THESIS REPORT
Master's Degree

S Y S T E M S
R E S E A R C H
C E N T E R

Supported by the National Science Foundation Engineering Research Center Program (NSFD CD 8803012), the University of Maryland, Harvard University, and Industry

Coordination of the Walking Stick Insect Using a System of Nonlinear Coupled Oscillators

by D.J. Marvin
Advisors: P.S. Krishnaprasad

M.S. 92-9

ABSTRACT

Title of Thesis: COORDINATION OF THE WALKING STICK INSECT USING A SYSTEM OF NONLINEAR COUPLED OSCILLATORS

Name of degree candidate: Daryl J. Marvin

Degree and Year: Master of Science, 1992

Thesis directed by: Professor P.S. Krishnaprasad
Department of Electrical Engineering

The area of walking machines is investigated. A design for a central pattern generator composed of nonlinear coupled oscillators which generates the characteristic gaits of the walking stick insect is presented. A full dynamic model of the walking stick insect is developed, including ground interaction. Furthermore, a method of numerically solving the dynamic equations is described. Next, a control system utilizing the system of nonlinear coupled oscillators is presented. Finally, a numerical/graphical simulation of the walking stick insect is described, and results presented.

COORDINATION OF THE WALKING STICK INSECT USING A SYSTEM OF NONLINEAR COUPLED OSCILLATORS

by

Daryl J. Marvin

Thesis submitted to the Faculty of the Graduate School
of The University of Maryland in partial fulfillment
of the requirements for the degree of
Master of Science
1992

Advisory Committee:

Professor P.S. Krishnaprasad, Chairman/Advisor
Assistant Professor W. Dayawansa
Assistant Professor M. Austin

ACKNOWLEDGEMENTS

I would like to begin by thanking my advisor Dr. P.S. Krishnaprasad for his help and guidance. In additions, thanks must go to Dr. Tsai, who got me invloved in legged locomotion through the walking machine project at the University of Maryland.

I would also like to thank Dr. Avis Cohen for taking the time for discussions and for obtaining a good deal of th source material used in this research. This interaction was very beneficial in the initial stages of this work.

Finally, and most importantly, I would like to thank my wife, Shu-er, with putting up with me while I finished up my thesis.

This work was supported in part by the National Science Foundation's Engineering Engineering Research Center Program NSFD CDR 8803012, and by the Air Force Office of Scientific Research Univeristy Research Initiative Program under grant AFOSR-90-0105. Support was also received through a Graduate School Fellowship and a NASA/USRA Teaching Assistantship.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

In the past, there has been much effort put into the study of man-made walking machines. Many different attempts have been made to build machines which walk, with varying results. In all cases, however, even one of the simplest of animals is a far superior walker than any machine yet built. With this in mind, we believe that an attempt should be made to study how nature has solved the problem of legged locomotion so that we can use similar techniques to build a better walking machine. Luckily for engineers, there has already been a great deal of study done in the biology/zoology field of the way that animals walk. While in the past there was very little interaction between engineers and biologists/zoologists, there appears to be an increasing interaction between these two groups.

This work consists of studying how certain animals may coordinate and control their movement in order to emulate the functionality of these biological solutions and apply them in an engineering setting. In addition, to facilitate further work in this area, a testbed consisting of a numerical dynamic simulation of a walking stick insect was created in which new and/or different schemes for control and coordination could be tested. But first, we take a look at why the study of legged locomotion is important, and briefly review some previous work in this area.

1.1.1 Importance of Legged Locomotion

Currently, practically all mechanical locomotion on land is done using wheeled or tracked machines. Since for most purposes these machines work fairly well, the question arises as to why we should study legged locomotion at all. One of the main reasons is that legged vehicles have greater mobility. While wheeled and tracked vehicles work well on prepared surfaces, they are not as good on rough terrain. Legged vehicles can use isolated footholds to cross uneven territory, step over obstacles, and climb up and down stairs. In addition, legged vehicles can achieve a smooth ride over rough ground by adjusting the effective length of the legs to match the variations in the terrain.

Mainly as a result of this adaptability to varied and rough terrain, walking machines have been proposed to be used for a variety of purposes. One of the

most commonly mentioned is the use of walking machines for use in planetary exploration. For example, on the rough Martian surface, a walking rover will have a much easier time than a wheeled rover. Another common use cited for walking machines is for use in potentially hazardous situations such as in nuclear reactor containment vessels where their increased mobility in cluttered environments would give them advantages over wheeled robots.

In addition to these potential uses for legged vehicles, study of legged locomotion also gives insight into human and animal control structures. Since animals, including humans, are examples of very useful, robust systems, a better understanding of how they work is quite beneficial. Such study can also guide biological research by suggesting different models for experimental testing.

1.1.2 Work done on Walking Machines

Much research has been done on walking machines, with many different machines having been built. Some of the more interesting are listed below [17][22]:

- General Electric walking truck, mid 1960's. The first design that allowed for freedom in the foot path. This vehicle used a human operator to provide the control of the hydraulically actuated legs. While the machine worked quite well, it demanded too much of the human operator to be useful over long periods of time.

- Ohio State University hexapod, mid 1970's. The first working use of a digital computer for purposes of controlling the walking. Used a central control strategy to control the six legs driven by electric drill motors. Used extensive planning and modeling requiring detailed knowledge of the terrain and the kinematics of the machine.

- Sutherland's hexapod, 1983. The first man-carrying, computer-controlled walking machine. Also very interesting since the control strategy was based to some extent on animal locomotion. The design used a distributed control in which retracting legs would inhibit nearby legs from lifting [9].

- Raibert's hopper, 1984. One-legged three dimensional hopper with pneumatic actuators. Two-legged and four-legged machines have also been built. Notable for its use of dynamic balancing.

Many other walking machines were also built, including a four legged machine by Hirose in Japan in 1980, a six legged machine by Kessis in France, and a six legged machine by Odetics Inc. in 1985.

1.1.3 Biological Studies

There has also been a great deal of relevant work in the biological/zoological literature. There are two main areas which have a direct bearing on research in walking machines: the study of animal mechanics, and the study of neurological

control.

While the mechanics of many different animals have been studied [2], two of the most commonly studied are the cat and the walking stick insect. The walking stick insect (Carausis morosus) has several advantages that make it conducive to study: slow insect movement, orderly gait pattern, and only three joints per leg [16]. As a result, there have been some models made of the leg coordination of the walking stick insect [7][8]. Due to the relatively simple nature of the walking stick insect, both in terms of mechanics and control, it is this animal which gets a great deal of attention in this thesis. Once the complexity of an insect's statically stable gait is mastered, then the more complex dynamically stable gaits of animals such as the cat should be attempted.

Most of the research on the neurological basis of motion control involved studying the signals of the spinal cord. There have been numerous studies of the so called spinal cat, in which the spinal cord of the cat is exposed [10]. There has also been a great deal of study of the spinal cord of the lamprey by Cohen et. al. [5][1][3][4][18]. In these experiments, the spinal cord has been removed from the animal, stimulated in a dish, and the resulting signals have been studied. This work is particularly notable for our purposes because of the effort made to mathematically model the functionality of the spinal cord [13][14].

While most of this work has been geared toward attempting to determine and model the actual biological basis for the observed phenomena, it can also

be used as a guide for the design of a man-made walking machine.

1.2 Terminology

Throughout this thesis, several terms will be used which, although they are common in biological literature, are not commonly used in engineering. A partial list of these terms used in this paper follows:

ipsilateral Along the length of the body.

contralateral Across the width of the body.

rostral Towards the head, or front of the body.

caudal Towards the tail, or rear of the body.

1.3 Overview

In Chapter 2, the concept of nonlinear coupled oscillators is investigated. In particular the objective of using nonlinear coupled oscillators for leg coordination is discussed, as well as the methods used to accomplish this objective. We give descriptions of the oscillators and methods of coupling between oscillators.

In Chapter 3, a model of the walking stick insect is presented. This model

includes the full kinematic and dynamic description of the insect, as well as modeling the ground interaction. This model will be used to test the coordination and control strategies developed in this thesis.

In Chapter 4, the full control strategy for the walking stick insect is presented, as well as a description of the simulation used, and the results obtained.

Finally, in Chapter 5, we conclude with several recommendations for future work in such areas as the dynamic modeling and control strategies.

Chapter 2

Nonlinear Coupled Oscillators

2.1 Introduction

In biological systems, movement is coordinated at least partially by a central pattern generator. In the work done by Cohen et. al. on the spinal cord of the *lamprey*, this central pattern generator was modeled as a system of nonlinear coupled oscillators [5]. This chapter discusses nonlinear coupled oscillators and their usefulness in the coordination of legged locomotion. First, a description is given of different kinds of gaits desired to be generated by a system of coupled oscillators. Then a description of an individual oscillator is given, followed by a description of the method of coupling between the oscillators.

2.2 Gaits

This section describes the type of gaits that are used by the walking stick insect, and the characteristics that are required of the system of nonlinear coupled oscillators in order to generate the desired gaits.

There are two main different styles of gaits: dynamically stable and statically stable. Statically stable gaits are characterized by the feature that if at any time all of the joints are locked, the machine or animal will not fall over. Dynamically stable gaits do not have this property; the system generally has to keep moving in order to not fall down. Statically stable gaits are typically used by simpler animals like insects, while dynamically stable gaits are typically used by more complex animals like mammals and some reptiles. Since statically stable gaits are much easier to produce than dynamically stable gaits, it is these types of gaits which are investigated in this thesis.

Statically stable gaits have the property that at all times the center of gravity is contained in the area formed by connecting all of the legs on the ground. In this situation, there always exists a set of ground reaction forces that can counteract gravity, and therefore keep the system from collapsing (i.e. it is statically stable). In order to maintain this property, there must be a minimum of three legs on the ground at all times. While it is possible to create a statically stable gait with as few as four legs, such gaits are quite slow and inflexible in

terms of foot placement. As a result, most animals that move with statically stable gaits have at least six legs. Such is the case with most insects, and in particular the walking stick insect, whose gaits will be looked at in more detail.

For our purposes, the most important features of the gaits of a walking stick insect are the following [7]:

- The swing duration is independent of walking speed. Assuming this swing is done in the minimum reasonable time, this enables the insect to keep as many feet on the ground as possible for the desired walking speed.

- Adjacent ipsilateral and contralateral leg swings do not overlap. This insures that two legs in close proximity are not both swinging at the same time, which would endanger the static stability of the system. For contralateral leg pairs, the phase difference is independent of walking speed. For ipsilateral leg pairs, the phase difference is dependent on walking speed.

- A leg begins to swing shortly after the swing of the next caudal (rearward) leg is completed. Because the swing duration is independent of walking speed, this creates a continuous change in phase difference between ipsilateral legs as a function of walking speed.

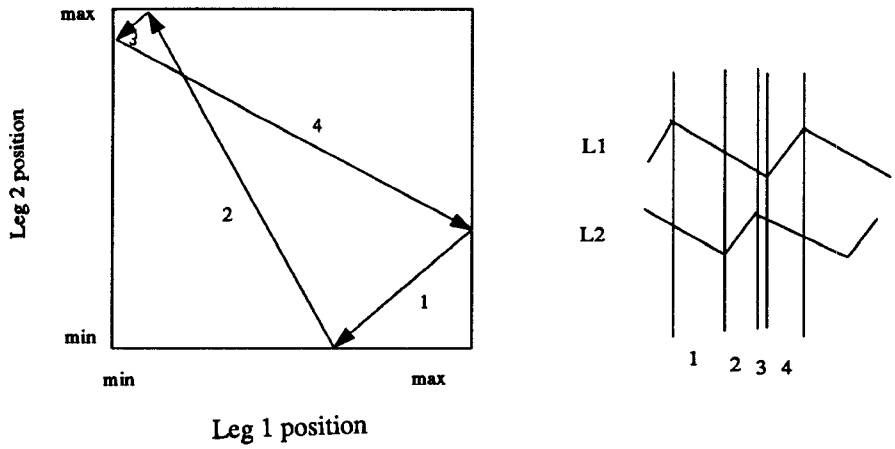


Figure 2.1: Phase-Plane Diagram (based on [7])

2.2.1 Phase-Plane Diagrams

One tool useful in describing the positional relationships between different legs as a result of different gaits is the phase-plane diagram [7]. Such a diagram plots the horizontal position of one leg versus the horizontal position of another. Assuming that the velocity of the leg in the horizontal direction is a constant during both the swing and the stance, this creates a graph consisting of straight line segments. A typical such diagram corresponding to a contraletral pair is shown in Figure 2.1. One reason the phase-plane diagram is useful is because the direction of the path through the diagram has a direct relationship to the status of the ground contact of the legs. If the motion is downward and to the left, both legs are in stance together. If the motion is downward and to the right, leg 1 is in swing while leg 2 is in stance. If the motion is upward and to the left, leg 1 is in stance while leg 2 is in swing. Finally, if the motion is upward and to the right, both legs are in swing together.

Furthermore, the slopes of the motions are related to the relative speed of stance and swing. The motions downward to the left and upward to the right will both have slopes of 1, since either both are in swing or both are in stance. The motion downward to the right has a slope of $-r$, and the motion upward to the left has a slope of $-\frac{1}{r}$, where r is the ratio of swing time to stance time. i.e. $r = \frac{t_{swing}}{t_{stance}}$

From the phase-plane diagram, it is easy to get a geometric picture of the phase relationship between two legs. As a result, we will be using phase-plane diagrams to illustrate the results of the coordination and control strategies discussed in this thesis.

2.3 Description of Oscillators

While the desired positional relationship between each leg could be achieved through a centrally directed computation, such a scheme is generally inflexible and non-robust. Complicated adjustments would be necessary to account for unexpected changes in the environment. In animals, whose coordination is flexible and robust, the leg coordination is accomplished through a central pattern generator that can be modeled as a system of nonlinear coupled oscillators. It is believed that the resulting system will be able to accomplish the determination of the positional relationship between each leg in a more flexible and robust

manner.

Each oscillator simply determines the desired phase of the associated leg. From this phase, the desired position of the leg can be determined from the desired path. An individual oscillator is described by the simple equation:

$$\dot{\theta} = \omega \tag{2.1}$$

where ω is a constant. The desired ω is easily determined from the quantities mentioned previously, t_{swing} and r.

In addition to the simple relationship for an individual oscillator, there are additional terms associated with coupling from other oscillators. The oscillator equation therefore becomes:

$$\dot{\theta}_i = \omega_i + \sum_j H_{ij} \left(\theta_i - \theta_j\right) \tag{2.2}$$

where H_{ij} is the coupling function between the ith and j^{th} oscillator. This is the form of the equations used by Cohen et. al. to model the central pattern generator of the lamprey [18][13].

2.4 Method of Coupling

This section describes the coupling between oscillators. First, the question of which oscillators should be coupled together is discussed, and then the coupling function itself is discussed.

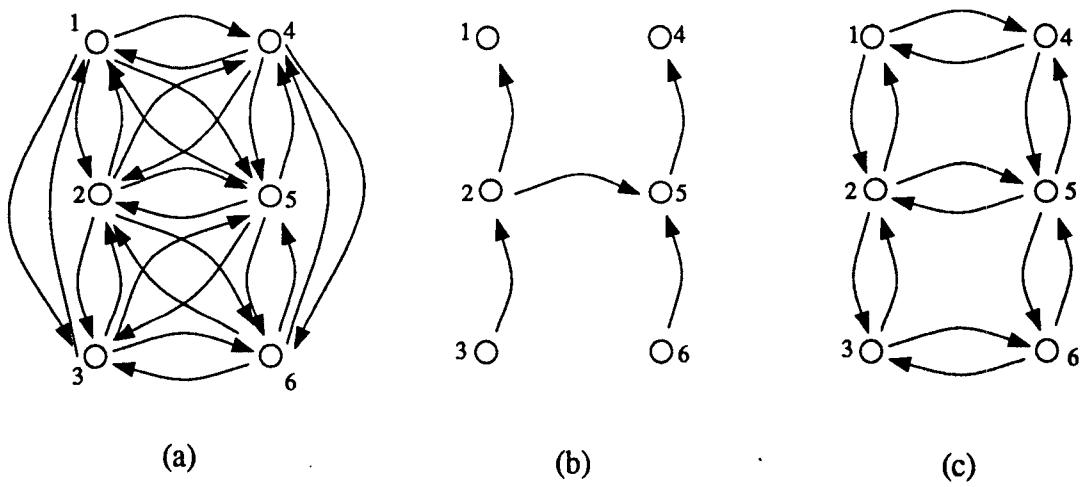


Figure 2.2: Coupling Options

2.4.1 Coupling Structure

The first, and perhaps most obvious, option of which oscillators to couple together is to simply have coupling between all of the legs (Figure 2.2a). While this yields the most flexible system, it is also the most complicated. Another option would be to have a limited amount of coupling (Figure 2.2b). This would be simpler to implement, but would be less flexible. *The choice made was to have only nearest neighbor coupling* (Figure 2.2c). This balances the amount of flexibility and simplicity, and is in basic agreement with what is found in the actual walking stick insect [9]. This arrangement yields the following set of equations:

$$\begin{aligned}
\dot{\theta}_1 &= \omega_1 + g_{12}H\left(\phi_{12_d} - (\theta_1 - \theta_2)\right) + g_{14}H\left(\phi_{14_d} - (\theta_1 - \theta_4)\right) \\
\dot{\theta}_2 &= \omega_2 + g_{21}H\left(\phi_{21_d} - (\theta_2 - \theta_1)\right) + g_{23}H\left(\phi_{23_d} - (\theta_2 - \theta_3)\right) + \\
&\quad g_{25}H\left(\phi_{25_d} - (\theta_2 - \theta_5)\right)
\end{aligned}$$

$$
\begin{aligned}
\dot{\theta}_3 &= \omega_3 + g_{32}H\left(\phi_{32_d} - (\theta_3 - \theta_2)\right) + g_{36}H\left(\phi_{36_d} - (\theta_3 - \theta_6)\right) \\
\dot{\theta}_4 &= \omega_4 + g_{45}H\left(\phi_{45_d} - (\theta_4 - \theta_5)\right) + g_{41}H\left(\phi_{41_d} - (\theta_4 - \theta_1)\right) \\
\dot{\theta}_5 &= \omega_5 + g_{54}H\left(\phi_{54_d} - (\theta_5 - \theta_4)\right) + g_{56}H\left(\phi_{56_d} - (\theta_5 - \theta_6)\right) + \\
&\quad g_{52}H\left(\phi_{52_d} - (\theta_5 - \theta_2)\right) \\
\dot{\theta}_6 &= \omega_6 + g_{65}H\left(\phi_{65_d} - (\theta_6 - \theta_5)\right) + g_{63}H\left(\phi_{63_d} - (\theta_6 - \theta_3)\right)
\end{aligned}
$$

where $\phi_{ij_d} = (\theta_i - \theta_j)_d$, the desired phase difference between oscillators i and j. To further investigate the above equations, it is useful to write them in terms of $\phi_{ij} = \theta_i - \theta_j$, the phase differences.

$$
\begin{aligned}
\dot{\phi}_{21} &= \xi_{21} + g_{21}H\left(\phi_{21_d} - \phi_{21}\right) - g_{23}H\left(\phi_{32_d} - \phi_{32}\right) - g_{25}H\left(\phi_{52_d} - \phi_{52}\right) + \\
&\quad g_{12}H\left(\phi_{21_d} - \phi_{21}\right) + g_{14}H\left(\phi_{41_d} - \phi_{41}\right) \\
\dot{\phi}_{32} &= \xi_{32} + g_{32}H\left(\phi_{32_d} - \phi_{32}\right) - g_{36}H\left(\phi_{63_d} - \phi_{63}\right) - g_{21}H\left(\phi_{21_d} - \phi_{21}\right) + \\
&\quad g_{23}H\left(\phi_{32_d} - \phi_{32}\right) + g_{25}H\left(\phi_{52_d} - \phi_{52}\right) \\
\dot{\phi}_{54} &= \xi_{54} + g_{54}H\left(\phi_{54_d} - \phi_{54}\right) - g_{56}H\left(\phi_{65_d} - \phi_{65}\right) + g_{52}H\left(\phi_{52_d} - \phi_{52}\right) + \\
&\quad g_{45}H\left(\phi_{54_d} - \phi_{54}\right) - g_{41}H\left(\phi_{41_d} - \phi_{41}\right) \\
\dot{\phi}_{65} &= \xi_{65} + g_{65}H\left(\phi_{65_d} - \phi_{65}\right) + g_{63}H\left(\phi_{63_d} - \phi_{63}\right) - g_{54}H\left(\phi_{54_d} - \phi_{54}\right) + \\
&\quad g_{56}H\left(\phi_{65_d} - \phi_{65}\right) - g_{52}H\left(\phi_{52_d} - \phi_{52}\right) \\
\dot{\phi}_{41} &= \xi_{41} - g_{45}H\left(\phi_{54_d} - \phi_{54}\right) + g_{41}H\left(\phi_{41_d} - \phi_{41}\right) + g_{12}H\left(\phi_{21_d} - \phi_{21}\right) + \\
&\quad g_{14}H\left(\phi_{41_d} - \phi_{41}\right) \\
\dot{\phi}_{52} &= \xi_{52} + g_{54}H\left(\phi_{54_d} - \phi_{54}\right) - g_{56}H\left(\phi_{65_d} - \phi_{65}\right) + g_{52}H\left(\phi_{52_d} - \phi_{52}\right) - \\
&\quad g_{21}H\left(\phi_{21_d} - \phi_{21}\right) + g_{23}H\left(\phi_{32_d} - \phi_{32}\right) + g_{25}H\left(\phi_{52}\right)
\end{aligned}
$$

$$\dot{\phi}_{63} = \xi_{63} + g_{65}H\left(\phi_{65_d} - \phi_{65}\right) + g_{63}H\left(\phi_{63_d} - \phi_{63}\right) - g_{32}H\left(\phi_{32_d} - \phi_{32}\right) + g_{36}H\left(\phi_{63_d} - \phi_{63}\right)$$

with $\xi_{ij} = \omega_i - \omega_j$, and where $H(\cdot)$ is assumed to be odd. In matrix form:

$$\dot{\phi} = \xi + G\mathcal{H}(\phi_d - \phi) \tag{2.3}$$

where

$$G = \begin{pmatrix} g_{12}+g_{21} & -g_{23} & 0 & 0 & g_{14} & -g_{25} & 0 \\ -g_{21} & g_{23}+g_{32} & 0 & 0 & 0 & g_{25} & -g_{36} \\ 0 & 0 & g_{45}+g_{54} & -g_{56} & -g_{41} & g_{52} & 0 \\ 0 & 0 & -g_{54} & g_{56}+g_{65} & 0 & -g_{52} & g_{63} \\ g_{12} & 0 & -g_{45} & 0 & g_{14}+g_{41} & 0 & 0 \\ -g_{21} & g_{23} & g_{54} & -g_{56} & 0 & g_{25}+g_{52} & 0 \\ 0 & -g_{32} & 0 & g_{65} & 0 & 0 & g_{36}+g_{63} \end{pmatrix}$$

and

$$\dot{\phi} = \begin{pmatrix} \dot{\phi}_{21} \\ \dot{\phi}_{32} \\ \dot{\phi}_{54} \\ \dot{\phi}_{65} \\ \dot{\phi}_{41} \\ \dot{\phi}_{52} \\ \dot{\phi}_{63} \end{pmatrix}, \quad \xi = \begin{pmatrix} \xi_{21} \\ \xi_{32} \\ \xi_{54} \\ \xi_{65} \\ \xi_{41} \\ \xi_{52} \\ \xi_{63} \end{pmatrix}, \quad \mathcal{H}(\phi_d - \phi) = \begin{pmatrix} H(\phi_{21_d} - \phi_{21}) \\ H(\phi_{32_d} - \phi_{32}) \\ H(\phi_{54_d} - \phi_{54}) \\ H(\phi_{65_d} - \phi_{65}) \\ H(\phi_{41_d} - \phi_{41}) \\ H(\phi_{52_d} - \phi_{52}) \\ H(\phi_{63_d} - \phi_{63}) \end{pmatrix}$$

Of these equations, only five are independent. There is a loop consistency equation due to the closed loops in the coupling that can be written as:

$$\begin{pmatrix} 1 & 0 & -1 & 0 & -1 & 1 & 0 \\ 1 & 1 & -1 & -1 & -1 & 0 & 1 \end{pmatrix} \begin{pmatrix} \phi_{21} \\ \phi_{32} \\ \phi_{54} \\ \phi_{65} \\ \phi_{41} \\ \phi_{52} \\ \phi_{63} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \tag{2.4}$$

or

$$L\phi = 0 \tag{2.5}$$

We now state a stability result for Equation 2.3.

Theorem 2.1 *Assume that $\xi = 0$, that ϕ_d is consistent (i.e. $L\phi_d = 0$), and that $g_{ij} > 0, \forall ij$. Furthermore, assume that $H(\sigma)$ is an odd, invertible function with positive slope over all possible σ, and that the coupling gains satisfy $g_{14}g_{45}g_{52}g_{21} = g_{12}g_{25}g_{54}g_{41}$ and $g_{23}g_{36}g_{65}g_{52} = g_{25}g_{56}g_{63}g_{32}$. Then, the origin is an asymptotically stable equilibrium point for Equation 2.3.*

Proof. First, a needed result about the matrix G is stated.

Lemma 2.2 *Assume that the constants g_{ij} in G are all chosen greater than zero, with $g_{14}g_{45}g_{52}g_{21} = g_{12}g_{25}g_{54}g_{41}$ and $g_{23}g_{36}g_{65}g_{52} = g_{25}g_{56}g_{63}g_{32}$. Then, the*

matrix G can be written as $A\Lambda$ where A is symmetric positive semi-definite with exactly two zero eigenvalues, and Λ is a diagonal matrix.

Proof. Choose the matrices A and Λ as follows:

$$A = \begin{pmatrix} 1+\frac{g_{12}}{g_{21}} & -1 & 0 & 0 & \frac{g_{12}}{g_{21}} & -1 & 0 \\ -1 & 1+\frac{g_{32}}{g_{23}} & 0 & 0 & 0 & 1 & -\frac{g_{32}}{g_{23}} \\ 0 & 0 & \frac{g_{52}}{g_{25}}\left(1+\frac{g_{45}}{g_{54}}\right) & -\frac{g_{52}}{g_{25}} & -\frac{g_{12}}{g_{21}}\frac{g_{41}}{g_{14}} & \frac{g_{52}}{g_{25}} & 0 \\ 0 & 0 & -\frac{g_{52}}{g_{25}} & \frac{g_{52}}{g_{25}}\left(1+\frac{g_{65}}{g_{56}}\right) & 0 & -\frac{g_{52}}{g_{25}} & \frac{g_{32}}{g_{23}}\frac{g_{63}}{g_{36}} \\ \frac{g_{12}}{g_{21}} & 0 & -\frac{g_{52}}{g_{25}}\frac{g_{45}}{g_{54}} & 0 & \frac{g_{12}}{g_{21}}\left(1+\frac{g_{41}}{g_{14}}\right) & 0 & 0 \\ -1 & 1 & \frac{g_{52}}{g_{25}} & -\frac{g_{52}}{g_{25}} & 0 & 1+\frac{g_{52}}{g_{25}} & 0 \\ 0 & -\frac{g_{32}}{g_{23}} & 0 & \frac{g_{52}}{g_{25}}\frac{g_{65}}{g_{56}} & 0 & 0 & \frac{g_{32}}{g_{23}}\left(1+\frac{g_{63}}{g_{36}}\right) \end{pmatrix} \tag{2.6}$$

$$\Lambda = \begin{pmatrix} g_{21} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & g_{23} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{g_{25}}{g_{52}}g_{54} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{g_{25}}{g_{52}}g_{56} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{g_{21}}{g_{12}}g_{14} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & g_{25} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \frac{g_{23}}{g_{32}}g_{36} \end{pmatrix} \tag{2.7}$$

It can easily be seen that the matrix A is symmetric if and only if $\frac{g_{12}}{g_{21}}\frac{g_{41}}{g_{14}} = \frac{g_{52}}{g_{25}}\frac{g_{45}}{g_{54}}$ and $\frac{g_{32}}{g_{23}}\frac{g_{63}}{g_{36}} = \frac{g_{52}}{g_{25}}\frac{g_{65}}{g_{56}}$, which is equivalent to the assumptions made.

A matrix is positive semi-definite if and only if all of the pivots for LU decomposition are greater than or equal to zero. Furthermore, the number of

zero pivots is equal to the number of zero eigenvalues [20]. The pivots for A can be shown to be:

$$\begin{pmatrix} 1+\frac{g_{12}}{g_{21}} \\ \frac{g_{12}}{g_{12}+g_{21}}+\frac{g_{32}}{g_{23}} \\ \frac{g_{52}}{g_{25}}\left(1+\frac{g_{45}}{g_{54}}\right) \\ \frac{g_{52}}{g_{25}}\left(\frac{g_{45}}{g_{45}+g_{54}}+\frac{g_{65}}{g_{56}}\right) \\ \frac{g_{12}g_{32}}{g_{12}g_{23}+g_{32}(g_{12}+g_{21})}+\frac{g_{52}}{g_{25}}\left(\frac{g_{45}g_{65}}{g_{45}g_{56}+g_{65}(g_{45}+g_{54})}\right) \\ 0 \\ 0 \end{pmatrix}$$

Therefore, $G = A\Lambda$, with A semi-positive definite with exactly two zero eigenvalues, and Λ diagonal. QED.

Select the Lyapanov function candidate:

$$V(\psi) = \int \Lambda\mathcal{H}(\psi)\cdot d(\psi)$$

where

$$\psi = \phi_d - \phi$$

Then, along trajectories of Equation 2.3,

$$\begin{aligned} \frac{d}{dt}V(\psi) &= -\left[\Lambda\mathcal{H}(\phi_d-\phi)\right]^T\dot{\phi} \\ &= -\left[\Lambda\mathcal{H}(\phi_d-\phi)\right]^T G\mathcal{H}(\phi_d-\phi) \\ &= -\left[\Lambda\mathcal{H}(\phi_d-\phi)\right]^T A\left[\Lambda\mathcal{H}(\phi_d-\phi)\right] \\ &\leq 0 \end{aligned}$$

Therefore, $\mathcal{H}(\phi_d - \phi) \rightarrow \mathcal{H}(\psi^*)$, with $A\Lambda\mathcal{H}(\psi^*) = G\mathcal{H}(\psi^*) = 0$.

Since $H(\cdot)$ is invertible, $\mathcal{H}(\cdot)$ can be considered a change of coordinates. Therefore, $G\mathcal{H}(\psi^*) = 0$ can be solved for ψ^*. Furthermore, the loop consistency equations add two equations of the form $L\psi^* = 0$. Therefore, the only solution is $\psi^* = 0 \Rightarrow (\phi_d - \phi) \rightarrow 0$. QED.

As Theorem 2.1 shows, the only necessary restrictions on the form of the coupling function to insure asymptotic stability are that the function be invertible and odd with positive slope. The only additional restriction on the coupling functions is that the coupling strengths around each loop be independent of the direction in which the loop is traversed.

In this thesis, the form chosen for the coupling function was

$$H(\sigma) = \sin(\frac{\pi}{2}\frac{\sigma}{\sigma_{max}}) \tag{2.8}$$

Notice that this has the desired characteristics for all possible values of σ. Furthermore, the coupling strengths were chosen such that $g_{ij} = g_{ji}, \forall ij$. This guarantees that the restriction on coupling strengths is achieved.

2.4.2 Desired Phase Relationship

The most important quantities for producing the desired gaits are the ϕ_{ij_d}'s. The choice of these constants determines the phase relationships between the legs. The desired gait is that which is described in Section 2.2.

First, some notation that will be used in the following:

- t_{swing} — The time taken for the swing portion of the leg motion.
- t_{stance} — The time taken for the stance portion of the leg motion.
- r — The ratio of swing time to stance time. $r = \frac{t_{swing}}{t_{stance}}$

One possible period for the oscillators is $1 + r$, where the $[0, 1]$ region designates the stance, and the $[1, 1 + r]$ region designates the swing. This choice of period sets the desired speed of the oscillators to be:

$$\omega = \frac{1}{t_{stance}} = \frac{r}{t_{swing}} \tag{2.9}$$

With this selection of the period, we now look at the desired phase difference for ipsilateral and contralateral legs.

As described before, ipsilateral legs begin to swing shortly after the swing of the next caudal leg is completed. The corresponding phase difference is simply r, the amount of phase for the swing portion. This amount of phase difference, however, leaves no margin for error. To insure that the next caudal leg is down before the leg lifts, it would be desirable to have a small amount of additional phase difference. This delay, however, must be dependent on the desired speed, since when $r = 1$ (the fastest speed) the delay must be zero. Therefore, we chose

$$\phi_{21_d} = \phi_{32_d} = \phi_{54_d} = \phi_{65_d} = r + \frac{1}{8}(1 - r) \tag{2.10}$$

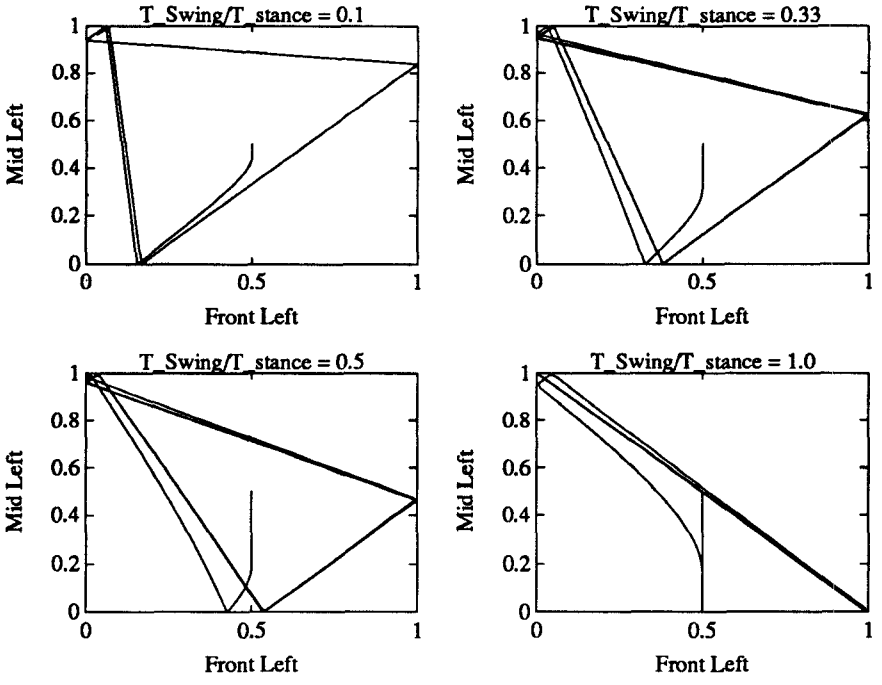


Figure 2.3: Phase-Plane Diagrams for Adjacent Ipsilateral Legs

The phase difference between contralateral legs must simply insure that the legs do not swing at the same time. A phase difference of 180° would insure this. In the chosen period, this means that

$$\phi_{41_d} = \phi_{52_d} = \phi_{63_d} = \pm\frac{1+r}{2} \tag{2.11}$$

The sign is determined based upon which is closer to the current phase difference.

The phase-plane diagrams using the described system of coupled oscillators are shown in Figures 2.3 and 2.4 for several choices of r (speed). Figure 2.3 shows the phase-plane diagrams for adjacent ipsilateral legs, while Figure 2.4 shows the phase-plane diagrams for contralateral legs. Figure 2.5 shows the corresponding plots of leg position versus time. As can be seen from the diagrams,

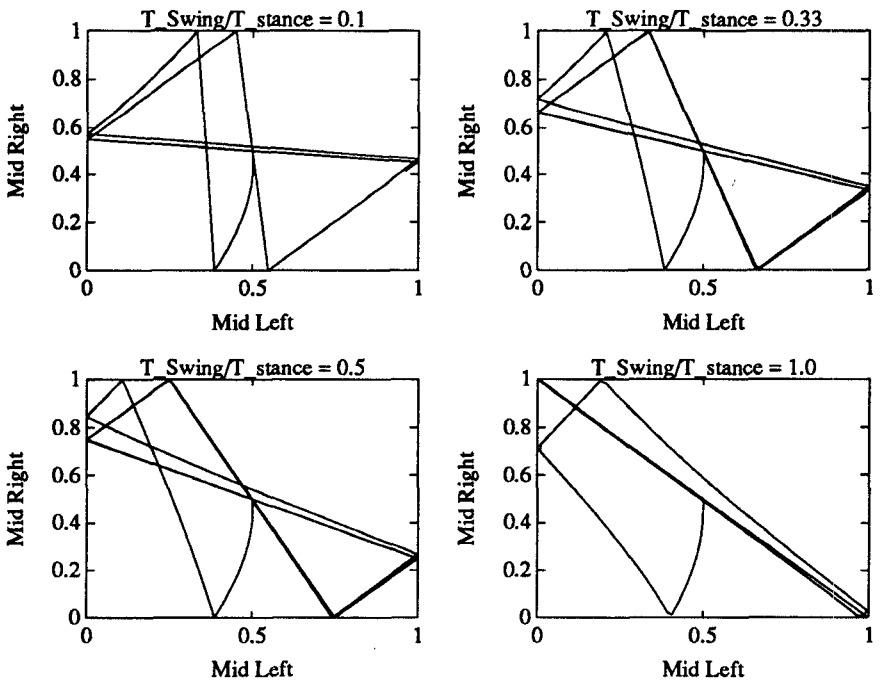


Figure 2.4: Phase-Plane Diagrams for Contralateral Legs

the phase relationship between the legs soon converges to the desired relationship. Furthermore, it can be seen that the generated gaits are the characteristic gaits of the walking stick insect. It should be noted that the so called alternating tripod gait is simply the metachronal wave gait taken to the limit where $t_{stance} = t_{swing}$.

Notice that with the system as previously described, if r changes, which is equivalent to changing the desired speed, then there could be a discontinuity in the desired leg position. To correct this problem, the following is done:

- The oscillators are given a period of 2, with the $[0, 1]$ region designating the stance, and the $[1, 2]$ region designating the swing.

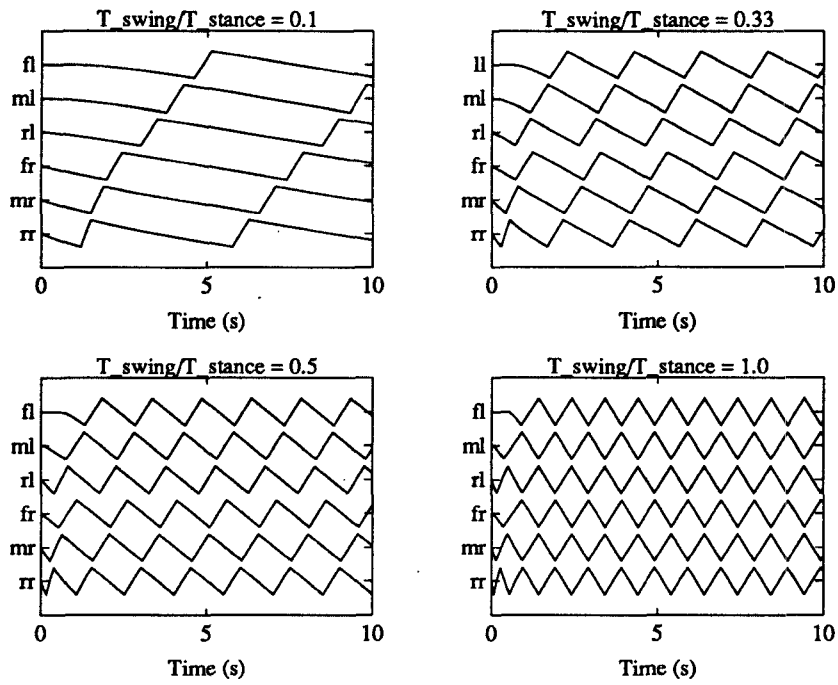


Figure 2.5: Generated Gaits for Several Speeds

- To make this equivalent to the previous discussion, the speed of the oscillator, ω is changed to:

$$\omega = \begin{cases} \frac{r}{t_{swing}} & \text{if } 0 \leq \theta \leq 1 \\ \frac{1}{t_{swing}} & \text{if } 1 \leq \theta \leq 2 \end{cases} \tag{2.12}$$

- The calculation of the coupling between legs is identical to that described above after the following transformation is done on the phase:

$$\tilde{\theta}_i = \begin{cases} \theta_i & \text{if } 0 \leq \theta_i \leq 1 \\ 1 + \frac{1}{r}(\theta_i - 1) & \text{if } 1 \leq \theta_i \leq 1 + r \end{cases} \tag{2.13}$$

Chapter 3

Model of the Walking Stick Insect

3.1 Introduction

In order to test the usefulness of different control strategies, it is desirable to have an accurate simulation of the system to be controlled. Toward this end, the equations of motion of the walking stick insect are derived, as well as a method for numerically solving them.

First, a fairly general formulation of the equations is discussed, without referring to any particulars of the actual system. This discussion includes the modeling of ground contact and the method of solving the dynamic equations with the nonholonomic constraints imposed by ground interaction. The specific

elements of the equations of motion are then derived for the walking stick insect. The dynamic equations are derived using a Lagrangian approach, while MACSYMA was used to perform the symbolic manipulation.

3.2 Formulation of Dynamic Equations

The basic equation of motion for a multibody system of rigid links without ground contact forces can be written as [6]:

$$\tau = M(q)\ddot{q} + V(q, \dot{q}) + G(q) \tag{3.1}$$

where

- q is the vector of generalized coordinates.
- τ is the vector of generalized forces.
- $M(q)$ is the mass matrix.
- $V(q, \dot{q})$ is the vector of coriolis and centrifugal forces.
- $G(q)$ is the vector of gravity forces.

Assuming the appropriate matrices are known, Equation 3.1 can be used to solve the dynamics for the system when there is no contact with the environment. One possible method would be to solve the following set of first order differential

equations:

$$\begin{bmatrix} \dot{q} \\ \ddot{q} \end{bmatrix} = \begin{bmatrix} \dot{q} \\ M^{-1}(q)\left[\tau - V(q,\dot{q}) - G(q)\right] \end{bmatrix} \tag{3.2}$$

However, once there is contact with the environment (i.e. a foot hitting the ground), there are additional constraints imposed on the system.

3.2.1 Ground Interaction

There are two components of ground interaction that must be considered: the constrained motion when the foot is in contact with the ground, and the actual impact with the ground.

Constrained Motion

When one or more feet are in contact with the ground, there are of course constraints on the possible motions of the leg joints. To make such constraints simpler, it was assumed that the feet do not ever slip on the ground (i.e. infinite friction). These nonholonomic constraints are of the form:

$$C(q)\dot{q} = 0 \tag{3.3}$$

These constraint equations are generally very complicated in the joint space coordinate system normally considered. One possible way to make the constrained system easier to solve is to change the coordinate system in the dynamic equations to one in which the constraint equations are much simpler. This is very

similar to that which is done in [12]. One such coordinate system (and the one that we use) is the coordinate system that uses the position of the feet in the inertial frame as coordinates rather than the rotation of the leg joints. Let us call these cartesian space coordinates x as opposed to the joint space coordinates q.

Notice that in this cartesian coordinate system, the constraint equations imposed by a foot being in contact with the ground are simply that the cartesian coordinates associated with that foot are constant. If we let ${}^{i}x$ denote the cartesian coordinates associated with leg i, then the constraint equations can be written as

$$ {}^{i}\dot{x}(t) = 0, \forall t, \quad \text{when leg } i \text{ is on the ground.} \tag{3.4} $$

The cartesian velocity and joint velocity are related by the appropriate jacobian:

$$ \dot{x} = J(q)\dot{q} \tag{3.5} $$

while the cartesian acceleration is:

$$ \ddot{x} = J(q)\ddot{q} + \dot{J}(q)\dot{q} \tag{3.6} $$

Combining Equations 3.2, 3.5, and 3.6 yields the following set of first order differential equations:

$$ \begin{bmatrix} \dot{q} \\ \ddot{x} \end{bmatrix} = \begin{bmatrix} J^{-1}(q)\dot{x} \\ J(q)M^{-1}(q)\left[\tau - V(q, J^{-1}(q)\dot{x}) - G(q)\right] + \dot{J}(q)J^{-1}(q)\dot{x} \end{bmatrix} \tag{3.7} $$

Assuming that $J(q)$ is never singular, such a formulation makes solving the constrained dynamic equations quite straightforward. The forces of constraint never even need to be calculated. It should be noted that the assumption that $J(q)$ is never singular is fine for the walking stick insect, since the legs are never fully stretched.

Impact Modeling

The actual impact with the ground may be modeled as totally elastic, totally inelastic, or as a combination of the two. While there is certainly some amount of elasticity in the impact with the ground, it is not of the instantaneous nature that is typically referred to as an elastic collision. Such an instantaneous response of a completely elastic collision would cause the legs to "bounce" when they contacted the ground. A much better model of ground interaction would consist of an inelastic collision with a spring modeled on the foot. However, for simplicity the collision is being modeled as totally inelastic, with the spring being saved for future improvements. Therefore, the equation of impact is

$$ {}^{i}\dot{x}(t^{+}) = 0, \quad \text{when leg } i \text{ hits the ground at time } t. \tag{3.8} $$

The strategy for solving the dynamic equations can be summarized as follows:

1. Check to see if any leg which is not thought to be on the ground is at ground level and moving downward. If so, set the cartesian velocity ${}^{i}\dot{x}$ of that leg to zero. (Equation 3.8)

2. Calculate the joint space velocity $\dot{q}$ and the cartesian space acceleration $\ddot{x}$. (Equation 3.7)

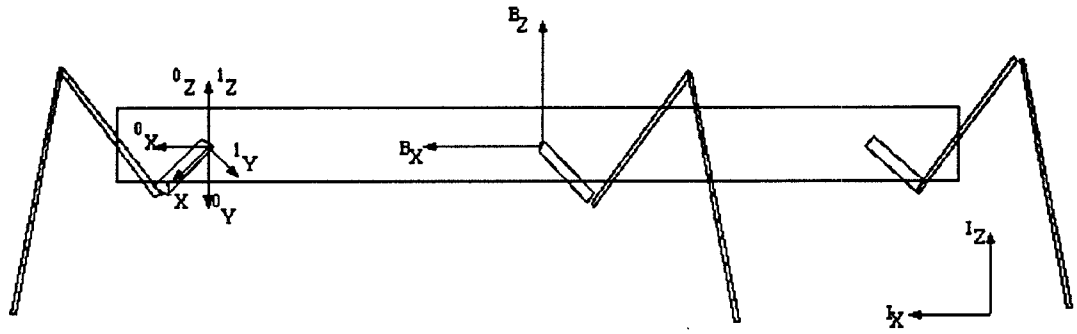
3. Set ${}^{i}\ddot{x}$ to zero if leg i is on the ground and the vertical component of the leg's acceleration is not positive (i.e. not upwards). (Equation 3.4)

3.3 Derivation of Walking Stick Model

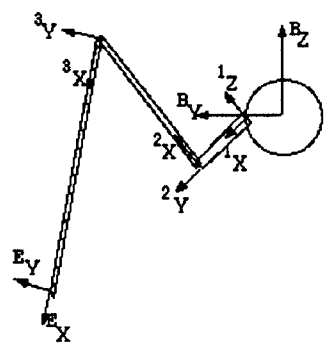
In order to solve the dynamic equations via the method summarized at the end of the last section, the various elements of the equations must first be determined. We begin by defining the particulars of the geometry of the walking stick insect, as well as the notation for describing the system. The kinematics of the system are then derived, followed by a Lagrangian derivation of the dynamics of the system. In large part the following is based on the methods described in [6].

3.3.1 Geometry and Notation

The walking stick insect is composed of six three-segment legs attached to one body segment, where all segments are considered to be rigid. The basic geometry of the system is as in Figure 3.1. Each leg segment is connected by a rotational joint, with the axis of rotation being the z-axis as drawn in the figure. The system therefore has 24 degrees of freedom (6 for the body and 3 per leg). The fundamental masses and geometry of the walking stick insect



(a)



(b)

Figure 3.1: Basic geometry of the walking stick insect

	mass (mg)	length (mm)	diameter (mm)
Main Body	872.4	72	5
Leg Segment 1	3.3	1.7	1.5
Leg Segment 2	9.1	12.3	1.0
Leg Segment 3	2.2	11.6	0.5

Table 3.1: Masses and Geometry

are as in Table 3.1 [16]. The six degrees of freedom of the body will be specified by the three translational positions (q_1, q_2, q_3) and the three ZYX Euler angles (q_4, q_5, q_6). The three degrees of freedom of each leg will be specified by $(\theta_{1_i}, \theta_{2_i}, \theta_{3_i})$ or by $(q_{3i+4}, q_{3i+5}, q_{3i+6})$.

In describing the system, the following frames of reference will be used:

- The inertial coordinate frame I. The x-axis is in the direction of desired movement, and the z-axis is directed vertically upwards.
- The body coordinate frame B. The origin is at the center of mass of the body segment, the x-axis is directed along the longitudinal axis of the body, and the z-axis is directed upwards perpendicular to the plane containing the leg connection points.
- The leg coordinate axis 0_i. The origin is at the connection point of leg i, the x-axis is the same as in frame B, and the z-axis is directed along the axis of rotation of the first leg link, tilted by an angle α from the z-axis of frame B. The subscript will often be omitted if a statement is true in

general for all of the legs.

- The leg coordinate axis 1_i. The origin is the same as frame 0_i, the x-axis is directed along the first leg link, and the z-axis is directed along the axis of rotation of the first leg link. The subscript will often be omitted if a statement is true in general for all of the legs.

- The leg coordinate axis 2_i. The origin is at the joint connecting the first and second leg links, the x-axis is directed along the second leg link, and the z-axis is directed along the axis of rotation of the second leg link. The subscript will often be omitted if a statement is true in general for all of the legs.

- The leg coordinate axis 3_i. The origin is at the joint connecting the second and third leg links, the x-axis is directed along the third leg link, and the z-axis is directed along the axis of rotation of the third leg link. The subscript will often be omitted if a statement is true in general for all of the legs.

- The leg coordinate axis E_i. The origin is at the end of the third leg link, and the orientation is the same as frame 3_i. The subscript will often be omitted if a statement is true in general for all of the legs.

The following notation will be used:

- ${}^iR_j$ — The 3x3 rotation matrix that denotes the orientation of frame j in

frame i coordinates.

- ${}^{i}p_j$ — The 3x1 vector that specifies the origin of frame j in frame i coordinates.

- ${}^{i}p_{c_j}$ — The 3x1 vector that specifies the center of mass of the link associated with frame j relative to the origin of frame j, written in frame i coordinates.

- ${}^{i}V_j^k$ — The linear velocity of frame j in relation to frame k, written in frame i coordinates. If the k frame reference is omitted, the velocity is relative to frame i.

- ${}^{i}\Omega_j^k$ — The angular velocity of frame j in relation to frame k, written in frame i coordinates. If the k frame reference is omitted, the velocity is relative to frame i.

- ${}^{i}v_j$ — The linear velocity of frame j relative to the inertial frame, written in frame i coordinates. i.e. ${}^{1}v_2 = {}^{1}V_2^I$

- ${}^{i}\omega_j$ — The angular velocity of frame j relative to the inertial frame, written in frame i coordinates. i.e. ${}^{1}\omega_2 = {}^{1}\Omega_2^I$

- ${}^{i}v_{c_j}$ — The linear velocity of the center of mass of the link associated with frame j relative to the inertial frame, written in frame i coordinates.

The link transformations between all of the links are summarized in Appendix A.

3.3.2 Kinematics

In this section, the kinematic elements required to solve the dynamic equations for the walking stick insect are developed. This includes the relevant Jacobians, as well as expressions for the velocities and accelerations of the links which will be needed for the dynamics.

Using the notation stated in Section 3.3.1, the position of the foot in the inertial frame is ${}^{I}p_{E_i}$, which can be calculated as follows:

$$\begin{aligned} {}^{2}p_E &= {}^{2}R_3\,{}^{3}p_E + {}^{2}p_3 \\ {}^{1}p_E &= {}^{1}R_2\,{}^{2}p_E + {}^{1}p_2 \\ {}^{B}p_E &= {}^{B}R_0\,{}^{0}R_1\,{}^{1}p_E + {}^{B}p_0 \\ {}^{I}p_E &= {}^{I}R_B\,{}^{B}p_E + {}^{I}p_B \end{aligned} \tag{3.9}$$

The necessary transformation from joint space coordinates to cartesian coordinates can then be found as:

$$\begin{aligned} x_i &= q_i \quad ; \quad 1 \le i \le 6 \\ [\, x_{(3i+4)} \quad x_{(3i+5)} \quad x_{(3i+6)} \,]^T &= {}^{I}p_{E_i} \quad ; \quad 1 \le i \le 6 \end{aligned} \tag{3.10}$$

and

$$J_{ij}(q) = \frac{\partial x_i}{\partial q_j} \tag{3.11}$$

$$\dot{J}_{ij}(q) = \sum_{k=1}^{n} \frac{\partial J_{ij}(q)}{\partial q_k} \dot{q}_k \tag{3.12}$$

Equations 3.10, 3.11, and 3.12 give all of the kinematic elements explicitly required to solve the equations of motion in the method suggested in the previous section.

We now turn our attention to the kinematic quantities needed in the next section to derive the dynamic elements of the equations of motion. In particular, we will determine the position (${}^{I}p_{c_i}$), angular velocity (${}^{i}\omega_i$), and linear velocity (${}^{i}v_{c_i}$) of the center of mass of each link. Using the link transformations, the position of the center of mass of each link can be found from:

$$
\begin{aligned}
{}^{I}p_1 &= {}^{I}R_B\,{}^{B}p_0 + {}^{I}p_B & {}^{I}p_{c_1} &= {}^{I}R_1\,{}^{1}p_{c_1} + {}^{I}p_1 \\
{}^{I}p_2 &= {}^{I}R_1\,{}^{1}p_2 + {}^{I}p_1 & {}^{I}p_{c_2} &= {}^{I}R_2\,{}^{2}p_{c_2} + {}^{I}p_2 \\
{}^{I}p_3 &= {}^{I}R_2\,{}^{2}p_3 + {}^{I}p_2 & {}^{I}p_{c_3} &= {}^{I}R_3\,{}^{3}p_{c_3} + {}^{I}p_3
\end{aligned}
\tag{3.13}
$$

The angular and linear velocities of the center of mass of each link can be determined recursively using the following equations:

$$
{}^{i+1}\omega_{i+1} = {}^{i+1}R_i\left({}^{i}\omega_i + {}^{i}\Omega^{i}_{i+1}\right) \tag{3.14}
$$

$$
{}^{i+1}v_{i+1} = {}^{i+1}R_i\left({}^{i}v_i + {}^{i}\omega_i \times {}^{i}p_{i+1} + {}^{i}V^{i}_{i+1}\right) \tag{3.15}
$$

$$
{}^{i}v_{c_i} = {}^{i}v_i + {}^{i}\omega_i \times {}^{i}p_{c_i} \tag{3.16}
$$

First we can calculate the angular and linear velocity of each link relative to the previous link:

$$
\begin{aligned}
{}^{I}\hat{\Omega}_B &= {}^{I}\dot{R}_B\,{}^{B}R_I & {}^{I}V_B &= [\ \dot{q}_1 \quad \dot{q}_2 \quad \dot{q}_3\ ]^T \\
{}^{B}\hat{\Omega}_0 &= {}^{B}\dot{R}_0\,{}^{0}R_B = 0 & {}^{B}V_0 &= 0 \\
{}^{0}\hat{\Omega}_1 &= {}^{0}\dot{R}_1\,{}^{1}R_0 & {}^{0}V_1 &= 0 \\
{}^{1}\hat{\Omega}_2 &= {}^{1}\dot{R}_2\,{}^{2}R_1 & {}^{1}V_2 &= 0 \\
{}^{2}\hat{\Omega}_3 &= {}^{2}\dot{R}_3\,{}^{3}R_2 & {}^{2}V_3 &= 0
\end{aligned}
\tag{3.17}
$$

where

$$\widehat{\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}} = \begin{bmatrix} 0 & -x_3 & x_2 \\ x_3 & 0 & -x_1 \\ -x_2 & x_1 & 0 \end{bmatrix} \text{ so } \hat{x}y = x \times y \tag{3.18}$$

The angular velocities with respect to the inertial frame can then be found as:

$$\begin{aligned} {}^B\omega_B &= {}^BR_I\,{}^I\Omega_B \\ {}^1\omega_1 &= {}^1R_0\left({}^0R_B\,{}^B\omega_B + {}^0\Omega_1\right) \\ {}^2\omega_2 &= {}^2R_1\left({}^1\omega_1 + {}^1\Omega_2\right) \\ {}^3\omega_3 &= {}^3R_2\left({}^2\omega_2 + {}^2\Omega_3\right) \end{aligned} \tag{3.19}$$

and the linear velocity with respect to the inertial frame can be found as:

$$\begin{aligned} {}^Bv_B &= {}^BR_I\,{}^IV_B \\ {}^1v_1 &= {}^1R_0\,{}^0R_B\left({}^Bv_B + {}^B\omega_B \times {}^Bp_1\right) \\ {}^2v_2 &= {}^2R_1\left({}^1v_1 + {}^1\omega_1 \times {}^1p_2\right) \\ {}^3v_3 &= {}^3R_2\left({}^2v_2 + {}^2\omega_2 \times {}^2p_3\right) \end{aligned} \tag{3.20}$$

Finally, the linear velocity of the center of masses with respect to the inertial frame can be found as:

$$\begin{aligned} {}^Bv_{c_B} &= {}^Bv_B \\ {}^1v_{c_1} &= {}^1v_1 + {}^1\omega_1 \times {}^1p_{c_1} \\ {}^2v_{c_2} &= {}^2v_2 + {}^2\omega_2 \times {}^2p_{c_2} \\ {}^3v_{c_3} &= {}^3v_3 + {}^3\omega_3 \times {}^3p_{c_3} \end{aligned} \tag{3.21}$$

Now that we have found expressions for all of the kinematic quantities needed, we will determine the dynamics for the system.

3.3.3 Dynamics

We will now derive the elements of Equation 3.1, the unconstrained dynamics, using Lagrangian mechanics. i.e. We will find expressions for $M(q)$, $V(q,\dot{q})$, and $G(q)$.

The Lagrangian for our system will be defined by:

$$L(q,\dot{q}) = K(q,\dot{q}) - P(q) \tag{3.22}$$

where

- K is the kinetic energy of the system.
- P is the potential energy of the system.

The unconstrained dynamic equations of motion are then:

$$\begin{aligned} \tau_j &= \frac{d}{dt}\frac{\partial L}{\partial \dot{q}_j} - \frac{\partial L}{\partial q_j} \\ &= \frac{d}{dt}\frac{\partial K}{\partial \dot{q}_j} - \frac{\partial K}{\partial q_j} + \frac{\partial P}{\partial q_j} \\ &= \sum_{k=1}^{n}\left(\frac{\partial^2 K(q,\dot{q})}{\partial \dot{q}_k \partial \dot{q}_j}\ddot{q}_k + \frac{\partial^2 K(q,\dot{q})}{\partial q_k \partial \dot{q}_j}\dot{q}_k\right) - \frac{\partial K(q,\dot{q})}{\partial q_j} + \frac{\partial P(q)}{\partial q_j} \end{aligned} \tag{3.23}$$

From examination of Equations 3.1 and 3.23 it is obvious that:

$$M_{jk}(q) = \frac{\partial^2}{\partial \dot{q}_k \partial \dot{q}_j} K(q,\dot{q}) \tag{3.24}$$

$$V_j(q,\dot{q}) = \sum_{k=1}^{n} \frac{\partial^2 K(q,\dot{q})}{\partial q_k \partial \dot{q}_j}\dot{q}_k - \frac{\partial K(q,\dot{q})}{\partial q_j} \tag{3.25}$$

$$G_j(q) = \frac{\partial}{\partial q_j} P(\dot{q}) \tag{3.26}$$

The kinetic and potential energies are defined by:

$$K(q,\dot{q}) = \sum_i \left(\frac{1}{2}\,{}^i\omega_i^T I_{c_i}\,{}^i\omega_i + \frac{1}{2}m_i\,{}^i v_{c_i}^T\,{}^i v_{c_i}\right) \tag{3.27}$$

$$\begin{aligned} P(q) &= g\sum_i m_i\,[\,0 \quad 0 \quad 1\,]\,{}^I p_{c_i} \\ &= g\sum_i m_i\left({}^I p_{c_i}\right)_3 \end{aligned} \tag{3.28}$$

where

- i is summed over all of the links in the system.
- m_i is the mass of link i.
- I_{c_i} is the inertial tensor of link i at the center of mass.

Since ${}^i\omega_i$ and ${}^i v_{c_i}$ are linear in terms of $\dot{q}$, this yields the following:

$$M_{jk}(q) = \sum_i \left[\frac{\partial\,{}^i\omega_i^T}{\partial\dot{q}_j} I_{c_i} \frac{\partial\,{}^i\omega_i}{\partial\dot{q}_k} + m_i \frac{\partial\,{}^i v_{c_i}^T}{\partial\dot{q}_j}\frac{\partial\,{}^i v_{c_i}}{\partial\dot{q}_k}\right] \tag{3.29}$$

$$\begin{aligned} V_j(q,\dot{q}) = \sum_i \Bigg\{ & \frac{\partial^2\,{}^i\omega_i^T}{\partial q_k \partial\dot{q}_j} I_{c_i}\,{}^i\omega_i\dot{q}_k + \left[\frac{\partial\,{}^i\omega_i^T}{\partial\dot{q}_j} I_{c_i}\frac{\partial\,{}^i\omega_i}{\partial q_k} - \frac{\partial\,{}^i\omega_i^T}{\partial\dot{q}_k} I_{c_i}\frac{\partial\,{}^i\omega_i}{\partial q_j}\right]\dot{q}_k + \\ & m_i \frac{\partial^2\,{}^i v_{c_i}^T}{\partial q_k \partial\dot{q}_j}\,{}^i v_{c_i}\dot{q}_k + m_i\left[\frac{\partial\,{}^i v_{c_i}^T}{\partial\dot{q}_j}\frac{\partial\,{}^i v_{c_i}}{\partial q_k} - \frac{\partial\,{}^i v_{c_i}^T}{\partial\dot{q}_k}\frac{\partial\,{}^i v_{c_i}}{\partial q_j}\right]\dot{q}_k \Bigg\} \end{aligned} \tag{3.30}$$

$$G_j(q) = g\sum_i m_i \frac{\partial\,({}^I p_{c_i})_3}{\partial q_j} \tag{3.31}$$

Using these equations, the unconstrained dynamic equations of motion can be determined. Furthermore, in conjunction with the kinematic elements derived in the previous section, and using the scheme described in Section 3.2.1, the constrained dynamic equations of motion can also be determined.

3.3.4 Implementation

Although in the preceding sections expressions were given which can be solved to find the necessary kinematic and dynamic equations, it would not be feasible to calculate these expressions by hand. MACSYMA, however, provides the tools needed to do the symbolic manipulations. Using MACSYMA, it is not only possible to calculate the needed expressions, but it is also possible to generate C source code which calculates $M(q)$, $V(q, \dot{q}$, $G(q)$, $J(q)$, and $\dot{J}(q)$, the elements needed to determine the kinematic and dynamic equations of motion for the walking stick insect.

Since the expressions produced contain many repetitions of the same subexpressions, it would be undesirable to have to recalculate long trigonometric expressions. While MACSYMA provides a means to accomplish this through the OPTIMIZE() function, it was not deemed feasible to optimize all expressions concurrently. Instead, while the code for each element is optimized, there are common terms between elements which are recalculated. Appendix B contains a sample of the MACSYMA source code which generated the appropriate C code. The code thus generated was then used to create a numerical simulation of the dynamics of the walking stick insect.

3.4 Numerical Methods

In the previous sections, we have discussed the development of a system of first order differential equations that models the dynamics of the walking stick insect. These equations are of the form:

$$\dot{\chi} = \mathcal{F}(\chi) \tag{3.32}$$

where χ is the state of the system.

In this particular case, the function $\mathcal{F}(\cdot)$ is very time consuming to compute, so a numerical method which does not require many evaluations of the derivative function is desirable. The function $\mathcal{F}(\cdot)$ is also not smooth due to ground interaction constraints which place discontinuities in the derivative function. Furthermore, as a result of such ground constraints, certain states are required to be constant, and slight changes in their value can create large changes in the resulting motion i.e. the foot will be thought to have left the ground when in fact it should still be on the ground. The ideal numerical method will be able to handle such discontinuities in $\mathcal{F}(\cdot)$, while using few computations of the derivative function.

Multistep numerical methods such as Adams-Bashforth and the predictor/corrector method Adams-Moulton have the desirable property of relatively few computations of $\mathcal{F}(\cdot)$. However, as a result of their multistep nature, they do not work well over discontinuities, and are thus not well suited for this problem.

The standard fourth order Runge-Kutta numerical method does generally work well over discontinuities, but within each step, it does not necessarily work well in terms of maintaining the constraint of the foot being in contact with the ground. As a result, the Runge-Kutta formulation was also found to be lacking.

The one numerical method that was found to work acceptably was the Euler method. While this method in general is not one of the better choices, in this case no other method was found to be better. Because of the one step nature of the Euler method, discontinuities were no problem and the constraints were maintained. While the Euler method requires very small step sizes in order to be accurate, no other acceptable numerical methods were found. While the Euler method was chosen to be used at this juncture, it is hoped that a better, more efficient algorithm can be found or developed.

3.5 Summary

In this chapter, a formulation of the dynamic equations of motion was derived which allows the simple handling of the nonholonomic constraints due to ground interaction. In addition, the specific equations for the walking stick insect were derived using a Lagrangian approach. The use of MACSYMA for the actual symbolic manipulations and C code generation was also discussed. In the next chapter, we will see how this code was used to make a simulation of the walking

stick insect.

Chapter 4

Control of the Walking Stick Insect

4.1 Introduction

This chapter discusses the design of the control system for the walking stick insect, whose equations of motion were derived in the previous chapter. This control system consists of the coordination of the legs as well as the determination of joint torques which result in the desired motions. The design of the distributed simulation used to test the control system is also discussed, as well as some of the results of the simulation.

The basic objective of the control system was to enable the system to be able to produce the characteristic gaits associated with the walking stick insect in

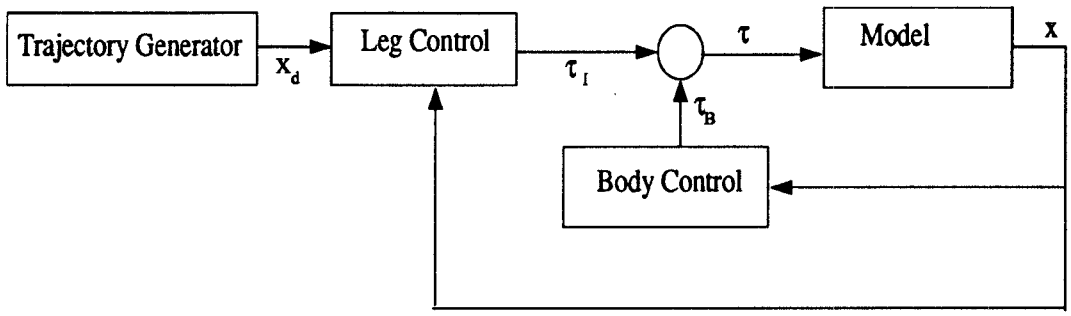


Figure 4.1: Control Structure

straight walking on flat ground. The systems of coupled nonlinear oscillators discussed in Chapter 2 are used to coordinate the legs to produce the characteristic gaits.

4.2 Control Strategy

The desired control structure is one which is highly distributed, with only certain functions being performed in a centralized manner. The basic structure of the control systems is as is shown in Figure 4.1. The trajectory generator determines the desired position of the legs at any given time. The leg controller calculates the joint torques necessary to follow the desired trajectory. The body controller calculates the joint torque necessary to maintain the body in the desired height and orientation. Of these components, only the body controller is implemented in a centralized manner. These three main components, the trajectory generator, the leg controller, and the body controller, will be discussed separately in detail.

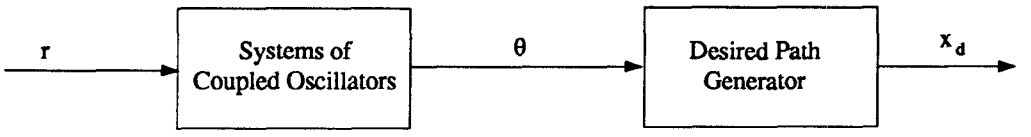


Figure 4.2: Trajectory Generator Structure

4.2.1 Desired Trajectory Generator

The trajectory generator must determine the desired position of the feet in relation to the body at every instant of time. The determination of the desired position is one of the most complicated problems that needs to be solved in order to have a useful walking machine. In general many things need to be taken into account, including variations in terrain. At this stage, only straight walking on flat ground is being considered, so some more advanced concerns about foot placement need not be considered.

The trajectory generator is composed of two main parts: a desired path generator and a desired phase generator (see Figure 4.2). Assuming the phase of the stride is known, the path generator provides the desired position of the foot. The phase generator provides the desired phase, and is thus responsible for the coordination and timing among the legs.

Desired Path

The path of the foot has several needed characteristics. First, the path while the foot is in stance (on the ground) should be flat, since in this case the body

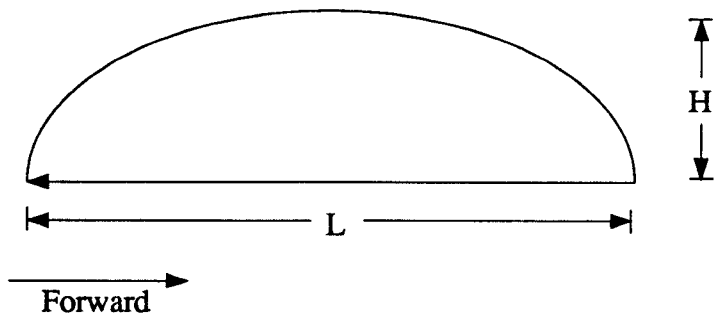


Figure 4.3: Desired Foot Path

height will remain constant, which saves energy. In addition, a curved foot path while in stance would create coordination problems between feet, since different feet are entering the stance region at varying times throughout the stride. The transition from stance to swing should involve a quick lift of the foot so that there is a smooth transition between the two regions. One foot path which provides all of these is the one shown in Figure 4.3. This path is described by:

$$x_d(\theta) = \begin{cases} \begin{bmatrix} L(\frac{1}{2} - \theta) \\ -H_B \\ W \end{bmatrix} & \text{if } 0 \leq \theta \leq 1 \text{ (stance)} \\ \begin{bmatrix} -\frac{1}{2}L\cos(\pi(\theta - 1)) \\ H\sin(\pi(\theta - 1)) - H_B \\ W \end{bmatrix} & \text{if } 1 \leq \theta \leq 2 \text{ (swing)} \end{cases} \tag{4.1}$$

where

- x_d is the desired position of the foot relative to the connection point of the leg to the body expressed in the body coordinates.
- L is the desired length of the stride.

- H is the desired height of lift during the swing.
- W is the desired horizontal distance of the foot away from the body.
- H_B is the desired height of the body.
- θ is the phase of the motion and varies between 0 and 2.

Notice that the swing path is simply the upper half of an ellipse.

Desired Phase

The desired phase for each leg is found using the system of coupled nonlinear oscillators described in Chapter 2. This system will accomplish the correct coordination of the legs to produce the characteristic gaits of the walking stick insect.

4.2.2 Leg Control

The function of the leg controller is to calculate the necessary joint torques to apply that will make the actual foot position closely follow the desired foot position. The basic method used to accomplish this is proportional-derivative plus gravity control of the form:

$$F_i = m_{\text{eff}}[K_P(x_{d_i} - x_i) + K_d(\dot{x}_{d_i} - \dot{x}_i)] \tag{4.2}$$

$$\tau_{l_i} = \begin{cases} J_{l_i}^T F_i & \text{if leg } i \text{ is in stance} \\ J_{l_i}^T F_i + G_{l_i}(q) & \text{if leg } i \text{ is in swing} \end{cases} \tag{4.3}$$

where

- x_{d_i} is the desired foot position of leg i in body coordinates.
- x_i is the actual foot position of leg i in body coordinates.
- K_p is the proportional gain.
- K_d is the differential gain.
- m_{eff} is the approximate effective mass seen by a force at the foot.
- F_i is the desired force to apply at the foot of leg i.
- J_{l_i} is the Jacobian which translates leg joint velocities into foot velocities in body coordinates for leg i.
- $G_{l_i}(q)$ is the relevant elements of G(q), the gravity vector, for leg i.
- τ_{l_i} is the desired leg joint torques for leg i.

This leg controller structure is shown in Figure 4.4.

The gravity compensation $G_l(q)$ is only included when the leg is in swing, since the ground force will counteract most of the gravitational force when the leg is in stance. In addition, the body controller will also help to counteract the

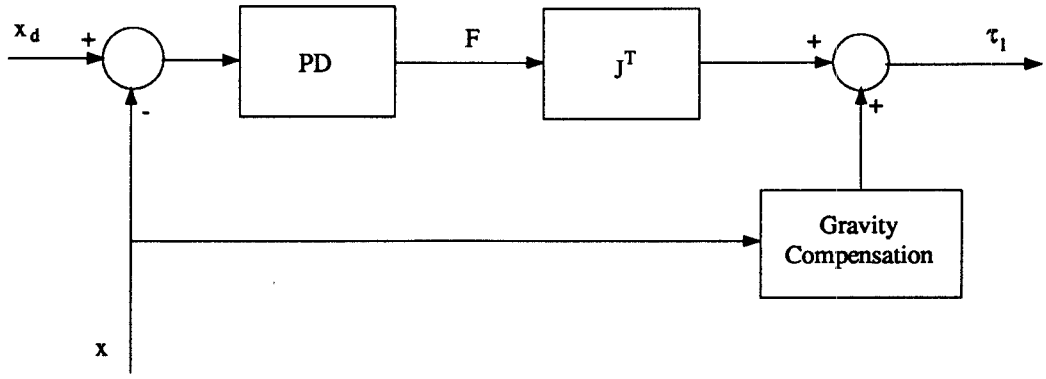


Figure 4.4: Leg Controller Structure

gravitational force when the leg is in stance (see Section 4.2.3). The effective mass m_{eff} is also dependent on whether the foot is on the ground or not, while all of the other quantities remain constant. The difference in the effective mass is due to the fact that when the foot is in the air the leg motion is not appreciably affected by the mass of the body, but if the foot is on the ground then the body has to be moved in order to change the leg position. As a result:

$$m_{\text{eff}} \doteq \begin{cases} m_{\text{leg}} & \text{if foot is in the air} \\ \frac{1}{4} m_{\text{body}} & \text{if foot is on the ground} \end{cases} \tag{4.4}$$

While such a control scheme will of course not achieve tremendous accuracy in following the desired trajectory, the numerical results showed that, with the gravity compensator, it did well enough to enable the insect to walk.

4.2.3 Body Control

The body controller must maintain the height and orientation of the body. In particular, the relatively large gravity force on the body must be counteracted to

keep the body from falling to the ground. Since the body itself does not contain any actuators, the only means to control the body is through forces generated from the leg actuators. As only the legs which are touching the ground can generate much of a force on the body, these are the only legs which will be used by the body controller, and the other legs will be assumed to have a negligible effect on the body.

First, let us look at the wrench that is felt on the body as a result of the reaction forces of the ground against the legs. The force and torque on the body due to leg i are:

$$f_i^B = \begin{cases} f_i & \text{if leg i is on the ground} \\ 0 & \text{otherwise} \end{cases} \tag{4.5}$$

$$\tau_i^B = \begin{cases} {}^B p_{E_i} \times f_i & \text{if leg i is on the ground} \\ 0 & \text{otherwise} \end{cases} \tag{4.6}$$

where

- f_i^B is the force on the body due to leg i.
- τ_i^B is the torque on the body due to leg i.
- f_i is the reaction force on the foot of leg i.
- ${}^B p_{E_i}$ is the position of the foot of leg i in body frame coordinates.

This can be written in matrix form as:

$$\begin{bmatrix} f^B \\ \tau^B \end{bmatrix} = \begin{bmatrix} D_1 & D_2 & D_3 & D_4 & D_5 & D_6 \\ \hat{r}_1 & \hat{r}_2 & \hat{r}_3 & \hat{r}_4 & \hat{r}_5 & \hat{r}_6 \end{bmatrix} \begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \\ f_5 \\ f_6 \end{bmatrix} \tag{4.7}$$

where

$$D_i = \begin{cases} I_3 & \text{if leg i is on the ground} \\ 0 & \text{otherwise} \end{cases}$$

$$r_i = \begin{cases} {}^B p_{E_i} & \text{if leg i is on the ground} \\ 0 & \text{otherwise} \end{cases}$$

$$\widehat{\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}} = \begin{bmatrix} 0 & -x_3 & x_2 \\ x_3 & 0 & -x_1 \\ -x_2 & x_1 & 0 \end{bmatrix}$$

If the desired force on the body was known, Equation 4.7 could be solved for the desired ground reaction forces. In general this linear system of equations could be underdetermined or overdetermined depending on how many legs are on the ground and where they are positioned. In the underdetermined case, the gravity force on the body cannot be exactly compensated, but the best compensation in the least squared sense can be found. In the overdetermined case there are infinite ways of compensating for the gravity, but Equation 4.7

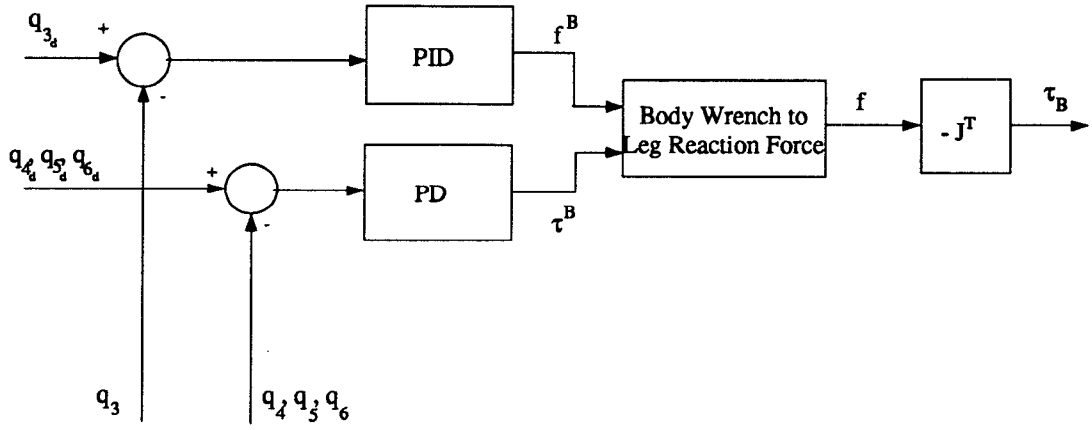


Figure 4.5: Body Controller Structure

can be solved for the solution with the smallest L_2 norm. In order to determine the solutions as mentioned above, a Singular Value Decomposition algorithm was used to solve the set of equations. This algorithm was used since it handles underdetermined and overdetermined sets of equations in exactly the desired manner.

One further complication in the solution of this set of equations is that we cannot achieve a reaction force in the negative vertical direction, although the equations may desire such a reaction force. For example, this would occur when all of the legs on one side are on the ground while the legs on the other side are in the air. In this case the torque balance is impossible without a negative vertical reaction force. This problem was solved by checking to see if a negative vertical reaction force was desired for any leg, and if it was, then resolving Equation 4.7 as if that leg was not on the ground.

While we have now outlined an approach to finding the reaction forces needed

to achieve a desired wrench on the body, we have not yet discussed the desired wrench, or how to achieve the desired reaction forces. The wrench on the body should not just compensate the gravitational force, but also attempt to maintain the body's desired position. This desired position is:

$$\begin{aligned} q_{3_d} &= H_B \\ q_{4_d} &= 0 \\ q_{5_d} &= 0 \\ q_{6_d} &= 0 \end{aligned} \tag{4.8}$$

where q_{i_d} is as defined in Section 3.3.1. i.e. the body should maintain a fixed height from the ground with no rotations in any direction. As a result, the desired wrench on the body, written in body coordinates, is chosen to be:

$$f^B = {}^B R_I \, m_B \begin{bmatrix} 0 \\ 0 \\ g + K_p(H - q_3) - K_d \dot{q}_3 - K_i \int q_3 dt \end{bmatrix} \tag{4.9}$$

$$\tau^B = {}^B R_I \, I_{c_B} \begin{bmatrix} -K_p q_6 - K_d \dot{q}_6 \\ -K_p q_5 - K_d \dot{q}_5 \\ -K_p q_4 - K_d \dot{q}_4 \end{bmatrix} \tag{4.10}$$

This wrench on the body should compensate the gravitational force as well as maintain the body at it desired height and orientation. Notice that while the orientation is only controlled with PD control, the height is controlled with PID control. The integral term is desired so that the gravitational force will be counteracted even if the model is not perfect.

In order to achieve the desired reaction force as calculated from Equation 4.7, the joint torques must be found which exert the negative force on the foot. This can be found as:

$$\tau_{B_i} = -J_{l_i}^T f_i \tag{4.11}$$

where J_{l_i} is the same Jacobian as in Equation 4.3.

The body controller structure is shown in Figure 4.5.

4.2.4 Summary

The torque τ_i to be applied at the joints of leg i is therefore:

$$\tau_i = \tau_{l_i} + \tau_{B_i} \tag{4.12}$$

where τ_{l_i} is determined by the leg controller and τ_{B_i} is determined by the body controller. It should again be noted that only the body controller is centralized in that it requires knowledge of all of the legs. All other aspects of the controller are decentralized, with only certain knowledge passing between legs through the coupling of the nonlinear oscillators. It should further be noted that the speed of the walk can be regulated by adjusting only one parameter, the ratio of swing time to stance time, which can vary between 0 (stopped) and 1 (full speed).

4.3 Simulation Description

In creating a real-time graphical simulation, there are two main jobs that need to be performed. First, there is a need for a graphics routine to provide the actual user interface. Second, there is a need for a numerical routine to solve the dynamic equations. In the Intelligent Servosystems Laboratory (ISL) at the University of Maryland, there has been much work done in the past to facilitate the development of such simulations. In particular, a library of routines for the development of a graphical user interface on the Silicon Graphics IRIS workstation has been created. In addition, a library of routines has been created to enable a process running on a numerical engine such as a SUN SPARCstation to interface with the graphics engine running on the IRIS.

4.3.1 Graphics

To enable the quick and easy development of a graphical user interface and graphical output, the Graphical Simulation Management System (GSMS) was developed [21]. This system provides tools for:

1. building interfaces (panels consisting of buttons, sliders, type in boxes, etc.)

2. arranging the simulation workspace

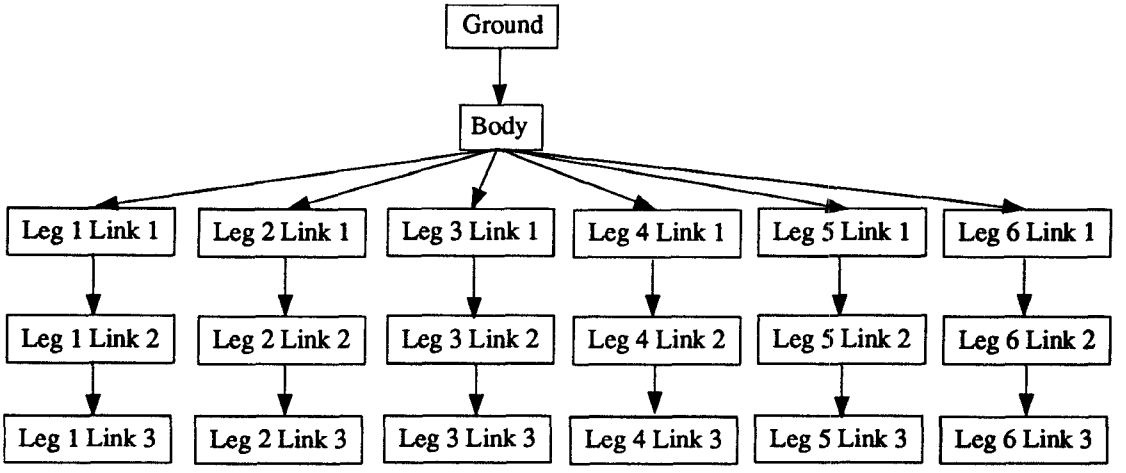


Figure 4.6: Object Network

3. describing complex 3D objects in terms of predefined primitives

4. endowing objects with properties used in conjunction with lighting models

Graphical Representation of the Walking Stick Insect

The graphical representation of the walking stick insect was constructed from predefined cylinder primitives. The primitives were then placed in the object network as shown in Figure 4.6. The transformation between each object in the object network is defined by the scale (sx,sy,sz), position (px,py,pz), and rotation (rx,ry,rz). Most of the transformation quantities are constants dependent on the geometry of the insect. The quantities that are not determined by the constant geometry can be determined from the joint space states of the simulation, consisting of one rotation for each leg link and all rotations and positions for the body. While there is a one-to-one relationship between the joint space position states and the needed values for the transformation, some conversion is needed.

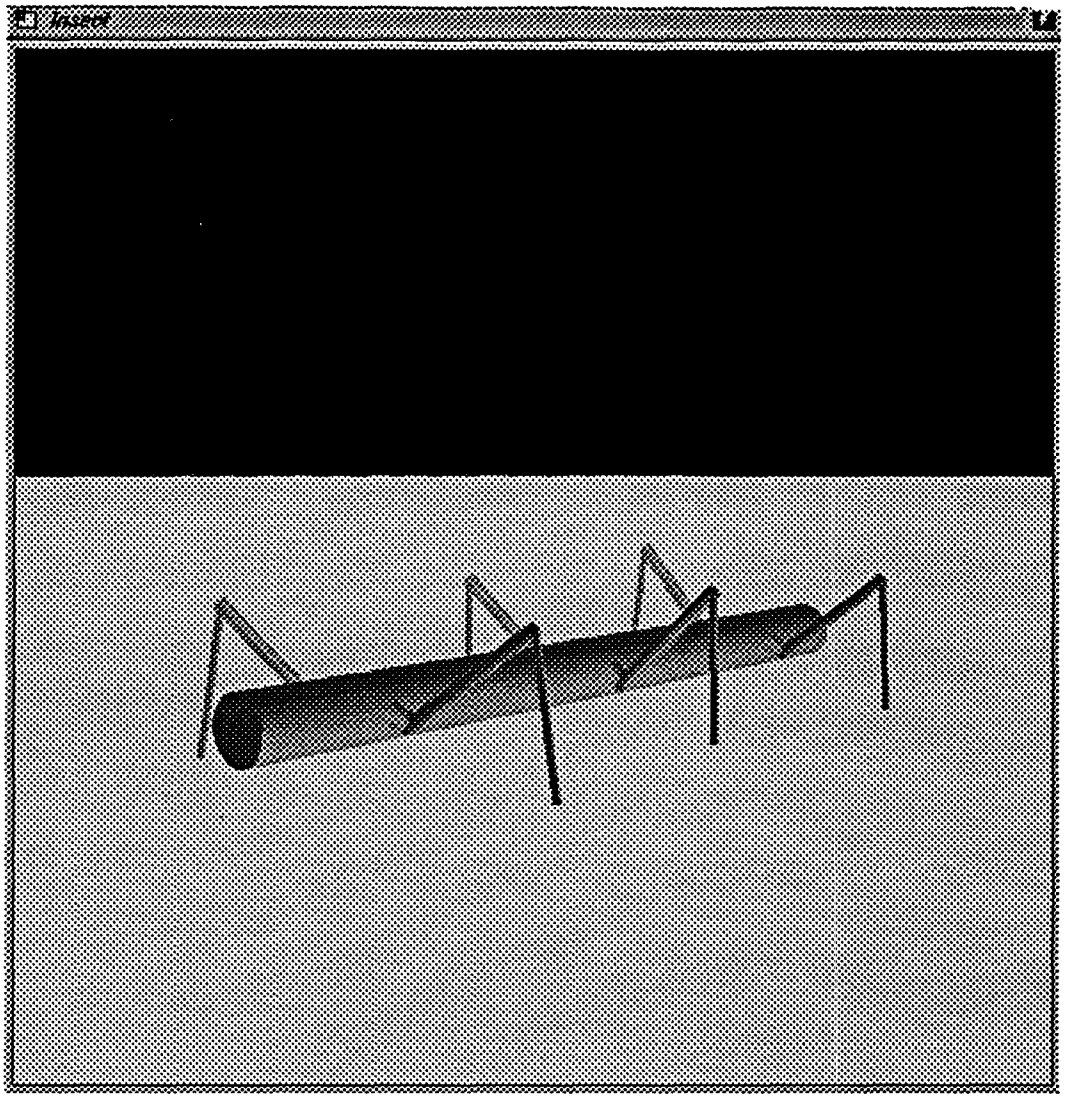


Figure 4.7: Graphical Representation of the Walking Stick Insect

In particular, the rotation angles are defined using XYZ Euler angles, while the body states were defined using ZYX Euler angles (see Section 3.3.1). Figure 4.7 shows the resulting graphical representation of the walking stick insect.

User Interface

Using GSMS it is quite easy to produce a user friendly graphical interface consisting of panels of buttons, sliders, etc. This type of user interface provides an easy means of manipulating a large number of inputs without erroneous inputs. The user interface for this simulation consists of three panels: Graphics, Legs, and Control Settings. The Graphics panel allows the user to change the position and orientation of the view of the walking stick insect on the screen. The Legs panel allows the user to move every degree of freedom of the walking stick insect. Furthermore, during a simulation the Legs panel displays the positions of all of the degrees of freedom. Finally, the Control Settings panel allows the user to select simulation parameters and to start/stop the simulation.

4.3.2 Interprocess Communication

While it would certainly be possible to have one machine do all of the desired numerical computations as well as handle the graphics necessary, it is much faster to split the tasks over more than one processor. It is with this goal in mind that the Interprocess Communication Library (libipc) was created [19]. This library provides the routines needed to set up a connection across the network between two different processes, and the means to transfer information across that connection. This allows one to set up a system where a SPARCstation runs the code requiring large amount of number crunching, while an IRIS workstation

Item	Description
tag	Command sequence number
command	The command in integer form
TimeStep	The desired integration time step
TimeInterval	The desired interval to integrate before updating the graphics
TimeRatio	The ratio of stance time to swing time
Pro_Gain[]	The proportional gains for leg control
Diff_Gain[]	The differential gains for leg control
state[]	The joint space position of the system

Table 4.1: Data sent from IRIS to SPARC

Item	Description
tag	Command sequence number
command	The command in integer form
Time	The current simulation time
state[]	The joint space position of the system

Table 4.2: Data sent from SPARC to IRIS

handles the graphics required. The graphics program is started first, with the desired name of the server machine passed on the command line. The server is then started with the graphics machine name and the socket number passed on the command line. The server then connects to the graphics process using the socket number. At this point, there is a connection between the two processes through which information can be passed. Table 4.1 shows the information that is passed from the graphics client to the server. Table 4.2 shows the information sent from the server to the graphics client. There are five basic commands which can be sent from the IRIS to the SPARC. They are summarized in Table 4.3.

Command	Description
EXIT	Exits the process
RESET	Resets the position to the default settings
SIM	Begins the simulation with the specified settings
STOP	Stops the simulation
STATE	Changes the current position of the system, or the system parameters

Table 4.3: Commands sent from IRIS to SPARC

4.4 Results

The controller described previously in Section 4.2 was shown to be capable of controlling the walking stick insect well enough to enable walking at speeds up to a ratio of swing time to stance time of $\frac{1}{2}$, which is two thirds of full walking speed. In this section, results from three different speeds are shown: a slow walking speed of $r = \frac{1}{10}$, a slightly faster walking speed of $r = \frac{1}{3}$, and a walking speed of $r = \frac{1}{2}$. These speed correspond to $\frac{2}{11}$, $\frac{1}{2}$, and $\frac{2}{3}$ of full walking speed, respectively. In all cases, the initial conditions were the same; each foot was displaced a constant distance away from the body, in the same horizontal position as the leg's attachment point to the body.

4.4.1 Leg Control

Figures 4.8, 4.9, and 4.10 show the foot motion in the horizontal direction for the three different speeds, respectively. The horizontal position of the foot relative

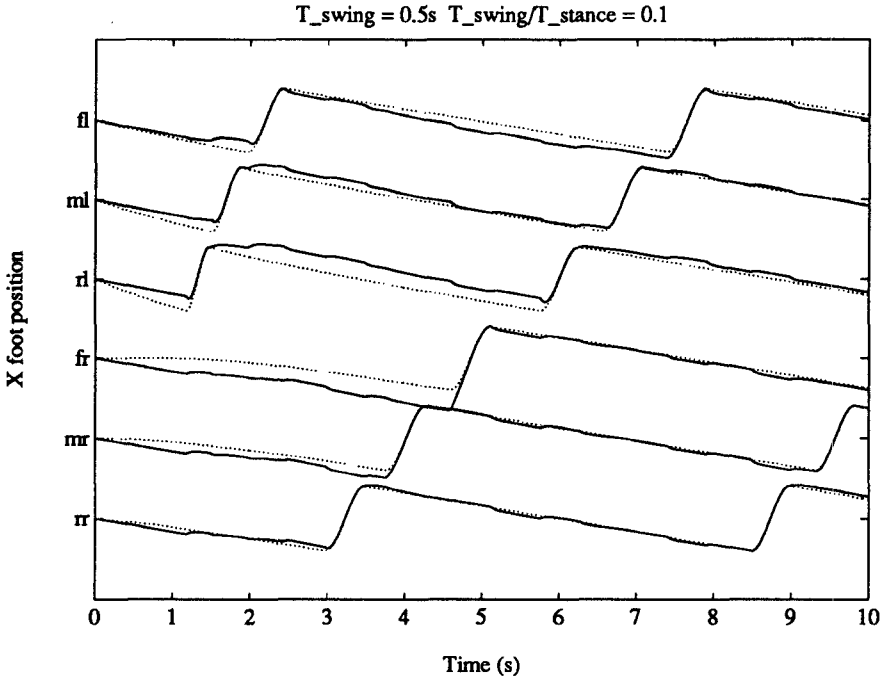


Figure 4.8: Horizontal Foot Motions, r = 0.1

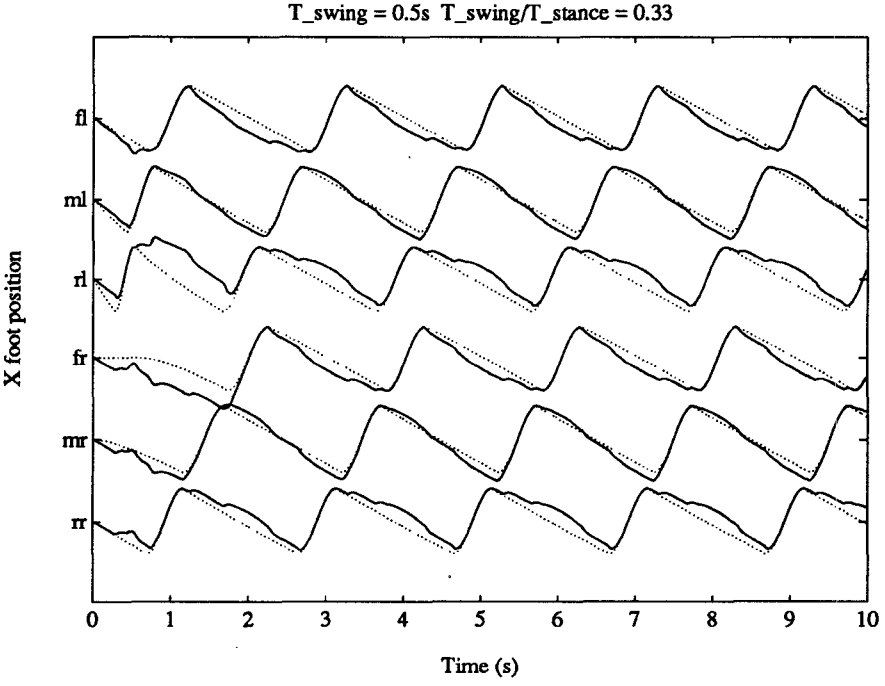


Figure 4.9: Horizontal Foot Motions, r = 0.33

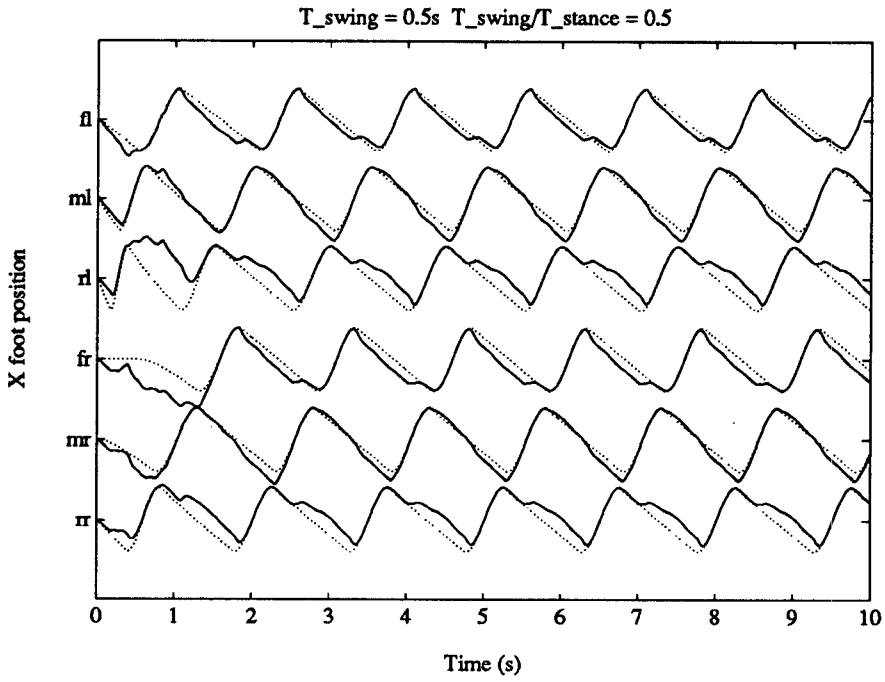


Figure 4.10: Horizontal Foot Motions, r = 0.5

to the body is very similar to the phase of the legs, so these figures show the phase relationship between the legs very clearly. The dotted lines indicates the desired position of the foot, while the solid line indicates the actual position. It can be seen that the actual position follows the desired position fairly well, especially at the lower speeds.

Figure 4.11 shows a representative graph of the foot path in the X-Z plane. It can be be seen that the foot path is close to the desired foot path shown in Figure 4.3.

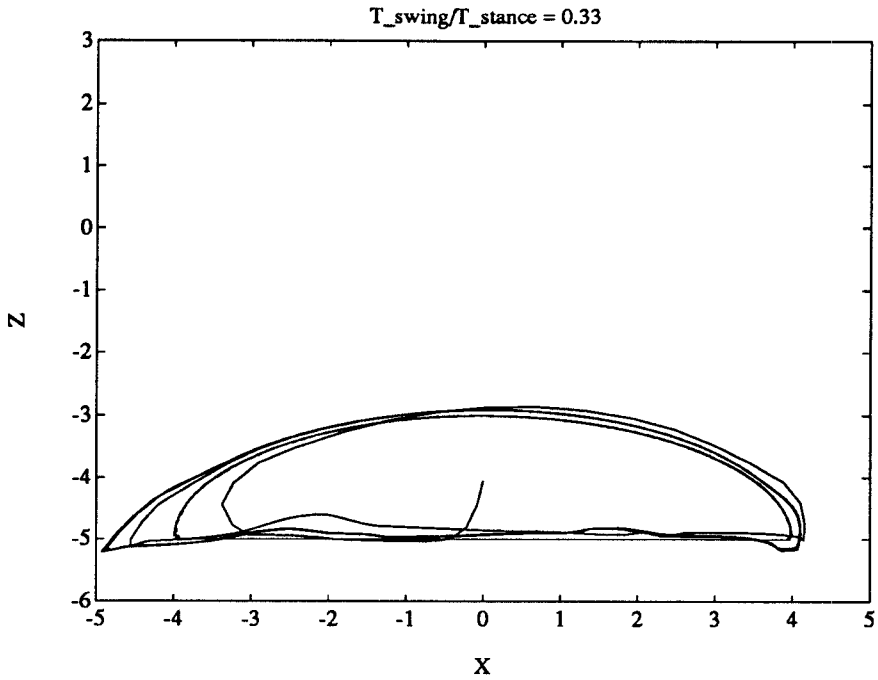


Figure 4.11: Foot Motions in X-Z Plane

4.4.2 Phase-Plane Diagrams

In Section 2.2.1, the concept of the phase-plane diagram was described, and in Section 2.4.2, the phase-plane diagram was used to show the desired phase relationship among the legs. In order to show the actual phase relationship among the legs, we will again use phase-plane diagrams, but instead of using the actual phase (which is difficult to calculate), we will use the horizontal position of the legs. Figures 4.12, 4.13, and 4.14 show the phase-plane diagram for adjacent ipsilateral legs for the three different speeds. Figures 4.15, 4.16, and 4.17 show the phase-plane diagrams for contralateral legs for the same three speeds. In all cases, the desired phase relationship is shown with a dotted line, and the actual phase relationship is shown with a solid line. It can be seen that the desired

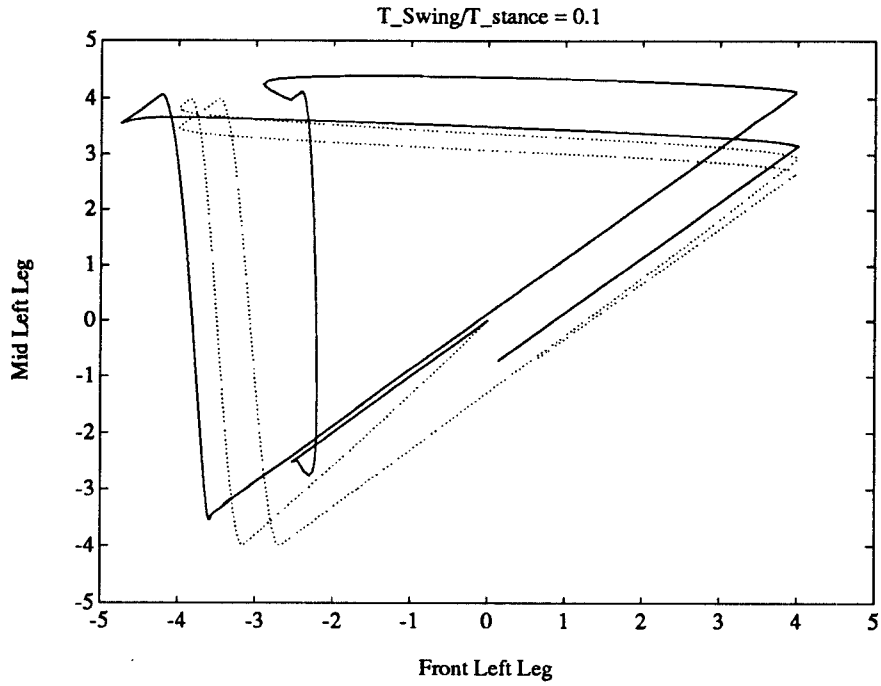


Figure 4.12: Ipsilateral Phase-Plane Diagram, r = 0.1

form of the phase-plane diagram is reached in all cases.

4.4.3 Body Control

In the body controller, four degrees of freedom of the body are controlled: the body height, and the ZYX Euler angles. Figures 4.18, 4.19 and 4.20 show these four quantities for each of the three speeds. While in all cases the body height is controller well, the rotation of the body gets increasingly worse as the speed increases. This is a result of the fact that when the insect is walking at a higher speed, there are fewer legs on the ground at any one time. When the walking speed is less than $r = \frac{1}{6}$, there are at least five legs on the ground at all times.

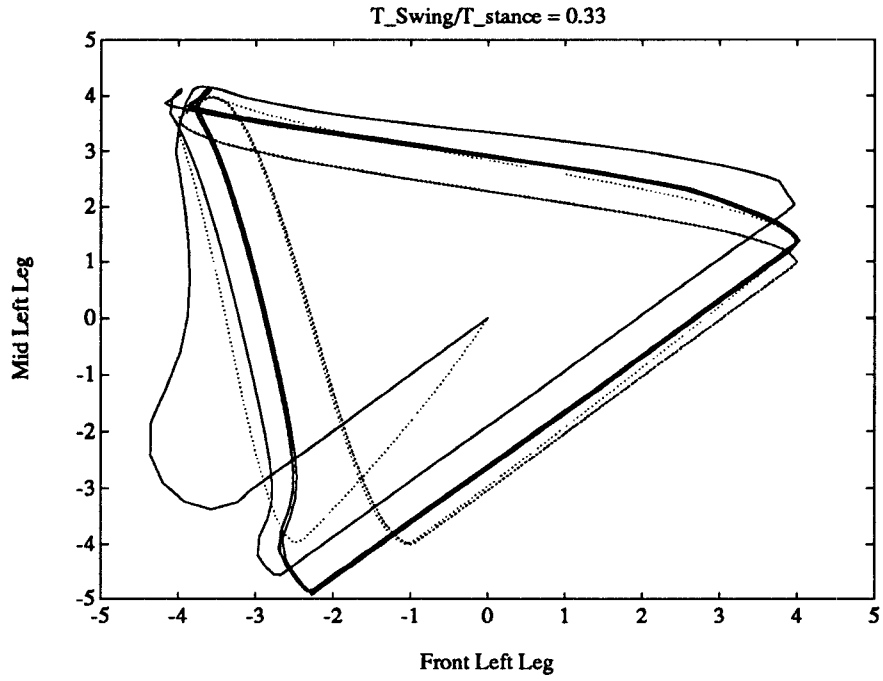


Figure 4.13: Ipsilateral Phase-Plane Diagram, r = 0.33

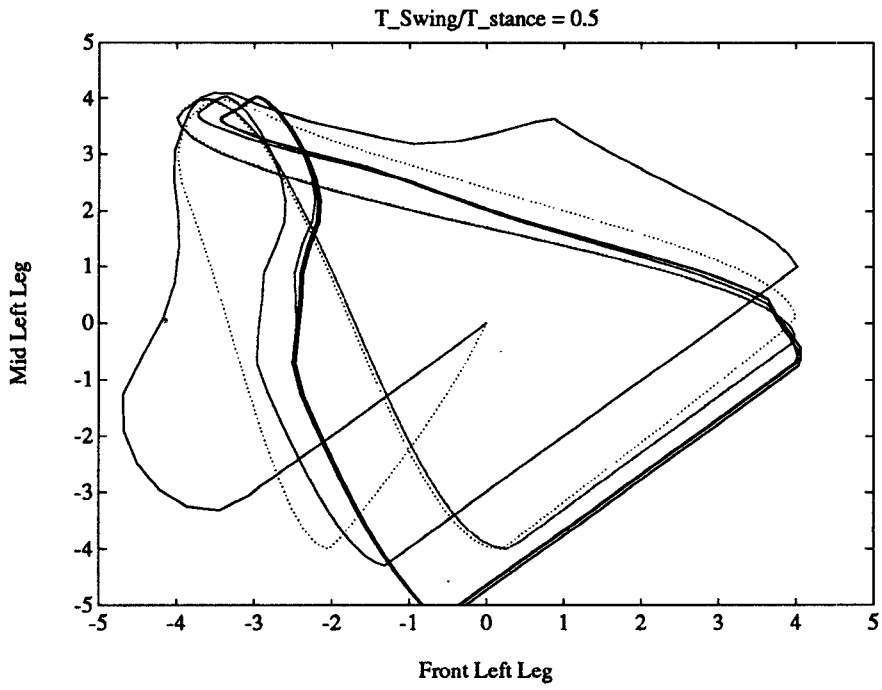


Figure 4.14: Ipsilateral Phase-Plane Diagram, r = 0.5

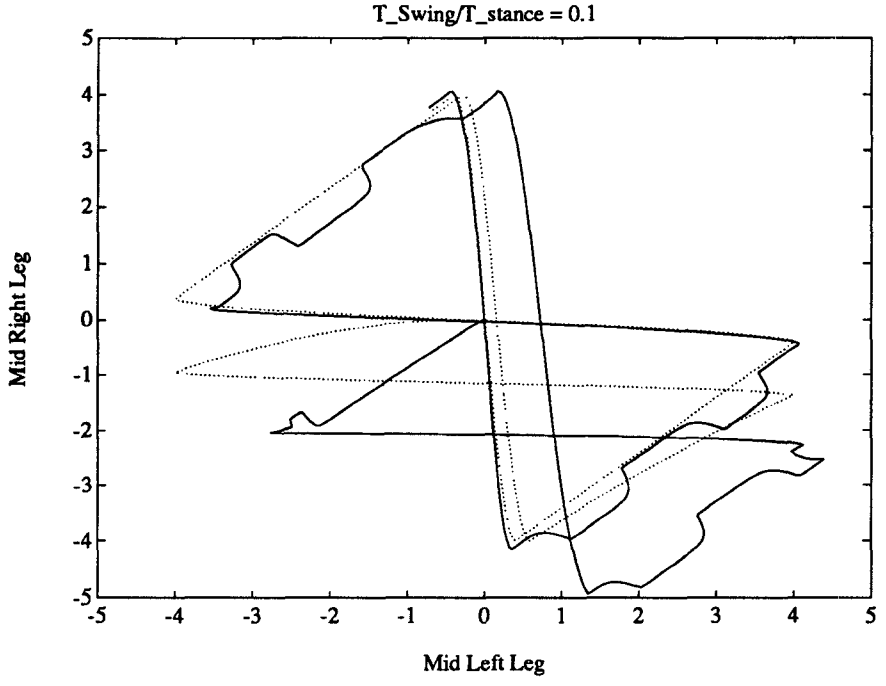


Figure 4.15: Contralateral Phase-Plane Diagram, r = 0.1

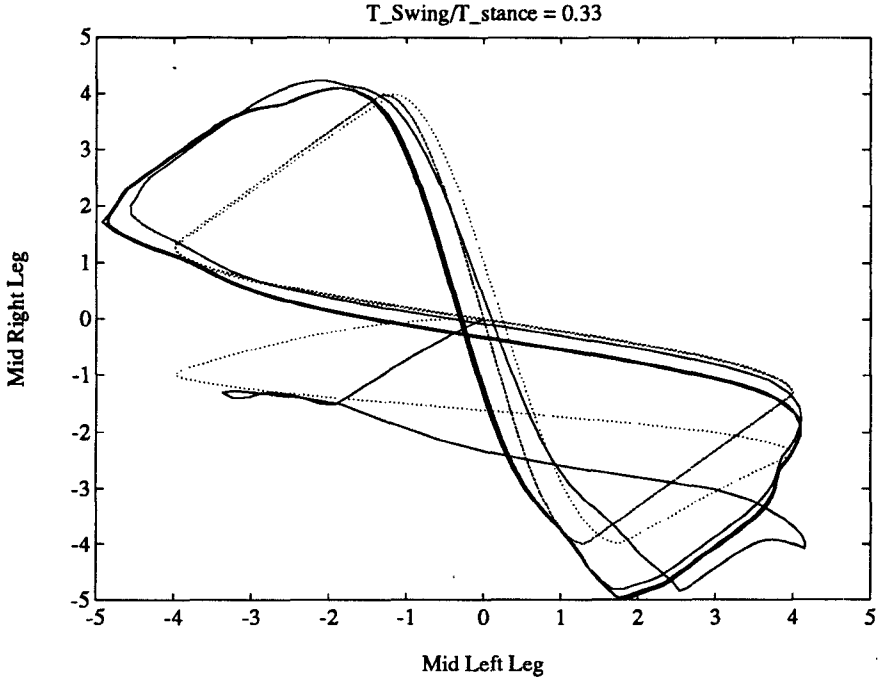


Figure 4.16: Contralateral Phase-Plane Diagram, r = 0.33

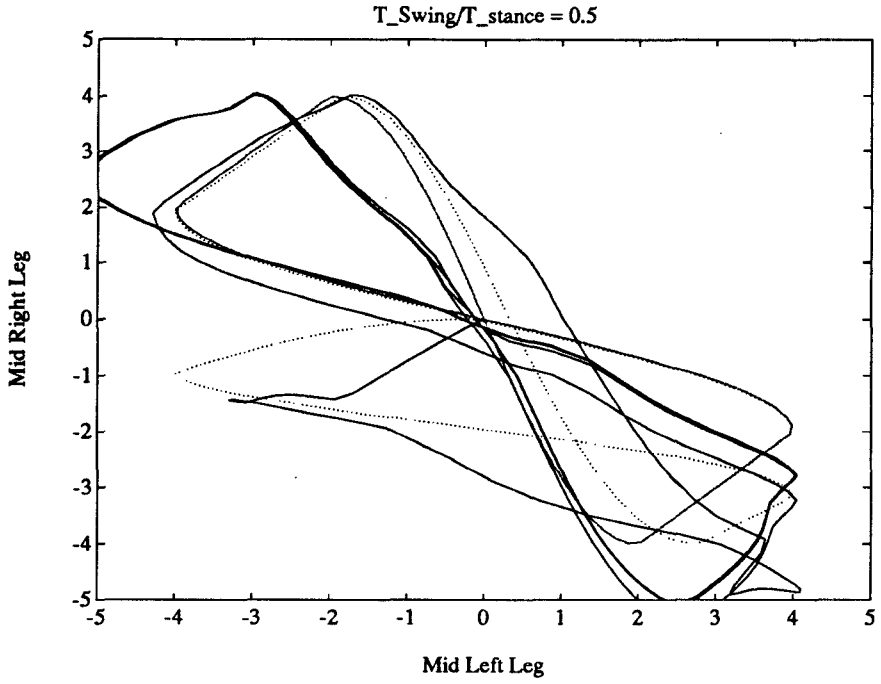


Figure 4.17: Contralateral Phase-Plane Diagram, r = 0.5

When the walking speed is less than $r = \frac{1}{2}$, there are at least four legs on the ground at all times. At speeds higher than $r = \frac{1}{2}$, there are times when only three legs are on the ground. Since the controller was not able to maintain stability at speeds of higher than $r = \frac{1}{2}$, it appears that it was unable to properly control the body with only three legs on the ground.

4.4.4 Summary

The results show that the coordination and control scheme worked well enough to enable walking at speeds where at least four legs remained on the ground at all times. Furthermore, the characteristic gait of the walking stick insect was

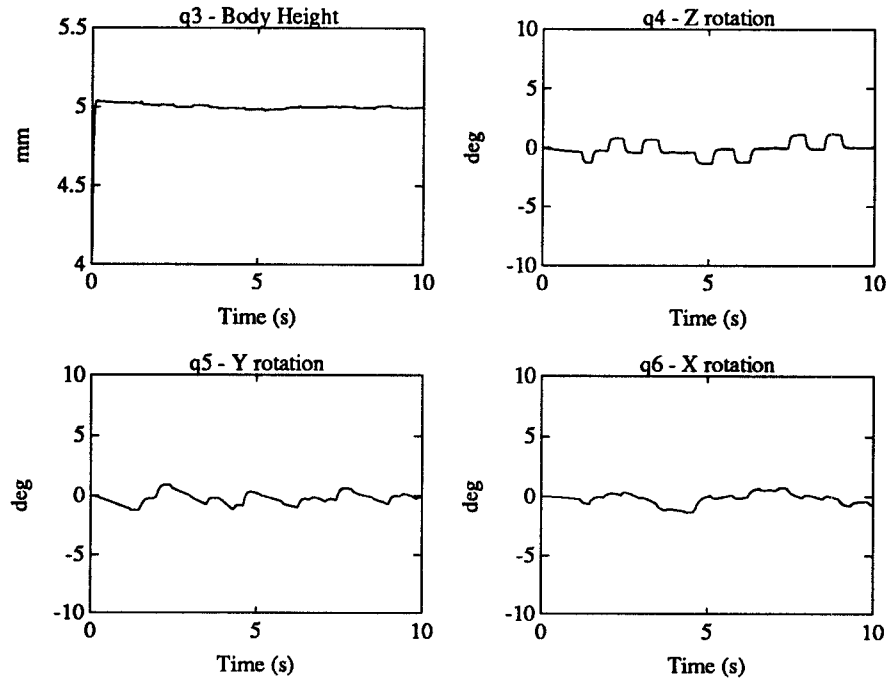


Figure 4.18: Body Movement, r = 0.1

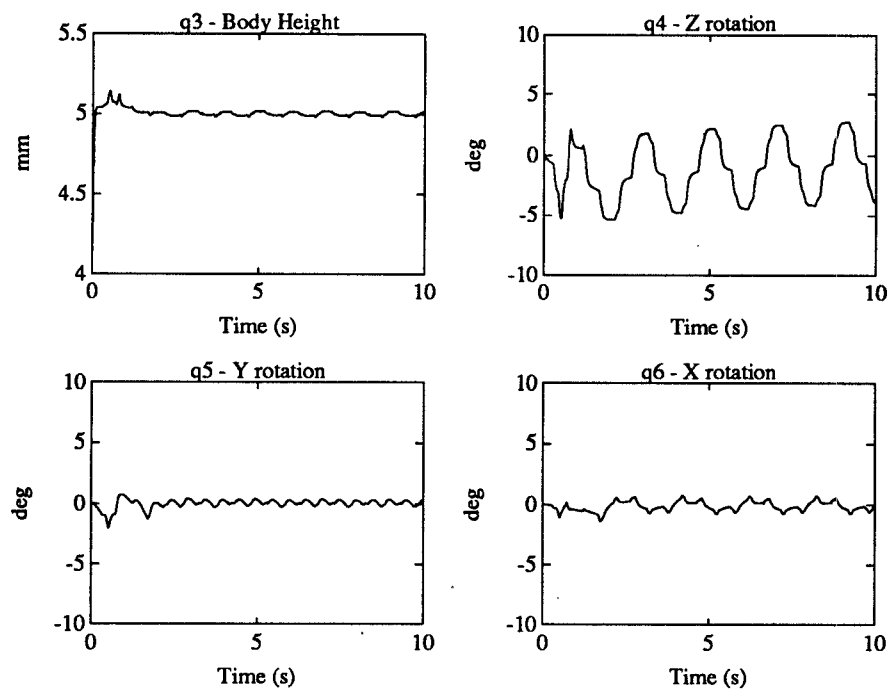


Figure 4.19: Body Movement, r = 0.33

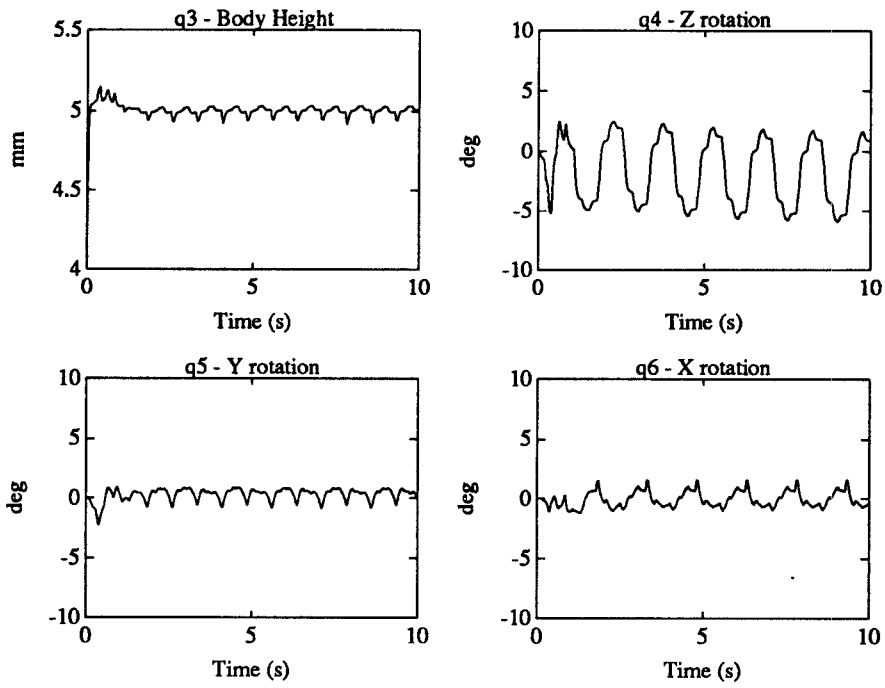


Figure 4.20: Body Movement, r = 0.5

produced. It is believed that with a better control strategy, the faster walking speeds could also be demonstrated.

Chapter 5

Conclusion

5.1 Summary

This thesis covers a great deal of territory in the area of walking machines. First, the coordination systems found in animals was investigated, and a model using nonlinear coupled oscillators was created. Secondly, the dynamics of the walking stick insect were derived, including the ground interaction constraints. Third, a control system was developed which could be used to create the normal gaits of the walking stick insect. Finally, a graphical dynamic simulation of the walking stick insect and its control system was developed to test the design.

While it is not felt that the control system as it is currently proposed would be a useful one in an actual system, it is felt that a useful control system could be developed from a continuation of this work. Furthermore, some basic tools

necessary for such a continuation were developed in terms of a dynamic simulation. Using this simulation tool, more sophisticated and accomplished control systems can be tested and give graphical as well as analytical feedback.

5.2 Recommendations for Future Work

The following are suggestions for future ideas and suggestions for future work in this area.

5.2.1 Numerical Methods

The numerical methods used in this research are probably not the best possible choices. Because of the non-smooth nature of the system whenever a contact condition changes, most common "good" numerical methods do not give good results in these regions. In addition since the computations required are so extensive, methods which require many evaluations are not desirable.

5.2.2 Dynamic Modeling

There are several improvements and/or additions in the area of dynamic modeling which we would like to see worked on in the future.

The dynamic modeling in this work made certain simplifying assumptions

that are not necessarily as close to reality as desired. In particular, the interaction with the ground was modeled as totally inelastic. It would not only be closer to reality, but would also probably be desirable to model this interaction with some degree of elasticity. The best way of accomplishing this would probably be to add a spring element to the model at the bottom of each leg. Modeling elasticity as an impulsive reaction would not be as desirable, since this would cause an unpleasant "hopping" effect.

Another simplification made with ground interaction was to assume no slippage, or infinite ground friction. This assumption should be changed to allow for feet slipping on the ground. This may require a bit more work, since in the current formulation, the ground interaction forces are never explicitly calculated.

Finally, an actuator model should be added. While currently the most commonly used actuator in such applications is the DC motor, we think that it would be interesting and useful to try implementing a muscle-tendon type actuator [11]. Such an actuator system is characterized by high damping in the muscle and a nonlinear spring in the tendon. As a result of the nonlinear nature of the spring, the compliance of the leg could be controlled separately from the position of the leg. Again, because of the nonlinear nature of the spring, one could approximate the force on the leg from the deflection of the leg. This might eliminate the need for direct force measurement or the computation of the gravitational load in the centralized manner done in this research. Such

actuator systems are also starting to become available in the form of air bladder mechanisms with the appropriate rubber tendons [15].

5.2.3 Control Strategies

There are many different coupling strategies that could be implemented besides that considered in this paper. In particular, one could implement a coupling more similar to that proposed in [7]. Instead of changing the velocity of the oscillator as we did in this research, Dean changed the transition between stance and swing to accomplish coordination. It would be fairly simple to implement such a scheme within the framework of this work.

Currently, the system of coupled oscillators is running open loop. Feedback from the environment should be added in order to enable the insect to adjust to obstacles, etc. Some suggestions for possible effects of this feedback can be found in [10], which describes how cats respond to different environmental stimuli.

Appendix A

Link Transformations

From the geometry of the walking stick insect, the link transformations $({}^{i}R_{i-1}, {}^{i}p_{i-1})$ can be specified as follows:

$$
{}^{I}R_{B} = \begin{pmatrix} cq_4cq_5 & cq_4sq_5sq_6 - sq_4cq_6 & cq_4sq_5cq_6 + sq_4sq_6 \\ sq_4cq_5 & sq_4sq_5sq_6 + cq_4cq_6 & sq_4sq_5cq_6 - cq_4sq_6 \\ -sq_5 & cq_5sq_6 & cq_5cq_6 \end{pmatrix}, \quad {}^{I}p_{B} = \begin{pmatrix} q_1 \\ q_2 \\ q_3 \end{pmatrix} \tag{A.1}
$$

$$
{}^{B}R_{0_i} = \begin{pmatrix} 0 & -1 & 0 \\ c\alpha & 0 & s\alpha \\ -s\alpha & 0 & c\alpha \end{pmatrix}, \quad {}^{B}p_{0_i} = \begin{pmatrix} {}^{B}p_{x_i} \\ {}^{B}p_{y_i} \\ 0 \end{pmatrix} \tag{A.2}
$$

$$
{}^{0_i}R_{1_i} = \begin{pmatrix} c\theta_{1_i} & s\theta_{1_i} & 0 \\ -s\theta_{1_i} & c\theta_{1_i} & 0 \\ 0 & 0 & 1 \end{pmatrix}, \quad {}^{0_i}p_{1_i} = \begin{pmatrix} l_1 \\ 0 \\ 0 \end{pmatrix} \tag{A.3}
$$

$$
{}^{1_i}R_{2_i} = \begin{pmatrix} c\theta_{2_i} & s\theta_{2_i} & 0 \\ 0 & 0 & 1 \\ s\theta_{2_i} & -c\theta_{2_i} & 0 \end{pmatrix}, \quad {}^{1_i}p_{2_i} = \begin{pmatrix} l_2 \\ 0 \\ 0 \end{pmatrix} \tag{A.4}
$$

$$
{}^{2_i}R_{3_i} = \begin{pmatrix} c\theta_{3_i} & -s\theta_{3_i} & 0 \\ s\theta_{3_i} & c\theta_{3_i} & 0 \\ 0 & 0 & 1 \end{pmatrix}, \quad {}^{2_i}p_{3_i} = \begin{pmatrix} l_3 \\ 0 \\ 0 \end{pmatrix} \tag{A.5}
$$

Appendix B

MACSYMA Source Code

```
c(x) := cos(x)$
s(x) := sin(x)$

/* Define the vector <=> skew symmetric transformations */
hat(x) := matrix( [    0,    -x[3,1],  x[2,1] ],
                  [  x[3,1],    0,    -x[1,1] ],
                  [ -x[2,1],  x[1,1],    0    ] )$
invhat(x) := matrix( [ -x[2,3] ],
                     [  x[1,3] ],
                     [ -x[1,2] ] )$

/* Define the vectors of coordinates for the main body  */
/*    q1, q2, q3 are x,y,z position                     */
/*    q4, q5, q6 are XYZ Euler angles                   */

depends( q, t )$
q : genvector( q, 24 )$
depends( q_d, t )$
q_d : genvector( q_d, 24 )$
depends( q_d_d, t )$
q_d_d : genvector( q_d_d, 24 )$

/* Define the rotation matrix for the inertial to body frame */

R11(x,y,z) := c(x)*c(y)$
R12(x,y,z) := c(x)*s(y)*s(z)-s(x)*c(z)$
R13(x,y,z) := c(x)*s(y)*c(z)+s(x)*s(z)$
```

```
R21(x,y,z) := s(x)*c(y)$
R22(x,y,z) := s(x)*s(y)*s(z)+c(x)*c(z)$
R23(x,y,z) := s(x)*s(y)*c(z)-c(x)*s(z)$
R31(x,y,z) := -s(y)$
R32(x,y,z) := c(y)*s(z)$
R33(x,y,z) := c(y)*c(z)$

Ri_b : matrix( [ R11(q[4,1],q[5,1],q[6,1]),
                     R12(q[4,1],q[5,1],q[6,1]),
                        R13(q[4,1],q[5,1],q[6,1]) ],
               [ R21(q[4,1],q[5,1],q[6,1]),
                     R22(q[4,1],q[5,1],q[6,1]),
                        R23(q[4,1],q[5,1],q[6,1]) ],
               [ R31(q[4,1],q[5,1],q[6,1]),
                     R32(q[4,1],q[5,1],q[6,1]),
                        R33(q[4,1],q[5,1],q[6,1]) ] )$

Rb_i : transpose( Ri_b )$

/* Calculate angular velocity of the body */

temp : diff( Ri_b, q[4,1] ) * q_d[4,1] +
       diff( Ri_b, q[5,1] ) * q_d[5,1] +
       diff( Ri_b, q[6,1] ) * q_d[6,1]$
wi_b_hat : temp.Rb_i$

wi_b : trigsimp( invhat( wi_b_hat ) )$

/* Define the rotation matrices for all the leg frames and */
/*   calculate all of the associated angular velocities    */

kill( wi_b_hat )$

Rb_0 : matrix( [      0,   -1,    0     ],
               [  c(alpha), 0, s(alpha) ],
               [ -s(alpha), 0, c(alpha) ] )$

R0_b : transpose( Rb_0 )$

R0_1 : matrix( [ c(q[7,1]), s(q[7,1]), 0 ],
               [-s(q[7,1]), c(q[7,1]), 0 ],
               [     0,         0,     1 ] )$
```

```
R1_0 : transpose( R0_1 )$

w0_1_hat : ( diff( R0_1, q[7,1] ) * q_d[7,1] ).R1_0$

w0_1 : trigsimp( invhat( w0_1_hat ) )$

R1_2 : matrix( [ c(q[8,1]), s(q[8,1]), 0 ],
               [     0,         0,     1 ],
               [ s(q[8,1]),-c(q[8,1]), 0 ] )$

R2_1 : transpose( R1_2 )$

w1_2_hat : ( diff( R1_2, q[8,1] ) * q_d[8,1] ).R2_1$

w1_2 : trigsimp( invhat( w1_2_hat ) )$

R2_3 : matrix( [ c(q[9,1]),-s(q[9,1]), 0 ],
               [ s(q[9,1]), c(q[9,1]), 0 ],
               [     0,         0,     1 ] )$

R3_2 : transpose( R2_3 )$

w2_3_hat : ( diff( R2_3, q[9,1] ) * q_d[9,1] ).R3_2$

w2_3 : trigsimp( invhat( w2_3_hat ) )$

kill( w0_1_hat, w1_2_hat, w2_3_hat )$

/* Define all of the vectors from frame origin to frame origin */

pb_1(i) := matrix( [ pb_1x[i] ],
                   [ pb_1y[i] ],
                   [     0    ] )$

p1_2 : matrix( [ l1 ],
               [ 0  ],
               [ 0  ] )$

p2_3 : matrix( [ l2 ],
               [ 0  ],
               [ 0  ] )$

/* Define all vectors from frame origin to center of mass */
```

```
p1_c1 : matrix( [ lc1 ],
                [  0  ],
                [  0  ] )$

p2_c2 : matrix( [ lc2 ],
                [  0  ],
                [  0  ] )$

p3_c3 : matrix( [ lc3 ],
                [  0  ],
                [  0  ] )$

GENTRANLANG : C$
CLINELEN : 65$
GENTRANOPT : TRUE$
TEMPVARTYPE : float$
TEMPVARNAME : o$
TEMPVARNUM : 1$
OPTIMPREFIX : o$
GENTRANSEG : FALSE$
GENFLOAT : TRUE$
NUMER : FALSE$

tmp : genvector( tmp, 1000 )$
count : 0$

save_it( mat, n, m, sym ) := block
(
  [ f, i, j, lower ],
  f : zeromatrix( n, m ),
  for i : 1 thru n do
  (
    if ( sym = TRUE ) then lower : i else lower : 1,
    for j : lower thru m do
    (
      if ( mat[i,j] # 0 ) then
      (
        count : count + 1,
        varlist[count,1] : mat[i,j],
        f[i,j] : tmp[count,1],
        if ( sym = TRUE ) then f[j,i] : tmp[count,1]
      )
```

```
    )
  ),
  f
)$

make_c( vlist, num ) := block
(
  list_opt : optimize( vlist ),
  opt_length : length( list_opt ),

  for i : 1 thru (opt_length-2) do
  (
    gentran( literal( "o", eval(i), "," ), ["dynam2.c"] )
  ),

  gentran( literal( cr ), ["dynam2.c"] ),

  for i : 1 thru (opt_length-2) do
  (
    gentran( literal( "o", eval(i), "=",
             eval( part( part( list_opt, i+1 ), 2 ) ), ";", cr ),
             ["dynam2.c"] )
  ),

  gentran( literal( cr ), ["dynam2.c"] ),

  for i : 1 thru num do
  (
    gentran( literal( "tmp[", eval(i), "] = ",
            eval( part( list_opt, opt_length )[i,1] ), ";", cr ),
             ["dynam2.c"] )
  )
)$

/* Define all of the inertias of the links */

load( matfuncs )$

I_cb : diag_matrix( I_cb[1], I_cb[2], I_cb[3] )$

I_c1 : diag_matrix( 0, I_c1[2], I_c1[3] )$

I_c2 : diag_matrix( 0, I_c2[2], I_c2[3] )$
```

```
I_c3 : diag_matrix( 0, I_c3[2], I_c3[3] )$


/* Calculate the mass matrix */

Calc_Mass( vel, mass, M, vars ) := block
(
  [ i, j, k, temp, newterm ],
  for i : 1 thru length( vars ) do
  (
    temp[i] : diff( vel, q_d[vars[i],1] )
  ),
  print( "Done with derivatives" ),
  for i : 1 thru length( vars ) do
  (
    print( "Starting pass", i ),
    for j : i thru length( vars ) do
    (
      if length( mass ) = 3 then
      (
        newterm : expand( transpose( temp[i] ).mass.temp[j] )
      )
      else
      (
        newterm : expand( mass * transpose( temp[i] ).temp[j] )
      ),
      for k : 1 thru length( vars ) do
      (
        newterm : ratsubst(1-cos(q[vars[k],1])*cos(q[vars[k],1]),
                  sin(q[vars[k],1])*sin(q[vars[k],1]), newterm )
      ),
      newterm : trigsimp( ratsubst( 1 - cos(alpha)*cos(alpha),
                          sin(alpha)*sin(alpha), newterm ) ),
      M[vars[i],vars[j]] : M[vars[i],vars[j]] + newterm,
      M[vars[j],vars[i]] : M[vars[i],vars[j]],
      print( "  Done with subpass", j )
    )
  ),
  M
)$

Calc_V( vel, mass, V, vars ) := block
```

```
(
  [ i, j, len, dv_dq_d, dv_dq, temp, temp2, D, E ],
  len : length( vars ),
  print( "Starting stage 1" ),
  dv_dq_d : zeromatrix( 3, len ),
  for i : 1 thru len do
  (
    for j : 1 thru 3 do
    (
      dv_dq_d[j,i] : diff( vel[j,1], q_d[vars[i],1] )
    ),
    print( "Done with substage", i )
  ),
  print( "Starting stage 2" ),
  dv_dq : zeromatrix( 3, len ),
  for i : 1 thru len do
  (
    for j : 1 thru 3 do
    (
      dv_dq[j,i] : diff( vel[j,1], q[vars[i],1] )
    ),
    print( "Done with substage", i )
  ),
  print( "Starting stage 3" ),
  temp : zeromatrix( len, 3 ),
  for i : 1 thru len do
  (
    temp : temp + diff( transpose( dv_dq_d ), q[vars[i],1] ) *
                  q_d[vars[i],1],
    print( "Done with substage", i )
  ),
  if length( mass ) = 3 then
  (
    D : transpose( dv_dq_d ) . mass . dv_dq,
    E : temp . mass . vel
  )
  else
  (
    D : mass * transpose( dv_dq_d ) . dv_dq,
    E : mass * temp . vel
  ),
  print( "Starting stage 4" ),
  temp2 : zeromatrix( len, 1 ),
```

```
  for i : 1 thru len do
  (
    for j : 1 thru len do
    (
      temp2[i,1] : temp2[i,1] + ( D[i,j] - D[j,i] ) * q_d[j,1]
    ),
    print( "Done with substage", i )
  ),
  for i : 1 thru len do
  (
    V[vars[i],1] : V[vars[i],1] + temp2[i,1] + E[i,1]
  ),
  V
)$

M : zeromatrix( 24, 24 )$
V : zeromatrix( 24, 1 )$

omega_b : trigsimp( Rb_i.wi_b )$

vi_b : matrix( [ q_d[1,1] ],
               [ q_d[2,1] ],
               [ q_d[3,1] ] )$

v_b : trigsimp( Rb_i.vi_b )$
v_cb : v_b$

M : Calc_Mass( v_cb, m_b, M, [ 1, 2, 3, 4, 5, 6 ] )$
M : Calc_Mass( omega_b, I_cb, M, [ 4, 5, 6 ] )$

V : Calc_V( v_cb, m_b, V, [ 1, 2, 3, 4, 5, 6 ] )$
V : Calc_V( omega_b, I_cb, V, [ 4, 5, 6 ] )$

MM : save_it( M, 24, 24, TRUE )$
VV : save_it( V, 24, 1, FALSE )$

varlist : genmatrix( varlist, count, 1 )$
make_c( varlist, count )$

gentran( M : eval( MM ), ["dynam2.c"] )$
gentran( V : eval( VV ), ["dynam2.c"] )$

kill( varlist, M, MM, VV, V )$
```

```
count : 0$
M : zeromatrix( 24, 24 )$
MM : zeromatrix( 24, 24 )$
V : zeromatrix( 24, 1 )$
VV : zeromatrix( 24, 1 )$

/* Calculate all of the angular velocities with respect to the */
/*   inertial frame for all of the link frames                 */

omega_1 : trigsimp( R1_0.( R0_b.omega_b + w0_1 ) )$

omega_2 : trigsimp( R2_1.( omega_1 + w1_2 ) )$

omega_3 : trigsimp( R3_2.( omega_2 + w2_3 ) )$

print( "Done with omega's" )$

/* Calculate all of the linear velocities of the links */

v_1 : trigsimp( R1_0.R0_b.( v_b + hat( omega_b ).pb_1(1000) ) )$

v_2 : trigsimp( R2_1.( v_1 + hat( omega_1 ).p1_2 ) )$

v_3 : trigsimp( R3_2.( v_2 + hat( omega_2 ).p2_3 ) )$

print( "Done with v's" )$

/* Calculate all the linear velocities of the */
/* center of mass of the links                */

v_c1 : trigsimp( v_1 + hat( omega_1 ).p1_c1 )$

v_c2 : trigsimp( v_2 + hat( omega_2 ).p2_c2 )$

v_c3 : trigsimp( v_3 + hat( omega_3 ).p3_c3 )$

print( "Done with v_c's" )$

M : Calc_Mass( v_c1, m_1, M, [ 1, 2, 3, 4, 5, 6, 7 ] )$
MM : MM + save_it( M, 24, 24, TRUE )$
kill(M)$
M : zeromatrix( 24, 24 )$
```

```
M : Calc_Mass( omega_1, I_c1, M, [ 4, 5, 6, 7 ] )$
MM : MM + save_it( M, 24, 24, TRUE )$
kill(M)$
M : zeromatrix( 24, 24 )$

M : Calc_Mass( v_c2, m_2, M, [ 1, 2, 3, 4, 5, 6, 7, 8 ] )$
MM : MM + save_it( M, 24, 24, TRUE )$
kill(M)$
M : zeromatrix( 24, 24 )$

M : Calc_Mass( omega_2, I_c2, M, [ 4, 5, 6, 7, 8 ] )$
MM : MM + save_it( M, 24, 24, TRUE )$
kill(M)$
M : zeromatrix( 24, 24 )$

M : Calc_Mass( v_c3, m_3, M, [ 1, 2, 3, 4, 5, 6, 7, 8, 9 ] )$
MM : MM + save_it( M, 24, 24, TRUE )$
kill(M)$
M : zeromatrix( 24, 24 )$

M : Calc_Mass( omega_3, I_c3, M, [ 4, 5, 6, 7, 8, 9 ] )$
MM : MM + save_it( M, 24, 24, TRUE )$
kill(M)$
M : zeromatrix( 24, 24 )$

V : Calc_V( v_c1, m_1, V, [ 1, 2, 3, 4, 5, 6, 7 ] )$
VV : VV + save_it( V, 24, 1, FALSE )$
kill(V)$
V : zeromatrix( 24, 1 )$

V : Calc_V( omega_1, I_c1, V, [ 4, 5, 6, 7 ] )$
VV : VV + save_it( V, 24, 1, FALSE )$
kill(V)$
V : zeromatrix( 24, 1 )$

V : Calc_V( v_c2, m_2, V, [ 1, 2, 3, 4, 5, 6, 7, 8 ] )$
VV : VV + save_it( V, 24, 1, FALSE )$
kill(V)$
V : zeromatrix( 24, 1 )$

V : Calc_V( omega_2, I_c2, V, [ 4, 5, 6, 7, 8 ] )$
VV : VV + save_it( V, 24, 1, FALSE )$
kill(V)$
```

```
V : zeromatrix( 24, 1 )$

V : Calc_V( v_c3, m_3, V, [ 1, 2, 3, 4, 5, 6, 7, 8, 9 ] )$
VV : VV + save_it( V, 24, 1, FALSE )$
kill(V)$
V : zeromatrix( 24, 1 )$

V : Calc_V( omega_3, I_c3, V, [ 4, 5, 6, 7, 8, 9 ] )$
VV : VV + save_it( V, 24, 1, FALSE )$
kill(V)$
V : zeromatrix( 24, 1 )$

varlist : genmatrix( varlist, count, 1 )$

make_c( varlist, count )$

gentran( M : eval( MM ), ["dynam2.c"] )$
gentran( V : eval( VV ), ["dynam2.c"] )$
```

Bibliography

[1] A.H.Cohen and P. Wallen. The neuronal correlate of locomotion in fish. *Experimental Brain Research*, 41:11–18, 1980.

[2] R. McNeill Alexander. *Animal Mechanics.* University of Washington Press, 1968.

[3] A.H. Cohen. Effects of oscillator frequency on phase-locking in the lamprey central pattern generator. *Journal of Neuroscience Methods*, 21:113–125, 1987.

[4] A.H. Cohen. Studies of the lamprey central pattern generator for locomotion: A close relationship between models and experimentation. In M.F. Shlesinger J.A.S. Kelso, A.J. Mandell, editor, *Dynamic Patterns in Complex Systems*, pages 143–161. World Scientific, 1988.

[5] A.H. Cohen, P.J. Holmes, and R.H. Rand. The nature of the coupling between segmental oscillators of the lamprey spinal generator for locomotion: A mathematical model. *Journal of Mathematical Biology*, 13:345–369, 1982.

[6] John J. Craig. *Introduction to Robotics, Mechanics and Control.* Addison-Wesley, second edition, 1989.

[7] J. Dean. A model of leg coordination in the stick insect, carausius morosus, i. *Biological Cybernetics*, 64:393–402, 1991.

[8] J. Dean. A model of leg coordination in the stick insect, carausius morosus, ii. *Biological Cybernetics*, 64:403–411, 1991.

[9] Marc D. Donner. *Real-Time Control of Walking.* Birkhäuser, 1987.

[10] H. Forssberg. *On Integrative Motor Functions in the Cat's Spinal Cord.* PhD thesis, Karolinska Institutet, Stockholm, 1979.

[11] J. He, W.S. Levine, and G.E. Loeb. Feedback gains for correcting small pertubations to standing posture. *IEEE Transactions on Automatic Control*, 36(3):322–332, March 1991.

[12] H.Hemami and F.C. Weimer. Modeling of nonholonomic dynamic systems with applications. *Journal of Applied Mechanics*, 48, March 1981.

[13] N. Kopell and G.B. Ermentrout. Symmetry and phaselocking in chains of weakly coupled oscillators. *Communications on Pure and Applied Mathematics*, 39:623–660, 1986.

[14] N. Kopell and G.B. Ermentrout. Coupled oscillators and the design of central pattern generators. *Mathematical Biosciences*, 90:87–109, 1988.

[15] J.K. Mills. Hybrid actuator for robot manipulators: Design, control and performace. *Proceedings of the IEEE Conference on Robotics and Automation*, pages 1872–1878, 1990.

[16] F. Pfeiffer, H.J. Weidermann, and P. Danowski. Dynamics of the walking stick insect. *Proceedings of the IEEE Conference on Robotics and Automation*, pages 1458–1463, 1990.

[17] Marc H. Raibert. *Legged Robots that Balance.* The MIT Press, 1986.

[18] R.H. Rand, A.H Cohen, and P.J. Holmes. Systems of coupled oscillators as models of central pattern generators. In A.H. Cohen, editor, *Neural Control of Rhythmic Movements in Vertibrates*, chapter 9, pages 333–367. Wiley, New York, 1988.

[19] R.H.Byrne. Interactive graphics and dynamical simulation in a distributed processing environment. Master's thesis, University of Maryland, 1990.

[20] Gilbert Strang. *Linear Algebra and Its Applications.* Academic Press, 1980.

[21] A. Teolis. A graphical simulation management system. Technical Report TR 91–75, Systems Research Center, University of Maryland, 1991.

[22] D.J. Todd. *Walking Machines, An Introduction to Legged Robots.* Chapman and Hall, 1985.