

## INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

**The quality of this reproduction is dependent upon the quality of the copy submitted.** Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps. Each original is also photographed in one exposure and is included in reduced form at the back of the book.

Photographs included in the original manuscript have been reproduced xerographically in this copy. Higher quality 6" x 9" black and white photographic prints are available for any photographs or illustrations appearing in this copy for an additional charge. Contact UMI directly to order.



University Microfilms International  
A Bell & Howell Information Company  
300 North Zeeb Road, Ann Arbor, MI 48106-1346 USA  
313/761-4700 800/521-0600



**Order Number 9232167**

**Theory and applications of hyper-redundant robotic  
manipulators**

**Chirikjian, Gregory Scott, Ph.D.**

**California Institute of Technology, 1992**

**Copyright ©1992 by Chirikjian, Gregory Scott. All rights reserved.**

**U·M·I**  
300 N. Zeeb Rd.  
Ann Arbor, MI 48106



**Theory and Applications  
of  
Hyper-Redundant Robotic Manipulators**

Thesis by  
**Gregory S. Chirikjian**

In Partial Fulfillment of the Requirements  
for the Degree of  
Doctor of Philosophy

California Institute of Technology  
Pasadena, California  
1992  
(Defended 22 May 1992)

©1992

Gregory S. Chirikjian  
All rights Reserved

## Abstract

*The term “hyper-redundant” refers to robotic manipulators and mobile robots with a very large, possibly infinite, number of actuatable degrees of freedom. These robots are analogous in morphology and operation to snakes, worms, elephant trunks, and tentacles. This thesis presents a novel kinematic framework for hyper-redundant manipulator motion planning and task implementation. The basis of this formulation is the use of a “backbone reference set” which captures the essential macroscopic geometric features of hyper-redundant robots. In the analytical part of this work, the backbone representation is developed and used to solve problems in obstacle avoidance, locomotion, grasping, and “optimal” end effector placement. The latter part of this thesis deals with the design and implementation of a thirty-degree-of-freedom planar hyper-redundant manipulator which is used to demonstrate these novel kinematic and motion planning techniques. Design issues such as robustness with respect to mechanical failure, and design for easy assembly and repair are also addressed. The analytical and design concepts are combined to illustrate tasks for which hyper-redundant robotic mechanisms are well suited.*

## Acknowledgements

I owe a great deal to my advisor, Joel Burdick, for his constant support and friendship. Joel has provided an excellent environment in which to work. Without Joel's open mind and flexibility, this work may never have matured. I would like to thank the other members of my committee: Richard Murray, James Knowles, Erik Antonsson, and Samad Hayati for taking the time to familiarize themselves with my thesis, and for their constructive comments.

I thank Joe Puett for his constant and enthusiastic help with everything from soldering circuit connections to first rate video production. Many thanks go to Howie Choset and Jim Ostrowski, without whom the computational environment needed to perform the experimental part of this work would not exist. I thank Dave Koenig for setting up a graphical environment which was necessary for initial simulations. I owe my thanks to Brian Rague for designing circuitry and his willingness to investigate and solve any electronics problem at a moment's notice. I thank Marty Gould for having the patience to teach me some of the subtleties of working in a machine shop, and for his input into the design of components for the robot. I also would like to thank Rodney Rojas for machining essential components.

A small army of undergraduate students each added to the development of the robot in their own way. Charles Cook helped with the initial testing of actuator

control. He also did quite a bit of machining during the "brainstorming" phase of the robot development. Bill Greene designed and machined one version of a bearing fixture. Kevin Park, Emerson Tongco, and Michelle Tseng designed and machined the twenty ratcheting wheel fixtures which are used for robot locomotion.

I thank my friends for all the things that friends do. I thank Bob M'Closkey and I-Ming Chen for putting up with my mess. I would also like to thank Mark Lusk, Kayo Ide, and Atsuko Fukui for their advice and support over the past several years. Most of all, I thank my parents for giving me the strength to pursue my dreams.

# Contents

<b>1</b>	<b>Introduction and Motivation</b>	<b>1</b>
1.1	Past Design Efforts . . . . .	3
1.2	Hyper-Redundancy in Biology . . . . .	7
1.3	Kinematic Redundancy Resolution . . . . .	9
1.4	Previous Hyper-Redundant Robot Motion Planning . . . . .	13
1.5	Organization of this Work . . . . .	16
<b>KINEMATICS</b>		
<b>2</b>	<b>Parametrization of Curves and Spherical Motion</b>	<b>20</b>
2.1	Parametrization of Time Varying Curves . . . . .	21
2.1.1	Classical Parametrization of Curves . . . . .	22
2.1.2	Alternate Parametrization of Backbone Curves . . . . .	26
2.2	Parametrization of Spherical Motion . . . . .	28
2.2.1	An Euler-Angle Representation . . . . .	28
2.2.2	Matrix Exponential Representation . . . . .	31
2.2.3	Parametrization via Unit Quaternions . . . . .	32
2.3	Parametrization of Planar Curves . . . . .	33
2.4	Discussion . . . . .	35
<b>3</b>	<b>Fitting Manipulators to Backbone Reference Sets</b>	<b>36</b>
3.1	Inverse Kinematics in Parallel . . . . .	37
3.2	A Planar Serial Chain Manipulator . . . . .	40
3.3	A Manipulator with Closed Loop Modules . . . . .	42

3.4	A Planar Truss Manipulator . . . . .	43
3.5	A Spatial VGTM . . . . .	45
3.6	Morphology-Specific Fitting Procedures . . . . .	47
3.7	Discussion . . . . .	48
<b>TASK ANALYSIS AND SIMULATION</b>		
<b>4</b>	<b>A Modal Approach to Hyper-Redundancy Resolution</b>	<b>50</b>
4.1	Motivation for Choice of Parametrization . . . . .	51
4.2	End-Effector Placement Using Bending Modes . . . . .	53
4.3	End-Effector Placement Using Extension Modes . . . . .	58
4.4	Combined Bending and Extension Modes . . . . .	60
4.5	Spatial Modes . . . . .	62
4.6	Algorithms for Modal Inverse Kinematics . . . . .	67
4.7	Singularities, Degeneracy, and Switching . . . . .	70
4.7.1	Modal Singularities . . . . .	70
4.7.2	Singularities of Bending Modes . . . . .	70
4.7.3	Singularities of Extension Modes . . . . .	72
4.7.4	Degenerate Modes . . . . .	72
4.7.5	Switching Among Sets of Modes . . . . .	74
4.8	Discussion . . . . .	75
<b>5</b>	<b>Optimal Manipulator Configurations</b>	<b>76</b>
5.1	Defining Configurations of Least Bending . . . . .	77
5.2	Defining Optimal Extensible Configurations . . . . .	78
5.3	Optimal Shape Equations for the Planar Case . . . . .	80
5.4	Equations for the Spatial Nonextensible Case . . . . .	83
5.4.1	Parametrization via $K$ , $T$ , and $R$ . . . . .	84
5.4.2	Parametrization via Unit Quaternions . . . . .	85
5.5	Equations for the Spatial Extensible Case . . . . .	86
5.6	Optimal Backbone Reparametrization . . . . .	88
5.6.1	Optimal Planar Curve Reparametrization . . . . .	88

5.6.2	Optimal Roll Distribution (Nonextensible Case) . . . . .	91
5.6.3	Optimal Spatial Curve Reparametrization . . . . .	93
5.6.4	Optimal Roll and Length Distributions . . . . .	94
5.7	Comparisons with the Modal Approach . . . . .	96
5.8	Discussion . . . . .	99
<b>6</b>	<b>Hyper-Redundant Obstacle Avoidance</b>	<b>101</b>
6.1	Nonextensible Backbone Tunnel Constraints . . . . .	103
6.1.1	Planar Tunnel Constraints . . . . .	104
6.1.2	Spatial Tunnel Constraints . . . . .	106
6.1.3	A Quasi-Modal Form for Exterior Segment Constraints	108
6.2	Extensible Manipulator Obstacle Avoidance . . . . .	109
6.3	A Specific Example . . . . .	110
6.4	Operator Specification of Obstacle Avoiding Paths . . . . .	116
6.5	Discussion . . . . .	119
<b>7</b>	<b>Hyper-Redundant Locomotion and Grasping</b>	<b>120</b>
7.1	Classification of Nonextensible Locomotion . . . . .	121
7.1.1	Stationary-Wave-Amplitude-Varying Locomotion . .	122
7.1.2	Traveling-Wave-Amplitude-Constant Locomotion . .	125
7.2	Classification of Extensible Locomotion . . . . .	127
7.3	Locomotion over Uneven Terrain . . . . .	129
7.3.1	Nonextensible Locomotion over Curved Terrain . .	129
7.3.2	Extensible Locomotion over Curved Terrain . . . .	132
7.4	Extrinsic Formulation for Extensible Mechanisms . . . . .	133
7.5	Applications to Grasping . . . . .	139
7.5.1	A Continuous Nonextensible Example . . . . .	141
7.5.2	A Nonsmooth Pure Bending Wave Example . . . .	144
7.5.3	Extensible Wave Examples . . . . .	145
7.5.4	Dual Arm Grasping . . . . .	147
7.6	Gaits Derived from Linkage Kinematics . . . . .	148

7.6.1	A Four-Bar Linkage Gait	150
7.6.2	A Five-Bar Linkage Gait	151
7.7	Discussion	152
<b>HARDWARE DEVELOPMENT AND DEMONSTRATION</b>		
<b>8</b>	<b>A Hardware Implementation</b>	<b>155</b>
8.1	Task Requirements	156
8.2	Review of Candidate Morphologies	157
8.3	System Description	159
8.3.1	Manipulator Description	159
8.3.2	Real-Time Control System	163
8.4	Lessons Learned the "Hard"ware Way	164
8.5	Discussion	165
<b>9</b>	<b>Demonstrations of Task Implementation</b>	<b>167</b>
9.1	Mechanism Extension/Bending Limits	168
9.2	Obstacle Avoidance	171
9.3	Locomotion	173
9.4	Grasping a Cylinder	176
9.5	Discussion	178
<b>10</b>	<b>Conclusions</b>	<b>179</b>
<b>Appendix</b>		<b>181</b>
<b>References</b>		<b>185</b>

## List of Figures

1.1	Scripps Institute Tensor-Arm	3
1.2	Cascade of Flexible Micro Actuators	4
1.3	A Variable Geometry Truss Manipulator	5
1.4	Classification of Hyper-Redundant Robot Morphologies	6
1.5	Locomotion of a Slug	7
1.6	Primary Modes of Snake Locomotion	8
1.7	Frames Affixed to a Robot	9
2.1	Frenet Frames	23
2.2	Modified Frenet Frames	24
2.3	Physical Interpretation of $K(s,t)$ , $T(s,t)$ , and $l(s,t)$	29
2.4	Physical Interpretation of $\theta$ and $l$	34
3.1	Backbone Reference Set and Mechanism Modules	37
3.2	Five Sets of 3R Modules Affixed to a Backbone	41
3.3	Closed Loop Modular Design	43
3.4	A Planar Variable Geometry Truss Manipulator	44
3.5	A Highly Articulated Spatial VGTM	45
4.1	Continuous Bending Modes	54
4.2	Mixed Continuous/Discontinuous Bending Modes	56
4.3	Discontinuous Bending Modes	58
4.4	Pure Extension Modes	60
4.5	Bending and Extension Modes	61

4.6	Modes which Mimic a Cartesian Manipulator . . . . .	66
4.7	Modes which Mimic a SCARA Manipulator . . . . .	67
4.8	Modes Without Closed-Form Inverse Kinematics . . . . .	69
4.9	An Example of Bending Mode Singularities . . . . .	71
4.10	Covering the Workspace . . . . .	74
5.1	Uniformly Weighted Planar Nonextensible Configurations . . . . .	81
5.2	Nonuniformly Weighted Planar Nonextensible Configurations . . . . .	82
5.3	Uniformly Weighted Planar Extensible Configurations . . . . .	83
5.4	Optimal Nonextensible Spatial Configurations . . . . .	86
5.5	Optimal Extensible Spatial Configurations . . . . .	87
5.6	Optimal Reparametrization of Planar Curves . . . . .	91
5.7	Specified Backbone Curve with Optimal Roll Distribution . . . . .	93
5.8	Optimal Roll and Extensibility for a Given Backbone Reference Set	95
5.9	Configurations Defined by Optimal Modes . . . . .	99
6.1	A Backbone Curve Constrained to a System of Tunnels . . . . .	103
6.2	Dimensions of the Obstacle Field . . . . .	113
6.3	An Obstacle Avoiding Backbone Curve . . . . .	113
6.4	Obstacle-Avoiding Configurations for Moving Obstacles . . . . .	114
6.5	Optimally Reparametrized Obstacle-Avoiding Configurations . . . . .	115
6.6	Avoiding Obstacles with Purely Longitudinal Motion . . . . .	116
6.7	An Arbitrary Obstacle Field with User Defined Interior Segment . .	117
6.8	Pure Bending Obstacle Avoidance . . . . .	118
6.9	Purely Longitudinal Obstacle Avoidance . . . . .	119
7.1	Nonextensible SWAV Locomotion: Example 1 . . . . .	123
7.2	Nonextensible SWAV Locomotion: Example 2 . . . . .	125
7.3	Nonextensible TWAC Locomotion . . . . .	127
7.4	Extensible SWAV Locomotion . . . . .	127
7.5	Extensible TWAC Locomotion . . . . .	128
7.6	Pure Longitudinal Locomotion over a Wall . . . . .	133

## LIST OF FIGURES

xii

7.7	Parameters Describing Extrinsic Extensible Robot Locomotion . . . . .	134
7.8	Extrinsically Defined Stationary Wave Locomotion . . . . .	136
7.9	Extrinsically Defined Traveling Wave Locomotion . . . . .	137
7.10	Hyper-Redundant Grasping of a Cylinder . . . . .	141
7.11	Turning an Object Fixed to an Axle . . . . .	145
7.12	Compression Wave Grasping of a Cylinder . . . . .	146
7.13	Expansion Wave Grasping of a Cylinder . . . . .	147
7.14	Dual Arm Grasping of a Cylinder . . . . .	147
7.15	Serial Chain Robot Locomotion: Four Bar . . . . .	149
7.16	Serial Chain Robot Locomotion: Five Bar . . . . .	149
8.1	Schematic and Photograph of the Current Truss Structure . . . . .	160
8.2	Assembly-Disassembly Procedure . . . . .	161
8.3	Photograph of One Bay of the Truss . . . . .	162
8.4	System Architecture . . . . .	163
9.1	Manipulator Fully Extended . . . . .	168
9.2	Manipulator Partially Contracted . . . . .	169
9.3	Manipulator Fully Contracted . . . . .	169
9.4	Partially Curled Manipulator . . . . .	170
9.5	Fully Curled Manipulator . . . . .	170
9.6	Manipulator Entering Obstacle Field . . . . .	171
9.7	Manipulator Partially in Obstacle Field . . . . .	171
9.8	Manipulator Mostly in Obstacle Field . . . . .	172
9.9	Manipulator Fully in Obstacle Field . . . . .	172
9.10	Stationary Wave Locomotion - Pure Bending . . . . .	173
9.11	Stationary Wave Locomotion - Pure Extension . . . . .	174
9.12	Traveling Wave Locomotion - Pure Extension . . . . .	175
9.13	Single-Arm Satellite Recovery . . . . .	176
9.14	Dual-Arm Satellite Recovery . . . . .	177

# Chapter 1

## Introduction and Motivation

The word *redundant* is commonly used to mean “exceeding what is necessary or normal” [Webs83]. Robotic manipulators which have more independent degrees of freedom than are required to perform a specified task are called *kinematically redundant*. A robot can also be referred to as having *redundant actuation* if more than the required number of actuators are available to insure task implementation despite possible actuator failure. A commonly specified task for robotic manipulators is the positioning and orientation of an object at the end-effector, or hand, relative to the base. The benefits of redundancy with regard to this task are enumerated in the literature [Bail86,Bur88,DaSS88,KirV86,NakHY87]. The term *hyper-redundant* is defined here to describe robotic mechanisms with a relative degree of kinematic and/or actuator redundancy which is large or infinite. Tasks which can be performed by nonredundant or mildly redundant robotic mechanisms will be referred to with the adjective hyper-redundant when performed by hyper-redundant mechanisms, e.g., hyper-redundant manipulation, hyper-redundant locomotion, etc.

While a wide variety of hyper-redundant mechanism morphologies can be imagined, this work focuses on manipulators which have a macroscopic serial structure, e.g., those which approximate the shape of snakes, elephants’ trunks, or tentacles.

Implementations of this class of hyper-redundant manipulators may consist of truly flexible physical structures, a cascade of modules composed of redundantly actuated parallel platforms, or a large number of rigid links in series.

As will be shown in the following chapters, applications of hyper-redundant manipulators include, but are not limited to, inspection in highly constrained environments, novel forms of robotic locomotion and grasping, and the manipulation of objects. While having a large number of redundant degrees of freedom enhances the ability of manipulators to perform these tasks, it also introduces new problems which will be addressed in the analytical part of this thesis.

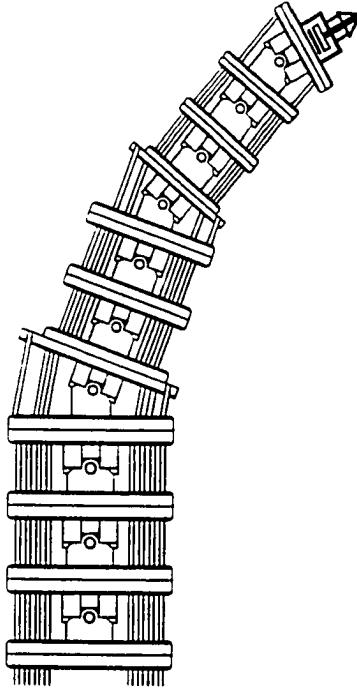
Methods by which redundancy can be used to augment task performance are referred to as *redundancy resolution*. The number of computations required by standard methods of redundancy resolution grows enormously with increasing redundancy. Thus, in this thesis, standard methods of redundancy resolution are rejected for hyper-redundant mechanisms, and a new approach is developed. In this new approach we assume that the geometric features of any hyper-redundant manipulator without macroscopic branches or closed loops can be captured with a continuum model. A “backbone reference set” consisting of a backbone curve and affixed set of reference frames is defined. A general framework for parametrizing hyper-redundant manipulator backbone reference sets in physically meaningful ways is developed. Tasks are implemented by specifying the geometry of the backbone, which in turn serves as the input to the kinematics of the manipulator.

The remainder of this chapter is organized as follows: Section 1.1 reviews previous efforts at the design and implementation of hyper-redundant mechanisms. Section 1.2 reviews how hyper-redundancy is used in biological systems. Section 1.3 examines the limitations of standard redundancy resolution techniques. Section 1.4 reviews previous efforts at hyper-redundant robot motion planning. Section 1.5 outlines the organization of the remaining chapters in this thesis.

## 1.1 Past Design Efforts

A plethora of hyper-redundant robot designs have been presented in the literature over the past quarter century. Hyper-redundant manipulator designs have varied widely in physical structure and actuation. As a result, no general kinematic techniques have been developed previously for this class of robots. This section examines many previously proposed hyper-redundant robot morphologies, while Chapters 2-7 develop a unifying analytical framework for the kinematics and motion planning of all hyper-redundant morphologies.

The Scripps Institute “tensor arm” [AndH67] was one of the first tendon driven hyper-redundant manipulators. This can be seen in Figure 1.1. Since then, other tendon driven robotic manipulators have been proposed [He85,Mo87], designed [BusFMZ91,MaH91,HiMa91], and built [HiKU83,TayLE83]. A tendon driven snakelike manipulator which is used in Sweden to paint auto chassis is the “Spine” robot [Dr84].

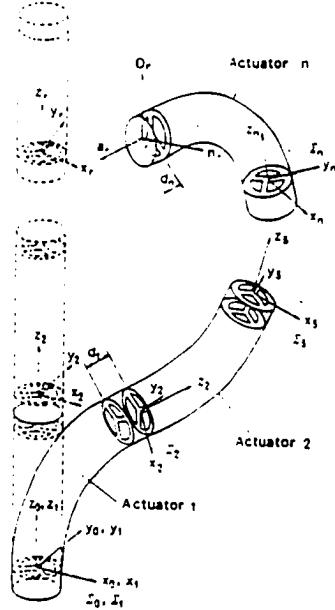


**Figure 1.1:** Scripps Institute Tensor-Arm (after [Sha87])

In [HiU76], an “active cord” mechanism was designed, built, and tested. This was

the first attempt at using a hyper-redundant robotic mechanism for the purpose of locomotion. Since then, numerous hyper-redundant morphologies have been developed. See [HiMo90, and references therein].

One of the first attempts at pneumatically actuated hyper-redundant arms was the Stanford “Orm.” More recently, [WilS88,WilM89] have studied pneumatic bellows in which variable air pressure provided a means of actuating flexible membranes to achieve motion. [FuHU89] investigated rubber gas actuator driven devices, in which electricity was used to initiate a chemical reaction to release gas from within the material structure of a rubber material instead of supplying air pressure. Figure 1.2 shows a cascade of flexible micro actuators developed in [SuIT91]. Each of three chambers in each actuator can be pressurized to implement bending in a variety of directions.

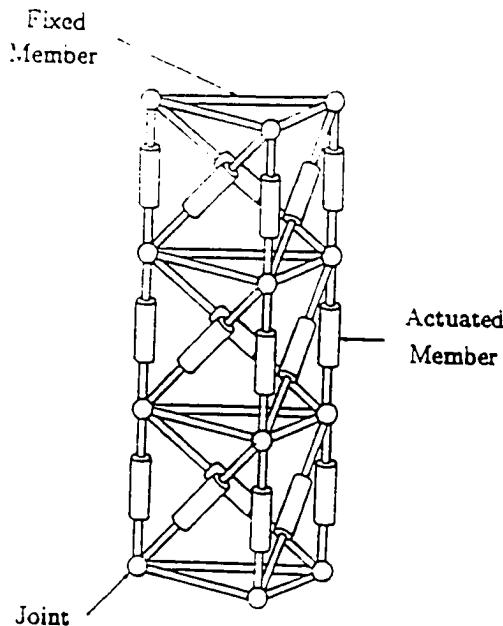


**Figure 1.2:** Cascade of Flexible Micro Actuators (after [SuIT91])

Serial chain morphologies consisting of a large number of rigid links have been studied in [Ch88,Sha87,Pe88]. In a sense, these are the canonical example of hyper-redundant manipulators, because of their simple kinematic structure and wide reference as examples of redundant manipulators in the literature [Bar-

rLL89,ScS88,HiMa89].

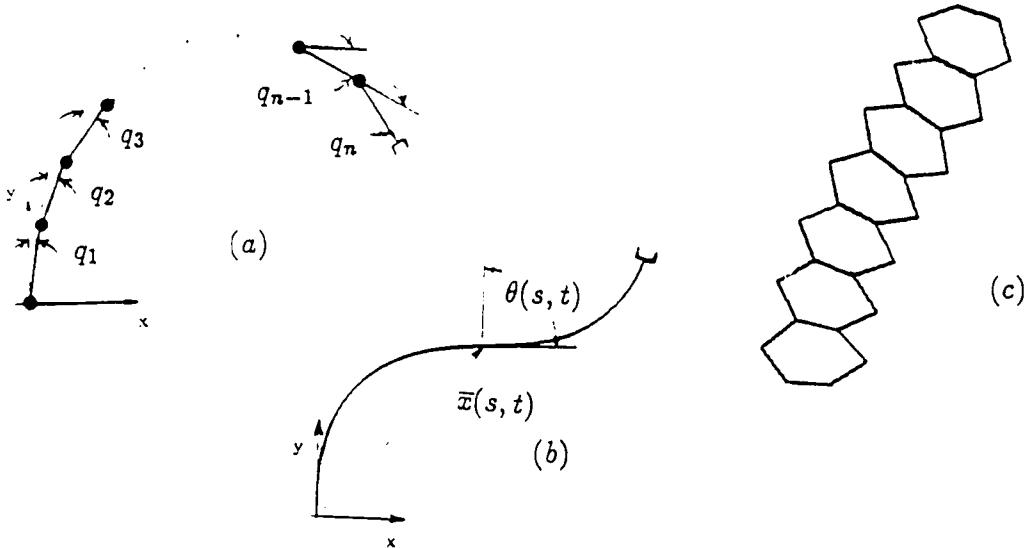
Another hyper-redundant morphology originated from research in large space structures [MiuF85,Wad90]. [NacH91,SaleRR88] have investigated the design and implementation of variable geometry truss manipulators (VGTMs). These usually consist of a serial chain of truss modules which are themselves parallel structures. This particular kind of structure is well suited for demonstrating both kinematic and actuator redundancy. Figure 1.3 shows the VGTM studied in [NacH89]. Each member is a linear actuator with the ability to radically change length. Design and implementation of a planar VGTM hyper-redundant manipulator is the subject of Chapter 8.



**Figure 1.3:** A Variable Geometry Truss Manipulator (after [NacH89])

Figure 1.4 classifies hyper-redundant morphologies into three categories. In Figure 1.4(a), a *discrete morphology* manipulator is shown, i.e., one with a large number of rigid links. This type of mechanical structure can be actuated with tendons, direct-drive motors, or gearhead motors. In Figure 1.4(b), a *continuous morphology* manipulator is shown, i.e., one whose actuation is distributed over the manipulator length. Such actuation could be achieved using shape memory alloy

[FuHK90], magnetic actuation [ShaKJ86], or pneumatic actuators. Figure 1.4(c) shows a cascade of modules which are actuated in parallel. Such modules could be the bays of a variable geometry truss, closed linkages with or without redundant actuation, or a combination of the two.



**Figure 1.4:** Classification of Hyper-Redundant Robot Morphologies

In addition to a wide variety of previous designs, “hyper-redundant” manipulators have been given a wide variety of names such as “swan’s neck” [HaPK90], “tentacle” [IB84], “highly redundant” [NacH89], “snakelike” [TeB89], “elephant trunk” [Mo87], “tensor arm” [AndH67], “Orm,” “flexible” [He85], “proboscis,” “massively redundant” [PeS89], “highly articulated,” “serpentine,” “Spine” [Dr84], and “active cord” [HiU76]. Applications of hyper-redundant robots include, but are not limited to, inspection in highly constrained environments [ChB89-2,CleI90], novel forms of locomotion and grasping [ChB89-3,PeS89,StuBDR90], and the fine manipulation of objects in space [ChB90-4]. As will be seen in Section 1.4, a variety of hyper-redundant robot motion planning algorithms for each of these morphologies have been attempted, but without any unified framework. But first we will exam-

ine how biological systems use hyper-redundancy, and examine standard methods of redundancy resolution.

## 1.2 Hyper-Redundancy in Biology

The biology literature is filled with examples of creatures with hyper-redundant morphologies which allow them to traverse land and water. Common creatures such as snakes, worms, and slugs have morphologies which can be considered hyper-redundant. A variety of small aquatic creatures have long narrow bodies which flex at high frequency for propulsion through fluid media [ClaH76,Web73]. Some parasites use hyper-redundant locomotion to embed themselves in the tissue of the host [GrL64]. Certain bacteria use hyper-redundant locomotion for propulsion by using flagellar bundles [WinK70]. Several works have been devoted to the movement of earthworms through soil [GrL38,Se69,Ya56]. Along similar lines the study of the locomotion of slugs has had a long history [Jo73,Vl07]. Lastly, a variety of papers have been devoted to the kinematics and kinetics of snake locomotion [BenMT74,Gr46,GrL50,Ja85,Ja86,Ja88,Li50].

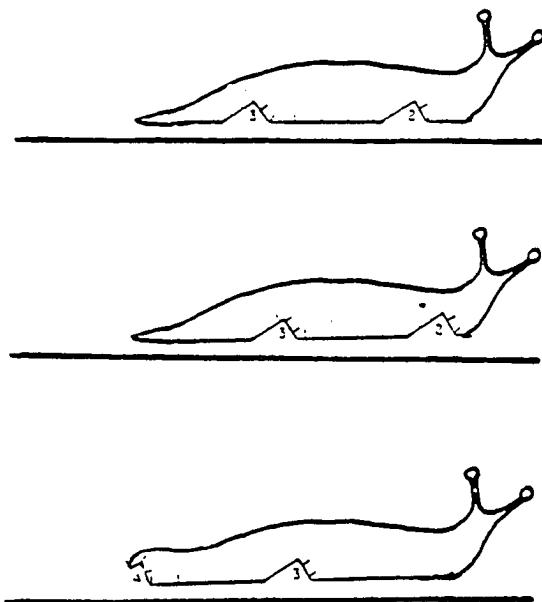
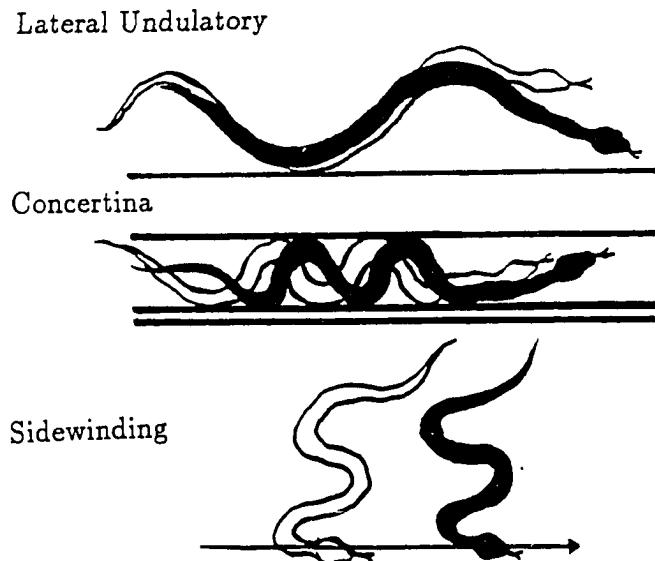


Figure 1.5: Locomotion of a Slug

Several terrestrial creatures such as inchworms, snakes, slugs, earthworms and caterpillars use forms of hyper-redundant locomotion. Figure 1.5 shows that several locomotory pulses, or *pedal waves*, travel along the length of the slug's body at one time [Ga62,Ga85]. Note that the pulses are not symmetric. This results in net forward motion due to a difference in propulsive force with the ground [Jo73].



**Figure 1.6:** Primary Modes of Snake Locomotion

Snake locomotion can be classified into three primary categories. These are lateral undulatory, sidewinding, and concertina modes, although snakes are known to use caterpillaring as well. The three primary modes are illustrated in Figure 1.6, as observed in [Ja86]. The choice of modes is highly situation dependent.

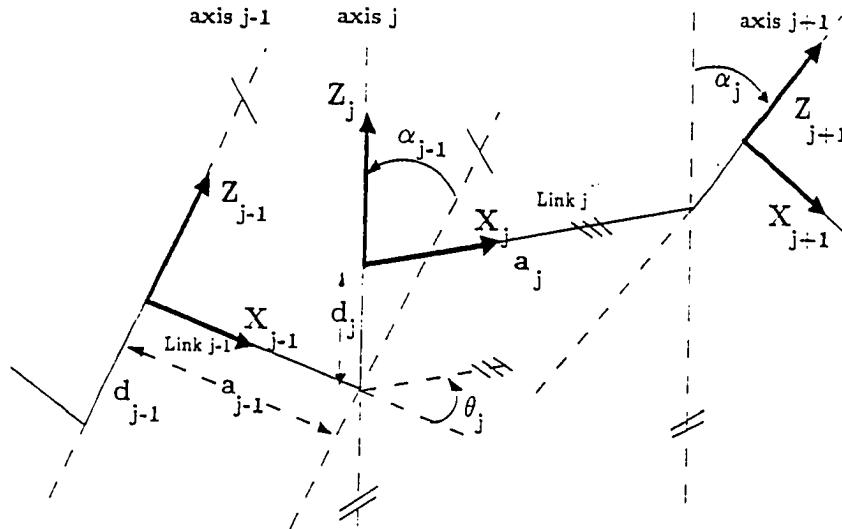
The focus of this section thus far has been creatures which have slender hyper-redundant bodies. However, other animals have limbs or tentacles which can be considered hyper-redundant. Examples are the octopus, squid, and elephant. The mechanics and morphological structure of such limbs have been considered in [Kie82,KieS85].

The methods of hyper-redundant locomotion found in nature provide examples which can be mimicked by hyper-redundant robots. Subsequent chapters will de-

velop a general kinematic framework for hyper-redundant robot task implementation. In particular, Chapter 7 will consider hyper-redundant robot locomotion. But first the next section reviews standard methods for resolving kinematic redundancy.

### 1.3 Kinematic Redundancy Resolution

The kinematic equations of robotic manipulators are usually formulated by affixing sets of frames to rigid elements in the mechanism. The relationship between these reference frames and the kinematic attributes of a manipulator, such as joint angles and link lengths, are generally expressed using the *Denavit-Hartenberg* notation [Crai86]. In this notation, homogeneous transforms are used to describe the relative rotation and translation of each reference frame with respect to adjacent frames.



**Figure 1.7:** Frames Affixed to a Robot (after [Bur88])

Figure 1.7 shows an example of this kind of description. The base frame, denoted by  $j = 0$ , provides an initial reference in space. Internal frames,  $j \in [1, n - 1]$ , describe the relative displacements of internal elements. The end-effector frame,

$j = n$ , specifies the position and orientation of the final link in the  $n$  degree of freedom serial-link robot structure. The information contained in the homogeneous transform relating the end-effector frame to the base frame is often expressed more compactly as a vector denoted  $\bar{x}_{ee}$ .

The *forward* or *direct kinematics* of serial chain robotic manipulators is generally written as:

$$\bar{x}_{ee} = \bar{f}(\bar{q}), \quad (1.3.1)$$

where  $\bar{q} \in \mathbf{R}^n$  is the vector of joint angles and  $\bar{x}_{ee} \in \mathbf{R}^m$  is the vector of end-effector positions and orientations. For the task of positioning and orienting the end-effector in space,  $m = 6$ . Equation (1.3.1) states that the forward kinematic function maps the joint angles of the manipulator to the vector describing the position and orientation of the end-effector. This can be written in differential form as:

$$\delta\bar{x}_{ee} = \mathbf{J}(\bar{q})\delta\bar{q} \quad (1.3.2)$$

where  $\mathbf{J}(\bar{q})$  is the Jacobian matrix of the transformation, which has elements  $J_{ij} = \frac{\partial f_i}{\partial q_j}$  where  $f_i$  and  $q_j$  are elements of the vectors  $\bar{f}(\cdot)$  and  $\bar{q}$  respectively. In the *inverse kinematics* problem, we are interested in determining the joint angles,  $\bar{q}$ , which will yield a desired end-effector position and orientation:

$$\bar{q} = \bar{f}^{-1}(\bar{x}_{ee}). \quad (1.3.3)$$

In the case of nonredundant manipulators, i.e., those for which

$$\dim(\bar{q}) = n = \dim(\bar{x}_{ee}) = m, \quad (1.3.4)$$

it has been shown that the inverse kinematics problem can always be solved in closed, or nearly closed, form [Pie68,DufC80,LeeL87]. However, it is nonetheless common to use a rate, or differential, formulation to solve for incremental displacements [Crai86]:

$$\delta\bar{q} = \mathbf{J}^{-1}(\bar{q})\delta\bar{x}_{ee}. \quad (1.3.5)$$

There are two limitations with this. First, if the Jacobian matrix is singular, i.e.,

$$\det(\mathbf{J}) = 0 \quad (1.3.6)$$

then there are problems inverting. This corresponds physically to an instantaneous loss of ability of the manipulator to move in one or more generalized directions (screw displacements).

Second, if the manipulator is redundant, i.e,

$$n = \dim(\bar{\mathbf{q}}) > \dim(\bar{\mathbf{x}}_{ee}) = m, \quad (1.3.7)$$

alternate methods have been developed, because the Jacobian matrix is no longer square. For redundant manipulators, two methods are generally used: the Moore-Penrose Pseudo-Inverse, and the Extended-Jacobian technique.

The Moore-Penrose generalized, or pseudo, inverse can be computed as:

$$\mathbf{J}^+ = \mathbf{W}^{-1} \mathbf{J}^T (\mathbf{J} \mathbf{W}^{-1} \mathbf{J}^T)^{-1} \quad (1.3.8)$$

when  $\mathbf{J}$  has full rank [RaoM71,KIH83]. For our applications  $\mathbf{J} \in \mathbf{R}^{m \times n}$ ,  $\mathbf{W} \in \mathbf{R}^{n \times n}$ , and  $\mathbf{J}^+ \in \mathbf{R}^{m \times m}$ . Note that  $\mathbf{J}^+ = \mathbf{J}^{-1}$  when  $n = m$ . In the case  $n > m$  most redundancy resolution in the literature (see [BarkM89,LuWP80,ScS87,ScS88,-SIY87]) is implemented with the equation:

$$\delta\bar{\mathbf{q}} = \mathbf{J}^+ \delta\bar{\mathbf{x}}_{ee} - (\mathbf{I}_n - \mathbf{J}^+ \mathbf{J}) \bar{\nabla} H \quad (1.3.9)$$

or similar formulations.  $\mathbf{I}_n$  is the identity matrix for  $\mathbf{R}^{n \times n}$ , and  $(\mathbf{I}_n - \mathbf{J}^+ \mathbf{J})$  is an operator which projects  $n$ -vectors onto the null space of the Jacobian matrix. The symmetric positive definite weighting matrix  $\mathbf{W}$  provides that the solution  $\delta\bar{\mathbf{q}}$  will instantaneously minimize the cost functional  $\delta\bar{\mathbf{q}}^T \mathbf{W} \delta\bar{\mathbf{q}}$  under the constraint that the incremental end-effector displacement  $\delta\bar{\mathbf{x}}_{ee}$  is adhered to.  $\bar{\nabla} H \in \mathbf{R}^n$  is the gradient of an additional scalar function  $H(\bar{\mathbf{q}})$  which we seek to minimize by projecting onto the null space of the Jacobian.

While this is the most popular method of redundancy resolution, there are two problems with this method. First, the algorithm is not necessarily *cyclic*, i.e., a closed-loop trajectory of the end-effector will not necessarily mean a closed-loop trajectory in the joint space of the robot. In other words, with this method, a manipulator end-effector can track a closed-loop trajectory, but the configuration of the robot will not return to its initial configuration. This can be quite a problem when dealing with robots with many degrees of freedom, because the manipulator can “get tangled.” However, recent work partially alleviates this problem [ShaY88,Bay92,RobM92]. The second major problem with this method is that it is computationally inefficient. The computational dependence on the number of manipulator degrees of freedom for this method is at least  $O(n^2)$ , and can be greater depending on the choice of  $H(\bar{q})$ . This becomes computationally cumbersome as  $n$  approaches infinity.

Another method of redundancy resolution is the use of an extended Jacobian. There are two versions of this method. In the simpler version, additional conditions of the form:

$$\bar{c} = \bar{g}(\bar{q}) \quad \text{or} \quad \delta\bar{c} = G(\bar{q})\delta\bar{q} \quad (1.3.10)$$

are specified where  $\bar{c}, \bar{g}(\cdot) \in \mathbb{R}^{n-m}$ , so that the matrix of an augmented system of equations becomes square:

$$\begin{bmatrix} \delta\bar{x}_{ee} \\ \delta\bar{c} \end{bmatrix} = \begin{bmatrix} J \\ G \end{bmatrix} \delta\bar{q} \quad \text{and so} \quad \delta\bar{q} = \begin{bmatrix} J \\ G \end{bmatrix}^{-1} \begin{bmatrix} \delta\bar{x}_{ee} \\ \delta\bar{c} \end{bmatrix} \quad (1.3.11)$$

One particular choice of the vector functions  $\bar{g}$  and  $\bar{c}$  is

$$\bar{g}(\bar{q}) = (I_n - J^+J)\bar{\nabla}H \quad (1.3.12)$$

and

$$\bar{c} = \bar{0} \quad (1.3.13)$$

which optimize the function  $H(\bar{q})$  by projecting the gradient on the null-space of the Jacobian, just as in the pseudo-inverse case. However, there are two major

differences. First, there is greater opportunity for algorithmic singularities to be encountered, i.e.,  $\mathbf{G}$  or  $\mathbf{J}$  may not always be full rank, or the combination of  $\mathbf{J}$  and  $\mathbf{G}$  may not be full rank.

The computational requirements for the extended Jacobian technique are also unacceptable for hyper-redundant robots. Inverting the matrix in Equation (1.3.11) requires  $O(n^3)$  arithmetic operations. If the augmented system of equations is based on Equation (1.3.12), additional pseudo-inverse computations must be performed as well.

While the number of computations required to implement Equations (1.3.9) and (1.3.11) involves the explicit inversion of matrices, it has recently been proposed to use the theory of sparse matrices to formulate iterative algorithms for the solution of inverse kinematic and dynamic equations [Alv91]. This method could potentially reduce the number of computations to  $O(n)$  for large  $n$ .

The methods developed in this thesis will also involve computations of  $O(n)$ , but are fully parallelizable, so that computation *time* can be  $O(1)$ , i.e, independent of the number of robot degrees of freedom.

## 1.4 Previous Hyper-Redundant Robot Motion Planning

As we saw in the previous section, standard kinematic redundancy resolution techniques are not desirable because they require the computation of a Jacobian pseudo-inverse or inversion of an augmented Jacobian. To circumvent the problems inherent in standard redundancy resolution techniques, some authors have developed motion planning techniques for particular hyper-redundant manipulator morphologies.

For instance, special motion planning algorithms have been developed for Variable Geometry Truss Manipulators (VGTMs) [NacH89,NacH91,SaleRR88]. In

those works, continuous spline-like curves were used to specify the macroscopic truss geometry. In [NacH89,NacH91] a resolved rate formulation was used to determine the kinematics of each module, or bay, of the truss in combination with a spline-like curve used to specify overall configuration. This is not desirable because use of pseudo-inverses in resolved rate formulations can result in non-cyclic behavior. While previous analyses were useful in demonstrating the use of VGTMs for obstacle avoidance and end-effector placement, past work has several drawbacks. First, for spatial manipulators, a curve alone is not sufficient to describe manipulator configuration. Second, unless the curve used to describe the manipulator is parametrized with meaningful physical variables, additional computations are required to specify manipulator shape and extensibility restrictions. Lastly, while [NacH89,NacH91,SaleRR88] deal exclusively with VGTMs, it is not clear how they would apply to other types of hyper-redundant manipulators.

Several authors have developed motion planning algorithms for hyper-redundant manipulators based on forces required to actuate the device. For instance [He85] uses tendon forces to determine acceptable motions of a “flexible” manipulator. [HiMo90] augments hyper-redundant mobile robot trajectories based on force feedback. [YoKT91] resolves redundancy in a variable geometry truss by minimizing forces with a Jacobian pseudo-inverse.

A continuum mechanics approach to the motion planning of hyper-redundant manipulators with distributed pneumatic actuation was introduced in [WilSn88,WilM89]. In [BrocS91], a similar analysis was used for the optimal shape design of long, thin elastic rods for implementing desired compliant robot behavior. In another recent work, [TavD90] defined an optimal shape synthesis problem for high degree of freedom variable geometry trusses using a differential geometric approach. The motion planning of a continuous curve manipulator based on intrinsic geometry is considered in [HaPK90]. Previous continuum approaches will be examined in greater detail when they are compared to the more general formulations in Chapters 2-5.

The above discussion provides a sample list of the various hyper-redundancy resolution techniques which have been developed for specific hyper-redundant morphologies. This list is however constantly expanding. For instance, neural network based approaches have recently been proposed for redundancy and hyper-redundancy resolution [At89,BarhGZ88,TanH91].

While previous hyper-redundant robot analyses have dealt with end-effector placement and obstacle avoidance, analytical treatment of problems such as hyper-redundant locomotion is rare. In fact, the hyper-redundant locomotion literature has focused primarily on mechanical and system design, rather than analysis, e.g., [HiMo90,StuBDR90]. This contrasts to the extensive body of literature devoted to the analysis of static walking [Ku87,McGh70,McGhP72,SoW89], dynamically stable walking and hopping [VakB90,Li-MR89,McGe90], and the more conventional wheeled vehicles [AM89,MuS90]. While a great quantity of work has been devoted to the kinematics and dynamics of legged locomotion, and sophisticated actuator technologies are being developed which can be used for hyper-redundant robots, little analysis has been performed for hyper-redundant mobile robots. In Chapter 7, this thesis contributes to the analysis and classification of the kinematics of hyper-redundant locomotion. While some of the mobile hyper-redundant robots in the literature are a hybrid between a snake-like vehicle and a tracked vehicle [HiMo90], the present investigation will be confined to motions caused strictly by the internally induced bending or twisting of the robot without the aid of actuatable wheels, tracks, or legs.

Several aspects of the general kinematic and motion planning framework presented in this thesis have been presented previously by the author in the papers [ChB89-1,2,3,4,ChB90-1,2,3, ChB91-1,2,3,4,ChB92]. The basic approach in the aforementioned papers and this thesis is the use of a “backbone curve” and associated set of reference frames which have a physically meaningful parametrization. The backbone curve is used to capture the macroscopic shape of the manipulator, while the associated frames specify twisting of the physical device about the imag-

inary backbone. Hyper-redundancy resolution is achieved by restricting backbone curve geometry and roll about the backbone curve.

In [ChB89-1], the word “hyper-redundant” was first used in reference to manipulators with a large number of actuatable degrees of freedom. In [ChB90-1], methods to analyze the kinematics and motion planning of planar *nonextensible* (fixed length) hyper-redundant manipulators were presented. In [ChB89-1,ChB90-2] modal expansions of the intrinsic curve parameters are used to resolve redundancy for the end-effector positioning task. In [ChB90-2,3], the methods in [ChB90-1] were generalized to include the spatial end-effector placement and obstacle avoidance problems. [ChB91-1,3] extends the work further to include manipulator *extensibility*, i.e., actuatable degrees of freedom in the longitudinal direction. Optimal hyper-redundant manipulator configurations were defined in [ChB92]. Applications of hyper-redundant manipulators such as grasping and locomotion were examined in [ChB89-3,ChB91-2,ChB91-4]. In [ChB91-3] a combined modal and traveling wave approach is used for obstacle avoidance with moving obstacles.

The following section explains the organization of the following chapters in this thesis.

## 1.5 Organization of this Work

The remaining chapters of this thesis are partitioned into three main areas: Kinematics - Chapters 2-3; Task analysis and simulation - Chapters 4-7; Hardware development and demonstration - Chapters 8-9.

Chapter 2 reviews and extends classical descriptions of the geometry of curves. A backbone reference set is defined as a backbone curve and associated set of frames. The backbone reference set is used to capture the macroscopic geometric features of hyper-redundant manipulators. Several physically meaningful parametrizations of the backbone reference set are examined.

Chapter 3 shows how the physical structure of hyper-redundant manipulators can be mapped to a backbone reference set. Hyper-redundant manipulator kinematics is analyzed by partitioning the manipulator into nonredundant segments, each of which has closed form inverse kinematic solutions. The kinematic constraints for each segment are specified independently by the backbone reference set. As a result, a parallel algorithm is developed.

Chapter 4 introduces the “modal approach” to hyper-redundancy resolution. The shape of the backbone curve is restricted by constraining the geometric functions developed in Chapter 2 to a sum of weighted mode functions. Criteria which the mode functions must satisfy and methods for continuously transforming the backbone curve from one set of modes to another are established. Singularities introduced with this method are also discussed.

Chapter 5 develops methods for determining the “optimal” configuration of a hyper-redundant manipulator which satisfies task constraints and minimizes a user-defined optimality criterion. The calculus of variations is used to develop differential equations, whose solution is the optimal backbone curve shape. Also considered are two other related optimization problems: the optimal reparametrization of backbone reference sets and the determination of optimal roll distributions about a spatial backbone curve.

Chapter 6 presents a strictly geometric algorithm for hyper-redundant manipulator obstacle avoidance which relies on the use of “tunnels” in the obstacle-filled workspace. Methods of the differential geometry of curves are used to formulate a generalized “follow the leader” approach which guarantees that sections of the manipulator are confined to the tunnels, and therefore avoid obstacles. A general formulation is presented with examples to illustrate this approach.

Chapter 7 explains how hyper-redundant robots can make use of their large number of degrees of freedom in the form of “waves” to manipulate, locomote, and grasp.

Chapter 8 gives a detailed description of the 30 degree of freedom variable geometry truss which was built to demonstrate the usefulness of hyper-redundancy. The mechanical design and assembly, as well as the electronic and computational elements, are discussed.

Chapter 9 presents the results of experiments performed to demonstrate and validate the analytical development of this thesis. Experiments in obstacle avoidance through a cluttered workspace, locomotion over flat terrain, and grasping of objects are demonstrated.

Chapter 10 is the conclusion. This contains a discussion of future applications of hyper-redundancy, and further problems which must be resolved.

## KINEMATICS

## Chapter 2.

# Parametrization of Curves and Spherical Motion

This chapter reviews and extends classical methods for describing curves and associated sets of reference frames in  $\mathbf{R}^3$ . Commonly used techniques from spherical kinematics and continuum mechanics are incorporated to establish a general framework for defining and parametrizing a *backbone reference set*. The backbone reference set consists of a backbone curve and a set of reference frames which evolve along the curve. Parametrizations of the backbone reference set are developed in a “natural” way for specifying hyper-redundant manipulator configurations. The backbone reference set is used in Chapter 3 to provide kinematic inputs for actual hyper-redundant robotic mechanisms.

The organization of this chapter is as follows: Section 2.1 reviews classical methods for parametrizing curves. The theory of curves and spherical motions is then combined with the fundamental idea of a *referential description* from continuum mechanics. The result is a general method for parametrizing curves and sets of frames, which together define the backbone reference set. Section 2.2 reviews a variety of specific parametrizations of spherical motion which are useful in this context. These parametrizations include Euler-Angles, unit quaternions, and matrix exponentials. Section 2.3 considers the degenerate case of planar curves.

## 2.1 Parametrization of Time Varying Curves

Time varying curves in  $\mathbf{R}^3$  can be represented in the parametric form:

$$\bar{x} = \bar{x}(s, t) \quad (2.1.1)$$

where  $s$  is called the *curve parameter*, which specifies a particular point in space at time  $t$ . Throughout this work, the curve parameter is normalized to the closed unit interval  $s \in [0, 1]$  because we are dealing strictly with curve segments of finite length. Much of the computer science literature concerning the parametrization of curves in  $\mathbf{R}^3$  [Fa90,SuL89,RogA76] is based on the specific *algebraic* structure of the vector valued function  $\bar{x}(s, t)$ . Throughout this work parametrizations which are purely algebraic in nature are referred to as *direct parametrizations*.

A well known direct parametrization from the computer graphics literature [SuL89] is the polynomial spline :

$$\bar{x}(s, t) = \sum_{i=1}^N s^i \bar{v}_i(t) \quad (2.1.2)$$

where  $\{\bar{v}_i(t)\}$  is a set of time varying vectors. Another example is the class of spline-like curves which have been used in [RogA76], and which have the form:

$$\bar{x}(s, t) = \sum_{i=1}^N w_i(s) \bar{v}_i(t) \quad (2.1.3)$$

where the weights  $\{w_i\}$  form a partition of unity:

$$\sum_{i=1}^N w_i(s) = 1. \quad (2.1.4)$$

For more information on other common curves, see [Kr79,MilP77,Str61].

The following sections will introduce *indirect parametrizations* in which integrals or differential equations must be solved in order to explicitly represent  $\bar{x}(s, t)$ . We resort to indirect parametrizations because direct parametrizations do not explicitly provide important geometric features of curves. These features will play a key role in Chapter 3 when actual mechanisms are considered.

For instance, Equations (2.1.2) or (2.1.3-4) do not explicitly state the length of the curve between two arbitrary points  $s = 0$  and  $s = \sigma$ . An additional computation of the form

$$L(\sigma, t) = \int_0^\sigma (\dot{\bar{x}} \cdot \dot{\bar{x}})^{\frac{1}{2}} ds \quad (2.1.5)$$

would be required to find the arc length, where a  $\cdot$  represents differentiation with respect to  $s$ . Thus, length constraints on a time varying curve can become complicated when using direct parametrizations. Other physically meaningful quantities such as how a curve bends and twists are also not explicit in direct parametrizations, and must be computed with additional equations. These issues are addressed in the following subsection.

### 2.1.1 Classical Parametrization of Curves

In the classical differential geometry of curves, curves are parametrized by arc-length:  $L(s, t) = s$ , which means that  $\dot{\bar{x}} \cdot \dot{\bar{x}} = 1$ . The *Frenet-Serret Apparatus* [MilP77,Str61] can be used to define how a curve locally bends and twists. The Frenet-Serret apparatus also defines a unique set of frames at each point on the curve. For each  $s$ , there is an associated frame consisting of the three orthonormal vectors:  $\bar{u}(s, t)$ ,  $\bar{n}(s, t)$ , and  $\bar{b}(s, t)$ , which are respectively termed the tangent, normal, and binormal vectors (see Figure 2.1):

$$\bar{u} = \dot{\bar{x}}. \quad (2.1.6)$$

$$\bar{n} = \frac{1}{\kappa} \dot{\bar{u}} \quad (2.1.7)$$

$$\bar{b} = \bar{u} \times \bar{n}. \quad (2.1.8)$$

where

$$\kappa(s, t) = (\dot{\bar{u}}(s, t) \cdot \dot{\bar{u}}(s, t))^{\frac{1}{2}} \quad (2.1.9)$$

is called the *curvature function*. Note that for nonzero curvature,  $\bar{u} \cdot \bar{u} = 1$  implies  $\bar{u} \cdot \bar{n} = 0$ , and for zero curvature the definitions no longer hold.

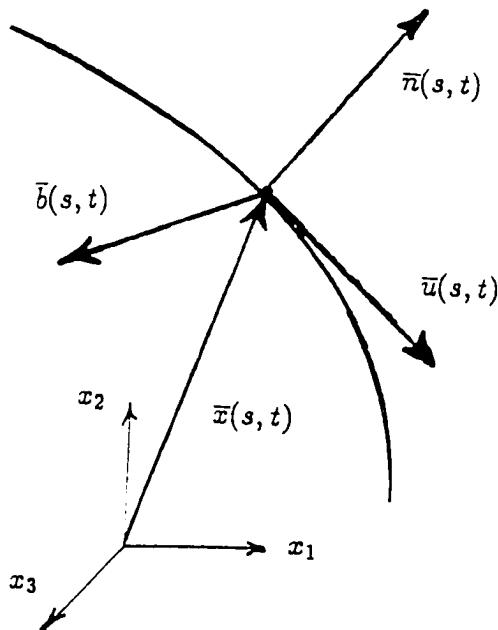


Figure 2.1: Frenet Frames

From the above definitions, the following relationships can be derived:

$$\dot{\bar{u}} = \kappa \bar{n}; \quad \dot{\bar{n}} = \tau \bar{b} - \kappa \bar{u}; \quad \dot{\bar{b}} = -\tau \bar{n}, \quad (2.1.10)$$

where  $\tau(s, t)$  is an additional intrinsic parameter of the curve, termed *torsion*:

$$\tau = \frac{1}{\kappa^2} \bar{u} \cdot (\dot{\bar{u}} \times \ddot{\bar{u}}). \quad (2.1.11)$$

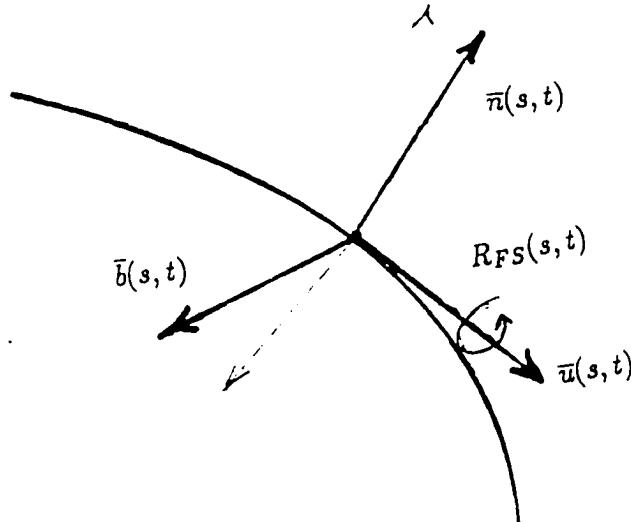
$\kappa(s, t)$  can be physically interpreted as the bending of the curve in the plane spanned by  $\bar{u}(s, t)$  and  $\bar{n}(s, t)$ , while  $\tau(s, t)$  measures the rate of change of orientation of that plane (whose normal is  $\bar{b}(s, t)$ ). Let  $\mathbf{Q}_{FS}(s, t) = [\bar{u}(s, t) \bar{n}(s, t) \bar{b}(s, t)]$  be the set of Frenet-Serret frames for a given curve. From Equation (2.1.10), it can be seen that the rate of change of  $\mathbf{Q}_{FS}$  is governed by the equation:

$$\dot{\mathbf{Q}}_{FS} = \mathbf{Q}_{FS} \boldsymbol{\Lambda}_{FS} \quad (2.1.12)$$

where

$$\boldsymbol{\Lambda}_{FS} = \begin{bmatrix} 0 & -\kappa & 0 \\ \kappa & 0 & -\tau \\ 0 & \tau & 0 \end{bmatrix} \quad (2.1.13)$$

The extension of this scheme to cases where  $s$  is no longer the arc length parameter is straightforward. If  $L(s, t) \neq s$ , then every  $d/ds$  operation performed in Equations (2.1.6-2.1.11) would be replaced with an operator of the form:  $1/(\dot{\bar{x}} \cdot \dot{\bar{x}})d/ds$ .



**Figure 2.2:** Modified Frenet Frames

An extension to the Frenet-Serret Apparatus which is relevant to hyper-redundant manipulator kinematics is to include a *relative roll distribution*,  $R_{FS}(s, t)$ , as seen in Figure 2.2.  $R_{FS}(s, t)$  is a measure of twist about the backbone curve tangent vector at each  $s$  measured in the Frenet Frames. While the macroscopic shape of hyper-redundant manipulators can be captured using a curve, incorporating the roll distribution allows the backbone reference set to specify how an actual mechanism twists about the curve. The *modified Frenet-Serret Frame*:

$$Q_M = \text{ROT}[\bar{v}, R_{FS}] Q_{FS} = Q_{FS} \text{ROT}[\bar{e}_1, R_{FS}] \quad (2.1.14)$$

can be used as a backbone reference set, where the symbol  $\text{ROT}[\bar{v}, \alpha]$  represents a counter-clockwise rotation about the vector  $\bar{v}$  by an angle  $\alpha$ .  $\bar{e}_i$  represents the  $i^{th}$  natural basis vector of  $\mathbf{R}^3$ . The rate of change of the modified Frenet Frame is then :

$$\dot{Q}_M = Q_M \Lambda_M \quad (2.1.15)$$

where

$$\Lambda_M = \begin{bmatrix} 0 & -\kappa \cos R_{FS} & \kappa \sin R_{FS} \\ \kappa \cos R_{FS} & 0 & -(\tau + \rho) \\ -\kappa \sin R_{FS} & \tau + \rho & 0 \end{bmatrix} \quad (2.1.16)$$

and  $\rho = \dot{R}_{FS}$  is the *rate of relative roll distribution* with respect to  $s$ . In Chapter 5, the norm of  $\Lambda_M$  will be used to define an *absolute roll distribution*.

It is possible to represent spatial curves directly in terms of the curvature and torsion. This can be achieved by manipulating Equations (2.1.6-11) to yield [Str61]:

$$\ddot{\bar{x}}^{(4)} - \left( 2\frac{\dot{\kappa}}{\kappa} + \frac{\dot{\tau}}{\tau} \right) \ddot{\bar{x}}^{(3)} + \left( \kappa^2 + \tau^2 - \frac{\kappa\ddot{\kappa} - 2(\dot{\kappa})^2}{\kappa^2} + \frac{\dot{\kappa}\dot{\tau}}{\kappa\tau} \right) \ddot{\bar{x}} + \kappa^2 \left( \frac{\dot{\kappa}}{\kappa} - \frac{\dot{\tau}}{\tau} \right) \dot{\bar{x}} = 0 \quad (2.1.17)$$

With specified curvature and torsion functions, and appropriate base conditions, such as

$$\bar{x}(0, t) = \bar{0}; \quad \dot{\bar{x}}(0, t) = \bar{e}_1$$

and

$$\dot{\bar{x}}(0, t) = \kappa(0, t)\bar{e}_2; \quad \ddot{\bar{x}}(0, t) = -\kappa^2(0, t)\bar{e}_1 + \dot{\kappa}(0, t)\bar{e}_2 + \kappa(0, t)\tau(0, t)\bar{e}_3,$$

Equation (2.1.17) can be integrated numerically to yield the backbone curve  $\bar{x}(s, t)$ . Similarly, a set of frames which smoothly evolve along the curve can be defined by integrating Equation (2.1.15) for a choice of  $\kappa, \tau, \rho$  with initial conditions such as  $\mathbf{Q}_M(0, t) = \mathbf{I}_3 = [\bar{e}_1, \bar{e}_2, \bar{e}_3]$ .

Equation (2.1.17) provides an indirect parametrization of  $\bar{x}(s, t)$  in terms of  $\kappa(s, t)$  and  $\tau(s, t)$ , and Equation (2.1.15) provides an indirect parametrization of an associated set of frames. While this is an elegant and physically meaningful indirect parametrization of curves, it has some drawbacks. For instance, when  $\kappa$  is close to zero, Equation (2.1.17) becomes singular. Because Equation (2.1.17) has no general closed form analytical solution, it would have to be solved numerically. These are substantial computational drawbacks when compared to direct parametrizations. In order to formulate a parametrization with physical meaning but without these drawbacks, the following sections introduce alternate parametrizations.

### 2.1.2 Alternate Parametrization of Backbone Curves

Both planar and spatial curves can be parametrized in the general form:

$$\bar{x}(s, t) = \int_0^s l(\sigma, t) \bar{u}(\sigma, t) d\sigma \quad (2.1.18)$$

where again  $s \in [0, 1]$  is the curve parameter which represents a particular point on the curve at time  $t$ .  $\bar{u}(s, t)$  is the unit tangent vector to the curve at  $s$  which, as a matter of convention, is taken here to be the *second* column of a rotation matrix:

$$\bar{u}(s, t) = Q(s, t) \bar{e}_2. \quad (2.1.19)$$

The columns of  $Q(s, t)$  define the axes of the set of backbone reference frames attached to the curve. The matrix  $Q(s, t)$  will sometimes be written in terms of its columns as

$$Q(s, t) = [\bar{\Psi}_1(s, t) \bar{\Psi}_2(s, t) \bar{\Psi}_3(s, t)], \quad (2.1.20)$$

where  $\bar{u}(s, t) = \bar{\Psi}_2(s, t)$ .

The general curve parametrization in Equation (2.1.18) has the following interpretation. The backbone curve is “grown” from the base by propagating the curve forward along the tangent vector, which is varying direction according to  $\bar{u}(s, t)$  and varying its magnitude (or “growth-rate”) according to  $l(s, t)$ .

$l(s, t)$  is termed the *rate of arc length* function because it is the rate of change of arc length with respect to the curve parameter. It controls the length of the curve tangent and can be written in the form:

$$l(s, t) = 1 + \epsilon(s, t) > 0 \quad (2.1.21)$$

where  $\epsilon(s, t)$  is the *rate of extensibility* of the curve at point  $s$  and time  $t$ . The length of the backbone curve between points  $s_1$  and  $s_2$  is simply  $L(s_2, t) - L(s_1, t)$ , where

$$L(s, t) = s + E(s, t) = \int_0^s l(\sigma, t) d\sigma. \quad (2.1.22)$$

$L(s, t)$  is the true, or classical, arc length measure (where  $\partial \bar{x}(L, t)/\partial L = 1$ ) and  $\partial L(s, t)/\partial s = \dot{L}(s, t) = l(s, t)$ .  $E(s, t)$  is the *extensibility* of the curve, which measures how the curve parameter deviates from arc length. In the nonextensible case,  $\dot{L} = 1$ ,  $E = 0$ , and  $s$  becomes the classical arc length parameter, as in Section 2.1.1. Thus, a positive value of  $\epsilon(s, t)$  indicates a local extension at  $s$ , while a negative value corresponds to a local contraction at  $s$ . Integrating all the local extensions and contractions of the curve over the interval  $[0, s]$  yields the extensibility of the curve segment over that interval. Because  $\epsilon(s, t)$  is a local property, some sections of an extensible curve may be extending while others are simultaneously contracting.

Let us assume that at time  $t = t_0$ :  $l(s, t_0) = 1$  (or equivalently  $L(s, t_0) = s$ ), and  $\bar{\Psi}_i(s, t_0) = \bar{e}_i$  for  $i = 1, 2, 3$ . These conditions dictate that the curve segment is straight, parametrized by arc length, and pointing along the  $x_2$ -axis at time  $t_0$ . In addition, all backbone reference frames differ only by translation along the  $x_2$ -axis. The geometry and parametrization of the backbone reference set at time  $t = t_0$  are referred to as the *nominal reference state*. Deviations from the nominal reference state are dictated by relative degrees of bending, twisting, rolling, and extensibility, which are all reflected in  $\{l(s, t), Q(s, t)\}$ .

The backbone reference set is completely specified by the homogeneous transformation matrix:

$$\mathcal{H}(s, t) = \begin{pmatrix} Q(s, t) & \bar{x}(s, t) \\ \bar{0}^T & 1 \end{pmatrix}. \quad (2.1.23)$$

Backbone reference set parametrizations can be induced by any parametrization of spherical motion. Common parametrizations of  $Q(s, t)$  from spherical kinematics are reviewed in the following section.

## 2.2 Parametrization of Spherical Motion

This section reviews parametrizations of spherical kinematics and their applications to backbone reference set analysis. Spherical kinematics plays a central role in defining the backbone reference set used throughout this work. Both the backbone curve shape and set of frames evolving about the curve depend on the rotation matrix  $\mathbf{Q}(s, t)$ .

### 2.2.1 An Euler-Angle Representation

Euler-Angles are perhaps the most widely used set of parameters for representing spatial rotations. The Euler-Angle description of spatial rotations consists of three consecutive rotations about three coordinate axes. There are many different ways of representing Euler-Angles depending on which axes are rotated about, and in what order the rotations are performed. One common choice is

$$\mathbf{Q} = \text{ROT}[\bar{\mathbf{e}}_3, \alpha_1] \text{ROT}[\bar{\mathbf{e}}_1, \alpha_2] \text{ROT}[\bar{\mathbf{e}}_3, \alpha_3]. \quad (2.2.1)$$

Throughout this work, an uncommon choice of

$$\mathbf{Q}_E = \text{ROT}[\bar{\mathbf{e}}_3, -K] \text{ROT}[\bar{\mathbf{e}}_1, T] \text{ROT}[\bar{\mathbf{e}}_2, R] \quad (2.2.2)$$

is chosen, because it is consistent with the convention introduced in Section 2.1.2, and it results in a simple expression for the backbone curve position vector. A subscript “*E*” is used in this section to denote Euler-Angle parametrization of a rotation matrix. The resulting rotation matrix is given explicitly as:

$$\mathbf{Q}_E = \begin{pmatrix} \cos K \cos R + \sin K \sin T \sin R & \sin K \cos T & \cos K \sin R - \sin K \sin T \cos R \\ -\sin K \cos R + \cos K \sin T \sin R & \cos K \cos T & -(\cos K \sin R + \sin K \sin T \cos R) \\ -\cos T \sin R & \sin T & \cos T \cos R \end{pmatrix}. \quad (2.2.3)$$

The position of points on a spatial backbone curve using this parametrization of  $SO(3)$  can be represented by the equations:

$$\bar{x}(s, t) = \begin{pmatrix} \int_0^s l(\sigma, t) \sin K(\sigma, t) \cos T(\sigma, t) d\sigma \\ \int_0^s l(\sigma, t) \cos K(\sigma, t) \cos T(\sigma, t) d\sigma \\ \int_0^s l(\sigma, t) \sin T(\sigma, t) d\sigma \end{pmatrix}. \quad (2.2.4)$$

$K(s, t)$  and  $T(s, t)$  have the physical interpretation of angles which fix the direction of the tangent for each value of  $s$ , while  $l(s, t)$  specifies the length of the tangent vector, as shown in Figure 2.3.

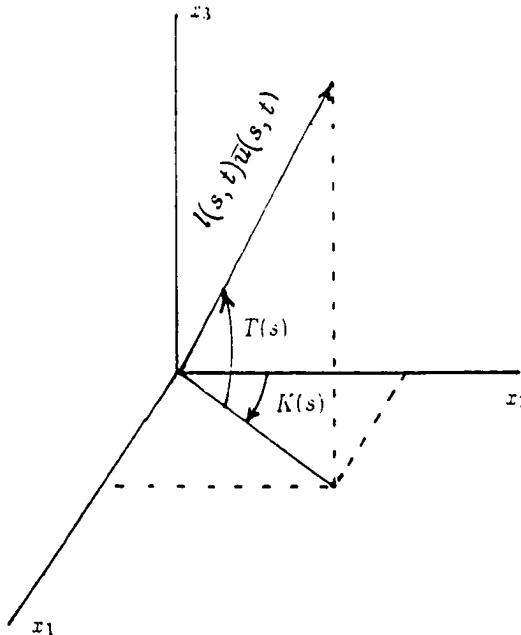


Figure 2.3: Physical Interpretation of  $K(s, t)$ ,  $T(s, t)$ , and  $l(s, t)$

By convention,  $K(0, t) = T(0, t) = 0$ . By abuse of language, the derivatives of  $K$  and  $T$  with respect to  $L$  are referred to here as *quasi-curvature* and *quasi-torsion* respectively. In other words,

$$K(s, t) = \int_0^s l(s, t) \mathcal{K}(\sigma, t) d\sigma \quad (2.2.5)$$

$$T(s, t) = \int_0^s l(s, t) \mathcal{T}(\sigma, t) d\sigma \quad (2.2.6)$$

where  $\mathcal{K}$  and  $\mathcal{T}$  are respectively the quasi-curvature and quasi-torsion.

A set of *parametrization induced* reference frames is defined as

$$\mathbf{Q}_{\text{PI}}(s, t) = \text{ROT}[\bar{\mathbf{e}}_3, -K] \text{ROT}[\bar{\mathbf{e}}_1, T]. \quad (2.2.7)$$

The column vectors of  $\mathbf{Q}_{\text{PI}}$  are denoted  $\bar{\Xi}_i(s, t)$  and are given explicitly as:

$$\bar{\Xi}_2 = \begin{pmatrix} \sin K \cos T \\ \cos K \cos T \\ \sin T \end{pmatrix}; \bar{\Xi}_3 = \begin{pmatrix} -\sin K \sin T \\ -\cos K \sin T \\ \cos T \end{pmatrix}; \bar{\Xi}_1 = \begin{pmatrix} \cos K \\ -\sin K \\ 0 \end{pmatrix}. \quad (2.2.8)$$

These are respectively called the *tangent*, *complementary vector*, and *planar-normal*. In this system,  $\Xi_1$  always lies in a plane parallel to the  $x$ - $y$  plane of the base frame. This frame will be useful in the next chapter when we merge real mechanisms to the backbone curve. The intrinsic parameters of the classical Frenet-Serret parametrization and this alternate parametrization can be related as follows:

$$\kappa^2 = T^2 + \mathcal{K}^2 \cos^2 T \quad (2.2.9)$$

$$\tau = -\mathcal{K} \sin T - \frac{\partial}{\partial L} [\text{Atan2}(T, \mathcal{K} \cos T)] \quad (2.2.10)$$

where  $\text{Atan2}(y, x)$  is the two argument arc tangent function [Crai86]. Note also that

$$\mathbf{Q}_{\text{FS}} = \text{ROT}[\bar{u}, \text{Atan2}(T, \mathcal{K} \cos T)] \hat{\mathbf{Q}}_{\text{PI}} \quad (2.2.11)$$

where

$$\hat{\mathbf{Q}}_{\text{PI}} = [\bar{\Xi}_2 \bar{\Xi}_3 \bar{\Xi}_1]$$

is a permuted version of  $\mathbf{Q}_{\text{PI}}$ . This is necessary because the classical formulation defines the first column of  $\mathbf{Q}_{\text{FS}}$  to be the tangent, whereas it is defined as the second column in this thesis.

The *induced roll distribution*  $R(s, t)$ , which was used in Equations (2.2.2-3), is defined with respect to the parametrization induced reference frames as follows: The backbone reference frame  $\{F_R(s, t)\}$  is assumed to be coincident with the parametrization induced reference frame for all  $s$  at time  $t = t_0$ .  $\{F_R(s, t)\}$  is then allowed to rotate about the  $\Xi_2$  axis by angle  $R(s, t)$ . The unit basis vectors of  $\{F_R(s, t)\}$  are given by:  $\{\bar{\Psi}_1(s, t), \bar{\Psi}_2(s, t), \bar{\Psi}_3(s, t)\}$ , where  $\bar{\Psi}_2(s, t) = \bar{\Xi}_2(s, t)$ . The roll distribution is the angle defined by

$$R = \cos^{-1} (\bar{\Xi}_1 \cdot \bar{\Psi}_1), \quad (2.2.12)$$

and measures how much the body fixed frame twists along the manipulator backbone curve with respect to the parametrization induced reference frame. The restriction  $R(0, t) = 0$  is imposed because the orientation of the manipulator base is fixed. The rate of induced roll with respect to arc length,  $\mathcal{R}$ , is defined such that

$$R(s, t) = \int_0^s l(s, t) \mathcal{R}(\sigma, t) d\sigma. \quad (2.2.13)$$

As indicated by Equation (2.2.11), the roll distribution defined with respect to the Frenet Frames and the parametrization induced reference frames differs by  $\text{Atan2}(\mathcal{T}, \mathcal{K} \cos T)$ . In Chapter 5, an “absolute” reference frame is defined as the reference frame of least twisting about the backbone curve. The absolute reference frame provides a means by which all others can be compared quantitatively.

The complete backbone reference set can thus be completely specified with the set of parameters:  $\{L, K, T, R\}$ , or their rate versions:  $\{l, \mathcal{K}, \mathcal{T}, \mathcal{R}\}$ .

While the Euler-Angle representation of spatial rotations is convenient and physically meaningful, it is often necessary to use 4-parameter descriptions. For instance, in Chapter 5, use of Euler-Angles will be restricted because of algorithmic singularities which can result. The next two sections review two common 4-parameter descriptions which avoid this problem.

## 2.2.2 Matrix Exponential Representation

Every rotation in  $SO(3)$  can be represented in terms of its axis of rotation, defined by a unit vector  $\bar{z} = [z_1, z_2, z_3]^T$ , and angle of rotation  $\alpha$ .

The corresponding matrix representation is

$$\mathbf{Q} = \exp[\alpha \mathbf{S}] = \mathbf{I} + \sin \alpha \mathbf{S} + (1 - \cos \alpha) \mathbf{S}^2 \quad (2.2.14)$$

where

$$\mathbf{S}(s, t) = \begin{bmatrix} 0 & -z_3(s, t) & z_2(s, t) \\ z_3(s, t) & 0 & -z_1(s, t) \\ -z_2(s, t) & z_1(s, t) & 0 \end{bmatrix}. \quad (2.2.15)$$

In matrix exponential form, a curve in  $\mathbf{R}^3$  can be written according to the framework in Equations (2.1.18) and (2.1.19) as

$$\bar{x}(s, t) = \int_0^s l(s, t) \exp[\alpha(\sigma, t) \mathbf{S}(\sigma, t)] \bar{e}_2 d\sigma. \quad (2.2.16)$$

It should be noted that the tangent is *not* the vector  $\bar{z}$ , but rather:

$$\bar{u}(s, t) = \exp[\alpha(s, t) \mathbf{S}(s, t)] \bar{e}_2. \quad (2.2.17)$$

In a more explicit computational form:

$$\bar{x}(s, t) = \begin{pmatrix} \int_0^s l(\sigma, t) [z_1(\sigma, t) z_2(\sigma, t) (1 - \cos \alpha(\sigma, t)) - z_3(\sigma, t) \sin \alpha(\sigma, t)] d\sigma \\ \int_0^s l(\sigma, t) [z_2^2(\sigma, t) (1 - \cos \alpha(\sigma, t)) + \cos \alpha(\sigma, t)] d\sigma \\ \int_0^s l(\sigma, t) [z_1(\sigma, t) z_2(\sigma, t) (1 - \cos \alpha(\sigma, t)) + z_1(\sigma, t) \sin \alpha(\sigma, t)] d\sigma \end{pmatrix}. \quad (2.2.18)$$

A backbone reference set can thus be parametrized with the set  $\{l, \alpha, \bar{z}\}$  under the constraint  $\bar{z} \cdot \bar{z} = 1$ .

### 2.2.3 Parametrization via Unit Quaternions

Unit quaternions are another 4-parameter set commonly used for describing spatial rotations. While the Euler-Angle parametrization defined in Section 2.2.1 will be used extensively, that description of spatial rotations becomes singular in some situations. As will be seen in Chapter 5, quaternions are particularly useful for formulating dynamical problems because the quaternion description of rotation does not become singular.

With this parametrization, the matrix describing the orientation of the backbone reference frame at any point is:

$$\mathbf{Q}_\lambda = \begin{pmatrix} \lambda_0^2 + \lambda_1^2 - \lambda_2^2 - \lambda_3^2 & 2(\lambda_1 \lambda_2 - \lambda_0 \lambda_3) & 2(\lambda_1 \lambda_3 + \lambda_0 \lambda_2) \\ 2(\lambda_1 \lambda_2 + \lambda_2 \lambda_3) & \lambda_0^2 - \lambda_1^2 + \lambda_2^2 - \lambda_3^2 & 2(\lambda_2 \lambda_3 - \lambda_0 \lambda_1) \\ 2(\lambda_1 \lambda_3 - \lambda_0 \lambda_2) & 2(\lambda_2 \lambda_3 + \lambda_0 \lambda_1) & \lambda_0^2 - \lambda_1^2 - \lambda_2^2 + \lambda_3^2 \end{pmatrix} \quad (2.2.19)$$

with the constraint :

$$\lambda_0^2 + \lambda_1^2 + \lambda_2^2 + \lambda_3^2 = \bar{\lambda} \cdot \bar{\lambda} = 1, \quad (2.2.20)$$

where  $\lambda_i = \lambda_i(s, t)$ , and  $\bar{\lambda} = [\lambda_1, \lambda_2, \lambda_3, \lambda_4]$ .

In this instance, the backbone curve is given by

$$\bar{x}(s, t) = \begin{pmatrix} 2 \int_0^s l(\sigma, t)(\lambda_2(\sigma, t)\lambda_3(\sigma, t) - \lambda_1(\sigma, t)\lambda_4(\sigma, t))d\sigma \\ \int_0^s l(\sigma, t)(\lambda_1^2(\sigma, t) - \lambda_2^2(\sigma, t) + \lambda_3^2(\sigma, t) - \lambda_4^2(\sigma, t))d\sigma \\ 2 \int_0^s l(\sigma, t)(\lambda_3(\sigma, t)\lambda_4(\sigma, t) + \lambda_1(\sigma, t)\lambda_2(\sigma, t))d\sigma \end{pmatrix}. \quad (2.2.21)$$

The backbone reference set is fully described with  $\{l, \bar{\lambda}\}$  and the constraint  $\bar{\lambda} \cdot \bar{\lambda} = 1$ .

## 2.3 Parametrization of Planar Curves

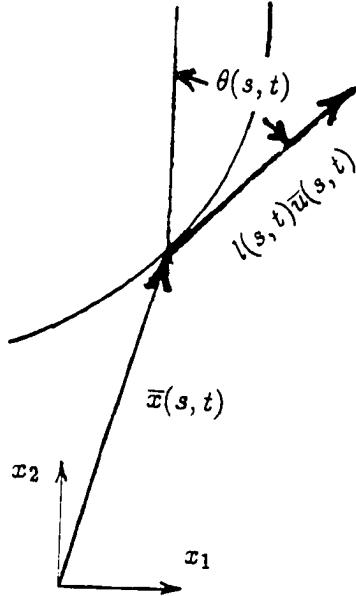
The parametrization of spatial curves discussed previously in this chapter simplifies in the planar case. A planar curve is the locus of points which have position defined by  $\bar{x}(s, t) = [x_1(s, t), x_2(s, t)]^T$ , where

$$x_1(s, t) = \int_0^s l(\sigma, t) \sin \theta(\sigma, t) d\sigma \quad (2.3.1)$$

$$x_2(s, t) = \int_0^s l(\sigma, t) \cos \theta(\sigma, t) d\sigma. \quad (2.3.2)$$

$\theta(s, t)$  is the clockwise measured angle which the tangent to the curve at point  $s$  makes with the  $x_2$ -axis at time  $t$ . Henceforth, the restriction  $\theta(0, t) = 0$  is observed to indicate that the base of the curve has fixed orientation. Figure 2.4 illustrates the physical meaning of  $l(s, t), \theta(s, t)$ . A simple relationship exists between the curvature function of the curve and the functions  $\theta$ , and  $l$ :

$$\kappa = \frac{\partial \theta}{\partial L} = \frac{1}{l} \frac{\partial \theta}{\partial s}. \quad (2.3.3)$$



**Figure 2.4:** Physical Interpretation of  $\theta$  and  $l$

Throughout this thesis Equations (2.3.1) and (2.3.2) are interchangeable with:

$$x_1(s, t) = \int_0^s l(\sigma, t) \sin \left( \int_0^\sigma l(\nu, t) \kappa(\nu, t) d\nu \right) d\sigma \quad (2.3.4)$$

$$x_2(s, t) = \int_0^s l(\sigma, t) \cos \left( \int_0^\sigma l(\nu, t) \kappa(\nu, t) d\nu \right) d\sigma. \quad (2.3.5)$$

Note that Equations (2.3.1-2) and (2.3.4-5) correspond to the Euler Angle parametrization with  $T(s, t) = R(s, t) = 0, K(s, t) = \theta(s, t)$ , and  $\kappa(s, t) = K(s, t)$ .

In the case of nonextensible curves,  $l = 1$ ,  $L = s$ , and  $\kappa$  is simply the derivative of  $\theta$  with respect to  $s$ . Note that strictly speaking  $\kappa$  is defined as a magnitude in Equation (2.1.9). A positive sense has been assigned to curvatures which cause the curve to bend clockwise, and a negative sense to counterclockwise bending.

Complex number representation of rotations can also be used. Using complex number representation of points in the plane, we have

$$z(s, t) = x_1(s, t) + jx_2(s, t) \quad (2.3.6)$$

with the real and imaginary parts corresponding to the natural basis vectors for  $\mathbf{R}^2$ . Counterclockwise rotation of an arbitrary vector  $\bar{v} = [v_1, v_2]^T$  in the plane by

an angle  $\alpha$  is achieved by representing the vector as a complex number  $v_1 + jv_2$ , and multiplying by the complex exponential:

$$\exp[j\alpha] = \cos \alpha + j \sin \alpha. \quad (2.3.7)$$

Equations (3.3.1-2), which can be written as:

$$\bar{x}(s, t) = \int_0^s l(\sigma, t) \text{ROT}[\bar{e}_3, -\theta(\sigma, t)] \bar{e}_2 d\sigma, \quad (2.3.8)$$

can thus be rewritten as:

$$z(s, t) = j \int_0^s l(\sigma, t) \exp[-j\theta(\sigma, t)] ds, \quad (2.3.9)$$

where in complex notation  $j$  represents the unit vector  $\bar{e}_2$ .

## 2.4 Discussion

This chapter reviewed and extended classical methods for describing curves and associated sets of frames in space. The classical geometry of curves is concerned almost exclusively with intrinsic curve shape. The analysis in this chapter added several new features to the old theory. These new features include the ability to specify an arbitrary roll distribution about a curve, and the ability to extend and contract curve segments in a controlled manner. The following chapters will show how this formulation is natural for specifying hyper-redundant manipulator configurations, and incorporating physical constraints.

## Chapter 3

### Fitting Manipulators to Backbone Reference Sets

Two issues are resolved in this chapter. First, methods for “fitting” hyper-redundant robots to backbone reference sets are developed. Second, relationships between the parameters  $\{L, K, T, R\}$  and joint limits are examined. In Section 3.1, a fully parallelizable algorithm is formulated for hyper-redundant manipulator inverse kinematics. In this algorithm, the backbone reference set is used to provide the kinematic inputs for each non-redundant section, or “module,” of a hyper-redundant manipulator. Using the Euler-Angle parametrization of Section 2.2.2, the curves can easily be partitioned into sections corresponding to physical manipulator modules. As a result, the mechanism kinematics problem completely decouples once the backbone reference set is specified. The relationship between mechanism joint limits and backbone reference set geometry is described within this framework. The presentation in Section 3.1 is appropriate for general modular hyper-redundant morphologies. Sections 3.2-3.5 present several examples of particular mechanisms which illustrate the theory developed in Section 3.1. Section 3.6 presents a fitting procedure tailored to a serial hyper-redundant manipulator composed of revolute joints in order to illustrate how alternatives to the general parallel algorithm can be developed.

### 3.1 Inverse Kinematics in Parallel

A parallel algorithm based on the formulation of Chapter 2 is introduced here. For the sake of simplicity, manipulators with a modular architecture are considered. For example, the modules of an extensible spatial hyper-redundant manipulator might be Stewart platforms. It is assumed for simplicity that the modules are uniform in structure and size.

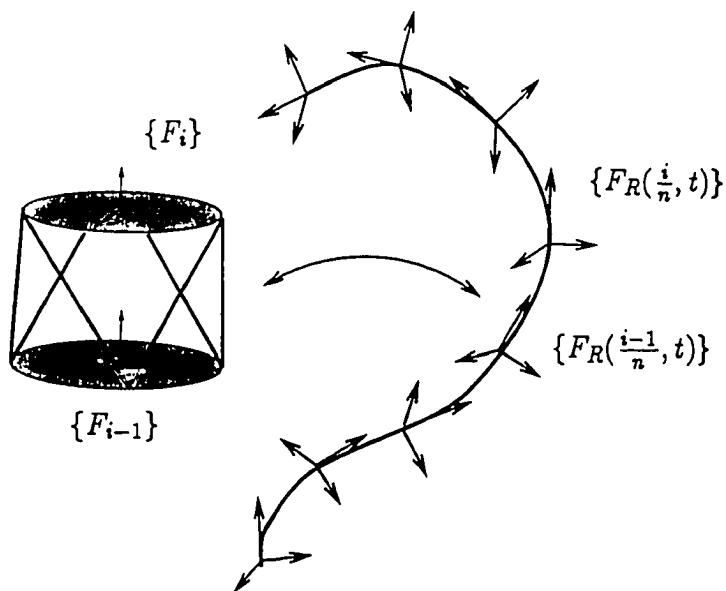


Figure 3.1: Backbone Reference Set and Mechanism Modules

The backbone reference set can be used to generate inverse kinematic solutions for modular manipulators as follows. Consider the  $i^{th}$  module in the manipulator chain consisting of  $n$  modules (see Figure 3.1). Attach a time-varying frame,  $\{F_{i-1}(t)\}$ , to the “input,” or base, of the module, and a frame,  $\{F_i(t)\}$ , to the “output,” or top, of the module. For the discretely segmented modular manipulator configuration to conform to the continuous curve geometry, the frames  $\{F_{i-1}(t)\}$  and  $\{F_i(t)\}$  are chosen to coincide with the backbone reference frames at points given by  $s = (i - 1)/n$  and  $s = i/n$  respectively. That is, equate  $\{F_i(t)\}$  to  $\{F_R(i/n, t)\}$ , which was defined in Section 2.2.1. Recall that equal partitioning of the curve parameter need not imply equal spacing along the curve, because

$L(s, t)$  can be chosen from a broad class of functions.

The  $4 \times 4$  homogeneous transform relating  $\{F_i\}$  to  $\{F_{i-1}\}$  is denoted by  $H_{i-1}^i$ . This consists of the relative translation,  $\bar{r}_{i-1}^i$ , and rotation,  $R_{i-1}^i$ , of  $\{F_i\}$  with respect to  $\{F_{i-1}\}$ , i.e.,

$$H_{i-1}^i(\bar{q}^{M_i}) = \begin{pmatrix} R_{i-1}^i(\bar{q}^{M_i}) & \bar{r}_{i-1}^i(\bar{q}^{M_i}) \\ \bar{0}^T & 1 \end{pmatrix}. \quad (3.1.1)$$

$\bar{q}^{M_i} \in \mathbf{R}^m$  is the vector of joint displacements which determine the geometry of the  $i^{th}$  module. In the plane,  $m = 3$ , while in space  $m = 6$ . It is assumed that the inverse kinematics of the module, which relate  $\{F_i\}$  to  $\{F_{i-1}\}$ , can be solved in a closed or efficient form.

The manipulator configuration will conform to the backbone reference set if:

$$H_{i-1}^i(\bar{q}^{M_i}) = \mathcal{H}^{-1}\left(\frac{i-1}{n}, t\right) \mathcal{H}\left(\frac{i}{n}, t\right) \quad (3.1.2)$$

where  $\mathcal{H}(s, t)$  is defined in Equation (2.1.23). Equation (3.1.2) can be rewritten as:

$$R_{i-1}^i(\bar{q}^{M_i}) = Q^T((i-1)/n, t)Q(i/n, t) \equiv P_{i-1}^i(t) \quad (3.1.3a)$$

and

$$\bar{r}_{i-1}^i(\bar{q}^{M_i}) = Q^T((i-1)/n, t)[\bar{x}(i/n, t) - \bar{x}((i-1)/n, t)] \equiv \bar{p}_{i-1}^i(t) \quad (3.1.3b)$$

$Q(s, t)$  is the rotation matrix relating the orientation of the backbone reference frame at point  $s$  to the base frame at  $s = 0$ . In other words,  $Q(s, t)$  describes the orientation of the frame  $\{F_R(s, t)\}$  with respect to the base frame.  $P_{i-1}^i(t)$  and  $\bar{p}_{i-1}^i(t)$  represent the orientation and position of the frame  $\{F_R(\frac{i}{n}, t)\}$  with respect to the frame  $\{F_R(\frac{i-1}{n}, t)\}$ . Equations (3.1.2-3) specify that the frames fixed in the manipulator,  $\{F_i(t)\}$ , coincide with the backbone reference frames,  $\{F_R(i/n, t)\}$ , at the discrete points  $s = i/n$  for all  $i \in [0, n]$ . In the planar case  $T(s, t) = R(s, t) = 0$  for all  $s, t$ , and the above equations reduce to:

$$\bar{p}_{i-1}^i = \begin{pmatrix} \int_{\frac{i-1}{n}}^{\frac{i}{n}} l(s, t) \sin[\theta(s, t) - \theta(\frac{i-1}{n}, t)] ds \\ \int_{\frac{i-1}{n}}^{\frac{i}{n}} l(s, t) \cos[\theta(s, t) - \theta(\frac{i-1}{n}, t)] ds \end{pmatrix} \quad (3.1.4)$$

and

$$\mathbf{P}_{i-1}^i(t) = \begin{pmatrix} \cos \theta_M^i & \sin \theta_M^i \\ -\sin \theta_M^i & \cos \theta_M^i \end{pmatrix}. \quad (3.1.5)$$

$\theta_M^i(t) = \theta(\frac{i}{n}, t) - \theta(\frac{i-1}{n}, t)$  is the relative rotation of the frame at  $s = \frac{i}{n}$  with respect to the one at  $s = \frac{i-1}{n}$ .

Assume that functions  $\{L(s, t), T(s, t), K(s, t), R(s, t)\}$  have been specified. Each  $\mathbf{R}_{i-1}^i$  and  $\bar{\mathbf{r}}_{i-1}^i$  can be computed in parallel as a function of the backbone reference set geometry. For example, in the plane let  $\theta(s, t) = a(t)s$  and  $l(s, t) = l(t)$ . Thus:

$$\bar{x}(s, t) = \begin{pmatrix} \frac{l(t)}{a(t)} [1 - \cos a(t)s] \\ \frac{l(t)}{a(t)} \sin a(t)s \end{pmatrix} \quad \mathbf{Q}(s, t) = \begin{pmatrix} \cos a(t)s & \sin a(t)s \\ -\sin a(t)s & \cos a(t)s \end{pmatrix} \quad (3.1.6)$$

and so

$$\bar{p}_{i-1}^i(t) = \begin{pmatrix} \frac{l(t)}{a(t)} \left[ 1 - \cos \frac{a(t)}{n} \right] \\ \frac{l(t)}{a(t)} \sin \frac{a(t)}{n} \end{pmatrix} \quad \mathbf{P}_{i-1}^i(t) = \begin{pmatrix} \cos \frac{a(t)}{n} & \sin \frac{a(t)}{n} \\ -\sin \frac{a(t)}{n} & \cos \frac{a(t)}{n} \end{pmatrix}, \quad (3.1.7)$$

which provide the kinematic inputs for each module in Equation (3.1.3). The inverse kinematics of each module can be performed in parallel, and so this method can in theory be applied to manipulators with an arbitrary number of degrees of freedom with the same computation time, provided each module has an associated processor. This method is applicable to a wide variety of morphologies. The computation of these quantities could be performed with a microprocessor dedicated to module kinematics, or analog circuitry designed specifically for this task.

Actuator limits can be accounted for using the backbone curve approach by establishing bounds on the parametric functions which define the curve bending and extension/contraction. The kinematics of the  $i^{th}$  module of a hyper-redundant manipulator consisting of  $n$  modules is mapped, or “fitted,” to the backbone curve with Equations (3.1.2) and (3.1.3). These equations can be rewritten as follows:

$$\bar{X}_i(t) = \bar{f}_i(\bar{q}^{M_i}(t)). \quad (3.1.8)$$

$\bar{f}_i(\cdot)$  is a vector function containing the same information as the homogeneous transformation  $H_{i-1}^i$ .  $\bar{X}_i(t)$  contains the same information as  $\mathcal{H}^{-1}(\frac{i-1}{n}, t)\mathcal{H}(\frac{i}{n}, t)$ . In the planar case  $\bar{X}_i, \bar{q}^{M_i}, \bar{f}_i(\cdot) \in \mathbf{R}^3$ , and in the spatial case  $\bar{X}_i, \bar{q}^{M_i}, \bar{f}_i(\cdot) \in \mathbf{R}^6$ . Examples of this will be seen in the following sections.

Within this framework, the module forward kinematic function can be inverted (at least in concept) to solve for the joint angles as a function of the curve parametrization:

$$\bar{q}^{M_i} = \bar{f}_i^{-1}(\bar{X}_i). \quad (3.1.9)$$

Joint limits can then be related to curve parameters through the module inverse kinematics as:

$$q_j^{\min} \leq q_j^{M_i} \leq q_j^{\max}, \quad (3.1.10)$$

where  $j \in [1, 3]$  in the planar case and  $j \in [1, 6]$  in the spatial case.  $q_j^{\min}$  and  $q_j^{\max}$  are respectively the minimum and maximum allowable joint displacements.

## 3.2 A Planar Serial Chain Manipulator

This section demonstrates the general parallel computational algorithm introduced in the previous sections for the specific example of a planar revolute 3n-link manipulator, where  $n \geq 1$ . The manipulator is broken down into  $n$  sets of three link modules. The 3R module was chosen because within its workspace it can arbitrarily position and orient the frame attached to the 3<sup>rd</sup> link with respect to the base. The forward kinematics of the manipulator is given by

$$x_{ee} = \sum_{i=1}^{3n} L_i \sin \left( \sum_{j=1}^i q_j \right); \quad y_{ee} = \sum_{i=1}^{3n} L_i \cos \left( \sum_{j=1}^i q_j \right) \quad (3.2.1)$$

where  $x_{ee}$  and  $y_{ee}$  are the end-effector position coordinates. Unless otherwise stated, it will be assumed that  $L_i = \frac{1}{3n}$  for all  $i \in [1, 3n]$ , so that the total manipulator length is normalized to unity.  $q_j$  is the angle of the  $j^{\text{th}}$  joint, i.e., the angle between the  $j - 1^{\text{st}}$  and  $j^{\text{th}}$  links measured clockwise. Figure 3.2(a) shows

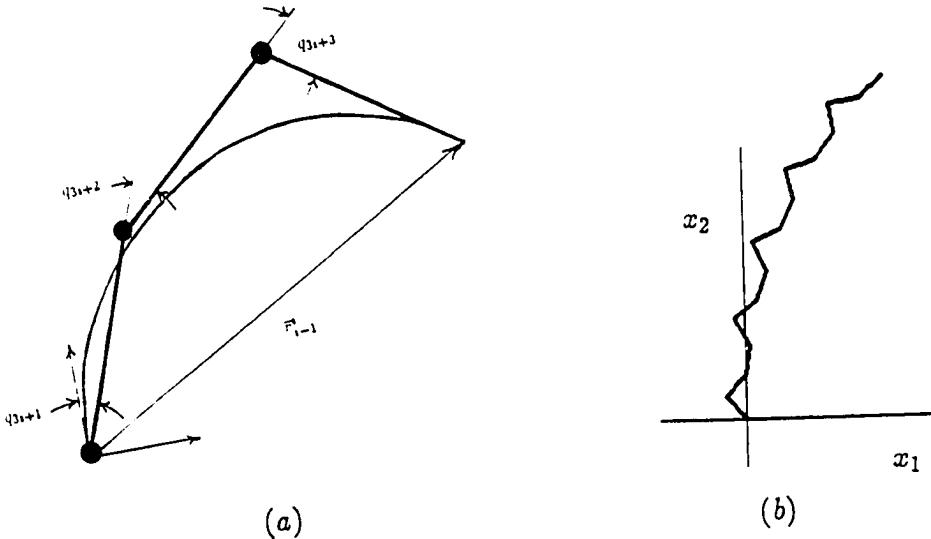
how the inverse kinematics of each triad is determined by matching it to a section of the backbone reference set.

The vector  $\bar{r}_{i-1}^i$  in Figure 3.2 is the relative position vector defined in Equation (3.1.1). For this example:

$$\bar{r}_{i-1}^i = \frac{1}{3n} \begin{pmatrix} \sum_{k=3i+1}^{3i+3} \sin(\sum_{j=3i+1}^k q_j) \\ \sum_{k=3i+1}^{3i+3} \cos(\sum_{j=3i+1}^k q_j) \\ 0 \end{pmatrix} \quad (3.2.2a)$$

$$R_{i-1}^i = \text{ROT}[-\bar{e}_3, q_{3i+1} + q_{3i+2} + q_{3i+3}] \quad (3.2.2b)$$

which are equated to Equations (3.1.4) and (3.1.5) respectively. Solving the inverse kinematics of each triad determines the joint angles  $q_{3i+1}, q_{3i+2}, q_{3i+3}$  as a function of curve geometry and the spacing of modules along the curve. The relationship between manipulator length and length of the curve segments can be specified to insure that each curve segment is in the reachable workspace of each triad of the actual manipulator.



**Figure 3.2: Five Sets of 3R Modules Affixed to a Backbone**

The inverse kinematics for the 3-link segment can be simply computed via the

following sequence:

$$\theta_M^i = \theta\left(\frac{i}{n}\right) - \theta\left(\frac{i-1}{n}\right) \quad (3.2.3a)$$

$$t_1^i = (p_{i-1}^i)_1 - \sin(\theta_M^i)/3n \quad (3.2.3b)$$

$$t_2^i = (p_{i-1}^i)_2 - \cos(\theta_M^i)/3n \quad (3.2.3c)$$

$$c_2 = (9n^2|t^i|^2 - 2)/2; \quad s_2 = \pm(1 - c_2^2)^{\frac{1}{2}} \quad (3.2.3d)$$

$$q_{3i+1} = \pi/2 - \text{Atan2}(t_2^i, t_1^i) + \text{Atan2}(s_2, 1 + c_2) \quad (3.2.3e)$$

$$q_{3i+2} = -\text{Atan2}(s_2, c_2) \quad (3.2.3f)$$

$$q_{3i+3} = \theta_{ee}^i - q_{3i+1} - q_{3i+2} \quad (3.2.3g)$$

where  $(p_{i-1}^i)_1$  and  $(p_{i-1}^i)_2$  are the first and second components of the vector  $\bar{p}_{i-1}^i$ . The up or down elbow solutions can be chosen so as to cause the manipulator to adhere to the curve segment as closely as possible, as long as this does not cause instantaneous changes in module pose.

Figure 3.2(b) shows how the  $3n$ -link manipulator considered in this section appears when  $n = 5$ . In this case, the backbone curve is a circular arc with  $\theta(s, t) = a(t)s$  and  $l(s, t) = l(t)$  for  $a(t) = 0.8$ , and  $l(t) = 0.8$ . With the backbone curve confined to this form, Equations (3.1.5-7) provide the kinematic inputs required in Equation (3.2.3).

In this case, it is fortunate that a closed form relationship exists between the backbone curve geometric factors  $\{l(t), a(t)\}$  and the mechanism joint angles. Joint limits are related to the backbone curve geometry by substituting Equations (3.1.5-7) into Equation (3.2.3) and observing the constraints in Equation (3.1.10), where in this case  $q_j^{M_i} = q_{3i+j}$  for  $i \in [0, 4]$  and  $j \in [1, 3]$ .

### 3.3 A Manipulator with Closed Loop Modules

The module design presented in this section has three actuatable degrees of freedom. For the  $i^{th}$  module, the actuatable degrees of freedom are denoted  $\theta_{3i+1}$ ,

$\theta_{3i+2}$ , and  $\theta_{3i+3}$  (see Figure 3.3(a)). This design can be viewed as two coupled 3-link planar manipulator triads. Figure 3.3(b) shows that the triad on the left is in an “up-elbow” pose, while the triad on the right is in a “down-elbow” pose.

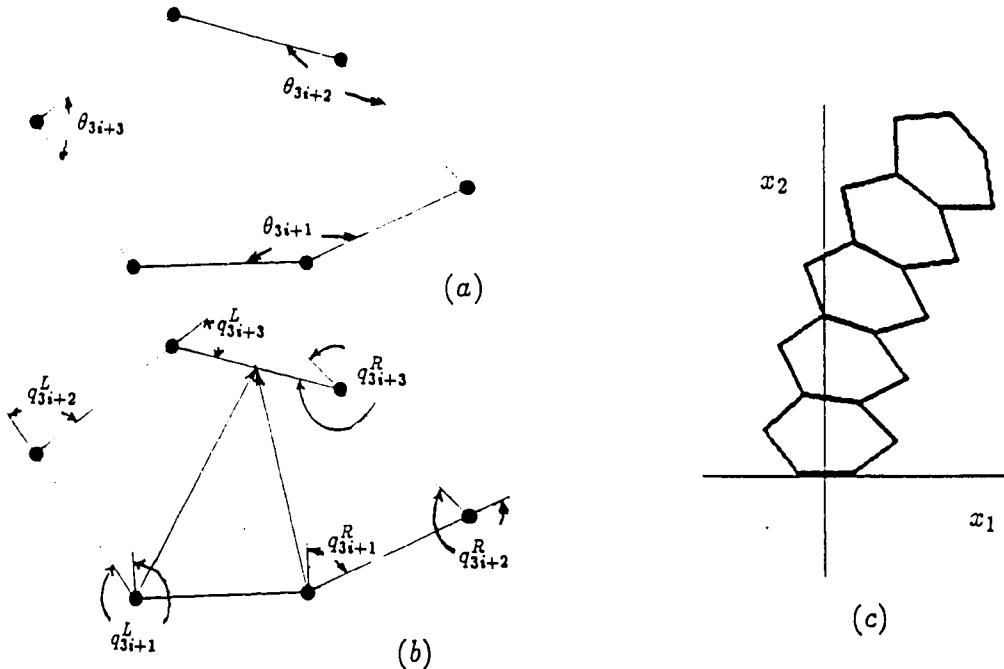


Figure 3.3: Closed Loop Modular Design

The inverse kinematics is trivially specified by using the solution from the previous section. In this case, however, the scaling of the link-lengths in relation to the curve segment length is different. In this case, the nondimensional link lengths are  $L_i = \frac{1}{\sqrt{3n}}$  where  $n$  is the number of modules. Figure 3.3(c) shows a cascade of five modules with the same backbone curve as the example in the previous section. Because the inverse kinematics is similar, determining joint limits as a function of the time dependence of the curve is also similar.

### 3.4 A Planar Truss Manipulator

This section demonstrates the methods of Sections 3.1 and 3.2 with a planar truss manipulator. Figure 3.4(a) shows one module of the planar truss manipulator.

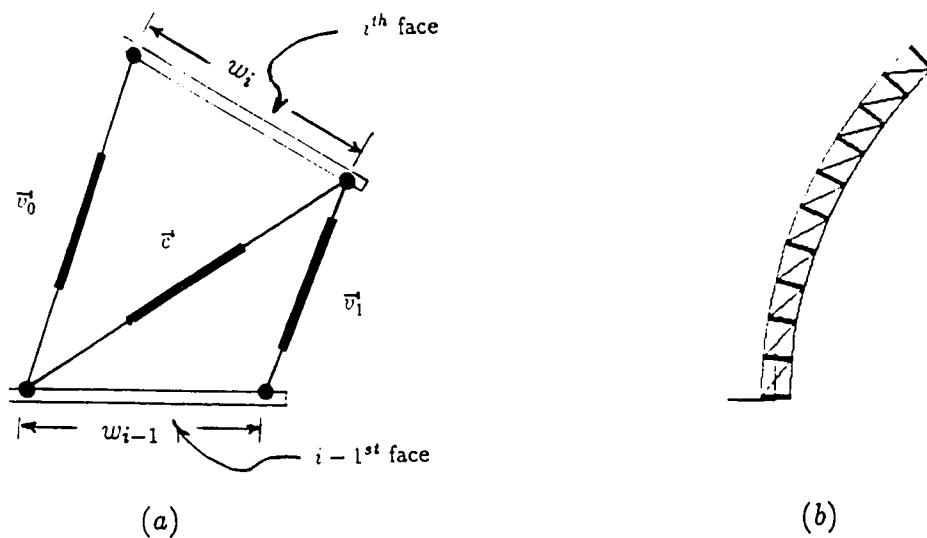
In this case, one segment of the truss is composed of side members and a cross element. The position vectors connecting like vertices in the truss are denoted  $\bar{v}_0^i$  on the left and  $\bar{v}_1^i$  on the right.  $\bar{c}^i$  denotes the cross element. These vectors can be determined from the continuous curve model as follows:

$$\begin{aligned}\bar{v}_j^i &= \bar{p}_{i-1}^i - \bar{n}_j^{i-1} + \text{ROT}(-\bar{e}_3, \theta_M^i) \bar{n}_j^i \quad j = 1, 2 \\ \bar{c}^i &= \bar{p}_{i-1}^i - \bar{n}_1^{i-1} + \text{ROT}(-\bar{e}_3, \theta_M^i) \bar{n}_2^i\end{aligned}\quad (3.4.1)$$

where  $\theta_M^i = \theta(i/n, t) - \theta((i-1)/n, t)$  and  $\bar{n}_j^i$  are the vectors to the  $j^{th}$  vertex of the  $i^{th}$  platform in the frame affixed to that platform. For this specific example,

$$\bar{n}_1^i = [-w_i/2, 0]^T; \quad \bar{n}_2^i = [w_i/2, 0]^T$$

where  $w_i$  is the width of each horizontal face of the truss, as denoted in Figure 3.4.



**Figure 3.4: A Planar Variable Geometry Truss Manipulator**

The controlled degrees of freedom are the lengths

$$\begin{aligned}L_j^i &= ||\bar{v}_j^i|| \\ L_3^i &= ||\bar{c}^i||\end{aligned}\quad (3.4.2)$$

for  $i = 1, \dots, n$ , and  $j = 1, 2$ . Thus, Equations (3.4.1) and (3.4.2) provide the inverse kinematics solution for this module geometry. The backbone curve example from Equations (3.1.6) and (3.1.7) is used again, resulting in the configurations shown in Figure 3.4(b). Joint limits are of the form given in Equation (3.1.10), where now  $q_j^{M_i} = L_j^i$  for  $j \in [1, 3]$  and  $i \in [1, 10]$ . Joint limits are related to the backbone curve geometry by observing Equations (3.1.5-7) and (3.4.1-2).

### 3.5 A Spatial VGTM

This section applies the methodology to a highly articulated spatial truss structure. Figure 3.5 shows the geometry of one possible type of highly articulated truss structure, which has been examined in [NacH89,SalRR88]. Here the geometry of the base and top faces of each truss segment is fixed, and the vertical and diagonal elements can expand and contract to change the truss geometry.

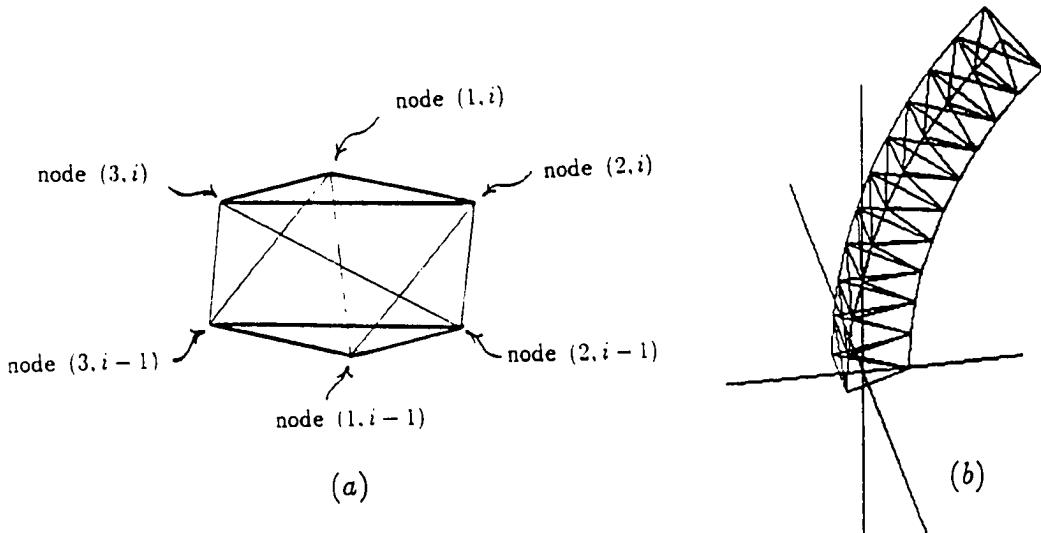


Figure 3.5: A Highly Articulated Spatial VGTM

The inverse kinematics problem reduces to the determination of the lengths of the

truss elements which cause the manipulator to conform as closely as possible to the continuous solution. The lengths of the truss elements required to generate a position and orientation of the  $i^{th}$  face relative to the  $i-1^{st}$  face can be determined as follows. Denote the position vector from node  $(j, i-1)$  to node  $(j, i)$  by  $\bar{v}_j^i$  for  $j = 1, 2, 3$ . These vectors are defined in the body fixed frame  $\{F_{i-1}\}$ , and can be computed as:

$$\bar{v}_j^i = \bar{r}_{i-1}^i + (P_{i-1}^i - I_3)\bar{n}_j. \quad (3.5.1)$$

$\bar{n}_j$  are the vectors to the vertices in the frame affixed to the center of each face, which is defined by the unit vectors  $\{\bar{\Xi}_i\}$ . For the particular example given, these vectors are:

$$\begin{aligned} \bar{n}_1 &= w[1, 0, 0]^T \\ \bar{n}_2 &= w[-\sin \pi/6, 0, -\cos \pi/6]^T \\ \bar{n}_3 &= w[-\sin \pi/6, 0, \cos \pi/6]^T \end{aligned} \quad (3.5.2)$$

where  $w$  determines the width of the truss.

The three cross elements are each denoted by  $\bar{c}_j^i$  for  $j = 1, 2, 3$ . They have the explicit form:

$$\bar{c}_1^i = \bar{r}_{i-1}^i - \bar{n}_1 + P_{i-1}^i \bar{n}_2 \quad (3.5.3a)$$

$$\bar{c}_2^i = \bar{r}_{i-1}^i - \bar{n}_2 + P_{i-1}^i \bar{n}_3 \quad (3.5.3b)$$

$$\bar{c}_3^i = \bar{r}_{i-1}^i - \bar{n}_3 + P_{i-1}^i \bar{n}_1. \quad (3.5.3c)$$

The actuatable degrees of freedom are the magnitudes of each of these vectors.

These are denoted by the symbol  $\lambda_j^i$ , where

$$\begin{aligned} \lambda_j^i &= \|\bar{v}_j^i\| \\ \lambda_{j+3}^i &= \|\bar{c}_j^i\|. \end{aligned} \quad (3.5.4)$$

The forward problem of determining the relative position and orientation of the top face with respect to the bottom one for each module is more difficult but need not be solved for this method to be applied.

Figure 3.5 shows a VGTM with 10 bays, adhering to the same backbone curve defined in Section 3.1.

## 3.6 Morphology-Specific Fitting Procedures

Previously in this chapter, a parallelizable algorithm for fitting modular hyper-redundant manipulators to backbone curves was developed and demonstrated. This section illustrates an alternate approach for a particular manipulator morphology: a planar n-link serial-chain manipulator. The purpose of this example is to illustrate that alternatives to the general fitting procedure exist. In some cases, it may even be desirable to use morphology-specific fitting procedures instead of the general procedure developed earlier in this chapter.

In this example, the joint angles  $\{q_j\}$  of the discrete manipulator are computed directly from the backbone curve as:

$$q_{j+1} = \theta \left( \frac{2j+1}{2n} \right) - \theta \left( \frac{2j-1}{2n} \right) + \epsilon_{j+1}. \quad (3.6.1)$$

In other words, the angles of the discrete case are taken to be approximately the change in angle over a corresponding section of the backbone curve (from  $s = (2j-1)/n$  to  $s = (2j+1)/n$ ) whose length corresponds to an individual link length. This approximation will lead to errors, and to account for these errors,  $n$  free “fitting” parameters,  $\{\epsilon_j\}$ , are introduced.

The  $\{\epsilon_j\}$  are computed to minimize a function which is the sum of squared distance between points on the backbone curve located at  $s = \frac{i}{n}$  (for  $i = 1, \dots, n$ ) and points on the discrete manipulator:

$$G = \frac{1}{2} \sum_{k=1}^n \left[ \left( \int_0^{\frac{k}{n}} \sin \theta ds - \frac{1}{n} \sum_{i=1}^k \sin \sum_{j=1}^i q_j \right)^2 + \left( \int_0^{\frac{k}{n}} \cos \theta ds - \frac{1}{n} \sum_{i=1}^k \cos \sum_{j=1}^i q_j \right)^2 \right]. \quad (3.6.2)$$

Assuming each  $\epsilon_i$  is small, Equation (3.6.2) can be linearized to provide  $n$  linear equations in the  $n$  unknown  $\{\epsilon_i\}$  (see [ChB89-1,2] for details). If the computed values of  $\{\epsilon_i\}$  are small, the linearization assumptions are justified. If the  $\{\epsilon_i\}$

are not sufficiently small, iteration of the above procedure or a nonlinear fitting technique can be used. This will generally only be the case when there are points on the backbone curve with large curvature magnitude. Such cases can be avoided by restricting the maximum curvature magnitude, possibly at the expense of reducing the volume of the workspace associated with a particular set of modes. For other methods of fitting discrete manipulators to backbone curves, see [ChB89-1,ChB91-3].

### 3.7 Discussion

This chapter introduced a general parallel computational algorithm for fitting robotic mechanisms to the backbone reference set developed in the previous chapter. The method was demonstrated with three planar and one spatial hyper-redundant manipulators. A variety of other candidate modules which could be cascaded to form spatial hyper-redundant manipulators can be found in the literature [Wu86,TeB89,PiFD91]. Chapters 4-7 will use the kinematic framework developed in Chapters 2 and 3 for the motion planning of hyper-redundant manipulators and mobile robots.

**TASK ANALYSIS  
AND  
SIMULATION**

# Chapter 4

## A Modal Approach to Hyper-Redundancy Resolution

In a broad sense, “modes” describe fundamental characteristics of any complex system, e.g., modes of operation of organized systems, or natural modes of linear systems. This chapter presents a “modal approach” to “hyper-redundancy resolution.” Hyper-redundancy resolution is a means by which the numerous excess degrees of freedom of a hyper-redundant system can be specified to perform useful tasks. Desirable characteristics of any hyper-redundancy resolution technique are computational efficiency, and cyclicity, i.e., a closed path motion of the end-effector causes a closed path in the joint space of the robot. The modal approach possesses both of these characteristics.

Using the modal approach, hyper-redundancy is resolved by confining the functions  $\{L(s, t), K(s, t), T(s, t), R(s, t)\}$  to a weighted sum of shape functions (or operating modes) which are sufficient to perform a specified task. As the task requirements change, the set of modes may be changed. The manipulator is artificially confined to modes via the backbone reference set. Several types of modes can be defined including bending (in two orthogonal planes), rolling, and extending. The task which is performed by hyper-redundant manipulators in this chapter is end-effector placement in a workspace without obstacles. As will be seen

in later chapters, the modal approach plays a central role in obstacle avoidance, locomotion, and grasping algorithms.

This chapter begins by motivating the choice of the set of parametric functions  $\{K, T, R, L\}$  defined in Section 2.2.1 to describe the geometry of the backbone reference set. In Sections 4.2-4.4 planar bending, extension, and mixed modes are defined and used to resolve hyper-redundancy. Section 4.5 examines the spatial problem. In Section 4.6, a *modal Jacobian* is defined and used in a manner analogous to the Jacobians of standard manipulators. In Section 4.7, methods for switching between sets of modes are discussed, and drawbacks of the modal approach are analyzed. One such drawback is the existence of “modal singularities,” which can be analogous to the singularities of mechanical systems. Alternate approaches to hyper-redundancy resolution, which are based on optimality criteria, are investigated in the next chapter.

## 4.1 Motivation for Choice of Parametrization

There are several compelling reasons to use Euler-Angle parametrizations of curves and their associated frames. The first is that all the parameters are physically meaningful. Second, unlike four parameter descriptions of spherical displacements (such as quaternions), Euler-Angles need no constraint equations. Lastly, the Euler-Angle description can be expressed as coupled planar problems. Some methods developed for planar problems are therefore automatically applicable to spatial problems.

The similarity in mathematical form between spatial and planar curves, as parametrized in Sections 2.2.1 and 2.3 respectively, can be seen from the trigonometric relationships:

$$\begin{aligned}\sin K \cos T &= \frac{1}{2} [\sin(K + T) + \sin(K - T)] \\ \cos K \cos T &= \frac{1}{2} [\cos(K + T) + \cos(K - T)].\end{aligned}$$

The equations for the position of the end point of a spatial curve, which corresponds to the end-effector of a hyper-redundant manipulator, can then be written as

$$\int_0^1 l(s, t) \sin \theta^+(s, t) ds + \int_0^1 l(s, t) \sin \theta^-(s, t) ds = 2x_1(1, t) = 2x_{ee}(t) \quad (4.1.1)$$

$$\int_0^1 l(s, t) \cos \theta^+(s, t) ds + \int_0^1 l(s, t) \cos \theta^-(s, t) ds = 2x_2(1, t) = 2y_{ee}(t) \quad (4.1.2)$$

$$\int_0^1 l(s, t) \sin \frac{1}{2}(\theta^+(s, t) - \theta^-(s, t)) ds = x_3(1, t) = z_{ee}(t) \quad (4.1.3)$$

where

$$\theta^+(s, t) = K(s, t) + T(s, t); \quad \theta^-(s, t) = K(s, t) - T(s, t).$$

Thus the geometry of space curves can be represented as "coupled" planar problems with this parametrization.

Furthermore, using this parametrization, it is easy to solve the inverse kinematics problem for orientation. If the end-effector orientation is specified by a vector of direction cosines  $\bar{\gamma}(t) = [\gamma_1(t), \gamma_2(t), \gamma_3(t)]^T$  and a roll  $R_{ee}(t)$  measured with respect to the parametrization induced frame  $\{\Psi_i\}$ , the conditions which must be satisfied are

$$\begin{aligned} \sin \theta^+(1, t) + \sin \theta^-(1, t) &= 2\gamma_1(t) \\ \cos \theta^+(1, t) + \cos \theta^-(1, t) &= 2\gamma_2(t) \end{aligned} \quad (4.1.4)$$

$$\sin \frac{1}{2}(\theta^+(1, t) - \theta^-(1, t)) = \gamma_3(t) \quad (4.1.5)$$

$$R(1, t) = R_{ee}(t) \quad (4.1.6)$$

Note that Equations (4.1.4) are equivalent to the forward kinematics equations of a two link revolute-jointed planar manipulator with unit length links, and the solution to that problem can be used to calculate two solutions:

$$\begin{aligned} \theta^-(1, t) &= \text{Atan2}(\gamma_1, \gamma_2) - \text{Atan2}(\gamma_3(1 - \gamma_3^2)^{\frac{1}{2}}, 1 - \gamma_3^2) \\ \theta^+(1, t) &= \theta^-(1, t) + \text{Atan2}(2\gamma_3(1 - \gamma_3^2)^{\frac{1}{2}}, 1 - 2\gamma_3^2). \end{aligned} \quad (4.1.7)$$

$$\begin{aligned} \theta^-(1, t) &= \text{Atan2}(\gamma_1, \gamma_2) - \text{Atan2}(-\gamma_3(1 - \gamma_3^2)^{\frac{1}{2}}, 1 - \gamma_3^2) \\ \theta^+(1, t) &= \theta^-(1, t) + \text{Atan2}(-2\gamma_3(1 - \gamma_3^2)^{\frac{1}{2}}, 1 - 2\gamma_3^2). \end{aligned} \quad (4.1.8)$$

It can also be shown that Equation (4.1.5) is automatically satisfied by the solutions of Equations (4.1.4). Equation (4.1.6) is solved independently of the other equations. Equations (4.1.7) or (4.1.8) are used to provide the end-effector orientation constraints  $K(1, t)$ ,  $T(1, t)$ , and  $R(1, t)$ .

The inverse kinematics problem of finding  $K$ ,  $T$ ,  $R$ , and  $L$  to achieve a particular end-effector position is not as easily accomplished. The “modal approach” presented in the next sections is one method to make this problem tractable.

## 4.2 End-Effector Placement Using Bending Modes

The forward kinematics of a planar backbone curve can be computed by exact or numerical integration of Equations (4.1.1-3). However, the inverse kinematic problem can have an infinite number of solutions. The inverse kinematic problem for planar nonextensible hyper-redundant manipulators can be simplified by constraining  $\theta(s, t)$  to the modal form:

$$\theta(s, t) = \sum_{i=1}^{N_\theta} a_i(t) \Phi_i(s), \quad (4.2.1a)$$

$$l(s, t) = 1 \quad (4.2.1b)$$

where  $\Phi_i$  is a *mode function*, and  $a_i$  is a *modal participation factor*. A hyper-redundant manipulator with nonextensible backbone curve is said to be confined to pure bending modes if Equations (4.2.1) hold, and  $T(s, t) = R(s, t) = 0$ . Under these conditions,  $\kappa(s, t) = \partial\theta(s, t)/\partial s$  is also of a modal form, with curvature modes  $\phi_i = \dot{\Phi}_i$ .  $N_\theta$  is the number of bending modes, which depends upon the number of end-effector or other task constraints. In this way, the end-effector coordinates become a function of  $\{a_i\}$ . The  $\{\Phi_i\}$  are specified functions, and thus the inverse kinematics problem reduces to finding the  $\{a_i\}$  which satisfy task constraints. When  $N_\theta$  equals the number of constraints, inverse kinematic solutions based on Equation (4.2.1) serve as a means of hyper-redundancy resolution for nonextensible planar manipulators.

For some choices of modes, exact closed form inverse kinematic solutions can be found. For example, consider the following choice of modes for  $N_\theta = 2$ :

$$\Phi_1(s) = \sin 2\pi s; \quad \Phi_2(s) = 1 - \cos 2\pi s \quad (4.2.2)$$

Substituting Equation (4.2.2) into Equations (4.2.1), (2.3.1), and (2.3.2), and evaluating at  $s = 1$ , it can be shown using identities in [Bow58,Dw61] that the forward kinematics equations reduce to

$$x_{ee} = \sin(a_2)J_0 \left[ (a_1^2 + a_2^2)^{\frac{1}{2}} \right] \quad y_{ee} = \cos(a_2)J_0 \left[ (a_1^2 + a_2^2)^{\frac{1}{2}} \right] \quad (4.2.3)$$

where  $J_0$  is the zero<sup>th</sup> order Bessel function.

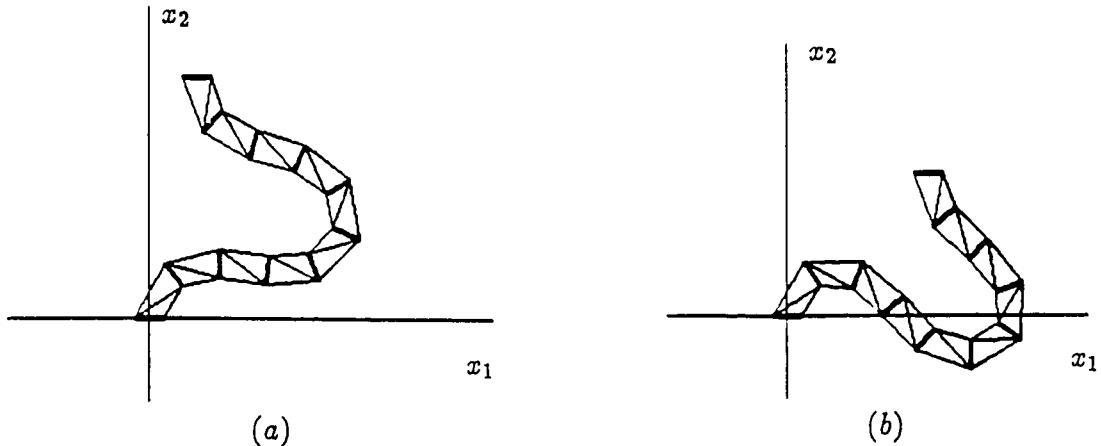


Figure 4.1: Continuous Bending Modes

The “inverse kinematics” (evaluation of modal participation factors) in this case can be computed as:

$$a_1 = \hat{a}_1^\pm(\bar{x}_{ee}) = \pm \left( \left[ J_0^{-1} \left[ (x_{ee}^2 + y_{ee}^2)^{\frac{1}{2}} \right] \right]^2 - [\text{Atan2}(x_{ee}, y_{ee})]^2 \right)^{\frac{1}{2}} \quad (4.2.4)$$

$$a_2 = \hat{a}_2(\bar{x}_{ee}) = \text{Atan2}(x_{ee}, y_{ee}). \quad (4.2.5)$$

$J_0^{-1}$  is the “restricted inverse Bessel function of zero order,” and is defined as the inverse of  $J_0(x)$  for  $0 < x < \mu$  where  $\mu \approx 3.832$  is the first local minimum of

$J_0$ . The plus and minus sign distinguishes two possible poses. Other poses would exist if other intervals of the argument were permitted in computing the inverse of  $J_0(x)$ . In general, inverse kinematic computations using the modal approach are analogous to the inverse kinematic computations of nonredundant manipulators, with modal participation factors serving as generalized joint displacements. However, one significant difference is that the modal approach can generate an infinite number of poses. In this case, there are two poses corresponding to each interval of the zeroth order Bessel function bounded by local minima and maxima, though not for all points in the workspace. Nonredundant manipulators only have a finite number of poses for a given end-effector position and/or orientation. Figure 4.1 shows a manipulator confined to the bending modes in Equation (4.2.2). The end-effector position and modal participation factors in Figures 4.1(a) and 4.1(b) are respectively  $(x_{ee}, y_{ee}, a_1, a_2) = (0.10, 0.50, 1.50, 0.20), (0.30, 0.30, 1.46, 0.80)$ . A VGTM has been fit to the curve using the method described in Section 3.4.

Other closed form modal solutions are easily found when smoothness constraints are relaxed. For instance, a revolute joint is mimicked with a Dirac delta function in curvature,  $\delta(s - s_0)$ , which integrates to a Heaviside unit step function in the tangent angle,  $H(s - s_0)$ . Consider the following set of modes:

$$\Phi_1(s) = \sin(2\pi s); \quad \Phi_2(s) = H(s - 0^+). \quad (4.2.6)$$

The corresponding forward kinematics is written as:

$$x_{ee} = J_0(a_1) \sin a_2; \quad y_{ee} = J_0(a_1) \cos a_2, \quad (4.2.7)$$

with inverse kinematics:

$$a_1 = J_0^{-1} \left[ (x_{ee}^2 + y_{ee}^2)^{\frac{1}{2}} \right]; \quad a_2 = \text{Atan2}(x_{ee}, y_{ee}). \quad (4.2.8)$$

Similarly, if

$$\Phi_1(s) = 1 - \cos(2\pi s); \quad \Phi_2(s) = H(s - 0^+) \quad (4.2.9)$$

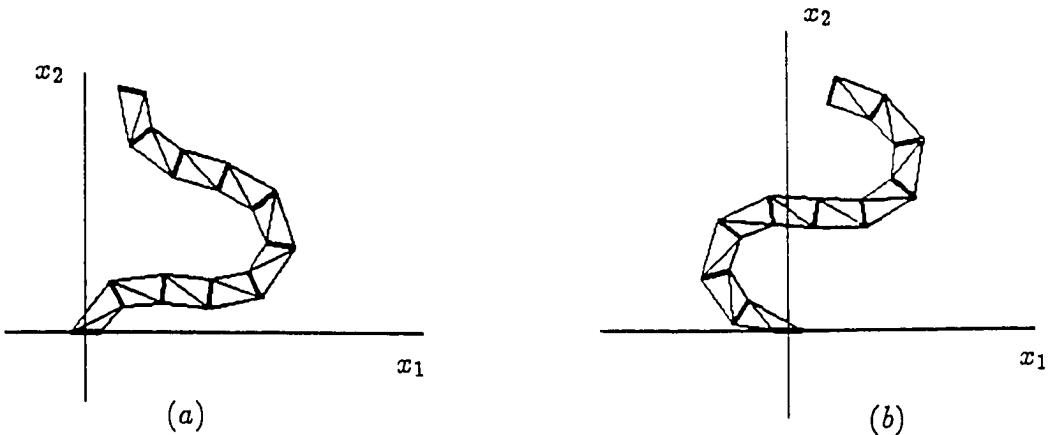
are chosen, the forward kinematics becomes

$$x_{ee} = J_0(a_1) \sin(a_1 + a_2); \quad y_{ee} = J_0(a_1) \cos(a_1 + a_2), \quad (4.2.10)$$

and the inverse kinematics is

$$a_1 = J_0^{-1} \left[ (x_{ee}^2 + y_{ee}^2)^{\frac{1}{2}} \right]; \quad a_2 = \text{Atan2}(x_{ee}, y_{ee}) - a_1. \quad (4.2.11)$$

In both of the above examples,  $\Phi_2(s) = H(s - 0^+)$  indicates a finite clockwise rotation about the base of the manipulator by an angle  $a_2$ . Configurations corresponding to the modes in Equations (4.2.6) and (4.2.9) are shown in Figures 4.2(a) and 4.2(b) respectively. The end-effector positions relative to the manipulator bases and participation factors are  $(x_{ee}, y_{ee}, a_1, a_2) = (0.10, 0.50, 1.52, 0.20), (0.10, 0.50, 1.50, -1.28)$  for Figures 4.2(a) and 4.2(b) respectively.



**Figure 4.2:** Mixed Continuous/Discontinuous Bending Modes

Using discontinuous modes exclusively, standard mechanism kinematics can be mimicked. In fact, the kinematics of many machines and mechanisms can be viewed as specific cases of the modal approach. An example of nonsmooth bending modes for  $N_\theta = 3$  is

$$\Phi_1(s) = H(s - L_1); \quad \Phi_2(s) = H(s - L_2); \quad \Phi_3(s) = H(s - L_3) \quad (4.2.12)$$

where again  $H(\cdot)$  is a unit step function, and  $0 < L_1 < L_2 < L_3 < 1$  are constants. Substituting these modes into Equations (2.3.1) and (2.3.2) and evaluating at

$s = 1$ , the forward kinematics equations reduce to

$$x_{ee} = \mathcal{L}_1 \sin a_1 + \mathcal{L}_2 \sin(a_1 + a_2) + \mathcal{L}_3 \sin(a_1 + a_2 + a_3) \quad (4.2.13a)$$

$$y_{ee} = L_1 + \mathcal{L}_1 \cos a_1 + \mathcal{L}_2 \cos(a_1 + a_2) + \mathcal{L}_3 \cos(a_1 + a_2 + a_3) \quad (4.2.13b)$$

$$\theta_{ee} = a_1 + a_2 + a_3 \quad (4.2.13c)$$

where  $\mathcal{L}_1 = L_2 - L_1$ ,  $\mathcal{L}_2 = L_3 - L_2$ , and  $\mathcal{L}_3 = 1 - L_3$ . This is nothing more than the forward kinematics of a three-link planar revolute manipulator which is translated along the  $y$ -axis by  $L_1$ , and with joint angles  $\{a_1, a_2, a_3\}$  and link lengths  $\{L_2 - L_1, L_3 - L_2, 1 - L_3\}$ . The “inverse kinematics” (evaluation of modal participation factors) in this particular case can then be computed in the usual way:

$$a_1 = \frac{\pi}{2} - \text{Atan2}(y_{ee} - l_2 \cos \theta_{ee}, x_{ee} - l_2 \sin \theta_{ee}) + \text{Atan2}(l_1 s_2, l_0 + l_1 c_2) \quad (4.2.14a)$$

$$a_2 = -\text{Atan2}(s_2, c_2) \quad (4.2.14b)$$

$$a_3 = \theta_{ee} - a_1 - a_2 \quad (4.2.14c)$$

where

$$c_2 = \frac{(x_{ee} - l_2 \sin \theta_{ee})^2 + (y_{ee} - l_2 \cos \theta_{ee})^2 - l_1^2 - l_2^2}{2l_0 l_1} \quad s_2 = \pm(1 - c_2^2)^{\frac{1}{2}}.$$

Figure 4.3 shows three different sets of modes of the form given by Equation (4.2.12) (corresponding to different “link lengths”) for the same end-effector location. The lengths in the figure are  $\{L_1, L_2, L_3\} = \{0.1, 0.3, 0.9\}$ ,  $\{0.1, 0.5, 0.8\}$ , and  $\{0.2, 0.5, 0.7\}$  and the end-effector position is  $(x_{ee}, y_{ee}) = (0.3, 0.9)$ .

In the usual fitting procedure for this mechanism morphology presented in Chapter 3, the fixed truss elements are chosen to be normal to the backbone curve. However, when the backbone curve is not continuously differentiable, the normal is not well defined, and smoothing is used to define alternate orientations for the transverse truss elements. The function  $\theta(s, t) = \sum_{i=1}^3 a_i(t)H(s - L_i)$  in the above

example was smoothed to form a new distribution of angle along the backbone curve denoted  $\theta_{sm}(s, t)$ .  $\theta_{sm}(s, t)$  is equal to  $\theta(s, t)$  everywhere on the backbone curve except on the intervals  $L_i - \epsilon < s < L_i + \epsilon$  for  $i = 1, 2, 3$ . On these intervals a polynomial spline replaces the value of the function so that a smooth transition over these intervals is achieved. The boundary conditions for the spline are provided by the function  $\theta(s, t)$  evaluated at  $s = L_i \pm \epsilon$  for  $i = 1, 2, 3$ . In this way, a smooth transition results across the singularities while matching the original function outside an  $\epsilon$  neighborhood of the singularities.  $\epsilon$  is small compared to the smallest link length. While  $\theta(s, t)$  is used in all backbone curve computations, such as Equations (2.3.1-2),  $\theta_{sm}(s, t)$  is used for the fitting procedure. For the configurations in Figure 4.3,  $\epsilon = 0.02$ .

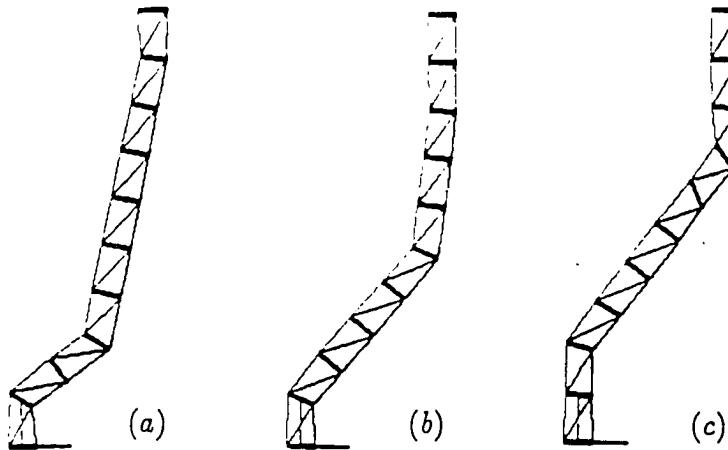


Figure 4.3: Discontinuous Bending Modes

### 4.3 End-Effector Placement Using Extension Modes

In contrast to the pure bending modes selected in the previous section, this section demonstrates how pure extension modes can be used to position the end-effector.

In this scenario,

$$\theta(s, t) = \theta_e(s) \quad (4.3.1)$$

is independent of time, i.e., the angle of the tangent to the backbone curve at each value of  $s$  does not change over time, while the rate of length function of the backbone curve is specified in modal form as:

$$l(s, t) = \sum_{i=1}^{N_l} a_i(t) \Phi_i(s). \quad (4.3.2)$$

Inequality constraints  $0 < l_{\min}(s) < l(s, t) < l_{\max}(s)$  are usually observed. The functions  $l_{\min}$  and  $l_{\max}$  are derived from the mechanism's ability to extend and contract with specified  $\theta_e(s)$ . It is important to note that because  $\theta_e$  is not a function of time, the end-effector orientation remains constant during pure extensions.

In this way, there is a linear dependence of the end-effector position on the participation factors given by:

$$x_{ee}(t) = \sum_{i=1}^{N_l} a_i(t) \int_0^1 \Phi_i(s) \sin \theta_e(s) ds \quad (4.3.3)$$

$$y_{ee}(t) = \sum_{i=1}^{N_l} a_i(t) \int_0^1 \Phi_i(s) \cos \theta_e(s) ds. \quad (4.3.4)$$

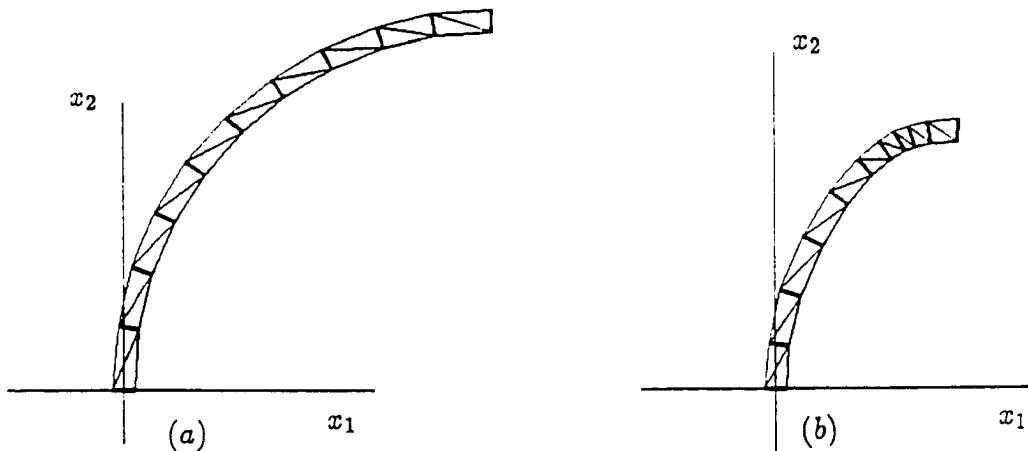
This can be written in the form

$$\bar{x}_{ee}(t) = \mathcal{J} \bar{a}(t), \quad (4.3.5)$$

where  $\mathcal{J}$  is a *modal Jacobian* matrix with constant elements which depend on the choice of modes but not the participation factors. Modal Jacobians will be dealt with in greater detail in Sections 4.6 and 4.7. In the case where  $N_l = 2$ , Equation (4.3.5) can be inverted to yield the "inverse kinematics" which provides the appropriate participation factors. It should also be noted that the integrals  $\int_0^1 \Phi_i \sin \theta_e ds$  and  $\int_0^1 \Phi_i \cos \theta_e ds$  need only be computed one time (and possibly off-line) because they are invariant with respect to participation factors and end-effector position. Figure 4.4 shows two configurations for a manipulator confined to the extension modes

$$\Phi_1(s) = \sin 2\pi s \quad \Phi_2(s) = 1 \quad (4.3.6)$$

with  $\theta_e(s) = \theta_{ee}s$ .  $\theta_{ee}$  is a constant end-effector orientation, which in this case is  $\pi/2$  radians. If  $\theta_{ee}$  were a function of time, this would constitute a set of mixed modes, which is the subject of the next section. The end-effector positions and participation factor values for Figures 4.4(a) and 4.4(b) are respectively  $(x_{ee}, y_{ee}, a_1, a_2) = (1.00, 1.00, 0.00, 1.56)$ ,  $(0.50, 0.70, 0.59, 0.94)$ .



**Figure 4.4:** Pure Extension Modes

## 4.4 Combined Bending and Extension Modes

The effective number of degrees of freedom of a planar hyper-redundant manipulator confined to both bending and extension modes is  $N = N_\theta + N_l$  where:

$$\theta(s, t) = \sum_{i=1}^{N_\theta} a_i(t) \Phi_i(s) \quad l(s, t) = \sum_{i=1}^{N_l} a_{\{i+N_\theta\}}(t) \Phi_{\{i+N_\theta\}}(s). \quad (4.4.1)$$

A class of manipulator configurations with closed form forward and inverse kinematic solutions which uses extension and bending modes can be defined as follows.

Consider for  $N_\theta = N_l = 1$ , the bending and extension modes:

$$\Phi_1(s) = v(s) \quad \Phi_2(s) = v'(s), \quad (4.4.2)$$

where  $v(s)$  is a strictly increasing function ( $v'(s) > 0$  for all  $s \in [0, 1]$ ) with  $v(0) = 0$  and  $v(1) = 1$ . The forward kinematics for this class of hyper-redundant manipulator configurations is:

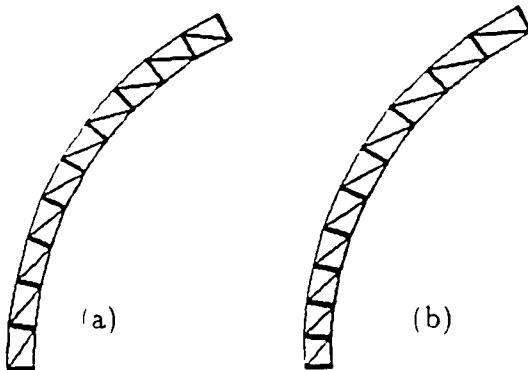
$$x_{ee} = \int_0^1 a_2 \Phi_2(s) \sin(a_1 \Phi_1(s)) ds = \frac{a_2}{a_1} (1 - \cos a_1) \quad (4.4.3a)$$

$$y_{ee} = \int_0^1 a_2 \Phi_2(s) \cos(a_1 \Phi_1(s)) ds = \frac{a_2}{a_1} \sin a_1. \quad (4.4.3b)$$

The inverse kinematics is:

$$a_1 = 2\text{Atan2}(x_{ee}, y_{ee}) \quad (4.4.4a)$$

$$a_2 = \frac{a_1 y_{ee}}{\sin a_1}. \quad (4.4.4b)$$



**Figure 4.5: Bending and Extension Modes**

Two examples of this class of modes are shown in Figure 4.5. In Figure 4.5(a),  $v(s) = s$ , and in Figure 4.5(b)  $v(s) = \frac{2}{3}(\frac{1}{2}s^2 + s)$ . In both cases, the end-effector and modal participation factor values are the same:  $(x_{ee}, y_{ee}, a_1, a_2) = (0.5, 0.8, 1.12, 0.99)$ . For all choices of  $v(s)$  chosen as mentioned above, the shape

of the backbone curve will be a circular arc. The distribution of actuator extensions will however depend on the particular choice of  $v(s)$ . Note that the curve used in Chapter 3, and defined in Equation (3.1.6), is a member of this class.

It should also be noted that there are other physically meaningful ways in which modes can be defined. Instead of restricting the function  $\theta(s, t)$  to modal form, the curvature function  $\kappa(s, t) = \frac{1}{l} \frac{d\theta}{ds}$  could have been restricted to modes. In the nonextensible case,  $l = 1$ , the two definitions of modes are the same but in the nonextensible case there is choice in specifying what quantities should be represented as a sum of weighted modes.

## 4.5 Spatial Modes

The modal method can be used to formulate spatial hyper-redundant manipulator kinematic algorithms by constraining  $\{R(s, t), K(s, t), T(s, t), l(s, t)\}$  to have the modal forms:

$$K(s, t) = \sum_{i=1}^{N_K} a_i(t) \Phi_i(s) \quad (4.5.1)$$

$$T(s, t) = \sum_{i=N_K+1}^{N_K+N_T} a_i(t) \Phi_i(s) \quad (4.5.2)$$

$$R(s, t) = \sum_{i=N_K+N_T+1}^{N_K+N_T+N_R} a_i(t) \Phi_i(s) \quad (4.5.3)$$

$$l(s, t) = \sum_{i=N_K+N_T+N_R+1}^{N_K+N_T+N_R+N_l} a_i(t) \Phi_i(s). \quad (4.5.4)$$

Solution techniques for the inverse kinematics of a spatial backbone curve are essentially the same as for the planar case, although it is somewhat more difficult to find closed form continuous mode solutions.

However, as stated in Section 4.1, closed form planar kinematics solutions can be used to find closed form spatial forward kinematic equations. For instance, by

restricting the functions  $\mathcal{K}$  and  $\mathcal{T}$  to the form:

$$\mathcal{K}(s, t) = 2\pi a_1 \cos 2\pi s + 2\pi a_2 \sin 2\pi s; \quad \mathcal{T}(s, t) = 2\pi a_3 \cos 2\pi s + 2\pi a_4 \sin 2\pi s, \quad (4.5.5)$$

with  $l(s, t) = 1$ , Equations (4.1.1-3) and (4.2.3) can be used to generate forward kinematic functions similar to the planar case. The forward kinematic equations corresponding to the modes in Equations (4.5.5) are:

$$\begin{aligned} x_{ee} &= \frac{1}{2} J_0[((a_1 + a_3)^2 + (a_2 + a_4)^2)^{\frac{1}{2}}] \sin(a_2 + a_4) \\ &\quad + \frac{1}{2} J_0[((a_1 - a_3)^2 + (a_2 - a_4)^2)^{\frac{1}{2}}] \sin(a_2 - a_4) \\ y_{ee} &= \frac{1}{2} J_0[((a_1 + a_3)^2 + (a_2 + a_4)^2)^{\frac{1}{2}}] \cos(a_2 + a_4) \\ &\quad + \frac{1}{2} J_0[((a_1 - a_3)^2 + (a_2 - a_4)^2)^{\frac{1}{2}}] \cos(a_2 - a_4) \\ z_{ee} &= J_0[(a_3^2 + a_4^2)^{\frac{1}{2}}] \sin a_4 \end{aligned} \quad (4.5.6)$$

which constrains the manipulator to four degrees of freedom. If a uniform extensibility and roll distribution are included, this would provide the six degrees of freedom needed to position and orient an object in space. Closed form inverse kinematic solutions for Equations (4.5.6) have not been found. Nonetheless, the existence of closed form forward kinematic solutions makes resolved rate formulations using the participation factors as generalized coordinates more efficient. Section 4.6 considers such numerical solutions, but first we examine several spatial modal configurations which do have closed form solutions.

For instance, we can take

$$\mathcal{K}(s, t) = W(s, 0, b)[a_1(t)\phi_1(s) + a_2(t)\phi_2(s)] \quad (4.5.7a)$$

and

$$\mathcal{T}(s, t) = a_3(t)\phi_3(s)W(s, b, 1) \quad (4.5.7b)$$

where  $b \in (0, 1)$  is a constant, and

$$W(s, s_0, s_1) = H(s - s_0) - H(s - s_1) \quad (4.5.8)$$

is a *window function* which is equal to unity on the interval  $s \in [s_0, s_1]$  and zero otherwise. By imposing the condition  $K(b, t) = 0$ , the effects of  $\mathcal{K}$  on the backbone reference frame at  $s = b$  are limited to translation in the  $x_1 - x_2$  plane. The effect of  $\mathcal{T}$  as defined in Equation (4.5.7b) is limited to the  $x_2 - x_3$  plane. The forward kinematic equations will take the form:

$$x_{ee} = x_1(a_1, a_2, b), \quad y_{ee} = x_2(a_1, a_2, b) + y_1(a_3, b), \quad z_{ee} = y_2(a_3, b) \quad (4.5.9)$$

the inverse of which are easily solvable for  $a_1, a_2, a_3$  as functions of  $x_{ee}, y_{ee}, z_{ee}$ , and  $b$  if the modes in the two orthogonal planes have closed form inverse kinematic solutions.

For example, if

$$\phi_1(s) = \frac{b}{2\pi} \sin \frac{2\pi s}{b} \quad \phi_2(s) = \frac{b}{2\pi} \cos \frac{2\pi s}{b} \quad \phi_3(s) = 1 \quad (4.5.10)$$

then the resulting forward kinematic equations are:

$$x_{ee} = b J_0 \left[ (a_1^2 + a_2^2)^{\frac{1}{2}} \right] \sin a_2 \quad (4.5.11a)$$

$$y_{ee} = b J_0 \left[ (a_1^2 + a_2^2)^{\frac{1}{2}} \right] \cos a_2 + \frac{1-b}{a_3} \sin a_3 \quad (4.5.11b)$$

$$z_{ee} = \frac{1-b}{a_3} (1 - \cos a_3) \equiv (1-b) F(a_3). \quad (4.5.11c)$$

The function  $F(\cdot)$  is defined for convenience. Equations (4.5.11) can be inverted to yield

$$a_3 = F^{-1} \left( \frac{z_{ee}}{1-b} \right) \quad (4.5.12a)$$

$$a_2 = \text{Atan2}(x_{ee}, y'_{ee}) \quad (4.5.12b)$$

$$a_1 = \pm \left( \left[ J_0^{-1} [(x_{ee}^2 + (y'_{ee})^2)^{\frac{1}{2}}] \right]^2 - a_2^2 \right)^{\frac{1}{2}} \quad (4.5.12c)$$

where  $y'_{ee} = y_{ee} - \frac{1-b}{a_3} \sin a_3$ , and  $F^{-1}(\cdot)$  can be defined over any part of the range of  $F(\cdot)$  for which  $F(\cdot)$  is monotonically increasing or decreasing. This example is simply the pure bending mode solution from Equations (4.2.2-4.2.5) with a circular arc bending in the  $x_2 - x_3$  plane.

Combined bending and extension modes can be used in the spatial case just as they were in the planar case. For instance, choosing

$$\begin{aligned} K(s, t) &= \pi s W(s, 0, \frac{1}{2}) + \frac{\pi}{2} W(s, \frac{1}{2}, 1) \\ T(s, t) &= 2a_1(s - \frac{1}{2}) W(s, \frac{1}{2}, 1) \\ l(s, t) &= a_2 W(s, 0, \frac{1}{2}) + a_3 W(s, \frac{1}{2}, 1), \end{aligned} \quad (4.5.13)$$

and substituting Equations (4.5.13) into the forward kinematic equations results in:

$$\bar{x}_{ee} = \begin{pmatrix} \frac{a_2}{\pi} + \frac{a_3}{2a_1} \sin a_1 \\ \frac{a_2}{\pi} \\ \frac{a_3}{2a_1} (1 - \cos a_1) \end{pmatrix}. \quad (4.5.14)$$

The corresponding inverse kinematics for this example have the closed form:

$$\begin{aligned} a_1 &= 2\text{Atan2}(z_{ee}, x_{ee} - y_{ee}) \\ a_2 &= \pi y_{ee} \\ a_3 &= \frac{2a_2(x_{ee} - y_{ee})}{\sin a_2} \end{aligned} \quad (4.5.15)$$

where  $z_{ee} = x_3(1)$ .

Another class of closed form spatial solutions is those which mimic standard nonredundant manipulators. For instance, if

$$K(s, t) = \frac{\pi}{2} H(s - \frac{1}{3}) \quad (4.5.16)$$

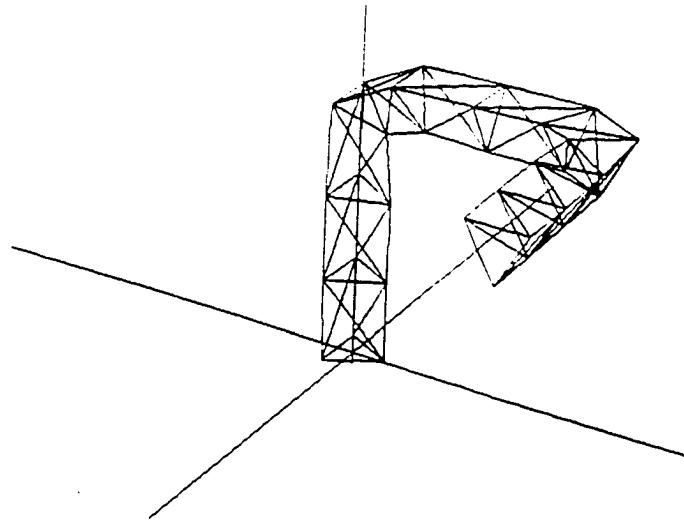
$$T(s, t) = \frac{\pi}{2} H(s - \frac{2}{3}) \quad (4.5.17)$$

$$l(s, t) = a_1(t) W(s, 0, \frac{1}{3}) + a_2(t) W(s, \frac{1}{3}, \frac{2}{3}) + a_3(t) W(s, \frac{2}{3}, 1) \quad (4.5.18)$$

the resulting curve mimics a three degree-of-freedom Cartesian manipulator, which has the simple kinematic equations:

$$a_1 = 3y_{ee}; \quad a_2 = 3x_{ee}; \quad a_3 = 3z_{ee}. \quad (4.5.19)$$

This choice of modes is shown in Figure 4.6 with a variable geometry truss.



**Figure 4.6:** Modes which Mimic a Cartesian Manipulator

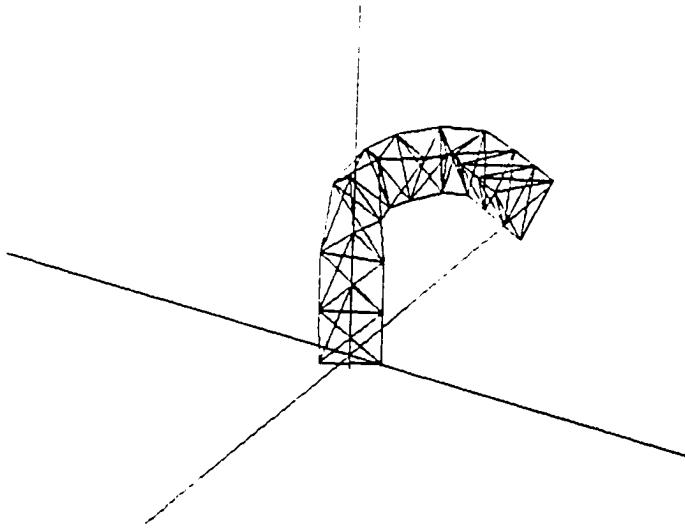
If we choose

$$K(s, t) = \frac{\pi}{2} W(s, \frac{1}{3}, 1) \quad (4.5.20)$$

$$T(s, t) = a_1(t) W(s, \frac{1}{3}, \frac{2}{3}) + a_2(t) W(s, \frac{2}{3}, 1) \quad (4.5.21)$$

$$l(s, t) = a_3(t) W(s, 0, \frac{1}{3}) + W(s, \frac{1}{3}, 1) \quad (4.5.22)$$

a SCARA-like manipulator is mimicked (see Figure 4.7). Likewise, a variety of other common manipulator morphologies can be mimicked while using the modal approach to hyper-redundancy resolution.



**Figure 4.7:** Modes which Mimic a SCARA Manipulator

## 4.6 Algorithms for Modal Inverse Kinematics

It is not necessary to have closed form inverse kinematic solutions to use the modal method. Look-up tables (or neural networks) can store the mapping between participation factors and end-effector coordinates for a given set of manipulator modes. Interpolation (or neural network generalization) can be used to interpolate the data. Thus the speed associated with closed form inverse kinematic solutions can be attained for a wide variety of modes.

Alternatively, an approach analogous to the “resolved rate” method reviewed in Chapter 1 can be used. The derivative of the forward kinematic map for a backbone curve restricted to modes is of the form:

$$\delta \bar{x}_{ee} = \mathcal{J} \delta \bar{a} \quad (4.6.1)$$

where  $\mathcal{J}$  is the *modal Jacobian*:

$$\mathcal{J}_{ik} = \frac{\partial x_i}{\partial a_k} \quad (4.6.2)$$

which is analogous to traditional manipulator Jacobians, only now participation factors are used instead of joint angles.

For a given initial set of modal participation factors,  $\bar{a}_0$ , Equation (4.6.1) can be solved to find incremental changes in modal participation factors to follow a desired end-effector trajectory. The following equation can be used to find the incremental changes in the modal participation factors:

$$\delta\bar{a} = \mathcal{J}^{-1}(\bar{a})\delta\bar{x}_{ee}. \quad (4.6.3)$$

These incremental changes can be used in an iterative procedure to modify the participation factors, i.e.,  $\bar{a}(t + \delta t) = \bar{a}(t) + \delta\bar{a}$ , starting with  $\bar{a}(0) = \bar{a}_0$ . For example, consider the two mode example of Equations (4.2.2-4.2.5). The modal Jacobian matrix is:

$$\mathcal{J} = \begin{bmatrix} \frac{\partial x_{ee}}{\partial a_1} & \frac{\partial x_{ee}}{\partial a_2} \\ \frac{\partial y_{ee}}{\partial a_1} & \frac{\partial y_{ee}}{\partial a_2} \end{bmatrix} \quad (4.6.4)$$

where:

$$\begin{aligned} \frac{\partial x_{ee}}{\partial a_1} &= -\frac{a_1}{(a_1^2 + a_2^2)^{\frac{1}{2}}} \sin(a_2) J_1[(a_1^2 + a_2^2)^{\frac{1}{2}}] \\ \frac{\partial x_{ee}}{\partial a_2} &= \cos(a_2) J_0[(a_1^2 + a_2^2)^{\frac{1}{2}}] - \frac{a_2}{(a_1^2 + a_2^2)^{\frac{1}{2}}} \sin(a_2) J_1[(a_1^2 + a_2^2)^{\frac{1}{2}}] \\ \frac{\partial y_{ee}}{\partial a_1} &= -\frac{a_1}{(a_1^2 + a_2^2)^{\frac{1}{2}}} \cos(a_2) J_1[(a_1^2 + a_2^2)^{\frac{1}{2}}] \\ \frac{\partial y_{ee}}{\partial a_2} &= -\sin(a_2) J_0[(a_1^2 + a_2^2)^{\frac{1}{2}}] - \frac{a_2}{(a_1^2 + a_2^2)^{\frac{1}{2}}} \cos(a_2) J_1[(a_1^2 + a_2^2)^{\frac{1}{2}}] \end{aligned} \quad (4.6.5)$$

where  $J_0$  and  $J_1$  are Bessel functions of the zeroth and first kind, and  $dJ_0(z)/dz = -J_1(z)$ .

If a closed form forward kinematics equation cannot be derived, the modal Jacobian matrix elements can also be computed numerically. For the case of nonextensible planar manipulators, the Jacobian elements can be numerically computed as:

$$\begin{aligned}\mathcal{J}_{1j} &= \frac{\partial x_{ee}}{\partial a_j} = \int_0^1 \Phi_j(\sigma) \cos[\theta(\sigma, t)] d\sigma \\ \mathcal{J}_{2j} &= \frac{\partial y_{ee}}{\partial a_j} = - \int_0^1 \Phi_j(\sigma) \sin[\theta(\sigma, t)] d\sigma\end{aligned}\quad (4.6.6)$$

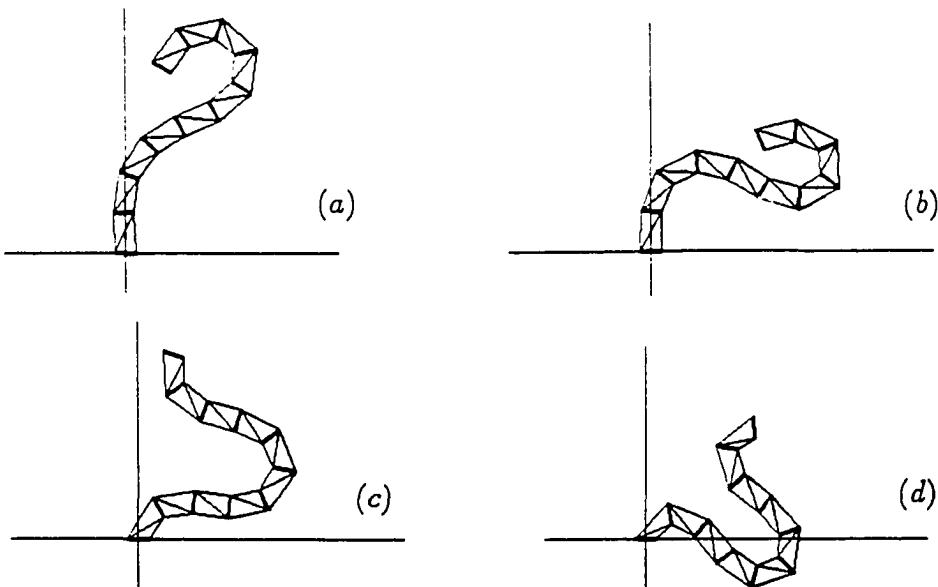
Figure 4.8 shows two examples of manipulator configurations confined to modes which do not have closed form inverse kinematic solutions. The rate formulation expressed in Equation (4.6.3) was used to solve for the participation factors as a function of end-effector position. In these cases, Equation (4.6.6) was used. Figures 4.8(a)-(b) show a manipulator confined to the two modes:

$$\Phi_1(s) = 1 - \cos 2\pi s \quad \Phi_2(s) = s \quad (4.6.7a)$$

for  $(x_{ee}, y_{ee}, a_1, a_2) = (0.10, 0.50, 1.22, -2.65), (0.30, 0.30, 1.54, -1.98)$  respectively.  
In Figures 4.8(c)-(d),

$$\Phi_1(s) = \sin 2\pi s \quad \Phi_2(s) = s \quad (4.6.7b)$$

and  $(x_{ee}, y_{ee}, a_1, a_2) = (0.10, 0.50, 1.64, 0.38), (0.30, 0.30, 2.12, 1.53)$  respectively.



**Figure 4.8: Modes Without Closed-Form Inverse Kinematics**

## 4.7 Singularities, Degeneracy, and Switching

While the introduction of modes reduces computational needs for robots with large numbers of degrees of freedom, it also introduces additional considerations. For example, “modal” singularities will arise. We will also see that modes cannot be chosen arbitrarily, i.e., it is possible to choose “degenerate” modes where  $N$  modes may not map to  $N$  end-effector degrees of freedom.

### 4.7.1 Modal Singularities

Modal singularities are analogous to the kinematic singularities of standard manipulators. In a kinematic singularity, an instantaneous motion of the joints is unable to provide an instantaneous twist of the end-effector about one or more screw axes in the workspace. For modal singularities, the loss of end-effector freedom is measured with respect to instantaneous changes in modal participation factors, and can be observed as a loss of rank of the modal Jacobian matrix,  $\mathcal{J}(\bar{a})$ . In the case when the number of modes is equal to the number of task constraints, modal singularities are identified by

$$\det(\mathcal{J}) = 0. \quad (4.7.1)$$

It should be noted that in most cases, the singularities of the physical manipulator must be accounted for separately.

### 4.7.2 Singularities of Bending Modes

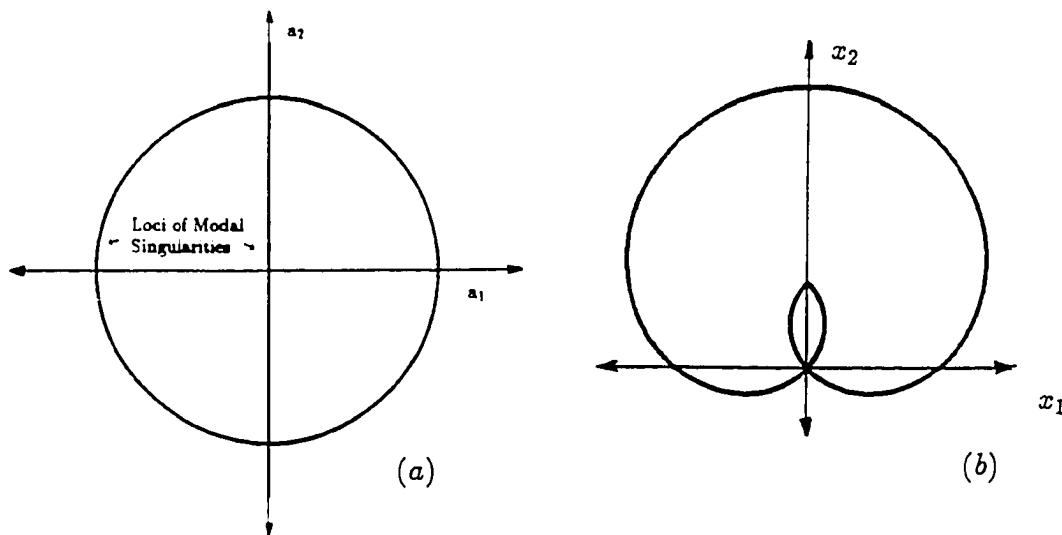
To illustrate modal singularities Equation (4.6.1) is evaluated with the choice of modes given by Equations (4.2.2-4.2.5). Since two participation factors are involved, and we are only interested in end-effector position in the plane, loss of rank can be determined by setting

$$\det(\mathcal{J}) = \frac{\partial x_{ee}}{\partial a_1} \frac{\partial y_{ee}}{\partial a_2} - \frac{\partial x_{ee}}{\partial a_2} \frac{\partial y_{ee}}{\partial a_1} = 0. \quad (4.7.2)$$

Substituting Equation (4.2.3) into Equation (4.7.2), the modal singularities of this manipulator occur when:

$$\frac{a_1}{(a_1^2 + a_2^2)^{\frac{1}{2}}} J_1 \left[ (a_1^2 + a_2^2)^{\frac{1}{2}} \right] J_0 \left[ (a_1^2 + a_2^2)^{\frac{1}{2}} \right] = 0. \quad (4.7.3)$$

Equation (4.7.3) will be satisfied for any one of the following conditions:  $a_1 = 0$ ,  $J_0 \left[ (a_1^2 + a_2^2)^{\frac{1}{2}} \right] = 0$ , or  $J_1 \left[ (a_1^2 + a_2^2)^{\frac{1}{2}} \right] = 0$ . The case of  $a_1 = 0$  corresponds to the workspace boundary, and physically means that for this set of modes the manipulator cannot extend further in a direction normal to the boundary imposed by these modes. The other conditions occur when  $(a_1^2 + a_2^2)^{\frac{1}{2}}$  is a zero of either of the Bessel functions  $J_0$  or  $J_1$ . Since the inverse kinematics solution uses a restricted Bessel function, the only time this happens is when  $(a_1^2 + a_2^2)^{\frac{1}{2}} = \mu_1$  where  $\mu_1 \approx 2.405$  is the first zero of  $J_0$ .



**Figure 4.9: An Example of Bending Mode Singularities**

Figure (4.9)(a) shows the  $a_1$ - $a_2$  modal participation factor space for this choice of modes and the loci of  $a_1$  and  $a_2$  values which lead to modal singularities. Figure (4.9)(b) shows the loci of end-effector positions where modal singularities occur.

### 4.7.3 Singularities of Extension Modes

Unlike the pure bending case, modal singularities do not result from pure extensions (provided the modes are not degenerate, see Section 4.7.4) because of the linearity of the mapping between participation factors and end-effector positions. In other words,  $\mathcal{J}$  in Equation (4.3.5) either has full rank all the time, or none of the time. The workspace boundary in this case is manipulator dependent, but not mode dependent.

### 4.7.4 Degenerate Modes

Care must be exercised in choosing mode functions to avoid *degenerate* choices. The following defines degeneracy of modes:

**Definition:** A choice of modes is *degenerate* if  $\mathcal{J}$  loses rank for all values of modal participation factors. A set of modes can be tested for degeneracy by simply evaluating the rank of the modal Jacobian for a random selection of  $\{a_i\}$ . If the modal Jacobian is full rank for any set of  $\{a_i\}$ , the modes are nondegenerate.

In other words, while modal singularities can occur for any choice of modes for *particular* values of the participation factors, the modes are degenerate only if *all* values of the participation factors result in modal singularities. This is illustrated in the following example.

#### An Example of Degenerate Bending Modes

In this section it is assumed that the manipulator is limited to pure bending in the plane. One choice of bending modes,  $\{\Phi_i\}$ , which is degenerate is the set of modes which are odd, or anti-symmetric, about the point  $s = \frac{1}{2}$ . This oddness implies:

$$\Phi_i(s) = -\Phi_i(1-s) \quad \text{for } i \in [1, N_\theta]. \quad (4.7.4)$$

It follows immediately that

$$\theta(s, t) = -\theta(1 - s, t). \quad (4.7.5)$$

The integral

$$\int_0^1 \sin \theta(s, t) ds \quad (4.7.6)$$

can then be evaluated over two half integrals as follows:

$$\int_0^{\frac{1}{2}} \sin \theta(s, t) ds + \int_{\frac{1}{2}}^1 \sin \theta(s, t) ds, \quad (4.7.7)$$

which can be written as

$$\int_{\frac{1}{2}}^1 \sin \theta(s, t) ds = \int_{\frac{1}{2}}^1 \sin -\theta(1 - s, t) ds. \quad (4.7.8)$$

Making the change of variables  $\sigma = 1 - s$ , and recognizing the oddness of the sine function, and linearity of integration,

$$\int_{\frac{1}{2}}^1 \sin -\theta(1 - s, t) ds = \int_{\frac{1}{2}}^0 \sin \theta(\sigma, t) d\sigma = - \int_0^{\frac{1}{2}} \sin \theta(\sigma, t) d\sigma. \quad (4.7.9)$$

Therefore,

$$\int_0^1 \sin \theta(s, t) ds = 0. \quad (4.7.10)$$

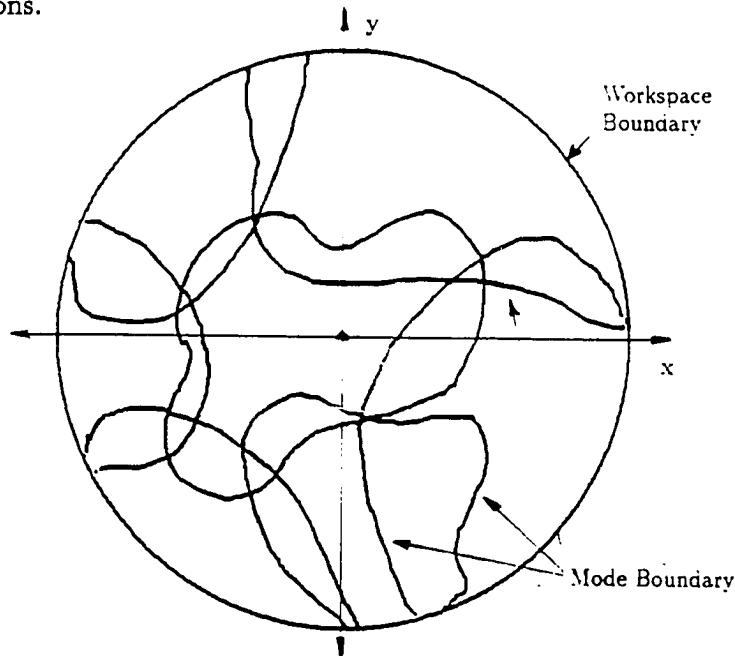
Note that what has been shown are sufficient, but *not* necessary, conditions for degeneracy of modes. This represents a loss in the freedom of the end-effector independent of the value of the participation factors, so this choice of modes is degenerate, i.e., even an infinite number of modes of the form given in Equation (4.7.4) will not allow the end-effector to depart from the  $x_2$ -axis.

### An Example of Degenerate Extension Modes

Any choice of extension modes will be degenerate if  $\theta_e(s)$  is taken to be zero. In this case, the end-effector will be constrained to move along the  $x_2$ -axis. It is easy to determine if a choice of extension modes is degenerate by simply checking the rank of the matrix  $\mathcal{J}$ . If it is not full rank then the modes are degenerate, otherwise the extension modes are nondegenerate and the modal approach does not introduce new singularities.

### 4.7.5 Switching Among Sets of Modes

The introduction of modes may restrict the manipulator to operate in a workspace which is smaller than dictated by physical limitations on the manipulator. However, the potential to switch among several sets of modes allows this approach to cover the workspace of the manipulator. Figure 4.10 shows a schematic picture of a manipulator's physical workspace which is "covered" by different overlapping modal regions.



**Figure 4.10:** Covering the Workspace

An easy way to switch from a set of modes and participation factors  $\{\Phi_i^I, \bar{a}^I\}$  to the set  $\{\Phi_i^O, \bar{a}^O\}$  is to specify a function  $f(t)$  such that  $f(t_0) = 0$  and  $f(t_1) = 1$  where the times  $t = t_0$  and  $t = t_1$  represent the beginning and end of the switching process respectively. In the planar case, the switching process is given by:

$$\theta(s, t) = [1 - f(t)] \sum_{i=1}^{N_\theta} a_i^I \Phi_i^I + f(t) \sum_{i=1}^{N_\theta} a_i^O \Phi_i^O, \quad (4.7.11)$$

with generalizations to the spatial case made by restricting  $\{K, T, R, L\}$  to similar forms. If however the end-effector must remain stationary during the procedure,

the following relation can be used to control the mode switching:

$$\mathcal{J}^I(\bar{a}^I, \bar{a}^O)\dot{\bar{a}}^I + \mathcal{J}^O(\bar{a}^I, \bar{a}^O)\dot{\bar{a}}^O = 0. \quad (4.7.12)$$

$\mathcal{J}^I$  and  $\mathcal{J}^O$  are the Jacobians associated with the original and new participation factors, which are written in component form as

$$\mathcal{J}_{ij}^I = \frac{\partial x_i}{\partial a_j^I} \quad \mathcal{J}_{ij}^O = \frac{\partial x_i}{\partial a_j^O}. \quad (4.7.13)$$

One of the two Jacobians needs to be full rank for Equation (4.7.12) to be useful. By defining  $\bar{a}^O$  such that  $\bar{a}^O \cdot \bar{a}^O$  is an increasing function of time, and inverting  $\mathcal{J}^I$ ,  $\dot{\bar{a}}^I$  can be found. Conversely, by defining  $\bar{a}^I$  such that  $\bar{a}^I \cdot \bar{a}^I$  is a decreasing function of time, and inverting  $\mathcal{J}^O$ ,  $\dot{\bar{a}}^O$  can be found. If both Jacobians are singular, the constraint of constant end-effector position as a function of time must be relaxed, or “transitional” modes can be introduced as an intermediate step to avoid these algorithmic singularities.

## 4.8 Discussion

This chapter has presented methods for resolving hyper-redundancy in both nonextensible and extensible manipulators. Hyper-redundant robots have failed to achieve wide-spread applicability due to inefficiency and ineffectiveness of previous kinematic modeling and motion planning techniques, complex mechanical design, and complexity in the programming of these devices arising from their non-anthropomorphic geometry. The algorithms developed in this chapter are a step toward efficient kinematic control of hyper-redundant robots. The next chapter presents a method for “optimal” hyper-redundancy resolution. Subsequent chapters then develop motion planning algorithms for particular hyper-redundant robot applications. In Chapter 8 these applications are demonstrated with one particular hardware implementation: a planar variable geometry truss.

## Chapter 5

# Optimal Manipulator Configurations

This chapter formulates a method of hyper-redundancy resolution based on the calculus of variations and optimality criteria. An “optimal” configuration is defined as one for which the set of backbone reference frames varies as little as possible between any two values of the curve parameter for given end-effector conditions. In other words, the backbone reference sets which satisfy task constraints while minimizing a measure of bending, twisting, rolling, and local extension/contraction will be referred to as optimal. While standard techniques can be used to optimize the configurations of particular hyper-redundant morphologies, the approach presented here is generally applicable, and becomes computationally more attractive with increasing degree of redundancy.

Section 5.1 considers the nonextensible case. Section 5.2 formulates the optimization of extensible backbone reference sets. Section 5.3 uses the parametrizations of Chapter 2 to generate explicit differential equations, via the calculus of variations, which are solved to find optimal planar configurations. Sections 5.4 and 5.5 generate systems of differential equations for optimal spatial backbone reference sets. Section 5.6 uses the optimality criteria to reparametrize existing curves in an optimal way. The rationale for this is straightforward: backbone reference sets consisting of frames which vary as little as possible along the backbone curve will

result in manipulator configurations which require smaller joint displacements. Section 5.7 compares the optimal configuration based hyper-redundancy resolution techniques presented here with the modal approach of Chapter 4.

## 5.1 Defining Configurations of Least Bending

As in Chapter 2,  $\mathbf{Q}(s, t)$  denotes the orientation matrix of the backbone reference frames which evolve along the backbone curve.  $\dot{\mathbf{Q}}(s, t)$  denotes the derivative of  $\mathbf{Q}(s, t)$  with respect to  $s$ . The optimality criterion in this case is the minimization of the integral of the weighted norm of  $\dot{\mathbf{Q}}(s, t)$  over the backbone curve:

$$I = \frac{1}{2} \int_0^1 \text{tr} \left( \dot{\mathbf{Q}}(s, t) \mathbf{W}_3(s) \dot{\mathbf{Q}}^T(s, t) \right) ds. \quad (5.1.1)$$

$\text{tr}(\mathbf{A})$  denotes the trace of matrix  $\mathbf{A}$ .  $\mathbf{W}_3(s)$  is a  $3 \times 3$  symmetric positive definite weighting matrix. It is assumed that there is no preferred direction of bending, and hereafter  $\mathbf{W}_3(s)$  is restricted to the isotropic form  $\mathbf{W}_3(s) = \alpha(s) \mathbf{I}_3$ , where  $\mathbf{I}_3$  is the  $3 \times 3$  identity matrix. The physical interpretation of Equation (5.1.1) is a measure of total bending and twisting of the backbone reference set.

At  $s = 0$ , the backbone reference frame must coincide with the base frame. At  $s = 1$ , the backbone reference frame must correspond to the desired end-effector orientation,  $\mathbf{Q}_{ee}(t)$ . Thus, the boundary conditions:

$$\mathbf{Q}(0, t) = \mathbf{I}_3; \quad \mathbf{Q}(1, t) = \mathbf{Q}_{ee}(t) \quad (5.1.2)$$

are imposed.

Using the convention established in Chapter 2,  $\bar{u}(s, t) = \mathbf{Q}(s, t) \bar{e}_2$  is the tangent to the curve. Recall that the position to any point on a nonextensible backbone curve is given by:

$$\bar{x}(s, t) = \int_0^s \bar{u}(\sigma, t) d\sigma. \quad (5.1.3)$$

The minimum bending problem can be stated as the minimization of Equation (5.1.1) subject to the isoperimetric constraints  $\bar{x}(1, t) = \bar{x}_{ee}(t)$  (the desired end-effector position) with boundary conditions of Equation (5.1.2). The classical

calculus of variations provides a means for solving this kind of problem. The appendix contains background information on variational calculus and terminology used throughout this chapter.

For the problem at hand, the associated *Lagrangian* is

$$\mathcal{L}(s, t) = \frac{1}{2}\alpha(s)\text{tr}(\dot{\mathbf{Q}}(s, t)\dot{\mathbf{Q}}^T(s, t)) + \bar{\mu}_c(t) \cdot \bar{u}(s, t) \quad (5.1.4)$$

where  $\bar{\mu}_c(t)$  is a vector of undetermined Lagrange multipliers arising from the isoperimetric end-effector constraint  $\bar{x}(1, t) = \bar{x}_{ee}(t)$ . Note that  $\bar{x}(0, t) = \bar{0}$  is automatic from Equation (5.1.3).

For the Frenet-Serret frames of Section 2.1,

$$\frac{1}{2}\text{tr}(\dot{\mathbf{Q}}_{FS}(s, t)\dot{\mathbf{Q}}_{FS}^T(s, t)) = \kappa^2(s, t) + \tau^2(s, t). \quad (5.1.5)$$

Since curvature and torsion tell us how a curve and its intrinsic frames bend and twist, this result meets with intuition, and reinforces confidence in Equation (5.1.1) as a measure of how frames vary along the backbone curve.

## 5.2 Defining Optimal Extensible Configurations

The optimality criterion in the extensible case will include contributions from bending, twisting, roll, and extension/contraction. For this case, the Lagrangian is defined as:

$$\mathcal{L}(s, t) = \frac{1}{2}\alpha(s)\text{tr}(\dot{\mathbf{Q}}(s, t)\dot{\mathbf{Q}}^T(s, t)) + \frac{1}{2}\beta(s)(\dot{L}(s, t) - 1)^2 + \dot{L}(s, t)\bar{\mu}_c(t) \cdot \bar{u}(s, t), \quad (5.2.1)$$

where  $\alpha(s)$  weights the relative cost of rotational change such as bending, twisting, and roll, while  $\beta(s)$  weights translational change such as extension and contraction.

Note that Equation (5.2.1) can also be viewed as a variant on the form:

$$\mathcal{L}(s, t) = \frac{1}{2}\text{tr}(\dot{\mathcal{H}}(s, t)\mathbf{W}_4(s)\dot{\mathcal{H}}^T(s, t)) + l(s, t)\bar{\mu}_c(t) \cdot \bar{u}(s, t) \quad (5.2.2)$$

where  $\mathcal{H}(s, t)$  is the homogeneous transform defined in Equation (2.1.23), and

$$\mathbf{W}_4(s) = \begin{pmatrix} \alpha(s)\mathbf{I}_3 & \bar{0} \\ \bar{0}^T & \beta(s) \end{pmatrix} \quad (5.2.3)$$

is a weighting matrix.

Equation (5.2.1) can be physically motivated in the planar case as follows. Let  $\bar{x}(s, t)$  denote the backbone curve (or centerline) of a planar tube in its distorted shape. Let  $\bar{x}_+$  and  $\bar{x}_-$  denote the respective sides of the planar tube, e.g. the curves parallel to the backbone curve.  $\bar{x}_+$  and  $\bar{x}_-$  are defined as:

$$\begin{aligned} \bar{x}_+(s, t) &= \bar{x}(s, t) + r\bar{n}(s, t) \\ \bar{x}_-(s, t) &= \bar{x}(s, t) - r\bar{n}(s, t) \end{aligned} \quad (5.2.4)$$

where  $\bar{n}(s, t)$  is the planar unit normal vector of  $\bar{x}(s, t)$  at  $s$  and  $r$  is the radius of the tube. A reasonable measure of the local deviation of the tube at a point  $s$  from its nominal configuration is the sum of the squared difference in length between the tube tangents and the length of the nominal reference tangent:

$$\begin{aligned} f(s, t) &= \frac{1}{2} ((||\dot{x}_+|| - 1)^2 + (||\dot{x}_-|| - 1)^2) \\ &= \frac{1}{2} ((||\dot{x} + r\dot{\bar{n}}|| - 1)^2 + (||\dot{x} - r\dot{\bar{n}}|| - 1)^2) \\ &= \frac{1}{2} ((||\bar{u}\dot{L} - r\dot{L}\kappa\bar{u}|| - 1)^2 + (||\bar{u}\dot{L} + r\dot{L}\kappa\bar{u}|| - 1)^2) \\ &= \frac{1}{2} (((\dot{L} - r\dot{L}\kappa)||\bar{u}|| - 1)^2 + ((\dot{L} + r\dot{L}\kappa)||\bar{u}|| - 1)^2) \\ &= (\dot{L} - 1)^2 + (r\dot{L}\kappa)^2 \\ &= (l - 1)^2 + (r\dot{\theta})^2. \end{aligned} \quad (5.2.5)$$

Thus, a measure of the total deviation of the planar tube from its nominal configuration is:

$$I(t) = \frac{1}{2} \int_0^1 f(s, t) ds. \quad (5.2.6)$$

Incorporating the isoperimetric end-effector constraints, the planar Lagrangian corresponding to Equation (5.2.6) is:

$$\mathcal{L} = \frac{1}{2}(l - 1)^2 + \frac{1}{2}(r\dot{\theta})^2 + \mu_1 l \cos \theta + \mu_2 l \sin \theta, \quad (5.2.7)$$

which is a specific case of Equation (5.2.1). The tube radius,  $r$ , is physically motivated by the width of the physical manipulator, and provides for a cost function which is dimensionally homogeneous. Comparing Equations (5.2.1) and (5.2.5),  $\alpha = \frac{1}{2}r^2$ , and  $\beta = 1$ .

The next section formulates equations for the planar case in detail.

## 5.3 Optimal Shape Equations for the Planar Case

This section presents examples of the generalized theory developed in the previous sections for planar configurations. Sections 5.4 and 5.5 present analogous examples for the spatial case.

In the planar case,  $\mathbf{Q}(s, t)$  consists of a rotation, by angle  $\theta(s, t)$ , about the axis normal to the plane. In this case:

$$\frac{1}{2}\text{tr}(\dot{\mathbf{Q}}(s, t)\dot{\mathbf{Q}}^T(s, t)) = \dot{\theta}^2(s, t). \quad (5.3.1)$$

Recall that  $\theta(s, t)$  is the clockwise measured angle between the backbone tangent and the  $x_2$ -axis.

In the nonextensible case,  $\dot{\theta}^2 = \kappa^2$ , and thus we seek to minimize:

$$I = \frac{1}{2} \int_0^1 \alpha \kappa^2 ds. \quad (5.3.2)$$

Minimizing a weighted integral of squared curvature is a problem which has been considered in detail in the mathematics, mechanics, and computer science literature [WilS88,SuLi90]. However, viewing the problem as a hyper-redundancy resolution technique is novel.

The forward kinematic constraint for a nonextensible planar manipulator is:

$$x_{ee} = \int_0^1 \sin \theta ds; \quad y_{ee} = \int_0^1 \cos \theta ds, \quad (5.3.3)$$

which is a degenerate form of the spatial Euler-Angle parametrization with  $T(s, t) = 0$  and  $K(s, t) = \theta(s, t)$ . Thus, the Euler-Lagrange equation is:

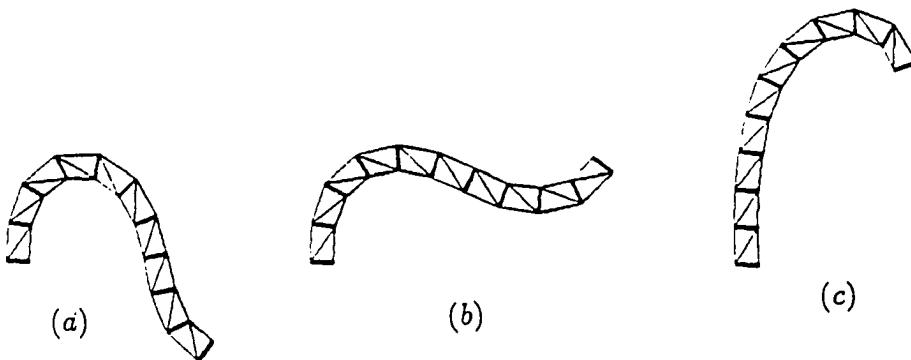
$$\alpha \ddot{\theta} + \dot{\alpha} \dot{\theta} - \mu_1 \cos \theta + \mu_2 \sin \theta = 0. \quad (5.3.4)$$

The solution of this equation can be computed numerically, subject to constraints of Equation (5.3.3), and the boundary conditions  $\theta(0, t) = 0$  and  $\theta(1, t) = \theta_{ee}$ . Numerical solutions for this type of problem can be found in [GruS80].

Let's first consider the case  $\alpha(s) = 1$ ; i.e., curvature is uniformly weighted along the manipulator length. In this case Equation (5.3.4) becomes

$$\ddot{\theta} - \mu_1 \cos \theta + \mu_2 \sin \theta = 0. \quad (5.3.5)$$

Configurations which are a solution to Equation (5.3.5) are shown in Figure 5.1 with a variable geometry truss which has been superimposed, or "fit," to the backbone curve. In Figures 5.1 (a)-(c),  $\dot{\theta}(0, t) = 0$  and  $(x_{ee}, y_{ee}, \mu_1, \mu_2) = (0.50, -0.25, 22.80, 1.87)$ ,  $(0.77, 0.22, 23.87, 14.07)$ ,  $(0.45, 0.51, 2.53, -13.13)$ .

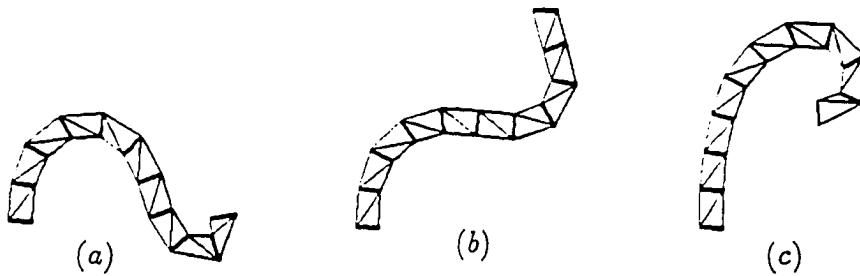


**Figure 5.1:** Uniformly Weighted Planar Nonextensible Configurations

In many practical cases,  $\alpha(s)$  should be defined as a decreasing function, so as to minimize bending at the manipulator base, and allow the distal end to perform a larger range of motion. For instance, the manipulator inertial properties can be approximately incorporated in our optimization by defining

$$\alpha(s) = \alpha_0 + \alpha_1 \int_s^1 \rho(\sigma) d\sigma \quad (5.3.6)$$

where  $\rho(s)$  is the normalized mass density of the manipulator per unit length measured in the nominal reference configuration.  $\int_s^1 \rho d\sigma$  is the normalized mass of the manipulator from point  $s$  to the distal end.  $\alpha_0$  and  $\alpha_1$  weight the relative importance of uniform bending versus inertially weighted bending. This choice of weighting produces configurations as shown in Figure 5.2, for  $\alpha_0 = 0.2$ ,  $\alpha_1 = 1$ , and  $\rho(s, t) = 1$  (uniformly distributed mass). In Figures 5.2 (a)-(c),  $\dot{\theta}(0, t) = 0$  and  $(x_{ee}, y_{ee}, \mu_1, \mu_2) = (0.55, 0.03, 21.67, 7.07)$ ,  $(0.52, 0.58, 23.73, 23.13)$ ,  $(0.29, 0.32, 4.47, -3.07)$ .



**Figure 5.2:** Nonuniformly Weighted Planar Nonextensible Configurations

In the extensible case, Equation (5.2.7) is the Lagrangian, i.e.,  $\alpha(s) = \frac{1}{2}r^2$ ,  $\beta(s) = 1$ . The Euler-Lagrange equations for this case are:

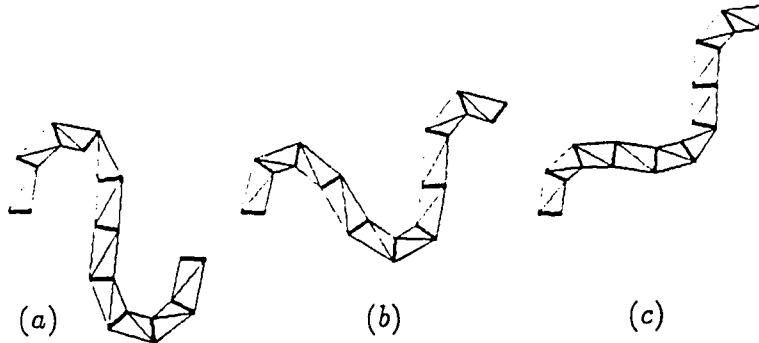
$$r^2 \ddot{\theta} - \mu_1 \dot{L} \cos \theta + \mu_2 \dot{L} \sin \theta = 0 \quad (5.3.7)$$

$$\frac{\partial}{\partial s} (\dot{L} - 1 + \mu_1 \sin \theta + \mu_2 \cos \theta) = 0. \quad (5.3.8)$$

Equation (5.3.8) has the exact first integral:

$$\dot{L}(s, t) + \mu_1(t) \sin \theta(s, t) + \mu_2(t) \cos \theta(s, t) = \dot{L}(0, t) + \mu_1(t) \quad (5.3.9)$$

The boundary conditions for Equations (5.3.7) and (5.3.8) are:  $\theta(0, t) = 0$ ,  $\theta(1, t) = \theta_{ee}(t)$ ,  $L(0, t) = 0$ ,  $L(1, t) = L_0(t)$ . In addition to these boundary conditions,  $\dot{\theta}(0, t)$ ,  $\dot{L}(0, t)$ ,  $\mu_1(t)$ , and  $\mu_2(t)$  specify the end-effector position. Note that optimal configurations for the nonextensible case will generally be different from optimal configurations for the extensible case with the same length and the same boundary conditions. This is because extensible backbone curves can internally reconfigure themselves by redistributing extensibility along the manipulator. The resulting configurations will be different from the nonextensible case where zero extensibility is allowed. Figure 5.3 shows configurations generated by Equations (5.3.7) and (5.3.8) with  $(r, \dot{L}(0, t), \dot{\theta}(0, t)) = (0.1, 1.0, 0.0)$  and  $(x_{ee}, y_{ee}, \theta_{ee}, L_0, \mu_1, \mu_2) = (0.47, -0.13, 0.05, 1.24, 0.38, -0.04)$ ,  $(0.69, 0.26, 2.31, 1.26, 0.64, 0.15)$ ,  $(0.61, 0.51, 1.72, 1.0, 0.63, 0.48)$ .



**Figure 5.3:** Uniformly Weighted Planar Extensible Configurations

## 5.4 Equations for the Spatial Nonextensible Case

This section examines the problem of generating equations which describe optimal nonextensible spatial backbone reference sets. Two different sets of parameters

are used: Euler-Angles and unit quaternions. As will be shown in Section 5.5, the results for the extensible case are similar, with an additional equation using the arc length function  $L(s, t)$  as the generalized coordinate.

### 5.4.1. Parametrization via $K$ , $T$ , and $R$

Using the Euler-Angle description of spatial rotations to parametrize the backbone reference set defined in Equation (2.2.10), the norm defined in Section 5.1 is explicitly computed as:

$$\frac{1}{2}\alpha \text{tr}(\dot{\mathbf{Q}}\dot{\mathbf{Q}}^T) = \alpha \left( \dot{K}^2 + \dot{T}^2 + \dot{R}^2 - 2\dot{K}\dot{R}\sin T \right). \quad (5.4.1)$$

Consequently, with weighting  $\alpha(s) = \frac{1}{2}$ , the constrained Euler-Lagrange equations are:

$$\mathbf{G}(T) \begin{pmatrix} \ddot{K} \\ \ddot{T} \\ \ddot{R} \end{pmatrix} + \bar{\mathbf{S}}(T, \dot{K}, \dot{T}, \dot{R}) + \mathbf{C}(T, K)\bar{\boldsymbol{\mu}}_c = 0, \quad (5.4.2)$$

where  $\bar{\boldsymbol{\mu}}_c = \bar{\boldsymbol{\mu}}_c(t) \in \mathbf{R}^3$  is the vector of undetermined Lagrange multipliers used to account for the end-effector position constraints, and

$$\begin{aligned} \mathbf{G}(T) &= \begin{pmatrix} 1 & 0 & -\sin T \\ 0 & 1 & 0 \\ -\sin T & 0 & 1 \end{pmatrix} \\ \bar{\mathbf{S}}(T, \dot{K}, \dot{T}, \dot{R}) &= \begin{pmatrix} -\dot{T}\dot{R}\cos T \\ \dot{K}\dot{R}\cos T \\ -\dot{K}\dot{T}\cos T \end{pmatrix} \\ \mathbf{C}(T) &= \begin{pmatrix} -\cos K \cos T & \sin K \cos T & 0 \\ \sin K \sin T & \cos K \sin T & -\cos T \\ 0 & 0 & 0 \end{pmatrix}. \end{aligned} \quad (5.4.3)$$

These equations are solved with initial conditions  $K(0, t) = T(0, t) = R(0, t) = 0$ . The final boundary conditions can be determined by equating  $\mathbf{Q}(1, t)$  to the desired end-effector orientation,  $\mathbf{Q}_{ee}(t)$ . One way to satisfy the orientation boundary conditions is:

$$\begin{aligned} T(1, t) &= \sin^{-1}(q_{ee,32}(t)) \\ K(1, t) &= \text{atan2} \left( \frac{q_{ee,12}(t)}{\cos T(1, t)}, \frac{q_{ee,22}(t)}{\cos T(1, t)} \right) \\ R(1, t) &= \text{atan2} \left( -\frac{q_{ee,31}(t)}{\cos T(1, t)}, \frac{q_{ee,33}(t)}{\cos T(1, t)} \right), \end{aligned} \quad (5.4.4)$$

where  $q_{ee,ij}(t)$  is the  $i-j$  element of  $\mathbf{Q}_{ee}(t)$ , which is the desired end-effector orientation matrix at time  $t$ .

However, note that  $\mathbf{G}(T)$  (which must be inverted in numerical solutions of Equation (5.4.2)) will become singular when  $\sin T = 1$ . The next section considers a 4-parameter description of orientation which avoids these singularities.

### 5.4.2. Parametrization via Unit Quaternions

Dynamical equations analogous to those of Section 5.4.1 can be generated with the quaternion representation presented in Section 2.2.3. With  $\alpha(s) = \frac{1}{2}$ , the Euler-Lagrange equations are

$$\mathbf{G}(\bar{\lambda})\ddot{\bar{\lambda}} + \bar{\mathbf{S}}(\bar{\lambda}, \dot{\bar{\lambda}}) + \mathbf{C}(\bar{\lambda})\bar{\mu}_c + 2\mu_v\bar{\lambda} = 0. \quad (5.4.5)$$

$\mathbf{G}(\bar{\lambda})$  is the  $4 \times 4$  matrix with matrix elements:

$$G_{ij} = \text{tr} \left( \frac{\partial \mathbf{R}}{\partial \lambda_i} \frac{\partial \mathbf{R}^T}{\partial \lambda_j} \right) = 4(2\delta_{ij} + \lambda_i \lambda_j). \quad (5.4.6)$$

$\bar{\mathbf{S}}(\bar{\lambda}, \dot{\bar{\lambda}})$  is a  $4 \times 1$  vector whose  $i^{th}$  component is:

$$\mathbf{S}_i = \sum_{j=1}^4 \sum_{k=1}^4 S_{ijk} \dot{\lambda}_j \dot{\lambda}_k, \quad (5.4.7)$$

where

$$\begin{aligned} S_{ijk} &= \text{tr} \left( \frac{\partial \mathbf{R}}{\partial \lambda_i} \frac{\partial^2 \mathbf{R}^T}{\partial \lambda_j \partial \lambda_k} \right) = 2\Lambda_{ijk} \Sigma_{ijk} \quad (\text{no sum}) \\ \Lambda_{ijk} &= \lambda_i \delta_{jk} + \lambda_j \delta_{ik} (1 - \delta_{jk}) (1 - \delta_{ij}) + \lambda_k \delta_{ij} (1 - \delta_{jk}) (1 - \delta_{ik}) \\ \Sigma_{ijk} &= -\delta_{jk} + 2\delta_{ij} + 2\delta_{ik} - \delta_{ij}\delta_{jk} - 2\delta_{ik}\delta_{jk} + 3\delta_{ij}\delta_{jk}\delta_{ik}. \end{aligned} \quad (5.4.8)$$

$\mathbf{C}(\bar{\lambda})$  is the  $4 \times 3$  matrix:

$$2 \begin{pmatrix} -\lambda_4 & \lambda_1 & \lambda_2 \\ \lambda_3 & -\lambda_2 & \lambda_1 \\ \lambda_2 & \lambda_3 & \lambda_4 \\ -\lambda_1 & -\lambda_4 & \lambda_3 \end{pmatrix}. \quad (5.4.9)$$

$\bar{\mu}_c(t)$  is again a  $3 \times 1$  vector of Lagrange multipliers for the end-effector position constraint, while  $\mu_v(s, t)$  is a Lagrange multiplier with functional dependence on the curve parameter required for the constraint  $\bar{\lambda} \cdot \bar{\lambda} = 1$ .

The Euler-Lagrange equations are solved with six free parameters: three Lagrange multipliers for end-effector position  $\bar{\mu}_c$ , and three arbitrary base conditions  $\{\dot{\lambda}_2(0, t), \dot{\lambda}_3(0, t), \dot{\lambda}_4(0, t)\}$ . The restriction that  $\mathbf{Q}(0, t) = \mathbf{I}_3$  requires that:  $\lambda_1(0, t) = 1$ , and  $\lambda_2(0, t) = \lambda_3(0, t) = \lambda_4(0, t) = 0$ . Thus, in order to be consistent with the constraint  $\bar{\lambda} \cdot \bar{\lambda} = 1$ ,  $\dot{\lambda}_1(0, t) = 0$ . Spatial configurations generated from the solution to Equation (5.4.5) are shown in Figures 5.4 (a)-(c) for  $(\mu_1, \mu_2, \mu_3, \lambda_2(0, t), \lambda_3(0, t), \lambda_4(0, t)) = (0.04, 0.00, 0.00, 0.00, 0.00, 0.00)$ ,  $(0.00, -0.34, 0.07, -0.01, -0.03, 0.02)$ ,  $(0.00, -0.34, 0.07, -0.23, -0.03, 0.02)$ . Here again, a variable geometry truss was “fitted” to the optimal backbone curve for clarity.

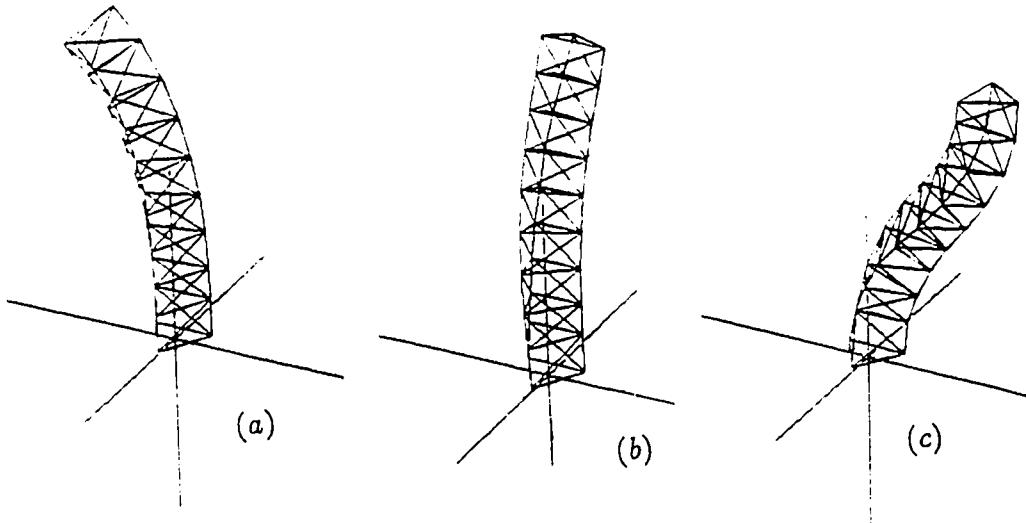


Figure 5.4: Optimal Nonextensible Spatial Configurations

## 5.5 Equations for the Spatial Extensible Case

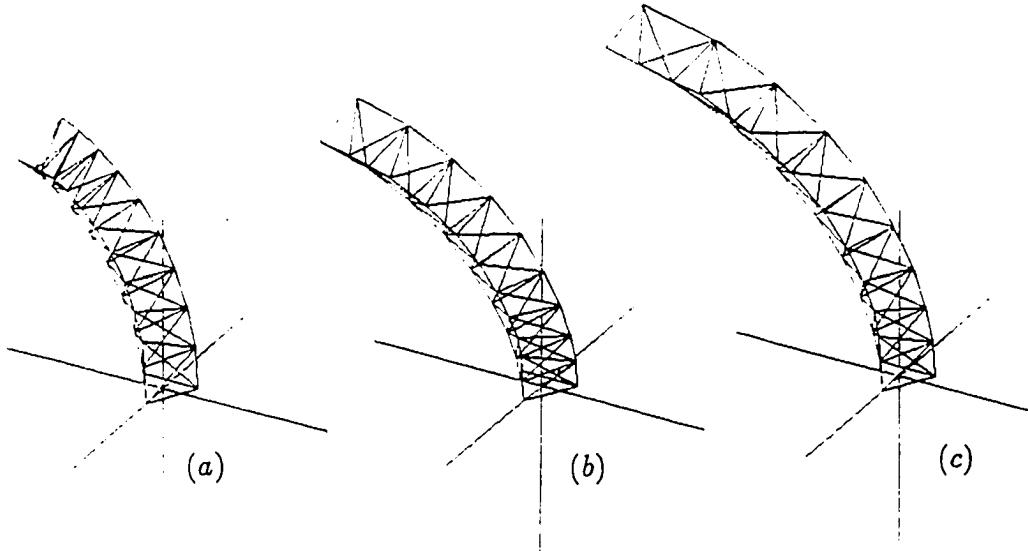
Using the planar case as motivation, the weighting functions  $\alpha(s) = \frac{1}{2}r^2$  and  $\beta(s) = 1$  are again chosen. The Euler-Lagrange equations corresponding to the

Lagrangian in Equation (5.2.1) expressed in terms of Euler-Angles are:

$$\mathbf{G}(T) \begin{pmatrix} \ddot{K} \\ \dot{T} \\ \ddot{R} \end{pmatrix} + \bar{\mathbf{S}}(T, \dot{K}, \dot{T}, \dot{R}) + \frac{\dot{L}}{r^2} \mathbf{C}(T, K) \bar{\mu}_c = 0 \quad (5.5.1a)$$

$$\frac{\partial}{\partial s} (\dot{L} + \bar{\mu}_c \cdot \bar{u}) = 0. \quad (5.5.1b)$$

These equations are solved with initial conditions  $K(0, t) = T(0, t) = R(0, t) = L(s, t) = 0$ . The final boundary conditions are:  $\mathbf{Q}(1, t) = \mathbf{Q}_{ee}(t)$  (or equivalently, Equation (5.4.4)),  $\bar{x}(1, t) = \bar{x}_{ee}(t)$ , and  $L(1, t) = L_0(t)$ .



**Figure 5.5:** Optimal Extensible Spatial Configurations

Similarly, the unit quaternion parametrization is used to yield the set of five equations:

$$\mathbf{G}(\bar{\lambda}) \ddot{\bar{\lambda}} + \bar{\mathbf{S}}(\bar{\lambda}, \dot{\bar{\lambda}}) + \frac{\dot{L}}{r^2} \mathbf{C}(\bar{\lambda}) \bar{\mu}_c + 2 \frac{\mu_v}{r^2} \bar{\lambda} = 0 \quad (5.5.2a)$$

$$\frac{\partial}{\partial s} (\dot{L} + \bar{\mu}_c \cdot \bar{u}) = 0 \quad (5.5.2b)$$

Just as in Section 5.4,  $\mu_c(t)$  is a  $3 \times 1$  vector of Lagrange multipliers for the end-effector position constraint, while  $\mu_v(s, t)$  is a Lagrange multiplier required for the constraint  $\bar{\lambda} \cdot \bar{\lambda} = 1$ . The boundary conditions are the same for this

problem as for the nonextensible case, with the one additional constraint on manipulator length  $L(1, t) = L_0(t)$ . Figures 5.5(a)-(c) show optimal extensible configurations for  $r = 10$  (which is exaggerated to illustrate the effects of extensibility),  $(\dot{\lambda}_2(0, t), \dot{\lambda}_3(0, t), \dot{\lambda}_4(0, t)) = (0.0, 0.0, 0.7)$ , and  $(\mu_1, \mu_2, \mu_3, \dot{L}(0, t)) = (0.2, -1.2, 0.6, 1.0), (0.6, 0.6, 0.6, 0, 0.5), (0.2, 1.6, 0.6, 0.8)$ .

## 5.6. Optimal Backbone Reparametrization

This section considers several related problems regarding the reparametrization of existing backbone reference sets in an “optimal” way. The motivation for optimal reparametrization is as follows. Often nonoptimal backbone curve shapes can be defined to perform useful tasks. The modal approach is an example of this. It is then easy to modify the parametrization of the original curve, and the roll distribution about the curve, so as to make the backbone reference set evolve as uniformly as possible for a given backbone curve.

In this section, the reparametrization of planar backbone curves is first considered. Next, the optimal roll distribution for spatial configurations is determined so that the backbone reference frames are twisting as little as possible about a given backbone curve for specified end-effector roll. Finally, it will be shown how optimal reparametrization of spatial backbone reference sets is expressed as first integrals in the length function  $L(s, t)$ , and roll distribution  $R(s, t)$ . In all of these cases, the distribution of backbone reference frames will vary as little as possible from one value of  $s$  to another according to the particular cost function.

### 5.6.1. Optimal Planar Curve Reparametrization

Let  $\bar{y}^*(\phi, t)$  be a time-dependent planar curve segment with curve parameter  $\phi \in [0, \phi_0]$ . We wish to find an alternate parameter,  $s$ , such that  $\bar{y}(s, t) = \bar{y}^*(\phi(s, t), t)$  has the same shape as  $\bar{y}^*(\phi, t)$ , but that the distribution of frames along the curve vary as little as possible from one value of  $s$  to another. In other words,

we wish to find a new parametrization which minimizes Equation (5.2.6). This reparametrization can be achieved as follows.

Define the angle which the tangent vector  $\bar{y}^*$ ' (where a' indicates differentiation with respect to  $\phi$ ) makes with the  $x_2$  coordinate axis as:

$$\theta^*(\phi, t) = \text{Atan2}(y_1^{*'}, y_2^{*'}), \quad (5.6.1)$$

where  $y_i^*$  for  $i \in [1, 2]$  are the components of  $\bar{y}^*$ . Equate the parametrizations in  $\phi$  and  $s$ :

$$\theta(s, t) = \theta^*(\phi(s, t), t). \quad (5.6.2)$$

Similarly, the rate of length function in the new parameter  $s$  must be:

$$l(s, t) = (\bar{y}^* \cdot \bar{y}^*)^{\frac{1}{2}} \dot{\phi}. \quad (5.6.3)$$

The goal is to find a new parametrization which minimizes Equation (5.2.6), which in this case is equivalent to minimizing

$$\frac{1}{2} \int_0^1 f(s, t) ds = \frac{1}{2} \int_0^1 (r^2 \dot{\theta}^2 + l^2) ds.$$

Written in terms of the initial parametrization,

$$f(s, t) = \frac{1}{2} \left( r^2 \left( \frac{\partial \theta^*}{\partial \phi} \right)^2 \dot{\phi}^2 + \left( \frac{\partial \bar{y}^*}{\partial \phi} \cdot \frac{\partial \bar{y}^*}{\partial \phi} \right) \dot{\phi}^2 \right) \quad (5.6.4)$$

which can be expressed in component form as

$$f(s, t) = \frac{\dot{\phi}^2}{2} \left[ (y_1')^2 + (y_2')^2 + r^2 \frac{y_1'' y_2' - y_1' y_2''}{(y_1')^2 + (y_2')^2} \right] = \dot{\phi}^2 g(\phi, t). \quad (5.6.5)$$

The superscript \* has been temporarily suppressed.

The Euler-Lagrange equations are simply:

$$2\ddot{\phi}g + \dot{\phi}^2 \frac{\partial g}{\partial \phi} = 0. \quad (5.6.6a)$$

Multiplying Equation (5.6.6a) by  $\dot{\phi}$  yields the exact differential:

$$\frac{\partial}{\partial s} (\dot{\phi}^2 g) = 0. \quad (5.6.6b)$$

Integrating with respect to  $s$ , dividing by  $g(\phi, t)$ , and taking the square root of what results,

$$\dot{\phi} = \frac{\partial \phi}{\partial s} = c_1(t) g^{-\frac{1}{2}}(\phi, t) \quad (5.6.7a)$$

where  $c_1(t)$  is an arbitrary function of time. Solving for  $s$  as a function of  $\phi$  and  $t$ ,

$$s(\phi, t) = \frac{1}{c_1(t)} \int_0^\phi g^{\frac{1}{2}}(\sigma, t) d\sigma + c_2(t). \quad (5.6.7b)$$

Since both parametrizations are defined such that  $s = 0$  when  $\phi = 0$  for all  $t$  (which means that both parametrizations begin at the origin of the manipulator base frame) then  $c_2 = 0$ . Likewise, end-effector conditions are matched recognizing that at the point on the curve where  $\phi = \phi_0$ ,  $s = 1$ . Using this condition with Equation (5.6.7b),  $c_1 = \int_0^{\phi_0(t)} g^{\frac{1}{2}}(\sigma, t) d\sigma$ , and so

$$s(\phi, t) = \frac{\int_0^\phi g^{\frac{1}{2}}(\nu, t) d\nu}{\int_0^{\phi_0} g^{\frac{1}{2}}(\nu, t) d\nu} \quad (5.6.8)$$

where

$$s(\cdot, t) = \phi^{-1}(\cdot, t). \quad (5.6.9)$$

When  $r = 0$ , the optimal reparametrization is the classical arc-length parametrization.

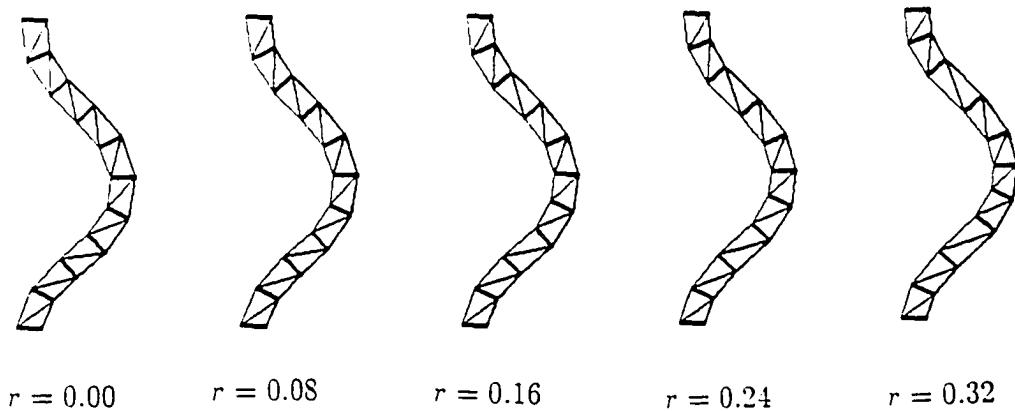
This is easily seen by considering a planar arc-length parametrized curve  $y(L, t)$ . The optimal reparametrization based on minimizing Equation (5.2.6) is:

$$s(L, t) = \frac{\int_0^L (1 + r^2 \kappa^2(\nu, t))^{\frac{1}{2}} d\nu}{\int_0^{L_0} (1 + r^2 \kappa^2(\nu, t))^{\frac{1}{2}} d\nu}. \quad (5.6.10)$$

For example, assume a nonextensible backbone curve shape which satisfies task constraints has been found using bending modes as in Chapter 4. For fitting purposes, the backbone curve can be reparametrized with a rate of length function of the form:

$$l(s, t) = \dot{L}(s, t) = 1 / \frac{\partial s}{\partial L} = \frac{\int_0^{L_0} (1 + r^2 \kappa^2(\nu, t))^{\frac{1}{2}} d\nu}{(1 + r^2 \kappa^2(L(s, t), t))^{\frac{1}{2}}}. \quad (5.6.11)$$

This procedure is illustrated in Figure 5.6 with curves whose shape was specified using the modes in Equation (4.2.2). A variable geometry truss is fitted to the backbone curve using the spacing between truss modules defined by Equation (5.6.11). While the backbone curve was initially assumed to be nonextendible, the variable geometry truss is extendible, and thus the truss modules can locally extend. With  $r = 0$ , the truss modules are uniformly spaced along the backbone curve. As  $r$  increases, the spacing of the truss modules becomes increasingly nonuniform for high curvature backbone curve segments. In effect, this spacing compresses modules to high curvature segments, and stretches modules over low curvature segments. A value of  $r = 0.03$  (half the truss width) would be chosen based on the deformed tube model discussed in Section 5.2. In general, the effect of reparametrization is rather minor in the case of slender manipulators with backbone curves initially parametrized by arc length.



**Figure 5.6:** Optimal Reparametrization of Planar Curves

### 5.6.2. Optimal Roll Distribution (Nonextensible Case)

This section considers the problem: given a backbone curve with  $L(s, t) = s$ , find

the roll distribution which provides the smallest “twist” about the backbone per unit curve parameter (which in this case is arc length).

Here the backbone reference set parametrization  $\{K, T, R, L\}$  is used, with  $L(s, t) = s$ .  $K = K^*(L, t)$  and  $T = T^*(L, t)$  are given functions. The goal is to determine  $R(L, t)$  which minimizes the integral in Equation (5.1.1) for given  $K$  and  $T$ . The bending criterion is:

$$\frac{1}{2} \text{tr}(\dot{\mathbf{Q}} \dot{\mathbf{Q}}^T) = \left( \frac{\partial K^*}{\partial L} \right)^2 + \left( \frac{\partial T^*}{\partial L} \right)^2 + \dot{R}^2 - 2\dot{R} \frac{\partial K^*}{\partial L} \sin T^*. \quad (5.6.12)$$

The Euler-Lagrange equation for this problem is trivial, with  $R$  as the only generalized coordinate:

$$\frac{\partial}{\partial s} \left( \dot{R}(s, t) - \dot{K}^*(s, t) \sin T^*(s, t) \right) = 0. \quad (5.6.13)$$

This equation has the solution:

$$R(s, t) = \dot{R}(0, t)s + \int_0^s \dot{K}^*(\sigma, t) \sin T^*(\sigma, t) d\sigma \quad (5.6.14)$$

where  $\dot{R}(0, t)$  is selected so that the constraint on  $R(1, t)$  is satisfied, i.e.,

$$\dot{R}(0, t) = R_{ee}(t) - \int_0^1 \dot{K}^*(s, t) \sin T^*(s, t) ds. \quad (5.6.15)$$

This problem could also be solved using the modified Frenet-Serret Apparatus from Chapter 1. Since the trace of a matrix is invariant with respect to coordinate transformations:

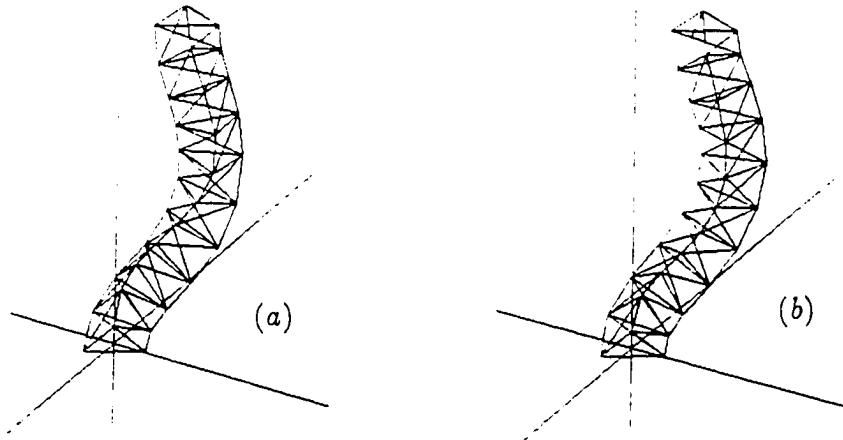
$$\begin{aligned} \text{tr}(\dot{\mathbf{Q}}_M \dot{\mathbf{Q}}_M^T) &= \text{tr}(\mathbf{Q}_M \Lambda_M (\mathbf{Q}_M \Lambda_M)^T) = -\text{tr}(\mathbf{Q}_M \Lambda_M \Lambda_M \mathbf{Q}_M^T) \\ &= -\text{tr}(\Lambda_M \Lambda_M) = 2(\tau + \rho)^2 + 2\kappa^2. \end{aligned} \quad (5.6.16)$$

Using the Euler-Lagrange equations with  $R_{FS}$  as the generalized coordinate, the optimal roll distribution is

$$R_{FS}(s, t) = R_{FS}(0, t) + \dot{R}_{FS}(0, t)s - \int_0^s [\tau(\sigma, t) - \tau(0, t)] d\sigma. \quad (5.6.17)$$

Independent of the parametrization used, the end result is the same: a backbone reference set of minimal roll distribution is generated.

Figure 5.7 shows configurations with  $K^*(s, t) = a_1(t) \sin 2\pi s$ ,  $T^*(s, t) = a_2(t)(1 - \cos 2\pi s)$ , and  $R_{ee}(t) = 0$ . In both figures  $(a_1, a_2) = (0.6, -0.4)$ . In Figure 5.7(a),  $R(s, t) = 0$ , whereas in Figure 5.7(b),  $R(s, t)$  is defined optimally, as in Equations (5.6.14-15).



**Figure 5.7:** Specified Backbone Curve with Optimal Roll Distribution

### 5.6.3 Optimal Spatial Curve Reparametrization

This section addresses the problem of defining the function  $L = L(s, t)$  such that the backbone reference frames vary as little as possible from one value of  $s$  to another for specified functions  $K^*(L, t)$ ,  $T^*(L, t)$ , and  $R^*(L, t)$ , and appropriate end-effector constraints.

This is achieved in the same way as the planar case. The goal is to minimize

$$\frac{1}{2} \int_0^1 \left\{ \frac{1}{2} r^2 \text{tr}(\dot{\mathbf{Q}} \dot{\mathbf{Q}}^T) + \dot{L}^2 \right\} ds. \quad (5.6.18)$$

When parametrizing with the Euler-Angles:

$$\frac{1}{2} \text{tr}(\dot{\mathbf{Q}} \dot{\mathbf{Q}}^T) = \left( \frac{\partial K^*}{\partial L} \right)^2 \dot{L}^2 + \left( \frac{\partial T^*}{\partial L} \right)^2 \dot{L}^2 + \left( \frac{\partial R^*}{\partial L} \right)^2 \dot{L}^2 - 2 \frac{\partial R^*}{\partial L} \frac{\partial K^*}{\partial L} \dot{L}^2 \sin T^*. \quad (5.6.19)$$

A function denoted  $h(L, t)$  can then be defined so that the integrand of Equation (5.6.18) assumes the form:

$$\frac{1}{2} r^2 \text{tr}(\dot{\mathbf{Q}} \dot{\mathbf{Q}}^T) + \dot{L}^2 = h(L, t) \dot{L}^2. \quad (5.6.20)$$

This is the same form as Equation (5.6.5), and so the Euler-Lagrange equations have a solution of the same form as Equation (5.6.8), with  $h(\cdot)$  replacing  $g(\cdot)$ .

#### 5.6.4 Optimal Roll and Length Distributions

This section considers the problem: given a backbone curve, what is the combined roll distribution,  $R(s, t)$ , and length distribution,  $L(s, t)$ , which provide the least varying backbone reference set for given curve geometry. Again,  $K^*(L, t)$  and  $T^*(L, t)$  are specified functions. The goal is to determine  $R(s, t)$ , and  $L(s, t)$  which minimizes the integral

$$\frac{1}{2} \int_0^1 \left\{ \frac{1}{2} r^2 \text{tr}(\dot{\mathbf{Q}} \dot{\mathbf{Q}}^T) + \dot{L}^2 \right\} ds. \quad (5.6.21)$$

Again using the Euler-Angle parametrization,

$$\frac{1}{2} \text{tr}(\dot{\mathbf{Q}} \dot{\mathbf{Q}}^T) = \left( \frac{\partial K^*}{\partial L} \right)^2 \dot{L}^2 + \left( \frac{\partial T^*}{\partial L} \right)^2 \dot{L}^2 + \dot{R}^2 - 2 \dot{R} \frac{\partial K^*}{\partial L} \dot{L} \sin T^*. \quad (5.6.22)$$

This problem must be solved with two Euler-Lagrange equations with  $R$  and  $L$  as generalized coordinates. The solution to the “ $R$ ”- equation is essentially the same as Equation (5.6.14):

$$R(s, t) = \dot{R}(0, t)s + \int_0^s \frac{\partial K^*}{\partial L}(L(\sigma, t), t) \dot{L}(\sigma, t) \sin T^*(L(\sigma, t), t) d\sigma \quad (5.6.23)$$

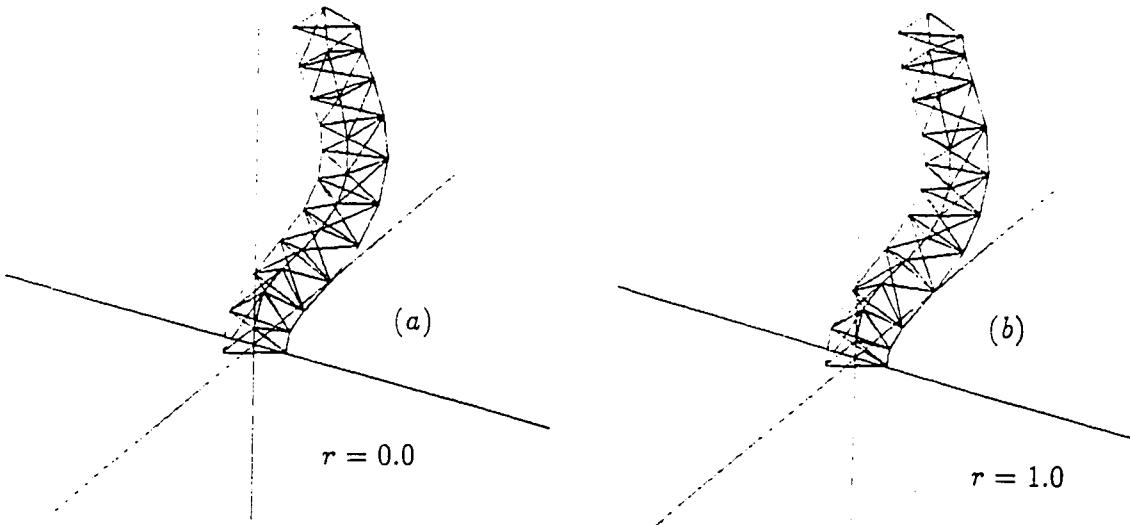
but now  $L(s, t) \neq s$ . Substituting this result back into Equation (5.2.22), and solving the Euler-Lagrange equation with “ $L$ ” as the generalized coordinate,

$$\dot{L}(s, t) = \frac{\int_0^{L_0} (1 + r^2 \kappa^2(\nu, t))^{\frac{1}{2}} d\nu}{(1 + r^2 \kappa^2(L(s, t), t))^{\frac{1}{2}}}, \quad (5.6.24)$$

which is the same as Equation (5.6.11) with

$$\kappa^2(L, t) = \left( \frac{\partial T^*}{\partial L} \right)^2 + \left( \frac{\partial K^*}{\partial L} \right)^2 \cos^2 T^*. \quad (5.6.25)$$

Recall that this is the definition of curvature in Equation (2.2.16). Thus, the criteria for combined optimal roll distribution and extensibility have a similar form to the two problems considered separately. Figure 5.8 shows the same configurations as Figure 5.7, but now optimal extensibility has been included. The value  $r = 1.0$  was chosen in Figure 5.8(b) to exaggerate the effects of the redistribution of frames along the backbone curve.



**Figure 5.8:** Optimal Roll and Extensibility for a Given Backbone Reference Set

## 5.7 Comparisons with the Modal Approach

In the previous chapter, the modal approach was used to compute backbone reference set inverse kinematic solutions which satisfy task constraints. Recall that in the modal approach, the physically meaningful functions  $T, K, R$ , and  $L$  are constrained to the modal form given by Equations (4.5.1-4). The mode functions are chosen by the user, and their number will depend upon the number of end-effector or other task constraints which are specified. With the restriction of the parameters  $\{K, T, R, L\}$  to a modal form, the backbone reference set geometry becomes strictly a function of the modal participation factors. Inverse kinematics reduces to the search for the proper values of the modal participation factors which satisfy task constraints.

This approach has the useful engineering property that it often leads to closed form or nearly closed form inverse kinematic solutions which can be computed with great efficiency. However, the resulting backbone curve shapes are not necessarily optimal in any sense. In this section, the relationship between the modal approach presented in the previous chapter and the calculus of variations approach presented in the preceding sections of this chapter are examined.

The modal and optimal configuration approaches can be combined. For example, one could define a set of basis modes,  $\phi_i$  for  $i \in [1, n]$  and use standard numerical methods to find the modal participation factors which best satisfy a given optimality criterion. This type of approach is common in the solution of variational problems in mechanics, and it is called the *Rayleigh-Ritz* method [HouV83]. One would expect that as the number of modes chosen becomes large, the configuration would approach the solution which can be found from the calculus of variations. For instance, optimal pure bending configurations could be approximated using a large number of orthonormal modes as

$$\kappa(s, t) = \sum_{i=1}^n a_i(t) \phi_i(s). \quad (5.7.1)$$

In this case, the modes would be chosen to be orthogonal with respect to the weighting of the optimality criterion  $\alpha(s)$ , so that

$$\int_0^1 \alpha(s) \phi_i(s) \phi_j(s) ds = \delta_{ij} \quad (5.7.2a)$$

where  $\delta_{ij}$  is the Kronecker delta function. In this case, Equation (5.3.2) becomes

$$\frac{1}{2} \int_0^1 \alpha \kappa^2 ds = \frac{1}{2} \sum_{i=1}^n a_i^2 = \frac{1}{2} \bar{a} \cdot \bar{a}. \quad (5.7.2b)$$

A constrained optimization problem of the form

$$f(\bar{a}, \bar{\mu}_c) = \bar{a} \cdot \bar{a} + [\bar{x}(1, \bar{a}) - \bar{x}_{ee}] \cdot \bar{\mu}_c \quad (5.7.3a)$$

can be written. The constrained optimization problem involves solving the equations  $\partial f / \partial a_i = 0$  and  $\partial f / \partial \mu_j$  for  $(i, j) \in [1, n] \times [1, m]$ , where  $n$  is the number of modes, and  $m$  is the number of end-effector coordinates. This is written as the  $n + m$  equations:

$$\bar{a} + \mathcal{J}^T(\bar{a})\bar{\mu}_c = \bar{0} \quad \text{and} \quad \bar{x}_{ee} = \bar{x}(1, \bar{a}). \quad (5.7.3b)$$

The modal participation factors can be found using standard numerical optimization techniques [GiMW81]. One method would be to take the time derivative of Equations (5.7.3b) and use a resolved rate technique similar to the extended Jacobian technique discussed in Section 1.2.

Similar ideas have been used in other investigations. In [Tav90] the case of a planar variable geometry truss manipulators was considered. They chose a particular set of curvature modes and found the set of participation factors which yielded “optimal” configurations according to a criterion they defined (although the terms “mode” and “participation factor” were not used).

Another connection between the modal and optimality condition based approaches arises when particular optimality criteria are used. Let  $\bar{\theta}(s, t)$  denote the angle  $\theta(s, t)$  and its first  $N - 1$  derivatives:

$$\bar{\theta} = [\theta, \theta^{(1)}, \dots, \theta^{(N-1)}]. \quad (5.7.4)$$

One can find an optimal configuration for cost functions of the form:

$$I = \frac{1}{2} \int_0^1 \bar{\theta}^T(s, t) \mathbf{W}(s) \bar{\theta}(s, t) ds \quad (5.7.5)$$

for a symmetric positive semi-definite matrix  $\mathbf{W}(s) \in \mathbb{R}^N$ , by using the Euler-Lagrange equations. In this case, the Euler-Lagrange equations will be linear differential equations of order  $2N - 2$  whose solutions can be superposed. That is, any solution can be represented as a sum of modes weighted by participation factors. These solutions can be loosely called “optimal modes.”

As an example, let the integral which is to be minimized take the form:

$$I = \frac{1}{2} \int_0^1 (\ddot{\theta}^2 + \omega^2 \dot{\theta}^2) ds. \quad (5.7.6)$$

The associated Euler-Lagrange equation is:

$$\theta^{(4)} - \omega^2 \theta^{(2)} = 0. \quad (5.7.7)$$

A solution to Equation (5.7.7) is

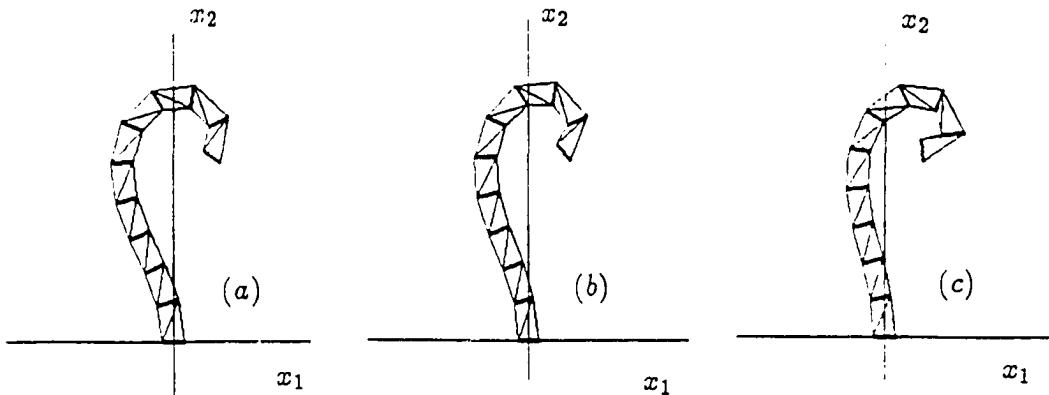
$$\theta(s, t) = a_1(t)s + a_2(t)(e^{\omega s} - 1) + a_3(t)(e^{-\omega s} - 1), \quad (5.7.8)$$

where the convention  $\theta(0, t) = 0$  has been incorporated. The participation factors,  $\{a_i\}$ , can be determined as in Chapter 4 to satisfy the end-effector constraints on position and orientation. Thus, solutions which use the modal approach and the three modes:

$$\begin{aligned} \Phi_1 &= s \\ \Phi_2 &= e^{\omega s} - 1 \\ \Phi_3 &= e^{-\omega s} - 1 \end{aligned} \quad (5.7.9)$$

result in planar backbone configurations which minimize Equation (5.7.6) and satisfy position and orientation constraints. Figure 5.9 shows configurations defined

by Equation (5.7.8) with  $a_1 = 0$ , and  $(x_{ee}, y_{ee}) = (0.1, 0.5)$ . In Figures 5.9(a)-(c),  $\omega = 1, 2, 3$ , respectively.



**Figure 5.9:** Configurations Defined by Optimal Modes

## 5.8 Discussion

This chapter presented methods for determining “optimal” configurations of hyper-redundant manipulators operating in unconstrained environments. These methods were based on the use of a continuous backbone reference set to capture essential macroscopic geometric properties of hyper-redundant manipulators in combination with the calculus of variations to compute the optimal backbone geometry.

This chapter focused on an optimality criterion which minimized a weighted sum of backbone curve bending and extension. However, other optimality criteria can be developed and treated in an analogous manner. In addition to shape optimization, means were developed to optimally reparametrize backbone curves and optimally distribute the backbone reference frame roll about a spatial backbone curve. These problems arise when fitting discretely segmented or modular hyper-redundant mechanisms to continuous backbone curves. Initial backbone curve shapes might

arise from optimal, modal, or spline-based inverse kinematic solution algorithms.

The relationship between the modal and optimal approaches was briefly considered. The advantage of the modal approach lies in the greater user control over manipulator shape through the choice of mode, or shape, functions. The “optimal” approach requires intuition in defining an appropriate cost function. The inverse kinematic solutions in both cases are cyclic when a particular branch of the solution is specified because manipulator configurations are determined by a reduced set of variables with the same dimension as the workspace. In the modal approach, the reduced set is the set of participation factors. In the optimal configuration approach, the Lagrange multipliers and initial conditions form the reduced set. These methods are desirable computationally because a look-up table or neural networks can be used to store the mapping between the reduced set and end-effector boundary conditions. Both approaches avoid computations whose complexity scales as the number of manipulator degrees of freedom.

## Chapter 6

# Hyper-Redundant Obstacle Avoidance

The hyper-redundant robot obstacle avoidance problem can be stated as follows: Given a field of obstacles which clutter the workspace, specify robot motions to avoid obstacles and perform useful tasks. Such tasks may include inspection, retrieval, or placement of objects in highly constrained environments.

The methods presented in this chapter are applicable to both stationary and moving obstacle fields. It is assumed that the layout and motion of obstacles in the workspace are well known, such as in an industrial setting, or that a sufficiently accurate sensing system is available, e.g., vision or proximity sensors. Executing a task in a field of obstacles is equated to defining a path around obstacles to which the robot must adhere. Such a path, as illustrated in Figure 6.1, provides a trajectory or “tunnel” in which a hyper-redundant robot can “slither” to circumvent the obstacles.

In the past, a variety of obstacle avoidance algorithms have been developed for robotic manipulators and mobile robots [Bail86,Kha86,BarrLL89]. Computational requirements for standard obstacle avoidance methods developed for manipulators grow dramatically with the number of manipulator degrees of freedom, and are therefore undesirable for practical use with hyper-redundant manipulators. In practice, an automatic means for selecting one or more feasible tunnels which

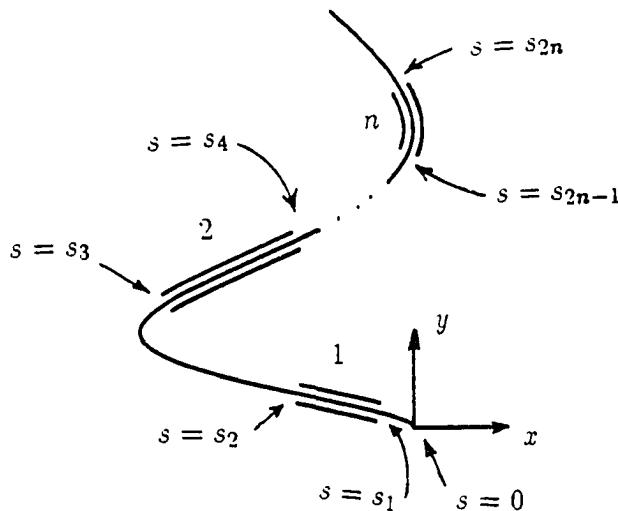
allow hyper-redundant manipulators to successfully negotiate obstacle fields could be generated using previously published methods developed for mobile robots. For instance, free-space methods [Broo83] based on generalized cones could be used to identify free path segments which could be assembled together to form the tunnels required by the algorithm presented here. Once the end-effector has passed through the obstacle field, the portion of a manipulator in the obstacle field can remain stationary while the unconstrained portion of the manipulator performs useful work. The methods presented in Chapters 4 and 5 can be used to control the geometry of the unconstrained manipulator segment outside of the obstacle field. Computational dependence on the number of manipulator degrees of freedom is limited to the fitting procedure used. This is a result of the fact that hyper-redundant manipulators have the ability to conform to prespecified curves, i.e., trajectories generated by algorithms designed for mobile robots can be used as the backbone curves for hyper-redundant manipulators.

In this chapter, two problems are solved to implement this obstacle avoidance scheme. First the proper geometrical constraints on the manipulator are determined so that it statically conforms to the tunnel constraints. This is accomplished by determining the appropriate functions  $\{L, T, K, R\}$  for those sections confined to tunnels, and defining compatibility equations for the free sections. Second the proper time rate of change of the manipulator configuration is computed. In this way the manipulator can "slither" through the tunnels from its starting configuration to its final configuration while obeying all of the geometric constraints. Several compelling reasons exist to investigate the rate problem. First, it is not always possible to find closed form inverse kinematic solutions for the positions of free manipulator sections. Second, the motions of some obstacles may be non-holonomic, e.g., the obstacle field could be on a moving truck with nonholonomic wheel constraints while the manipulator is on another moving platform. A third reason for formulating the velocity problem is to better deal with control algorithms which require rate information. No particular actuation scheme is assumed

or precluded. Because both position and rate information is computed, the current formulation is compatible with most actuator servo control algorithms.

This chapter is organized as follows. Section 6.1 develops an obstacle avoidance algorithm for nonextensible hyper-redundant manipulators. Section 6.2 generalizes this to the extensible case. Section 6.3 demonstrates these algorithms with a detailed example. Section 6.4 illustrates a method in which a human operator can specify obstacle avoiding paths in the planar case. Section 6.5 discusses future applications.

## 6.1 Nonextensible Backbone Tunnel Constraints



**Figure 6.1:** A Backbone Curve Constrained to a System of Tunnels

Figure 6.1 shows a hyper-redundant manipulator backbone curve in which certain segments of its length are constrained to pass through tunnel segments in order to avoid obstacles. Let the segments which are constrained to fit inside a tunnel be termed *interior* segments, while the unconstrained segments are termed *exte-*

*prior* segments. Number the segments sequentially starting from the base of the manipulator, assuming that the first segment is always an exterior segment. The interior and exterior segments will respectively have even and odd indices. The following subsections develop constraint equations on backbone curves to ensure that the tunnels are adhered to. The planar case is considered first.

### 6.1.1 Planar Tunnel Constraints

To begin the analysis the planar nonextensible case is examined. The next section generalizes these results to the spatial case. Curvature functions which will satisfy the section by section constraints have the form:

$$\kappa(s, t) = \sum_{i=1}^{2n} \kappa_i(s, t) W(s, s_{i-1}, s_i) \quad (6.1.1)$$

where  $s_0 = 0$  and  $s_i = s_i(t)$  for  $i > 0$ , and  $i$  indexes the manipulator segments.  $\kappa_i(s, t)$  is a *local curvature function*. Recall the definition of a window function:

$$W(s, s_{i-1}, s_i) = H(s - s_{i-1}) - H(s - s_i), \quad (6.1.2)$$

where  $H$  is the Heaviside, or unit, step function. The curvature function defined in Equation (6.1.1) is therefore a piecewise continuous function, where each segment of the manipulator is assigned a different curvature function to satisfy its local constraints. For interior segments the curvature function, termed an *interior* local curvature function, must take the form of a traveling wave:

$$\kappa_{2i} = \kappa_{2i}(s - s_{2i-1}(t)), \quad (6.1.3)$$

where  $\kappa_{2i}(\cdot)$  is the curvature function for the  $i^{\text{th}}$  tunnel. This can be explained as follows: As sections of the backbone enter tunnels, the shape of the entering sections of the backbone curve will assume the same shape as the tunnels. Likewise, as sections of the backbone exit tunnels, the shape of the exiting sections of the backbone curve will be free to change shape. Thus the shape of the tunnel propagates along the manipulator as it moves through the tunnel.

The curvature function describing exterior segments (an *exterior* local curvature function) can have the more general form:

$$\kappa_{2i+1} = \kappa_{2i+1}(s, t). \quad (6.1.4)$$

However, the exterior segments have kinematic restrictions on their relative position and orientation so as to connect the exit of the  $i^{\text{th}}$  tunnel to the entrance of the  $i + 1^{\text{st}}$  tunnel:

$$\int_{s_{2i}}^{s_{2i+1}} \kappa_{2i+1} ds = \theta_{2i}^{2i+1}, \quad (6.1.5)$$

$$\int_{s_{2i}}^{s_{2i+1}} \sin \theta(s, t) ds = x_{2i}^{2i+1}, \quad (6.1.6a)$$

and

$$\int_{s_{2i}}^{s_{2i+1}} \cos \theta(s, t) ds = y_{2i}^{2i+1}. \quad (6.1.6b)$$

$\theta_{2i}^{2i+1}$  and  $\bar{x}_{2i}^{2i+1} = (x_{2i}^{2i+1}, y_{2i}^{2i+1})$  are the orientation and position of the frame at the entrance of the  $i + 1^{\text{st}}$  tunnel with respect to the frame at the exit of the  $i^{\text{th}}$  tunnel, measured in the base frame. That is,  $\theta_{s_{2i}}^{s_{2i+1}}(t) = \theta(s_{2i+1}, t) - \theta(s_{2i}, t)$  and  $\bar{x}_{2i}^{2i+1}(t) = \bar{x}(s_{2i+1}, t) - \bar{x}(s_{2i}, t)$ . In this way, the manipulator backbone curve is at least a once continuously differentiable curve along its whole length. The constraint in Equation (6.1.5) can be relaxed if nonsmooth backbone curves are allowed.

The velocity constraints on the endpoints of the free sections of the manipulator are found by simply taking the time derivative of the position constraints represented by Equations (6.1.5) and (6.1.6) to yield:

$$\frac{d}{dt} \int_{s_{2i}}^{s_{2i+1}} \kappa_{2i+1} ds = \frac{d\theta_{2i}^{2i+1}}{dt}, \quad (6.1.7)$$

$$\frac{d}{dt} \int_{s_{2i}}^{s_{2i+1}} \sin \theta(s, t) ds = \frac{dx_{2i}^{2i+1}}{dt}, \quad (6.1.8a)$$

and

$$\frac{d}{dt} \int_{s_{2i}}^{s_{2i+1}} \cos \theta(s, t) ds = \frac{dy_{2i}^{2i+1}}{dt}. \quad (6.1.8b)$$

Since  $s_k = s_k(t)$ , Leibnitz's rule can be used to write the velocity constraints of Equations (6.1.7) and (6.1.8) explicitly as:

$$\left[ \frac{\partial \theta(\nu, t)}{\partial t} + \frac{d\nu}{dt} \kappa(\nu, t) \right]_{\nu=s_{2i}}^{s_{2i+1}} = \frac{d\theta_{2i}^{2i+1}}{dt} \quad (6.1.9)$$

$$\int_{s_{2i}}^{s_{2i+1}} \frac{\partial \theta(s, t)}{\partial t} \cos \theta(s, t) ds + \left[ \frac{d\nu}{dt} \sin \theta(\nu, t) \right]_{\nu=s_{2i}}^{s_{2i+1}} = \frac{dx_{2i}^{2i+1}}{dt} \quad (6.1.10)$$

and

$$-\int_{s_{2i}}^{s_{2i+1}} \frac{\partial \theta(s, t)}{\partial t} \sin \theta(s, t) ds + \left[ \frac{d\nu}{dt} \cos \theta(\nu, t) \right]_{\nu=s_{2i}}^{s_{2i+1}} = \frac{dy_{2i}^{2i+1}}{dt} \quad (6.1.11)$$

where the following notation has been used:  $[f(\nu, d\nu/dt, t)]_{\nu=a}^b = f(b, db/dt, t) - f(a, da/dt, t)$  for arbitrary functions  $a = a(t)$  and  $b = b(t)$ . Note that for the parts of the nonextensible backbone which fully occupy a tunnel,  $s_{2i+1} - s_{2i} = \text{const}_i$ , which means that

$$\dot{s}_{2i+1} = \dot{s}_{2i}. \quad (6.1.12)$$

The velocity of every point of the backbone curve is of the form:

$$\bar{v}(s, t) = \begin{pmatrix} \int_0^s \frac{\partial \theta}{\partial t} \cos \theta d\sigma \\ - \int_0^s \frac{\partial \theta}{\partial t} \sin \theta d\sigma \end{pmatrix}. \quad (6.1.13)$$

### 6.1.2 Spatial Tunnel Constraints

Equations very similar in nature to those defined for the planar obstacle avoidance problem are now defined for the spatial case. Figure 6.1 is still applicable, and the same indexing system is used.

The functions  $\mathcal{K}$  and  $\mathcal{T}$  are taken to be of the form:

$$\begin{aligned} \mathcal{K}(s, t) &= \sum_{i=1}^n \mathcal{K}_i(s, t) W(s, s_{i-1}, s_i) \\ \mathcal{T}(s, t) &= \sum_{i=1}^n \mathcal{T}_i(s, t) W(s, s_{i-1}, s_i) \end{aligned} \quad (6.1.14)$$

where

$$\mathcal{K}_{2i} = \mathcal{K}_{2i}(s - s_{2i-1}(t)), \quad \mathcal{T}_{2i} = \mathcal{T}_{2i}(s - s_{2i-1}(t)) \quad (6.1.15)$$

are of the form of traveling waves, and

$$\mathcal{K}_{2i+1} = \mathcal{K}_{2i+1}(s, t), \quad \mathcal{T}_{2i+1} = \mathcal{T}_{2i+1}(s, t) \quad (6.1.16)$$

are of a more general form, but must obey the constraint equations

$$\int_{s_{2i}}^{s_{2i+1}} \mathcal{K}_{2i+1} ds = \Delta K_i, \quad \int_{s_{2i}}^{s_{2i+1}} \mathcal{T}_{2i+1} ds = \Delta T_i \quad (6.1.17)$$

and

$$\bar{x}(s_{2i+1}, t) - \bar{x}(s_{2i}, t) = \bar{d}_i \quad (6.1.18)$$

where  $\Delta K_i$  and  $\Delta T_i$  are the difference in the angles  $K(s, t)$  and  $T(s, t)$  which specify the orientation of the  $i+1^{st}$  tunnel entrance with respect to the  $i^{th}$  tunnel exit.  $\bar{d}_i$  is the constant vector measuring the difference in position of the entrance of the  $i+1^{st}$  tunnel with respect to the exit of the  $i^{th}$  tunnel as measured in the base frame.

Rate equations corresponding to Equation (6.1.16) are of the same form as Equation (6.1.9). The explicit rate form of Equation (6.1.18) is a complicated expression when written in terms of the quasi-curvature and quasi-torsion, and can be written symbolically as

$$\left[ \frac{d\nu}{dt} \bar{v}(\nu, t) + \bar{v}(\nu, t) \right]_{\nu=s_{2i}}^{s_{2i+1}} = \frac{d\bar{d}_i}{dt} \quad (6.1.19)$$

where the velocity vector  $\bar{v}(s, t)$  of every point on the manipulator can be written

as

$$\bar{v}(s, t) = \begin{pmatrix} - \int_0^s \frac{\partial T}{\partial t} \sin K \sin T d\sigma + \int_0^s \frac{\partial K}{\partial t} \cos K \cos T d\sigma \\ - \int_0^s \frac{\partial T}{\partial t} \cos K \sin T d\sigma - \int_0^s \frac{\partial K}{\partial t} \sin K \cos T d\sigma \\ \int_0^s \frac{\partial T}{\partial t} \cos T d\sigma \end{pmatrix} \quad (6.1.20)$$

by taking the time derivative of Equation (2.2.11) (with  $l(s, t) = 1$ ).

Equations (6.1.9-11) and (6.1.19) hold for generally defined exterior segments. Simplifications can be made if the exterior segments are restricted to a "quasi-modal" form. The next subsection defines and illustrates the use of this quasi-modal form.

### 6.1.3 A Quasi-Modal Form for Exterior Segment Constraints

Previously in this section, position and velocity constraints for the exterior segments of nonextensible hyper-redundant manipulator backbone curves were formulated. This section shows how the modal approach of Chapter 4 can be used to satisfy these constraints. Recall the definition of the modal form represented in Equations (4.5.1-4). A modified “quasi-modal” form is defined for exterior backbone segments as follows:

$$\begin{aligned} \mathcal{K}_{2i+1}(s, t) &= \frac{1}{s_{2i+1}(t) - s_{2i}(t)} \sum_{j=1}^{N_K} a_j(t) \phi_j \left( \frac{s - s_{2i}(t)}{s_{2i+1}(t) - s_{2i}(t)} \right) \\ \mathcal{T}_{2i+1}(s, t) &= \frac{1}{s_{2i+1}(t) - s_{2i}(t)} \sum_{j=N_K+1}^{N_K+N_T} a_j(t) \phi_j \left( \frac{s - s_{2i}(t)}{s_{2i+1}(t) - s_{2i}(t)} \right). \end{aligned} \quad (6.1.21)$$

This can be written equivalently as

$$\begin{aligned} K_{2i+1}(s, t) &= \sum_{j=1}^{N_K} a_j(t) \Phi_j \left( \frac{s - s_{2i}(t)}{s_{2i+1}(t) - s_{2i}(t)} \right) \\ T_{2i+1}(s, t) &= \sum_{j=N_K+1}^{N_K+N_T} a_j(t) \Phi_j \left( \frac{s - s_{2i}(t)}{s_{2i+1}(t) - s_{2i}(t)} \right). \end{aligned} \quad (6.1.22)$$

In the planar case,  $\mathcal{T}_{2i+1} = T(s, t) = 0$ , and  $\mathcal{K}_{2i+1}(s, t) = \kappa_{2i+1}(s, t)$ , and the quasi-modal form is also applicable.

In this way, if we choose bending modes,  $\{\Phi_i\}$ , which yielded closed form solutions for the end-effector problem in Chapter 4, they will yield closed form solutions for the exterior segment constraints expressed in Equations (6.1.5) and (6.1.6) in the planar case, or Equation (6.1.18) in the spatial case. This is observed by a simple change of variables:

$$\sigma = \frac{s - s_{2i}(t)}{s_{2i+1}(t) - s_{2i}(t)} \quad (6.1.23)$$

which transforms all the exterior segment constraints into exactly the same problem as the end-effector placement problem:

$$\int_{s_{2i}}^{s_{2i+1}} \bar{u} \left( \frac{s - s_{2i}(t)}{s_{2i+1}(t) - s_{2i}(t)}, t \right) ds = (s_{2i+1}(t) - s_{2i}(t)) \int_0^1 \bar{u}(\sigma, t) d\sigma. \quad (6.1.24)$$

The quasi-modal form also ensures that the rate formulation of exterior segment constraints will mimic the rate problem for the modal end-effector case:

$$\mathcal{J}_i(\bar{a})\dot{\bar{a}} = \frac{d}{dt} \left( \frac{\bar{d}_i}{s_{2i+1} - s_{2i}} \right). \quad (6.1.25)$$

Thus, we have a natural choice for implementing rate constraints for nonextensible exterior segments. Using the method depicted in Equation (6.1.25), a modified resolved rate modal approach results. Examples of this approach will be given in Section 6.3. But first the next section shows how extensible manipulator obstacle avoidance can be treated naturally using the nonextensible formulation of this section.

## 6.2 Extensible Manipulator Obstacle Avoidance

The methods developed in Section 6.1 for nonextensible planar and spatial hyper-redundant manipulators generalize quite easily for extensible manipulators. Features of extensible hyper-redundant manipulators can be exploited to make them adhere to a specified obstacle avoiding trajectory in a variety of useful ways.

Assume an arc length parametrized time-varying set of obstacle-avoiding backbone curves:

$$\bar{x}(s, t) = \bar{x}_{av}(L, t) \quad \text{for} \quad (L, t) \in [L_{min}, L_{max}] \times [t_0, t_1], \quad (6.2.1)$$

where  $L_{min} > 0$  and  $L_{max} > L_{min}$  are respectively the shortest and longest length attainable by the hyper-redundant manipulator for configurations defined by the given backbone curve. Extensible manipulators can be fit to this set of time-varying configurations in several ways. For instance,  $L(s, t) = s$  could be chosen. This results in fitting the manipulator to this set of curves as if it were nonextensible, as was done in the previous section. Alternately,  $L(s, t)$  can be chosen arbitrarily from the set of strictly increasing functions with boundary conditions  $L(0, t) = 0$  and  $L(1, t) = L_0 \in [L_{min}, L_{max}]$ . Possible choices are the optimal

reparametrization of Equation (5.6.11) or a uniform extension  $L(s, t) = L_0(t)s$ . In either of these cases, the backbone curve  $\bar{x}(s, t) = \bar{x}_{av}(L(s, t), t)$  will avoid obstacles, because  $\bar{x}_{av}(L, t)$  does.

Extensible hyper-redundant manipulator motions can be viewed as combinations of extensibility and bending. In Chapter 4, pure bending and extension modes were defined in a way which simplified the end-effector placement problem. *Pure longitudinal* motions are defined below to simplify the analysis of extensible manipulator obstacle avoidance.

Recall that pure bending motions result if  $L = L(s, t_i)$  for some constant  $t_i$ , so that  $\bar{x}(s, t) = \bar{x}_{av}(L(s, t_i), t)$ . In this case, changes in manipulator configuration are determined solely due to bending because the length function is independent of time, and so extensibility cannot be varied. On the other extreme, obstacle avoidance for extensible manipulators in static environments can be implemented without any bending by using pure longitudinal motions. Pure longitudinal motions are defined by restricting the hyper-redundant manipulator to the time-varying curve  $\bar{x}(s, t) = \bar{x}_{av}(L(s, t), t_i)$ . The curve shape does not change in the case of pure longitudinal motion, but the amount of curve covered by a given manipulator changes as a function of time. Note that this is different from the definition of pure extensible modes in Chapter 4. In practice, extensible manipulator obstacle avoidance can be implemented as a combination of bending and longitudinal modes. An example of this mode of obstacle avoidance is demonstrated as part of a comprehensive example in the next section.

### 6.3 A Specific Example

Figures 6.2 and 6.3 illustrate a specific example of the planar formulation presented in Section 6.1.1. The hyper-redundant manipulator must pass through a single maze-like tunnel and reach a goal on the other side. In the first unconstrained exterior section, i.e.,  $0 < s < s_1$ , the curvature function is restricted to a form

which ensures that the backbone curve is free of the maze. For this example,

$$\kappa_1(s, t) = \frac{2\pi a(t)}{s_1(t)} \cos \frac{2\pi s}{s_1(t)} \quad (6.3.1)$$

is chosen, where  $a(t)$  will be determined such that the point  $\bar{x}(s_1(t), t)$  on the moving, or slithering, manipulator is always coincident with the tunnel entrance. The form for the curvature function was chosen to yield a closed form inverse kinematic solution for the exterior manipulator segment, i.e., the point  $s = s_1(t)$  moves along the backbone curve, but remains stationary in space. Integrating Equations (2.3.4) and (2.3.5) for the case  $l(s, t) = 1$ ,

$$x_1(s_1, t) = \int_0^{s_1} \sin \left( a \sin \frac{2\pi s}{s_1} \right) ds = 0 \quad (6.3.2)$$

and

$$x_2(s_1, t) = \int_0^{s_1} \cos \left( a \sin \frac{2\pi s}{s_1} \right) ds = J_0(a) s_1, \quad (6.3.3)$$

where  $J_0$  is the zeroth order Bessel function.

Let  $h > 0$  be the distance from the base of the manipulator to the entrance of the tunnel (Figure 6.2). Since  $h$  is a constant, the following must be true:

$$h = J_0(a) s_1. \quad (6.3.4)$$

For  $h$  to be positive, it is necessary that

$$0 < a < \mu_1$$

where  $\mu_1 \approx 2.405$  is the first zero of  $J_0$ .

The condition for stationarity of  $\bar{x}(s_1, t)$  at the entrance of the tunnel, while  $s_1$  varies, is

$$a(t) = J_0^{-1} \left( \frac{h}{s_1(t)} \right) \quad (6.3.5)$$

and so

$$\kappa_1(s, t) = \frac{2\pi}{s_1(t)} J_0^{-1} \left( \frac{h}{s_1(t)} \right) \cos \frac{2\pi s}{s_1(t)} \quad (6.3.6)$$

where  $J_0^{-1}$  is a restricted inverse Bessel function, defined in Chapter 4. Other closed form inverse kinematic solutions can be found using the modal approach.

The obstacle environment is illustrated in Figure 6.2. With  $\bar{x}(s_1, t) = \bar{d}_1 = [0, h]^T$  fixed, an appropriate curvature function for the section of manipulator confined to the tunnel can be determined. This curvature function is denoted as  $\kappa_2(s, t)$ .

For this particular obstacle field  $\kappa_2$  can be defined as follows:

$$\begin{aligned}\kappa_2(s, t) = & -\frac{\pi}{L_c} [W(s, s_1, s_1 + L_c) - W(s, s_1 + L_c, s_1 + 2L_c) \\ & + W(s, s_1 + 2L_c, s_1 + 3L_c)].\end{aligned}\quad (6.3.7)$$

This choice of  $\kappa_2$  corresponds to three consecutive semi-circular arcs. The magnitude of the curvature over each of the three sections, each arc with length  $L_c$ , is  $\pi/L_c$ , corresponding to semicircles of radius  $r_c = L_c/\pi$ . The window functions take the value of unity over each of the semicircles, and the sign indicates the sense in which the arc turns. A positive sign indicates clockwise bending, and a negative sign indicates counterclockwise bending of the manipulator. The fact that  $\kappa_2(s, t)$  is a traveling wave in curvature is evident from the equality  $W(s, s_1 + a, s_1 + b) = W(s - s_1, a, b)$ , for all  $a < b \in \mathbf{R}$ .

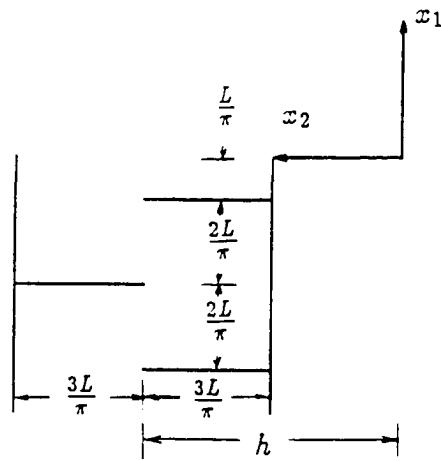
The composite curvature function for this example is then

$$\kappa(s, t) = \kappa_1(s, t)W(s, 0, s_1) + \kappa_2(s, t)W(s, s_1, 1). \quad (6.3.8)$$

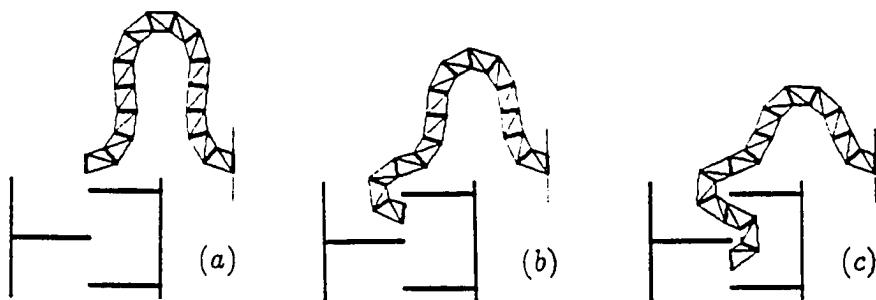
Integrating Equation (6.3.8) in the variable  $s$ ,

$$\begin{aligned}\theta(s, t) = & J_0^{-1}\left(\frac{h}{s_1}\right) \sin\left(\frac{2\pi s}{s_1}\right) - \frac{\pi}{L_c}(s - s_1)W(s, s_1, s_1 + L_c) \\ & + [\frac{\pi}{L_c}(s - s_1 - L_c) - \pi]W(s, s_1 + L_c, s_1 + 2L_c) \\ & - \frac{\pi}{L_c}(s - s_1 - 2L_c)W(s, s_1 + 2L_c, s_1 + 3L_c) \\ & - \pi W(s, s_1 + 3L_c, 1).\end{aligned}\quad (6.3.9)$$

A time history of this nonextensible tunneling obstacle avoidance maneuver is shown in Figure 6.3, corresponding to the obstacle with dimensions shown in Figure 6.2. The configurations shown correspond to  $h = 0.4$ ,  $L_c = 0.2$ , and  $s_1 = 1.00, 0.83, 0.68$ .



**Figure 6.2:** Dimensions of the Obstacle Field



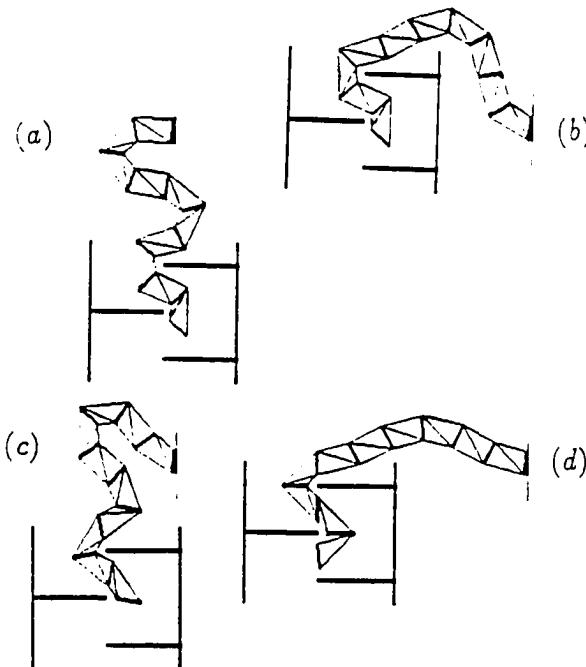
**Figure 6.3:** An Obstacle Avoiding Backbone Curve

The procedure used for the stationary obstacle is easily generalized to the case of a moving obstacle field. The backbone curve is partitioned into two sections as in

Equation (6.3.8), but now:

$$\kappa_1(s, t) = \frac{2\pi}{s_1(t)} \hat{a}_1^+(\bar{d}_1(t)/s_1(t)) \cos \frac{2\pi s}{s_1(t)} + \frac{2\pi}{s_1(t)} \hat{a}_2(\bar{d}_1(t)/s_1(t)) \sin \frac{2\pi s}{s_1(t)} \quad (6.3.10)$$

where  $\hat{a}_1^+(\cdot)$  and  $\hat{a}_2(\cdot)$  are the inverse kinematic solutions in Equations (4.2.4) and (4.2.5).  $\kappa_1$  now assures that a point on the manipulator backbone at  $s = s_1(t)$  has fixed position with respect to the obstacle field entrance (independent of how the obstacle translates in the plane) provided the entrance is within reach. That is,  $\kappa_1$  assures that  $\bar{x}(s_1(t), t) = \bar{d}_1(t)$ , which is the position vector of the entrance to the obstacle field in the base frame. Note that this is an example of the quasi-modal form introduced in Section 6.1.3.  $\kappa_2$  again has the form of a traveling wave (composed of three semicircular arcs) for the section of the manipulator which is inside the tunnel given by Equation (6.3.7). Figures 6.4(a)-(d) show this solution for  $\bar{d}_1 \in [(-0.3, 0.0), (0.2, 0.46), (-0.17, 0.19), (0.0, 0.57)]$ , and  $s_1 \in \{0.65, 0.65, 0.69, 0.61\}$ .



**Figure 6.4: Obstacle-Avoiding Configurations for Moving Obstacles**

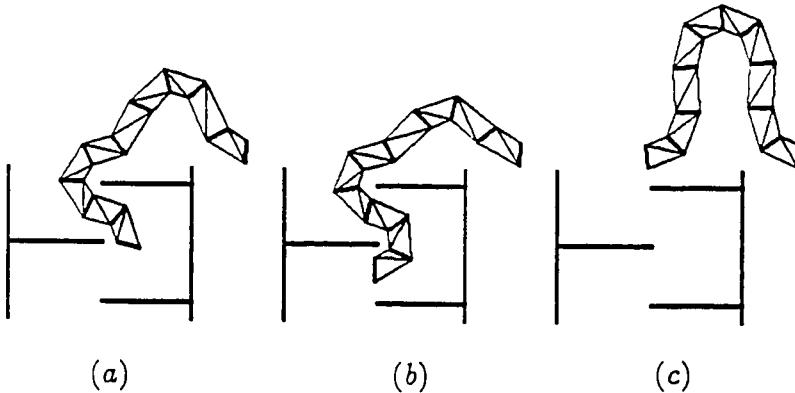
Incorporating extensibility into this example is rather straightforward. Only now,

$$\begin{aligned}\kappa_1(s, t) &= \frac{2\pi}{L(s_1(t), t)} \hat{a}_1^+(\bar{d}_1(t)/L(s_1(t), t)) \cos \frac{2\pi L(s, t)}{L(s_1(t), t)} \\ &+ \frac{2\pi}{L(s_1(t), t)} \hat{a}_2(\bar{d}_1(t)/L(s_1(t), t)) \sin \frac{2\pi L(s, t)}{L(s_1(t), t)}\end{aligned}\quad (6.3.11)$$

and

$$\begin{aligned}\kappa_2(s, t) &= -\frac{\pi}{L_c} [W(L(s, t) - L(s_1(t), t), 0, L_c) - W(L(s, t) - L(s_1(t), t), L_c, 2L_c) \\ &+ W(L(s, t) - L(s_1(t), t), 2L_c, 3L_c)].\end{aligned}\quad (6.3.12)$$

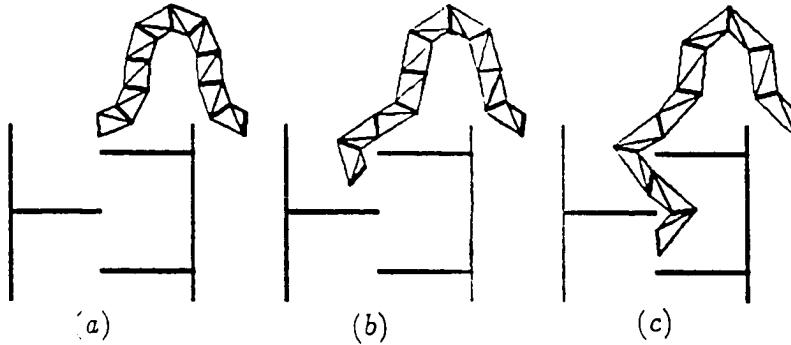
Equations (6.3.11) and (6.3.12) are simply generalizations of Equations (6.3.10) and (6.3.7) with  $L(s, t)$  replacing  $s$ .  $L(s, t)$  is chosen within the constraints  $\dot{L}(s, t) = l(s, t) > 0$ ,  $L(0, t) = 0$ ,  $L_{min} < L(1, t) < L_{max}$ , and whatever constraints are introduced by the actual mechanism joint limits. In Figure 6.5, the same backbone curve as in Figure 6.4 is used, only now  $\dot{L}(s, t) \neq 1$ , but is rather defined as in Equation (5.6.11) with  $(L_0, r) = (1.0, 0.1)$ .



**Figure 6.5:** Optimally Reparametrized Obstacle-Avoiding Configurations

If the obstacle is stationary, then  $\bar{d}_1$  is constant, and purely longitudinal obstacle avoidance can be implemented. Recall that pure longitudinal locomotion can be implemented only when the backbone curve shape has no time dependence.

For this example, fixing the time dependence of the obstacle avoidance curve is achieved by fixing  $s_1(t)$  for constant  $\bar{d}_1$ . The behavior of the function  $L(s, t)$  can be chosen in a variety of ways. In Figure 6.6,  $\bar{d}_1 = [0.3, 0]^T$ ,  $s_1 = 0.5$ , and  $L(s, t) = L_0(t)s$  for  $L_0 \in \{0.75, 1.0, 1.25\}$ .



**Figure 6.6:** Avoiding Obstacles with Purely Longitudinal Motion

## 6.4 Operator Specification of Obstacle Avoiding Paths

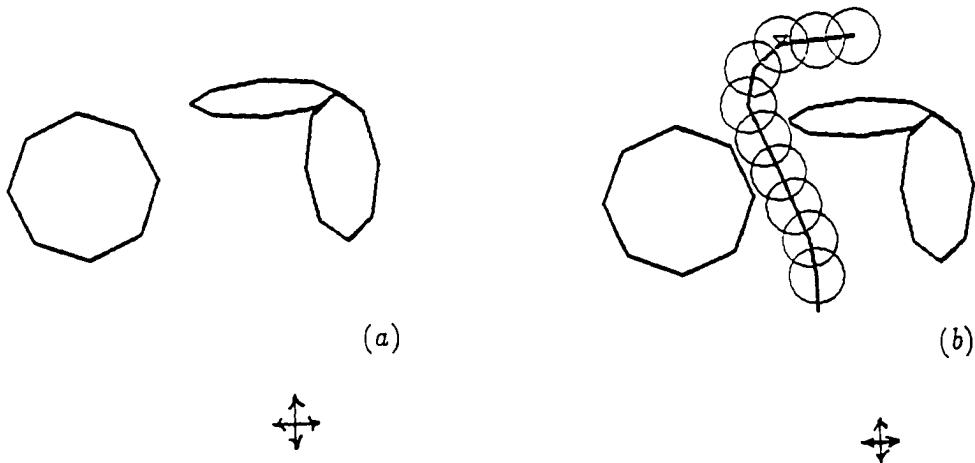
Often obstacle avoidance problems will only require a single tunnel to be defined. This section introduces a user-interactive obstacle avoidance algorithm for the planar single tunnel case. Using this approach, the operator of the hyper-redundant robot chooses paths graphically to avoid obstacles. The basis of this approach is straightforward. The interior segment curvature function is restricted to a class which has closed form forward kinematics. In particular, the following form will be used:

$$\kappa_2(s, t) = \sum_{i=1}^{N_\delta} \theta_i \delta \left( s - \sum_{j=1}^i L_j \right), \quad (6.4.1)$$

which yields a path divided into  $N_\delta$  connected line segments each with correspond-

ing length  $L_i$ , and angle between the  $i - 1$ st and  $i$ th segments denoted by  $\theta_i$  for  $i \in [1, N_\theta]$ . A computer interface allows the user to alter the variables  $L_i$  and  $\theta_i$  so as to achieve a particular desired obstacle avoiding path.

In effect Equation (6.4.1) results in a kinematic structure which mimics the revolute manipulator studied in Section 3.3. Only now the user is specifying link lengths and joint angles to shape an obstacle avoiding path. Once the operator of the hyper-redundant robot is satisfied with the path chosen, the interior segment curvature function is matched to an exterior segment, just as in the example of the previous section. In this section, the same exterior segment curvature function as the previous example is used, i.e., that given by Equation (6.3.11). The composite curvature function is again of the form of Equation (6.3.8), with  $\kappa_1(s, t)$  given by Equation (6.3.7). Fitting of the actual hyper-redundant manipulator to this nonsmooth trajectory is achieved in the same way as for the nonsmooth modal example in Equation (4.2.12).



**Figure 6.7:** An Arbitrary Obstacle Field with User Defined Interior Segment

As an example, consider the arbitrarily defined obstacle field shown in Figure 6.7(a). The discretely segmented straight-line path shown in Figure 6.7(b) was defined by the user to form a tunnel through the obstacle field. A collection

of circles surrounds the path, the envelope of which forms a tube in which the manipulator can enter or exit. By ensuring that this tube does not intersect obstacles, the manipulator is guaranteed not to intersect obstacles. Figures 6.8 and 6.9 show hyper-redundant manipulators penetrating an obstacle field with pure bending and purely longitudinal motions respectively.

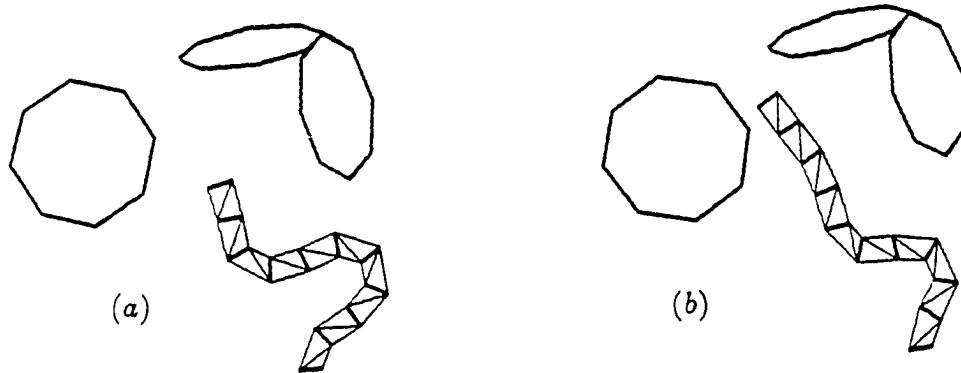


Figure 6.8: Pure Bending Obstacle Avoidance

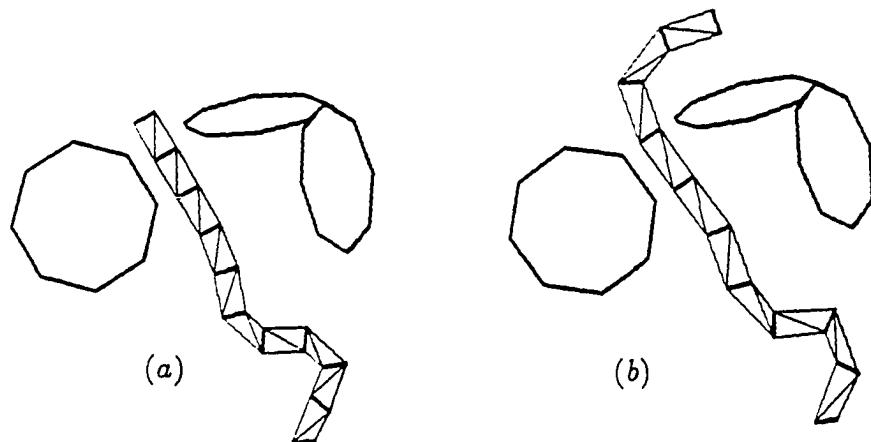


Figure 6.9: Purely Longitudinal Obstacle Avoidance

Previously developed automatic methods for generating obstacle avoiding paths for mobile robots can be used in place of operator input to generate obstacle-

avoiding backbone curves of hyper-redundant manipulators. One such trajectory generation technique developed by [AnA90] automatically generates obstacle-avoiding paths consisting of straight-line segments. If this automatic method were chosen, the interior segment curvature function would still be of the form of Equation (6.4.1), only the parameters  $L_i$ , and  $\theta_i$  would be generated automatically instead of by a human operator.

## 6.5 Discussion

This chapter presented a novel obstacle avoidance concept, based on “tunneling,” for hyper-redundant manipulators of constant and variable length through stationary and moving obstacle fields. A general formulation was given which allows a manipulator to maneuver through a complicated sequence of interior and exterior segments. Computer simulations were presented for a particular obstacle field. In principle, connecting circular arcs (or more complicated blending curves) and line segments can be used to construct a system of tunnels for maneuvering in arbitrarily complex obstacle fields. The benefit of this method over existing potential field and optimal methods for application to hyper-redundant manipulators is that a comparatively efficient set of kinematic equations based on differential geometry is computed, allowing much faster solutions. However, the present formulation does not preclude the use of other methods to define the trajectory to which the hyper-redundant manipulator is to adhere. The methods of Chapter 3 are used to fit the actual hyper-redundant manipulators to the backbone curve, independent of how that curve is generated. While the applicability of the present model diminishes with diminishing degree of redundancy, it works quite well in situations such as the “infinitely redundant” manipulator case, where conventional methods of analysis clearly do not apply.

## Chapter 7

# Hyper-Redundant Locomotion and Grasping

In Chapters 2-6, a kinematics and motion planning framework was developed for robots with fixed bases. In this chapter, the analysis presented earlier is extended to the locomotion of mobile hyper-redundant robots. Hyper-redundant locomotion is defined here to mean motions resulting in net displacements of hyper-redundant robots due to the internally induced bending and twisting of the mechanism. Actuatable wheels, tracks, or legs are not necessary. It will also be shown that hyper-redundant robot locomotion has applications to the grasping and manipulation of objects. The analysis performed in this chapter will be for planar problems, but the ideas are generalizable to the spatial case.

The position vector to all points on a moving hyper-redundant robot backbone curve relative to a fixed inertial frame are written as:

$$\bar{y}(s, t) = \mathcal{R}(t)\bar{x}(s, t) + \bar{c}(t).$$

$\mathcal{R}(t)$  is a  $3 \times 3$  rotation matrix which describes the orientation of the backbone curve base frame relative to the inertial frame, and  $\bar{c}(t)$  is the  $3 \times 1$  vector from the origin of a fixed frame to the origin of the backbone curve base frame. Because of the invariance of the intrinsic description of backbone curves with respect to

rigid body motion, all geometric features of mobile hyper-redundant robots are still captured by  $\kappa(s, t)$  and  $l(s, t)$  in the planar case. In this chapter several types of locomotion and grasping schemes will be classified and demonstrated using intrinsic geometry.

This chapter is organized as follows. Section 7.1 introduces two categories of nonextensible (bending mode) hyper-redundant locomotion: stationary and traveling wave. Section 7.2 illustrates analogous forms of extensible (longitudinal mode) locomotion. Section 7.3 shows how the intrinsically defined locomotion methods developed in Sections 7.1 and 7.2 can be used for locomotion over arbitrary terrain. Section 7.4 develops an alternate approach to the intrinsic formulation: describing locomotion with respect to the geometry of the terrain. Section 7.5 shows how ideas developed for hyper-redundant locomotion apply to the grasping and manipulation of objects. Section 7.6 discusses potential applications for the locomotion and grasping techniques developed in this chapter.

## 7.1 Classification of Nonextensible Locomotion

This section presents two classes of gaits which are highly idealized forms of biological locomotion. In this context, a “gait” is a repetitive sequence of mechanism deformations which cause net motion of the robot. The term “wave” will often be used to describe the mechanism deformations associated with these hyper-redundant locomotion gaits. The two categories presented here are by no means the only possible modes of locomotion, but were chosen for their simplicity and wide range of applicability.

The first class of gaits uses an amplitude varying wave which remains stationary with respect to body coordinates, and is thus referred to as *stationary wave locomotion*. For instance, roughly speaking, an inchworm extends and contracts its body in a manner in which the “hump” remains in approximately the same location of the body. A second category of gaits which mimics the pedal waves of

the slug (see Figure 1.5) or the locomotion of a caterpillar is also considered, and is referred to as *traveling wave locomotion*. In this category of gaits, the amplitude of the wave remains relatively constant, but travels the length of the body. Complicated forms of locomotion may not fit into either of these categories. Examples of these are the whirling and flagellar movements used by some creatures for swimming, and the sidewinding and concertina modes of snake locomotion.

The next two subsections consider nonextensible stationary and traveling wave gaits separately. Examples illustrate suitable curvature functions for implementing locomotion. In these subsections, it is assumed that the robot locomotes over a level surface. Section 7.3 shows how this formulation can be applied to locomotion over uneven terrain.

### 7.1.1 Stationary-Wave-Amplitude-Varying Locomotion

As stated in Section 2.3, the kinematics of planar hyper-redundant robot backbone reference sets can be reduced to the determination of the spatial and temporal behavior of the backbone curvature function,  $\kappa(s, t)$ , and the rate of length function,  $l(s, t)$ . Nonextensible *stationary-wave-amplitude-varying* (SWAV) locomotion over flat terrain can be represented with curvature and rate of length functions of the form:

$$\kappa_s(s, t) = \alpha_s(t)\phi_s(s) \quad (7.1.1a)$$

$$l(s, t) = 1 \quad (7.1.1b)$$

where  $\phi_s(s)$  is a curvature mode function and  $\alpha_s(t)$  is the associated modal participation factor. Figures 7.1 and 7.2 show computer-generated time-varying sequences of variable geometry trusses demonstrating SWAV locomotion over flat terrain. In SWAV locomotion, forward travel is achieved by varying the amplitude of the robot's curvature function. This is achieved via the time dependence of the participation factors, i.e.,  $\alpha_s(t)$  is taken to be a periodic function. SWAV locomotion implicitly assumes that the bottom of the mechanism has greater resistance

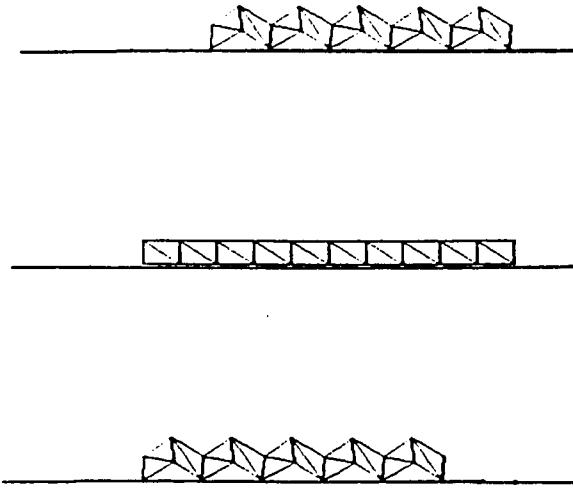
to motion in the forward direction than in the reverse direction. In practice, this can be achieved with passive racheting wheels which only turn in one direction, or scales which slide during forward motion and exert traction on the ground preventing retrograde motion (see Chapters 8 and 9 for an actual implementation).

One possible curvature function appropriate to model SWAV locomotion is

$$\kappa_s(s, t) = \alpha_s(t) \cos 2\pi m s, \quad (7.1.2a)$$

for some integer  $m$ , where  $\alpha_s(t)$  is chosen to be of the form

$$\alpha_s(t) = 2\pi m(\alpha_0 + \alpha_1 \cos \omega t). \quad (7.1.2b)$$



**Figure 7.1:** Nonextensible SWAV Locomotion: Example 1

Figure 7.1 shows a variable geometry truss which has been fit to a backbone curve with curvature defined in Equations (7.1.2) for  $m = 5$ . In practice, choosing  $m \gg 1$  lowers the robot center of gravity.  $\alpha_0$  and  $\alpha_1$  in Equation (7.1.2b) must obey  $\alpha_{max} > \alpha_0 > |\alpha_1| > 0$ , where specifying  $\alpha_{max}$  limits the allowable curvature at any point in the robot. Bounds on the curvature may be imposed by constraints which arise in the physical implementation of the robot, or may be imposed to

prevent self-intersection. The constraint  $\alpha_0 > |\alpha_1| > 0$  is necessary to prevent a violation of the kinematic terrain constraints. That is, these constraints prevent the “humps” from penetrating the terrain surface.

Substituting Equation (7.1.2) into Equations (2.3.4) and (2.3.5), the location of the front of the robot relative to the back is:

$$x_1(1, t) = 0; \quad x_2(1, t) = J_0(\alpha_0 + \alpha_1 \cos \omega t) \quad (7.1.3)$$

where  $J_0(\cdot)$  again is the Bessel function of zeroth order.  $x_2(1, t)$  will be referred to as the *nominal body length*. The *stride length*,  $L_{SL}$ , which is normally defined as the distance traveled by a legged animal or robot over one complete cycle of foot placements, is modified to mean the distance traveled over one temporal period of the curvature function, which in this case is

$$L_{SL} = J_0(\alpha_0 - \alpha_1) - J_0(\alpha_0 + \alpha_1). \quad (7.1.4)$$

Note that the stride length is the difference between the maximum and minimum nominal body length. Other gait analytic terms such as stepping sequence and duty factor [SoW89] have little useful meaning here, primarily because there are no foot placements.

The previous choice of curvature is not by any means unique for SWAV locomotion. As a second example, a curvature function is defined as a piecewise continuous function composed of constant sections:

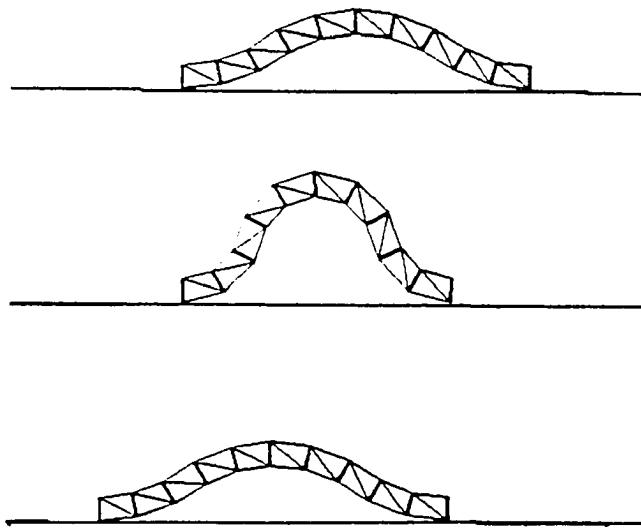
$$\begin{aligned} \kappa(s, t) = \sum_{i=1}^m & \left[ a_{3i}(t) W\left(s - \frac{i-1}{m}, 0, \frac{1}{4m}\right) - a_{3i+1}(t) W\left(s - \frac{i-1}{m}, 1/4m, 3/4m\right) \right. \\ & \left. + a_{3i+2}(t) W\left(s - \frac{i-1}{m}, 3/4m, 1/m\right) \right] \end{aligned} \quad (7.1.5)$$

where the window function  $W(s, s_0, s_1)$  is defined as in Equation (4.5.8), i.e., it is unity on the interval  $s_0 \leq s < s_1$  and zero otherwise. The resulting wave is a combination of  $3m$  circular arcs, as shown in Figure 7.2 for the case  $m = 1$ . The

kinematic constraint that no point of the robot should penetrate the terrain is satisfied by setting:

$$a_{3i}(t) = a_{3i+1}(t) = a_{3i+2}(t) = \alpha_s(t) > 0 \quad \text{for } i \in [1, m] \quad (7.1.6)$$

for a chosen oscillatory  $\alpha_s(t)$ . As  $\alpha_s(t)$  varies, Equation (7.1.6) also ensures that the backbone curve is tangent to the flat terrain at all points of contact.



**Figure 7.2:** Nonextensible SWAV Locomotion: Example 2

### 7.1.2 Traveling-Wave-Amplitude-Constant Locomotion

Many terrestrial creatures with hyper-redundant morphologies are capable of sending waves along their bodies for propulsion. In this section, we idealize this form of locomotion as a traveling body wave which has constant amplitude. This *traveling-wave-amplitude-constant* (TWAC) form of nonextensible locomotion can be implemented with a curvature function of the form:

$$\kappa_t(s, t) = \phi_t(s - \alpha_t(t)). \quad (7.1.7)$$

For example, let

$$\kappa_t(s, t) = \epsilon W(s - \alpha_t(t), 0, 1/m) \cos 2\pi m(s - \alpha_t(t)) \quad (7.1.8)$$

where  $m$  is any integer such that  $m > 2$  and  $\epsilon$  is the maximum curvature of the wave. The integer  $m$  is chosen so that the wave constitutes  $1/m^{\text{th}}$  of the total arc length of the robot.

One possible choice for the time dependence of the curvature function is

$$\alpha_t(t) = \omega \left[ t - I \left( t, \frac{m+1}{m\omega} \right) \right] - 1/m. \quad (7.1.9)$$

The function  $I(t, t_p)$ , which will be used extensively in this chapter, is defined as

$$I(t, t_p) = t_p (\text{int}) \left( \frac{t}{t_p} \right) \quad (7.1.10)$$

where (int) is the greatest integer function, and  $t_p > 0$  is the period of the locomotion cycle.  $\omega$  is the instantaneous wave speed when the wave is traveling along a robot of unit length. However, the average wave speed dictated by Equation (7.1.9) is  $\frac{m\omega}{m+1}$  because there is an interval of time when no complete wave is present in the robot during each cycle.

The time varying curvature function in Equation (7.1.8) causes a “hump” of constant magnitude (with the same shape as the first example of Subsection 7.1.1) to travel from one end of the robot to the other. With each additional increase in time of  $\frac{m+1}{m\omega}$  the process is repeated, and the robot advances by one stride length,  $L_{SL}$ , which for this example is given by:

$$L_{SL} = \frac{1}{m} \left[ 1 - J_0 \left( \frac{\epsilon}{2\pi m} \right) \right]. \quad (7.1.11)$$

As a second example of TWAC locomotion, a wave with the same shape as in the second SWAV example is chosen, but with temporal behavior:

$$\begin{aligned} \kappa_t(s, t) = & a_1 W(s - \alpha_t(t), 0, 1/4m) - a_2 W(s - \alpha_t(t), 1/4m, 3/4m) \\ & + a_2 W(s - \alpha_t(t), 3/4m, 1/m). \end{aligned} \quad (7.1.12)$$

$\alpha_t(t)$  is the same as Equation (7.1.9) and the  $\{a_i\}$  are constants which obey the constraints of Equation (7.1.6). The resulting robot shape for this choice of curvature is shown in Figure 7.3 for  $m = 3$ .

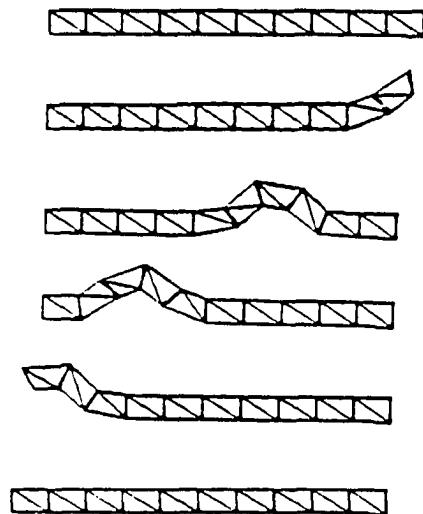


Figure 7.3: Nonextensible TWAC Locomotion

## 7.2 Classification of Extensible Locomotion

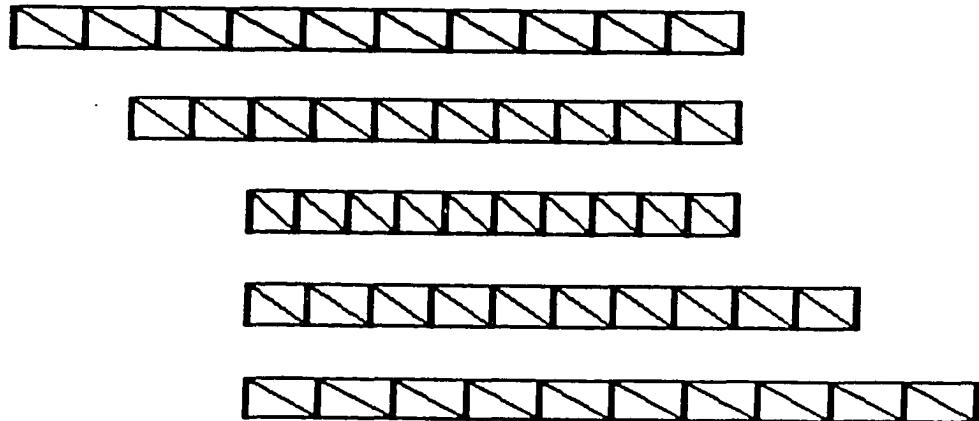


Figure 7.4: Extensible SWAV Locomotion

The two modes of nonextensible hyper-redundant locomotion classified in the previous section have extensible counterparts. For flat terrain, SWAV locomotion

can be defined as

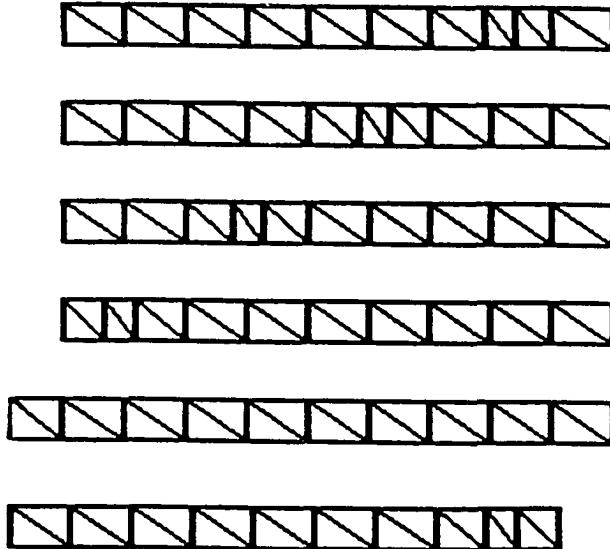
$$\kappa(s, t) = 0 \quad (7.2.1a)$$

$$l(s, t) = 1 + \alpha_s(t)\phi_s(s) \quad (7.2.1b)$$

where  $\alpha_s(t)$  is an oscillatory function, which will generally be chosen as in Equation (7.1.2b). There is great freedom in the choice of  $\phi_s(s)$ , but a uniform extension and contraction of the form

$$\phi_s(s) = 1 \quad (7.2.2)$$

is chosen here. This is shown in Figure 7.4 for  $m = 1$ ,  $\alpha_0 = 0$ , and  $\alpha_1 = \frac{1}{5}$  in Equation (7.1.2b). The stride length in this case is  $L_{SL} = 2\alpha_1$ .



**Figure 7.5:** Extensible TWAC Locomotion

For a purely longitudinal traveling wave over flat terrain, we define

$$\kappa(s, t) = 0 \quad (7.2.3a)$$

$$l(s, t) = 1 + E_0 \phi_l(s - \alpha_t(t)). \quad (7.2.3b)$$

This is shown in Figure 7.5 for  $\phi_l(\cdot) = W(\cdot, 0, 1/5)$  and  $E_0 = -\frac{1}{2}$ . Note that this is a contraction wave. If  $E_0 > 0$ , the locomotion wave would be an expansion wave which must travel in the direction opposite to the direction of robot travel.

In practice, traveling waves are required not to contact the terrain so that strides can be achieved without relying on differential friction. This is achieved for the case of flat terrain with a composite bending and longitudinal traveling wave so that Equations (7.2.3b) and (7.1.7) hold simultaneously.

## 7.3 Locomotion over Uneven Terrain

The formulation in Sections 7.1 and 7.2 is naturally extended to locomotion over curved terrain. The following subsections formulate the locomotion problem for arbitrary terrain.

### 7.3.1 Nonextensible Locomotion over Curved Terrain

The curvature functions given in Equation (7.1.5) or (7.1.12) can be used for a wide variety of terrain shapes when the constraints in Equation (7.1.6) are modified. The waves which result from the curvature function in Equation (7.1.12) have three degrees of freedom (much like a 3 mode inverse kinematic solution) when not constrained by Equation (7.1.6). The free degrees of freedom correspond to free choices for  $a_1$ ,  $a_2$ ,  $a_3$ . Equation (7.1.6) was a constraint imposed by the flat terrain which could be expressed in closed form. While a variety of three degree of freedom waves can be imagined, those defined in Equations (7.1.5) and (7.1.12) are used in the following examples. Nonetheless, the results generalize to other waveforms.

Assume the terrain is described by an arc length parametrized time-varying curve, written as  $\bar{T}(\mu, t) = [T_1(\mu, t), T_2(\mu, t)]$ , where  $\mu$  is the terrain arc length.  $\bar{T}(\mu, t)$  is defined so that  $\partial \bar{T}(0, t) / \partial \mu = [0, 1]^T = \bar{e}_2$ . Also assume that at  $t = 0$ , the base frame of the backbone curve coincides with  $\{\bar{e}_1, \bar{e}_2\}$ . After  $N_{cyc}$  traveling waves, the base frame of the robot is at  $\bar{T}(N_{cyc}L_{SL}, t)$ . Terrain constraints will generally not be expressible in closed form. In fact,  $\bar{T}(\mu, t)$  may not even have a closed form description.

In order to gain insight into the nonextensible locomotion problem over curved terrain, the intrinsic functions used to define the shape of the locomotion wave are written in the explicit form:

$$\begin{aligned}\kappa(s, t) = & \kappa_T(s + N_{cyc}L_{SL}, t)W(s, 0, s_0(t)) \\ & + \kappa^*(s, \bar{a}(t), t)W(s, s_0(t), s_1(t)) \\ & + \kappa_T(s + (N_{cyc} - 1)L_{SL}, t)W(s, s_1(t), 1)\end{aligned}\quad (7.3.1)$$

where  $s = s_1$  and  $s = s_0$  denote the front and back of the wave (stationary or traveling) measured along the backbone curve from its base, and  $\kappa_T(\mu, t) = |\partial^2 \bar{T}/\partial^2 \mu|$  is terrain curvature. This notation illustrates explicit dependence of the wave shape upon the variables  $\{\alpha_i\}$ . Strictly speaking, this is not of the modal form defined in Chapter 4.

The vector  $\bar{\mathcal{X}}(\bar{a}, t) = [\mathcal{X}_1(\bar{a}, t), \mathcal{X}_2(\bar{a}, t), \Delta\theta(\bar{a}, t)]^T$  is defined as the relative position and orientation of a frame attached to the robot at  $s = s_1$  with respect to a frame attached to the robot at  $s = s_0$ . In the case of TWAC locomotion,  $s_0 = \alpha_t(t)$  and  $s_1(t) - s_0(t) = L_w$ , which is the wave length. In the case of SWAV locomotion  $s_0 = 0$  and  $s_1 = 1$ . In either case, the components of  $\bar{\mathcal{X}}(\bar{a}, t)$  are given explicitly as:

$$\mathcal{X}_1(\bar{a}, t) = \int_{s_0}^{s_1} \sin \theta^*(s, \bar{a}, t) ds \quad (7.3.2a)$$

$$\mathcal{X}_2(\bar{a}, t) = \int_{s_0}^{s_1} \cos \theta^*(s, \bar{a}, t) ds \quad (7.3.2b)$$

$$\Delta\theta(\bar{a}, t) = \theta^*(s_1, \bar{a}, t) - \theta^*(s_0, \bar{a}, t) \quad (7.3.2c)$$

where

$$\theta^*(s, \bar{a}, t) = \int_{-N_{cyc}L_{SL}}^s \kappa(\sigma, t) d\sigma = \theta_T(t) + \int_{s_0}^s \kappa^*(\sigma, \bar{a}, t) d\sigma \quad (7.3.3)$$

where  $\theta_T(t)$  is the difference in angle between the vectors tangent to the terrain at  $\mu = 0$  and  $\mu = s_0 + N_{cyc}L_{SL}$ . In this way,  $\{\mathcal{X}_1, \mathcal{X}_2\}$  are defined relative to the frame  $\{\bar{e}_1, \bar{e}_2\}$ .

We seek to determine the variables  $\bar{a} = [a_1, a_2, a_3]^T$  such that

$$\bar{\mathcal{X}}(\bar{a}, t) = \bar{\mathcal{X}}^{des}(t) \quad (7.3.4a)$$

where

$$\mathcal{X}_1^{des}(t) = T_1(s_1(t) + (N_{cyc} - 1)L_{SL}, t) - T_1(s_0(t) + N_{cyc}L_{SL}, t) \quad (7.3.4b)$$

$$\mathcal{X}_2^{des}(t) = T_2(s_1(t) + (N_{cyc} - 1)L_{SL}, t) - T_2(s_0(t) + N_{cyc}L_{SL}, t) \quad (7.3.4c)$$

$$\mathcal{X}_3^{des}(t) = \cos^{-1} \left[ \frac{\partial \bar{T}}{\partial s}(s_1(t) + (N_{cyc} - 1)L_{SL}, t) \cdot \frac{\partial \bar{T}}{\partial s}(s_0(t) + N_{cyc}L_{SL}, t) \right] \quad (7.3.4d)$$

is the desired trajectory (position and orientation) of the front of the wave with respect to the back of the wave such that the (possibly moving) terrain constraints are observed. The length  $L_{SL}$  is not only the stride length, but also the *slack*, i.e., the difference in length measured along the robot's backbone curve, and measured along the terrain. After each cycle of locomotion, the robot progresses along the terrain curve by the amount  $L_{SL}$ .  $N_{cyc} = (\text{int})(t/t_p)$  is the number of cycles of locomotion which have been performed up until the current time.  $t_p$  is the period of each locomotion cycle (which was  $\frac{m+1}{m\omega}$  in Equation (7.1.9)).

Unfortunately, unlike the flat terrain case, it is generally difficult to solve Equation (7.3.4a) for  $\bar{a}$  in closed form. However, a method analogous to the resolved rate scheme used for end-effector placement in Section 4.6 can be used. The time rate-of-change of Equation (7.3.4a) is expressed as

$$\dot{\mathcal{X}}^{des} = \mathcal{J}(\bar{a})\dot{\bar{a}} + \frac{\partial \bar{\mathcal{X}}}{\partial t} \quad (7.3.5)$$

which can be solved for the  $\{\dot{a}_i\}$ . By assuming the robot originates on flat ground at possibly fictitious initial time  $t_0$ , the initial conditions  $\{a_i(t_0)\}$  can be specified by Equation (7.1.6). Solving Equation (7.3.5) for  $\dot{\bar{a}}$  and integrating yields a set of  $\{a_i\}$  which defines a wave that tracks the terrain.

This method will not work for all terrains because there is no guarantee that the specified wave shapes will not intersect the terrain, and/or that the Jacobian in Equation (7.3.5) will not become singular. Algorithms using this technique must check for both of these situations. This procedure will be illustrated with a

specific example in Section 7.5.1 in the context of a grasping and fine manipulation algorithm.

### 7.3.2 Extensible Locomotion over Curved Terrain

The analytical description of purely longitudinal locomotion over arbitrary terrain follows naturally from the purely longitudinal obstacle avoidance algorithm presented in Chapter 6.

Suppose again that the terrain curve is parametrized by arc-length,  $\bar{T} = \bar{T}(\mu, t)$ . Purely longitudinal locomotion will be implemented when the backbone curve is parametrized as follows:

$$\bar{x}(s, t) = \bar{T}(L(s, t) + d(t), t) \quad (7.3.6)$$

for appropriate choice of the functions  $L(s, t)$  and  $d(t)$ . These functions respectively describe how the backbone curve stretches along the terrain relative to its base, and how that base moves. By choosing  $l(s, t)$  as in Equation (7.2.1b) or (7.2.3b),  $L(s, t)$  can generate purely longitudinal stationary or traveling wave locomotion.  $d(t)$  is defined so that the progression of the robot along the given curve is consistent with the type of locomotion. The only difference between purely longitudinal locomotion and obstacle avoidance is the fact that  $d(t) = 0$  in the case of obstacle avoidance because the base is fixed.

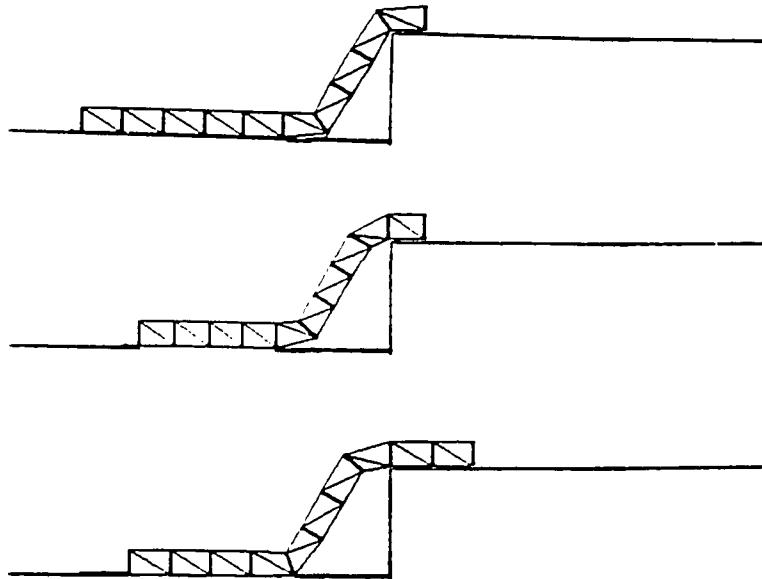
Figure 7.6 shows purely longitudinal stationary wave locomotion over a wall. The terrain curve in this case is piecewise linear, including three line segments: an initial horizontal line, a “ramp” line, and a final horizontal line. In this example,  $l(s, t)$  is defined as in Equation (7.2.1b) with

$$a_s(t) = 1 - \alpha + 2\alpha \sin^2(\pi t/t_p), \quad (7.3.7a)$$

and

$$d(t) = 2\alpha \sum_{i=0}^{N_{cyc}} \left[ H(t - it_p) + \cos^2(\pi t/t_p) W(t - it_p, \frac{t_p}{2}, t_p) \right], \quad (7.3.7b)$$

where  $\alpha = 0.2$  and  $t_p = 1$ .



**Figure 7.6:** Pure Longitudinal Locomotion over a Wall

## 7.4 Extrinsic Formulation for Extensible Mechanisms

Sections 7.1-7.3 established an intrinsic kinematic formulation for the locomotion of hyper-redundant mobile robots over flat and uneven terrain. While the intrinsic formulation of the previous sections is applicable to both nonextensible and extensible morphologies, in some situations it is easier to formulate extensible robot locomotion *extrinsically*. Curves describing the mechanism geometry need not be parametrized intrinsically if the nonextensibility constraint is not imposed. Instead, it is natural to describe the kinematics of extensible mobile robots relative to the terrain.

Consider a curve which represents a time dependent terrain with planar profile. The terrain profile is again parametrized in an  $x_1$ - $x_2$  coordinate system by its arc length,  $\mu$ , and every point on this terrain curve is represented by  $\bar{T}(\mu, t) = [T_1(\mu, t), T_2(\mu, t)]^T$  for the interval  $\mu \in [\mu_{-\infty}, \mu_{+\infty}]$ .  $\bar{T}(\mu, t)$  is defined such that

as  $\mu$  increases, the area to the left of the curve is solid ground, and the robot is on the right side of the curve, where forward is defined as the direction of the terrain curve tangent. With this parametrization, the outward (upward) unit normal is always of the form:

$$\bar{n}(\mu, t) = \begin{bmatrix} \frac{\partial T_2}{\partial \mu}(\mu, t) \\ -\frac{\partial T_1}{\partial \mu}(\mu, t) \end{bmatrix}. \quad (7.4.1)$$

The robot backbone curve is parametrized with respect to the terrain by equating the backbone curve parameter to the terrain arc length:  $s = \mu$ . The backbone curve of the robot is defined relative to the terrain as:

$$\bar{x}(s, t) = \bar{T}(s - s_b(t), t) + \rho(s - s_b(t), h(t), t)W(s - s_b(t), s_0(t), s_1(t))\bar{n}(s - s_b(t), t), \quad (7.4.2)$$

for  $s \in [s_b(t), s_f(t)]$ , where  $s_f(t)$  and  $s_b(t)$  respectively denote the front and back of the backbone curve with respect to a reference value of arc length on the terrain curve. The function  $\rho(s, h(t), t)$  is defined over the domain  $(s, h, t) \in [s_0(t), s_1(t)] \times [h_{min}, h_{max}] \times [t_{init}, t_{final}]$ , and determines the shape of the locomotion wave.  $s_0(t)$  is the distance measured along the terrain to the back of the wave, and  $s_1(t)$  is the distance to the front of the wave (see Figure 7.7). Note that  $s \neq L(s, t)$ .

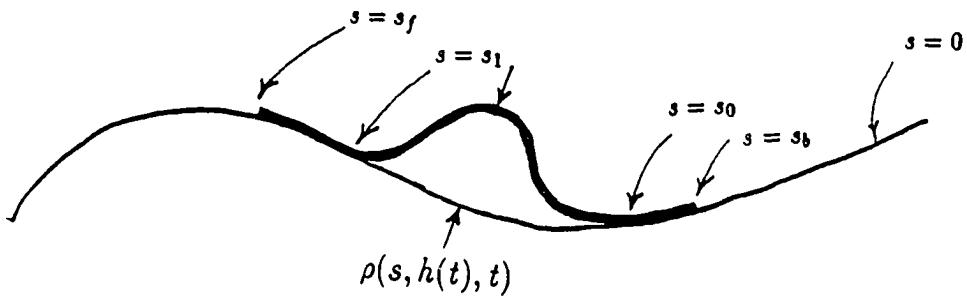


Figure 7.7: Parameters Describing Extrinsic Extensible Robot Locomotion

The interval  $[h_{min}, h_{max}]$  is defined such that  $h_{min} \leq h(t) \leq h_{max}$  where the

function  $h(t)$  is a measure of the amplitude of the locomotion wave. The conditions

$$0 \leq \rho(s, h_{\min}, t) \quad (7.4.3a)$$

and

$$\rho(s, h_{\max}, t) | << \min_{s \in [s_0, s_1]} \frac{1}{\kappa_T(s, t)} \quad (7.4.3b)$$

are imposed to satisfy the constraint of solid terrain, and prevent self intersections of the wave.  $\kappa_T(s, t) = \left| \frac{\partial^2 T}{\partial \mu^2} \right|_{\mu=s}$  is the terrain curvature for  $s \in [s_b, s_f]$ .

Appropriate choices for the functions  $s_b(t)$ ,  $s_0(t)$ ,  $s_1(t)$ , and  $s_f(t)$  further determine the character of the locomotion; e.g., a wave which travels along the robot, one which is stationary with respect to the body, or a combination of the two.

Written intrinsically, extrinsically defined locomotion can be defined as

$$l(s, t) = ([1 - \rho \kappa_T]^2 + \dot{\rho}^2)^{\frac{1}{2}} \quad (7.4.4a)$$

$$\begin{aligned} \kappa(s, t) &= \kappa_T(s, t) [1 - W(s - s_b(t), s_0(t), s_1(t))] \\ &+ W(s - s_b(t), s_0(t), s_1(t)) \left( \frac{\dot{\rho}(2\dot{\rho}\kappa_T - \rho\dot{\kappa}_T) + (1 - \rho\kappa_T)(\kappa_T - \rho\kappa_T^2 + \ddot{\rho})}{([1 - \rho\kappa_T]^2 + \dot{\rho}^2)^{\frac{3}{2}}} \right), \end{aligned} \quad (7.4.4b)$$

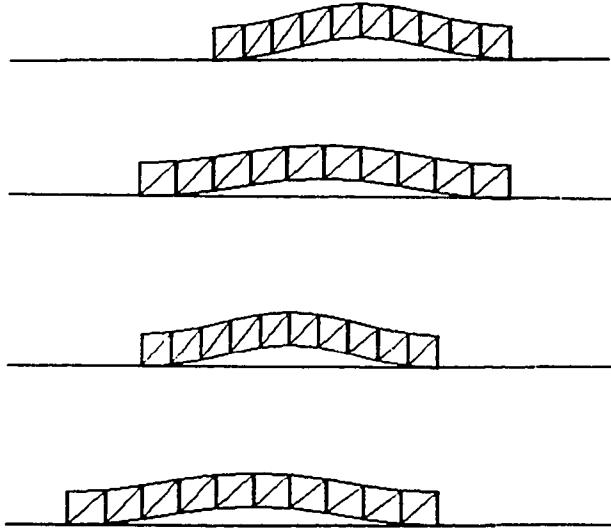
which represents a particular kind of combined bending and extensibility wave. Equation (7.4.4) follows from the general definitions of the curvature and rate of length functions applied to Equation (7.4.2).

The conditions

$$\rho(s_0, h, t) = \rho(s_1, h, t) = \frac{\partial \rho}{\partial s}(s_0, h, t) = \frac{\partial \rho}{\partial s}(s_1, h, t) = 0, \quad (7.4.5)$$

and that  $\rho(s, h, t)$  must be continuously differentiable for all  $(s, h, t) \in [s_0, s_1] \times [h_{\min}, h_{\max}] \times [t_{\text{init}}, t_{\text{final}}]$ , are needed so that the curve representing the robot is continuously differentiable.

Figure 7.8 demonstrates a two phase gait in which the locomotion wave is stationary in the body. The first phase has a duration of  $t_1$ , while the second phase has duration  $t_2 - t_1$ . For this form of locomotion,  $s_0(t) = s_b(t)$  and  $s_1(t) = s_f(t)$ . In the first phase, the wave contracts, i.e.  $ds_0/dt > 0$  and  $ds_1/dt = 0$ . Figure 7.8(a) shows  $\rho(s, h, t)$  increasing in amplitude. In the nonextensible case this would be necessary to preserve the constraint of constant length, but here it is chosen arbitrarily. The second phase is defined by the conditions  $ds_1/dt > 0$  and  $ds_0/dt = 0$ , while  $\rho(s, h, t)$  returns to its original value as shown in Figure 7.8(b). After the completion of the second phase, the cycle repeats. Any point in time can be classified as being in either a contraction or expansion phase, i.e., either  $t \in [I(t, t_2), t_1 + I(t, t_2)]$  or  $t \in [t_1 + I(t, t_2), t_2 + I(t, t_2)]$ . In the example shown in Figure 7.8,  $\bar{T}(\mu, t) = [-\mu, 0]^T$ , which represents flat terrain, though this method will work for any arc length parametrized terrain such that Equation (7.4.3) holds.



**Figure 7.8:** Extrinsically Defined Stationary Wave Locomotion

Figure 7.8 shows configurations of a robot for one choice of the functions  $s_0(t)$ ,  $s_1(t)$ ,  $h(t)$ , and  $\rho(s, h, t)$ . A wide variety of choices for these functions is possible, as long as the aforementioned guidelines are observed. In this particular example,

the functions  $s_0(t)$ ,  $s_1(t)$ , and  $\rho(s, h, t)$  are:

$$\begin{aligned} s_0(t) &= I(t, t_2)L_0/t_2 + L_0 W(t - I(t, t_2), t_1, t_2) \\ &+ L_0 \sin^2 \frac{\pi}{2} \left( \frac{t - I(t, t_2)}{t_1} \right) W(t - I(t, t_2), 0, t_1) \end{aligned} \quad (7.4.6a)$$

$$\begin{aligned} s_1(t) &= I(t, t_2)L_0/t_2 + L_0 W(t - I(t, t_2), t_1, t_2) + L_1 \\ &- L_0 \cos^2 \frac{\pi}{2} \left( \frac{t - t_1 - I(t, t_2)}{t_2 - t_1} \right) W(t - I(t, t_2), t_1, t_2) \end{aligned} \quad (7.4.6b)$$

and

$$\rho(s, h(t), t) = h(t) \sin^2 \pi \left( \frac{s - s_0(t)}{s_1(t) - s_0(t)} \right) \quad (7.4.7)$$

where

$$h(t) = h_0/(s_1 - s_0). \quad (7.4.8)$$

$L_0$  is the length of a single forward stride measured along the terrain,  $L_1$  determines the maximum nominal body length measured along the terrain, and  $h_0$  is a constant which specifies the size of the “hump.” Note how the window functions are used here in the time domain instead of the arc length domain.

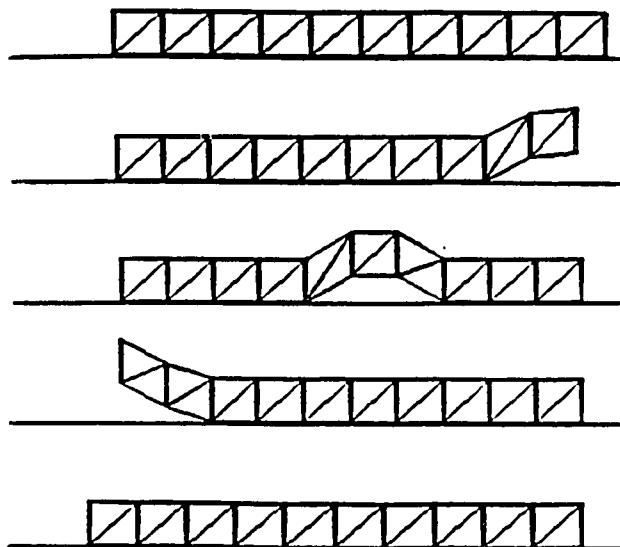


Figure 7.9: Extrinsic Traveling Wave Locomotion

Figure 7.9 demonstrates the three phases of extrinsically defined traveling wave gaits. By definition, the first, second, and third phases of locomotion respectively

have durations  $t_1$ ,  $t_2 - t_1$ , and  $t_3 - t_2$ . In the first phase, which occurs during each time interval  $t \in [I(t, t_3), t_1 + I(t, t_3)]$ , the wave starts to traverse the length of the robot, but the whole wave is not yet in the robot, i.e.,  $s_0(t) < s_b(t) \leq s_1(t)$ . This phase lifts the rear of the robot to initiate a traveling wave. In the second phase, which occurs when  $t \in [t_1 + I(t, t_3), t_2 + I(t, t_3)]$ , the wave is fully contained in the robot, i.e.,  $s_b(t) \leq s_0(t) < s_1(t) \leq s_f(t)$ . In the third phase, the wave exits the front of the robot. The time interval of this phase is  $t \in [t_2 + I(t, t_3), t_3 + I(t, t_3)]$  and  $s_0(t) < s_f(t) < s_1(t)$ .

Letting  $\omega$  denote wave-speed, the wave is defined such that  $s_1 - s_0 = L_1$ , and the distance the robot progresses with each wave is  $L_0$ . Thus,  $t_1 = L_1/\omega$ ,  $t_2 = 1/\omega$ , and  $t_3 = (1 + L_1)/\omega$ . Expressions for the other relevant variables are:

$$s_1(t) = (1 + L_0)[(t - I(t, t_3))\omega + L_0 I(t, t_3)/t_3], \quad (7.4.9)$$

$$s_0(t) = s_1(t) - L_1, \quad (7.4.10)$$

and

$$\rho(s, h, t) = h \sin^2 \left( \frac{\pi[s - s_0(t)]}{L_1} \right) \quad (7.4.11)$$

where  $h = h_0$  is again a constant which specifies the height of the “hump.” Again, the wave shape, determined by  $\rho(s, h, t)$ , is arbitrary up to the constraints of Equations (7.4.3) and (7.4.5).  $s_0$  and  $s_1$  are defined so that the wave travels with constant speed from back to front of the robot, with the process repeating with a period  $t_3$ . The positions of both ends of the robot, denoted by values of the curve parameter  $s_b$  and  $s_f$ , do not coincide with the front and back ends of the wave in this case. They are defined by

$$\begin{aligned} s_b(t) &= (L_0/t_3)I(t, t_3) + (L_0/t_1)[t - I(t, t_3)]W(t - I(t, t_3), 0, t_1) \\ &\quad + L_0 W(t - I(t, t_3), t_1, t_3) \end{aligned} \quad (7.4.12)$$

and

$$\begin{aligned} s_f(t) &= 1 + (L_0/t_3)I(t, t_3) \\ &\quad + \frac{L_0}{t_3 - t_2}[t - t_2 - I(t, t_3)]W(t - I(t, t_3), t_2, t_3). \end{aligned} \quad (7.4.13)$$

This choice for  $s_b$  and  $s_f$  causes the robot to locally stretch and contract as the wave passes. However, no residual stretching or contracting of the robot is passed on between cycles, i.e., the length of the robot is the same at the times  $t = nt_3$  for all integers  $n$ .

The above choices of the functions  $a, \rho, s_0, s_1, s_b, s_f$  constitute one possible gait of the category of extrinsically defined traveling wave gaits. One can imagine other gaits which use different wave shapes, combine stationary and traveling waves, or incorporate multiple simultaneous waves.

## 7.5 Applications to Grasping

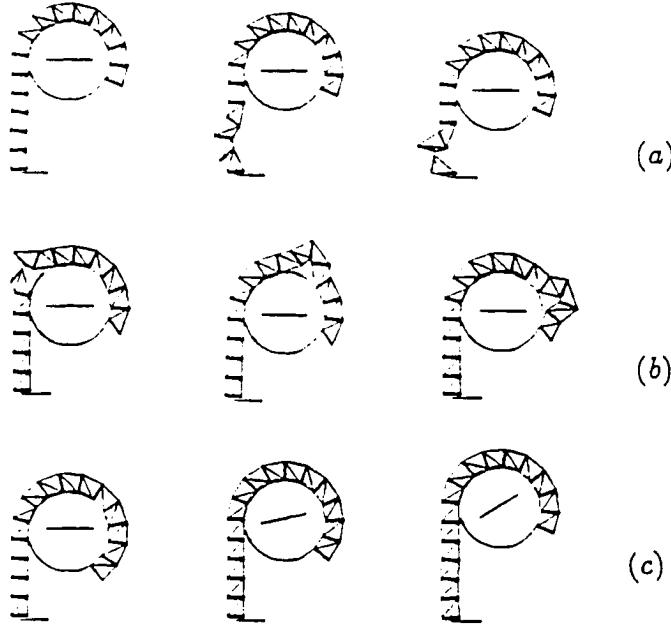
This section applies some of the locomotion ideas to grasping and object reorientation with hyper-redundant manipulators. The hyper-redundant robot, which in this case is fixed to a stationary base, wraps around the solid object to be grasped. In previous work, [Pe88,PeS89], “massively redundant” tentacle grasping of objects has been considered. Here, we consider the novel combination of a hyper-redundant grasp with a locomotion wave used to reorient the object. This form of grasping is hyper-redundant in two senses. First, the robot used in the grasping procedure is hyper-redundant. Second, the grasp has many more contact points than are necessary to maintain stable force closure grasps. In this presentation, only the kinematic aspect of grasping and object reorientation is considered. That is, internal grasp forces required for wrench closure are not computed.

The proposed grasping wave method for manipulating objects consists of the following phases.

- **Shape Initialization:** The hyper-redundant mechanism wraps around the object. This is considered the zeroth phase. The section of the manipulator in contact with the object is termed the *grasp contact segment*, and is specified by the backbone curve segment  $s \in [s_g, 1]$ . The remainder is termed the *noncontact segment*, and is specified by  $s \in [0, s_g]$ .

- **First Phase:** A section of the mechanism,  $s \in [s_0, s_1]$ , in the noncontact segment distorts to a wave form. This could be a bending, longitudinal, or combined wave. As a result, the object may be displaced by a small amount. This phase is shown in Figure 7.10(a).
- **Second Phase:** The wave generated in the first phase travels along the mechanism toward the distal end without changing the position or orientation of the object over which it passes. This phase is shown in Figure 7.10(b), and is similar to traveling wave locomotion where in this case the grasped object is the terrain. When the wave has traveled to the distal end of the manipulator, the grasp contact segment will be longer by an amount  $L_{SL}$ .
- **Third Phase:** The manipulator “unwraps” part of the grasp contact segment from the object by an amount  $L_{SL}$  by straightening the mechanism in the section of the grasp contact segment which is closest to the noncontact segment:  $s \in [s_g, s_g + s_L]$ , where  $s_L$  is defined such that  $L_{SL} = L(s_g + s_L, t) - L(s_g, t)$  (in the nonextensible case  $s_L = L_{SL}$ ). This results in a rotation and displacement of the object, and restores the length of the grasp contact segment to its original value.

In the case where the object to be manipulated is a cylinder, the displacements resulting from the first and third phases cancel, leaving only a net rotation. When the third phase is complete, the cycle repeats, starting with the first phase. This repetition results in repeated object rotations, the magnitude of which depends on the value of  $L_{SL}$  and the size and shape of the grasped object. The cycle shown in Figure 7.10 can be used as depicted to cause counter-clockwise rotations. Alternately, the cycle can be reversed to yield clockwise rotations. For arbitrary objects, net translations can also occur from cycle to cycle. These can be compensated for with the methods developed earlier in this thesis for hyper-redundant manipulator end-effector placement such as the modal approach or configuration based optimal hyper-redundancy resolution.



**Figure 7.10: Hyper-Redundant Grasping of a Cylinder**

The details involved in performing the grasping cycle discussed above are best illustrated with examples. The next subsections consider several examples of the grasping wave with differently shaped wave-forms and a variety of bending and extension waves.

### 7.5.1 A Continuous Nonextensible Example

A continuous bending wave for the reorientation of a disc within the grasp of a hyper-redundant mechanism is discussed in this section. The wave for this example has curvature defined in Equation (7.1.12).

The center of the disc is denoted by the Cartesian coordinates  $(r_c, h_c)$  measured in the  $x_1$ - $x_2$  frame at the base of the manipulator.  $r_c$  is the constant radius of the disc and  $h_c = h_c(t)$  is the height of the center of the disc which is variable in the first and third phases of the grasping cycle, but constant in the second phase.

The distance from the base of the manipulator to the rear of the wave is denoted by  $s_0(t)$ . The constant arc length of this nonextensible traveling wave is  $s_1(t) - s_0(t) =$

$L_w$ , the first section having length  $L_w/4$ , the middle section having length  $L_w/2$ , and the last section having length  $L_w/4$ , just as in Equation (7.1.12) (in that case  $L_w = 1/m$ ).

The arc length measured along the manipulator to a point on the manipulator whose height ( $x_2$  value) in the  $x_1$ - $x_2$  frame is  $h_c(t)$  is denoted by  $h_m(t)$ . In other words,  $h_c(t) = x_2(h_m(t), t)$ . If part of the wave is in the interval  $0 < s < h_m$ , then  $h_c < h_m$ , otherwise they are equal.

It is assumed that at time  $t = 0$ , the shape initialization phase has already occurred. The time intervals of the remaining three grasping phases are:  $t \in [I(t, t_3), t_1 + I(t, t_3)]$ ,  $t \in [t_1 + I(t, t_3), t_2 + I(t, t_3)]$ , and  $t \in [t_2 + I(t, t_3), t_3 + I(t, t_3)]$ .  $s_0(t)$  is constant during the first phase. During the second phase,  $s_0(t)$  is strictly increasing, causing the wave to propagate from the manipulator base to the end-effector with period  $t_3$ :

$$s_0 = [t - t_1 - I(t, t_3)]W(t - I(t, t_3), t_1, t_2)/(t_2 - t_1). \quad (7.5.1)$$

During the grasping wave cycle, the backbone curve curvature function,  $\kappa(s, t)$ , takes the form:

$$\begin{aligned} \kappa(s, t) = & \kappa_1(s, t)W(t - I(t, t_3), 0, t_1) \\ & + \kappa_2(s, t)W(t - I(t, t_3), t_1, t_2) \\ & + \kappa_3(s, t)W(t - I(t, t_3), t_2, t_3) \end{aligned} \quad (7.5.2)$$

where the window functions are used to “section-off” the time intervals in which the curvature functions for each of the three phases are used. In the first and second phases, the curvature function is of the form

$$\kappa_i(s, t) = W(s, h_m, 1)[1 - W(s, s_0, s_0 + L_w)]/r + \kappa_w W(s, s_0, s_0 + L_w) \quad (7.5.3)$$

for  $i = 1, 2$ , where

$$\begin{aligned} \kappa_w(s, t) = & -a_1(t)W(s, s_0, s_0 + L_w/4) + a_2(t)W(s, s_0 + L_w/4, s_0 + 3L_w/4) \\ & - a_3(t)W(s, s_0 + 3L_w/4, s_0 + L_w) \end{aligned} \quad (7.5.4)$$

is the curvature of the wave. Note the signs of the coefficients of this curvature function are opposite to those in Equation (7.1.2) because the wave travels in the opposite sense.

$s_0(t)$  and  $\{a_i(t)\}$  assume different forms in phases 1 and 2. In phase 1,  $s_0 = 0$ , and

$$a_i(t) = \frac{2\pi}{L_w t_1} [t - I(t, t_3)] W(t - I(t, t_3), 0, t_1) \quad (7.5.5)$$

for  $i = 1, 2, 3$ . This causes the segment of the backbone curve in the range  $s_0 < s < s_1$  to form a wave. In the second phase,

$$a_i(t) = W(t - I(t, t_3), t_1, t_2) \left( \frac{2\pi}{L_w} + \int_{t_1+I(t,t_3)}^t \sum_{k=1}^3 \mathcal{J}_{ik}^{-1} (\dot{\mathcal{X}}_k^{des} - \frac{\partial \mathcal{X}_k}{\partial t}) dt \right). \quad (7.5.6)$$

This form for  $\{a_i(t)\}$  causes the traveling wave to "track" the terrain, which in this case is the surface of the grasped object.  $\mathcal{J}_{ik} = \partial \mathcal{X}_i / \partial a_k$ , where the elements of  $\bar{\mathcal{X}}(\bar{a}, t)$  are given in Equation (7.3.2).

The terrain curve varies with time since the grasped object translates and rotates. For this particular example,

$$\begin{aligned} \bar{T}(\mu, t) = & W(\mu, 0, h_c(t)) [0, \mu]^T + W(\mu, h_c(t), 1) \times \\ & \left[ h_c(t) + \frac{1}{r_c} \sin \left( \frac{\mu - h_c(t)}{r_c} \right), h_c(t) + \frac{1}{r_c} \left( 1 - \cos \left( \frac{\mu - h_c(t)}{r_c} \right) \right) \right]^T \end{aligned} \quad (7.5.7)$$

which consists of a straight section for  $0 \leq \mu < h_c$ , and a circular arc for  $h_c \leq \mu \leq 1$ . The vector  $\bar{\mathcal{X}}^{des}$  in Equation (7.5.6) is defined as in Equation (7.3.4) where the value of  $L_{SL}$  remains constant at  $L_{SL} = (1 - 2/\pi)L_w$  throughout the second grasping phase.

In the third phase, the wave has exited the end of the manipulator, and the curvature function assumes the simpler form:

$$\kappa_3(s, t) = W(s, h_m, 1)/r_c, \quad (7.5.8)$$

where  $h_m(t) = h_c(t)$  throughout this phase. This curvature function implements the "unwrapping" of the object as  $h_c(t)$  increases.

This example has shown how a particular type of wave can be generated and applied to locomotion and the grasping-manipulation of objects. It is possible to use similar formulations with different waveforms, or more than one wave at a time for grasping. The next subsection illustrates this notion with another bending wave. In this case, there exists a closed form inverse kinematic solution, thus eliminating the need for Equation (7.5.6) and the associated computations.

### 7.5.2 A Nonsmooth Pure Bending Wave Example

The grasping/manipulation scheme is again illustrated with a planar VGT manipulator that manipulates a disc. The backbone curve is chosen to be nonextensible, and thus a pure bending wave is used. For simplicity, a triangular grasping wave is selected, employing the nondifferentiable three mode example used for end-effector positioning in Equations (4.2.12)-(4.2.14). This wave will travel along the backbone curve and must track the boundary of the disc. The distance from the base of the manipulator to the rear of the wave as measured along the backbone curve is denoted by  $s_0(t)$ . The arc length of the traveling wave is again  $s_1(t) - s_0(t) = L_w$ . The triangle wave has the curvature function:

$$\kappa_w(s, t) = a_1 \delta(s - s_0) + a_2 \delta\left(s - \frac{1}{2}(s_0 + s_1)\right) + a_3 \delta(s - s_1) \quad (7.5.9)$$

where  $\delta(\cdot)$  is the Dirac delta function.  $\{a_1, a_2, a_3\}$  provide sufficient kinematic freedom for the grasping wave to track the disc boundary.  $s_0 = s_0(t)$  specifies the grasping wave progression. Closed form inverse kinematics result from this choice of curvature function, thereby reducing the computational cost.

In the first grasping phase (see Figure 7.11(a)), a portion of the non-contact segment contracts to form a wave while tracking a line segment directed from the base to the grasped object. During this phase,  $a_1 = a_3 = \hat{\alpha}$ ,  $a_2 = -2\hat{\alpha}$ , where  $\hat{\alpha} = \hat{\alpha}(t)$  specifies the slope of the wave. At the beginning of the wave formation,  $\hat{\alpha}(t_0) = 0$ , and at the end,  $\hat{\alpha}(t_1) = \alpha$ . For this wave shape,  $L_{SL} = (1 - \cos \alpha)L_w$ . Because the resulting backbone curve is nonsmooth, the modified fitting procedure used for nonsmooth backbone curves discussed in Chapter 4 is used here.

The above procedure can be altered so that no translation occurs. This is achieved by combining the first and third phases discussed at the beginning of Section 7.5 so that the translations of the object which occurred between phases cancel. This is shown in Figure 7.11.

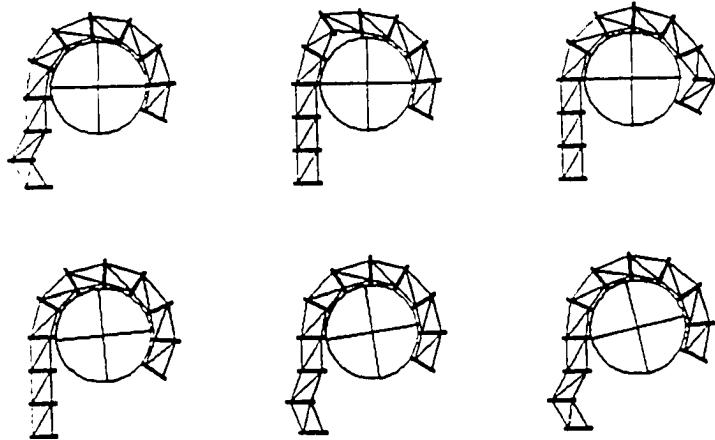


Figure 7.11: Turning an Object Fixed to an Axe

### 7.5.3 Extensible Wave Examples

In this subsection, longitudinal motions are used to define expansion and compression waves which have the same effect on the grasping/manipulation problem as the bending waves presented earlier. Longitudinal waves are analytically easier to manipulate than bending waves. Assume that the preshaping phase has been completed and the “terrain” curve,  $\bar{T} = \bar{T}(\mu, t)$ , is known. A pure longitudinal pulse wave is represented by

$$l(s, t) = 1 + \epsilon_0 W(s, s_0(t), s_1(t)). \quad (7.5.10)$$

When  $\epsilon_0 < 0$  this is a compression wave. When  $\epsilon_0 > 0$  this is an expansion wave.

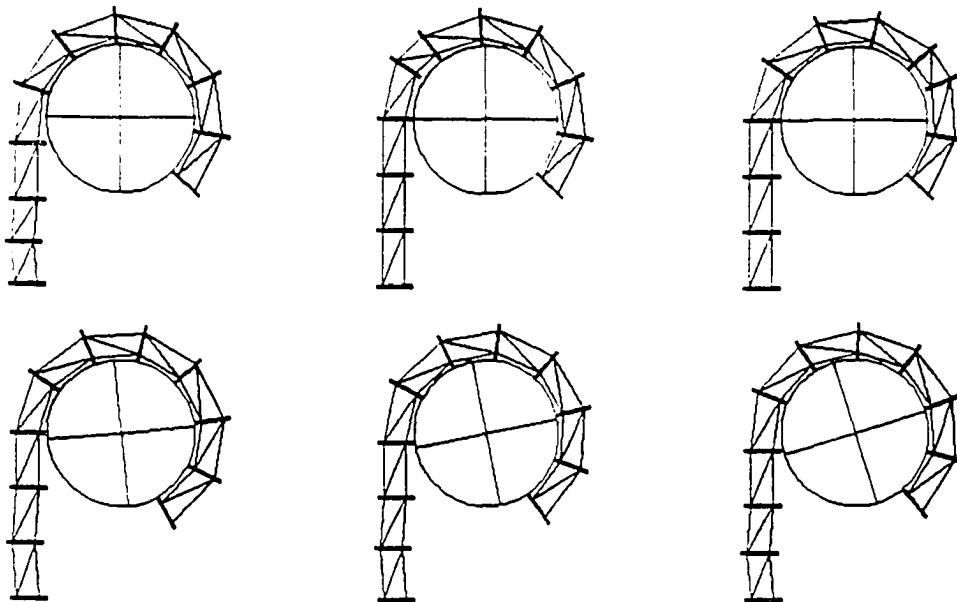
In order for the manipulator not to rub against the grasped objects while longitudinal waves pass, a small amount of bending is always included with longitudinal traveling waves. This is most easily accomplished by using the extrinsic approach

of Section 7.4. For the following examples, Equation (7.5.10) holds *approximately* with an infinitesimal extrinsically defined wave of the form:

$$\begin{aligned}\rho(s, \nu, t) = & \frac{\nu(s - s_0 + s_e)}{s_e} W(s, s_0 - s_e, s_0) + \nu W(s, s_0, s_1) \\ & + \frac{\nu(s_1 + s_e - s)}{s_e} W(s, s_1, s_1 + s_e).\end{aligned}\quad (7.5.11)$$

This is a symmetric trapezoidal waveform with base width  $s_1 - s_0 + 2s_e$ , top width  $s_1 - s_0$ , and height  $0 < \nu \ll 1$ . The value of  $s_e$  determines how gradually or abruptly a traveling wave will separate from a grasped object for fixed wave speed. The smoothness constraints in Equation (7.4.5) have been relaxed in this example.

Introducing an extrinsically defined wave alters the functional relationship between the curve parameter  $s$  and arc length  $L(s, t)$ . Therefore Equation (7.5.10) will deviate from the actual  $l(s, t)$ . The dependence of this deviation on  $\rho$  is given by Equation (7.4.4a). However, by choosing  $\nu$  in Equation (7.5.11) to be infinitesimal, the extrinsic effects on the backbone curve parametrization become negligible, while still implementing the desired effect of preventing the traveling wave from contacting the object being grasped.



**Figure 7.12: Compression Wave Grasping of a Cylinder**

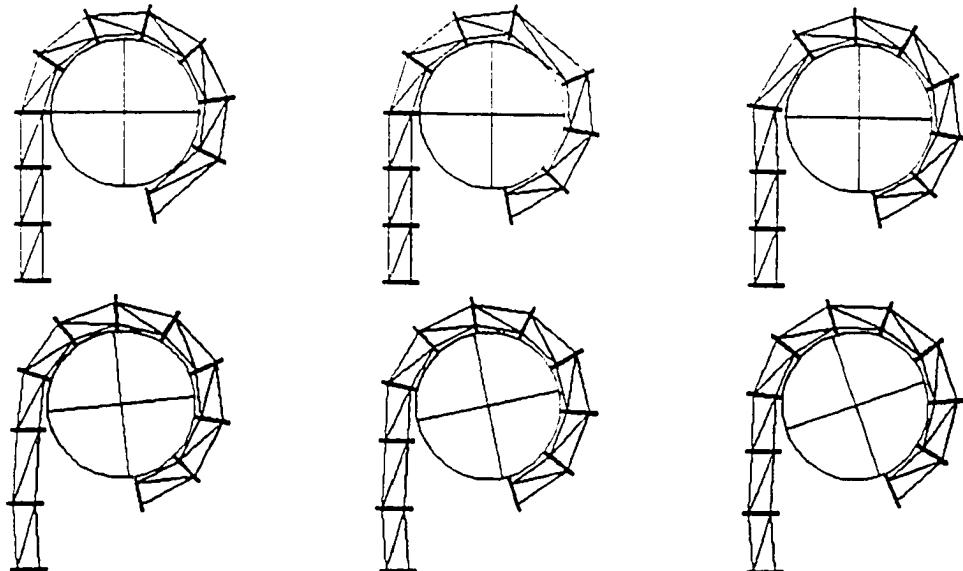


Figure 7.13: Expansion Wave Grasping of a Cylinder

Note that unlike bending and compression waves, expansion waves turn objects in the direction that the waves travel. In other words, waves originate at the distal end of the manipulator and travel towards the base to achieve the same effect as bending or compression waves traveling in the opposite direction. Both types of object manipulation schemes are shown in Figures 7.12 and 7.13.

#### 7.5.4 Dual Arm Grasping

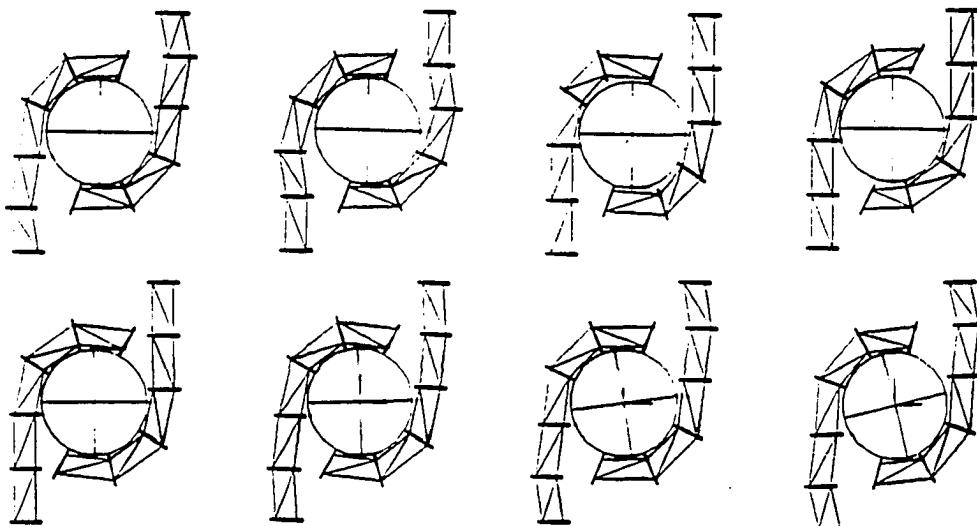


Figure 7.14: Dual Arm Grasping of a Cylinder

Symmetries of the grasping process can be exploited for dual arm hyper-redundant grasping and manipulation of objects as seen in Figure 7.14. While bending waves are shown, any of the waveforms used in this section is acceptable for dual arm grasping as well. In practice, dual arm grasping may be more useful because the preshaping phase of the grasp cycle is less problematical.

## 7.6 Gaits Derived from Linkage Kinematics

As illustrated in Chapter 4, the kinematics of many linkages can be mimicked with nonsmooth backbone reference sets. In this section, locomotion gaits based on the kinematics of planar linkages are developed. These gaits are defined by curvature functions of the form:

$$\kappa(s, t) = \sum_{i=1}^n q_i(t) \delta(s - \frac{i}{n}). \quad (7.6.1)$$

The functions  $\{q_i(t)\}$  are derived using the kinematics of four and five-bar linkages.

Figures 7.15 and 7.16 show a 10-link revolute-jointed hyper-redundant robot passively assuming the shape of the ground, except for a wave made up of a small number of robot links. In this idealized model, the robot configuration is stable because of gravitational and frictional effects. For this example, the traveling wave is restricted to groups of adjacent links which form a four or five-bar linkage, with the ground forming one of the links. In general, the traveling wave may consist of six or more links, though gaits with a large number of links are not necessarily advantageous. Two gait parameters are used to describe this type of locomotion: the number of links which define the wave, and a measure of the wave shape. In this analysis, wave shapes are a function of the steepness of the angle which the back of the wave makes with the ground. This angle is denoted by  $\alpha$  in Figures 7.15 and 7.16.

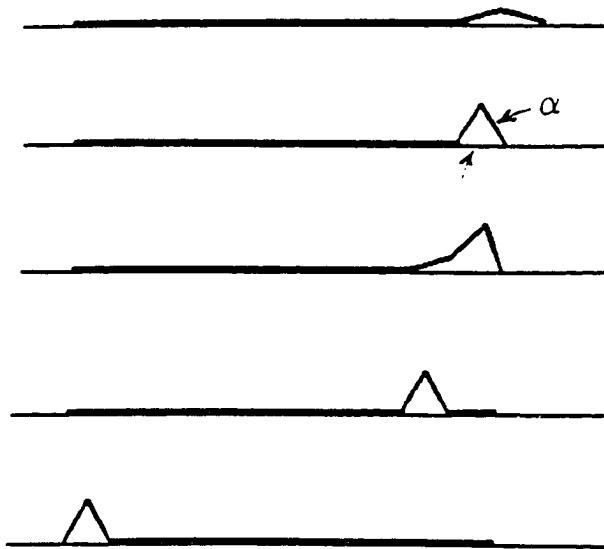


Figure 7.15: Serial Chain Robot Locomotion: Four Bar

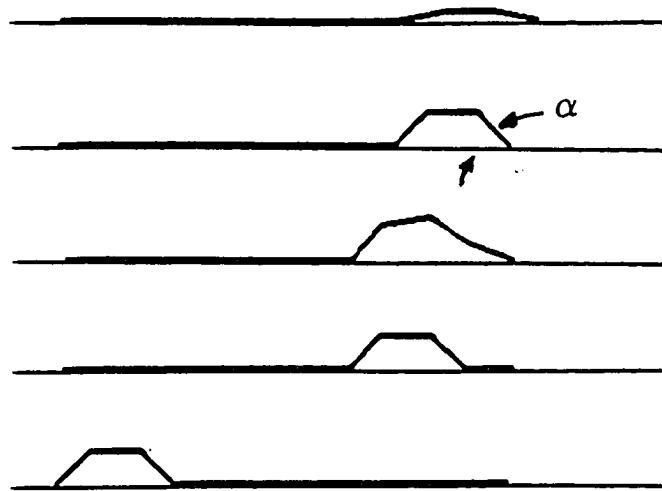


Figure 7.16: Serial Chain Robot Locomotion: Five Bar

In the examples considered here, the ground is flat, and so all the joint angles are zero except those involved in the traveling "mechanical wave." These angles are derived from appropriate linkage kinematics. This locomotion scheme consists of

three phases for both the four and five bar gaits:

- In the first phase, the back of the robot curls to form a wave. This is similar in nature to the SWAV locomotion of Section 7.1.1.
- In the second phase, the wave generated in the first phase travels along the mechanism from back to front.
- In the third phase, the wave exits the front of the robot, leaving the robot with its initial shape, but having traversed a small distance.

### 7.6.1 A Four-Bar Linkage Gait

The gait shown in Figure 7.15 is now described quantitatively. In the first phase,  $0 \leq t - I(t, t_3) < t_1$ , only the first three links are used to form a wave at the back of the robot. The joint angles of these links are given by:

$$q_1 = \alpha[t - I(t, t_3)]/t_1; \quad q_2 = -2\alpha[t - I(t, t_3)]/t_1; \quad q_3 = \alpha[t - I(t, t_3)]/t_1. \quad (7.6.2)$$

This first phase determines the initial conditions for the shape of the wave. The second phase, denoted by the interval  $t_1 \leq t - I(t, t_3) < t_2$ , is broken down into  $n-2$  subphases. In the  $i^{th}$  subphase the wave is treated as three bars of a four bar linkage specified by the joint angles  $q_i, q_{i+1}, q_{i+2}, q_{i+3}$ , with the fourth bar being the terrain. All other joint angles for this example are zero, although this could be generalized so that the robot passively assumes the shape of an arbitrary terrain. The  $i^{th}$  subphase is confined to the time interval  $t_1 + (t_2 - t_1)\frac{i-1}{n-3} \leq t - I(t, t_3) < t_1 + (t_2 - t_1)\frac{i}{n-3}$ . In the first subphase of the second phase, denoted by  $i = 1$ , the initial configuration of the wave is specified by the first phase of the locomotion evaluated at  $t = t_1$ . The kinematics of four-bar linkages is then used to transform the configuration of this wave so that the “hump” which is initially on the right side of the four-bar linkage is moved to the left side. This is shown in Figure 7.15. Because of symmetry, the final configuration in the variables  $q_1, q_2, q_3, q_4$  provides an initial configuration for the joint angles  $q_2, q_3, q_4, q_5$ , with  $q_5 = 0$ , which is the same as the initial configuration established by the first phase for the joint

angles  $q_1, q_2, q_3$ , with  $q_4 = 0$ . The process is then repeated in each of the  $i$  subphases. When the  $i^{th}$  phase is over, an initial configuration for the joint angles  $q_{(i+1)}, q_{(i+1)+1}, q_{(i+1)+2}$ , with  $q_{(i+1)+3} = 0$  is provided and the process continues through each of the  $n - 2$  subphases. The details of the kinematics of the wave are exactly those of a four-bar linkage with the top bars having length  $\frac{1}{n}$  and the bottom having length  $\frac{1+2\cos\alpha}{n}$ .

In the third phase,  $t_2 \leq t - I(t, t_3) < t_3$

$$q_{n-1} = \alpha[1 - (t - I(t, t_3) - t_2)/(t_3 - t_2)] \quad (7.6.3a)$$

$$q_n = -2\alpha[1 - (t - I(t, t_3) - t_2)/(t_3 - t_2)] \quad (7.6.3b)$$

extends the front end of the robot as the wave is flattened. As a result, on flat ground the configuration is the same as the initial configuration, only now the robot has progressed by a distance

$$L_{SL} = \frac{2}{n}(1 - \cos\alpha). \quad (7.6.4)$$

### 7.6.2 A Five-Bar Linkage Gait

Figure 7.16 shows another possible gait. Again the locomotion is broken into three phases. These phases are analogous to those of the previous gait, only now four of the robot's links define each wave instead of three.

In the first phase, denoted by the time interval:  $0 \leq t - I(t, t_3) < t_1$ , the only nonzero joint angles are:

$$q_1 = \alpha[t - I(t, t_3)]/t_1 \quad (7.6.5a)$$

$$q_2 = -\alpha[t - I(t, t_3)]/t_1 \quad (7.6.5b)$$

$$q_3 = -\alpha[t - I(t, t_3)]/t_1 \quad (7.6.5c)$$

$$q_4 = \alpha[t - I(t, t_3)]/t_1. \quad (7.6.5d)$$

The second phase,  $t_1 \leq t - I(t, t_3) < t_2$ , is broken into  $n - 3$  subphases. In the first half of the  $i^{th}$  subphase, the  $i + 1^{st}$  joint angle is held constant at  $q_{i+1} = \alpha$ , and

$q_i, q_{i+2}, q_{i+3}, q_{i+4}$  are determined by using the kinematics of four-bar linkages. In the second half of the  $i^{th}$  subphase, the joint angle  $q_{i+1}$  is “unlocked” and  $q_{i+2} = \alpha$  is enforced. Just as in the previous example, the final states of the joint angles in the  $i^{th}$  subphase determine the initial conditions for the  $i + 1^{st}$  subphase. In the third phase,  $t_2 \leq t - I(t, t_3) < t_3$

$$q_{n-2} = \alpha[1 - (t - I(t, t_3) - t_2)/(t_3 - t_2)] \quad (7.6.6a)$$

$$q_{n-1} = -\alpha[1 - (t - I(t, t_3) - t_2)/(t_3 - t_2)] \quad (7.6.6b)$$

$$q_n = -\alpha[1 - (t - I(t, t_3) - t_2)/(t_3 - t_2)]. \quad (7.6.6c)$$

The stride length is again given by Equation (7.6.4).

## 7.7 Discussion

This chapter modeled the kinematics of hyper-redundant locomotion and grasping using methods of the differential geometry of planar curves and kinematics of planar linkages. Two gait models, which are idealizations of inchworm and caterpillar locomotion, were introduced. These “Stationary-Wave-Amplitude-Varying” and “Traveling-Wave-Amplitude-Constant” gait models were applied to the locomotion of a variable geometry truss robot and an n-link revolute chain. It was shown how these gaits can be used for both extensible and nonextensible modes of locomotion over flat and curved terrain. A novel hyper-redundant fine manipulation scheme based on a “grasping wave” follows naturally from the formulation of locomotion problems.

Mobile hyper-redundant robots have many potential advantages. Neither wheeled nor legged robots can overcome an obstacle on the order of their largest linear dimension. However, hyper-redundant mobile robots can potentially surmount such obstacles. In addition, the calculations required for adequate foot placement of legged robots over complicated terrain present a large computational burden. The inherent stability of the traveling wave locomotion introduced in this chapter

could allow a hyper-redundant robot to traverse rugged terrain with relative ease. Other novel attributes of hyper-redundant robots include mixed operation modes in which the robot simultaneously locomotes, grasps, and manipulates objects. This operation could be achieved by bracing portions of the robot against the ground while using traveling wave techniques for manipulation as well as locomotion. The methods developed in this chapter are an important component needed to implement these complicated behaviors.

**HARDWARE DEVELOPMENT  
AND  
DEMONSTRATION**

## Chapter 8

# A Hardware Implementation

The usefulness of the analytical framework developed in Chapters 2-7 hinges on one important factor: “Can practical hyper-redundant robotic mechanisms be designed and built?” Part of this problem depends on whether or not it is possible to modularize hyper-redundant robot designs, and reduce the complexity of the design problem to a manageable level. This chapter discusses the design and implementation of a thirty degree of freedom variable geometry truss manipulator which was developed to implement, validate, and demonstrate concepts developed in previous chapters.

As discussed in Chapter 1, hyper-redundant robot morphologies can vary widely in mechanical structure and actuation. The selection of a particular morphology depends heavily on the functional and operational requirements of the application domain under consideration. In this chapter, criteria which determined the selection of the current design are discussed. Section 8.1 reviews task requirements which a hyper-redundant robot testbed must demonstrate. Section 8.2 reviews the candidate morphologies from which the VGT design was chosen. Section 8.3 describes the structure and operation of the manipulator and its support system, e.g., actuators, motor controllers, and computer interface. Section 8.4 discusses properties of the current system which generalize to other hyper-redundant sys-

tems. Actual experiments performed with this device are presented in Chapter 9.

## 8.1 Task Requirements

This section reviews task requirements for which the current system was designed. The primary purpose for designing, building, and implementing this hyper-redundant robot testbed was not only to demonstrate algorithms developed in Chapters 2-7, but also to demonstrate applications for which hyper-redundant robots are well suited. To obtain the maximum possible benefit from their highly articulated structures, hyper-redundant robots must not only be able to place their end-effectors at specified locations, but must also be able to:

- maneuver through highly constrained workspaces.
- locomote like inchworms, snakes, and slugs.
- grasp and manipulate objects in a tentacle-like fashion.

These functional requirements arise in the following anticipated application scenarios:

- operations in highly constrained environments, such as: nuclear reactor steam generator maintenance; chemical sampling in buried toxic waste containers; and medical endoscopy.
- tentacle-like grasping and object reorientation such as: enveloping and capturing free floating satellites; replacement of mechanical components in hard to reach areas of an automobile; and complex “whole arm” manipulation experiments.
- novel locomotion in complex environments such as: emergency response vehicles in burning or collapsed buildings; and sensor placement in complicated geological formations, such as lava tubes.

Naturally, no single design can be suitable for all applications. Rather than discuss the morphology-application relation, the next section will focus on the enumeration of broad design classes and their strengths and weaknesses. Sections 8.3 and 8.4 review the selected VGT design in the context of this discussion.

## 8.2 Review of Candidate Morphologies

Hyper-redundant manipulators can be physically implemented using a variety of actuators and structural materials. Examples are pneumatic bellows, rubber gas actuator driven devices, serial chains consisting of a large number of rigid links, or variable geometry trusses. As stated in Chapter 1, these morphologies can be roughly categorized into three main types: (1) *discrete morphologies* which have a large, but finite, number of rigid links connected in series; (2) *continuous morphologies* in which the actuation is distributed over the length of a continuously deformable manipulator; and (3) *modular morphologies* in which the physical structure is viewed as a cascade of closed loop modules. For the most part, the distinction in morphologies arises from the types of actuators chosen, and their means of deployment in the robot's structure.

In continuous morphologies, actuation is distributed through the robot structure, resulting in the ability to continuously deform the robot's geometry. Continuous morphologies include robots constructed of "smart" materials, and pneumatic/hydraulic tubes and bundles of tubes. Continuous manipulators have the most general geometry, and at first glance would appear to be more maneuverable and highly articulated. However, their actual implementation can be problematical. Truly distributed actuation (akin to muscles in a snake) is difficult to engineer in practice, although there are several ongoing research activities in this area [SegWAS92]. Further, pneumatic and hydraulic schemes are not very suitable for locomoting hyper-redundant robots, as it is difficult to carry the pumps, regulators, and other required components in a self-propelled package.

Discrete morphologies are comprised of a finite number of non-distributed actuators. An example of such systems includes serial chain rigid link robots (with revolute and/or prismatic joints). Serial chain rigid link systems are an extension of traditional manipulator design methodology, have a simple kinematic structure, and have simple “fitting” algorithms, as discussed in Chapter 3. However, this morphology has inherently poor mechanical advantage, making it difficult to build one which can support its own weight when fully extended. In addition, serial chain designs are not robust with respect to actuator failure, i.e., if a single actuator in the chain fails, the manipulator will cease to function.

The category of modular morphologies includes variable geometry truss manipulators and a variety of serially connected platform manipulators. Variable geometry truss systems have many advantages for practical hyper-redundant manipulator implementation. VGTs were originally studied for application to large space-structures, such as precision segmented mirrors. Examples in the literature are numerous (See Section 1.1). Traditional VGTs and VGTMs can be differentiated by their actuation requirements. For large space-structures, the primary role of the actuators and control system is to control structural resonances. Thus, these systems have actuators with high bandwidth and very little stroke. Conversely, VGTs for robotics applications require actuators with substantial stroke to effect significant changes in geometry.

Most actuator technologies and implementations fall clearly into one category or the other. Some actuators, such as tendons or shape memory alloys, can be deployed in either a continuous or discrete fashion. For example, tendons can be distributed throughout a hyper-redundant structure to provide continuous or discrete changes in geometry, depending upon the way they are attached to the robot structure. Tendons can either be active or passive. Active tendons can be constructed of contracting shape memory alloys or contracting polymers. Passive tendons transmit tensile forces which are typically supplied by d.c. servomotors. A number of tendon based structures have been suggested and constructed (see

Section 1.1). Tendon based structures can be very light in weight. However, active tendons based on SMAs are typically slow in response. Passive tendons require remotely mounted actuators, and tendon routing becomes very complicated and impractical for the large number of tendons required to implement versatile spatial hyper-redundant robots. Further, nearly all tendon based designs implemented to date do not have complete kinematic functionality. That is, these designs cannot provide twisting motion about the backbone curve tangent, nor do they have a general capability to simultaneously implement both torsional and curvature bending. Consequently, tendon driven structures can only assume a restricted class of shapes.

A VGT design was chosen for the current system because of its inherent mechanical advantage, modularity, and static stability. This is required for several of the envisioned experiments. Sections 8.3 and 8.4 will elaborate on this design, and other relative merits of VGT systems.

## 8.3 System Description

This section describes the hyper-redundant manipulator and system architecture chosen to implement the current hyper-redundant robot testbed. For simplicity's sake, a planar hyper-redundant robot was developed, leaving the development of a spatial system for future work. The primary function of this robot is to demonstrate and validate the analytical work in Chapters 2-7, and serve as the basis for future work in hyper-redundant robot analysis, design, programming, and sensor-based operation. The following subsections describe the manipulator, and the supporting system architecture.

### 8.3.1 Manipulator Description

The particular manipulator shown in Figure 8.1 was designed with the intent of easy assembly and disassembly. Approximately one half of an hour was required to assemble the mechanical components with three people working in parallel at a

comfortable rate. This speed was achieved because the design allows for actuators to be connected in interlocking "chains." These actuator chains are assembled independently by simply inserting pins into bearing structures which are attached to the ends of each actuator. Two of the chains are then used to form the side elements of the truss structure, and the third is laid on top in a "zig-zag" fashion to form the diagonal elements. This assembly procedure is shown schematically in Figure 8.2.

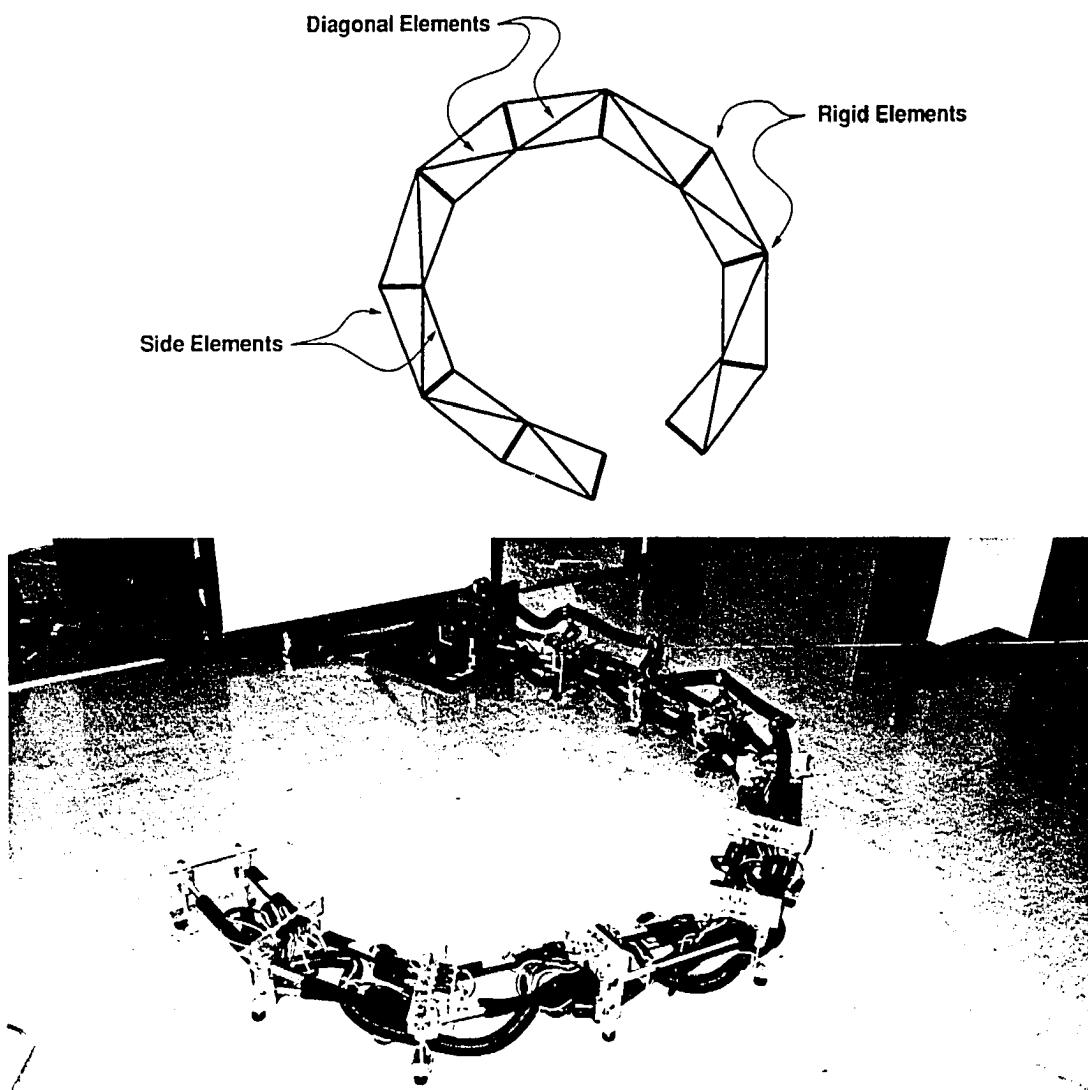
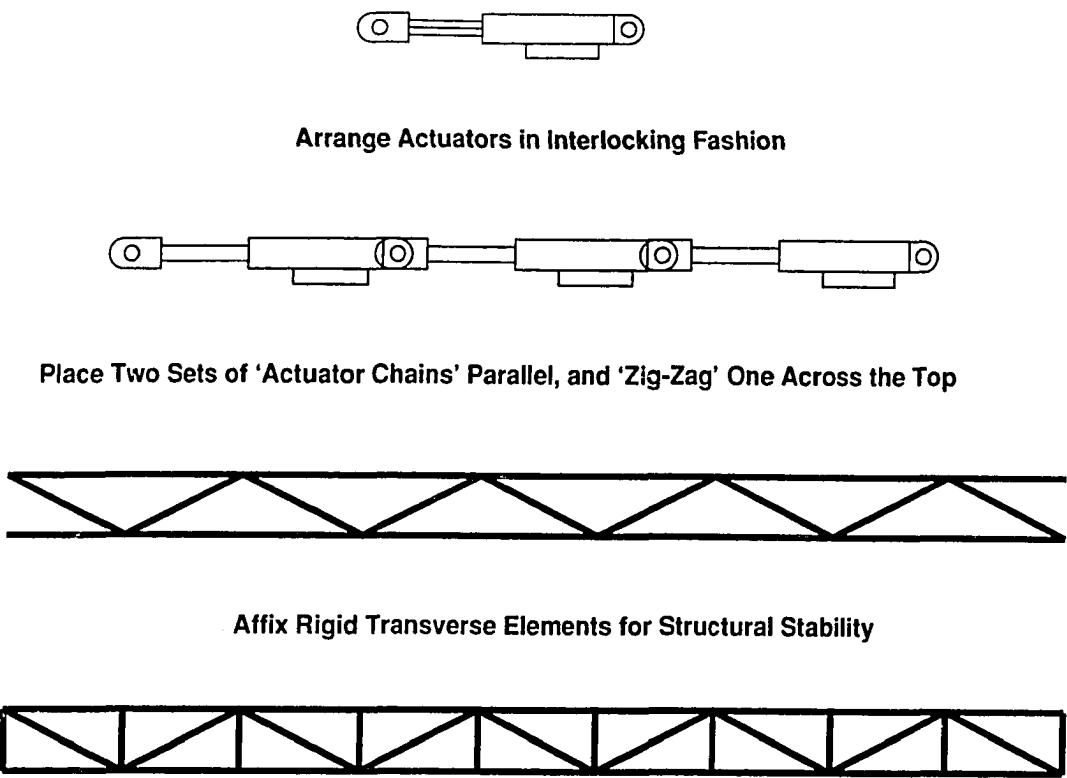


Figure 8.1: Schematic and Photograph of the Current Truss Structure



**Figure 8.2: Assembly-Disassembly Procedure**

To disassemble, the pins which fix the relative position of each actuator in the overall structure can be removed, and any particular actuator can be replaced. When dual-arm or half-arm tasks are to be demonstrated in Chapter 9, this 10-bay truss is “split” into two 5-bay units (which are still hyper-redundant).

Each truss module (Figure 8.3) contains three prismatic joints. In effect, each bay is a planar parallel manipulator. The prismatic actuators consist of d.c. servomotors and lead screw drives. The actuators can vary in length from a minimum of 12 inches to a maximum of 19 inches, and can generate a force of 75 pounds during motion, and sustain 225 pounds statically. Each actuator is equipped with a linear potentiometer to measure its absolute displacement. The accuracy of the feedback potentiometer and the backlash in the lead screw are both approximately 1 percent of maximum extension.

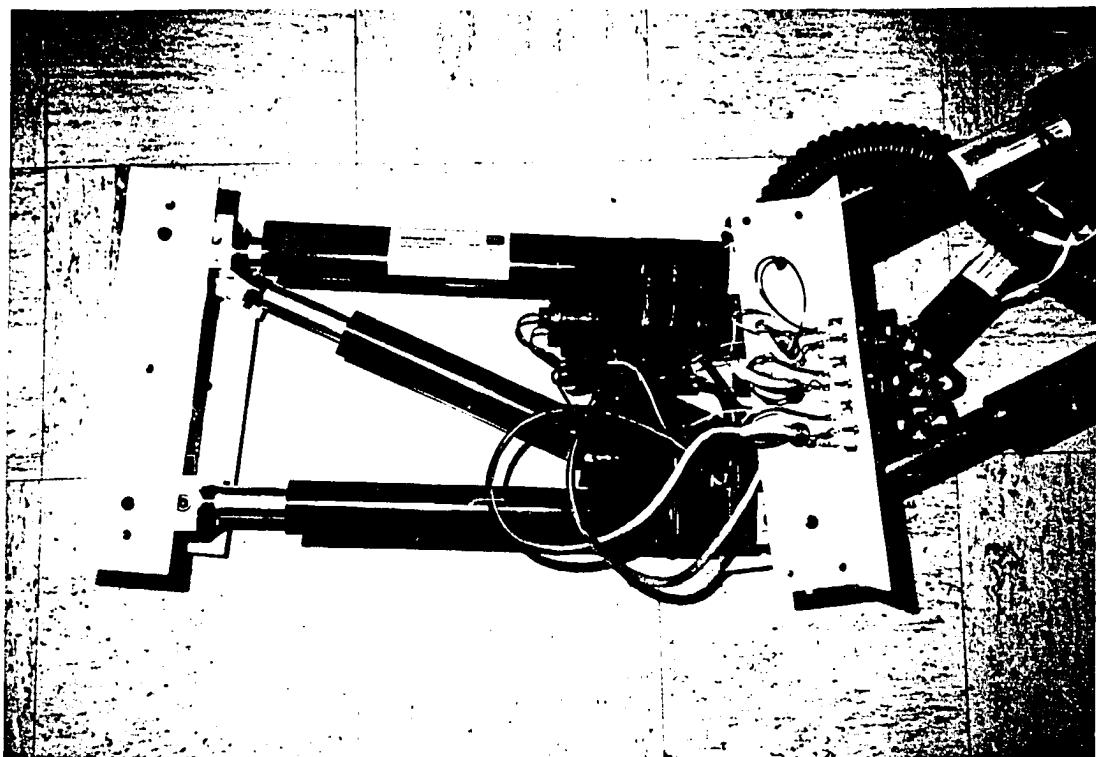


Figure 8.3: Photograph of One Bay of the Truss

For manipulation and grasping experiments, ball castors are attached to the fixed elements of the modules in order to allow low friction motion over the laboratory floor, which is the plane of manipulation. For locomotion experiments, the manipulator is detached from its base, and turned on its side. Each module is equipped with rubber faced ratchet wheels which permit rolling in one direction only, which is required for stationary wave locomotion. For traveling wave locomotion, the wheels are locked in place, and the wheels' rubber surfaces are used as passive feet. Foam covered extensions adorn the opposing side and serve as soft finger contacts for grasping experiments.

Let the length of a module from base to top (as measured in a reference configuration) divided by the width of the module (distance between actuator pivot points) be called the *module aspect ratio*. The fixed elements of the truss modules are designed so that each module's aspect ratio can be varied for different applica-

tions. A very high module aspect ratio leads to greater dexterity, but has reduced mechanical advantage. High aspect ratios are advantageous for obstacle avoidance in tightly constrained spaces. Conversely, low aspect ratios provide greater mechanical strength at the expense of dexterity. For most applications, the module aspect ratios are uniformly sized, though they can be nonuniform when necessary. For example, in manipulation, the modules closest to the base are subjected to greater forces and moments. Their aspect ratios can be reduced to improve their strength.

### 8.3.2 Real-Time Control System

Figure 8.4 shows a schematic of the current system architecture.

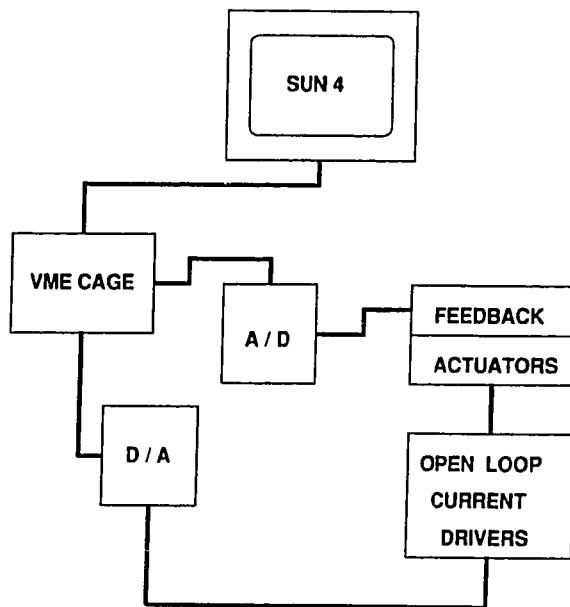


Figure 8.4: System Architecture

The robot system is controlled by a multiprocessing computer, which consists of Heurikon 68030 processing boards in a VME bus, which is in turn connected to a Sun Microsystems 4/360 computer. The WindRiver Systems VxWorks software system provides operating system and system management functions. The 68030 processors compute the kinematic, path planning, sensor processing, and task

level coordination functions. Low level control of the actuator positions is also performed by 68030 processors. The Sun computer is used primarily for user interface and data storage.

The digital-to-analog and analog-to-digital conversion hardware is commercially available. The actuator current amplifiers were built in house, using a standard amplifier circuit with op-amps and push-pull transistor pairs. Each actuator draws up to 2 amperes of current at 24 volts during operation.

## 8.4 Lessons Learned the “Hard”ware Way

As demonstrated by this design, hyper-redundant robots have several inherent design and operating advantages. These are enumerated below.

### Modularity:

Hyper-redundant robot design is often amenable to simple modular implementation. In this case, each module consists of a planar parallel manipulator. For dual-arm experiments, the 10-bay truss can be “split” into two 5-bay units in a matter of minutes. For spatial manipulators, the module geometry could be a Stewart platform.

### Duty Cycle:

In many operational scenarios, only a small fraction of the robot actuators needs to be used, minimizing their duty cycle and prolonging their life. Even so, such systems are highly fault tolerant with respect to numerous actuator failures. For the planar thirty degree of freedom manipulator shown in Figure 8.1, 90 percent of the actuators can fail, and the system will retain the ability to position and orient objects at the end-effector (though obstacle avoidance capability and workspace volume can be severely reduced).

**Assembly and Maintenance:**

Assembly and maintenance of this hyper-redundant manipulator is extremely simple. The entire mechanism can be assembled from its constituent parts in less than one half hour (with three people working in parallel). Partial disassembly to replace actuators is trivial. The pins which fix the relative position of each actuator in the overall structure can be removed, and an actuator can be replaced in less than 5 minutes.

**Strength-to-Weight Ratio:**

Such systems can also be weight efficient. The thirty degree-of-freedom robot can contract to a minimum length of 12 feet and extend to a maximum length of 16 feet. Yet the robot weighs less than 55 kg. Also, simple mechanical adjustments can alter the mechanical advantage properties of the modules to suit different needs.

Of course, hyper-redundant robots have obvious disadvantages, and are best suited to niche applications. Small errors in each actuator position can accumulate into large base-to-tip position errors. Thus, hyper-redundant manipulators are at a disadvantage for precise positioning tasks, unless they are instrumented with numerous sensors and equipped with sensor-based control strategies which adjust for local errors. However, inspection and locomotion tasks do not require ultra-precise accuracy.

## 8.5 Discussion

This chapter discussed the system requirements and design of a thirty degree-of-freedom planar hyper-redundant robot. The benefits and drawbacks of other hyper-redundant morphologies were also enumerated. This system was developed to validate the analytical methods presented in Chapters 2-7. Construction of this system illustrates that hyper-redundant manipulator designs need not be complex.

It is hoped that the motion planning framework and design criteria developed here for hyper-redundant systems will bring the concept of hyper-redundancy closer to a level of practical implementation. The following chapter reviews a series of experiments which demonstrate the operation of the robot described in this chapter using the algorithms of Chapters 2-7.

## Chapter 9

# Demonstrations of Task Implementation

This chapter is an exposition of the task scenarios which have been tested with the thirty degree-of-freedom hyper-redundant variable geometry truss described in the previous chapter. The sections of this chapter contain photographs of the tasks explained below.

**Range of Motion:** Figures 9.1-5 show several configurations of the manipulator designed and built according to the specifications enumerated in the previous chapter. The ability of the mechanism to extend, contract, and bend is illustrated.

**Obstacle Avoidance:** The theory developed in Chapter 6 is implemented in Figures 9.6-9.9. In fact, the same obstacle-avoiding set of backbone curves used in the examples of Chapter 6 is used in this demonstration. In Figures 9.6-9.9, a time progression of the manipulator entering an obstacle field is shown.

**Locomotion:** The theory for hyper-redundant robot locomotion developed in Chapter 7 is demonstrated in Figures 9.10-9.12. Figure 9.10 shows stationary bending wave locomotion over flat terrain. Figure 9.11 shows stationary longitudinal wave locomotion. Figure 9.12 shows longitudinal traveling wave locomotion.

**Grasping and Object Reorientation:** Figures 9.13 and 9.14 show how a hyper-redundant manipulator can be used as a grasping device. In Figure 9.13, the whole

manipulator is used to recover a mock satellite from orbit. In Figure 9.14, dual-arm satellite recovery is demonstrated.

## 9.1 Mechanism Extension/Bending Limits

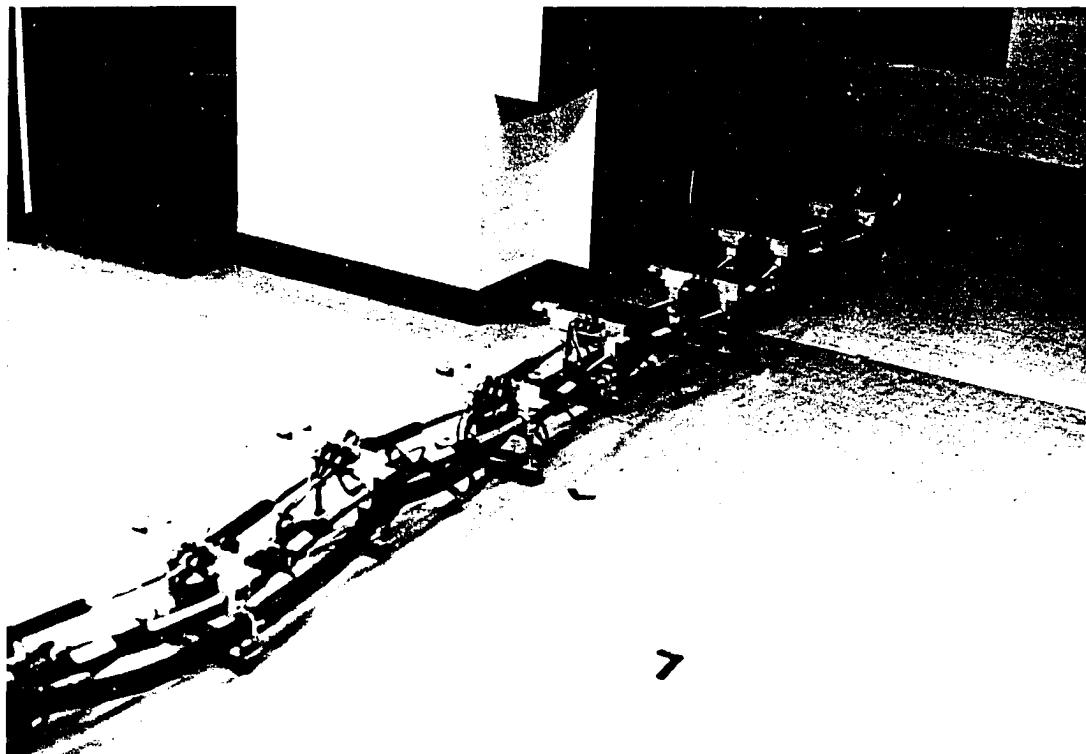


Figure 9.1: Manipulator Fully Extended

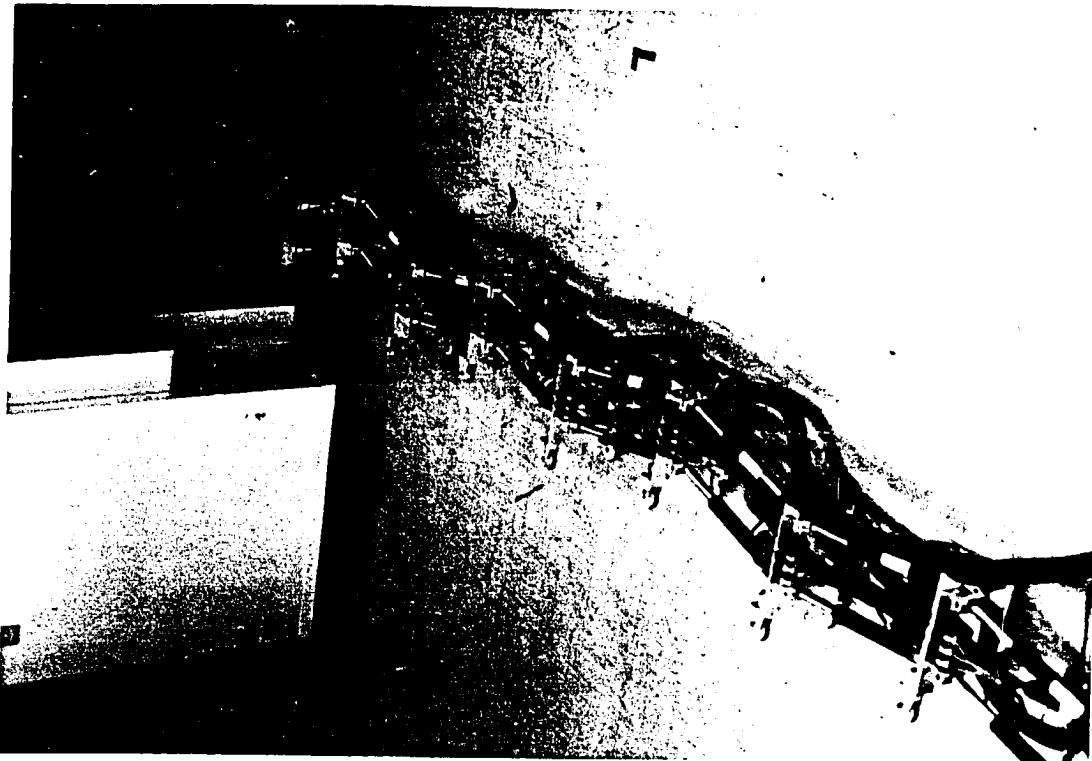


Figure 9.2: Manipulator Partially Contracted

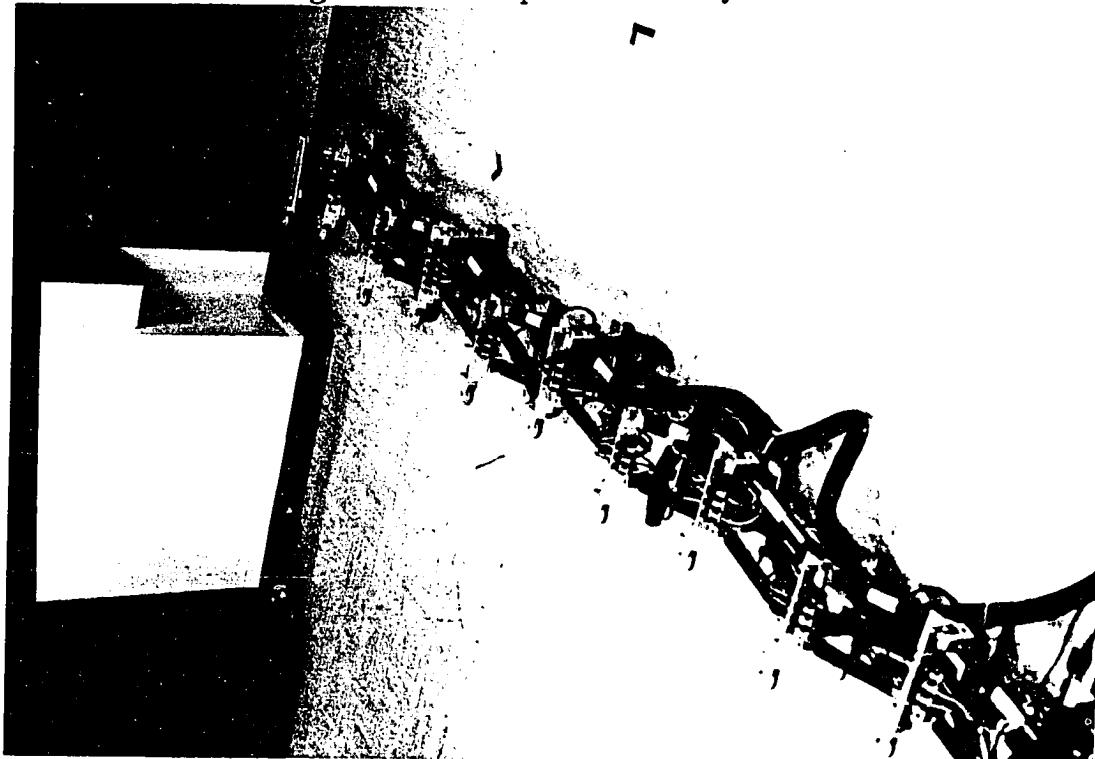


Figure 9.3: Manipulator Fully Contracted

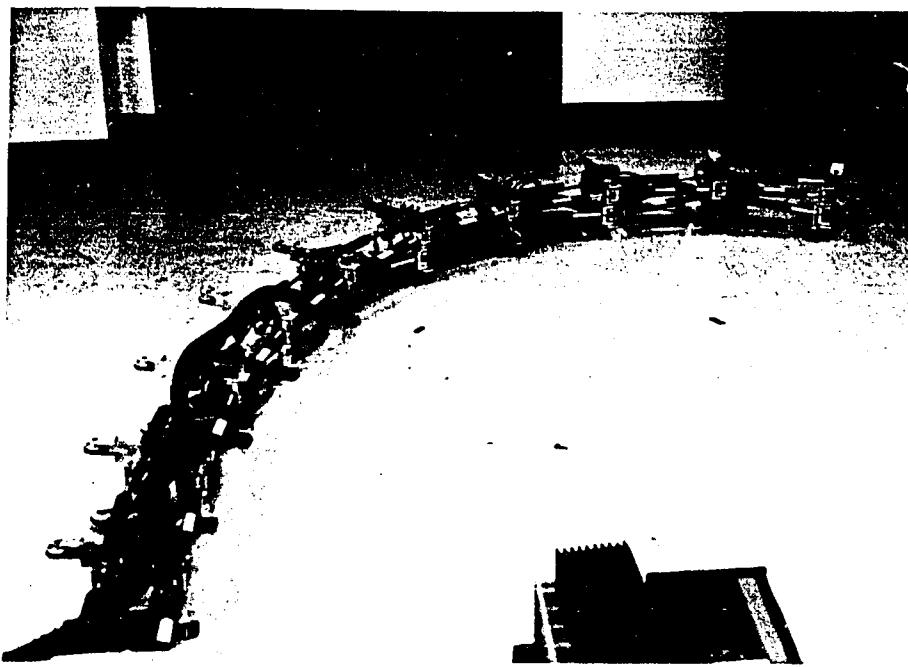


Figure 9.4: Partially Curled Manipulator

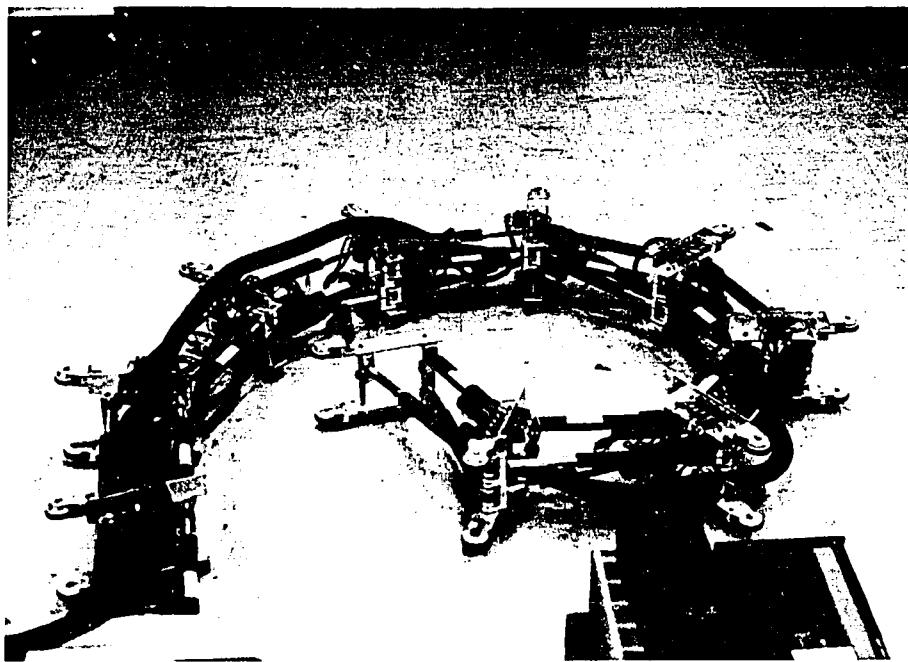


Figure 9.5: Fully Curled Manipulator

## 9.2 Obstacle Avoidance

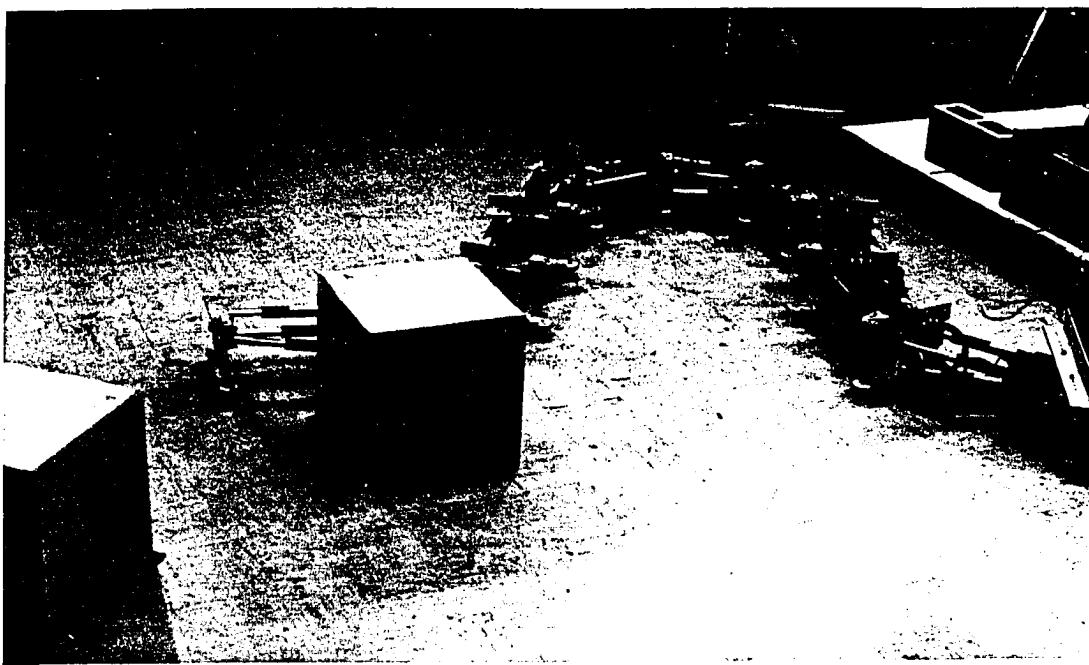


Figure 9.6: Manipulator Entering Obstacle Field

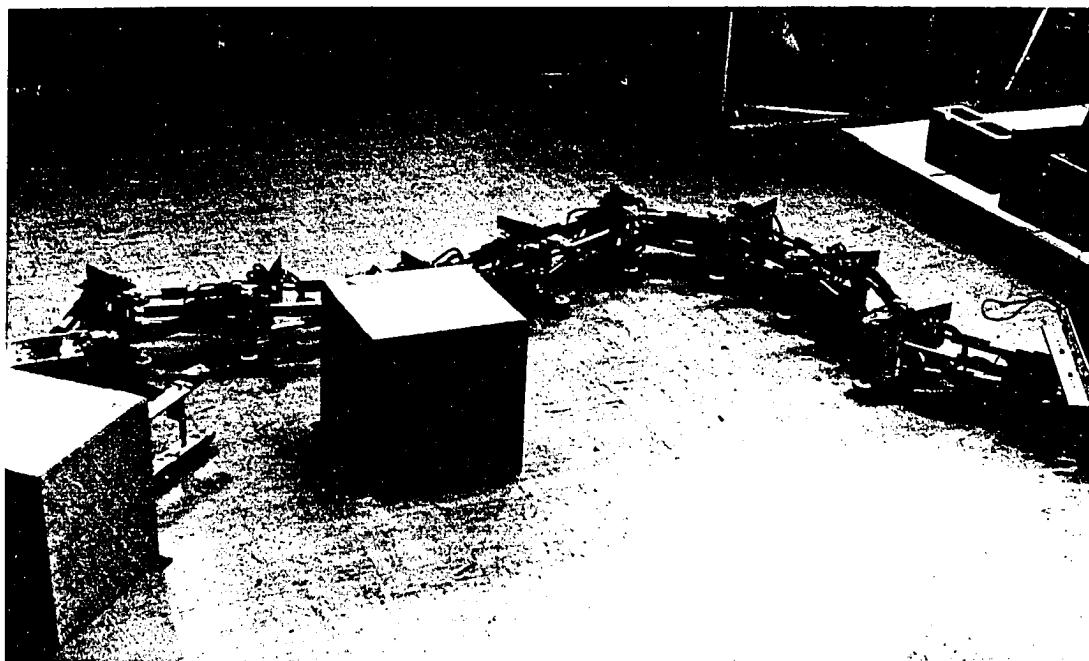


Figure 9.7: Manipulator Partially in Obstacle Field

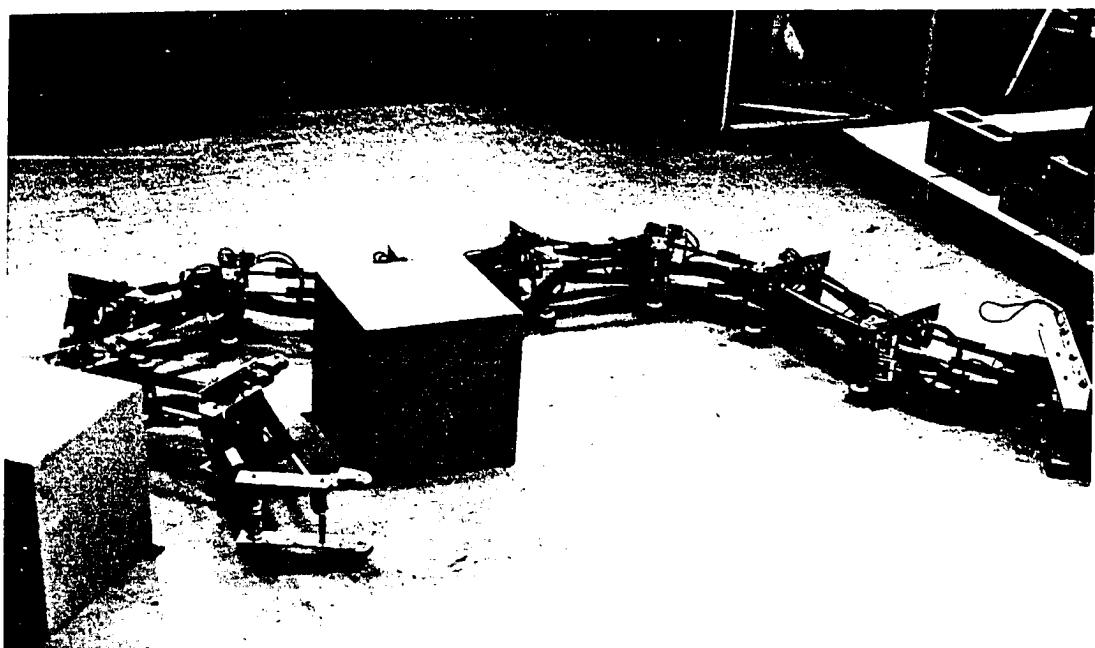


Figure 9.8: Manipulator Mostly in Obstacle Field

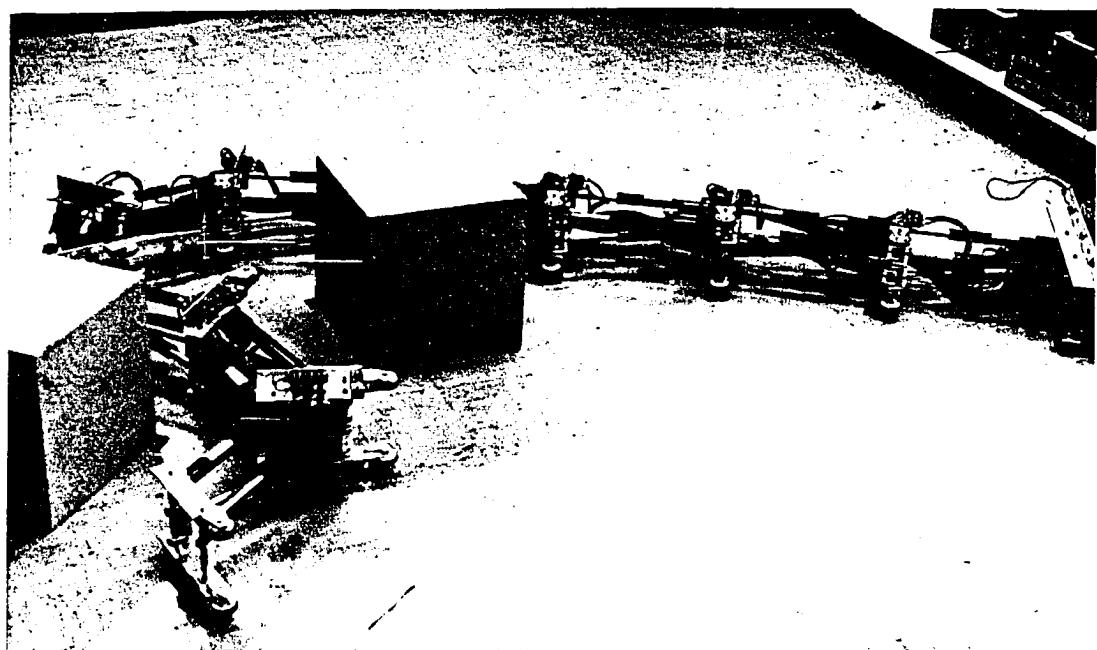


Figure 9.9: Manipulator Fully in Obstacle Field

### 9.3 Locomotion

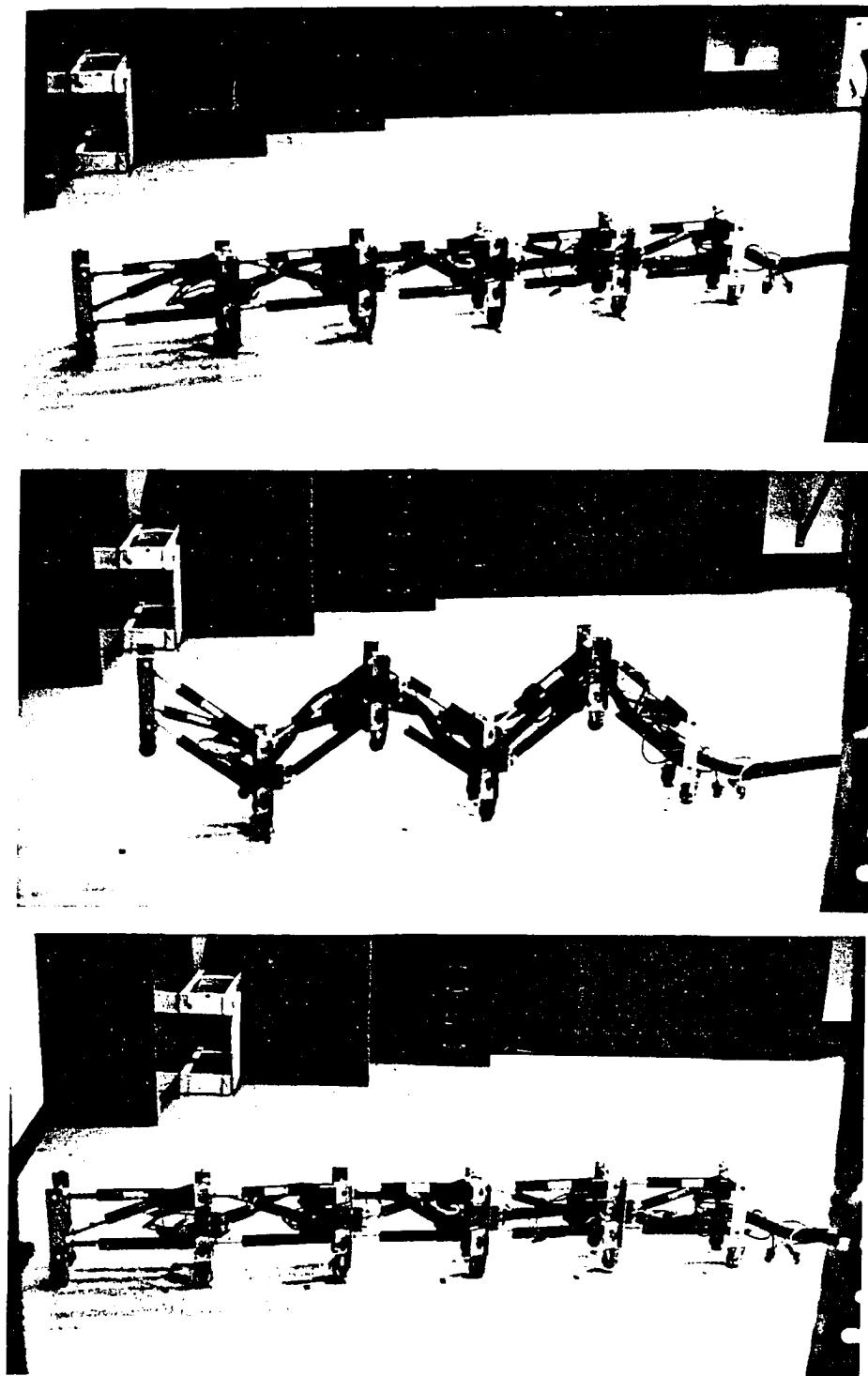
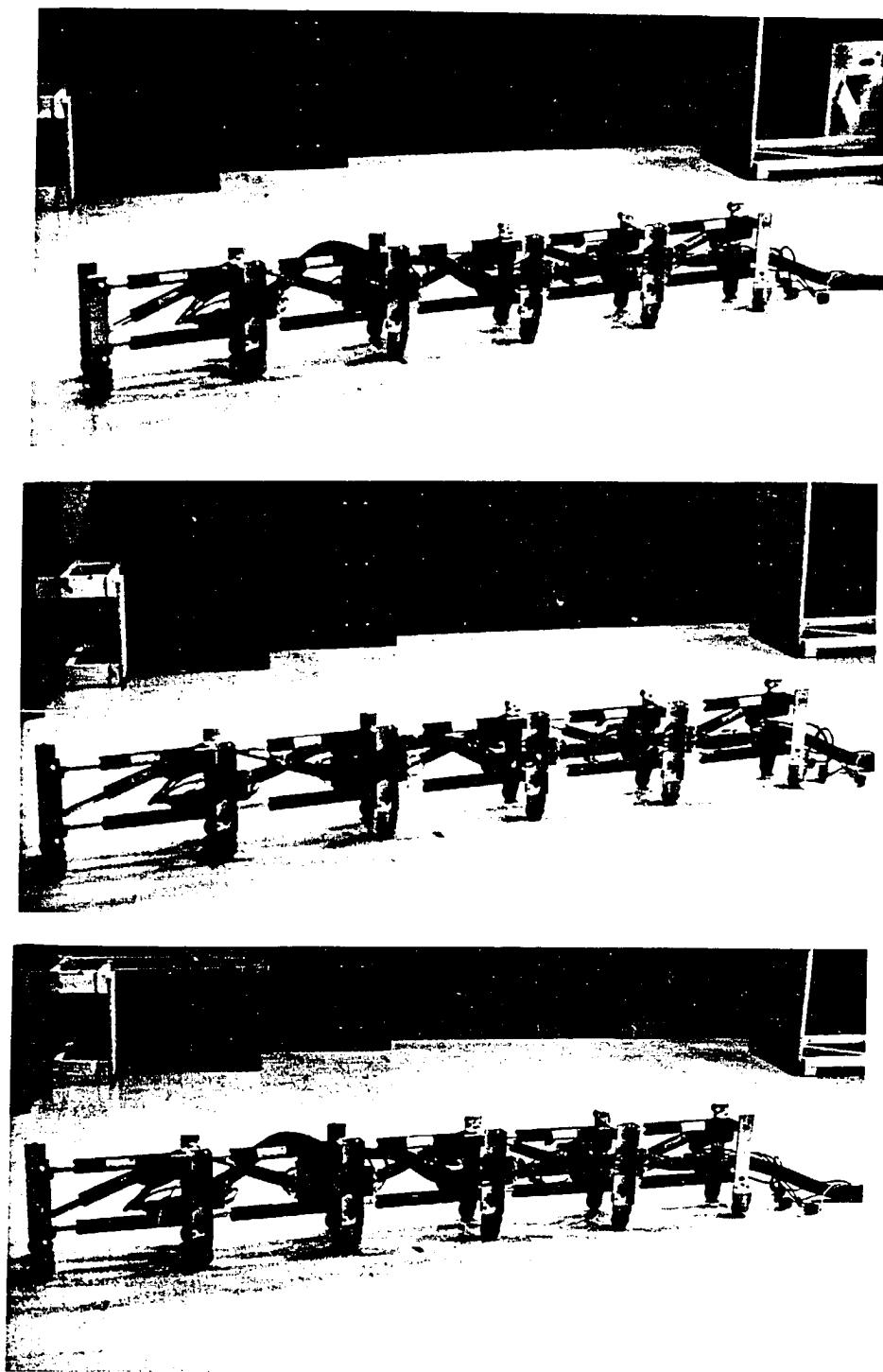


Figure 9.10: Stationary Wave Locomotion - Pure Bending



**Figure 9.11:** Stationary Wave Locomotion - Pure Extension

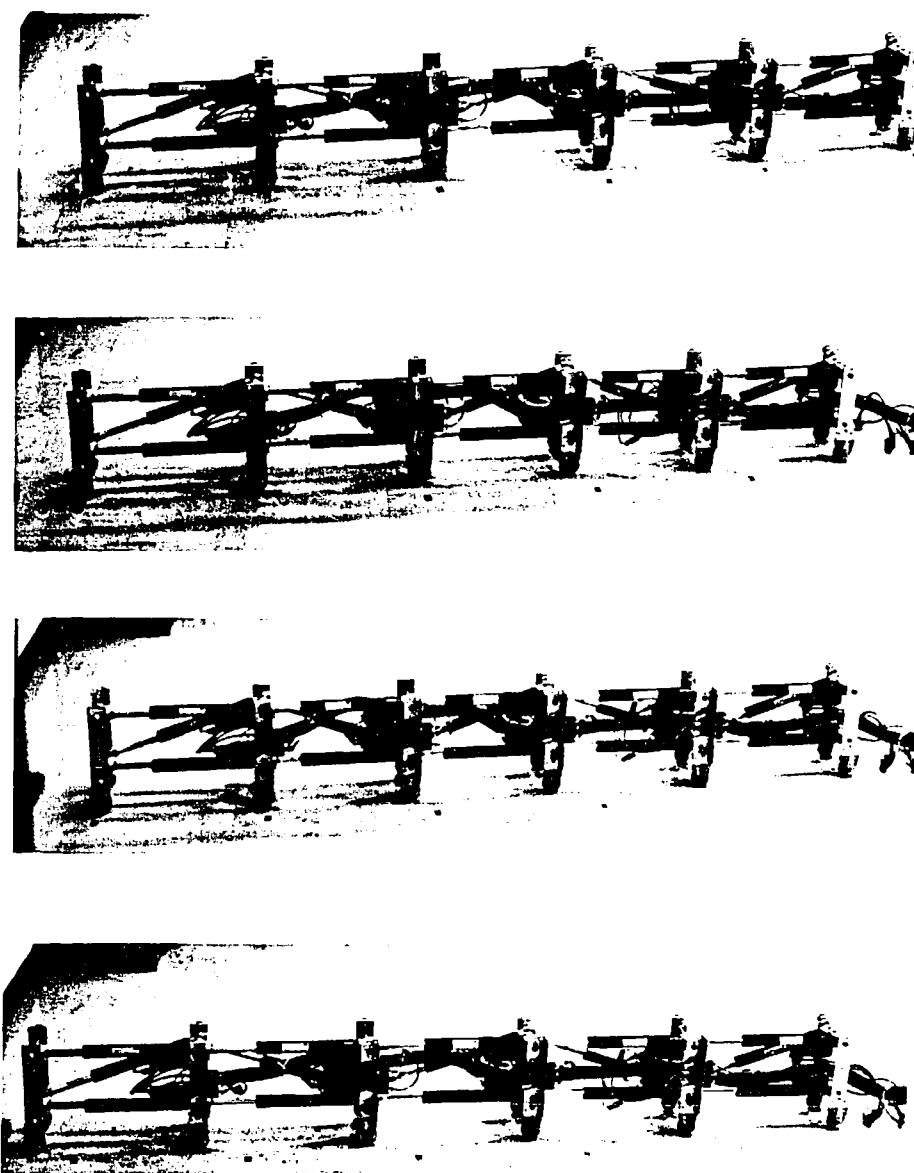


Figure 9.12: Traveling Wave Locomotion - Pure Extension

## 9.4 Grasping a Cylinder

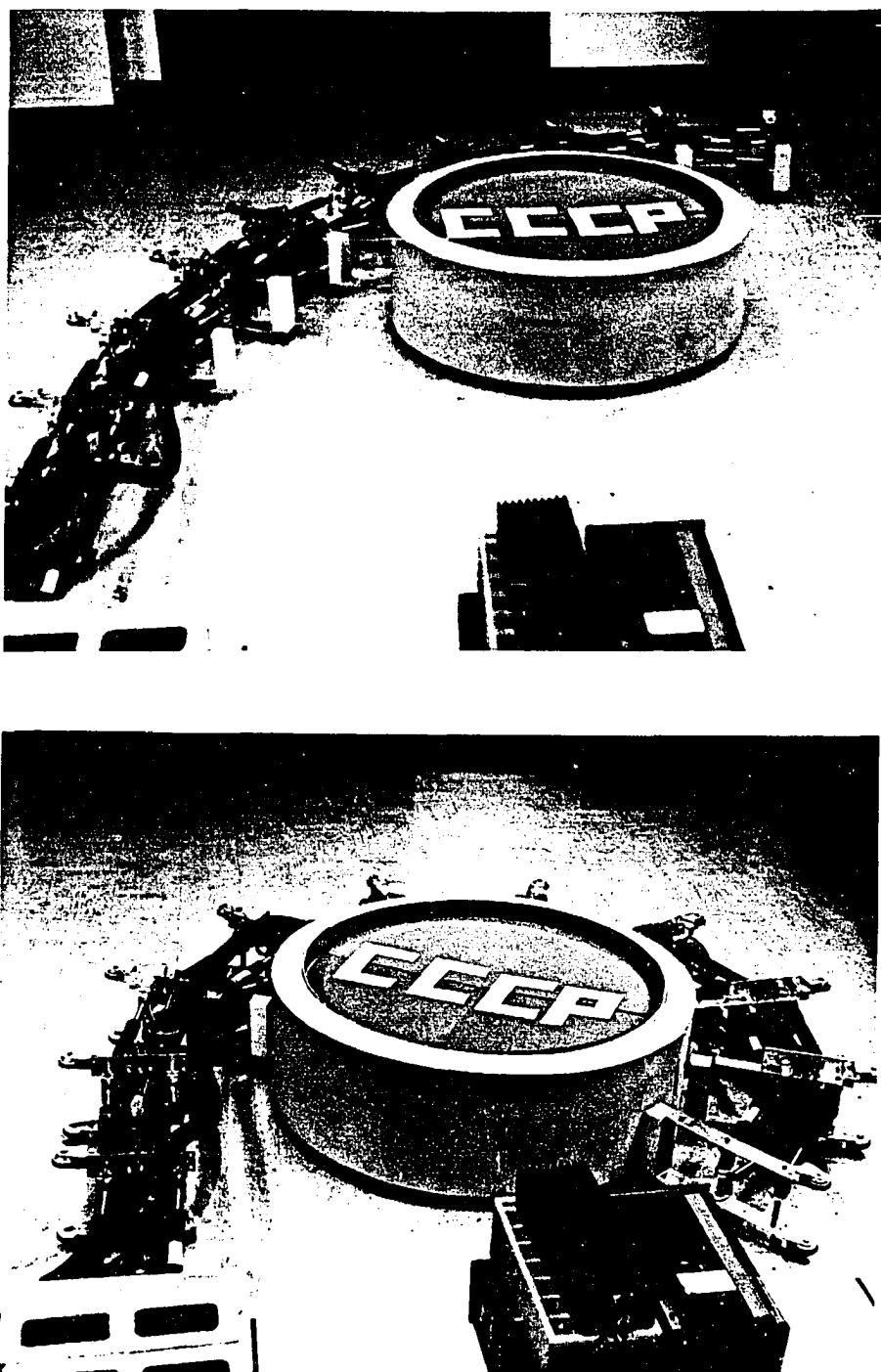


Figure 9.13: Single-Arm Satellite Recovery

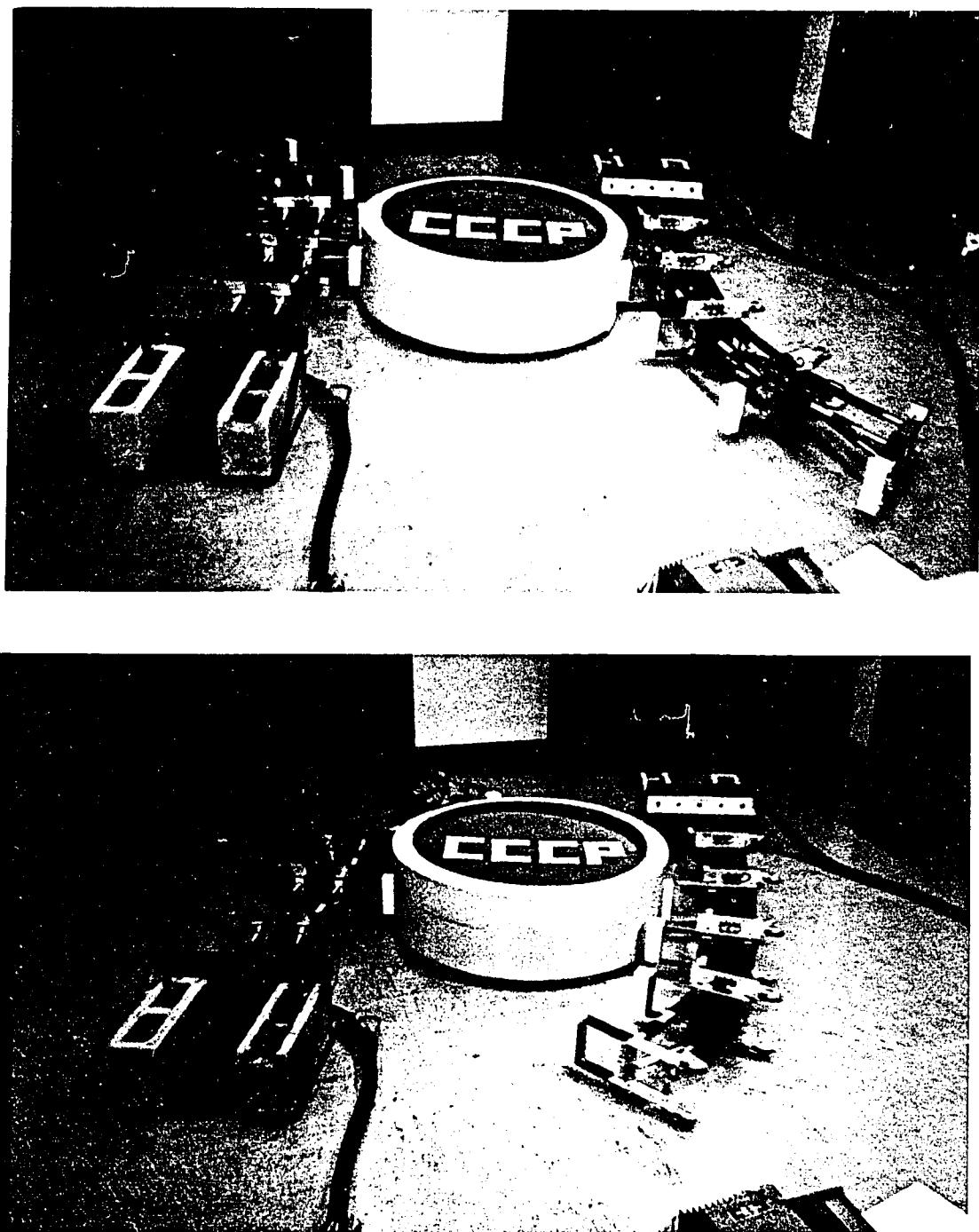


Figure 9.14: Dual-Arm Satellite Recovery

## 9.5 Discussion

The experiments performed and illustrated in this chapter demonstrate and confirm three important conclusions of this work. First, the analytical framework developed in Chapters 2-7 can be successfully employed to control real hyper-redundant robots. Second, hyper-redundant robot design and implementation are not necessarily complex. Third, hyper-redundant robots controlled using the methodology developed in this thesis can be used for a variety of task scenarios such as locomotion, grasping, and obstacle avoidance.

# Chapter 10

## Conclusions

The three divisions of this thesis addressed three broad issues in hyper-redundancy: kinematics, task analysis and simulation, and hardware development and demonstration.

In the first division (Chapters 2 and 3) a general kinematic framework was developed to capture the macroscopic geometric features of general hyper-redundant manipulators. While the applicability of this kinematic framework diminishes with diminishing degree of redundancy, it works quite well in situations such as continuous morphology manipulators, where conventional methods of robot analysis do not apply.

The second division (Chapters 4-7) developed useful techniques for resolving hyper-redundancy for particular tasks using the kinematic framework developed in the first division. These tasks include end-effector positioning, obstacle avoidance, locomotion, and object reorientation and transport using peristaltic-like waves.

The third division (Chapters 8 and 9) addressed issues in the implementation of hyper-redundant robotic systems. The design and implementation of a thirty degree-of-freedom planar hyper-redundant robot was described. The algorithms developed in the first two divisions were implemented on this system, and task performance was demonstrated. A design paradigm and associated methodology have

resulted from the experimental section of this thesis. Namely, hyper-redundant robotic mechanisms need not be complicated mechanical devices.

In the past, little progress has been made in the application of hyper-redundant robots to practical problems. Hyper-redundant robots have failed to achieve widespread applicability due to inefficiency and ineffectiveness of previous kinematic modeling techniques, complex mechanical design, and complexity in the programming of these devices arising from their non-anthropomorphic geometry. Using methods developed in this thesis, the kinematic complexity has been overcome, and applying hyper-redundant robots to solve real problems in manipulation, obstacle avoidance, and locomotion is becoming more feasible. Challenging work in mechanical implementation and high performance control still remains.

Great potential exists for future work in the area of hyper-redundant robotic mechanisms. One area of research could be the investigation of different hyper-redundant morphologies and system architectures. Another is the mechanics of whole arm manipulation and peristaltic transport of objects. Algorithmic problems in the area of automatic task selection, sensor based motion planning, and sensor integration with kinematic motion planning need to be addressed. Furthermore, development of user interfaces to enable nonanthropomorphic devices to be readily controlled by human operators could have great potential for inspection in highly constrained and hazardous environments. These issues provide a rich source of problems in hyper-redundancy to be investigated in the future.

# Appendix

This appendix contains a brief review of the calculus of variations. Primarily, results of the calculus of variations are enumerated. In Section A.1, the general one-dimensional variational problem is reviewed in the context of the analysis in Chapter 5. In Section A.2, the form of the Euler-Lagrange equations for the current problem is examined. Explicit time dependence is suppressed in this exposition, because it plays no role in the derivation of equations. For more complete coverage of the calculus of variations, and efficient formulation of associated dynamical equations, see [E69,GruS80,Wei74].

## A.1 Review of the Calculus of Variations

The optimal configuration problem developed in Chapter 5 required the minimization of integrals of the form:

$$I = \int_{s_0}^{s_1} f(s, \bar{q}(s), \bar{q}^1(s), \dots, \bar{q}^n(s)) ds. \quad (A.1.1)$$

$\bar{q} \in \mathbf{R}^N$  is a set of configuration variables. In the current context, these configuration variables could be  $\{L, K, T, R\}$ ,  $\{\bar{\lambda}, L\}$ , or any parametrization of the backbone reference set. The scalar function  $f(\cdot) \in \mathbf{R}$  is a physically motivated function.  $\bar{q}^i$  is shorthand notation for higher derivatives of  $\bar{q}$ :  $\bar{q}^i = \frac{d^i \bar{q}}{ds^i}$ . Many of the optimization problems in Chapter 5 have integral, or isoperimetric, constraints of the form:

$$\int_{s_0}^{s_1} \bar{g}(s, \bar{q}(s), \bar{q}^1(s), \dots, \bar{q}^n(s)) ds = \bar{c}_0, \quad (A.1.2)$$

where  $\bar{g}(\cdot), \bar{c}_0 \in \mathbf{R}^M$ , and  $\bar{c}_0$  is independent of  $s$ . For the problems in this thesis, constraints of the form of Equation (A.1.2) arise from end-effector position constraints. Some optimization formulations may also have finite constraints of the form:

$$\bar{h}(s, \bar{q}(s), \bar{q}^1(s), \dots, \bar{q}^n(s)) = \bar{0}. \quad (A.1.3)$$

The calculus of variations provides a means for finding a  $\bar{q}(s)$  which yields stationary, or extremal, values of the integral in Equation (A.1.1) with constraints of Equations (A.1.2) and/or (A.1.3). To solve such problems, define a function which will be referred to as the Lagrangian:

$$\mathcal{L} = f + \bar{\mu}_c \cdot \bar{g} + \bar{\mu}_v \cdot \bar{h}. \quad (A.1.4)$$

The Lagrange multipliers  $\bar{\mu}_c$  and  $\bar{\mu}_v$  are respectively independent and dependent on the parameter  $s$ , as required for constraint satisfaction.

The vector function  $\bar{q}(s) \in \mathbf{R}^N$  which extremizes Equation (A.1.1) with constraints of Equation (A.1.2) or (A.1.3) can be found as the solutions to the Euler-Lagrange equations:

$$\sum_{i=0}^n (-1)^i \frac{d^i}{ds^i} \left( \frac{\partial \mathcal{L}}{\partial q_j^i} \right) = 0 \quad j = 1, \dots, N \quad (A.1.5)$$

which together with constraint Equation (A.1.2) or (A.1.3) and boundary conditions  $\bar{q}^i(s_0) = \bar{q}_0^i$  and  $\bar{q}^i(s_1) = \bar{q}_1^i$  can generally be solved to find the extremizing functions,  $\bar{q}$ , and Lagrange multipliers  $\bar{\mu}_c$  and  $\bar{\mu}_v(s)$ . Sufficient conditions, which guarantee that  $\bar{q}(s)$  minimize Equation (A.1.1) can be found in [E69, GruS80]. For most physically motivated problems,  $f(\cdot)$  is a positive definite quadratic form which means  $\bar{q}(s)$  will minimize Equation (A.1.1) if it solves Equation (A.1.5).

As will be seen in Section A.2, with some backbone curve parametrizations and some optimality criteria, the Euler-Lagrange equations assume a fairly simple form.

## A.2 Explicit Form of the Euler-Lagrange Equations

Note that in the following derivation, the following notation is used:

$$\mathbf{A} \cdot \mathbf{B} = \text{tr}(\mathbf{AB}^T) \quad (A.2.1)$$

where  $\mathbf{A}, \mathbf{B} \in \mathbf{R}^{3 \times 3}$ . The backbone reference set frames,  $\mathbf{Q}(s)$ , are often expressed as a function of other parametric functions,  $\bar{\mathbf{q}}(s) \in \mathbf{R}^N$ , such as Euler angles:

$$\mathbf{Q}(s) = \mathbf{R}(\bar{\mathbf{q}}(s)) \quad (A.2.2)$$

To write the Euler-Lagrange equations in a compact form, the chain rule is used:

$$\dot{\mathbf{Q}} = \dot{\mathbf{R}} = \sum_{i=1}^N \dot{q}_i \frac{\partial \mathbf{R}}{\partial q_i}. \quad (A.2.3)$$

Thus,

$$\frac{\partial \dot{\mathbf{R}}}{\partial \dot{q}_i} = \frac{\partial \mathbf{R}}{\partial q_i}. \quad (A.2.4)$$

The contribution of  $f = \frac{1}{2}\text{tr}(\dot{\mathbf{R}}\dot{\mathbf{R}}^T)$  in the first term in the Euler-Lagrange equations can be rewritten as:

$$\frac{d}{ds} \left( \frac{\partial f}{\partial \dot{q}_i} \right) - \frac{\partial f}{\partial q_i} = \ddot{\mathbf{R}} \cdot \frac{\partial \mathbf{R}}{\partial q_i}. \quad (A.2.5)$$

Using the chain rule twice relates  $\ddot{\mathbf{R}}$  to partial derivatives as follows:

$$\ddot{\mathbf{R}} = \sum_{k=1}^N \frac{\partial \mathbf{R}}{\partial q_k} \ddot{q}_k + \sum_{k=1}^N \sum_{i=1}^N \frac{\partial^2 \mathbf{R}}{\partial q_i \partial q_k} \dot{q}_i \dot{q}_k. \quad (A.2.6)$$

Thus, the  $i^{th}$  of  $N$  equations which must be solved in the nonextensible case with unity weighting are of the form:

$$\sum_{k=1}^N \frac{\partial \mathbf{R}}{\partial q_i} \cdot \frac{\partial \mathbf{R}}{\partial q_k} \ddot{q}_k + \sum_{j=1}^N \sum_{k=1}^N \frac{\partial \mathbf{R}}{\partial q_i} \cdot \frac{\partial^2 \mathbf{R}}{\partial q_j \partial q_k} \dot{q}_j \dot{q}_k - \bar{\mu}_i \cdot \frac{\partial \bar{u}}{\partial q_i} = 0, \quad (A.2.7)$$

which can be expressed in terms of a particular parametrization, as in Chapter 5. In the extensible case, the analogous equations are:

$$\sum_{k=1}^N \frac{\partial \mathbf{R}}{\partial q_i} \cdot \frac{\partial \mathbf{R}}{\partial q_k} \ddot{q}_k + \sum_{j=1}^N \sum_{k=1}^N \frac{\partial \mathbf{R}}{\partial q_j} \cdot \frac{\partial^2 \mathbf{R}}{\partial q_i \partial q_k} \dot{q}_j \dot{q}_k - \dot{L} \bar{\mu}_c \cdot \frac{\partial \bar{u}}{\partial q_i} = 0 \quad (A.2.8)$$

and the one additional equation:

$$\beta_0 \ddot{L}(s) + \sum_{i=1}^N \dot{q}_i \bar{\mu}_c \cdot \frac{\partial \bar{u}}{\partial q_i} = 0. \quad (A.2.9)$$

The boundary conditions on the variables  $\bar{q}(s)$  which parametrize the rotation matrix must be such that

$$\mathbf{R}(\bar{q}(0)) = \mathbf{I} \quad (A.2.10)$$

and

$$\mathbf{R}(\bar{q}(1)) = \mathbf{R}_{ee}, \quad (A.2.11)$$

and the vector of Lagrange multipliers,  $\bar{\mu}_c$ , must be determined such that  $\bar{x}(1) = \bar{x}_{ee}$ . In the extensible case, the boundary conditions for Equation (A.2.9) are  $L(0) = 0$ ,  $L(1) = L_0$ , whereas in the nonextensible case,  $L(s) = s$ , and we are not free to choose the length.

## References

- [AD74] Abdy, P.R., Dempster, M.A.H., Introduction to Optimization Methods, Chapman and Hall, New York, 1974.
- [AlM89] Alexander, J.C., Maddocks, J.H., "On the Kinematics of Wheeled Mobile Robots," *International Journal of Robotics Research*, Vol. 8, No. 5, 1989.
- [Alv91] Alvarado, F.L., Personal Correspondence, University of Wisconsin, Nov. 1991.
- [AndH67] Anderson, V.V., Horn, R.C., "Tensor-arm Manipulator Design," ASME Trans., Vol. 67-DE-57, pp. 1-12, 1967.
- [Ang88] Angeles, J., Rational Kinematics, Springer-Verlag, New York, 1988.
- [AnA90] Anthimopoulou, M., Aspragathos, N., "Kinematic Control of Planar Redundant Manipulators Moving Between Obstacles," *2<sup>nd</sup> International Workshop on Advances in Robot Kinematics*, Linz, Austria, Sept. 1990.
- [At89] Atkeson, C.G., "Learning Arm Kinematics and Dynamics," *Ann. Rev. Neurosci.*, Vol. 12, pp 157-183, 1989.
- [Bail86] Baillieul, J., "Avoiding Obstacles and Resolving Kinematic Redundancy," *Proceedings of the 1986 IEEE International Conference on Robotics and Automation*, San Francisco, California, pp. 1698-1704, 1986.

- [BarhGZ88] Barhen, J., Gulati, S., Zak, M., "Real-Time Neuromorphic Algorithms for Inverse Kinematics of Redundant Manipulators," *SPIE Vol. 1002*, pp. 686-696, 1988.
- [BarkM89] Barker, L.K., McKinney, W.S., "Optimized Resolved Rate Control of Seven-Degree-of-Freedom Laboratory Telerobotic Manipulator (LTM) With Application to Three-Dimensional Graphics Simulation," *NASA Technical Paper 2938*, 1989.
- [BarrLL89] Barraquand, J., Langlois, B., Latombe, J-C., "Numerical Potential Field Techniques for Robot Path Planning," *STAN-CS-89-1285*, Dept. of Comp. Sci., Stanford University, Oct. 1989.
- [Bay92] Bay, J.S., "Geometry and Prediction of Drift-Free Trajectories for Redundant Machines Under Pseudoinverse Control," *International Journal of Robotics Research*, Vol. 11, No. 1, pp. 41-52, Feb. 1992.
- [BenMT74] Bennet, S., McConnel, T., Trubatch, S.L., "Quantitative analysis of the speed of snakes as a function of peg spacing," *J. Exp. Biol.*, Vol. 60, pp. 161-165, 1974.
- [Bo58] Bowman, F., Introduction to Bessel Functions, Dover Inc, New York, 1958.
- [Broo83] Brooks, R.A., "Solving the Find-Path Problem by Good Representation of Free Space," *IEEE Transaction on Systems, Man, and Cybernetics*, Vol. SMC-13, No. 3, pp. 190-197, March 1983.
- [BrocS91] Brockett, R.W., Stokes, A., "On the Synthesis of Compliant Mechanisms," *1991 IEEE Conference on Robotics and Automation*, Sacramento, California, pp. 2168-2173, May 1991.
- [Bur88] Burdick, J.W., "Kinematic Analysis and Design of Redundant Robot Manipulators," PhD Dissertation, Stanford University, March 1988.
- [BurCB91] Burdick, J.W., Cetin, B., Barhen, J., "Efficient Global Redundant Configuration Resolution via Sub-Energy Tunneling and Terminal Repelling,"

1991 *IEEE Conference on Robotics and Automation*, Sacramento, California, pp. 929-944, May 1991.

[BušFMZ91] Buško, Z., Fraczek, J., Moreckis, A., Zielińska, T., "Modelling and Design of Elastic Manipulators and a Walking Machine," *Proceedings of the 8<sup>th</sup> World Congress on the Theory of Machines and Mechanisms*, Prague Czechoslovakia, pp. 1177-1180, August 1991.

[Can88] Canny, J., The Complexity of Robot Motion Planning, The MIT Press, Cambridge, MA, 1988.

[Ch88] Chirikjian, G.S., "Lagrangian Formulation of Dynamical Equations Governing Appendage Deployment from Spacecraft," M.S.E. Thesis, Department of Mechanical Engineering, The Johns Hopkins University, Baltimore, MD, 1988.

[ChB89-1] Chirikjian, G.S., Burdick, J.W., "A Modal Approach to the Kinematics of Hyper-Redundant Manipulators," *Robotics and Mechanical Systems Report* No. RMS-89-3, Dept. of Mechanical Engineering, California Institute of Technology, Pasadena, CA 91125, September 1989.

[ChB89-2] Chirikjian, G.S., Burdick, J.W., "An Obstacle Avoidance Algorithm for Hyper-Redundant Manipulators," *Robotics and Mechanical Systems Technical Report* No. RMS-89-4, Dept. of Mechanical Engineering, California Institute of Technology, Sept. 1989.

[ChB89-3] Chirikjian, G.S., Burdick, J.W., "Hyper-Redundant Robotic Locomotion: Locomotion Without Wheels, Tracks, or Legs," *Robotics and Mechanical Systems Technical Report* No. RMS-89-6, Dept. of Mechanical Engineering, California Institute of Technology, Sept. 1989.

[ChB90-1] Chirikjian, G.S., Burdick, J.W., "An Obstacle Avoidance Algorithm for Hyper-Redundant Manipulators," *Proceedings of the 1990 IEEE Conference on Robotics and Automation*, Cincinnati, Ohio, pp. 625-631, May 14-17, 1990.

- [ChB90-2] Chirikjian, G.S., Burdick, J.W., "Kinematics of Hyper-Redundant Manipulators," *Proceedings of the ASME Mechanisms Conference*, Chicago, IL, DE-Vol. 25, pp. 391-396, Sept. 16-19, 1990.
- [ChB90-3] Chirikjian, G.S., Burdick, J.W., "A Geometric Approach to Hyper-Redundant Manipulator Obstacle Avoidance," *Proceedings of the ASME Mechanisms Conference*, Chicago, IL, DE-Vol. 24, pp. 337-342, Sept. 16-19, 1990.
- [ChB90-4] Chirikjian, G.S., Burdick, J.W., "Applications of Hyper-Redundant Manipulators for Space Robotics and Automation," i-SAIRAS Meeting, Kobe, Japan, Nov. 18-21, 1990.
- [ChB91-1] Chirikjian, G.S., Burdick, J.W., "Parallel Formulation of the Inverse Kinematics of Modular Hyper-Redundant Manipulators," *1991 IEEE Conference on Robotics and Automation*, Sacramento, California, pp. 708-713, April 1991.
- [ChB91-2] Chirikjian, G.S., Burdick, J.W., "Kinematics of Hyper-Redundant Locomotion with Applications to Grasping," *1991 IEEE Conference on Robotics and Automation*, Sacramento, California, pp. 720-725, April 1991.
- [ChB91-3] Chirikjian, G.S., Burdick, J.W., "Theory and Applications of Hyper-Redundant Robotic Mechanisms," *8<sup>th</sup> World Congress on the Theory of Machines and Mechanisms*, Prague, Czechoslovakia, August 1991.
- [ChB91-4] Chirikjian, G.S., Burdick, J.W., "Hyper-Redundant Robot Mechanisms and Their Applications," *IROS'91*, Osaka, Japan, November 1991.
- [ChB92] Chirikjian, G.S., Burdick, J.W., "On the Determination of Kinematically Optimal Hyper-Redundant Manipulator Configurations," *1992 IEEE Conference on Robotics and Automation*, Nice, France, May 10-15, 1992.
- [ClaH76] Clark, R.B., Hermans, C.O., "Kinetics of swimming in some smooth-bodied polychaetes," *J. Zool.*, Lond., Vol. 178, pp. 147-159, 1976.
- [CleI90] Clement, W.I., Inigo, R.M., "Design of a snake-like manipulator," *Robotics and Automated Systems*, Vol. 6, pp. 265-282, 1990.

- [Crai86] Craig, J.J., Introduction to Robotics, Mechanics and Control, Addison-Wesley, Reading, Mass., 1986.
- [CravT89] Craver, W.M., Tesar, D., "Structural Analysis and Design of a Three Degree-of-Freedom Robotic Shoulder Module," Department of Mechanical Engineering, the University of Texas at Austin, January 1989. Based on a thesis presented to the University of Texas at Austin, May 1989.
- [DaSS88] Das, H., Slotine, J.J.E., Sheridan, T.B., "Inverse Kinematic Algorithms for Redundant Systems," *Proceedings of the 1988 IEEE International Conference on Robotics and Automation*, Philadelphia, 1988.
- [Dr84] Drozda, T.S., "Spine Robot ... The Verdict's Yet to Come," *Manufacturing Engineering*, Vol. 93, No. 3, pp. 110-112, Sept. 1984.
- [DufC80] Duffy, J., Crane, C., *Mechanism and Machine Theory*, Vol. 16, pp. 153-169, 1980.
- [Dw61] Dwight, H.B., Tables of Integrals and Other Mathematical Data, MacMillan Inc., New York, 1961.
- [E69] Ewing, G.M., Calculus of Variations With Applications, W.W. Norton and Co. Inc., New York, 1969.
- [Fa90] Farin, G.E., Curves and Surfaces for Computer Aided Geometric Design: A Practical Guide, 2nd ed., Academic Press, Boston, 1990.
- [FuHK90] Fukuda, T., Hosokai, H., Kikuchi, I., "Distributed Type of Actuators by Shape Memory Alloy and its Application to Underwater Mobile Robotics Mechanism," *Proceedings of the 1990 IEEE International Conference on Robotics and Automation*, Cincinnati, Ohio, May 14-17, pp. 1316-1321, 1990.
- [FuHU89] Fukuda, T., Hosokai, H., Uemura, M., "Rubber Gas Actuator Driven by Hydrogen Storage Alloy for In-pipe Inspection Mobile Robot with Flexible

Structure," *Proceedings 1989 IEEE International Conference on Robotics and Automation*, Scottsdale, Arizona, pp. 1847-1852, May 1989.

[Ga62] Gans, C., "Terrestrial Locomotion without Limbs," *Amer. Zool.*, Vol. 2, pp. 167-182, 1962

[Ga85] Gans, C., "Limbless Locomotion - A Current Overview," *Proceedings of International Symposium on Vertebrate Morphology. Forschritte der Zoologie*, Gustav Fischer Verlag, Stuttgart and New York, pp. 13-22, 1985.

[GiMW81] Gill, P.E., Murray, W., Wright, M.H., Practical Optimization, Academic Press, Inc., New York, 1981.

[Gr46] Gray, J., "The mechanism of locomotion in snakes," *J. Exp. Biol.*, Vol. 23, pp. 101-120, 1946.

[GrL38] Gray, J., Lissman, H.W., "Studies in animal locomotion. VII. Locomotory reflexes in the earthworm," *J. Exp. Biol.*, Vol. 15, pp. 506-517, 1938.

[GrL50] Gray, J., Lissmann, H.W., "The kinetics of locomotion of the grass snake," *J. Exp. Biol.*, Vol. 26, pp. 354-367, 1950.

[GrL64] Gray, J., Lissmann, H.W., "The Locomotion of Nematodes," *J. Exp. Biol.*, Vol. 41, pp. 135-154, 1964.

[GruS80] Gruver, W.A., Sachs, E., Algorithmic Methods in Optimal Control, Pitman Publishing, Ltd., Boston, 1980.

[HaPK90] Hayashi, A., Park, J., Kuipers, B.J., "Toward Planning and Control of Highly Redundant Manipulators," *Fifth IEEE International Symposium on Intelligent Control*, 1990.

[He85] Hemami, A., "Studies on a Light Weight and Flexible Robot Manipulator," *Robotics*, Vol. 1, pp. 27-36, 1985

- [HiKU83] Hirose, S., Kado, T., Umetani, Y., "Tensor Actuated Elastic Manipulator," *Proc. 6th IFToMM World Congress*, New Delhi, India, Vol. 2, pp. 978-981, 1983.
- [HiMa89] Hirose, S., Ma, S., "Redundancy Decomposition for Multi-Joint Manipulator," *Proceedings 1989 IEEE International Conference on Robotics and Automation*, pp. 119-124, May 1989.
- [HiMo90] Hirose, S., Morishima, A., "Design and Control of a Mobile Robot with an Articulated Body," *International Journal of Robotics Research*, Vol. 9., No. 2, pp. 99-114, 1990.
- [HiMa91] Hirose, S., Ma, S., "Coupled Tendon-driven Multijoint Manipulator," *1991 IEEE International Conference on Robotics and Automation*, Sacramento, California, pp. 1268-1275, May 1991.
- [HiU76] Hirose, S., Umetani, Y., "Kinematic Control of Active Cord Mechanism With Tactile Sensors," *Proceedings of Second International CISM-IFT Symposium on Theory and Practice of Robots and Manipulators*, pp. 241-252, 1976.
- [HouV83] Housner, G.W., Vreeland, T., The Analysis of Stress and Deformation, California Institute of Technology, 1983.
- [IB84] Ivanescu, M., Badea, I., "Dynamic Control for a Tentacle Manipulator," *Proceedings of the Int. Conf. on Robotics and Factories of the Future*, Charlotte, North Carolina, pp. 317-328, Dec. 4-7, 1984,
- [Ja85] Jayne, B.C., "Swimming in constricting (*Elaphe g. guttata*) and nonconstricting (*Nerodia fasciata pictiventris*) colubrid snakes," *Copeia*, pp. 195-208, 1985.
- [Ja86] Jayne, B.C., "Kinematics of Terrestrial Snake Locomotion," *Copeia*, No. 4, pp. 915-927, 1986.
- [Ja88] Jayne, B.C., "Mechanical behaviour of snake skin," *J. Zool., Lond.*, Vol. 214, pp. 125-140, 1988.

- [Jo73] Jones, H.D., "The mechanism of locomotion of *Agriolimax reticulatus* (Mollusca: Gastropoda)," *J. Zool. Lond.*, Vol. 171, pp. 489-498, 1973.
- [JoT70] Jones, H.D., Trueman, E.R., "Locomotion of the Limpet, *Patella Vulgata L.*," *J. Exp. Biol.*, Vol. 52, pp. 201-216, 1970.
- [Kha86] Khatib, O., "Real-Time Obstacle Avoidance for Manipulators and Mobile Robots," *Int. J. Robotics Research*, Vol. 5, No. 1, 1986.
- [KhoV88] Khosla, P., Volpe, R., "Superquadric Artificial Potentials for Obstacle Avoidance and Approach," *Proceedings of the 1988 IEEE International Conference on Robotics and Automation*, Philadelphia, PA, pp. 1778-1784, 1988.
- [Kie82] Kier W.M., "The Functional Morphology of the Musculature of Squid (Loliginidae) Arm and Tentacles," *Journal of Morphology*, Vol. 172, pp. 307-324, 1982.
- [KieS85] Kier, W.M., Smith, K.K., "Tongues, tentacles and trunks: the biomechanics of movement in the muscular-hydrostats," *Zoological Journal of the Linnean Society*, Vol. 83, pp. 307-324, 1985
- [KirV86] Kircanski, M., Vukobratovic, M., "Contribution to the Control of Redundant Robotic Manipulators in an Environment with Obstacles," *Int. J. Robotics Research*, Vol. 5, No. 4., 1986.
- [KIH83] Klein, C.A., Huang, C.H., "Review of the Pseudoinverse for Control of Kinematically Redundant Manipulators," *IEEE Transactions on Systems, Man and Cybernetics*, Vol. SMC-13, No. 2, pp. 245-250, March 1983.
- [Kr79] Kreyzig, E., Advanced Engineering Mathematics, Fourth Ed., Wiley, New York, 1979.
- [Ku87] Kumar, V.R., "Motion Planning for Legged Locomotion Systems on Uneven Terrain," Ph.D. dissertation, The Ohio State University, Columbus, Ohio, 1987.

- [Le72] Lebedev, N.N., trans. by Silverman, R., Special Functions and Their Applications, Dover Inc., New York, 1972.
- [LeeL87] Lee, H.Y., Liang, C.G., "Displacement Analysis of the General Spatial 7-Link 7R Mechanism" (to be published in *Mechanism and Machine Theory*).
- [Li-MR89] Li, Z., Montgomery, R., Raibert, M., "Dynamics and Optimal Control of Legged Lococmotion Systems" (preprint), Oct. 1989.
- [Li50] Lissmann, H.W., "Rectilinear locomotion in a snake (*Boa occidentalis*)," *J. Exp. Bio.*, Vol. 26, pp. 368-379, 1950.
- [LuWP80] Luh, J.Y.S., Walker, M.W., Paul, R.P.C., "Resolve-Acceleration Control of Mechanical Manipulators," *IEEE Transactions on Automatic Control*, Vol. AC-25, No. 3, pp. 468-474, 1980.
- [MaH91] Ma, S., Hirose, S., "Development of Coupled Tendon-driven Multijoint Manipulator," *IROS'91*, Osaka, Japan, pp. 725-730, Nov. 1991.
- [Mac89] Maciejewski, A.A., "Kinetic Limitations on the Use of Redundancy in Robotic Manipulators," *Proceedings 1989 IEEE International Conference on Robotics and Automation*, Scottsdale, Arizona, pp. 113-118, May 1989.
- [MacK85] Maciejewski, A.A., Klein, C.A., "Obstacle Avoidance for Kinematically Redundant Manipulators in Dynamically Varying Environments," *International Journal of Robotics Research*, Vol. 1, No. 3, pp. 109-117, 1985.
- [McC90] McCarthy, J.M., Introduction to Theoretical Kinematics, MIT Press, 1990.
- [McGe90] McGee, T., "Passive Dynamic Walking," *International Journal of Robotics Research*, Vol. 9, No. 2, 1990.
- [McGh70] McGhee, R.B., "A mathematical theory for legged locomotion systems," *Proceedings of the 1970 Midwest Symposium on Circuit Theory*, 1970.

- [McGhP72] McGhee, R.B., Pai, A.L., "An approach to computer control for legged vehicles," *J. Terramechanics*, Vol. 11, pp. 9-26, 1972.
- [MilP77] Millman, R.S., Parker, G.D., Elements of Differential Geometry, Prentice-Hall Inc., Englewood Cliffs, NJ, 1977.
- [MiuF85] Miura, K., Furuya, H., "Variable Geometry Truss and Its Application to Deployable Truss and Space Crane Arm," *Acta Astronautica*, Vol. 12, No. 7/8, pp. 599-607, 1985.
- [Mo87] Morecki, A., et al., "Robotic System - Elephant Trunk Type Elastic Manipulator Combined with a Quadruped Walking Machine," *Proc. of Second Int. Conf. on Robotics and Factories of the Future*, San Diego, pp. 649-656, July 1987.
- [MuS90] Murray, R.M., Sastry, S.S., "Steering Nonholonomic Systems Using Sinusoids," *IEEE Control and Decision Conference*, Honolulu, Hawaii, Dec. 5-7, 1990.
- [NacH89] Naccarato, F., Hughes, P.C., "An Inverse Kinematics Algorithm for a Highly Redundant Variable-Geometry-Truss Manipulator," *Proceedings of the 3rd Annual Conference on Aerospace Computational Control*, ed. by D.E. Bernard and G.K. Man, Oxnard, CA, JPL Publication 89-45, Dec. 15, 1989.
- [NacH91] Naccarato, F., Hughes, P.C., "Inverse Kinematics of Variable-Geometry Truss Manipulators," *Journal of Robotic Systems* Vol. 8, No. 2, pp. 249-266, 1991.
- [NakG89] Nakamura, Y., Ghodoussi, M., "Dynamics Computation of Closed-Link Robot Mechanisms with Nonredundant and Redundant Actuators," *IEEE Trans. on Robotics and Automation*, Vol. 5, No. 3, June 1989.
- [NakHY87] Nakamura, Y., Hanafusa, H., Yoshikawa, T., "Task-Priority Based Redundancy Control of Robot Manipulators," *International Journal of Robotics Research*, Vol. 6, No. 2., pp. 3-15, 1987.

- [NakM89] Nakamura, Y., Mukherjee, R., "Nonholonomic Path Planning of Space Robots," *Proceedings of the 1989 IEEE International Conference on Robotics and Automation*, Scottsdale, AZ, pp. 1050-1055, May 15-19, 1989.
- [Pa75] Paul, B., "Analytical dynamics of mechanisms - A computer oriented overview," *Mech. Mach. Theory*, Vol. 10, pp. 481-507, 1975.
- [Pe88] Pettinato, J.S., "Manipulability and Stability of a Tentacle Based Robot Manipulator," M.S. Thesis, George Mason University, Dept. of Electrical and Computer Engineering, Dec. 1988.
- [PeS89] Pettinato, J.S., Stephanou, H.E., "Manipulability and Stability of a Tentacle Based Robot Manipulator," *Proceedings of the 1989 IEEE International Conference on Robotics and Automation*, Scottsdale, AZ, pp. 458-463, May 15-19, 1989.
- [Pie68] Pieper, D., "The Kinematics of Manipulators Under Computer Control," Ph.D. Thesis, Stanford University, 1968.
- [PiDF91] Pierrot, F., Dauchez, P., Fournier, A., "Fast Parallel Robots," *Journal of Robotic Systems*, Vol. 8, No. 6, pp. 829-840, 1991.
- [PiFD91] Pierrot, F., Fournier, A., Dauchez, P., "Towards a Fully-Parallel 6 DOF Robot for High-Speed Applications," *Proceedings, 1991 IEEE International Conference on Robotics and Automation*, Sacramento, CA, pp. 1288-1293, April 9-11, 1991.
- [Rai84] Raibert, M.H., "Hopping in Legged Systems - Modeling and Simulation for the 2D One-Legged Case," *IEEE Trans. Systems, Man, and Cybernetics*, 14:451-463, 1984.
- [Rai86] Raibert, M.H., Legged Robots That Balance, The MIT Press, Cambridge, Mass., 1986.
- [RaoM71] Rao, C.R., Mitra, S.K., Generalized Inverse of Matrices and Its Applications, Wiley, New York, 1971.

- [Ri74] Rivlin, T.J., The Chebyshev Polynomials, Wiley, New York, 1974.
- [RobM92] Roberts, R.G., Maciejewski, A.A., "Nearest Optimal Repeatable Control Strategies for Kinematically Redundant Manipulators," (preprint).
- [RogA76] Rogers, D.F., Adams, J.A., Mathematical Elements for Computer Graphics, McGraw-Hill, New York, 1976.
- [SaleRR88] Salerno, R.J., Reinholtz, C.F., Robertshaw, H.H., "Shape Control of High Degree-of-Freedom Variable Geometry Trusses," *Proceedings of the Workshop on Computational Aspects in the Control of Flexible Systems, Part 2*, Williamsburg, VA, July 12-14, 1988.
- [ScS87] Sciavicco, L., Siciliano, B., "A Dynamic Solution to the Inverse Kinematic Problem for Redundant Manipulators," *Proc. 1987 IEEE Int. Conf. on Robotics and Automation*, Raleigh, NC, March/April 1987.
- [ScS88] Sciavicco, L., Siciliano, B., "A Solution Algorithm to the Inverse Kinematic Problem for Redundant Manipulators," *IEEE Trans. on Robotics and Automation*, Vol. 4, No. 4, June 1988.
- [SegWAS92] Segalman, D.J., Witkowski, W.R., Adolf, D.B., Shahinpoor, M., "Theory and Application of Electrically-Controlled Polymeric Gels," to appear in *Int. J. Smart Materials and Structures* vol. 1, 1992.
- [Ser89] Seraji, H., "Configuration Control of Redundant Manipulators: Theory and Implementation," *IEEE Transactions on Robotics and Automation*, Vol. 5., No. 4, pp. 472-490, Aug. 1989.
- [Se69] Seymour, M.K., "Locomotion and Coelomic Pressure in *Lumbricus Terrestris L.*," *J. Exp. Biol.* Vol. 51, pp. 47-58, 1969.
- [Sha87] Shahinpoor, M., A Robot Engineering Textbook, Harper and Row, New York, 1987.

- [ShaKJ86] Shahinpoor, M., Kalhor, H., Jamshidi, M., "On Magnetically Activated Robotic Tensor Arms," *Proceedings of the International Symposium on Robot Manipulators: Modeling, Control, and Education*, Albuquerque, New Mexico, Nov. 12-14, 1986.
- [ShaY88] Shamir, T., Yomdin, Y., "Repeatability of Redundant Manipulators: Mathematical Solution of the Problem," *IEEE Trans. Auto. Control*, Vol. 33, No. 11, pp. 1004-1009, 1988.
- [SIY87] Slotine, J.J.E., Yoerger, D.R., "A Rule Based Inverse Kinematics Algorithm for Redundant Manipulators," *International Journal of Robotics and Automation*, Vol. 2, No. 2, 1987.
- [SoW89] Song, S.M., Waldron, K.J., Machines That Walk, The MIT Press, Cambridge, Mass., 1989.
- [StuBDR90] Stulce, J.R., Burgos, W.E., Dhande, S.G., Reinholtz, C.F., "Conceptual Design of a Multibody Passive-Legged Crawling Vehicle," *Proceedings of the ASME Mechanisms Conference*, Chicago, IL, Sept. 16-19, 1990.
- [Str61] Struik, D.J., Lectures on Classical Differential Geometry, Second Ed., Dover Inc., 1961.
- [SuLi90] Su Bu-Qing, Liu Ding-Yuan, Computational Geometry: Curve and Surface Modeling, Academic Press, Inc., Boston, 1989.
- [SuIT91] Suzumori, K., Iikura, S., Tanaka, H., "Development of Flexible Microactuator and Its Applications to Robotic Mechanisms," *1991 IEEE Conference on Robotics and Automation*, Sacramento, CA, April 1991.
- [TanH91] Tanaka, M., Hanahara, K., "Motion Planning of Truss-Type Robot: Acquisition and Representation of Criterion by Modular Neural Network," *IROS'91*, Osaka, Japan, Nov. 3-5, 1991.

- [TavD90] Tavakkoli, S., Dhande, S.G., "Shape Synthesis and Optimization Using Intrinsic Geometry," *Proceedings of the ASME Design Conference*, Chicago, IL, Sept. 16-19, 1990.
- [TayLE83] Taylor, W.K., Lavie, D., Esat, I.I., "A curvilinear snake arm robot with gripper-axis fibre-optic image processor feedback," *Robotica*, Vol. 1, pp. 33-39, 1983.
- [TeB89] Tesar, D., Butler, M.S., "A Generalized Modular Architecture for Robot Structures," *ASME Manufacturing Review*, Vol. 2, No. 2, pp. 91-118, June 1989.
- [UY89] Umetami, Y., Yoshida, K., "Resolved Motion Rate Control of Space Manipulators with Generalized Jacobian Matrix," *IEEE Transactions on Robotics and Automation*, Vol. 5, No. 3, June 1989.
- [VakB90] Vakakis, A.F., Burdick, J.W., "Chaotic Motions of a Simplified Hopping Robot," *Proceedings, 1990 IEEE Conference on Robotics and Automation*, Cincinnati, OH, pp. 1464-1469, May 14-18, 1990.
- [Vl07] Vles, F., "Sur les ondes pedieuses des mollusques reptateurs," *C.r. hebd. Seanc. Acad. Sci. Paris*, Vol. 145, pp. 276-278, 1907.
- [VuK84] Vukobratovic, M., Kircanski, M., "A Dynamic Approach to Nominal Trajectory Synthesis for Redundant Manipulators," *IEEE Trans. Syst., Man, Cybern.*, Vol. SMC-14, No. 4, pp. 580-586, July/August 1984.
- [Wad90] Wada, B., "Adaptive Structures : An Overview," *J. Spacecraft*, Vol. 27, No. 3, May/June 1990.
- [Was65] Wass, C.A.A., Garner, K.C., Introduction to Electronic Analogue Computers, Pergamon Press, New York, 1965.
- [Web73] Webb, J.E., "The role of the notocord in forward and reverse swimming and burrowing in the amphioxus *Branchiostoma lanceolatum*," *J. Zool.*, Lond., Vol. 170, pp. 325-338, 1973.

- [Webs83] Webster's Ninth New Collegiate Dictionary, Merriam-Webster Inc., Publishers, Springfield, Mass., 1983.
- [Wei74] Weinstock, R., Calculus of Variations with Applications to Physics and Engineering, Dover, New York, 1974.
- [WilS88] Wilson, J.F., Snyder, J.M., "The Elastica with End-Load Flip-Over," *Journal of Applied Mechanics*. Vol. 55, December 1988.
- [WilM89] Wilson, J.F., Mahajan, U., "The Mechanics and Positioning of Highly Flexible Manipulator Limbs," *Journal of Mechanisms, Transmissions, and Automation in Design*, Vol. 111, June 1989.
- [WinK70] Winet, H., Keller, S.R., "Spirillum Swimming: Theory and Observations of Propulsion by the Flagellar Bundle," *J. Exp. Biol.*, Vol. 65, pp. 577-602, 1970.
- [Wu86] Wurst, K.H., "The Conception and Construction of a Modular Robot System," *Proceedings of the 16th International Symposium on Industrial Robotics*, pp. 37-44, 1986.
- [Ya56] Yapp, W.B., "Locomotion of worms," *Nature, Lond.*, Vol. 177, pp. 614-615, 1956.
- [YoKT91] Yokoi, K., Komoriya, K., Tanie, K., "A Method for Solving Inverse Kinematics of Variable Structure Truss Arm with High Redundancy," *Proceedings of the 2<sup>nd</sup> Joint Japan-USA Conference on Adaptive Structures*, Nagoya, Japan, Nov. 12-14, 1991.