

Copyright © 1991, by the author(s).  
All rights reserved.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission.

**ON REORIENTING LINKED RIGID BODIES  
USING INTERNAL MOTIONS**

by

Gregory C. Walsh and Shankar Sastry

Memorandum No. UCB/ERL M91/35

2 May 1991

COVER ME

**ON REORIENTING LINKED RIGID BODIES  
USING INTERNAL MOTIONS**

by

Gregory C. Walsh and Shankar Sastry

Memorandum No. UCB/ERL M91/35

2 May 1991

**ELECTRONICS RESEARCH LABORATORY**

College of Engineering  
University of California, Berkeley  
94720

TITLE PAGE

**ON REORIENTING LINKED RIGID BODIES  
USING INTERNAL MOTIONS**

by

Gregory C. Walsh and Shankar Sastry

Memorandum No. UCB/ERL M91/35

2 May 1991

**ELECTRONICS RESEARCH LABORATORY**

College of Engineering  
University of California, Berkeley  
94720

### **Abstract**

The goal of this paper is to provide kinematic algorithms for reorienting linked rigid bodies floating in space. Since they are floating in space, they conserve both angular and linear momentum. The conservation of linear momentum is an integrable constraint, the conservation angular momentum is not. We dualize the angular momentum constraint so we may view the problem of reorientation as a steering problem for a drift free control system which obeys these constraints. Explicit solutions via an algorithm are given are given for several example systems and a general system.

<sup>1</sup>Supported in part by and NSF Graduate Fellowship and NSF grant IRI9014490

<sup>2</sup>Supported in part by NSF grant IRI9014490

# Contents

<b>1</b>	<b>Introduction</b>	<b>3</b>
<b>2</b>	<b>Derivation of Kinematic Equations</b>	<b>5</b>
2.1	The Lagrangian . . . . .	5
2.2	The Momentum Map . . . . .	7
2.3	The Kinematic Equations . . . . .	8
2.4	The Two Dimensional Case . . . . .	9
<b>3</b>	<b>Steering the Two Dimensional System</b>	<b>12</b>
3.1	Controllability . . . . .	12
3.2	Reconstruction . . . . .	13
3.3	Steering Algorithms . . . . .	16
3.4	Car Parking and Optimization . . . . .	18
<b>4</b>	<b>Simulations and Hardware</b>	<b>20</b>
4.1	Hardware Design . . . . .	20
4.2	Results . . . . .	24
<b>5</b>	<b>Steering the Three Dimensional System</b>	<b>27</b>
5.1	Special Case: A Satellite with Two Momentum Wheels . . . . .	27
5.1.1	The Kinematic Equations . . . . .	28
5.1.2	Parallel Parking the Satellite . . . . .	30
5.2	The Shape Space for the General Case . . . . .	33
5.3	The Steering Algorithm . . . . .	34
<b>6</b>	<b>Conclusion</b>	<b>36</b>
A	Proof of Proposition 1 . . . . .	37
B	Example Satellite 1 . . . . .	41
C	Proof of Proposition 3 . . . . .	42
D	Example Satellite 2 . . . . .	44
E	Computation of the Lie Bracket . . . . .	45

# List of Figures

2.1	A Space Robot Consisting of Three Linked Rigid Bodies . . . . .	6
2.2	The Planar Skater: A Two Dimensional Restriction . . . . .	10
2.3	The Unicycle . . . . .	11
3.1	Three Dimensional Plot of the Determinant . . . . .	14
3.2	Phase Shift as a Function of Input Amplitude . . . . .	16
3.3	Steering in Singular Configurations . . . . .	17
3.4	Parallel Parking for the Unicycle . . . . .	18
4.1	System Layout . . . . .	21
4.2	Options for the Configuration . . . . .	22
4.3	Direct Joint Design . . . . .	23
4.4	Experimental Data Showing Controller Performance . . . . .	24
4.5	Plot of $\psi_3$ on the Hardware . . . . .	25
4.6	Simulated $\psi_3$ with Similar Inputs . . . . .	26
5.1	Satellite with Two Rotors . . . . .	28
5.2	The Rotation Group as a Circle Bundle . . . . .	31
5.3	The Parallel Park Maneuver for the Satellite . . . . .	33
5.4	Reducing the Three Dimensional System to the Planar Case . . . . .	34
6.1	Robot in Home Configuration . . . . .	37

# Chapter 1

## Introduction

The problem of reorienting series of linked rigid bodies has been studied before. The motivation for the current paper was a paper written by Nakamura and Mukerjee [NM89] which used a Lyapunov design methodology to find a smooth feedback law which would reorient a satellite which had a Puma arm attached. Although there was much success, the control law failed to achieve some orientations because of the nonholonomic nature of the system. The control system would get stuck and stop. A natural question would then be: is there a possible path at all and how to find such a path.

The goal of this paper is to provide control inputs to joint motors which will drive this kind of system from one orientation to another. Although this work may be further generalized to other systems, we will focus on systems of linked rigid bodies floating in space. For a treatment of the subject in the plane, [Byr90] proves interesting. If one is not concerned with controlling the orientation of the body as well as the positions of the manipulator, the reduced system will have less complicated velocity constraints. See [PD90] for more details on the control of such systems.

Other interesting, related systems include falling cats [Mon90], floating astronauts [KHY72], spring board divers [Fro79], satellites with rotors [LG90], and oddly enough car parking and swimming. The similarities to car parking will be carefully explored because we are familiar with steering a car and with some modification the procedure used to park a car can be used in this problem. In discussing car parking, we will refer to the treatment of the subject given in Murray and Sastry [MS90] and in [LS91b] or in [NvdS90].

In particular we will focus on the problem of reorienting three linked rigid bodies. The problem of two linked bodies is well examined by Richard Montgomery [Mon90] in his paper on falling cats, and no other degrees of freedom are obtained by adding more links than three. As the algorithm developed will respect joint angle limitations, the multi-link problem can be solved using trajectories for the three link case by ignoring the extra degrees of freedom.



It is interesting to note that this satellite problem belongs to a large class of engineering problems that do not have nilpotent vector fields. A nilpotent vector field is one where the Lie bracket operation will terminate (yield zero as the resultant vector field) after some fixed number of operations. For more information on the Lie bracket see [Spi70]. The satellite problem's vector fields are left invariant and belong to the tangent bundle of the space of rotations,  $SO(3)$ , like many other mechanical devices. The tangent space at the identity is commonly identified with  $\mathbb{R}^3$  and taking Lie brackets there amounts to taking the cross product of the vectors in  $\mathbb{R}^3$  given we have the left invariance. Given any two independent vector fields the termination will never occur as the cross product will yield a vector perpendicular to the starting vectors. Therefore work on nilpotent Lie algebras due to [LS90] or [LS91a] apparently can not be applied here.

The main point to the method presented here is the organization of the configuration space into a part without velocity constraints and a part which we will not exercise direct control over. The space without velocity constraints is aptly called shape space, for it corresponds exactly to relative positions of each of the links of the satellite, or in other words its shape. The problem of finding the motion in the indirect part of the space given the motion in shape space is called reconstruction, and the net motion we find for some path in shape space will be called the geometric phase of the path.

## Chapter 2

# Derivation of Kinematic Equations

In this chapter first the expression for the Lagrangian is developed for the three linked rigid bodies. The next part derives the equations for the angular momentum using the concept of momentum mapping. The next section derives a kinematic map for the system. Finally in §2.4 the equations for the restricted planar case are developed so that they may be used in examples.

### 2.1 The Lagrangian

First we will derive the expression for the Lagrangian for the system of three linked rigid bodies. If we assume the effect of the inhomogeneity of the gravitational field to be negligible, notice that the Lagrangian is also the Hamiltonian or energy. It is interesting to note that because there is no potential energy, all solution trajectories to the system without input will be geodesics with the Riemannian structure induced by the Lagrangian on the configuration space. The solution trajectories will be geodesics because the Lagrangian is conserved, and since the Lagrangian is only kinetic energy, this implies that the length of the velocity vector will never change when measure with the Lagrangian metric. Therefore, minimizing the integral of the Lagrangian is equivalent to finding the shortest path between two points in the configuration space.

In keeping with Patrick [Pat90], the orientation of each of the three bodies will be represented by the  $3 \times 3$  matrix needed to rotate the body's frame onto some fixed inertial frame. These matrices will be denoted  $R_1$ ,  $R_2$ , and  $R_3$  respectively. The space of such matrices is denoted  $SO(3)$ , or special orthogonal  $3 \times 3$  matrices. Besides being orthogonal, they have determinant equal to +1.

Given the orientation of each body and how the bodies are connected, now only the position in three-space of a point on any one of the links is needed to

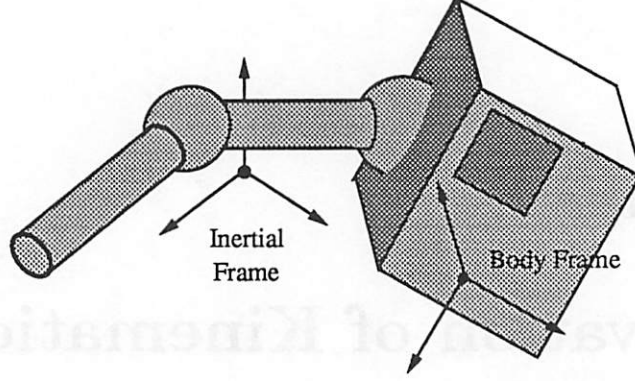


Figure 2.1: A Space Robot Consisting of Three Linked Rigid Bodies

completely specify any other point on the system. We will choose the center of mass of the middle or second link to be that reference point. Call this vector  $c \in \mathbb{R}^3$ .

To summarize, the configuration space is  $Q = SO(3)^3 \times \mathbb{R}^3$ . Each point in  $Q$  is given by three rotation matrices  $R_1, R_2, R_3 \in SO(3)$ ; and one vector  $c \in \mathbb{R}^3$ . In addition, define the relative rotation matrices to be  $R_{ij} = R_i^T R_j$ . The angular velocity of some body will be measured in that body's coordinate frame. So,  $(\omega_i \times) := R_i^T \dot{R}_i$ . In words,  $\omega_i \in \mathbb{R}^3$  is the angular velocity of  $i^{th}$  body in the body's own frame.

**Proposition 1** *The Lagrangian of the system is given by:*

$$L = [\omega_1^T, \omega_2^T, \omega_3^T] J(R_{12}, R_{13}, R_{23}) \begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{bmatrix} \quad (2.1)$$

with:

$$J(R_{12}, R_{13}, R_{23}) := \begin{bmatrix} M_{11} & M_{12}R_{12}K_{12} & M_{13}R_{13}K_{13} \\ K_{12}^T R_{12}^T M_{12}^T & M_{22} & M_{23}R_{23}K_{23} \\ K_{13}^T R_{13}^T M_{13}^T & K_{23}^T R_{23}^T M_{23}^T & M_{33} \end{bmatrix}$$

where the  $M_{ij}, K_{ij} \in \mathbb{R}^{3 \times 3}$  are constant  $3 \times 3$  matrices which depend on the physical parameters of the system.

The proof may be found in the appendix. One interesting quantity which arises in the proof is a vector representing the direction of the line connecting the two “shoulders” or joints of the robot. This is defined as:

$$z = r_1 - r_3 \quad (2.2)$$

As will be shown later, this axis is the most difficult to rotate about. The next section in the appendix computes an example finding the  $3 \times 3$  matrices in the above proposition.

## 2.2 The Momentum Map

The Lie Group  $SO(3)$ , consisting of  $3 \times 3$  unitary matrices with positive determinant, acts on the configuration space  $Q$  by rotations. Define the action for some  $A \in SO(3)$  to be  $\mathcal{A} : (c, R_1, R_2, R_3) \rightarrow (Ac, AR_1, AR_2, AR_3)$ . Heuristically, this action just picks up the entire system and rotates it with respect to the inertial frame by  $A \in SO(3)$ . Notice that under this action the value of the Lagrangian is invariant because  $\mathcal{A} : R_{ij} \rightarrow R_i^T A^T A R_j = R_{ij}$ . It will prove useful to find the the quotient of the configuration space and the group under which the Lagrangian and Hamiltonian are constant. Define this quotient  $Q/SO(3) = Sh_1$ , or shape space one. It will identify all points in the configuration space for which there exists a matrix  $A \in SO(3)$  such that the action maps the first point into the other. In this manner we can view  $Q$  as a trivial  $SO(3)$  bundle over shape space.

**Remark:** The quotient space formed above is indeed a shape space by the definition given earlier. In other words, there are no velocity constraints and thus any path in the space may be achieved. The proof is straightforward. The coordinates of this quotient space can be represented by the matrices  $R_{12}, R_{13} \in SO(3)$ . By assumption, we have control over the joints and thus the relative position of the three bodies. Thus any continuous path in  $Sh_1$  may be followed.

This action lifts to an action on the velocity phase space. It has been noted in general [MR90] that when one has an action preserving the Lagrangian, there exists a conserved quantity in the Lie algebra of the group of the action. See Noether's theorem [MR90] for details. It will be expressed in terms of the reduced coordinates of the quotient space. As it turns out, this quantity is the total angular momentum in this case. There is a general formula for computing this quantity, and if we identify the Lie algebra of  $SO(3)$  with  $\mathbb{R}^3$  in the usual way, then the formula to find this conserved quantity is as follows. Define  $AM$  to be this angular momentum.

$$AM = [I \ I \ I] J(R_{12}, R_{13}, R_{23}) \begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{bmatrix}$$

Note that the total angular momentum is just the sum of the the momentum vectors obtained through the Legendre transform.

## 2.3 The Kinematic Equations

Next we will put the equations of motion in a more convenient form. They will show how the system evolves depending on the joint velocities. To do this we will rewrite the constraint of conservation of angular momentum, now using the assumption that the total angular momentum of the system is zero. To do this, as in Li and Gurvitz [LG90], we will find the null space of the  $J$  matrix. In other words, find the directions the system can move without changing the angular momentum. These motions will be called internal motions. Define a few quantities first.

$$\begin{aligned}
 A_1 &:= M_{11} + M_{12}R_{12}K_{12} + M_{13}R_{13}K_{13} \\
 A_2 &:= K_{12}^T R_{12}^T M_{12}^T + M_{22} + M_{23}K_{23}R_{23} \\
 A_3 &:= K_{13}^T R_{13}^T M_{13}^T + K_{23}^T R_{23}^T M_{23}^T + M_{33} \\
 AM &= [A_1 \ A_2 \ A_3] \begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{bmatrix} \\
 V &= [A_1 + A_2 + A_3]^{-1}
 \end{aligned}$$

Assume that the inverse in the formula for  $V$  exists. Now pick the nullspace vectors so they will correspond to inputs we know about: the relative velocities of the bodies. Each  $B_i : Q \rightarrow \mathbb{R}^{9 \times 3}$  gives at each point in the configuration space the three vectors associated with the internal motions of the  $i^{th}$  3 degree of freedom joint.

$$\begin{aligned}
 B_1 &= \begin{bmatrix} I - VA_1 \\ -VA_1 \\ -VA_1 \end{bmatrix} \\
 B_2 &= \begin{bmatrix} I - VA_3 \\ -VA_3 \\ -VA_3 \end{bmatrix}
 \end{aligned}$$

Given these as our input vector fields for the control system, a simple change of coordinates will divide the space along the lines of the shape space  $Sh_1$ . This will also make the problem of steering the system look more like the parallel parking problem [MS90]. Label the inputs  $u_1, u_2 : t \rightarrow \mathbb{R}^3$ . Define  $\phi_1, \phi_2, \phi_3 \in \mathbb{R}^3$ :

$$\phi_1 = \omega_1 - \omega_2$$

$$\begin{aligned}\phi_2 &= \omega_3 - \omega_2 \\ \phi_3 &= \omega_2\end{aligned}$$

Then we derive:

$$\begin{bmatrix} \dot{\phi}_1 \\ \dot{\phi}_2 \\ \dot{\phi}_3 \end{bmatrix} = \begin{bmatrix} I \\ 0 \\ -VA_1 \end{bmatrix} u_1 + \begin{bmatrix} 0 \\ I \\ -VA_3 \end{bmatrix} u_2 \quad (2.3)$$

An interpretation of the inputs is required. We claim that these inputs correspond to the control inputs at the joints. We will refer to these joints as the shoulders.

$$\begin{aligned}\dot{R}_{12} &= \dot{R}_1^T R_2 + R_1^T \dot{R}_2 \\ &= (\omega_1 \times)^T R_1^T R_2 + R_1^T R_2 (\omega_2 \times) \\ &= R_{12} ((\omega_2 - \omega_1) \times) \\ &= -R_{12} (u_1 \times)\end{aligned} \quad (2.4)$$

So the two sets of equation, (2) and (3) above, specify a differential equation describing how the matrices  $R_1, R_2, R_3$  change depending on how we drive the joints.

**Remark:** Assuming there is zero angular momentum, then the trajectories of the system corresponding to  $AM = 0$  will also be the solutions of (3) provided  $u_1, u_2$  are chosen to match the systems joint velocities.

This fact is due to conservation of angular momentum. By contradiction one may see that if the two solutions do not agree, then one may find an instance where angular momentum was not conserved for the real system.

## 2.4 The Two Dimensional Case

Now we restrict attention to the simpler two dimensional case. We will assume that the plane of restriction is the  $xy$  plane. Only the twist about the  $z$  axis is permitted, so set  $\theta_i = (0, 0, \omega_i)^T$ , with  $\omega_i \in \mathbb{R}$ .

For simplicity, assume that in the home configuration, the centers of mass of each body lie along the  $x$  axis. Notice that in this reduced system, each  $SO(3)$  is restricted to be rotations in the plane or  $SO(2)$  and this space may be parameterized by  $\theta_i \in S^1$ . Thus  $\theta_i$  will be the amount the  $i^{th}$  body must rotate to get it back to the orientation it had in the home configuration.

One way to visualize the two dimensional restriction is as a planar skater. A planar skater consists of three rigid bodies connected by revolute joints. It is placed on a ice rink or air table and so can translate and rotate with little friction. The friction will be ignored.

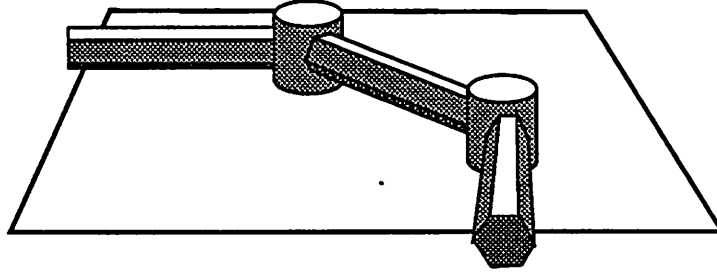


Figure 2.2: The Planar Skater: A Two Dimensional Restriction

**Proposition 2** Set  $u_1, u_2 : \mathbb{R} \rightarrow \mathbb{R}$  to be the relative angular velocities of the two joints as a function of time. If we assume the total angular momentum is zero, the equations of motion for the three rigid, linked bodies restricted to move in the plane are given by the following:

$$\begin{bmatrix} \dot{\psi}_1 \\ \dot{\psi}_2 \\ \dot{\psi}_3 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ b_1(\psi_1, \psi_2) \end{bmatrix} u_1 + \begin{bmatrix} 0 \\ 1 \\ b_2(\psi_1, \psi_2) \end{bmatrix} u_2 \quad (2.5)$$

With the following definitions:

$$\begin{aligned} \psi_1 &= \theta_1 - \theta_2 \\ \psi_2 &= \theta_3 - \theta_2 \\ \psi_3 &= \theta_2 \end{aligned}$$

$$\begin{aligned} a_1(\psi_1, \psi_2) &= k_{11} + k_{12} \cos(\psi_1) + k_{13} \cos(\psi_1 - \psi_2) \\ a_2(\psi_1, \psi_2) &= k_{12} \cos(\psi_1) + k_{22} + k_{23} \cos(\psi_2) \\ a_3(\psi_1, \psi_2) &= k_{13} \cos(\psi_1 - \psi_2) + k_{23} \cos(\psi_2) + k_{33} \\ b_1(\psi_1, \psi_2) &= -a_1(\psi_1, \psi_2) (a_1(\psi_1, \psi_2) + a_2(\psi_1, \psi_2) + a_3(\psi_1, \psi_2))^{-1} \\ b_2(\psi_1, \psi_2) &= -a_3(\psi_1, \psi_2) (a_1(\psi_1, \psi_2) + a_2(\psi_1, \psi_2) + a_3(\psi_1, \psi_2))^{-1} \end{aligned}$$

Where  $k_{ij}$ 's are scalar constants given by the physical parameters of the system.

**Remarks:** First, the variables  $\psi_1$  and  $\psi_2$  parameterize a shape space for the planar skater. Second, the form of the equations looks a lot like the form for steering a car. To support this assertion, suppose we look at the kinematic equations of the unicycle. Following Murray and Sastry [MS90], we will denote the positions of the center of the vehicle by two coordinates  $x, y \in \mathbb{R}$  and the angle the wheel makes with the  $x$  axis as  $\theta \in S^1$ . Set  $v_2$  to be the velocity of the steering input, and  $v_1$  to be  $\cos \theta$  times the forward velocity of the car.

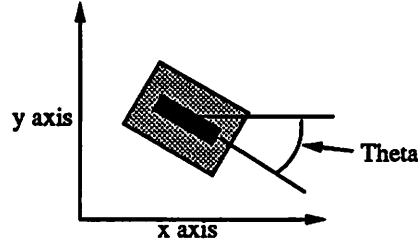


Figure 2.3: The Unicycle

If we assume that  $\theta$  is not  $\frac{\pi}{2}$  then we can transform the system so the system has the canonical form as in [MS90]. We are introducing this example now for the analogy it provides.

$$\begin{aligned}\dot{x} &= v_1 \\ \dot{\theta} &= v_2 \\ \dot{y} &= (\tan \theta) v_1\end{aligned}$$

Examination of these equations reveals the similar structure to the planar skater's. The difference lies in the how the difficult steering direction,  $y$ , is related to motions in shape space. Here shape space, the space we can move holonomically in, is parameterized by  $x \in \mathbb{R}$  and the steering direction  $\theta \in S^1$ .

Further, one can linearize this model following Brockett's [Bro81] example as an approximation. In this case the  $\tan \theta$  becomes  $\theta$ .

**Proof:** The proof of the proposition is once again more algebra and can be found in the appendix. It involves taking the general equations and restricting them to the plane.



## Chapter 3

# Steering the Two Dimensional System

The two dimensional system is analyzed first because it manages to embody most of what makes these systems an interesting problem while only involving one level of brackets. By one level of brackets, we will mean that the input vector fields and their Lie brackets will span enough the tangent space to the configuration space. Higher level of brackets will just mean that movement in some directions will require taking brackets of the vector fields produced by brackets. A movement along the direction given by the bracket of two vector fields will be called a bracket motion. The three dimensional case involves two levels and the path planning proposed for it will use the two dimensional case as a building block. The traditional method of testing the controllability of the two dimensional system produces some results which will motivate the path planning algorithm. In fact, some of the results will generalize to other systems. In this spirit we will examine car parking. The algorithm bears some resemblance to car parking and will be discussed as well as some methods towards finding paths requiring the least control effort.

### 3.1 Controllability

At first, one would like to apply Chow's theorem [Isi89] to check the controllability of the the planar system developed in chapter 2. We will define controllability as follows. Given any initial point in configuration space  $\psi_1, \psi_2, \psi_3$ ; and any final point there exist a time  $T$  and some input to the system defined on the interval  $[0, T]$  such that at time 0 the system is at the initial point and at time  $T$  the system is at the final point.

Notice as the system is autonomous, we can scale the inputs as to fit the same result into a smaller and smaller time interval, so the  $T$  requirement is not

important.

Chow's theorem states that a system with analytic vector fields is controllable if the involutive closure of the input vector fields spans the entire configuration space. To find the involutive closure, we must start taking Lie brackets. See [War83] for Lie brackets. So we will take the Lie bracket of the input vector fields labeled them  $v_1(\psi_1, \psi_2), v_2(\psi_1, \psi_2)$ .

In coordinates, the Lie bracket of two vector fields  $v_1(\psi_1, \psi_2), v_2(\psi_1, \psi_2)$  is given by:

$$[v_1, v_2] = \frac{\partial v_2}{\partial \psi} v_1 - \frac{\partial v_1}{\partial \psi} v_2 \quad (3.1)$$

The Lie bracket of the input vector fields of the planar system is given by the following, using the definitions given in the last section.

$$[v_1, v_2] = \begin{bmatrix} 0 \\ 0 \\ \xi(\psi_1, \psi_2) \end{bmatrix}$$

$$\xi(\psi_1, \psi_2) := 2v^2 (a_1 k_{23} \sin(\psi_2) - a_3 k_{12} \sin(\psi_1) + a_2 k_{13} \sin(\psi_1 - \psi_2)) \quad (3.2)$$

The proof is a matter of computation and is left to the appendix.

Now we would like to check the controllability condition. To do so, we put  $v_1, v_2, [v_1, v_2]$  into a matrix and find its determinant. We will call this the first level controllability matrix. It is easy to check that its determinant will just be  $\xi(\psi_1, \psi_2)$  which one may see is zero at many places inside the shape space.

Although this appears superficially to imply the loss of controllability, the result is really inconclusive because more Lie brackets may yield enough rank vector fields to span the tangent space. However, this matrix being singular is a poor sign for most schemes in non-linear control. We will refer to the configurations where the determinant of the first level controllability matrix is zero as singular configurations. From the formula (7) one may see that the folded and extended configurations of any type are singular. That is,  $(\psi_1, \psi_2) = (0, 0), (\pi, 0), (0, \pi), (\pi, \pi)$  are singular. In fact, the set of singular configurations is a one dimensional manifold in shape space.

## 3.2 Reconstruction

We will now proceed to view the problem from a more modern mechanical point of view. Given any path in shape space, we would like to reconstruct the full state of the system. This just involves finding  $\psi_3$ . The reconstruction will tell us directly how to plan paths.

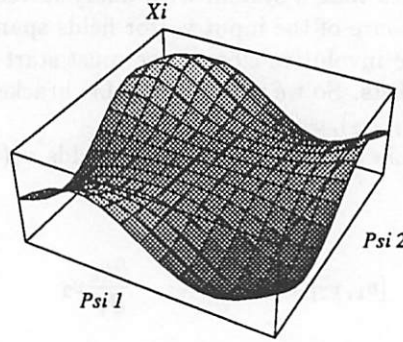


Figure 3.1: Three Dimensional Plot of the Determinant

The solution to the reconstruction of  $\psi_3$  is an application of Green's theorem. We have a formula for finding  $\psi_3$  as a line integral in shape space. First, we start with the case where the path in shape space is a simple closed curve. Then the following results, assuming without loss of generality that the path is one second long.

$$\begin{aligned}
 \psi_3(1) - \psi_3(0) &= \int_{\text{time}} (-va_1 u_1 - va_3 u_2) dt \\
 &= \int_{\text{path}} (-va_1 d\psi_1 - va_3 d\psi_2) \\
 &= \int_{\text{area}} \left( \frac{\partial}{\partial \psi_1} (-va_1) - \frac{\partial}{\partial \psi_2} (-va_3) \right) dA
 \end{aligned}$$

The term in the brackets, however, is familiar. We had just calculated it when finding the Lie bracket. Thus we may write the following.

$$\psi_3(1) - \psi_3(0) = \int_{\text{area}} \xi(\psi_1, \psi_2) dA \quad (3.3)$$

This quantity is referred to as the geometric phase, and for a positively oriented simple closed curve in shape space it is equal to the determinant of the first level controllability matrix integrated over the area enclosed by that loop. In general, the procedure for finding the geometric phase can be broken down into three increasingly complicated cases.

**Case 1** The path is a simple closed curve. The positive orientation case is already done, and if the curve was negatively oriented the result just changes sign.

**Case 2** The path is a self-intersecting closed curve. Then divide the curve into simple closed pieces and apply case 1 to each piece, adding the result together.

**Case 3** The path is an open curve. Join the end points with a straight line and compute the line integral over that line. Keep the result. Now look at the path with this line attached, and note it is a possibly self intersecting closed curve. Apply case 1 or 2 and then subtract the result of the straight line integral.

This tantalizing relationship, the equality between the integrand and determinant of the first level controllability matrix, is unfortunately not true in general. It is true, however, in many interesting systems including the unicycle of Murray and Sastry [MS90], a satellite with two rotors, and with this case. One may ask what classes of systems have this intuitive property.

**Proposition 3** Consider a 3 dimensional system with two inputs of the form:

$$\begin{aligned}\dot{x}_1 &= u_1 \\ \dot{x}_2 &= u_2 \\ \dot{x}_3 &= g_1 u_1 + g_2 u_2 \\ g_1, g_2 : (x_1, x_2, x_3) &\rightarrow \mathbb{R}\end{aligned}$$

Further, assume that it is strictly triangular in the sense that  $g_1, g_2$  depend only on  $x_1, x_2$ . Then given any path in the space parameterized by  $x_1, x_2$  which is a simple closed curve, the net change in  $x_3$  after traversing this loop will be equal to the determinant of the first level controllability matrix integrated over the area enclosed by the loop.

**Remark:** Notice that many systems may be transformed by coordinate transformations (the planar skater) and input transformations (the unicycle) to the canonical form. Strict triangularity is a notion from Murray and Sastry, and it can be employed for multiple bracket levels in the same way for reconstruction as it is with steering with sinusoids in their paper.

**Proof:** To prove the proposition, one may notice that because of the special form of the input vector fields that the determinant of the the first level controllability matrix will be as follows.

$$\det = \frac{\partial g_2}{\partial x_1} - \frac{\partial g_1}{\partial x_2} + \frac{\partial g_2}{\partial x_3} g_1 - \frac{\partial g_1}{\partial x_3} g_2$$

Now if we employ the assumption that the system is strictly triangular, then the result follows.  $\square$

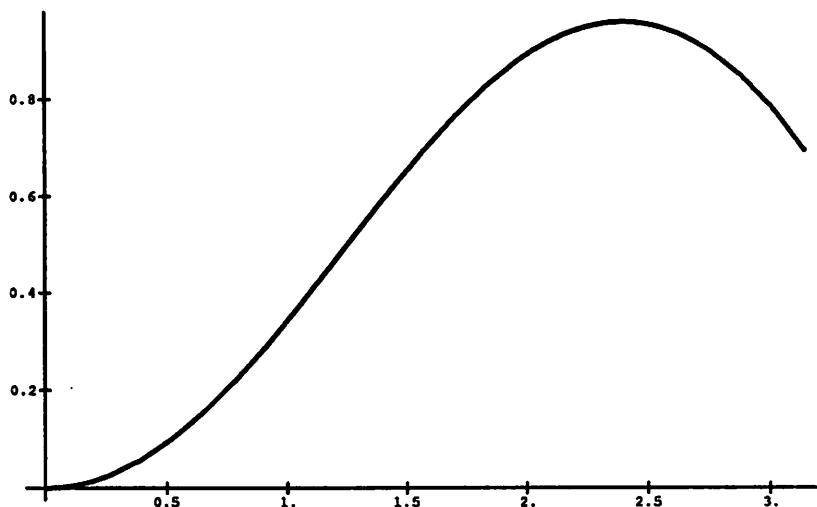


Figure 3.2: Phase Shift as a Function of Input Amplitude

### 3.3 Steering Algorithms

With the reconstruction procedure before us, one method of choosing paths is fairly clear. Notice that the weighting factor  $\xi$  is not zero at the point  $(\psi_1, \psi_2) = (\frac{1}{2}\pi, -\frac{1}{2}\pi)$  and so  $\xi$  being continuous implies that there exists an open subset of the shape space where it is not zero and consequently of the same sign. Any simple closed curve enclosing this area will have non-zero geometric phase which, by the way, changes sign if the direction traveled in this path is reversed. This result implies controllability in the sense of Chow in a constructive manner.

The algorithm that follows is based on that principle. All that is needed is a way to choose these loops in shape space. In doing so, we will apply the steering by sinusoids method given in Murray and Sastry for parking the unicycle.

Given any initial configuration  $(\psi_1, \psi_2, \psi_3)$ , and a final one, the final one may be reached using this procedure.

**Step 1** Using constant inputs, drive the system to the desired  $\psi_1, \psi_2$  ignoring the drift in the  $\psi_3$  term. Measure the amount of phase shift required to bring the system to the exact desired location.

**Step 2** Again using constant inputs, drive the system to  $(-\pi/2, -\pi/2 + k)$  where  $k$  is the amplitude chosen from the graph below (figure 5). For given parameter values, this graph gives the shift in  $\psi_3$  as a function of the amplitude the driving sinusoids, applied for one period. The offset is to ensure the path is centered on  $(\psi_1, \psi_2) = (-\pi/2, \pi/2)$ .

Notice that this function reaches a maximum, because bigger circles start to enclose areas of the shape space where the  $\xi$  changes sign.

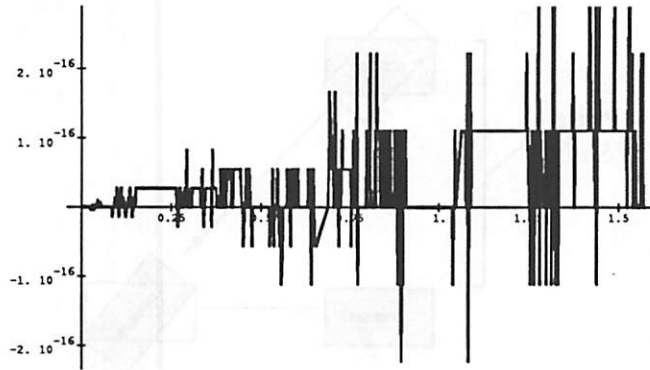


Figure 3.3: Steering in Singular Configurations

$k$  is selected to get this desired phase shift in  $\psi_3$ . If the needed phase shift is larger than the maximum value shown in figure 5, then two or more loops may be required.

**Step 3** Drive the system with  $u_1 = k \sin(t)$ ,  $u_2 = k \cos(t)$  with enough cycles to obtain the desired phase shift.

**Step 4** Return the system back to the desired  $\psi_1, \psi_2$  by the negative of the constant inputs used in step two. Because this return path and the outgoing path enclose no area, the phase shift of this path is equal to the phase shift obtained in step three only.

Note that if the system did loops in shape space about a singular configuration, it would enclose roughly equal areas of positive and negative  $\xi$ , and so the motion would have little if any net effect. Figure 6 is a similar graph, the only difference being that the loops are now centered at  $(\psi_1, \psi_2) = (0, 0)$ .

Notice the small values and discontinuity in figure 6. The round off error inherent in the computer is enough to account for the shape of the graph. This is why the traverse step is needed: to ensure that the sinusoidal steering input has any net effect. Of course, step one may be skipped if one is willing to compute the needed phase shift by hand.

Notice that, by using multiple loops if necessary, we may move in the  $\psi_3$  direction. Even with joint angle limitations, there still will exist an open subset of the shape space in which the system is free to move. As the set of points in shape space where  $\xi = 0$  is a one dimensional manifold, there will exist a feasible closed path enclosing an area of shape space where  $\xi$  has only one sign. With this loop and other similar ones inside of it one may design an analogous path planner for the restricted system.

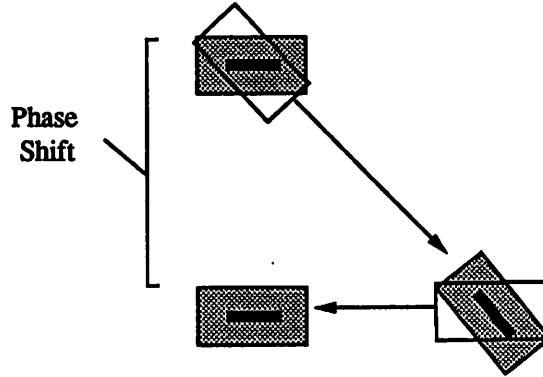


Figure 3.4: Parallel Parking for the Unicycle

### 3.4 Car Parking and Optimization

One may check, in the case of the linearized unicycle, that the determinant of the first level controllability matrix is just equal to  $+1$ . therefore, the geometric phase for the system is just the area enclosed by the path. As a consequence, the result will not depend on the linearized unicycle's position in its shape space unlike the planar skater.

The difficult maneuver for a linearized unicycle is to get a side ways translation. It is exactly in the direction of the Lie bracket of the input vector fields. Figure 7 shows a parallel park maneuver consisting of straight line segments forming a square in shape space.

In Murray and Sastry [MS90], an optimization is performed on this linearized version of the unicycle and the optimal paths for the parallel park maneuver are circles.

Viewed from the context of the reconstruction, the result follows because the length of the path in shape space is exactly the control effort they try to minimize. It is well known that the circle encloses the maximum area with the minimum circumference.

The two dimensional system has an analogous situation in step three of the path planning algorithm. At this point we wish to move exclusively in the bracket direction, or in other words, parallel park the satellite.

Once again the length of the path in shape space is exactly the control effort required to move the two dimensional system. So the problem of optimizing this maneuver reduces to finding the minimum length path which will have the desired geometric phase. This situation is considered also in [Kri91].

**Proposition 4** *Given any strictly triangular system as in proposition (4) the*

optimal control inputs must follow the following differential equation:

$$\frac{d}{dt} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = \begin{bmatrix} 0 & -2\lambda\xi(\psi_1, \psi_2) \\ 2\lambda\xi(\psi_1, \psi_2) & 0 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$$

The function  $\xi(\psi_1, \psi_2)$  is, once again, the determinant of the controllability matrix.

**Remark:** Note that in the case of the linearized unicycle, the density factor  $\xi$  is constant and thus the optimal inputs are sinusoidal, as noted earlier, see [MS90] for details.

**Proof:** One may apply the method of Lagrange multipliers to minimize the control effort. We wish to minimize:

$$\int_0^1 (u_1^2 + u_2^2) dt = \int_0^1 \dot{\psi}_1^2 + \dot{\psi}_2^2$$

Recast the system as a constraint so that the minimization will be subject to the following constraint:

$$\dot{\psi}_3 - b_1(\psi_1, \psi_2)\dot{\psi}_1 - b_2(\psi_1, \psi_2)\dot{\psi}_2 = 0$$

Necessary conditions that an extremal solution should satisfy are the Euler-Lagrange equations with the constraint being multiplied by the multiplier  $\lambda$ .

$$\begin{aligned} L &= \frac{1}{2} (\dot{\psi}_1^2 + \dot{\psi}_2^2) + \lambda (\dot{\psi}_3 - b_1(\psi_1, \psi_2)\dot{\psi}_1 - b_2(\psi_1, \psi_2)\dot{\psi}_2) \\ \frac{\partial L}{\partial \dot{\psi}} &= \begin{bmatrix} \dot{\psi}_1 - \lambda b_1(\psi_1, \psi_2) \\ \dot{\psi}_2 - \lambda b_2(\psi_1, \psi_2) \\ \lambda \end{bmatrix} \\ \frac{\partial L}{\partial \psi} &= \begin{bmatrix} -\lambda \left( \frac{\partial b_1}{\partial \psi_1} \dot{\psi}_1 + \frac{\partial b_2}{\partial \psi_1} \dot{\psi}_2 \right) \\ -\lambda \left( \frac{\partial b_1}{\partial \psi_2} \dot{\psi}_1 + \frac{\partial b_2}{\partial \psi_2} \dot{\psi}_2 \right) \\ 0 \end{bmatrix} \\ \frac{d}{dt} \left( \frac{\partial L}{\partial \dot{\psi}} \right) &= \begin{bmatrix} \ddot{\psi}_1 - \lambda \left( \frac{\partial b_1}{\partial \psi_1} \dot{\psi}_1 + \frac{\partial b_2}{\partial \psi_1} \dot{\psi}_2 \right) \\ \ddot{\psi}_2 - \lambda \left( \frac{\partial b_1}{\partial \psi_1} \dot{\psi}_1 + \frac{\partial b_2}{\partial \psi_1} \dot{\psi}_2 \right) \\ \dot{\lambda} \end{bmatrix} \end{aligned}$$

Now apply the Euler-Lagrange equations. Notice that  $u_1 = \dot{\psi}_1$  and that  $u_2 = \dot{\psi}_2$ . Since  $\xi(\psi_1, \psi_2) = \frac{\partial b_1}{\partial \psi_2} - \frac{\partial b_2}{\partial \psi_1}$ , the proposition follows from the first two equations above.  $\square$



## Chapter 4

# Simulations and Hardware

We targeted the planar skater for simulations in both software and hardware because its restriction to the plane simplified both tasks. The goal was to demonstrate in software and hardware the reorientation algorithm presented in section 3. The simulations were run on Sun workstation and the results ported over to Silicon Graphics workstations for viewing. This viewing required new graphics software, the hardware project required the fabrication of an entirely new piece of hardware.

Free flotation in the plane was not attempted as it is a labor intensive effect to produce and achieving some approximation to conservation to angular momentum could be achieved by having one undriven joint. Provided there is no friction at this joint there will be conservation of angular momentum though no conservation of linear momentum. The apparatus was built to minimize the amount of friction in the undriven or free joint.

This friction in the rotation of the entire system, the torques induced by the tether, and the tilt error of the undriven joint shaft constitute the most serious differences between the hardware and the model. All are unmodeled in the simulator. As it turns out, the friction does not swamp the effect we wish to exploit and the tether only starts affecting the robot when it has been wound many times. The system, however, is very sensitive to any tilt error and so the apparatus is mounted on an adjustable platform to correct the error. This has successfully dealt with that error.

### 4.1 Hardware Design

The implementation was simplified by using LYMPH, a VME based multiprocessor environment currently in the E.E.C.S. Robotics Lab at U.C. Berkeley. It uses 68020 based computers with a VME bus to communicate. Communication with the robot is realized using a DT1401 (Data Translation) interface board.

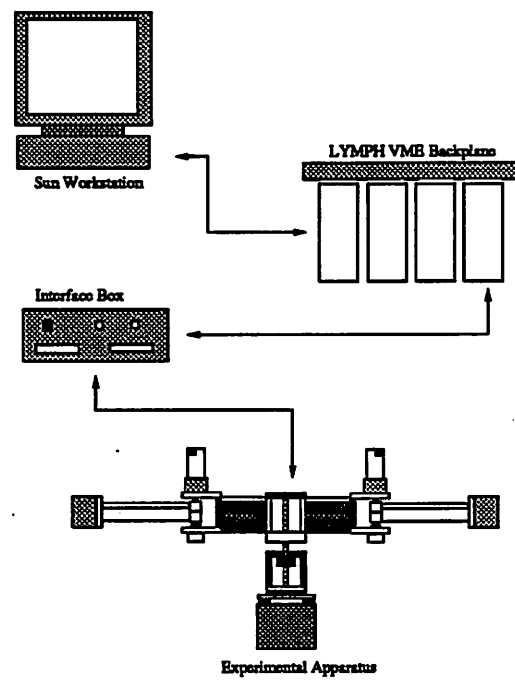


Figure 4.1: System Layout

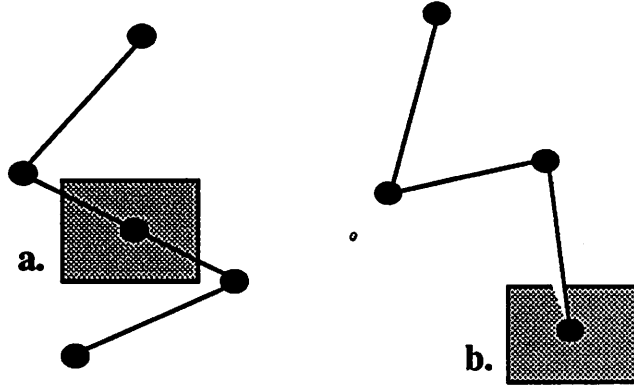


Figure 4.2: Options for the Configuration

The undriven joint is built simply with bearings, using an optical encoder to minimize the friction. Deciding where to put the undriven joint is the next order of business. The two points of most interest are the middle of the second link (a) and at the end of the last link (b). This last configuration makes the robot more like a planar acrobot. However attractive configuration (b) may be, it was not chosen simply because the robot would be at all times off balance and this would stress the bearings at the undriven joint, thus probably inducing large amounts of nonlinear friction invalidating the results.

A tether was chosen to transmit information and power to and from the robot. At first some sort of telephone handset cord suspended from the ceiling was used. However, it became clear that a simple length of 10 conductor ribbon cable induced far less torque.

Plastic was picked as the body material for two reasons. First, its light weight meant that the strategically placed metal weights would dominate the moments of inertias and would simplify their estimation. Second the problems associated with little chips of metal would be eliminated.

Most of the weight of the robot is in the motors and in the brass weights on the end of each arm. Each weight weighs 253 grams, each arm without the weight weighs 60 grams, each motor weighs 203 grams, and the center body without the motors weighs 115 grams. The distance from the pivot of the free joint to the pivot of each arm is 12.5 centimeters. The distance to the center of mass of the brass weight from the pivot joint is 16.2 centimeters.

After deciding that home configuration is the entirely folded one, which is not physically realizable, the  $k_{ij}$ 's may be computed. They are as follows, in  $\text{kgm}^2$ :

$$\begin{aligned} k_{11} &= k_{33} = 0.013825 \\ k_{12} &= k_{23} = -0.00624 \end{aligned}$$

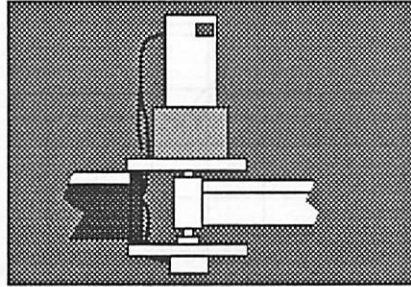


Figure 4.3: Direct Joint Design

$$k_{13} = -0.002176$$

$$k_{22} = 0.023575$$

The two arms do not have full  $2\pi$  rotation, so by the home configuration that was picked one may compute the off sets given the physical dimensions of the robot. So at the low end of the scale, the robot will hit physical limits at 0.848 radians, and at the higher end it will be stopped at 5.370 radians.

All other parts were chosen by availability. Pittman DC servo motors give the power, and those angles are read by potentiometers directly the joints. A belt design was considered but the direct drive design seemed safer and easier to implement.

The interface box contains all of the routing, the decoding for the optical encoder, the amplifiers, and any filtering for the potentiometers that was needed. The cable running to the ceiling tends to act like an antenna, picking up the noise for the lab. fortunately, this has not proved to be too serious a problem and three op amps are left for any future improvements.

The software written for this robot consists of three pieces. The first is `monitor.c`, which runs on the 68020 in the LYMPH cage. It is closest to the robot, providing all interfacing and low level control. It has many functions including being a basic monitor for the robot from which one may issue commands to move, ask about current positions and motor commands, implement a software emergency stop, calibrates, load in and run scripts, and save data. The control loops for each joint are simple sliding mode controllers with a bit of PE which run at 2500 hertz.

The second and third do interface work. The second, `scripter.c`, takes high level commands and writes open loop scripts for the robot to try to follow. The third runs real time graphing so one may watch the robot in action and check the performance of the joint controllers.

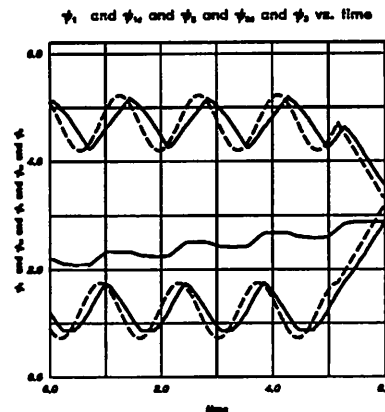


Figure 4.4: Experimental Data Showing Controller Performance

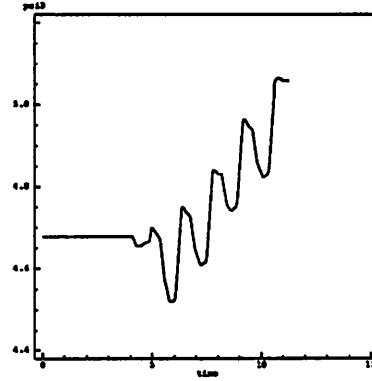
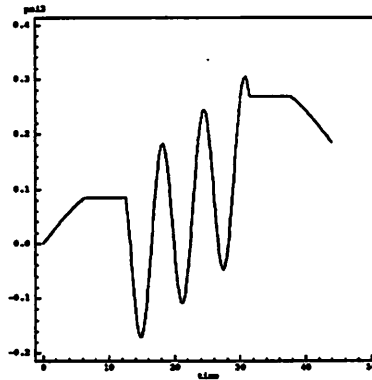
## 4.2 Results

Several tests were run to ascertain whether the system was actually demonstrating the effect desired. First, the tests were repeated with the tether unwound and wound in both directions, and it had very little effect for several rotations. The second test was checked the tilting problem. The arms may be run to move the system clockwise or counter clockwise for a full revolution provided the undriven joint shaft is straight up and down. The effect dies as the magnitude of the joint velocity dies, an effect due to the friction and not witnessed at all in the simulation.

Below is a graph of the robot running to generate positive motion as measured by the optical encoder. The dotted lines represent the desired trajectories and the solid lines represent the measured trajectories. The top two lines and the bottom two lines are the joints, the center being the undriven shaft as seen by the optical encoder. Notice the phase lag in the controlled joints. As the phase lags between the actual and desired position in the two joints are roughly equivalent the system is not hurt by this, and a 90 degree phase difference between the two is maintained. All units are in radians. Notice the steady ramping in the  $\psi_3$  variable.

What is shown below are two graphs, first a graph of the of the  $\psi_3$  variable as a the algorithm is executed, second, a plot of the simulation for similar inputs in the sense they have the same center points and amplitudes.

Notice that on the straight line maneuvers the hardware hardly moves at all, as stiction is swamping the meager torques at the undriven shaft. This of course is not witnessed in the simulation. Second the amplitudes are extremely

Figure 4.5: Plot of  $\psi_3$  on the HardwareFigure 4.6: Simulated  $\psi_3$  with Similar Inputs

different, the hardware being damped by friction so much.

In general, motion in the directions given by the Lie brackets of the input vector fields depends on the small difference between two much larger actions. If the system is not modeled correctly, the effect is greatly perturbed, for a small change in the large actions could easily consume this small difference.

## Chapter 5

# Steering the Three Dimensional System

With the two dimensional system carefully analyzed we may now use it as a building block for the three dimensional algorithm. It will be shown that all but a one dimensional subset of the configuration space may be reached arbitrarily by executing motions of the planar variety. Bracket motions of the planar variety are first level bracket motions. Motions that require a set of first level bracket motions will be called second level, and that is what will be needed to correct error in this one dimensional subset of the configuration space.

The chapter itself will be divided into three sections. The first presents a special case for our problem, the satellite with two rotors. This example will provide the model for the second level of bracket motions. The next section will define a first level shape space for the original system, motion in which will require first level brackets. Finally, the last section will present the algorithm based on the second part's results. This method on making layered shape spaces should prove useful in developing algorithms for any strictly triangular system requiring multiple levels of brackets, such as cars with multiple trailers.

### 5.1 Special Case: A Satellite with Two Momentum Wheels

Attitude control of a satellite can be achieved using carefully placed rocket thrusters. This method consumes rocket fuel and so has proven expensive. In addition, reservicing the craft often is out of the question, and so once the limited supply of rocket fuel is exhausted, control is lost. The solution to this problem is to attach rotors orthogonally to the craft. By spinning them, the attitude of the craft may be controlled. We will study the system where we

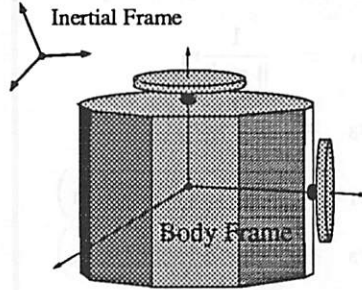


Figure 5.1: Satellite with Two Rotors

have only two momentum wheels. For path planning in this special case we will assume that the final orientation of the rotors is not significant.

**Assumption 1** The rotors are symmetric about their axis of rotation.

**Assumption 2** The rotors are attached so that their axis of rotation will intersect and will do so at only one point.

### 5.1.1 The Kinematic Equations

We can represent the satellite with two rotors as three linked, rigid bodies. The body of the satellite will correspond to the second link of the general system. The notation will carry over with some simplifications. First, the rotors can be modeled as having the joint located at the center of the mass of the satellite. In that case  $d_1 = 0$  and  $d_3 = 0$ . Consequently the form of the  $J$  will become diagonal and independent of configuration.

The angular momentum will of course have the same formula, so this part will carry over entirely.

$$\begin{aligned} AM &= [I \ I \ I] \begin{bmatrix} M_{11} & 0 & 0 \\ 0 & M_{22} & 0 \\ 0 & 0 & M_{33} \end{bmatrix} \begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{bmatrix} \\ &= M_{11}\omega_1 + M_{22}\omega_2 + M_{33}\omega_3 \end{aligned}$$

Again, to write the kinematic equations, we will assume the total angular momentum is zero. This simplifies the dynamics greatly and makes reconstructing the state of the system from the reduced coordinates easy. Now we should apply the hinge constraints. In doing so, it is useful to define the normals in the direction of  $r_1$  and  $r_3$  which were defined in appendix A. These vector constants will point from the center of mass of second body towards the joint of the  $i^{th}$  link as measured in the inertial frame when the robot is in the home configuration.



The rate of each wheel will be denoted  $\dot{\theta}_1, \dot{\theta}_2$  respectively.

$$\begin{aligned} n_1 &= \frac{1}{\|r_1\|} r_1 \\ n_3 &= \frac{1}{\|r_3\|} r_3 \\ \omega_1 &= R_1^T R_2 (\omega_2 + n_1 \dot{\theta}_1) \\ \omega_3 &= R_3^T R_2 (\omega_2 + n_3 \dot{\theta}_2) \end{aligned}$$

Given these hinge constraints, solve for the angular momentum in terms of  $\omega_2, \dot{\theta}_1, \dot{\theta}_2$ .

$$AM = M_{22}\omega_2 + M_{11}R_1^T R_2 (\omega_2 + n_1 \dot{\theta}_1) + M_{33}R_3^T R_2 (\omega_2 + n_3 \dot{\theta}_2)$$

Notice that because of our choice of  $n_1$  and  $n_3$  and because each rotor is symmetric about its axis of rotation, that the following identities will hold.

$$\begin{aligned} R_1^T R_2 n_1 \dot{\theta}_1 &= n_1 \dot{\theta}_1 \\ R_3^T R_2 n_3 \dot{\theta}_2 &= n_3 \dot{\theta}_2 \\ M_{11}R_1^T R_2 &= M_{11} \\ M_{33}R_3^T R_2 &= M_{33} \end{aligned}$$

Now we can write the angular momentum without the dependencies on configuration that we had earlier.

$$AM = (M_{11} + M_{22} + M_{33})\omega_2 + M_{11}n_1 \dot{\theta}_1 + M_{33}n_3 \dot{\theta}_2$$

As before, define  $V := (M_{11} + M_{22} + M_{33})$  and assume it is invertible. Because each  $M_{ii} > 0$  this will be true. In this way, we may solve for  $\omega_2$  and obtain the kinematic map.

$$\omega_2 = V^{-1}M_{11}n_1 \dot{\theta}_1 + V^{-1}M_{33}n_3 \dot{\theta}_2$$

Again as before, if we consider the velocities of the rotors as our inputs to this system, we now have a kinematic model of the satellite. This is the model we will use to plan trajectories for the space craft. Define  $so(3)$  to be the Lie algebra of  $SO(3)$  identified with  $\mathbb{R}^3$ . It is useful to define  $b_1, b_2 \in \mathbb{R}^3$ , associated with  $so(3)$  in the normal way.

$$\begin{aligned} b_1 &= V^{-1}M_{11}n_1 \\ b_2 &= V^{-1}M_{33}n_3 \end{aligned}$$

The reconstruction of the actual state of the system is fairly straightforward with zero momentum. Given an initial configuration of the body,  $A_0 \in SO(3)$ ,

and the two inputs to the rotors  $\dot{\theta}_1(t), \dot{\theta}_2(t)$  we can solve for the body twist  $\omega_2(t)$  for any time.

The trajectory of the body 2's orientation will just be the solution of this differential equation.

$$\dot{R}_2 = R_2(\omega_2(t) \times)$$

In the special case of constant  $\omega_2(t)$ ; we can solve for the final orientation more directly.

$$\begin{aligned} \xi &\equiv \omega_2(t) \quad \forall t \in [0, 1] \\ R_2(1) &= R_2(0)e^{\xi \times} \end{aligned}$$

This just corresponds to a twist in the body frame about the axis  $\xi$  by one radian. This solution will suffice for most of the motions the path planner will generate.

### 5.1.2 Parallel Parking the Satellite

For systems without drift like ours, Chow's theorem assures controllability if the involutive closure of the input vector fields spans the tangent space at every point in the configuration space.

Finding the involutive closure involves taking the Lie Brackets of the input vector fields. For our case only one level of Brackets are needed to span the tangent space at every point. Taking the Lie Bracket in  $so(3)$  amounts to taking the cross product of the twist axis. Provided that they are not dependent, this will trivially complete the basis.

Assumption number two of the introduction assures controllability by demanding that the axis of rotation are not collinear.

Define  $b_0$  as follows.

$$b_0 = \frac{b_1 \times b_2}{\|b_1 \times b_2\|}$$

To form the shape space, identify any two configurations related to each other by a rotation through the axis given by  $b_0$ . Then shape space is  $SO(3)/S^1 = S^2$ . To visualize this space, one can think of Poincaré's representation of  $SO(3)$  as a circle bundle over the two sphere.

One may embed this shape space in  $\mathbb{R}^3$  by mapping  $R_2$  to the unit sphere with the projection map  $SO(3) \rightarrow S^2 : R_2 \rightarrow R_2 b_0$ . Observe that indeed if the system is rotated about the  $b_0$  axis then the projection map will map to the same point on the unit sphere in  $\mathbb{R}^3$ .

**Proposition 5** *There are no velocity constraints on the two sphere resulting from the projection of  $R_2$ , and thus the two sphere is a shape space.*

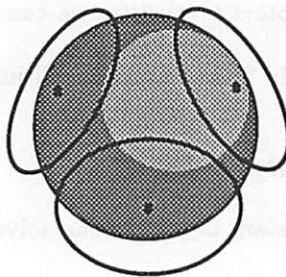


Figure 5.2: The Rotation Group as a Circle Bundle

**Proof:** By construction of this space, the rotors may provide a projected velocity which spans the tangent space to the unit sphere at any point.  $\square$

So given any path in this shape space, we wish to find geometric phase. For this case the geometric phase corresponds to finding the amount the system has rotated about the  $b_0$  axis.

**Proposition 6** *Provided the projected path is a positively oriented simple closed curve, the geometric phase for the satellite with two rotors is equal to the area, as induced from  $\mathbb{R}^3$  in the standard way, enclosed by the projected path on the unit sphere as embedded in  $\mathbb{R}^3$ .*

**Proof:** Consider the projected initial point of the path, and attach a reference frame which will span the tangent space of the unit sphere at that point. As we travel along this path, this frame will keep track of any spin about the  $b_0$  axis. But, by construction, the input vector field may induce spins about axes which are strictly perpendicular to the  $b_0$  axis and thus there can be no spin about the  $b_0$  axis. Therefore, the frame is parallel transported along this path. Assume the path is constructed out of finitely many smooth segments parameterized by arc length. Label each one of these segments  $C_i$  and the region they enclose  $R$ . An application of the Gauss-Bonnet Theorem [Car84] will then finish the proof, for it states:

$$2\pi\chi(R) = \sum \int_{C_i} k_g(s)ds + \int_R Kd\sigma + \sum \theta_i$$

Where  $\chi(R)$  is the Euler-Poincaré characteristic of the region,  $K$  being the curvature at each point,  $k_g$  being the geodesic curvature of curve  $C_i$ , and  $\theta_i$  being the exterior angle at each discontinuity in the path.

In this case,  $\chi(R)$  is equal to 1, so we may disregard the term on the left hand side since it is a multiple of  $2\pi$ . In addition, the curvature of the unit sphere is constant and equal to 1 therefore the surface integral will just give us area enclosed by the path. Finally the two summations give the net phase shift in the parallel transported frame.  $\square$

This result may be generalized to more complicated paths in the following manner. We will divide the problem into three increasingly complicated pieces.

**Case 1** The path is a simple closed curve. We will also assume it is positively oriented. If not, the sign of the geometric phase is reversed. The rotation around the  $b_0$  axis is then just the equal to the area enclosed by the path.

**Case 2** The path is a self-intersecting closed curve. In this case the path may be broken into simple closed curves, some positively oriented and others negatively oriented. Apply case 1 to each piece, and add the result together to obtain the amount rotated around the  $b_0$  axis.

**Case 3** The path is an open smooth curve. Then, join the end points by a geodesic. In this case, this would just be a great circle. This will form the second case. From the result gained by applying case 2. We are now done as the phase change due to the motion along the geodesic is zero.

With this reconstruction, one algorithm for steering the satellite becomes apparent. We could apply the same method, in principle, as was applied in the case of the planar skater. Given any starting configuration  $A_0 \in SO(3)$  and any final configuration  $A_1 \in SO(3)$ , a two step planner can always find a path. Observe that sinusoidal steering will work starting in any configuration because the phase will be the area enclosed and is not dependent on where it is located in the shape space. The procedure is then as follows.

**Step 1** Drive the two input fields with constant inputs as to drive the system in shape space coordinates to the destination configuration projected down in the shape space.

This is akin to lining the unicycle up in front of the desired parking space before the parallel park maneuver. In that case, the side ways direction of the car is the difficult direction to move in and in this step is ignored. This maneuver is also akin to step one of the planar skater algorithm.

**Step 2** Now that we have lined the system up in shape space, all that is left to do is move along the fiber attached at this point. We wish to enclose a sufficient amount of area of shape space to correct the error. What follows is one method. Keep a note of how far this error is. Now drive the system through the  $b_1$  vector field until  $R_2 b_0$  has moved through 90 degrees.

Now move in along the great circle perpendicular to the last motion. Move the system until we have rotated through the same amount as the error. The final leg will just be driving along the geodesic connecting us back to  $A_1 b_0$ .

So the problem of steering the satellite with two rotors is solved. In this system the motion in the difficult direction was executed with a first level bracket

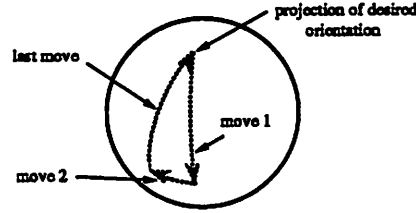


Figure 5.3: The Parallel Park Maneuver for the Satellite

motion. The original and more general system will have an almost identical shape space except changes it will achieve will involve second level brackets.

Observe that if we make two additional assumptions, the optimization with respect to control effort of the maneuver correcting the twist about the  $b_0$  axis becomes straight forward. If we assume that the rotors are identical and that they are mounted orthogonally, then the metric measuring distance on the two sphere induced from the ambient space is the same, up to a positive constant, as the control effort for the path. Thus the optimal path will enclose the most area on the unit sphere while having the minimum length. These are of course circles on the sphere.

## 5.2 The Shape Space for the General Case

Now we will return to the general three linked space robot. First we will define a few quantities. In keeping with the last section, define  $b_0$  as follows. It will play the same role as it did before. Recall that  $z$  is the vector pointing in the direction of the line connecting the two shoulders. We want it as measure in the second body's frame when the system is in the home configuration.

$$b_0 = \frac{z}{\|z\|}$$

Parameterize a new shape space,  $Sh_2$ , by  $R_2 b_0 \in \mathbb{R}^3$  as in the last section. Notice that  $Sh_2$  is just  $S^2$  embedded in  $\mathbb{R}^3$ . First we must establish when this space is indeed a shape space of some sort.

Consider the linked bodies in a fully extended configuration. In a fully extended configuration,  $b_0$  points to the center of mass of links one and three as measured in the second body's frame. Look at any plane which includes the  $b_0$  makes and notice the system projected down onto this plane matched exactly the assumptions required for the planar skater. So if one restricts the motion of the three dimensional system to this plane, the system will behave as the planar skater would.

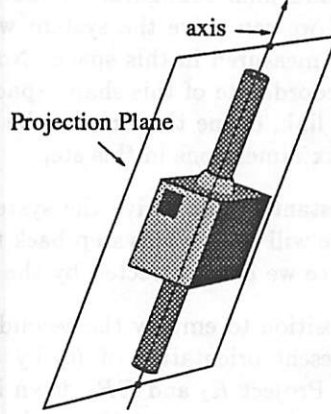


Figure 5.4: Reducing the Three Dimensional System to the Planar Case

Therefore, using the planar skater algorithm, we can obtain a net rotation of the entire system in this plane. Clearly, the algorithm will produce velocities in this plane only, and thus if we look at this plane in the inertial frame as in the figure, it will remain invariant under this motion.

Look at the projection of the resultant trajectory in  $Sh_2$ . The formula for this would be  $R_2(t)b_0 \in \mathbb{R}^3$ . Notice that this gives a path in the inertial frame coordinates. The entire trajectory will lie on the great circle created by the intersection of the projection plane with the shape space as embedded in  $\mathbb{R}^3$ .

Thus if we start with the system in the fully extended configuration, we can move the satellite to any point desired in  $Sh_2$  by traveling in a great circle to it. Moreover, the solution trajectory of the system will be confined to this circle.

In this way the space  $Sh_2$  may be considered a shape space. The projection plane may be rotated at will about the  $b_0$  axis and thus there are no velocity constraints.  $Sh_2$  is a second shape space because to move in it requires first level bracket motions. Recall that for the general system,  $Sh_1$  corresponds the space parameterized by  $R_{12}, R_{23}$  and is  $SO(3) \times SO(3)$ .

### 5.3 The Steering Algorithm

Now we will solve the problem of steering the three dimensional system. Given any desired configuration in  $A_1, A_2, A_3 \in SO(3)^3 = Q$  and any initial configuration in  $Q$ , the algorithm below generates inputs which will steer the system from the initial to the final configuration.

The algorithm is completely analogous to the planar skater's path planner.

- Step 1** Project the initial and final configurations down to  $Sh_1$ . This space is a shape space, thus we can drive the system with constant inputs as to correct the error as measured in this space. Now we can measure the error in  $R_2$ , the drift coordinate of this shape space. If  $R_2$  is the present attitude of the second link, define this error to be  $E = A_1 R_2^T$ . The error has been reduced by six dimensions in this step.
- Step 2** With piecewise constant inputs, drive the system to a fully extended configuration. Later we will retrace this step back to the projection of the desired orientation, once we have corrected by the error in  $R_2$ .
- Step 3** Now we are in a position to employ the second shape space  $Sh_2$ . We wish to adjust the present orientation of  $R_2$  by the error matrix  $E$  as measured in step one. Project  $R_2$  and  $ER_2$  down into  $Sh_2$ . We can move to any point in shape space, so correct the projected error by executing the planar style maneuvers. This will reduce the dimension of the error by two dimensions. If we retraced step two at this point, the configuration would be correct except for a rotation about the axis given by  $R_2 b_0$ .
- Step 4** We have already learned how to correct this type of error in this shape space when discussing the satellite with two rotors. Note the algorithm developed there used motions in great circles only, a motion we can execute in this shape space. So using three arcs, correct the rotational error about the  $b_0$  axis.
- Step 5** Retrace step two. The path is then complete.

All told, three piecewise constant motions in the joints and four planar skater motions are needed to achieve any point in configuration space.

## Chapter 6

# Conclusion

The algorithm presented here is based on correcting errors in levels. The non-holonomic nature of the system gives us lower dimension shape spaces. In the case of a holonomic system there are no velocity constraints and thus  $Sh_1 = Q$ . In general, each level of brackets required to span the entire configuration space will add another  $Sh$ .

For example, the car with trailers will have an onion-like structure if this procedure is carried to the limit. This organization of the configuration space together with the reconstruction procedure between each level or shell will immediately provide the path planner with a method of construction feasible paths between any two points in the configuration space as it did in the case of three rigid, linked bodies.

Three subjects are of future interest. One is finding methods of optimization of the paths generated with respect to control effort or some other method. Another is of course avoidance of obstacles and planning a path in an obstacle filled environment. Finally, designing controllers that would assure that the generated paths are actually followed present quite a challenge. These systems are impossible to stabilize by smooth state feed back. [Bro83] The paths themselves are relatively benign however they are very sensitive to deviations in initial conditions and from errors modeling. This is especially evident as seen in Murray [MS90] with the car with trailers.



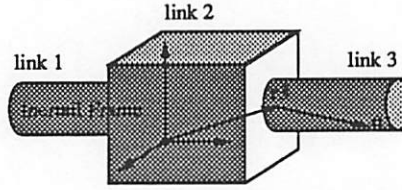


Figure 6.1: Robot in Home Configuration

## A Proof of Proposition 1

Recall that the configuration space is  $Q = SO(3)^3 \times \mathbb{R}^3$ . Each point in  $Q$  is given by three rotation matrices  $R_1, R_2, R_3 \in SO(3)$ ; and one vector  $c \in \mathbb{R}^3$ . In addition, define the relative rotation matrices to be  $R_{ij} = R_i^T R_j$ . The angular velocity of some body will be measured in that body's coordinate frame. Thus,  $\omega_i \in \mathbb{R}^3$  is the angular velocity of  $i^{th}$  body in the body's own frame.

**Proof:** The proof of this is a matter of algebra. In this derivation of the Lagrangian, we will at first parallel the work of Patrick [Pat90] wherever generalizations are clear. Thus, define  $\rho_i : \mathbb{R}^3 \rightarrow \mathbb{R}$  to be the density of the  $i^{th}$  link at a point  $q \in \mathbb{R}^3$  as measured when the system is in the home configuration from the  $i^{th}$  joint. For the middle joint, the position will be measured from the origin. The home configuration is any fixed reference position. Define  $r_i \in \mathbb{R}^3$  to be the distance of the  $i^{th}$  link joint from the center of mass of the second joint when the system is in home configuration.

So,  $c + R_i q + R_2 r_i$  is the position of a particle with label  $q \in \mathbb{R}^3$  on body  $i$  at some point in the configuration space. The derivative will give us the material velocity of the point. The Lagrangian is the kinetic energy with velocities measured in the inertial frame. Thus, the variable  $q \in \mathbb{R}^3$  will be the dummy variable of integration.

$$\begin{aligned}
 L = & \frac{1}{2} \int_{\mathbb{R}^3} \left| \dot{c} + \dot{R}_1 q + \dot{R}_2 r_1 \right|^2 \rho_1(q) dq + \\
 & \frac{1}{2} \int_{\mathbb{R}^3} \left| \dot{c} + \dot{R}_2 q \right|^2 \rho_2(q) dq + \\
 & \frac{1}{2} \int_{\mathbb{R}^3} \left| \dot{c} + \dot{R}_3 q + \dot{R}_2 r_3 \right|^2 \rho_3(q) dq
 \end{aligned}$$

The square terms represent the Euclidean norm squared. Using the usual dot product, expand these terms and factor out all terms independent of  $q$ . Following Patrick, notice the following identity:

$$q^T q = \text{trace}(qq^T)$$

$$\begin{aligned}
&= \text{trace} \begin{bmatrix} q_1^2 & q_1 q_2 & q_1 q_3 \\ q_1 q_2 & q_2^2 & q_2 q_3 \\ q_1 q_3 & q_2 q_3 & q_3^2 \end{bmatrix} \\
&= q_1^2 + q_2^2 + q_3^2
\end{aligned}$$

Then the Lagrangian is:

$$\begin{aligned}
L = & \frac{1}{2} \text{trace} \left( \dot{R}_1 \left( \int_{\mathbb{R}^3} q q^T \rho_1(q) dq \right) \dot{R}_1^T \right) + \dot{c}^T \dot{R}_1 \left( \int_{\mathbb{R}^3} q \rho_1(q) dq \right) + \frac{1}{2} \dot{c}^2 \int_{\mathbb{R}^3} \rho_1(q) dq + \\
& \frac{1}{2} \left( \left( \dot{R}_2 r_1 \right)^2 + 2 r_1^T \dot{R}_2^T \dot{c} \right) \int_{\mathbb{R}^3} \rho_1(q) dq + r_1^T \dot{R}_2^T \dot{R}_1 \int_{\mathbb{R}^3} q \rho_1(q) dq + \\
& \frac{1}{2} \text{trace} \left( \dot{R}_2 \left( \int_{\mathbb{R}^3} q q^T \rho_2(q) dq \right) \dot{R}_2^T \right) + \dot{c}^T \dot{R}_2 \left( \int_{\mathbb{R}^3} q \rho_2(q) dq \right) + \frac{1}{2} \dot{c}^2 \int_{\mathbb{R}^3} \rho_2(q) dq + \\
& \frac{1}{2} \text{trace} \left( \dot{R}_3 \left( \int_{\mathbb{R}^3} q q^T \rho_3(q) dq \right) \dot{R}_3^T \right) + \dot{c}^T \dot{R}_3 \left( \int_{\mathbb{R}^3} q \rho_3(q) dq \right) + \frac{1}{2} \dot{c}^2 \int_{\mathbb{R}^3} \rho_3(q) dq + \\
& \frac{1}{2} \left( \left( \dot{R}_2 r_3 \right)^2 + 2 r_3^T \dot{R}_2^T \dot{c} \right) \int_{\mathbb{R}^3} \rho_3(q) dq + r_3^T \dot{R}_2^T \dot{R}_3 \int_{\mathbb{R}^3} q \rho_3(q) dq
\end{aligned}$$

For each link, label the mass  $m_i$  and call the vector to the center of mass from the joint as measured in the home configuration  $d_i \in \mathbb{R}^3$ . For the middle link, this will be measured from the origin of the inertial coordinate system. Given that we can place the inertial frame anywhere, we will place it at the center of mass of the middle body when the system is in the home configuration. Therefore  $d_2$  will be zero. With these definitions, the following identities hold.

$$\begin{aligned}
m_i &= \int_{\mathbb{R}^3} \rho_i(q) dq \\
m_i d_i &= \int_{\mathbb{R}^3} q \rho_i(q) dq \\
I_i &= \int_{\mathbb{R}^3} q q^T \rho_i(q) dq - m_i d_i d_i^T
\end{aligned}$$

Using these identities we may replace the integrals with these constants. Denote the total mass by  $m$ .

$$\begin{aligned}
L = & \frac{1}{2} \text{trace} \left( \dot{R}_1 I_1 \dot{R}_1^T + \dot{R}_2 I_2 \dot{R}_2^T + \dot{R}_3 I_3 \dot{R}_3^T \right) + \\
& \frac{m_1}{2} \left( \left( \dot{R}_1 d_1 \right)^2 + \left( \dot{R}_2 r_1 \right)^2 \right) + \frac{m_3}{2} \left( \left( \dot{R}_3 d_3 \right)^2 + \left( \dot{R}_2 r_3 \right)^2 \right) + \\
& m_1 r_1^T \dot{R}_2^T \dot{R}_1 d_1 + m_3 r_3^T \dot{R}_2^T \dot{R}_3 d_3 +
\end{aligned}$$

$$\left( m_1 (\dot{R}_1 d_1 + \dot{R}_2 r_1) + m_2 \dot{R}_2 d_2 + m_3 (\dot{R}_3 d_3 + \dot{R}_2 r_3) \right)^T \dot{c} + \frac{m}{2} \dot{c}^2$$

As in Patrick [Pat90], one can add any vector to  $c$  and not change the value of the Lagrangian. Thus  $L(c, R_1, R_2, R_3, \dot{c}, \dot{R}_1, \dot{R}_2, \dot{R}_3) = L(c + \delta, R_1, R_2, R_3, \dot{c}, \dot{R}_1, \dot{R}_2, \dot{R}_3)$ . This defines an action  $l_\delta : Q \rightarrow Q$ , taking  $(c, R_1, R_2, R_3) \rightarrow (c + \delta, R_1, R_2, R_3)$  leaving  $L$  invariant. This tells us we can find a reduced coordinate system, the coordinates of which will characterize the conserved quantity. Assume that total linear momentum is zero. System center of mass will be denoted by  $CM$ .

$$\begin{aligned} CM &= \frac{m_1}{m} (R_1 d_1 + R_2 r_1 + c) + \frac{m_2}{m} (R_2 d_2 + c) + \frac{m_3}{m} (R_3 d_3 + R_2 r_3 + c) \\ 0 &= \frac{d}{dt}(CM) \\ 0 &= \frac{m_1}{m} (\dot{R}_1 d_1 + \dot{R}_2 r_1 + \dot{c}) + \frac{m_2}{m} (\dot{R}_2 d_2 + \dot{c}) + \frac{m_3}{m} (\dot{R}_3 d_3 + \dot{R}_2 r_3 + \dot{c}) \\ \dot{c} &= -\frac{1}{m} \left( m_1 (\dot{R}_1 d_1 + \dot{R}_2 r_1) + m_2 \dot{R}_2 d_2 + m_3 (\dot{R}_3 d_3 + \dot{R}_2 r_3) \right) \end{aligned}$$

So this gives a formula for  $\dot{c}$  in terms of the  $\dot{R}_i$ 's. Substitute this into the Lagrangian and define the reduced masses to be  $\epsilon_{ij} = \frac{m_i m_j}{m}$ . Continuing,

$$\begin{aligned} L &= \text{trace} \left( \dot{R}_1 I_1 \dot{R}_1^T + \dot{R}_2 I_2 \dot{R}_2^T + \dot{R}_3 I_3 \dot{R}_3^T \right) + \frac{m_1}{2} (\dot{R}_1 d_1 + \dot{R}_2 r_1)^2 + \frac{m_3}{2} (\dot{R}_3 d_3 + \dot{R}_2 r_3)^2 - \frac{1}{m} \left( m_1 (\dot{R}_1 d_1 + \dot{R}_2 r_1) + m_3 (\dot{R}_3 d_3 + \dot{R}_2 r_3) \right)^2 + \frac{m}{2} \left( \frac{m_1}{m} (\dot{R}_1 d_1 + \dot{R}_2 r_1) + \frac{m_3}{m} (\dot{R}_3 d_3 + \dot{R}_2 r_3) \right)^2 \\ L &= \frac{1}{2} \text{trace}(\dot{R}_1 I_1 \dot{R}_1^T + \dot{R}_2 I_2 \dot{R}_2^T + \dot{R}_3 I_3 \dot{R}_3^T) + \frac{\epsilon_{12}}{2} (\dot{R}_1 d_1 + \dot{R}_2 r_1)^2 + \frac{\epsilon_{13}}{2} (\dot{R}_1 d_1 + \dot{R}_2 r_1 - (\dot{R}_3 d_3 + \dot{R}_2 r_3))^2 + \frac{\epsilon_{23}}{2} (\dot{R}_3 d_3 + \dot{R}_2 r_3)^2 \end{aligned}$$

Now we will substitute in for  $\dot{R}_i$  in the following standard way. The only difference will be having the angular velocity vector in body coordinates. I choose

this representation because later this will make more sense from a control point of view. So as a reminder, if  $R_i$  is a rotation matrix,  $\dot{R}_i$  will be denoted as  $R_i(\omega_i \times)$  where  $(\omega_i \times)$  is the skew symmetric matrix representing the cross product of the angular velocity vector  $\omega_i \in \mathbb{R}^3$  as measured from the  $i^{\text{th}}$  coordinate frame. The only other trick involved is noting that a similarity transform does not change the eigenvalues of a matrix, which implies it does not change the trace [Str88]. The following identity is useful:

$$\begin{aligned}
 \frac{1}{2} \text{trace} \left( \dot{R}_1 I_1 \dot{R}_1^T \right) &= \frac{1}{2} \text{trace} \left( R_1^T \dot{R}_1 I_1 \dot{R}_1^T R_1 \right) \\
 &= \frac{1}{2} \text{trace} \left( (\omega_1 \times) I_1 (\omega_1 \times)^T \right) \\
 &= \frac{1}{2} \text{trace} \left( \omega_1^T \omega_1 I_1 - \omega_1 \omega_1^T I_1 \right) \\
 &= \frac{1}{2} \omega_1^T (\text{trace}(I_1) I - I_1) \omega_1 \\
 &:= \frac{1}{2} \omega_1^T J_1 \omega_1 \\
 J_1 &:= \text{trace}(I_1) I - I_1
 \end{aligned}$$

The other terms will follow in a similar manner giving us the  $3 \times 3$  matrices  $J_2, J_3$ . Another simple calculation shows how to rewrite the other terms so that  $\omega_i$ 's may be factored out easily.

$$\begin{aligned}
 \epsilon_{12} \left( R_1 R_1^T \dot{R}_1 d_1 + R_2 R_2^T \dot{R}_2 r_1 \right)^2 &= \\
 &= \epsilon_{12} (R_1 (\omega_1 \times) d_1 + R_2 (\omega_2 \times) r_1)^2 \\
 &= \epsilon_{12} \left( \omega_1^T (d_1 \times)^T (d_1 \times) \omega_1 + \omega_2^T (r_1 \times)^T (r_1 \times) \omega_2 \right) + \\
 &\quad \epsilon_{12} \left( 2 \omega_1^T (d_1 \times)^T R_1^T R_2 (r_1 \times) \omega_2 \right)
 \end{aligned}$$

The cross term may be written as sum of two terms, one the transpose of the other. This is done to get a symmetric matrix when the  $\omega$ 's are factored out. In doing so it is convenient to write the orientation matrixes in terms of their relative orientations. Recall that we defined  $R_{ij} = R_i^T R_j$ . These matrices will later have the interpretation of coordinates of the shape space, the part of the configuration space in which we may move the robot without any velocity constraints. Further, define  $z \in \mathbb{R}^3$  to simplify some notation. This quantity physically is the direction of the line connecting the two shoulders and as will be shown later, it is also the most difficult axis to rotate the system about.

$$z = r_1 - r_3 \quad (6.1)$$

$$L = \frac{1}{2}(\omega_1^T, \omega_2^T, \omega_3^T) J(R_{12}, R_{13}, R_{23}) \begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{bmatrix}$$

The matrix  $J$  is as follows.

$$\begin{bmatrix} \overset{J_1+}{(\epsilon_{12} + \epsilon_{13})(d_1 \times)^T(d_1 \times)} & (d_1 \times)^T R_{12} \begin{pmatrix} \epsilon_{12}(r_1 \times)^+ \\ \epsilon_{13}(z \times) \end{pmatrix} & -\epsilon_{13}(d_1 \times)^T R_{13}(d_3 \times) \\ \left( (d_1 \times)^T R_{12} \begin{pmatrix} \epsilon_{12}(r_1 \times)^+ \\ \epsilon_{13}(z \times) \end{pmatrix} \right)^T & \overset{J_2+}{\epsilon_{12}(r_1 \times)^T(r_1 \times) + \epsilon_{23}(r_3 \times)^T(r_3 \times) + \epsilon_{13}(z \times)^T(z \times)} & (d_3 \times)^T R_{23} \begin{pmatrix} \epsilon_{23}(r_3 \times)^- \\ \epsilon_{13}(z \times) \end{pmatrix} \\ \left( -\epsilon_{13}(d_1 \times)^T R_{13}(d_3 \times) \right)^T & \left( (d_3 \times)^T R_{23} \begin{pmatrix} \epsilon_{23}(r_3 \times)^- \\ \epsilon_{13}(z \times) \end{pmatrix} \right)^T & \overset{J_3+}{(\epsilon_{23} + \epsilon_{13})(d_3 \times)^T(d_3 \times)} \end{bmatrix}$$

Because every term in the matrix  $J$  is either a constant or a linear function of one of the  $R_{ij}$ 's we can compress some of this notation by defining the constant  $3 \times 3$  matrices  $M_{ij}, K_{ij}$  so that the following holds:

$$J(R_{12}, R_{13}, R_{23}) =: \begin{bmatrix} M_{11} & M_{12}R_{12}K_{12} & M_{13}R_{13}K_{13} \\ K_{12}^T R_{12}^T M_{12}^T & M_{22} & M_{23}R_{23}K_{23} \\ K_{13}^T R_{13}^T M_{13}^T & K_{23}^T R_{23}^T M_{23}^T & M_{33} \end{bmatrix}$$

This completes the proof of the proposition.  $\square$

## B Example Satellite 1

This first satellite example will emulate an anthropomorphic tele-operated robot similar to some proposed versions of NASA's Flight Telerobotic Servicer. To simplify the model each of the arms will have three degree of freedom shoulders and no elbows. It is probably not the most useful robot but it will serve to illustrate this procedure. This appendix will contain the calculations of the  $M_{ij}, K_{ij}$  matrixes.

This could also serve as a rough model for an astronaut who is holding his elbows rigid. The robot here will be light by FTS standards, a mere 180 kilograms. In order to exaggerate the reorientation, the arms contain a significant amount of mass relative to the body. See figure (12) for a picture of the robot.

Each arm will be one meter long and weigh 40 kilograms, with an approximate diameter of 25 centimeters. The main body will be modeled as a 100

kilogram cylinder, one meter in diameter and height. The following quantities are, with all units meters:

$$\begin{aligned} r_1 &= d_1 &= \begin{pmatrix} -\frac{1}{2} & 0 & 0 \end{pmatrix} \\ r_3 &= d_3 &= \begin{pmatrix} \frac{1}{2} & 0 & 0 \end{pmatrix} \\ z &= r_1 - r_3 &= \begin{pmatrix} -1 & 0 & 0 \end{pmatrix} \end{aligned}$$

The inertial matrixes can be computed assuming homogeneous density of mass in the robot. Careful choice of frames and of the home configuration makes the computation simpler in this case. All units are  $kgm^2$ .

$$\begin{aligned} J_{11} &= J_{33} = \begin{bmatrix} 0.32 & 0 & 0 \\ 0 & 13.5 & 0 \\ 0 & 0 & 13.5 \end{bmatrix} \\ J_{22} &= \begin{bmatrix} 14.5 & 0 & 0 \\ 0 & 14.5 & 0 \\ 0 & 0 & 12.5 \end{bmatrix} \end{aligned}$$

The reduced masses are, in kilograms,  $\epsilon_{12} = \epsilon_{23} = 22.2$  and  $\epsilon_{13} = 8.88$ . The computation then yields:

$$\begin{aligned} M_{11} &= M_{33} = \begin{bmatrix} 0.32 & 0 & 0 \\ 0 & 21.3 & 0 \\ 0 & 0 & 21.3 \end{bmatrix} \\ M_{12} &= -M_{23} = -M_{13} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -0.5 \\ 0 & 0.5 & 0 \end{bmatrix} \end{aligned}$$

$$\begin{aligned} M_{22} &= \begin{bmatrix} 14.5 & 0 & 0 \\ 0 & 34.4 & 0 \\ 0 & 0 & 32.4 \end{bmatrix} \\ K_{12} &= \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 19.9 \\ 0 & -19.9 & 0 \end{bmatrix} \\ K_{13} &= \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -4.4 \\ 0 & 4.4 & 0 \end{bmatrix} \\ K_{23} &= \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -2.3 \\ 0 & 2.3 & 0 \end{bmatrix} \end{aligned}$$

## C Proof of Proposition 3

We will use the general equations of the first proposition and just look at their restriction. First, in this context the equations for the Lagrangian simplify.

Notice that as  $J(R_{12}, R_{13}, R_{23})$  is symmetric therefore so is the reduced matrix we will find. Call this reduced matrix  $M$ .

$$M = \begin{bmatrix} J_{33} & J_{36} & J_{39} \\ J_{36} & J_{66} & J_{69} \\ J_{39} & J_{69} & J_{99} \end{bmatrix}$$

$$L = [\dot{\theta}_1^T, \dot{\theta}_2^T, \dot{\theta}_3^T] M \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \\ \dot{\theta}_3 \end{bmatrix}$$

Now we examine the elements of this matrix  $M$ . In doing so we employ the assumption that in the home configuration all of the centers of mass fall along the  $x$  axis.

$$\begin{aligned} J_{33} &= I_1^{11} + I_1^{22} + (\epsilon_{12} + \epsilon_{13})(d_1)^2 \\ J_{66} &= I_2^{11} + I_2^{22} + \epsilon_{12}(y)^2 + \epsilon_{23}(x)^2 + \epsilon_{13}(z)^2 \\ J_{99} &= I_3^{11} + I_3^{22} + (\epsilon_{23} + \epsilon_{13})(d_3)^2 \\ J_{36} &= d_1^1(\epsilon_{12}y^1 + \epsilon_{13}z^1) \cos(\theta_2 - \theta_1) \\ J_{39} &= \epsilon_{13}d_1^1d_3^1 \cos(\theta_3 - \theta_1) \\ J_{69} &= d_3^1(\epsilon_{23}x^1 - \epsilon_{13}z^1) \cos(\theta_2 - \theta_3) \end{aligned}$$

The system of equations looks like those of Li and Gurvits [LG90]. Also, one may look at [SOKM88]. Compute the momentum map. The calculation of the momentum map is the same as before but restricted. The group which acts on the configuration space is  $S^1$ . The action is defined also in a similar manner, it amounts to just physically rotating the entire system. Define the angular momentum to be  $AM \in \mathbb{R}$ , which is the Lie Algebra of  $S^1$ .

$$AM = [1 \ 1 \ 1] M \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \\ \dot{\theta}_3 \end{bmatrix}$$

In a similar manner, define  $v, a_i$  as the scalar counterparts to  $V, A_i$ . Then we get  $b_1, b_2$  and can perform the same transform as in the three dimensional case. The equations then transform as follows.

$$\begin{bmatrix} \dot{\psi}_1 \\ \dot{\psi}_2 \\ \dot{\psi}_3 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ -va_1 \end{bmatrix} u_1 + \begin{bmatrix} 0 \\ 1 \\ -va_3 \end{bmatrix} u_2$$

Again note that the inputs correspond to the velocities of the the joints in the shape space. The proposition is now proved.  $\square$

## D Example Satellite 2

This section contains the calculations for the symmetric planar skater. See figure (2) for a picture of this robot. Each link will be modeled as a one meter long square tube 10 kilograms in mass. The reduce masses will be just 3.3 kilograms each. The  $r_1, r_3$  and the  $d_1, d_3$  will be the same as in the first example satellite.

The calculations are standard and so the results are:

$$\begin{aligned}k_{11} &= k_{33} = 6.92kgm^2 \\k_{12} &= k_{23} = -5kgm^2\end{aligned}$$

$$\begin{aligned}k_{13} &= 3.31kgm^2 \\k_{22} &= 5.25kgm^2\end{aligned}$$

Simulations were implemented on the SPARC stations and the data ported over to the IRIS machines for viewing.



## E Computation of the Lie Bracket

The proof is a matter of computation. Notice first that given the structure of  $v_1, v_2$  that the terms that will effect  $\dot{\psi}_1, \dot{\psi}_2$  will be zero. Also as the input vector fields are not functions of  $\psi_3$ , the formula for the bracket will simplify greatly.

$$\begin{aligned}
 [v_1, v_2] &= (Dv_2)v_1 - (Dv_1)v_2 \\
 &= \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ \frac{\partial(-va_3)}{\partial\psi_1} & \frac{\partial(-va_3)}{\partial\psi_2} & 0 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ -va_1 \end{bmatrix} - \\
 &\quad \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ \frac{\partial(-va_1)}{\partial\psi_1} & \frac{\partial(-va_1)}{\partial\psi_2} & 0 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ -va_3 \end{bmatrix} \\
 &= \begin{bmatrix} 0 \\ 0 \\ \frac{\partial(va_1)}{\partial\psi_2} - \frac{\partial(va_3)}{\partial\psi_1} \end{bmatrix}
 \end{aligned}$$

Now define  $\xi$  to be the function given in the last entry. In showing the rest of the proposition, the following identities prove useful.

$$\begin{aligned}
 \frac{\partial a_1}{\partial \psi_2} &= k_{13} \sin(\psi_1 - \psi_2) \\
 \frac{\partial a_3}{\partial \psi_1} &= -k_{13} \sin(\psi_1 - \psi_2) \\
 \frac{\partial v}{\partial \psi_1} &= 2(k_{12} \sin(\psi_1) + k_{13} \sin(\psi_1 - \psi_2)) v^2 \\
 \frac{\partial v}{\partial \psi_2} &= -2(k_{23} \sin(-\psi_2) + k_{13} \sin(\psi_1 - \psi_2)) v^2
 \end{aligned}$$

Substitution is the next step. The equation will then take on the proper form, after all is added together.

$$\begin{aligned}
 \xi(\psi_1, \psi_2) &= -2 \frac{\partial v}{\partial \psi_2} a_1 + \frac{\partial a_1}{\partial \psi_2} v - \frac{\partial v}{\partial \psi_1} a_3 - \frac{\partial a_3}{\partial \psi_1} v \\
 &= -2v^2 a_3 (k_{12} \sin(\psi_1) + k_{13} \sin(\psi_1 - \psi_2)) \\
 &\quad + 2v^2 a_1 (k_{23} \sin(\psi_2) - k_{13} \sin(\psi_1 - \psi_2)) \\
 &\quad + 2v^2 ((a_1 + a_2 + a_3) k_{13} \sin(\psi_1 - \psi_2)) \\
 &= 2v^2 (a_1 k_{23} \sin(\psi_2) - a_3 k_{12} \sin(\psi_1) + a_2 k_{13} \sin(\psi_1 - \psi_2))
 \end{aligned}$$

The proposition is now proven. This result is similar, modulo some signs, to the result given in Li and Gurvits [LG90].  $\square$ .

# Bibliography

- [Bro81] R. W. Brockett. *Control Theory and singular Riemannian geometry*. Springer-Verlag, New York, 1981.
- [Bro83] R. W. Brockett. Asymptotic stability and feedback stabilization. In R.W. Brockett, R.S. Millman, and H.J. Sussman, editors, *Differential Geometric Control Theory*, pages 181–191. Birkhauser, 1983.
- [Byr90] R. H. Byrne. Interactive graphics and dynamical simulation in a distributed processing environment. Master's thesis, University of Maryland at College Park, 1990. Advisor: P. S. Krishnaprasad.
- [Car84] Manfredo P. Do Carmo. *Differential Geometry of Curves and Surfaces*. Prentice-Hall, Inc., Englewood Cliffs, New Jersey, second edition, 1984.
- [Fro79] Cliff Frohlich. Do springboard divers violate angular momentum conservation? *Am. J. Phys.*, 47(7):583 – 592, 1979.
- [Isi89] A. Isidori. *Nonlinear Control Systems*. Springer-Verlag, second edition, 1989.
- [KHY72] T. R. Kane, M. R. Headrick, and J. D. Yatteua. Experimental investigation of an astronaut maneuvering scheme. *J. Biomechanics*, 5:313 – 320, 1972.
- [Kri91] P.S. Krishnaprasad. Geometric phases, and optimal reconfiguration for multibody systems. Systems Research Center at the University of Maryland, College Park, 1991.
- [LG90] Z. Li and L. Gurvits. Theory and application of nonholonomic motion planning. Technical report, Courant Institute of Mathematical Sciences, July 1990.
- [LS90] G. Lafferriere and H. J. Sussmann. Motion planning for controllable systems without drift: a preliminary report. Technical report, New York University and Rutgers University, June 1990.

- [LS91a] G. Lafferriere and H. J. Sussmann. Motion planning for controllable systems without drift. In *IEEE International Conference on Robotics and Automation*, pages 1148 – 1153, 1991.
- [LS91b] J-P. Laumond and T. Siméon. Motion planning for a two degrees of freedom mobile robot with towing. In *IEEE International Conference on Control and Applications*, 1991.
- [Mon90] R. Montgomery. Isoholonomic problems and some applications. *Communications in Mathematical Physics*, 128:565 – 592, 1990.
- [MR90] J. Marsden and T. Ratiu. *Mechanics and Symmetry*. Pre-Print, 1990.
- [MS90] R. Murray and S. Sastry. Steering nonholonomic systems using sinusoids. In *IEEE Conference on Decision and Control*, pages 2097 – 2101, 1990.
- [NM89] Y. Nakamura and R. Mukherjee. Nonholonomic path planning of space robots via bi-directional approach. In *IEEE International Conference on Robotics and Automation*, pages 1764 – 1769, 1989.
- [NvdS90] H. Nijmeijer and A. J. van der Schaft. *Nonlinear Dynamical Control Systems*. Springer-Verlag, 1990.
- [Pat90] G. W. Patrick. *Two Axially Symmetric Coupled Rigid Bodies: Relative Equilibria, Stability, Bifurcations, and a Momentum Preseving Symplectic Integrator*. PhD thesis, University of California at Berkeley, 1990.
- [PD90] Evangelos Papadopoulos and Steven Dubowsky. Steering nonholonomic systems using sinusoids. In *IEEE International Conference on Robotics and Automation*, pages 1102 – 1108, 1990.
- [SOKM88] N. Sreenath, Y. G. Oh, P. S. Krishnaprasad, and J. E. Marsden. The dynamics of coupled planar rigid bodies. *Dynamics and Stability of Systems*, 3:25 – 49, 1988.
- [Spi70] Michael Spivak. *Differential Geometry*. Publish or Pershish, Inc., Houston, Texas, second edition, 1970.
- [Str88] G. Strang. *Linear Algebra and its Applications*. Harcourt Brace Jovanovich, San Diego, second edition, 1988.
- [War83] F. W. Warner. *Foundations of Differentialble Manifolds and Lie Groups*. Springer-Verlag, 1983.