

# **Analysis, Design, and Control of Hopping Robot**

Hamidreza Dokht Taghirad

Department of Mechanical Engineering  
McGill University

Master of Engineering  
Project

April 1, 1993

Copy 1

# **Analysis, Design, and Control of Hopping Robot**

by

**Hamidreza Dokht Taghirad**

B.S. (Sharif University of Technology), 1989

Department of Mechanical Engineering

McGill University, Montreal

Canada

A project report submitted to the Faculty of Graduate Studies and Research  
in partial fulfillment of the requirements for the degree of  
Masters of Engineering

April 1, 1993

© Hamidreza Dokht Taghirad

# Abstract

The control of dynamically stable legged locomotion has made great strides in the past decade. Usually powerful hydraulic actuators are harnessed for control of the legged robot. In this report, we focus on a planar one-legged hopper, which is electrically actuated. This type of actuation technology is clean, safe and cheap, suitable for indoor use and autonomous robots. However, because DC motors have much smaller torque/mass ratio than hydraulic actuators, special consideration must be paid to the modelling, design and control of the hopper.

For the vertical hopper we derived a dynamical model including actuator dynamics, and developed different control strategies based on the total energy of the system. These strategies are verified by simulation. For planar motion we used the passive dynamics concept to model the hopper. This modelling leads to an energy efficient system, and might simplify the control.

A one-legged hopper was designed and constructed, and some of the control algorithms were implemented. Using the experimental results, the estimated parameters in the simulations were modified and the simplified model-based control was verified.

# Résumé

La commande d'un robot sauteur unijambiste dynamiquement stable a été l'objet de progrès considérables au cours des dix dernières années. Habituellement, de puissants actuateurs hydrauliques sont utilisés pour la commande du robot unijambiste. Dans ce rapport, nous nous concentrons sur un robot sauteur unijambiste planaire qui est manoeuvré électriquement. Ce type de manoeuvre est propre, sécuritaire et est une technologie peu couteuse. Elle est appropriée à l'utilisation intérieure et aux robots autonomes. Parcontre, à cause des limitations de puissance disponible, une attention toute particulière doit être portée sur la modélisation, la conception et la commande du robot sauteur.

Pour le sauteur vertical nous avons dérivé un modèle dynamique incluant la dynamique des actuateurs et développé des stratégies de commande différentes, basées sur l'énergie totale du système. Ces stratégies sont vérifiées par simulation. Pour un déplacement planaire, nous avons utilisé le concept des dynamiques passives afin de modéliser le sauteur. Cette modélisation résulte en un system énergiquement efficace et simplifie la commande.

Un sauteur vertical unijambiste est conçu, construit, et quelques algorithmes de commande y sont réalisés. Les paramètres estimés lors de la simulation furent modifiés à partir des résultats expérimentaux, et la commande à base de modèle simplifiée fut vérifiée.

# Acknowledgements

I would like to extend my sincere thanks and gratitude to my advisor, Professor Martin Buehler, for his assistance and continuous guidance during my research. His invaluable comments, and suggestions, motivated me to improve the quality of this research.

I am indebted to the graduate students in my group for their faithful comments. Special thanks to my fellow graduate student Majid Moghaddam, who unconditionally helped me in design, construction, and modification of the prototype presented in this report, Pedro Gregorio who continues this research, and Gerard Blais for translating the abstract into French.

The support I received from the Department of Mechanical Engineering, and McGill Research Center for Intelligent Machines is truly acknowledged. I am grateful to the machine shop staff of the Department of Mechanical Engineering, for their valuable suggestions to complete the final design and machining of the parts of the prototype presented in this report. This research has been supported in part by a Major Equipment Grant from the McGill Graduate Faculty, an Equipment Grant from the McGill Faculty of Graduate Studies and Research, and an NSERC/CIAR Junior Industrial Research Chair. The researcher grant was provided by the Ministry of Culture and Higher Education of Islamic Republic of Iran.

I thank God for his silent support in all the moments of accomplishing this research.

**To Azam**

# Contents

<b>Abstract</b>	<b>iii</b>
<b>Résumé</b>	<b>iv</b>
<b>Acknowledgements</b>	<b>v</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Why Study Legged Robots? . . . . .	2
1.2 Background . . . . .	2
1.3 Contributions of This Research . . . . .	4
<b>2 Analysis and Simulation</b>	<b>5</b>
2.1 Vertical Hopping Dynamics . . . . .	5
2.1.1 Modelling . . . . .	5
2.1.2 Dynamics . . . . .	7
2.1.3 Control and Simulation . . . . .	13
2.2 Planar Passive Dynamics . . . . .	23
2.2.1 Modelling . . . . .	24
2.2.2 Dynamics . . . . .	26
2.2.3 Simulation . . . . .	31
2.2.4 Active Stability . . . . .	32
<b>3 Design</b>	<b>35</b>

3.1	Actuator Design . . . . .	36
3.2	Spring Design . . . . .	38
3.3	Leg Design . . . . .	43
<b>4</b>	<b>Experiment</b>	<b>45</b>
4.1	Hardware . . . . .	45
4.2	Software . . . . .	46
4.3	DC Motor Control . . . . .	48
4.4	Hopping Experiments . . . . .	53
4.4.1	Passive Hopping . . . . .	53
4.4.2	Open Loop Torque Control . . . . .	54
4.4.3	Modified Torque Control . . . . .	56
4.5	Simulation Verification . . . . .	57
4.5.1	Passive Hopping . . . . .	58
4.5.2	Actuated Hopping . . . . .	60
4.6	Design Modification . . . . .	61
<b>5</b>	<b>Conclusion and Future Work</b>	<b>64</b>
5.1	Conclusion . . . . .	64
5.2	Future Work . . . . .	65
	<b>Appendices</b>	<b>70</b>
<b>A</b>	<b>General Planar Dynamics</b>	<b>70</b>

# List of Tables

2.1	<i>Parameters of vertical hopper model . . . . .</i>	14
2.2	<i>Parameters of one-legged passive model . . . . .</i>	32
3.1	<i>Simulation comparison of different motors, regarding <math>k=10</math>, <math>r=5</math> . . . . .</i>	37
3.2	<i>Spring materials tensile strength . . . . .</i>	41
3.3	<i>Final spring design parameters . . . . .</i>	42

# List of Figures

2.1	<i>Vertical Hopper model in stance and flight phases</i>	6
2.2	<i>Vertical Hopper and actuator model during flight</i>	7
2.3	<i>Vertical Hopper and actuator model during stance</i>	10
2.4	<i>Open loop position step; State variables</i>	16
2.5	<i>Open loop position step; Phase plot and total energy</i>	16
2.6	<i>Closed loop position step; State variables</i>	18
2.7	<i>Closed loop step; Phase plot and total energy</i>	18
2.8	<i>Open loop torque control; State variables</i>	19
2.9	<i>Open loop torque control; Phase plot and total energy</i>	19
2.10	<i>Closed loop torque control; State variables</i>	20
2.11	<i>Closed loop torque control; Phase plot and total energy</i>	20
2.12	<i>Actuator velocity control; State variables</i>	22
2.13	<i>Actuator velocity control; Phase plot and total energy</i>	22
2.14	<i>Planer one-legged model</i>	24
2.15	<i>Passive trajectory of the robot</i>	25
2.16	<i>Ground forces during stance phase</i>	27
2.17	<i>Harmonic oscillation of the hip</i>	29
2.18	<i>Passive dynamic running simulation of planner one-legged hopper for running speed 2m/s</i>	30
2.19	<i>Passive dynamic running simulation of planner one-legged hopper for running speed 4m/s</i>	31

2.20	<i>Phase plot; Pitch angle versus hopping height for horizontal speed of 4m/s . . .</i>	32
2.21	<i>Simulation results for controller with <math>K_c = 1000</math> and horizontal velocity 4m/s</i>	33
2.22	<i>Phase plot of controller with <math>K_c = 1000</math> and horizontal velocity 4m/s . . . . .</i>	33
3.1	<i>Simulation result for ball screw verification; <math>k=10</math> KN/m, Lead=2 . . . . .</i>	39
3.2	<i>Simulation result for ball screw verification; <math>k=10</math> KN/m, Lead=5 . . . . .</i>	39
3.3	<i>Simulation result for ball screw verification; <math>k=10</math> KN/m, Lead=10 . . . . .</i>	39
3.4	<i>The result of the Spring test; * denotes compression, and + denotes decom- pression readings . . . . .</i>	42
3.5	<i>The final leg drawing (left) and the picture of McGill Hopper (right) . . . . .</i>	44
4.1	<i>Block diagram of the system hardware . . . . .</i>	46
4.2	<i>Block diagram of the system software . . . . .</i>	47
4.3	<i>Block diagram of motor controller . . . . .</i>	48
4.4	<i>Step Response comparison of theory and experiment . . . . .</i>	50
4.5	<i>Step Response comparison of theory without current limit and experiment with current limit . . . . .</i>	52
4.6	<i>Step response of the system for a large initial error . . . . .</i>	52
4.7	<i>Hopping height, total energy, and stance energy of Passive Hopper . . . . .</i>	54
4.8	<i>Hopping height, actuator length, total energy, and stance energy of open loop torque control . . . . .</i>	55
4.9	<i>Hopping height, actuator length, total energy, and stance energy of modified torque control . . . . .</i>	57
4.10	<i>Comparison of simulation and experiment; Passive hopping . . . . .</i>	58
4.11	<i>Simulation and Experiment result during the first stance; Passive hopping . .</i>	59
4.12	<i>Comparison of simulation and experiment; Actuated case . . . . .</i>	60
4.13	<i>Simulation and Experiment result during the first stance; Actuated case . . .</i>	61
4.14	<i>Stable hopping experiment, starting from large hopping height . . . . .</i>	62
4.15	<i>Stable hopping experiment, starting from small hopping height . . . . .</i>	62
A.1	<i>The general planar running robot model . . . . .</i>	71

# Chapter 1

## Introduction

In this report we deal with a group of robots named *Legged Robots*. The research on this subject has a long history; however, only the last decades saw some successful experimental prototypes. In this chapter we explain why this research is important, and what benefits it could present to humans. Then we review a short history of the research done in this subject. Afterwards we will discuss the introductory concepts to develop the idea, and the contributions of this research.

In chapter two we address the analysis and modelling of a planar one-legged robot. Two different ways of modelling the system will be studied and the simulation results will be discussed. In chapter three we speak about the design of the leg and the modular artificial muscle system, the analysis to identify the system parameters, and the dimension and stress analysis of the parts will be explained. Chapter four is about the experimental results. The overall scheme of the hardware and software which is used in the experiments will be discussed first, and then the methods of experimentation and the results will be studied. At the end a comparison between the experimental and analytical results will be made, and an approach to match these result will be explained. Then the results of the simplified model will be verified with the experimental results. The conclusion and suggestions for further work will be given in the last chapter.

## 1.1 Why Study Legged Robots?

This report is about the machines that use legs to run. The purpose of these machines is to study the principles of legged locomotion. Such principles can help us to understand animal locomotion and to build useful legged vehicles.

There are three main reasons for exploring the use of legs for locomotion. First reason is the study of the mobility in difficult terrain. Today's vehicles use wheels to move, and wheels can provide motion only on prepared surfaces, such as rails and roads. However, most of places have not been paved. It seems possible to build vehicles which use legs like animals for locomotion.

The second reason to study legged machines is to understand human and animal locomotion. The principles of control which is used in human and animal locomotion is still not understood. One way to learn more about plausible mechanisms for animal locomotion is to build machines that locomote using legs [21].

The third reason which motivated the study of legged locomotion is the need to build artificial legs for amputees. For below-knee amputee and above-knee prostheses some practical feet have been built, but there is still a long way to find appropriate mechanisms which can be substitute the real organs [14].

## 1.2 Background

Before introducing the main topic, we turn briefly an account of previous work on legged machines. The scientific study of legged locomotion began just over a century ago when Muybridge studied the trotting motion of a horse. His photographic data are still of considerable value and survive as a landmark in locomotion research [18].

During the ninety years that followed, the field viewed that of building walking machines as the task of designing kinematic linkages that would generate suitable stepping motion. However, by the late 1950s it had become clear that a linkage providing fixed motion would not do the trick of walking or running, and useful walking machines would need control.

One approach to control was to harness human. Mosher used this approach in building a four legged walking truck [17]. Another approach is the use of digital computer for control. McGhee's group was the first to use this approach successfully [11]. They built an insect-like hexaped that could walk with a number of standard gaits, turn, crab, and negotiate simple obstacles.

Gurfinkel and his group built a machine quite similar to McGhee's, which used analog computer (hybrid) for control [4]. Hirose designed clever and unusual mechanisms to simplify the control of locomotion and improve their efficiency [5]. McGhee, Gurfinkel, and Hirose's walking machine groups represent a class called *Static Crawlers*. Several other machines that fall into this class have been studied in the intervening years.

Another class turned to the study of dynamic machines that balance actively. Shannon was probably the first to build a machine that balanced an inverted pendulum in top of a small powered truck. This study forwarded by his students to demonstrate controllers for two pendulums at once, and finally the case that two pendulums were mounted on top of each other. Later, they extended these techniques to provide balance for a flexible inverted pendulum [24]. Miura and Shimoyama[15, 16] built the first walking machine that really balanced actively. The control of their biped relied on an inverted pendulum model. Matsuoka was the first to build a machine that was able to hop on one leg.

The field of dynamically stable legged locomotion has made great strides in the past decade, led primarily by Marc Raibert [21]. He built a variety of running robots, starting with a planar one-legged machine [20], followed by a 3D one-legged, a two-legged planar robot, and a four-legged quadruped. His latest robots include a 3D two-legged robot, where each leg has four actuated degrees of freedom. Except for the very first one legged planar hopper, which was pneumatically actuated, his subsequent designs are actuated by powerful hydraulic actuators and rely on pneumatics for the leg spring only [21]. Papantoniou designed an electrically powered planar robot, capable of operating at maximum speed of 0.3 m/s. In order to obtain that, an original mechanical design of an articulated leg and leg attitude control has been realised [19].

### 1.3 Contributions of This Research

We want to analyse again the legged robots, where Raibert has started a few years ago. His research clearly shows that this area is a platform for new ideas. The variety of designs and controller could be examined, and simplified model could be justified with experiment. Generalizing simple controller for more complex robots is another feature of these research. We decided to start with simple prototypes, but with the image of generalizing the results to a more complex one. Therefore, we studied a one-legged running robot with elastic prismatic leg. We focus on electrical actuation, instead of hydraulics, as a clean, safe and cheap technology, suitable for indoor use and autonomous robots.

There are three main contributions of the present research. First, we present a new robot leg design for a vertical hopper actuated by a low power electric motor. The leg is a modular component which in fact is part of a planar robot, and which can be used for 3D and multi-legged robots. Second, we derive carefully a dynamical model for our robot which we validate by comparing simulations to experiments. This model is to our knowledge the first one for running or hopping robots which includes actuator dynamics. For robots with electrical actuators, these can no longer be neglected. Third, we developed different continuous control strategies, which stabilize the hopper, even with limited motor power available.

# Chapter 2

## Analysis and Simulation

### 2.1 Vertical Hopping Dynamics

In synthesizing control system for legged machines, there are many design alternatives [9]. Several functions must be performed, and each function can be accomplished in several different ways. For feasibility of experimentation, we first analyse the possibilities and then determine which alternatives will be implementable.

There exist some control algorithms for vertical hopper based on step actuator position input during stance phase [21]. We are looking for a stable control algorithm which is based on a *continuous* input, and we want to implement it on a one-legged hopping robot.

#### 2.1.1 Modelling

Figure (2.1) illustrates the model of the vertical hopper used for analysis and simulation. The essential features of the model are a body of mass  $M$ , a compliant leg of mass  $m$ , and stiffness  $K$  with damping  $C$ . The total length of the leg is fixed; however, the initial preload of the spring can be adjusted. The actuator, which is a position source operating in series with the spring, consists of a DC motor and a ball screw transmission with lead  $r$ . The actuator and spring act together to lengthen and shorten the leg, and to exert forces between

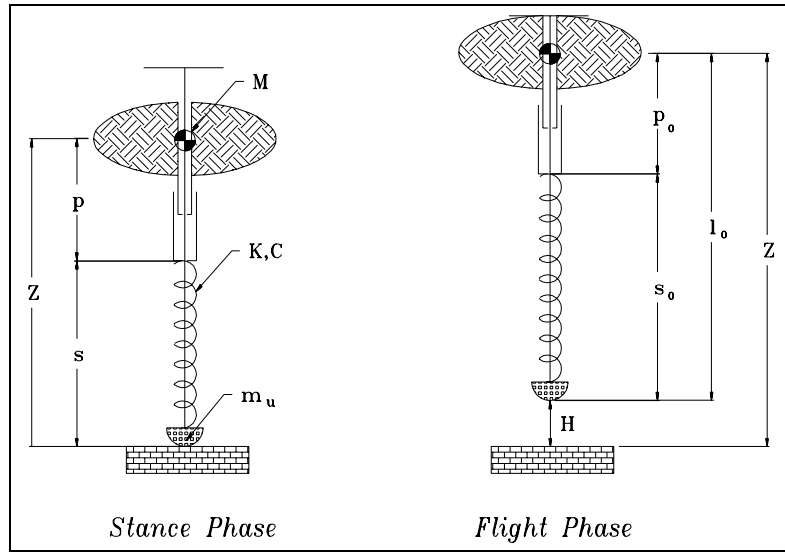
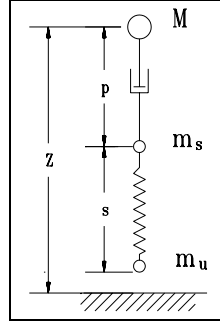


Figure 2.1: *Vertical Hopper model in stance and flight phases*

the foot and the body.

The leg spring absorbs energy by shortening under load of the body and returns energy by lengthening, accelerating the body upward. A mechanical stop prevents the leg from extending beyond a fixed length. The unsprung mass  $m_u$  represents that portion of the leg that is functioning below the spring, the rest being included in body mass  $M$ .

When the actuator changes length, it does work on the leg spring to increase or decrease its stored energy. This arrangement of actuator, leg, spring, and mechanical stop permits the model to hop. During stance phase the actuator excites the spring-mass system. As the leg reaches maximum length, the mechanical stop permits a fraction of kinetic energy to transfer from the body to the leg, enabling the foot to leave the ground. There are two sources of energy loss: impact of unsprung mass to the ground which we assume to be perfectly inelastic, and friction which is modelled as a dry friction  $F_{fr}$ , and viscous friction. The idea of maintaining the desired hopping height of the system is to measure the energy in the vertical motion during stance phase, and to control the leg actuation to inject energy to the system to reach to the desired hopping height.

Figure 2.2: *Vertical Hopper and actuator model during flight*

## 2.1.2 Dynamics

The motion is divided into two phases, flight and stance phase, and two transients, touch down and lift off. In modelling we consider the dynamics of the actuator. Suppose the actuator consist of DC motor and ball screw transmission, where the rotating parts have a moment of inertia  $J$  and sliding mass is represented by  $m_s$ . The ball screw transmission ratio is  $r$ . The input to the system is the motor torque  $\tau$ . Moreover, we suppose the motor and ball screw transmission have an overall efficiency  $\eta$ . For the modelling we use the lagrange approach. The Lagrange formulation can be written as

$$\frac{d}{dt} \left( \frac{\partial \mathbf{L}}{\partial \dot{q}_j} \right) - \frac{\partial \mathbf{L}}{\partial q_j} = \mathbf{Q}_j \quad (2.1)$$

Where Lagrangian,  $\mathbf{L}$  is subtraction of potential energy from kinetic energy, namely  $\mathbf{L} = \mathbf{T} - \mathbf{V}$ ,  $q_j$  is the generalized coordinate, which for our system is  $z$  and  $p$ , and the  $\mathbf{Q}_j$  is the generalized nonconservative force, which could be derived using virtual work principle.

### 2.1.2.1 Flight Phase

Figure (2.2) represents the flight model. During flight phase we can write the kinetic and potential energies as follows:

$$\mathbf{T} = \frac{1}{2} J \left( \frac{\dot{p}}{r} \right)^2 + \frac{1}{2} (M + m_u) \dot{z}^2 + \frac{1}{2} m_s (\dot{z} - \dot{p})^2 \quad (2.2)$$

$$\mathbf{V} = Mgz + m_s g(z - p) + m_u g(z - l_0) + \frac{1}{2} k (s_0 - l_0 + p)^2 \quad (2.3)$$

where  $p$  is the downward linear motion of the actuator, and is related to the DC motor angular position  $\theta$  with:

$$\theta = \frac{p}{r} \quad (2.4)$$

therefore, Lagrangian  $\mathbf{L}$  is as follows:

$$\mathbf{L} = \frac{1}{2}\alpha\dot{p}^2 + \frac{1}{2}(m_t)z^2 - m_s\dot{z}\dot{p} - m_tgz + m_sgp + m_u gl_0 - \frac{1}{2}k(s_0 - l_0 + p)^2 \quad (2.5)$$

where  $m_t$  is the total mass, and

$$\begin{aligned} m_t &= M + m_s + m_u \\ \alpha &= \frac{J}{r^2} + m_s \end{aligned} \quad (2.6)$$

Before taking derivatives of Equation (2.5) let's derive the generalized forces. Assume the friction forces are divided into a viscous damping due to the spring, and a dry friction act on the body due to the planariser  $F_{fr,p}$ . The friction of the sliding part of the actuator added to the ball screw friction and motor bearing frictions acts in the formulation by considering an overall efficiency  $\eta$  for the actuator. In the virtual work approach we assume infinitesimal virtual displacement on the system and find the virtual work done on the system, which will be equal to the generalized force times virtual displacement. To decouple the generalized forces,  $\mathbf{Q}_z$  and  $\mathbf{Q}_p$ , we apply decoupled virtual displacements  $dz$  and  $dp$  respectively.

Suppose we have a virtual displacement  $dz$  and no displacement  $dp$  then the virtual work will be:

$$d\mathbf{W}_z = \mathbf{Q}_z dz = F_{fr,p} \text{sign}(-\dot{z}) dz \quad (2.7)$$

and the generalized force turn out to be:

$$\mathbf{Q}_z = -F_{fr,p} \text{sign}(\dot{z}) \quad (2.8)$$

Now suppose a virtual displacement  $dp$  and no displacement  $dz$ , the virtual work will be:

$$d\mathbf{W}_p = \mathbf{Q}_p dp = -c\dot{s}ds + \tau_s d\theta \quad (2.9)$$

where  $\tau_s$  is the torque acting on the sliding mass and is equal to  $\eta\tau$  where  $\tau$  is the motor torque. Moreover,

$$s = s_0 - p \quad \implies \quad \dot{s} = -\dot{p} \quad ; \quad ds = -dp \quad (2.10)$$

Finally using Equation (2.4) and Equation (2.10) into Equation (2.9), the generalized force will be exerted.

$$\mathbf{Q}_p = -c\dot{p} + \frac{\eta\tau}{r} \quad (2.11)$$

Using Lagrangian Equation (2.1) considering Equation (2.8) and Equation (2.11) we end up with,

$$m_t\ddot{z} - m_s\ddot{p} + m_tg = -F_{fr,p}\text{sign}(\dot{z}) \quad (2.12)$$

$$\alpha\ddot{p} - m_s\ddot{z} - m_sg = \frac{\eta}{r}\tau - k(s_0 - l_0 + p) - c\dot{p} \quad (2.13)$$

The final state equation will be:

$$\begin{Bmatrix} \ddot{z} \\ \ddot{p} \end{Bmatrix} = \frac{1}{\gamma} \begin{bmatrix} \alpha & m_s \\ m_s & m_t \end{bmatrix} \begin{Bmatrix} -m_tg - F_{fr,p}\text{sign}(\dot{z}) \\ m_sg + \frac{\eta\tau}{r} - F_s \end{Bmatrix} \quad (2.14)$$

where  $F_s$  is the spring force, and

$$\begin{aligned} \alpha &= \frac{J}{r^2} + m_s \\ \gamma &= \alpha m_t - m_s^2 \\ F_s &= k(s_0 - l_0 + p) + c\dot{p} \end{aligned} \quad (2.15)$$

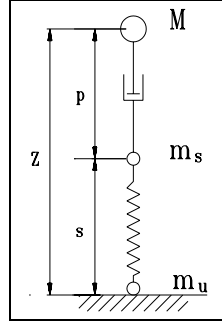
The final state equation can be shown in the standard form:

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\tau + \mathbf{E} \quad (2.16)$$

where:

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & \frac{-m_sc}{\gamma} \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & \frac{-m_tc}{\gamma} \end{bmatrix}; \quad \mathbf{B} = \frac{\eta}{\gamma r} \begin{Bmatrix} 0 \\ m_s \\ 0 \\ m_t \end{Bmatrix}; \quad \mathbf{x} = \begin{Bmatrix} z \\ \dot{z} \\ p \\ \dot{p} \end{Bmatrix} \quad (2.17)$$

$$\mathbf{E} = \frac{1}{\gamma} \begin{Bmatrix} 0 \\ \alpha(-m_tg - F_{fr,p}\text{sign}(\dot{z})) + m_s(m_sg - k(s_0 - l_0 + p)) \\ 0 \\ m_s(-m_tg - F_{fr,p}\text{sign}(\dot{z})) + m_t(m_sg - k(s_0 - l_0 + p)) \end{Bmatrix} \quad (2.18)$$

Figure 2.3: *Vertical Hopper and actuator model during stance*

### 2.1.2.2 Touchdown

Touchdown occurs when  $z(t_{td}) = l_0$  and  $\dot{z} < 0$ . Suppose the velocity of the center of mass at touchdown just before impact is  $\dot{z}_{td-}$ . At touchdown the unsprung mass dissipates its kinetic energy to the ground when it suddenly is brought to rest; however, because of the leg spring the kinetic energy of the body remains unchanged. Thus the energy lost at touch down will be :

$$\Delta E_{td} = \frac{1}{2} m_u \dot{z}_{td-}^2 \quad (2.19)$$

and

$$\dot{z}_{td+} = \dot{z}_{td-} \quad (2.20)$$

The amount of energy dissipation of the unsprung mass at touchdown reaches to approximately  $1.2J$  for our physical machine, at the nominal hopping height with a touchdown velocity of  $-2m/s$ .

### 2.1.2.3 Stance Phase

Considering Figure (2.3), the stance model is derived using the Lagrangian  $\mathbf{L} = \mathbf{T} - \mathbf{V}$ , where

$$\mathbf{T} = \frac{1}{2} J \left( \frac{\dot{p}}{r} \right)^2 + \frac{1}{2} M \dot{z}^2 + \frac{1}{2} m_s (\dot{z} - \dot{p})^2 \quad (2.21)$$

$$\mathbf{V} = Mgz + m_s g(z - p) + \frac{1}{2} k (s_0 - z + p)^2 \quad (2.22)$$

Therefore,

$$\mathbf{L} = \frac{1}{2}\alpha\dot{p}^2 + \frac{1}{2}(M + m_s)\dot{z}^2 - m_s\dot{z}\dot{p} - Mgz - m_s g(z - p) - \frac{1}{2}k(s_0 - z + p)^2 \quad (2.23)$$

where:

$$\alpha = \frac{J}{r^2} + m_s \quad (2.24)$$

The friction forces are modelled as viscous and dry frictions. In this case the dry frictions have two components. The first part is due to the planariser, namely  $F_{fr,p}$ , which is also present in the flight phase, and the second part is due to the sliding of the lower leg and upper leg, called  $f_{fr,s}$ , which is present only in the stance phase. Similar to the method used in the flight phase, we will use the virtual work approach to find the generalized forces.

First suppose a virtual displacement  $dz$  and no  $dp$ ,

$$d\mathbf{W}_z = \mathbf{Q}_z dz = -c\dot{s}ds + (F_{fr,p} + F_{fr,s})\text{sign}(-\dot{z})dz \quad (2.25)$$

But in this case,

$$s = z - p \implies \dot{s} = \dot{z} - \dot{p} \quad (2.26)$$

In general  $ds = dz - dp$ , but here we don't have  $dp$ ; therefore,

$$ds = dz \quad (2.27)$$

Thus the generalized force  $\mathbf{Q}_z$  will be derived as,

$$\mathbf{Q}_z = -c(\dot{z} - \dot{p}) - (F_{fr,p} + F_{fr,s})\text{sign}(\dot{z}) \quad (2.28)$$

Now consider a virtual displacement  $dp$  and no  $dz$ ,

$$d\mathbf{W}_p = \mathbf{Q}_p dp = -c\dot{s}ds + \tau_s \dot{\theta} \quad (2.29)$$

Equation (2.26) holds for this case but,  $ds$  in this case is equal to,  $ds = -dp$ . Again  $\tau_s$  is the torque acting on the sliding nut of the actuator, and considering all the losses come through the efficiency  $\eta$ ,

$$\tau_s = \eta \tau \quad (2.30)$$

where  $\tau$  is the motor torque. Finally generalized force will be simplified to,

$$\mathbf{Q}_p = c(\dot{z} - \dot{p}) = \frac{\eta \tau}{r} \quad (2.31)$$

Using Equation (2.1), by differentiation from Equation (2.23) and inserting the generalized forces from Equation (2.28) and Equation (2.31) we end up with:

$$(M + m_s)\ddot{z} - m_s\ddot{p} + (M + m_s)g = k(s_0 - z + p) + c(\dot{p} - \dot{z}) - (F_{fr,p} + F_{fr,s})\text{sign}(\dot{z}) \quad (2.32)$$

$$\alpha\ddot{p} - m_s\ddot{z} - m_s g = \frac{\eta \tau}{r} - k(s_0 - z + p) - c(\dot{p} - \dot{z}) \quad (2.33)$$

The final state equation will be:

$$\begin{Bmatrix} \ddot{z} \\ \ddot{p} \end{Bmatrix} = \frac{1}{\beta} \begin{bmatrix} \alpha & m_s \\ m_s & M + m_s \end{bmatrix} \begin{Bmatrix} -(M + m_s)g + F_s - (F_{fr,p} + F_{fr,s})\text{sign}(\dot{z}) \\ m_s g + \frac{\eta \tau}{r} - F_s \end{Bmatrix} \quad (2.34)$$

where

$$\begin{aligned} \alpha &= \frac{J}{r^2} + m_s \\ \beta &= \left(M + \frac{J}{r^2}\right) m_s + \frac{MJ}{r^2} \\ F_s &= k(s_0 - z + p) + c(\dot{p} - \dot{z}) \end{aligned} \quad (2.35)$$

The final state equation can be shown in the standard form:

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\tau + \mathbf{E} \quad (2.36)$$

where:

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & \frac{-J_c}{r^{\bar{a}}\beta} & 0 & \frac{J_c}{r^{\bar{a}}\beta} \\ 0 & 0 & 0 & 1 \\ 0 & \frac{M_c}{\beta} & 0 & \frac{-M_c}{\beta} \end{bmatrix}; \quad \mathbf{B} = \frac{\eta}{\beta r} \begin{Bmatrix} 0 \\ m_s \\ 0 \\ M + m_s \end{Bmatrix}; \quad \mathbf{x} = \begin{Bmatrix} z \\ \dot{z} \\ p \\ \dot{p} \end{Bmatrix} \quad (2.37)$$

$$\mathbf{E} = \frac{1}{\beta} \begin{Bmatrix} 0 \\ \alpha(-(M + m_s)g + k(s_0 - z + p) - (F_{fr,p} + F_{fr,s})\text{sign}(\dot{z})) + m_s(m_s g - k(s_0 - z + p)) \\ 0 \\ m_s(-(M + m_s)g + k(s_0 - z + p) - (F_{fr,p} + F_{fr,s})\text{sign}(\dot{z})) + (M + m_s)(m_s g - k(s_0 - z + p)) \end{Bmatrix} \quad (2.38)$$

#### 2.1.2.4 Liftoff

When the leg is fully extended and  $z(t_{lo}) = l_0$  and  $\dot{z} > 0$ , liftoff occurs. The leg velocity just before liftoff is zero, and the body velocity is  $\dot{z}_{lo-}$ . There are no external forces acting on the total system so due to the conservation of linear momentum before and after lift off:

$$(M + m_s)\dot{z}_{lo-} - m_s\dot{p} = (M + m_u)\dot{z}_{lo+} + m_s(\dot{z}_{lo+} - \dot{p}) \quad (2.39)$$

then

$$\dot{z}_{lo+} = \mu \dot{z}_{lo-} \quad (2.40)$$

where  $\mu$  is defined as:

$$\mu = \frac{m_t - m_u}{m_t} \quad (2.41)$$

The amount of the energy dissipation at liftoff is less than that at touchdown. For our prototype with nominal hopping height and liftoff velocity of  $2m/s$  this value will be approximately  $75mJ$ , while the energy dissipation at touchdown is 16 times this.

### 2.1.3 Control and Simulation

The control of hopping height is based on energy evaluation and energy injection to the system. There are many control algorithms we can examine here [10]. In the present work we examine a few of them.

A simulation in C language is developed, which is designed for more complete models of robots. The body of the simulation has the capacity of implementing multi-legged robot models with different actuators and complexities. In this section we introduce the control principles examined in simulation and give the results of simulation. The data used in the simulation is given in the Table (2.1).

#### 2.1.3.1 Proportional Energy Error Control

The basic idea of this method is to measure the components of instantaneous energy of the system and find its error to the desired energy level for a desired hopping height, and to

Symbol	Description	Data
$M$	Sprung mass	9.0 <i>Kg</i>
$m_u$	Unsprung mass	600 <i>gr</i>
$m_s$	Sliding mass	250 <i>gr</i>
$m_t$	Total mass	9.850 <i>Kg</i>
$J$	rotating part moment of inertia	$2.7 \times 10^{-4}$ <i>Kg/m<sup>2</sup></i>
$r$	Ball-screw lead	5 <i>mm/rev</i>
$k$	Spring stiffness	11 <i>KN/m</i>
$c$	Viscous damping coefficient	0
$s_0$	Spring free length	0.23 <i>m</i>
$l_0$	Leg free length	0.61 <i>m</i>
$\eta$	Motor and ball-screw efficiency	%90
$\tau$	Maximum Motor torque input (torque control)	1.0 <i>Nm</i>
$F_{fr}$	Friction force during stance	10 <i>N</i>
$F_{fr,p}$	Friction force due to the planeariser	5 <i>N</i>
$F_{fr,s}$	Friction force due to the leg sliding parts	5 <i>N</i>
$k_e$	Energy control gain	0.2
$k_v$	Velocity control gain	30
$k_p$	Position Control gain	0.2
$k_{p,f}$	Position gain for flight control law	$8 \times 10^4$
$k_{v,f}$	Velocity gain for flight control law	300
$E_d$	Desired energy	70 <i>J</i>
$l_1$	Actuator travel in step position control	10 <i>mm</i>

Table 2.1: *Parameters of vertical hopper model*

inject energy to the system equal to the amount of error during stance phase. If  $\Delta E_{add}$  represents the injected energy to the system, and  $E_D$  and  $E(t)$  are the desired energy and the instantaneous energy respectively, then

$$\Delta E_{add} = E_D - E(t) \quad (2.42)$$

Suppose the potential reference level is ground. Then the desired energy level for hopping height  $H_D$  is :

$$E_D = Mg(l_0 + H_D) + m_s \cdot g(s_0 + H_D) + m_u \cdot g \cdot H_D \quad (2.43)$$

The energy of the system during stance can be derived adding Equation (2.21) and Equation (2.22).

$$E(t) = \frac{1}{2}J\dot{\theta}^2 + \frac{1}{2}M\dot{z}^2 + \frac{1}{2}m_s(\dot{z} - r\dot{\theta})^2 + Mgz + m_s g(z - p) + \frac{1}{2}k(s_0 - z + p)^2 \quad (2.44)$$

If we measure spring force  $F_m$ , then

$$\frac{1}{2}K(s_0 - z + p)^2 = \frac{1}{2} \frac{F_m^2}{K} \quad (2.45)$$

We need to measure  $z$ ,  $\dot{z}$ ,  $\theta$ , and  $\dot{\theta}$  to find the total amount of energy.

The energy added to the system is due to the actuator velocity which pushes the spring. If  $F_s$  represents the spring force, then:

$$\Delta E_{add} = \int_0^t F_s(t) \cdot \dot{p}(t) dt \quad (2.46)$$

by differentiation

$$\dot{p}(t) = \frac{1}{F_s(t)} \cdot \frac{d}{dt}(\Delta E_{add}) \quad (2.47)$$

By integrating we reach to

$$p(t) = \left[ \int_0^t \frac{1}{F_s(t)} \cdot \frac{d}{dt}(\Delta E_{add}) dt \right] + p(t_0) \quad (2.48)$$

These equations is too complicated to be implement in practice, however, it leads us to simple position or velocity control of the motor. The energy error introduced here could be used in other control strategies. A torque control law for the motor, or actuator step position could be based on this formulation. We elaborate them in separate sections.

### 2.1.3.2 Open Loop Position Step

This control law is basically a step position control of the actuator during decompression of spring. The purpose of control is to gain the losses by injecting energy during the stance phase. We examine this law since it is simple and some research is done based on this law [23], although with a DC motor and ball-screw transmission is not realistic.

The amount of  $p$  is :

$$p(t) = l_1 u_t(t - t_1) \quad (2.49)$$

where  $u_t$  represents the step function and  $t_1$  is the time of maximum spring compression, and the beginning of the decompression phase. For  $l_1 = 10mm$  and the other data given in

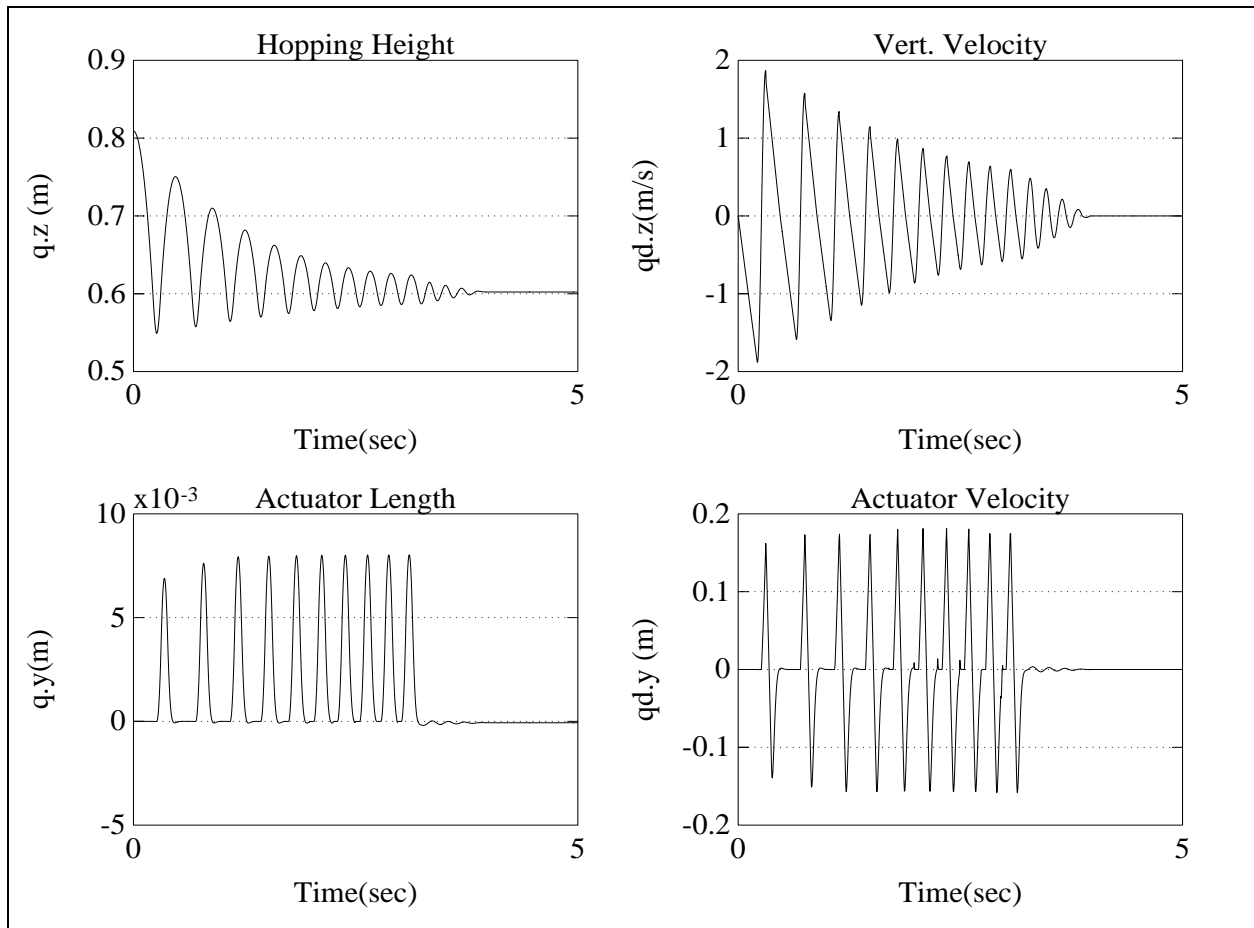


Figure 2.4: *Open loop position step; State variables*

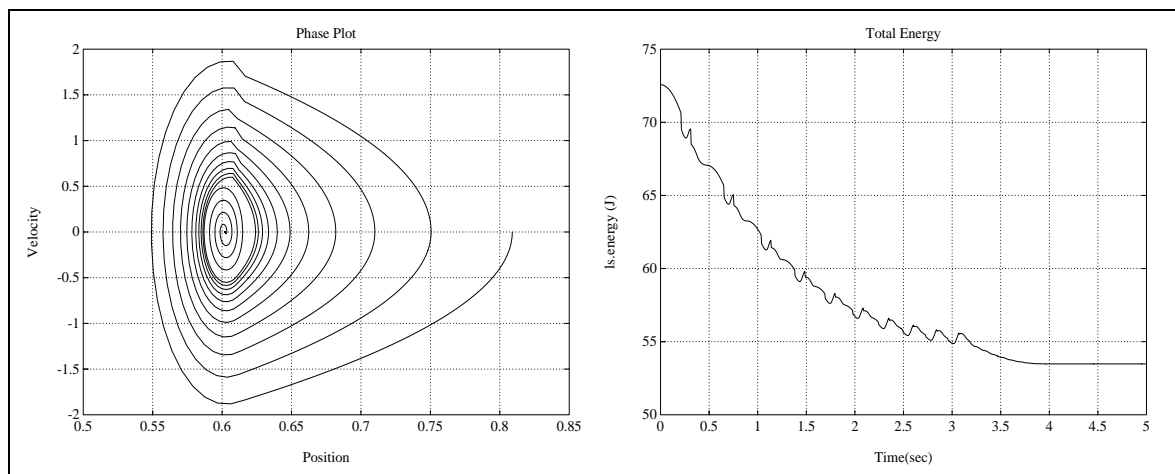


Figure 2.5: *Open loop position step; Phase plot and total energy*

Table (2.1), the state parameters are illustrated in Figure (2.4). The hopping height never stabilizes to a fixed height and the phase plot illustrated in Figure (2.5) does not show a stable limit cycle, for this case. The total energy of the system decreases continuously, and the control input could not gain the energy losses due to friction and impact momentum transfer. This is because of the friction we have assumed for the system. We can reach to some stable limit cycles, provided to have more actuation position input, or less friction.

### 2.1.3.3 Closed Loop Position Step

This control law is very similar to the control law of Section (2.1.3.2), but we consider a position step as a function of the energy-error. The amount of position step is proportional to the energy error:

$$p = k_p (E_D - E(t))u_t(t - t_1) \quad (2.50)$$

To find an estimate value for  $k_p$  we simplified Equation (2.48) and find a normalized amount for  $k_p$ :

$$k_p = \frac{E_D - E(t)}{K(z(t_1) - z(t_{td}))} \quad (2.51)$$

Where  $t_{td}$  correspond to touch down instance. The simulation results with  $k_p = 0.2$  is illustrated in Figure (2.7) and Figure (2.6). In this case we observe a reentrant behaviour of the hopper with a reasonable amount of actuator position input. The energy and phase plot confirms the stability of the system.

### 2.1.3.4 Torque control

For DC motors the most realistic control law should be based on torque control. In most of typical servo motors, there is a current feedback, which could be used in control law. Since the torque of the motor is proportional to its input current, a control law based on torque is easily implementable. In this case the input torque could be fixed or a function of energy error. Fixed torque control law is in fact a stable open loop control law, thus by setting

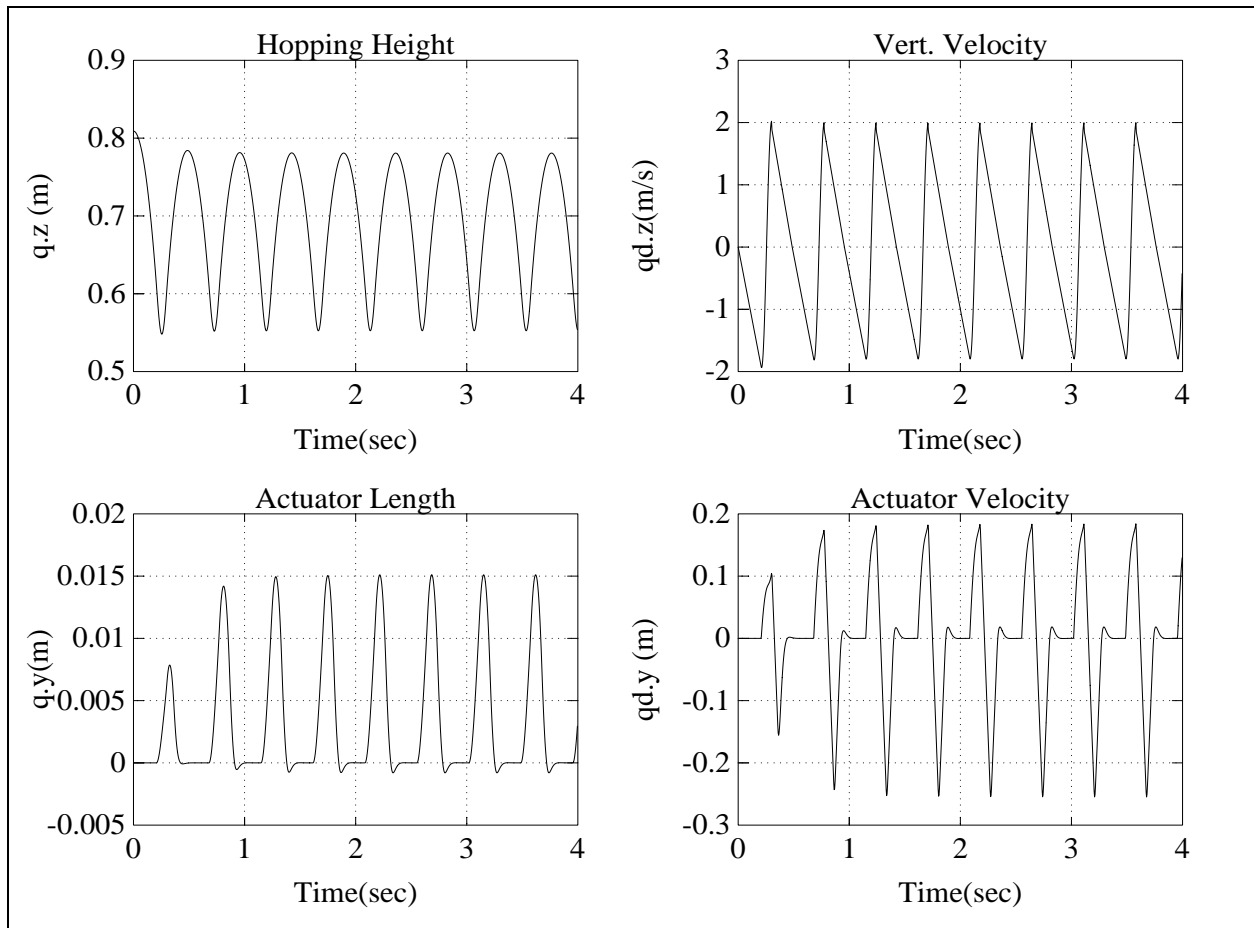


Figure 2.6: Closed loop position step; State variables

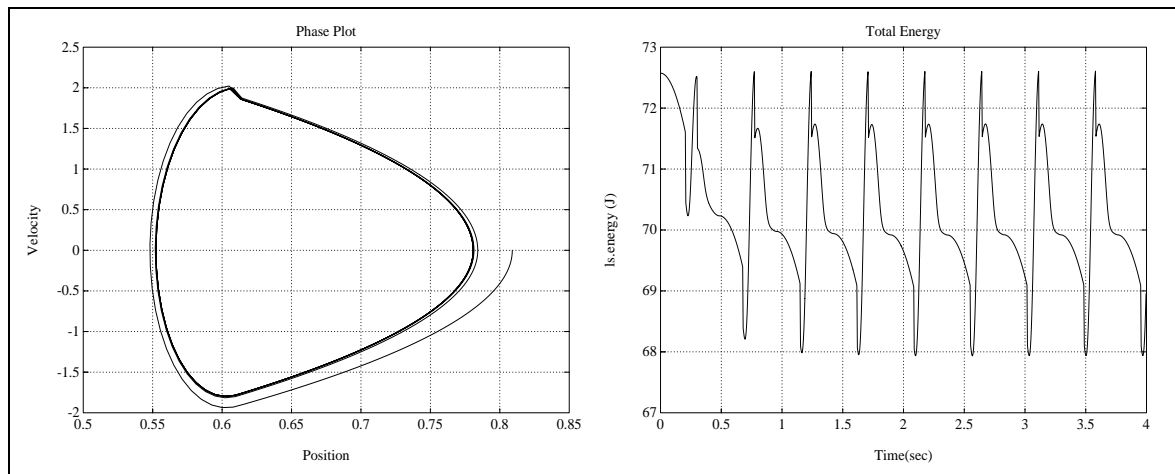


Figure 2.7: Closed loop step; Phase plot and total energy

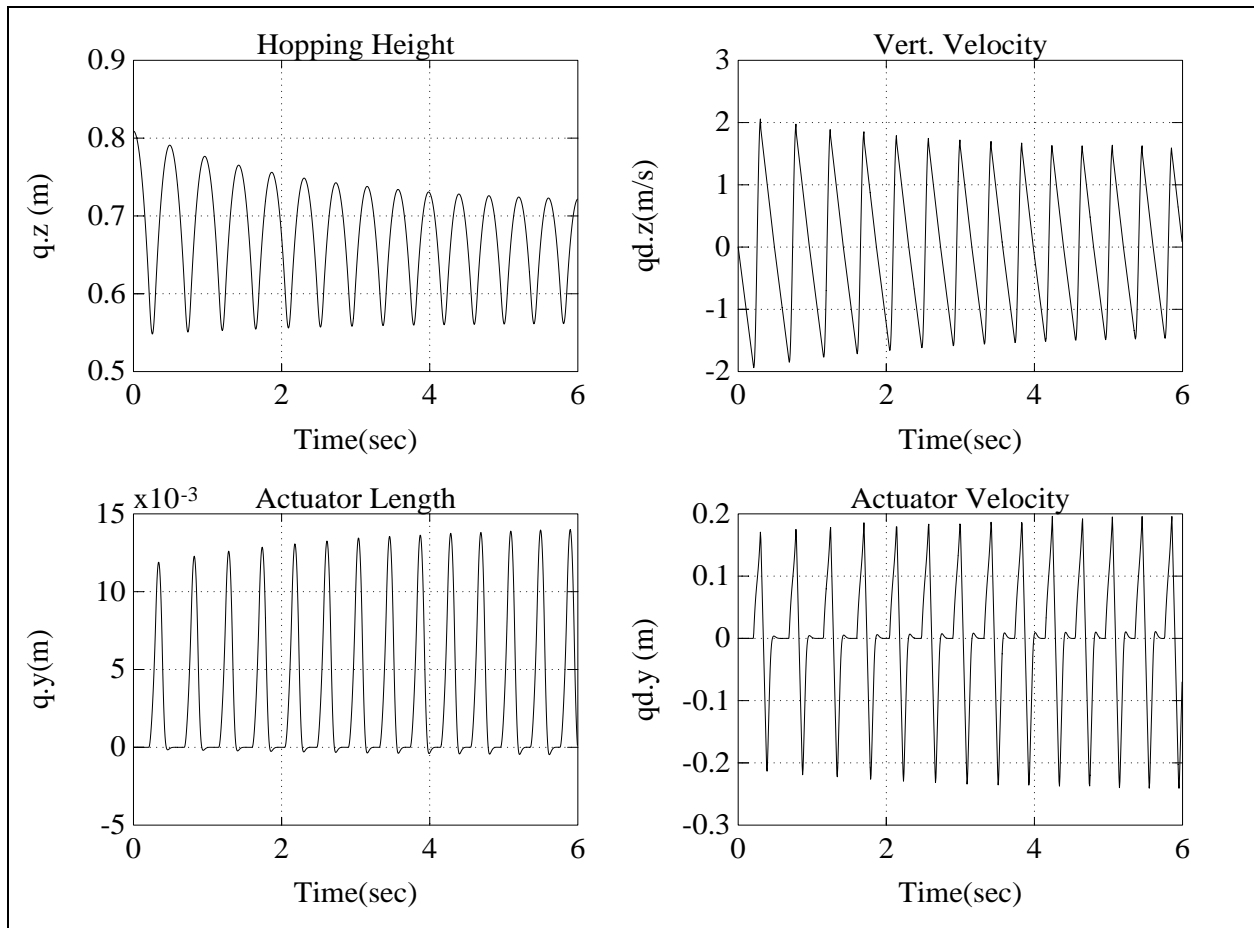


Figure 2.8: *Open loop torque control; State variables*

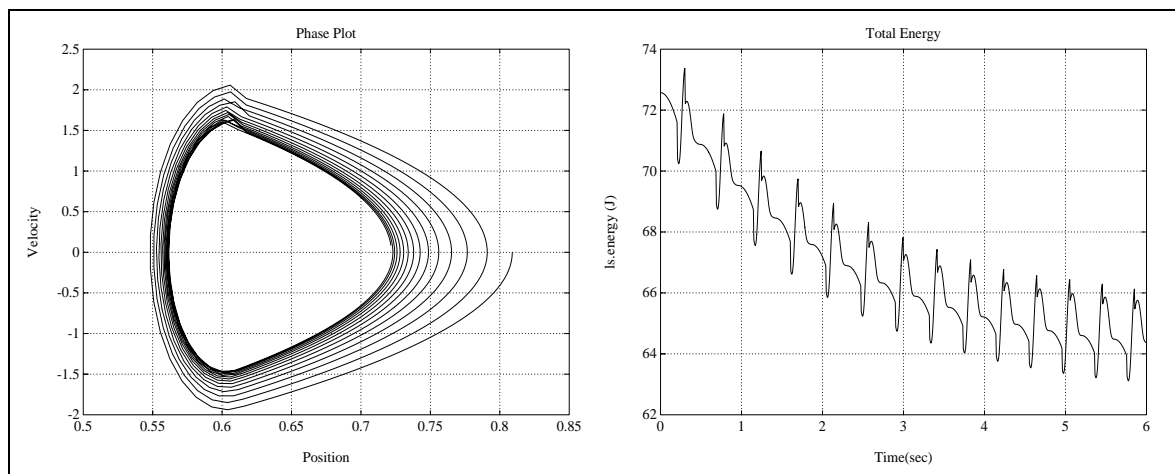


Figure 2.9: *Open loop torque control; Phase plot and total energy*

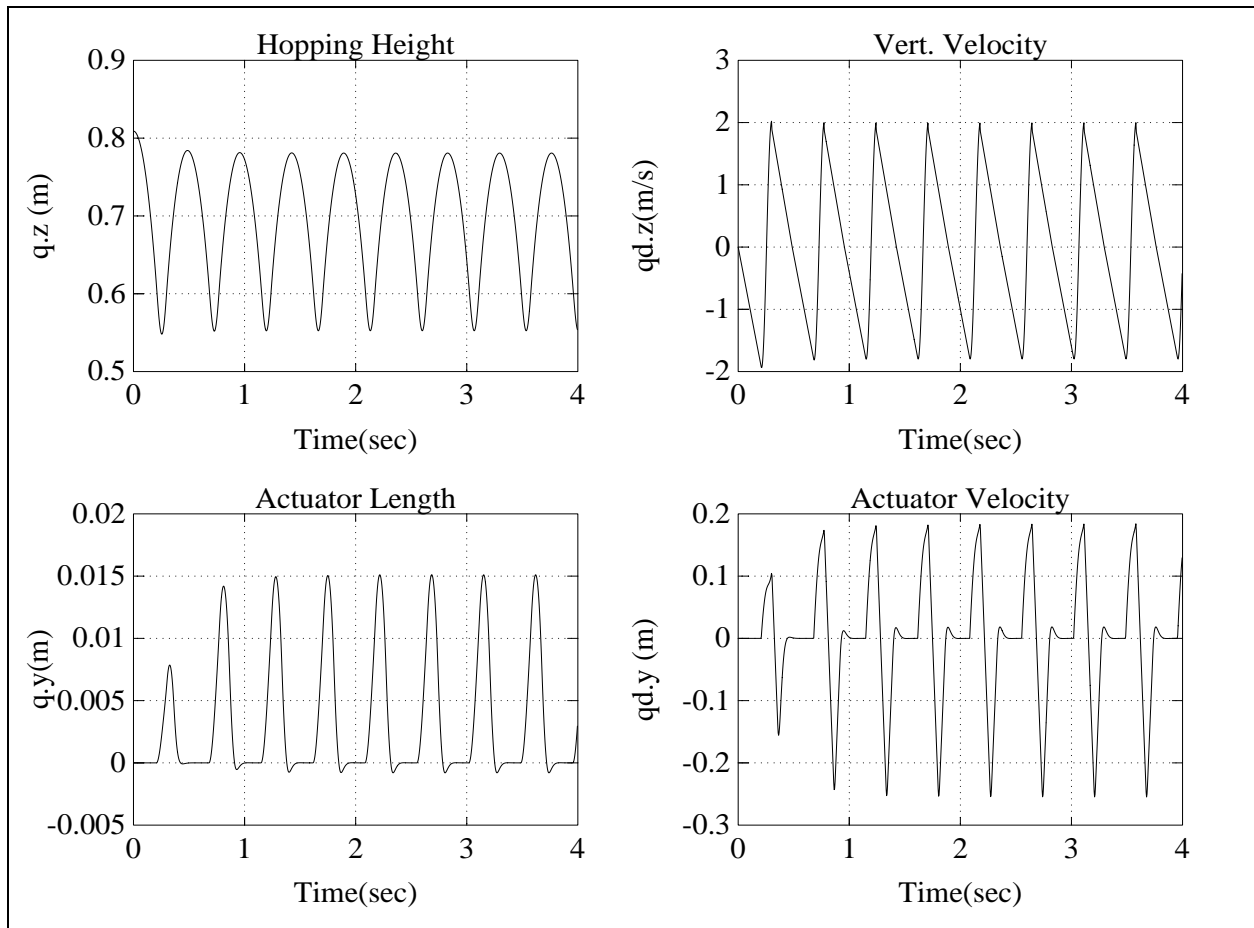


Figure 2.10: Closed loop torque control; State variables

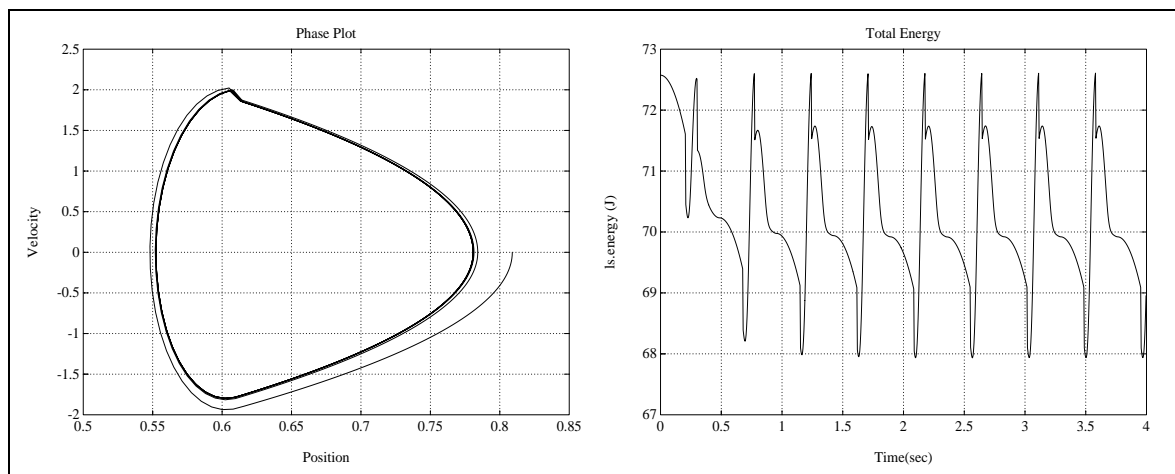


Figure 2.11: Closed loop torque control; Phase plot and total energy

the amount of fixed torque we will reach to a specific steady state hopping height. However the other approach lets us define a desired energy level and force the system to our desired hopping height. Figure (2.9) and Figure (2.8) illustrate the case where we apply a continuous torque of  $1Nm$  during the stance phase. We observe a stable duty cycle and a asymptotic energy level at steady state which is a function of the applied torque to the system.

The results of simulation with a closed loop torque control law based on :

$$T(t) = -K_e (E(t) - E_D) \quad (2.52)$$

and  $K_c = 0.2$  is illustrated in Figure (2.10) and Figure (2.11). The difference of these two approach are clearly understood by comparing the phase plots. In this case we have a very fast converging to the desired energy level, although in the open loop torque control we observe a delayed convergence.

### 2.1.3.5 Actuator Velocity Control

Regarding Equation (2.47) the actuator velocity is directly proportional to energy error. This leads us to the velocity control law given by following equation:

$$\dot{p}(t) = -K_v (E(t) - E_D) \quad (2.53)$$

The simulation results for  $K_v = 30$  is illustrated in Figures 2.12 and Figure (2.13). Again we observe a stable duty cycle. The success in these control options motivates constructing a prototype and examining them in practice. It's obvious that there are some other important aspect that should be considered in practice. For example, consider this fact that in this strategy there is no limitation on the maximum torque applied by the motor. Therefore, in practice we anticipate not the same response as in the simulation.

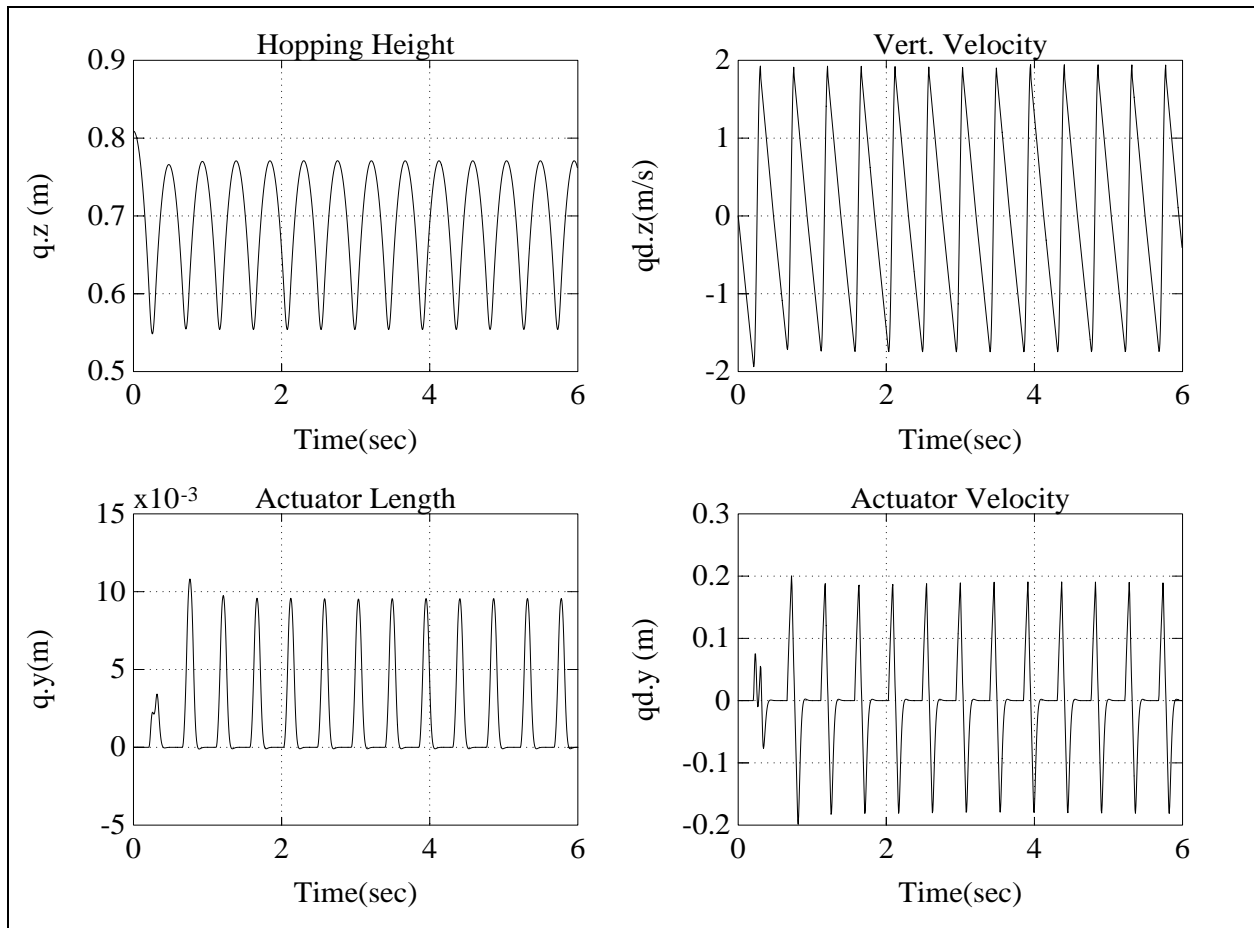


Figure 2.12: Actuator velocity control; State variables

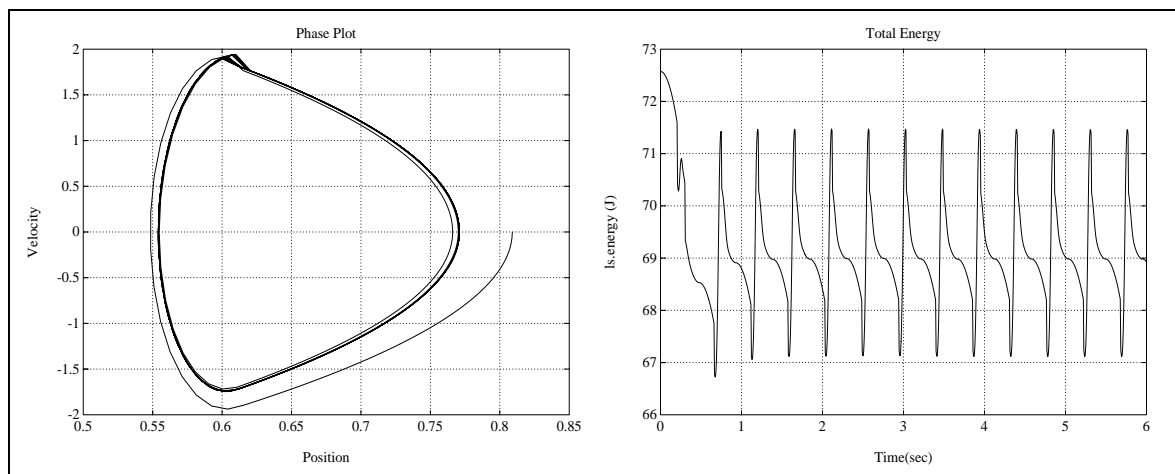


Figure 2.13: Actuator velocity control; Phase plot and total energy

## 2.2 Planar Passive Dynamics

In this chapter we review the passive dynamics modelling of planar robot given by Thompson and Raibert [27]. This modelling is used to simulate the planar passive robot and to derive one active controller for it. It seems necessary to bring the important part of modelling here, since it help us to understand the physics of running robot.

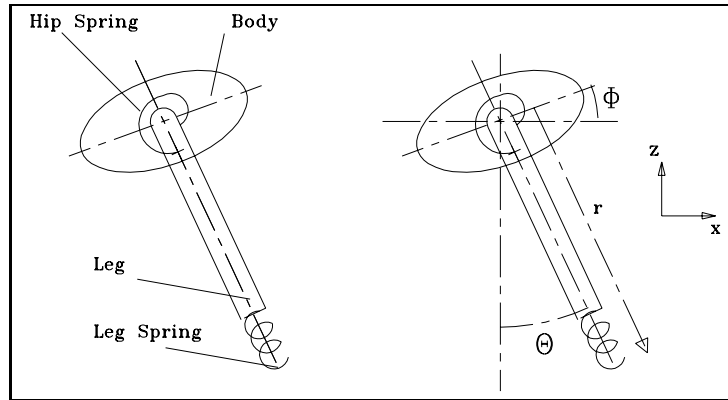
Running is a motion that combines a vertical oscillation of the body with a fore-aft oscillation of the legs. In passive dynamic approach we study how stored elastic energy can be used to generate vertical and angular motion of the body, without requiring a large expenditure of energy on each step. We introduce the definition of stance and flight phase for running. Stance phase represents the time interval, which the leg contacts the ground, while during flight phase, the leg has no contact with the ground.

The body can bounce on springy legs during stance phase, storing a portion of the kinetic energy, and releasing it later to help power the next step. In the same way if hip joint is also springy, the same result will be gained in fore-aft oscillation of each leg. The goal is to avoid loosing the kinetic energy of the legs each time they reverse their fore-aft sweeping motion. This approach is appealing because it offers energetically efficient locomotion of the body and contributes to simplified control. The legs can be made into harmonic oscillation by introducing torsional springs at the hip joints. The resulting leg oscillation move the foot backward with respect to the hip during stance phase, and forward in preparation for the next step during flight phase.

The objective of passive dynamic study is to see if it is possible to design a mechanical system such that its unactuated motion is close to the desired one. Depending on which model we are using, we can accomplish the following steps to design a passive dynamic system.

- a)* Manipulating the running trajectory by tuning the natural frequency of the vertical bouncing motion to be a specific fraction of the natural frequency of the leg swinging oscillation.
- b)* Choosing appropriate initial conditions according to the running speed, by which the phase plots of the variables repeated on itself, one step after another.

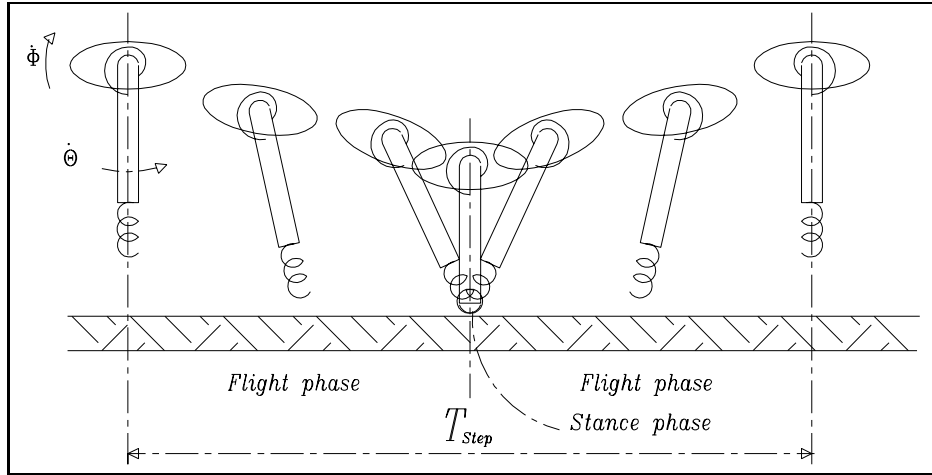
The systems we consider are passive in the way that they are made up of springs, links, and

Figure 2.14: *Planer one-legged model*

masses with no actuators. By this study, we don't claim that a physical legged system can operate passively for sustained periods of time. A source of energy is needed to make up for the losses, and to use for control to maintain the reentrant running trajectory. Once the passive part of the system is understood, it is possible to introduce actuators and control algorithms that provide energy and control, but we expect that still the system be very efficient in energy consumption.

### 2.2.1 Modelling

To study passive dynamic running, the simplest model is a planer one-legged model, which is shown in Figure (2.14). The model has a body of mass  $m_b$ , moment of inertia  $J_b$ , measured about hip, and the center of mass lies at the hip. The leg possess a mass  $m_l$ , moment of inertia  $J_l$  measured from center of mass which lies at hip. The model has two springs. One torsional spring which acts between leg and body with a stiffness  $k_h$ , and a compression spring acts along the leg axis between the lower part of the leg and the support surface. The leg spring is mass less, exerts force only during stance phase, and its stiffness is  $k_l$ . Moreover, we suppose that there exist no slip during stance, no impact load at touch down, and no other energy loss.

Figure 2.15: *Passive trajectory of the robot*

### 2.2.1.1 Operation

Before starting the mathematical analysis it is better to have a physical discussion on how the hopper works and which motion trajectory is ideal.

The motion begins at the initial point in which we prepare a certain initial condition for the hopper. Leg and body must have an appropriate angular velocity at initial point where they are at zero angular position ( $\phi_0 = \theta_0 = 0$ ). On the other hand the hopper must be dropped from an initial height  $z_{max}$  without initial velocity in vertical direction, and with the desired forward velocity. In fact appropriate initial condition in addition to right mechanical parameters assures passive dynamic motion of the robot. (In practice reaching to this specific initial conditions is a precise and difficult task) Now we can follow one step of the robot motion, which is illustrated in Figure (2.15). At initial point robot is at maximum height and maximum angular velocity. The hip and the leg springs are both at rest. By starting the motion, the kinetic energy of the body and the leg first stored at the hip spring as an elastic energy up to the touch down. At touch down the toe remains motionless on the ground, leg spring stores energy by compression, and the direction of  $\theta$  and  $\phi$  will change. The leg spring compression reduces the vertical velocity and ideally at the zero vertical velocity we expect zero angles for leg and body. From this point the motion continues in a mirror-shape trajectory. If at the end of the step all the conditions return to their initial ones the system

would have a reentrant behavior.

Now the effect of choosing initial conditions and appropriate mechanical parameters is more clear. If the hip spring characteristic period becomes equal to the flight time and the leg spring characteristic period becomes equal to stance time this condition would be accomplished. We will see in Section (2.2.3) that reaching to a fully reentrant behavior is not possible but the close we are to the passive dynamic condition the less energy is needed to get the desired reentrant trajectory.

### 2.2.2 Dynamics

Using Newton-Euler approach, the governing equations of motion are:

$$(m_b + m_l)\ddot{x} = F_x \quad (2.54)$$

$$(m_b + m_l)\ddot{z} = F_z - (m_b + m_l)g \quad (2.55)$$

$$J_b\ddot{\phi} = k_h(\theta - \phi) \quad (2.56)$$

$$J_l\ddot{\theta} = k_h(\phi - \theta) \quad (2.57)$$

where during stance phase:

$$F = k_l(r_0 - r) \quad (2.58)$$

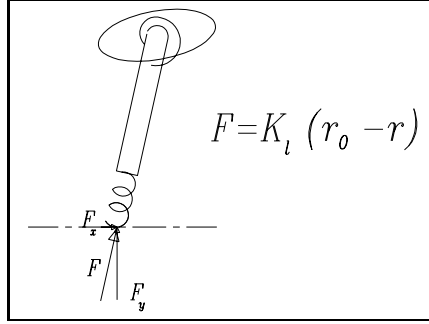
$$F_x = -F \sin(\theta) \quad (2.59)$$

$$F_z = F \cos(\theta) \quad (2.60)$$

and during flight phase:

$$F = F_x = F_y = 0 \quad (2.61)$$

A variable-step Runge-Kutta routine was used to integrate the equations of motion. For each simulation, we chose initial conditions and adjusted parameters to get the desired reentrant behavior. Now we want to derive explicitly the relation among the mechanical parameters, the initial conditions, and the desired motion trajectory.

Figure 2.16: *Ground forces during stance phase*

### 2.2.2.1 Vertical Bouncing

During flight, the center of mass of the system travels along a parabolic trajectory determined by the vertical position and velocity at lift off.

$$z(t) = z_{l0} + \dot{z}_{l0}t - \frac{gt^2}{2} \quad (2.62)$$

where  $z_{l0}, \dot{z}_{l0}$  are the vertical position and velocity of the body at lift off, and  $g$  is the acceleration of gravity. The peak altitude and duration of the flight phase are:

$$z_{max} = \frac{\dot{z}_{l0}^2}{2g} \quad (2.63)$$

$$T_f = \sqrt{\frac{8z_{max}}{g}} = \frac{2\dot{z}_{l0}}{g} \quad (2.64)$$

During stance the vertical motion is harmonic rebound determined by the system mass bouncing on the leg spring. The natural frequency of this rebound and stance phase period are:

$$\omega_l = \sqrt{\frac{k_l}{m_b + m_l}} \quad (2.65)$$

$$T_s = \frac{\pi}{\omega_l} \quad (2.66)$$

the exact  $T_s$  is slightly different [12]:

$$T_s = \frac{2(\pi - \arctan(|\dot{z}_{l0}|\dot{\omega}_l/g))}{\omega_l} \quad (2.67)$$

but Equation (2.66) gives reasonable results when vertical velocity at touch down is large compared to  $g/\omega_l$  [27].

### 2.2.2.2 Hip Oscillation

The natural frequency and the characteristic period of the hip oscillation are:

$$\omega_h = \sqrt{\frac{k_h}{J_{eff}}} \quad (2.68)$$

$$T_h = \frac{2\pi}{\omega_h} \quad (2.69)$$

where the effective inertia  $J_{eff}$  is defined as

$$J_{eff} = \frac{J_b J_l}{J_b + J_l} \quad (2.70)$$

Equation (2.56) and Equation (2.57) give simple harmonic oscillation for  $\theta$  and  $\phi$ . So we can easily find the displacement of the foot from the hip which is :

$$x_{fh} = r \sin(\theta(t)) \quad (2.71)$$

where

$$\theta(t) = \frac{\dot{\theta}_0}{\omega_h} \sin(\omega_h t) \quad (2.72)$$

and  $r$  is the leg length, and we assumed that hip spring is at rest when  $\theta = \phi = 0$ .

### 2.2.2.3 Choosing Parameters for Passive Dynamic Running

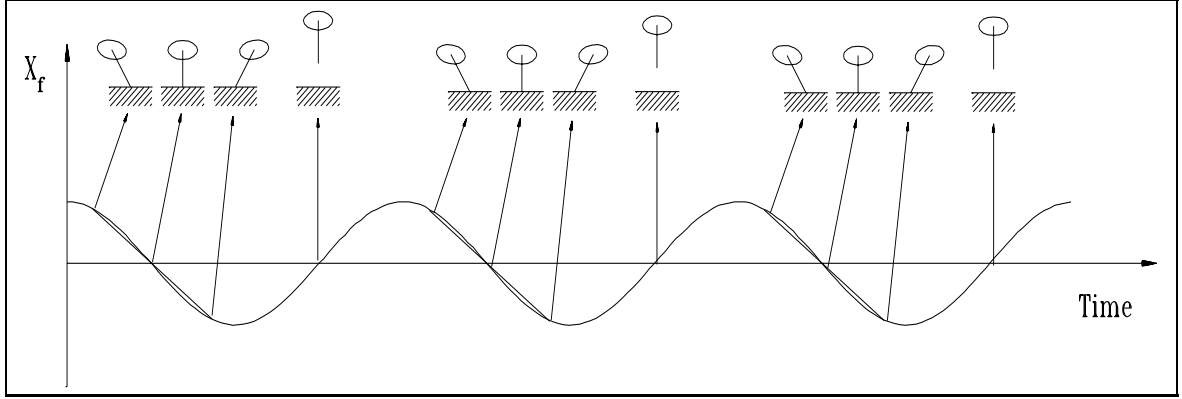
Figure (2.17) shows a foot motion produced by a harmonic hip oscillation that was tuned to approximate the foot motion used in ideal speed travel. The approximation is based on the linearity of the sin function for small values of its argument. The basic approach to find passive reentrant trajectories is based on this fact that harmonic hip motion can generate foot motion which closely approximates the foot motion found in ideal forward travel.

Now we want to discuss how to choose system parameters and initial conditions to find reentrant trajectories for the model :

a– Assume fixed values for  $m_b, m_l, J_b, J_l$ , and nominal leg length  $r_0$ .

b– Given a desired running speed  $\dot{x}_d$  and step period  $T_{step}$  and the fraction of stance time to step time  $\rho$ .

c– We choose spring constants  $k_h, k_l$  and initial conditions  $\dot{\theta}_0, \dot{\phi}_0, z_{max}$ .

Figure 2.17: *Harmonic oscillation of the hip*

#### 2.2.2.4 Spring Constants

The stiffness of the hip is chosen so the hip undergoes one complete oscillation during one complete step.

$$\omega_h = \frac{2\pi}{T_{step}} = \left( \frac{k_h}{J_{eff}} \right)^{\frac{1}{2}} \implies k_h = \left( \frac{2\pi}{T_{step}} \right)^2 J_{eff} \quad (2.73)$$

The stiffness of the leg spring is chosen to establish the duration of the stance phase as a fraction of the step time. We define  $\rho$  as:

$$\rho = \frac{T_s}{T_{step}} = \frac{\omega_h}{2\omega_l} \quad (2.74)$$

Small values for  $\rho$  ensures that stance phase occur during linear portion of the foot fore-aft travel, but result in larger leg peak forces and longer flight duration. Also as we will see later this parameter plays an important role in the stability of the system. If  $\rho$  is big the forces exerted to the system act in a longer time and disturb the system earlier. On the other hand, only during stance phase we can add control forces to stabilize the system. We experimented the values 0.125 to 0.25 which are good compromises. Given the natural frequency of the hip and the value of  $\rho$ ,  $k_l$  is:

$$k_l = m \left( \frac{\pi}{\rho T_{step}} \right)^2 \quad (2.75)$$

Since we don't have an exact expression for  $T_s$ , and it also changes slightly in each step, Equation (2.74) gives an approximate value for  $\rho$ . The desired value is obtained by adjusting  $k_l$  iteratively on a series of trials.

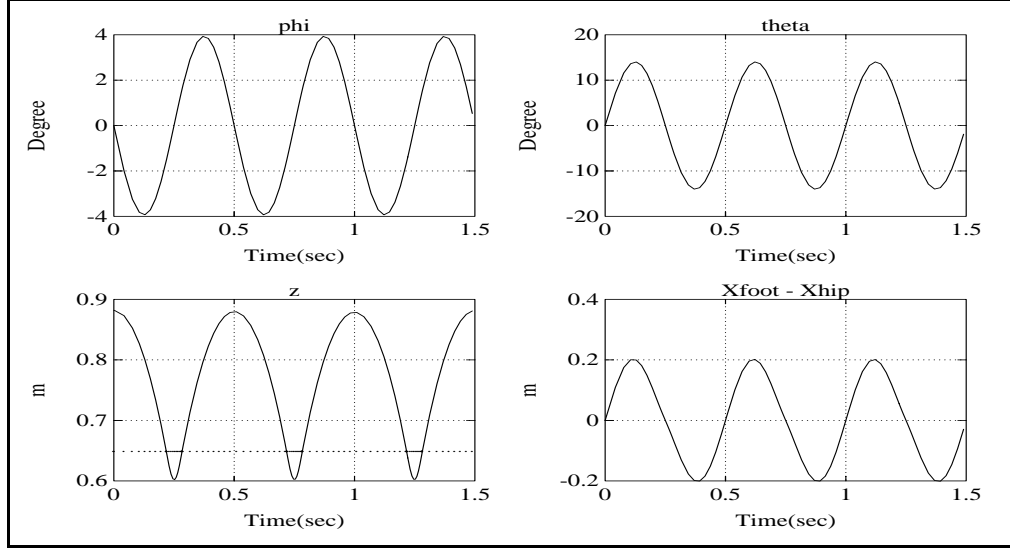


Figure 2.18: *Passive dynamic running simulation of planner one-legged hopper for running speed 2m/s*

### 2.2.2.5 Initial Conditions

The simulation begins by dropping the hopper from a specified height. Therefore, the initial state of the system is equivalent to the state at mid flight. The angular leg rate  $\dot{\theta}$ , is chosen so that the backward foot velocity at mid stance is matched to the desired forward speed, namely

$$\dot{\theta}_0 = \frac{\dot{x}_d}{r} \quad (2.76)$$

To maintain zero angular momentum during flight phase, we should have

$$\frac{\dot{\phi}}{J_l} = -\frac{\dot{\theta}}{J_d} \implies \dot{\phi}_0 = -\frac{J_l}{J_b} \dot{\theta}_0 \quad (2.77)$$

The initial altitude that provides the correct flight duration  $T_{step}(1 - \rho)$  is:

$$z_{max} = \frac{g}{8} T_f^2 + r_0 \cos \left( \frac{\dot{\theta}_0}{\omega_h} \sin \left( \frac{\omega_h T_f}{2} \right) \right) \quad (2.78)$$

Examining Equation (2.74) to Equation (2.77) we see that three independent parameters are required to specify the passive dynamic running motion which are:  $T_{step}$ ,  $\rho$ , and  $\dot{x}_d$ .

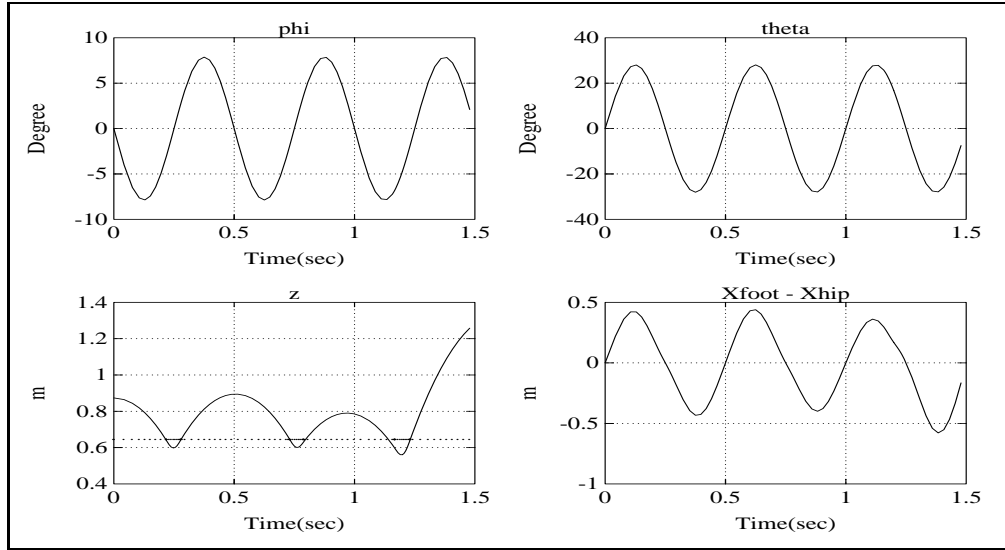


Figure 2.19: *Passive dynamic running simulation of planner one-legged hopper for running speed 4m/s*

### 2.2.3 Simulation

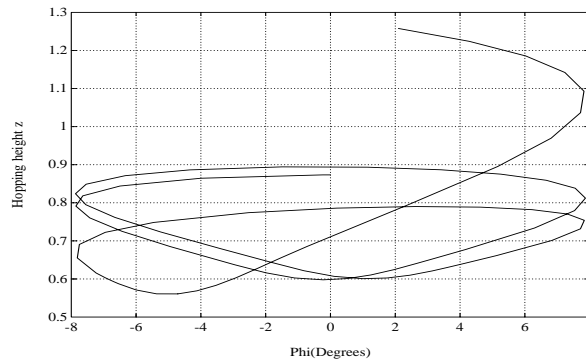
A variable-step Runge-Kutta routine in Matlab was used to integrate the equation of motion. To verify the results, we first run the program with Thompson and Raibert data [27]. The results were identical; therefore, we tried our desired data of one-legged robot which is shown in Table (2.2). In the simulation in addition to mass and dimension parameters three key parameters ( $\rho$ ,  $T_{step}$ , and  $\dot{x}_d$ ) are given and the other parameters will be calculated from passive dynamic formulation. The result of simulation for two running speeds 2 and 4 m/s are given in Figure (2.18) and Figure (2.19). In Figure (2.19), the hopping height becomes unstable. This will occur in almost all cases when the running speed is 4 m/s, and implies that the system have not a complete reentrant behavior. This effect is shown more clearly in Figure (2.20), where the pitch angle of the body is plotted versus hopping height. This figure shows a gradual drift in phase trajectory. If the system were stable, it should remain in a certain limit cycle. A complete implementation of passive dynamic running would include a control mechanism to stabilize the oscillation and eliminate this sort of drifts. This simulation is written for design purposes; thus, the effect of each parameter could be studied on the trajectory and stability. It is verified that the ratio of body moment inertia to

Symbol	Description	Data
$g$	the acceleration of gravity	$9.81m/s^2$
$m_b$	body mass	$12.5kg$
$J_b$	body moment of inertia	$0.82kgm^2$
$r_0$	leg rest length	$0.65m$
$m_l$	leg mass	$3.5kg$
$J_l$	leg moment of inertia	$0.23kgm^2$
$k_h$	hip torsional spring constant	$28.4Nm/rad$
$k_l$	leg spring constant	$40425N/m$
$T_{step}$	step time	$0.5s$
$\rho$	the ratio of the stance time to the step time	$0.125$
$\dot{x}_d$	desired velocity	$2.0, 4.0$
$r$	leg length	
$F_x$	horizontal force exerted on C.G.	
$F_z$	vertical force exerted on C.G.	
$\dot{z}_{l0}$	vertical lift off velocity	

Table 2.2: *Parameters of one-legged passive model*

leg moment inertia determines the amount of body pitching. A larger body moment inertia results in smaller body pitch angle. The ratio of body to leg mass also determines the amount of variation of vertical and forward velocities. The higher this ratio is, the smaller the variation of  $\dot{x}$  and  $\dot{z}$  are, but it results to a larger ground forces during stance.

## 2.2.4 Active Stability

Figure 2.20: *Phase plot; Pitch angle versus hopping height for horizontal speed of 4m/s*

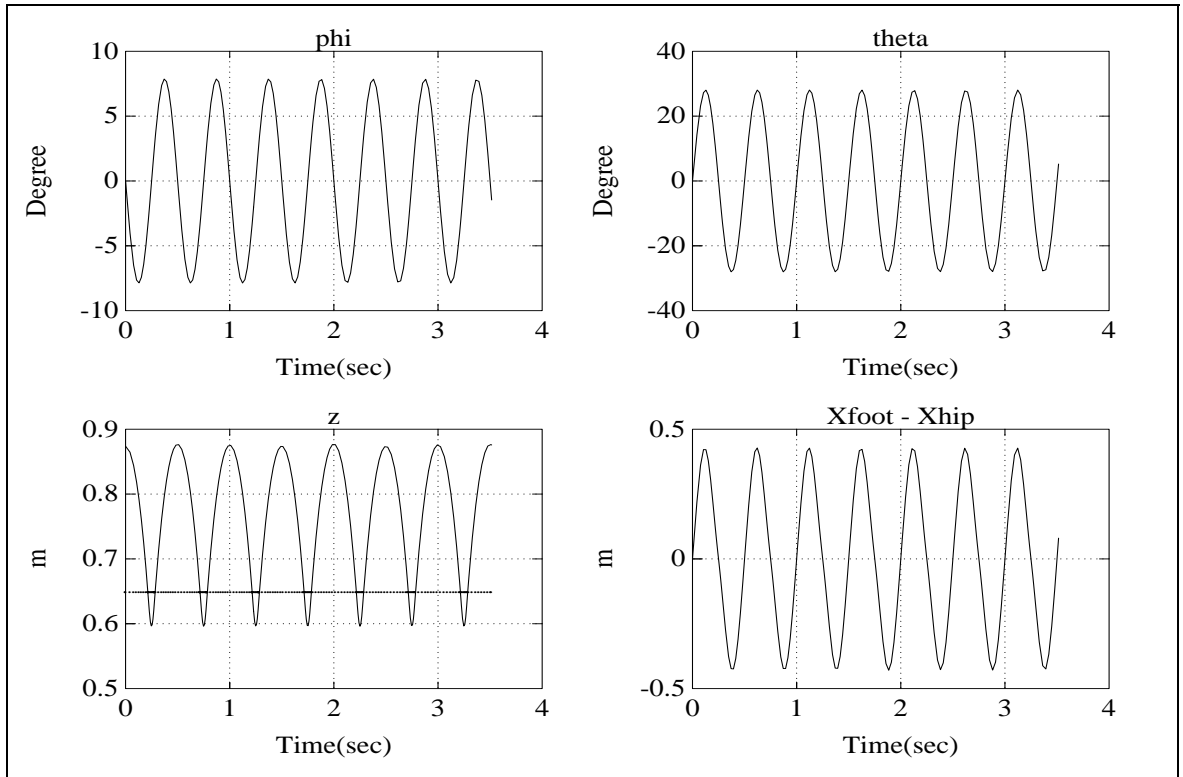


Figure 2.21: Simulation results for controller with  $K_c = 1000$  and horizontal velocity  $4\text{m/s}$

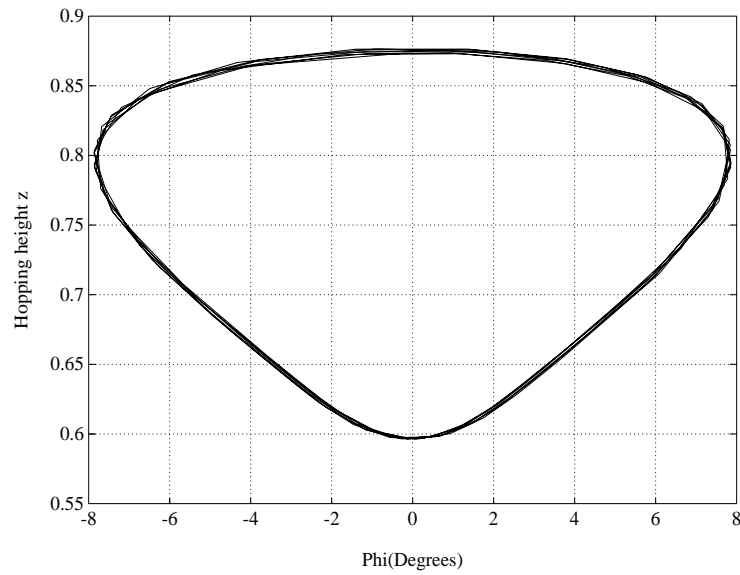


Figure 2.22: Phase plot of controller with  $K_c = 1000$  and horizontal velocity  $4\text{m/s}$

As it was shown in Figure (2.19) and Figure (2.20) the passive system alone is not enough to gain a reentrant behavior, and it is necessary to find a practical control algorithm to reduce the effect of disturbances. We can add energy to the system at hip joint and at the leg. We propose a new stabilizing control algorithm for this system. It controls forward velocity by regulating the ground force. We introduce a control force  $F_c$  which is proportional to the forward velocity error. This force acts only during the stance time and will be added to the ground forces ideally by means of a linear actuator which exerts  $F_c$  on the leg spring. Thus we have:

$$F_c = K_c(\dot{x}_d - \dot{x}(t)) \quad (2.79)$$

$$\sum F = F - F_c = k_l(r_0 - r) - F_c \quad (2.80)$$

Equation (2.80) is now used instead of Equation (2.58). We chose the amount of controller gain  $K_c$ , so that the maximum of controller force will be less than one tenth of maximum ground force. For this criteria  $K_c$  will be less than  $1000(m/Ns)$ . The results of simulation for the case of forward velocity  $4m/s$  is given in Figure (2.21) and Figure (2.22). Now we gain a complete reentrant behavior. The following explanation should be mentioned to understand why forward velocity error is related to the stability. During the stance phase the ground forces acting on the system cause to the system forward and vertical velocity and also the swinging motion of the hip and the leg deviate from the desired values. These motion are coupled together by the nature of the passive system. Therefore it is not surprising that we can tune the vertical motion by the forward velocity error.

In this case we neglect all the losses and this control mechanism was designed just to control the stability. Therefore, in practice we should design a more complete control mechanism to gain the losses and ensure stability. There is a complete model for planar dynamics given in Appendix A.

# Chapter 3

## Design

Human muscle has a maximum power density of about 100W/kg (for a comparison of human muscle with other contractive materials see [7]), which is readily achievable with today's "fractional horsepower" electric motors. These can attain peak power/mass ratios of up to 1kW/kg [1]! The reason is that muscles are "direct drive" actuators – they develop their maximum force at the low speeds needed to drive the biological joints directly. Small motors, with a mass of about 1kg, on the other hand, are severely torque limited, and the peak power is achieved only at high speeds (5000-10,000 RPM). Peak torque/mass ratios of 6Nm/kg are only reached by large direct drive motors whose intricate geometry (at present) cannot be scaled down to small motors [6]. This calls for a speed reducer (gear) in order to match the high torque–low velocity requirements of biological/robotic joint motion to the available low torque–high velocity of small motors.

This has two main consequences: First, virtually all large gear reducing mechanisms introduce additional dynamics which cannot be ignored. A significant fraction of the motor power can be consumed for merely driving the gear masses. Second, most high gear ratio mechanisms introduce significant loss of driving torque, up to 61% for a high quality three stage planetary gear head for a gear ratio range of 80 to 200 [1]. This precludes the use of fractional horsepower motors as direct drive force controlled actuators in legged robots in an analogous fashion to muscles. In addition, we need to convert the rotary motion of the motor

into linear motion suitable for our prismatic leg (Figure (3.5)). The only practical transmission that provides us both with linear motion and high efficiency (95%) is a ball screw. The design challenge then is to select a motor, lead screw pitch (lead), and spring constant such that the energy added during hopping is maximized. In other words, we want to achieve a motor torque-speed curve during operation as close as possible to allowable boundaries. Preliminary simulations pointed out that the main design challenge was to maximize the energy added during the short stance phase. In our case, the complete motor – ball screw – spring system must be optimized to this effect at some nominal operating point. Unfortunately, there are many practical considerations, which stand in the way of formulating and solving a standard optimization problem. For example, the stance time can be increased by lowering the spring constant. However, this increases the actuator travel during stance. This in turn requires a longer ball screw (only available at discrete lengths), which increases its moment of inertia (subject to catalog tables), which actually *limits* the energy added. In addition there is only a finite number of choices for suitable motors and ball screw lengths and leads. This results in a highly nonlinear and complex optimization problem. To simplify this task, we also discretized the number of available spring constants, and performed an exhaustive computer search. With the main components selected, the remainder of the leg is kept light and simple, as shown in Figure (3.5). The leg design is based on the analytical analysis made on the planar and vertical hopper. Like Raibert's first hopper, we designed the leg consisting of a prismatic leg connected to a body via a revolute joint. For vertical hopping experiments, we have not connected the body via the hip, and blocked the swinging motion of the leg. The design challenge for the leg is then to compromise the weight and the strength. On the other hand the simplicity of the design is of great importance, for the disassembling and assembling should be easy and fast.

### 3.1 Actuator Design

The muscle system consist of three major parts: Motor, ball screw transmission, and spring. There are different range of DC servo motor applicable for robotic application; moreover,

Motor Type	Stall Torque (Nm)	Energy Added (J)
Maxon 100 WEC	0.382	1.8
Maxon 100 EC	0.470	1.5
Maxon 80	2.000	15.0
Pittman 14006	2.000	14.8

Table 3.1: *Simulation comparison of different motors, regarding  $k=10$ ,  $r=5$*

the standard ball screw transmission have different categories. In this section we want to elaborate the method we chose suitable standard motor and ball screw for our application. The most important characteristic that they should possess for this type of application is high power to weight ratio. Regarding the availability in short period and the cost, there are four kind of DC motor assigned for the verification test. Ball screws are categorized with their leads, and springs are determined with their stiffness. There is a simulation written in Matlab to calculate the maximum energy added to the system during first stance phase of vertical hopper model discussed in Section (2.1). In this simulation we suppose the motors work on their maximum power, i.e. they are working on the maximum permissible speed torque curve. For several leads and springs a search is done to find the maximum energy added to the system. Table (3.1) express the final results of the simulation. Two kind of motors namely Maxon 80 and Pittman 14006 show a similar characteristics with respect to the energy and power during stance phase, where among them Maxon 80 was chosen [1]. One of the insights in this comparison is that the component selection is far from intuitive, since the robot – actuator is a fourth order dynamical system. For example, a higher power (100W), lower weight (0.3 kg) brush less motor (Maxon 100 EC) fared far worse due to the smaller torque at low speed; the velocities where high power is available are never attained during the short stance time. This comparison shows that the motors with high stall torque are preferable, even if their nominal power is not high.

Again we used the same simulation for optimum ball screw lead. The available ball screws have a nominal lead of 2,3,4,5, and 10 and up [25]. By increasing the lead we are tuning two aspects. The gear ratio is increasing; therefore, the motor works in lower speed, thus lower motor torque is available. On the other hand the spring stroke increases as the lead is larger;

therefore, the spring force increases and the power increases. Figures 3.1, 3.2, and 3.3 are the simulation results for ball screws with lead 2, 5, and 10 millimeters, and spring stiffness 10 KN/m. Considering added energy plots, and power output, the ball screw with lead 5 millimeters gives the best result. We reach to the similar results for different spring stiffness; therefore, Maxon 80 watt motor and balls-screw with lead 5 is chosen for the muscle system. We chose standard NSK ball screw with lead 5mm/rev and its standard support. Since the moving nut is in touch with the spring, we could have only one end support for ball screw, which make the buckling of the ball screw of great importance. In the design of the ball screw and support the impact loading is considered in addition to the buckling problem [25]. The other feature of design was safety consideration. In the case of hardware or software crash we should have some safety stop for the system. This feature is accomplished for the ball screw by putting two mechanical stops at the both end of the stroke. The stroke of the ball screw is set to 50 millimeters based on the simulation result.

The spring stiffness is the last parameter to be verified. Refer to Sections 2.2 and 2.1 we know that the spring constant is important since it change the stance time and frequency of oscillation. Especially stance time is important since it is the time for injecting energy to the system. On the other hand at this time the disturbance forces act to the system and for planar hopper it makes trouble for long term stability. A good compromise is found by the simulation results given in Sections 2.2 and 2.1. After doing the detail design and getting better estimates of weights and dimensions we tried that simulation again to find a suitable value for spring constant. A 20 KN/m spring for planar hopper, and a 10 KN/m spring for vertical hopper are the suitable springs. Therefore, we decided to design two springs with these constants and the same overall dimensions for the hopper.

## 3.2 Spring Design

The standard high duty springs [3] with our prescribed stiffness have long length, and buckling problem [2]. Therefore, we decided to design our own springs. The shear stress in a

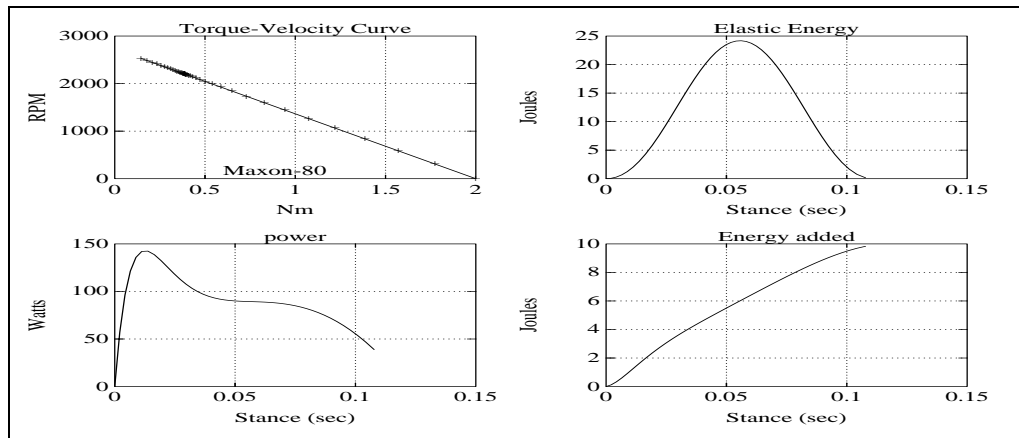


Figure 3.1: Simulation result for ball screw verification;  $k=10$  KN/m, Lead=2

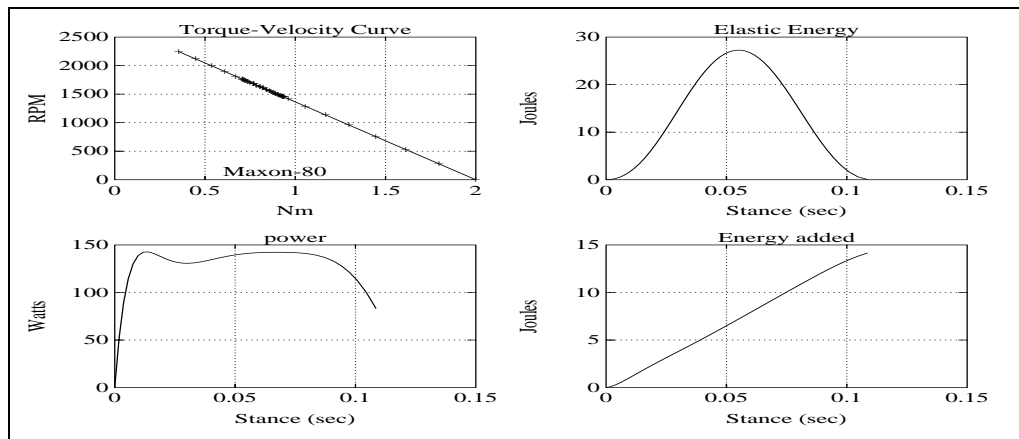


Figure 3.2: Simulation result for ball screw verification;  $k=10$  KN/m, Lead=5

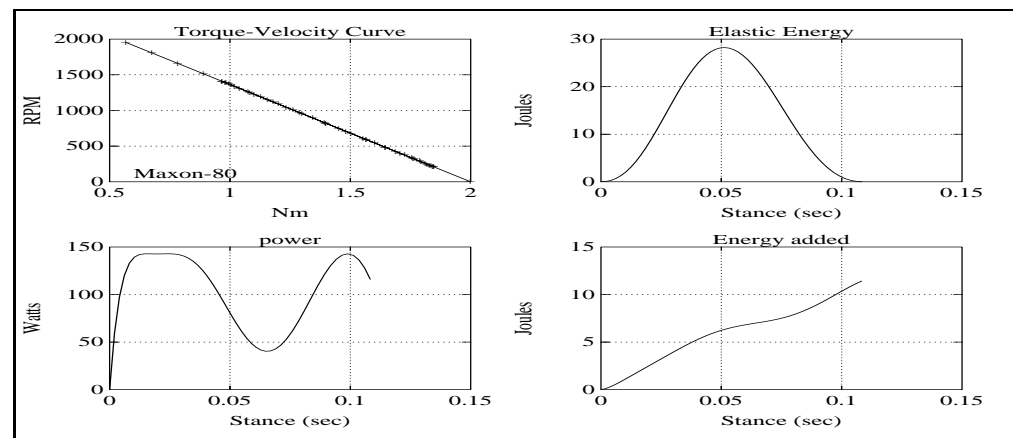


Figure 3.3: Simulation result for ball screw verification;  $k=10$  KN/m, Lead=10

round-wire helical compression spring, loaded by the static axial force  $F$  is [26]:

$$\tau = K_s \frac{8FD}{\pi d^3} \quad (3.1)$$

where  $d$  is the wire diameter and  $D$  is the mean spring diameter and  $K_s$  is a shear-stress concentration factor given:

$$K_s = \frac{2C + 1}{2C} \quad ; \quad C = \frac{D}{d} \quad (3.2)$$

The spring rate  $k$  is given by:

$$k = \frac{d^4 G}{8D^3 N} \quad (3.3)$$

where  $N$  is the number of active coils and  $G$  is module of rigidity of the spring material.

For fatigue load with maximum and minimum loading  $F_{max}$  and  $F_{min}$ . we define the force amplitude and the force mean value as:

$$F_a = \frac{F_{max} - F_{min}}{2} \quad ; \quad F_m = \frac{F_{max} + F_{min}}{2} \quad (3.4)$$

Then the shear stress amplitude and the shear stress mean value would be

$$\tau_a = K_B \frac{8F_a D}{\pi d^3} \quad ; \quad \tau_m = K_s \frac{8F_m D}{\pi d^3} \quad (3.5)$$

where  $K_B$  is Bergsträsser factor which is defined as

$$K_B = \frac{4C + 2}{4C - 3} \quad (3.6)$$

The tensile strength of spring materials is obtained from [26]:

$$S_{ut} = \frac{A}{d^m} \quad (3.7)$$

where the parameters  $A$  and  $m$  is given in Table (3.2) for different spring materials. The maximum allowable torsional stress  $\tau_{all}$  is about one half of the tensile strength [26]. Some other parameters of squared and ground compression spring is obtained by:

$$\begin{aligned} \text{End Coils :} \quad N_e &= 2 \\ \text{Total coils :} \quad N_t &= N + 2 \\ \text{Free length :} \quad L_0 &= pN + 2d \\ \text{Solid length :} \quad L_s &= dN_t \\ \text{Pitch :} \quad p &= (L_0 - 2d)/N \end{aligned}$$

Material	ASTM No.	$m$	$A$ , kpsi
Music wire	A228	0.163	186
Oil-tempered wire	A229	0.193	146
Hard-drawn wire	A227	0.201	137
Chrome vanadium	A232	0.155	173
Chrome silicon	A401	0.091	218

Table 3.2: *Spring materials tensile strength*

Our design requirement are:

$$\text{Spring constant : } k = 10, 20 \text{ KN/m}$$

$$\text{Outer diameter : } D_o = 52 \text{ mm}$$

$$\text{Solid length : } L_s = 235 \text{ mm}$$

$$\text{Minimum stroke : } S_{min} = 100 \text{ mm}$$

With the wires available we made some trial to find the final shape of the spring we discuss the design of one spring here and the final design of the other, will be given in Table (3.3)

For  $k = 10$  we tried music wire material with diameter  $0.218''$ ; therefore:

$$d = 0.218'' = 5.53 \text{ mm}$$

$$D_o = 52 \rightarrow D = D_o - d = 46.46 \text{ mm}$$

$$N = 9.744 \rightarrow N = 10$$

$$N_t = 12 ; p = 22.4 \text{ mm}$$

$$L_s = 66.36 \rightarrow S = 168.8 > S_{min}$$

The dimension analysis of the spring is complete, now consider stress analysis.

$$S_{ut} = 1650 \text{ MPa} \rightarrow \tau_{all} = 821.36 \text{ MPa} \quad (\text{Equation(3.7)})$$

$$C = 8.4 \rightarrow K_s = 1.05 ; K_B = 1.16$$

$$\text{Allowable static force} = 1180 \text{ N} \quad (\text{Equation(3.1)})$$

$$\text{Allowable fatigue force} = 1070 \text{ N} \quad (\text{Equation(3.5)})$$

$$\text{Allowable Stroke} = 107 \text{ mm} > S_{min}$$

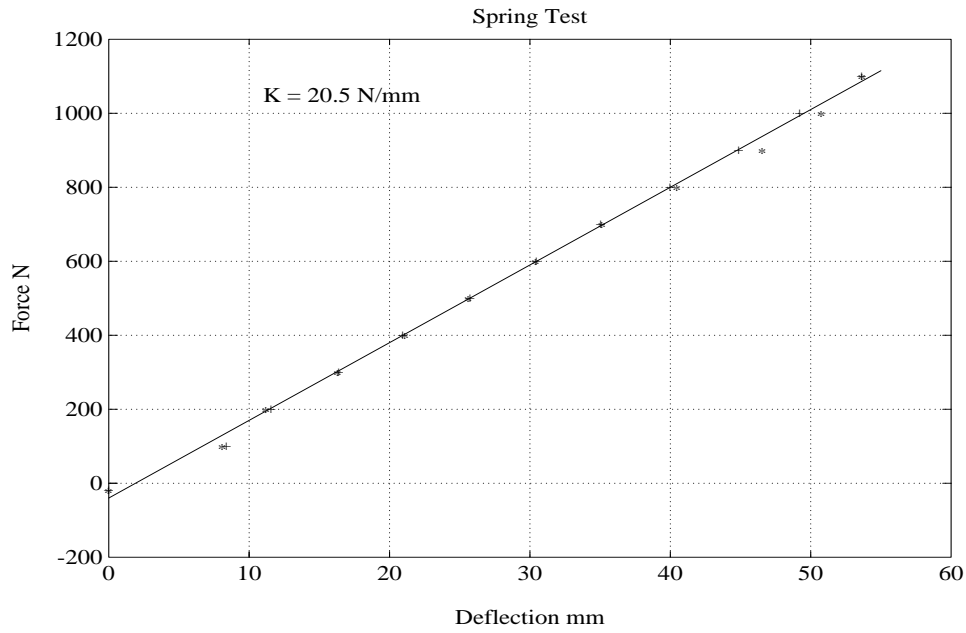


Figure 3.4: *The result of the Spring test; \* denotes compression, and + denotes decompression readings*

The allowable stroke is 107 mm, which is more than what we required. Therefore, our dimension design is acceptable. The final design for our two springs are summarized in Table (3.3).

Certainly, we don't reach to the exact values given in Table (3.3) due to the manufacturing tolerances. To feel how far we are in practice we tested the 20 KN/m spring, with a ETS

Stiffness KN/m	10	20
Material	Music wire	Chrome vanadium
Wire diameter mm	5.53	7.06
Outer Diameter mm	52	52
Free length mm	235	235
Total Stroke mm	168.6	118
Safe Stroke mm	107	118
pitch mm	22.4	15.8
weight gr	315	670
Energy/weight J/kg	180	205

Table 3.3: *Final spring design parameters*

compression machine. The final Spring rate obtained from the test was 20.5 KN/m. Figure (3.4) shows the test results, which is only %2 off. The stress analysis shows that we could reach to the solid height of the spring without getting any plastic deformation. By trying to reach to the solid height several times in the test, this claim was confirmed.

### 3.3 Leg Design

The design of the leg was recast to the design of upper leg, lower leg, and toe. The upper leg is the base containing ball screw, spring, measurement units, and the mechanical stops. Besides, we use that for preventing ball nut from rotation (Figure (3.5)). The lower leg design was tricky since it should be very light to minimize the impact losses. The spring force can be measured by a miniature load cell mounted on the top of the lower leg and the prismatic motion is produced by using five point contact Delrin rings between the lower and the upper legs. The toe is subject to the impact load and could be used to place the ground force measuring units. In our design we simply used a high compliance material as the toe, but some parallel project are working on the design of measurement units that will be assembled in the toe later. A Cartesian planariser is designed to keep the robot in a plane. There are two couples of ball bushings assembled on two couple of perpendicular shafts to get a planar motion. The alignment of the shafts and ball bushings are important to reduce the amount of rolling friction [8].

There are four sensor used in the hopper to measure the important states. Optical encoders are measuring the motor rotation angle directly, and the hopping height via a pulley and rope designed in the planariser. Spring force is measured by a miniature load cell, which is placed between the spring and lower leg, and a linear potentiometer measures the leg compression during stance (Figure (3.5)).

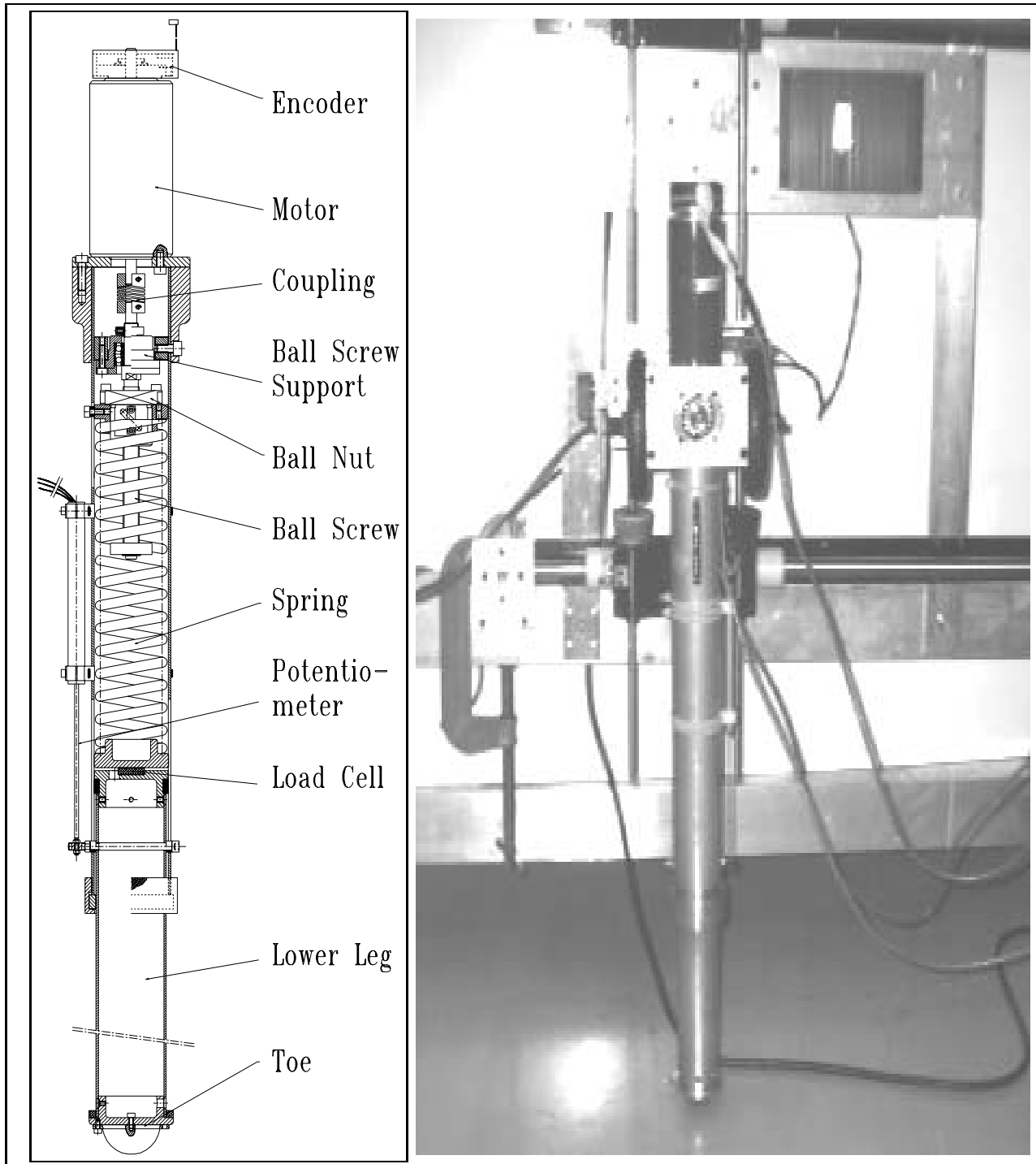


Figure 3.5: *The final leg drawing (left) and the picture of McGill Hopper (right)*

# Chapter 4

## Experiment

### 4.1 Hardware

In this chapter we will discuss the structure of the hardware, and the software of the system. In practice we have a mechanical hardware, the hopper, which has its dynamical characteristics. Some important states of the system should be measured for control purpose. The sensors data should be collected, filtered, analysed, and stored. On the other hand the control command should be fed to the servo amplifier for the motor control. This calls for an I/O board, and a software to manage and control the process. The overall structure of the system hardware is illustrated in Figure (4.1). The mechanical hardware, the hopper, interacts with measurement units and servo amplifier. In fact the input of the the hopper is the servo amplifier current, and the output of the mechanical hardware is the state signals that are measured by digital and analog sensors. The optical encoders which measure the motor rotation angle and hopping height are the digital parts, and the linear potentiometer and the load cell which measure the leg length and spring force are the analog parts of the measurement unit. The servo amplifier obtains a control command from the I/O board in the form of a voltage, and send the amplified current signal to the DC motor. The I/O board consist of A/D and D/A converters, some data acquisition and memory chips. There is a watch-dog timer mounted on the I/O board which controls the sampling time sequences. If

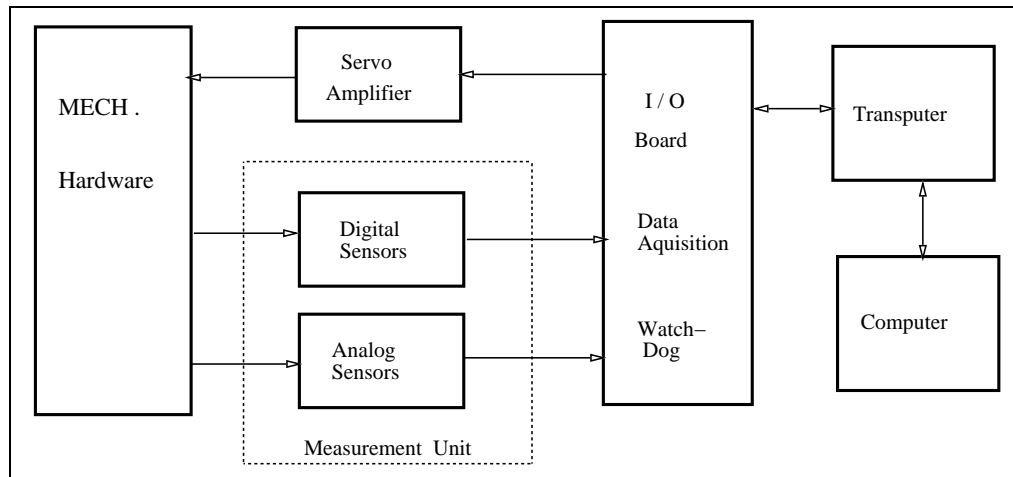
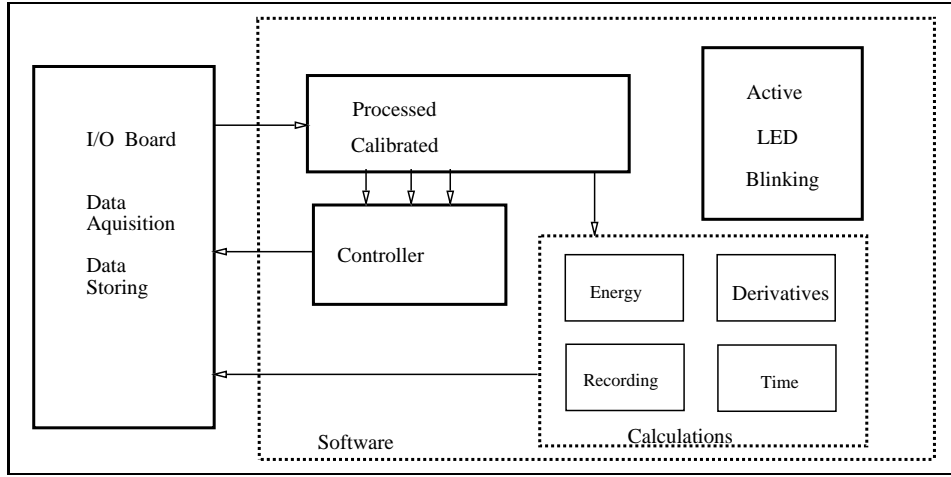


Figure 4.1: *Block diagram of the system hardware*

there is any computer failure, this part terminates the execution of the compiled code. It seems that I/O board can directly connected to the computer; however, we are using a Sun work station, and for the interactive communication, a transputer is needed to control the real time interaction. The transputer we have used has 20 MHz clock frequency CPU. In overall the memory capacity and speed of it is quite adequate for this kind of application [13]. The compiled C code will be down loaded onto the booted transputer in each experiment. The real time interaction of mechanical hardware, measurement unit, servo amplifier, I/O board, and the transputer is controlled by this code. Moreover, the required data will be stored in the transputer memories. After experiment termination, the stored data will be transmitted through the computer, for further analysis.

## 4.2 Software

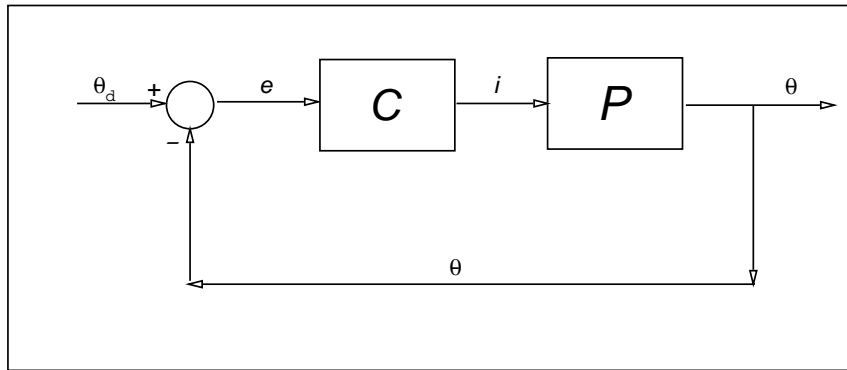
Now that the components of the hardware were explained, we can have an overlook of the software, which is written compatible with the hardware. The software has one main process, which consists of reading the data from the I/O board, processing, and calibrating them and sending them to the controller or calculation parts. In the controller parts, from the received data the control command is produced and sent to the I/O board. The output command is a digital signal. This signal is converted to the voltage command through a digital to

Figure 4.2: *Block diagram of the system software*

analog converter unit available in the I/O board. The calculation part consist of calculating the non measured states or function which are required to be stored. Moreover, the format of recording data, and the number of the recorded data is specified in this section. The calculation part also covers the time producing part. The real time of the operation is produced through converting the clock ticks of the transputer CPU to seconds. Sometimes the derivative of a data is required. This derivative can be derived using the ratio of the finite difference of two steps over the step time. This is not an ideal way to differentiate the measurement signals since it amplifies noise. A filter algorithm is considered for this respect. Assigning a filter gain  $G$ , we use two inputs for deriving the derivative: The finite difference, and the last step derivative. Suppose you want to derive the velocity signal  $v$  knowing the measured signal  $x$ . We assign weights to the two parts of the derivative algorithm, by setting a filter gain  $G$  smaller than one and with this formulation:

$$v_n = (G)v_{n-1} + (1 - G)\frac{x_n - x_{n-1}}{dt} \quad (4.1)$$

If the filter gain  $G = 0$ , then the total weight is given to the finite difference part, and if  $G = 1$  this part has zero weighting. This algorithm is successful to clean up the noises, but the accuracy of the velocity will be compromised. In the cases where the velocity is used in the control it's more precise to set filter gain near zero. The data calibration and process unit handles eight A/D channels, one D/A, and two optical encoders. The controller has different

Figure 4.3: *Block diagram of motor controller*

subroutines which will be explained in the proceeding section, and the calculation and recording part consist of calculating velocities and energies, and recording different states and data.

### 4.3 DC Motor Control

DC motor position control is a well-studied problem in the control theory. In most of the text books you can find several method of expressing position and velocity control for DC motors. In this section we want to brush up the classical theory and its limitation in experiments. Then we give a combined theory and experimental solution to the problem. Before starting the analysis we turn an account to the importance of this result to the hopper control. We develop several control ideas for injecting energy during stance. However, we didn't explain how the ball-screw will be back driven during flight. This subject is more important in the small hopping height where the flight phase is short. If the ball-screw back drive did not get complete it has a great influence in the stance phase energy injection. This study help us to establish the ball-screw back drive control law.

The Control of the DC motor then could be formulated as a position control where the amplifier current  $i$  is the input to the plant  $\mathbf{P}$  and the out put of the system  $\theta$  should regulate to the set point  $\theta_d$ . This could be accomplished via several algorithms. We tried a PD feedback control for this concern. Figure (4.3) illustrates the block diagram representation

for the problem. The objective of this analysis is to find the PD feedback gains, to give a desired damping ratio and settling time to the closed loop system.

The plant transfer function could be easily found from the nature of the system. If there is no output torque:

$$\ddot{\theta} = \frac{k_m}{J_e} i \quad (4.2)$$

where  $J_e$  is the effective moment of inertia acting on the motor and  $k_m$  is the motor constant. Therefore, the plant transfer function is:

$$\mathbf{P} = \frac{A}{s^2} ; \quad A = \frac{k_m}{J_e} \quad (4.3)$$

The PD controller transfer function is simply:

$$\mathbf{C} = k_d s + k_p \quad (4.4)$$

where  $k_d$  and  $k_p$  are the controller gains. The closed loop transfer function will be:

$$\frac{\theta}{\theta_d} = \frac{\mathbf{CP}}{1 + \mathbf{CP}} = \frac{A(k_d s + k_p)}{s^2 + A k_d s + A k_p} \quad (4.5)$$

This is a standard second order system, which can be rewritten in the form of:

$$\frac{\theta}{\theta_d} = \frac{2\eta\omega_n s + \omega_n^2}{s^2 + 2\eta\omega_n s + \omega_n^2} \quad (4.6)$$

where

$$\omega_n = \sqrt{A k_p} ; \quad \eta = \frac{1}{2} \sqrt{\frac{A}{k_p}} k_d \quad (4.7)$$

For  $\eta < 1$  the unit step response will be:

$$\theta(t) = 1 + \frac{1}{\sqrt{1 - \eta^2}} e^{-\eta\omega_n t} \{\cos(\omega_d t + \phi)\} \quad (4.8)$$

where

$$\omega_d = \omega_n \sqrt{1 - \eta^2} ; \quad \phi = \tan^{-1} \frac{\eta}{\sqrt{1 - \eta^2}} \quad (4.9)$$

The settling time  $t_s$  is defined as the time where the step response reach %95 set point. It can be easily verified from Equation (4.8) that settling time can be expressed in the form of:

$$t_s = -\frac{1}{\eta\omega_n} \ln(0.05\sqrt{1 - \eta^2}) \quad (4.10)$$

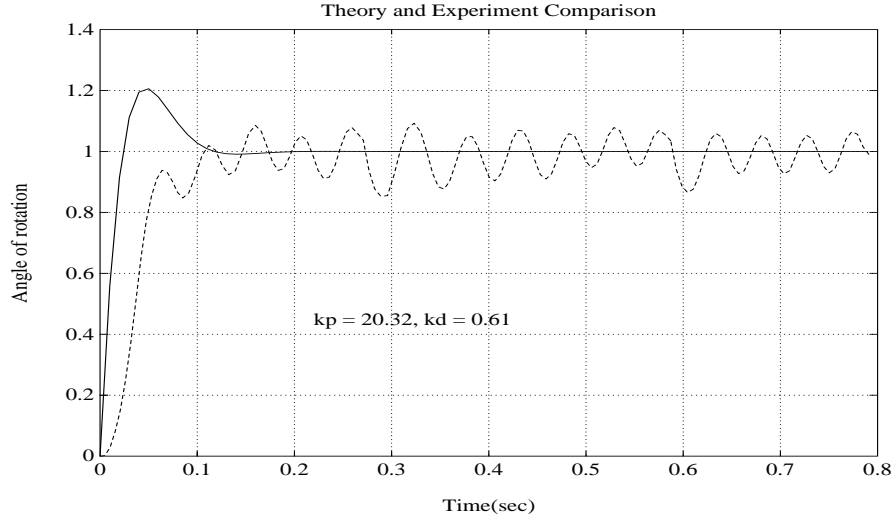


Figure 4.4: *Step Response comparison of theory and experiment*

Now the formulation is complete and by setting a desired damping ratio  $\eta$  and settling time  $t_s$  we can find the controller gains  $k_p$  and  $k_d$ . To summarize, Equation (4.10) can be written as:

$$\omega_n = \frac{3}{\eta t_s} - \frac{1}{2\eta t_s} \ln(1 - \eta^2) \quad (4.11)$$

and  $k_p$  and  $k_d$  could be assigned by:

$$k_p = \frac{\omega_n^2}{A} \quad ; \quad k_d = \frac{2\eta\sqrt{k_p}}{\sqrt{A}} \quad (4.12)$$

where  $A$  is given in Equation (4.3). There is a simulation written in Matlab, to find the step response of the system with the desired values of damping ratio  $\eta$  and settling time  $t_s$  as input and controller gain calculated from Equation (4.11) and Equation (4.12). These values of gain is given into the real controller for comparison. For  $\eta = 0.707$  and  $t_s = 0.1$  the resultant gains will be:

$$k_p = 20.3152 \quad ; \quad k_d = 0.6077 \quad (4.13)$$

Figure (4.4) illustrates the difference of the simulation and the experiment results. In the experiment we observe some oscillation around the set point which is not expected by the simulation. This is not surprising, since in the theory there was no limitation on the

current. However, in practice the amplifier has maximum current limitation. The oscillation of the experiment response is due to the high current output of the controller, which is never achieved because of the amplifier limits. Instead we have jumping from positive current bound to negative current bound, which cause the oscillation of the motor shaft.

By tuning the controller gains we observed that this phenomena is more dependent on the  $k_p$ . Therefore, we considered another strategy to calculate controller gains and avoid this instability of the response.

In the new strategy we calculate the ratio of the controller gains due to the settling time and damping requirements, then we find the optimum  $\eta$  where  $k_p$  becomes minimum and then by adjusting  $k_p$  we can find the maximum bounds of controller gains by experiment. From Equation (4.12) we can conclude:

$$\frac{k_p}{k_d} = \frac{3}{2\eta^2 t_s} - \frac{1}{4\eta^2 t_s} \ln(1 - \eta^2) \quad (4.14)$$

on the other hand we get minimum  $k_p$

$$\frac{\partial k_p}{\partial \eta} = 0 \quad (4.15)$$

Since  $\omega_n \propto k_p$ , This is equivalent to:

$$\frac{\partial \omega_n}{\partial \eta} = \frac{1}{t_s} \left( \frac{-3}{\eta^2} + \frac{1}{1 - \eta^2} + \frac{1}{2\eta^2} \ln(1 - \eta^2) \right) = 0 \quad (4.16)$$

Notice that this condition depends only on  $\eta$ , and by solving it we reach to:

$$\eta_{opt} = 0.8893275 \quad (4.17)$$

By setting  $\eta = \eta_{opt}$  Equation (4.14) simplifies to:

$$k_p = 2.39124 \frac{k_d}{t_s} \quad (4.18)$$

The upper bound of the gains should obtained by experiment. We found the upper bound of  $k_d$  to be 0.01 for the motor. It should be mentioned that by bounding controller gains we are sacrificing our requirements, which in this case turns out to be the settling time.

This strategy is used with  $t_s = 0.1$  in practice. Figure (4.5) illustrates the comparison

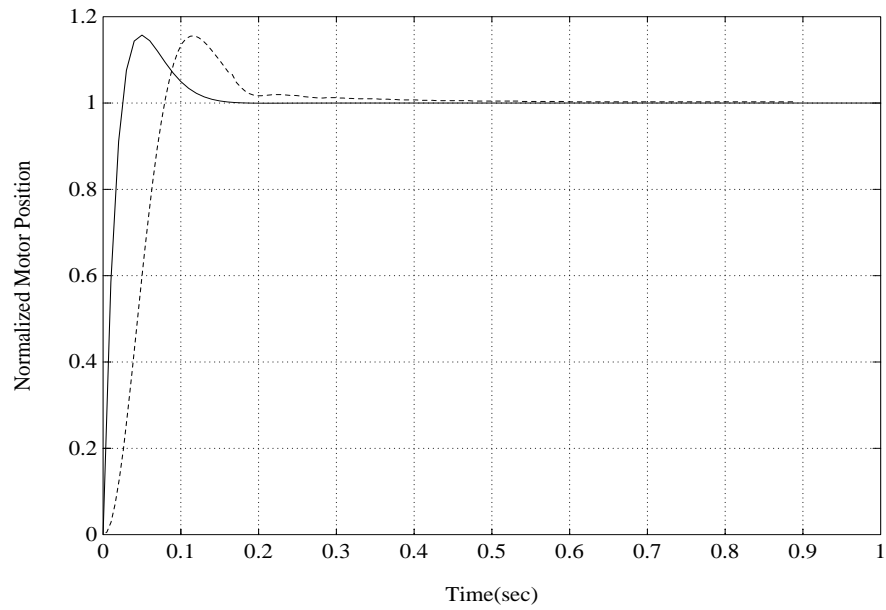


Figure 4.5: *Step Response comparison of theory without current limit and experiment with current limit*

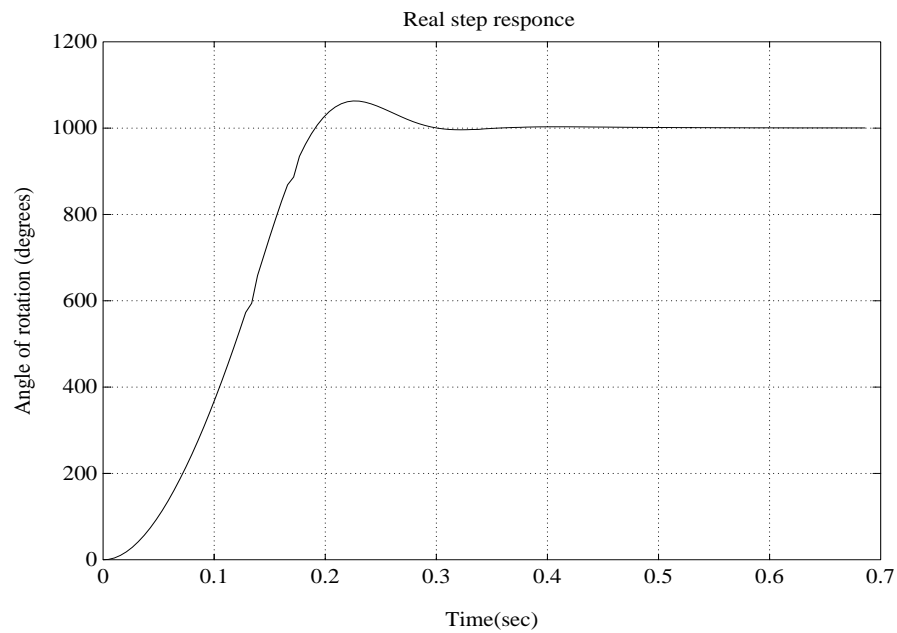


Figure 4.6: *Step response of the system for a large initial error*

of theory and experiment. In the theory we didn't consider current limit; however, in the experiment we bound controller gains to  $k_d = 0.001$  and  $k_p = 0.024$ . The responses are quite similar, except the settling time. The lose of this requirement is because of controller gain limits. This strategy is used in a large step response of the system. In this test we want to check the reliability of this method for different inputs. As it is shown in Figure (4.6) the shape of the response is quite similar to that of Figure (4.5), but the settling time is longer. We end up this discussion here and used this strategy to derive controller gains for flight phase control of the hopper. This analysis could be continued for different control methods.

## 4.4 Hopping Experiments

In this section we give the result of the experiment for three cases. Passive hopping, open loop torque control, and modified torque control.

### 4.4.1 Passive Hopping

Let us first examine Passive hopping. This experiment is valuable, because we can understand the physics of the hopper, and estimate the unknown parameters in the simulations. On the other hand it is a good test to check the data acquisition units, and the software. Figure (4.7) illustrates the result of the passive hopping experiment. The system loses its energy very fast and after 1.5 second it becomes stationary. Comparing this result to the result of simulation, we notice that the simulation parameters are not assigned correctly. In fact the friction losses are much more than we expected. This results helps us to estimate the unknown variables in the simulation and to verify the simulation results. This will be done in Section (4.5). The other thing which is also important, is the amount of energy loss during stance and flight. These energy loss is almost four Joules for both first stance phase, and first flight phase. This result is also surprising, since we had some sliding motion in the stance phase, and we expected to have most energy losses during stance. However, the experiment shows that the flight friction which is due to the planariser system (the system

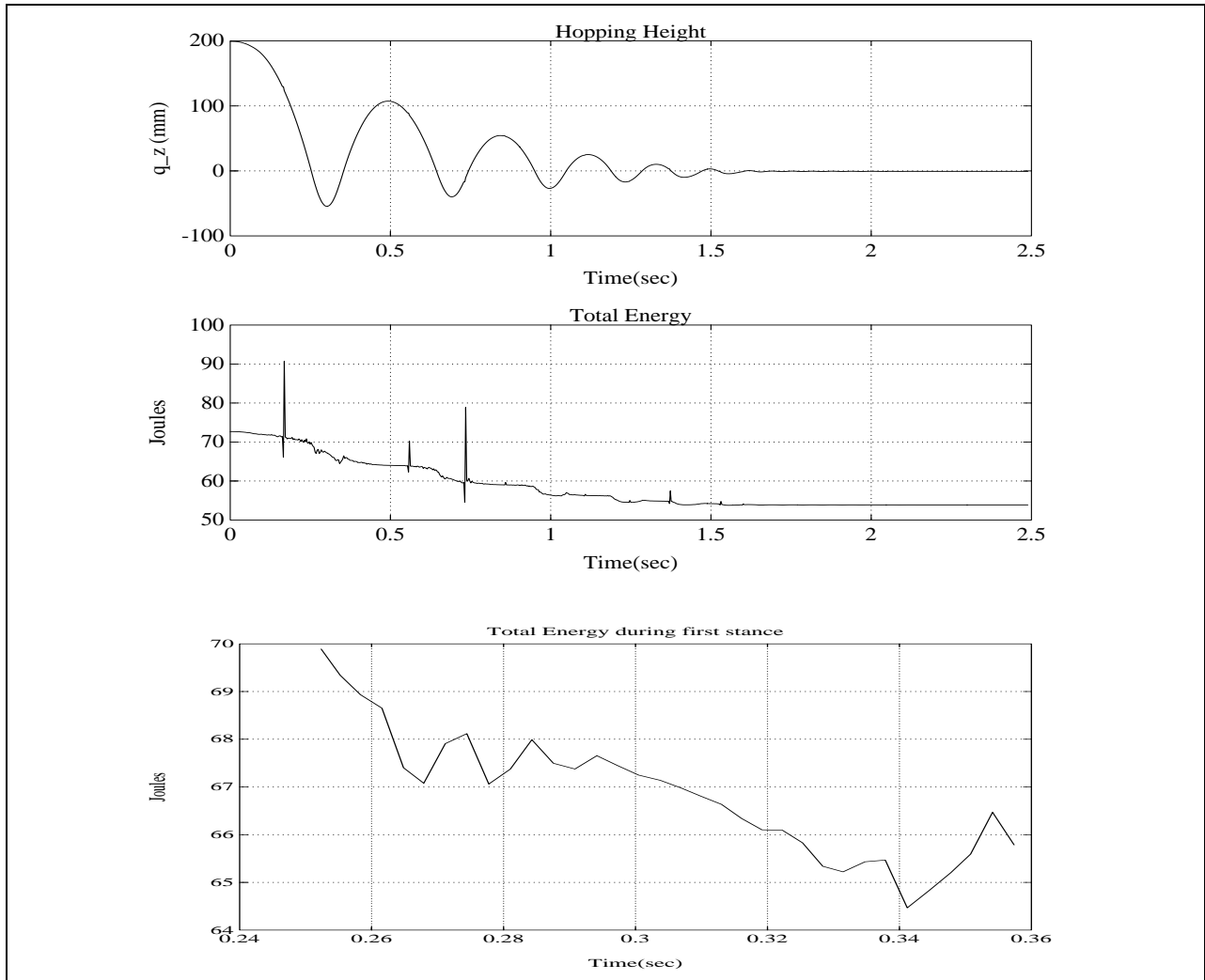


Figure 4.7: *Hopping height, total energy, and stance energy of Passive Hopper*

which keeps the hopper in the plane) has the same order of importance. In fact the longer flight time, magnifies the flight phase energy losses. It should be considered that the glitches observed in the energy plot is due to a electronic problem of I/O board, which is under study.

#### 4.4.2 Open Loop Torque Control

To examine our motor power to the unexpected losses, we tried a simple control algorithm. By giving the maximum permissible current to the motor during the stance phase, we expect

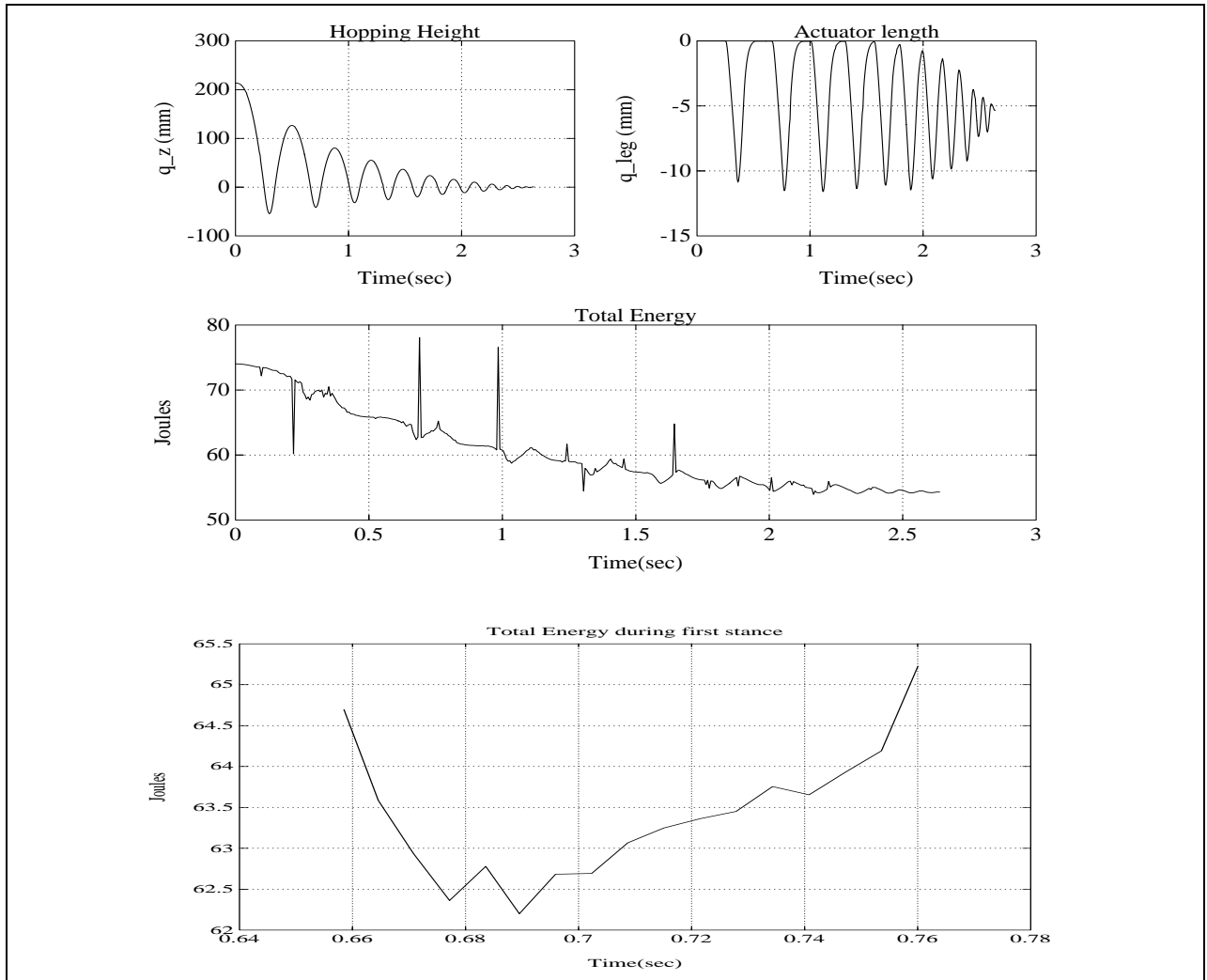


Figure 4.8: *Hopping height, actuator length, total energy, and stance energy of open loop torque control*

to gain almost maximum power of the motor. During flight we use the PD control explained in Section (4.3) to drive the ball screw back to its initial position. Figure (4.8) illustrates the experiment results. Hopping time becomes almost twice but still we didn't reach to the reentrant trajectory. The PD control during flight is successful for the few first steps, and by decreasing the flight time it can not back drive the ball screw perfectly. This is acceptable since we require at least 20 mm stable hopping height and within this region the PD control is successful. We claimed that the energy injected to the system is almost maximum. This could be improved by slightly changing the control law. We know that DC motors are

severely torque limited, this became worse when there is no time for the motor to reach to its nominal speed. In our case, the stance time is just 100 ms which is very short for the motor to reach to its nominal speed. Therefore, the power output of the motor is even less than one third of the nominal power. We can improve this situation with a modified torque control law.

### 4.4.3 Modified Torque Control

In this case we are using the maximum torque of the motor, not only during stance phase, but also for a variable time interval before stance phase. This time interval should be variable to be robust for different hopping height. By a series of trials we set the starting instance of control law be the instance where the hopper has %40 of its maximum hopping height. In the first hopping cycles this time intervals are relatively long, but in the last hopping cycles it becomes short. The effect of this modification to the system is that the motor reaches to higher velocity, and therefore, we gain higher power during stance phase. On the other hand, by this modification, the spring compression get more and it has two disadvantages. First the stance time get shorter, and second there will be more amount of stored energy in the spring at the lift off. Since the leg length is constant in our design, this amount of energy will be lost after lift off.

Figure (4.9) illustrates the result of this case. By comparison to Figure (4.8) we observe that the hopping duration changes slightly. On the other hand the added energy during stance has increased about two Joules. However, this approach is also not complete to bring this situation to reentrant hopping.

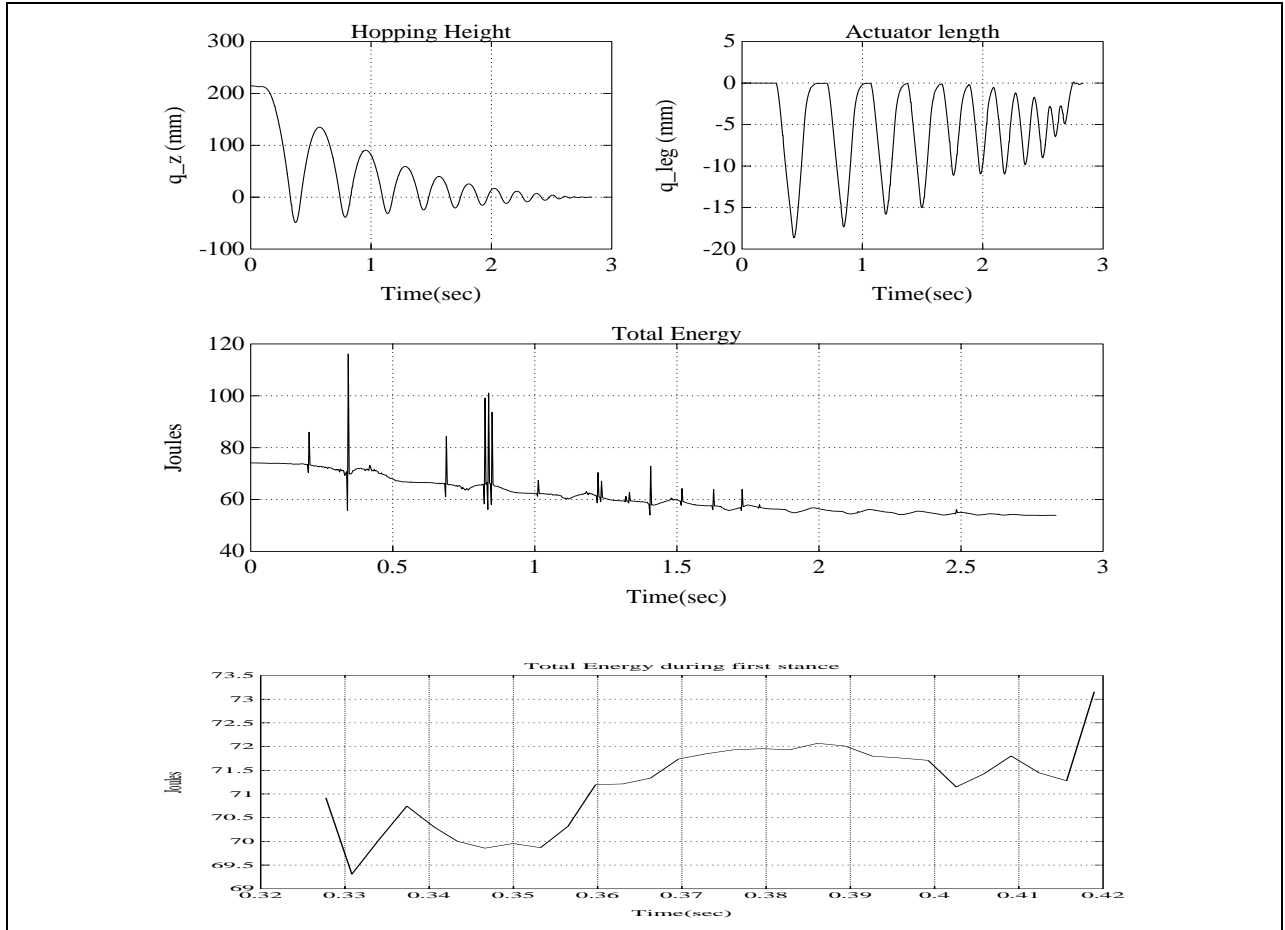


Figure 4.9: *Hopping height, actuator length, total energy, and stance energy of modified torque control*

## 4.5 Simulation Verification

In Section (4.4) we observed that the simulation assigned parameters, which are basically the friction forces, are underestimated, and this cause the difference between the simulation results obtained in Section (2.1) and the experimental results in Section (4.4). This calls the purpose of this section, which is to verify the unknown parameters in the simulation by the use of experimental results and to compare the simulation and experiment. This approach help us to state two important result. First, to verify that how the simplified model can forecast the real physics of the system, and second to use the simulation for further design modification. This parameter identification can be improved to a more formal identification

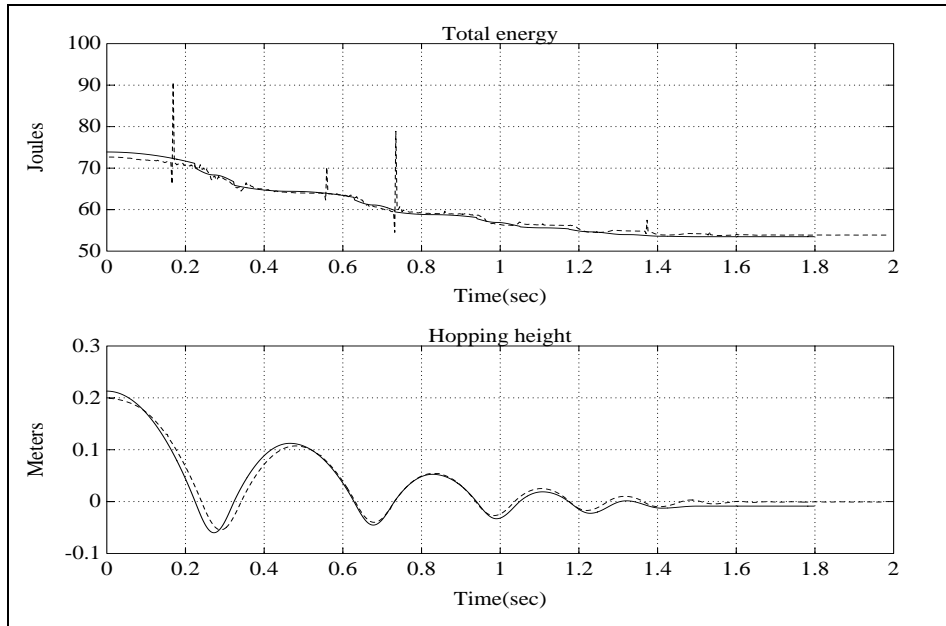


Figure 4.10: *Comparison of simulation and experiment; Passive hopping*

process, which could be pursued online concurrently with the robot controller improvement. In the parameter identification we divide the discussion into two parts. Passive hopping will be first studied to gain the friction forces during stance and flight, and actuated hopping will be studied to estimate the motor realistic characteristic during stance time. In each part we focus our attention to overall matching of total energy, and hopping height. Moreover, we will study more precisely the first stance phase total energy, hopping height, and hopping velocity. Unfortunately the resolution of our instrumentation is not high enough to make a concrete quantitative conclusion in some special cases; however, the qualitative comparison and the simulation validation can be examined in this study.

### 4.5.1 Passive Hopping

For the passive hopping, we want just to identify the friction forces during stance and flight. All the dimension and masses are assigned in the simulation by direct measurement or manufacturer's specification data. After some trial we reach to the point that viscous

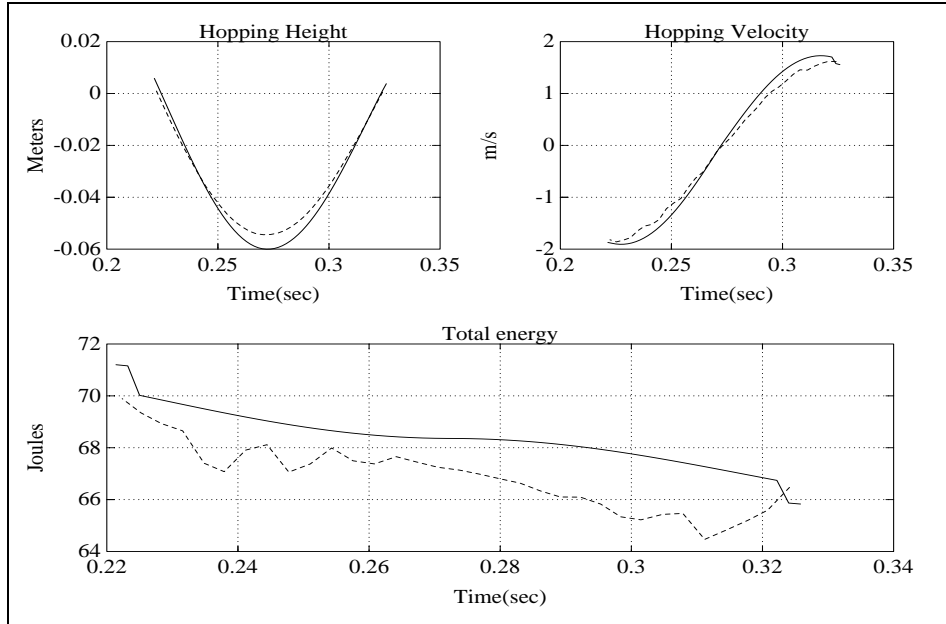


Figure 4.11: *Simulation and Experiment result during the first stance; Passive hopping*

friction can be neglected, and the dry friction forces in the stance and flight phase are:

$$\mathbf{F}_{fr} = 28 \text{ N} \quad ; \quad \mathbf{F}_{fr,p} = 13 \text{ N} \quad (4.19)$$

Since the stance friction is the total friction due to the sliding parts and planariser during stance, the sliding part friction will be 15 N which is in the same order of planariser friction force. This amount of friction is three times more than that we expected before the experiment, and this is the main reason why the reentrant trajectory is not achievable. In the design of the planariser, two ball bushings and two shafts is considered, which should be parallel. The reason of the friction there could be the misalignment of the shafts or the moment exerted by the hopper on the ball bushings. The friction can be reduced by precise alignment of the shaft.

Figure (4.10) illustrates the overall comparison of the total energy and hopping height. In this scale the results are very compatible. Figure (4.11) shows the result of simulation and experiment for the first stance phase. Quantitative behaviour of the results are quite similar, but, because of the non sufficient instrumentation resolution, especially the derivative of the measured states, we can not give a concrete qualitative comparison between the results.

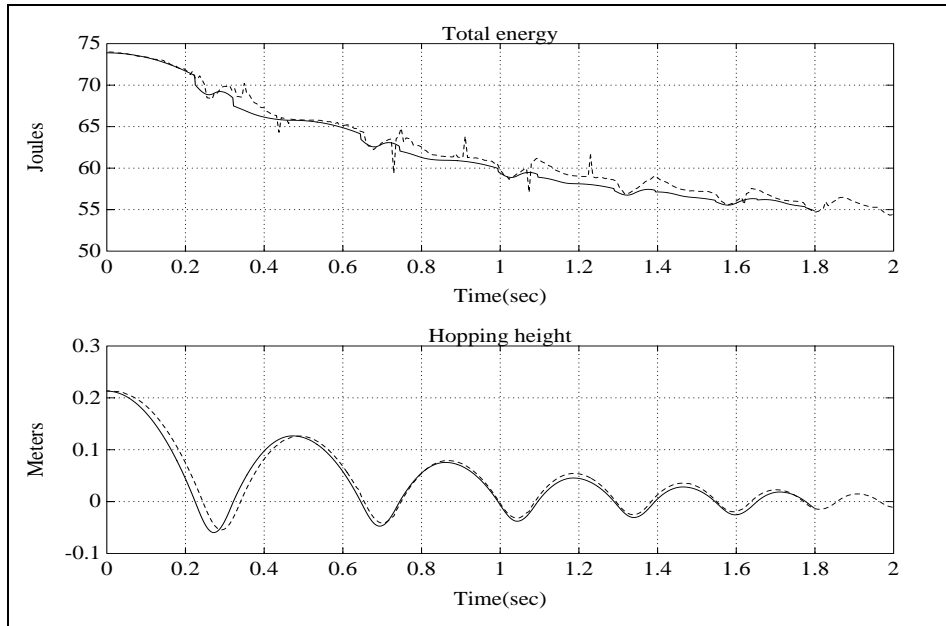


Figure 4.12: *Comparison of simulation and experiment; Actuated case*

### 4.5.2 Actuated Hopping

Now that the friction forces are identified we want to compare the open loop torque control law of the simulation to the real experiment. For this case the result of Section (4.4.2) is considered for comparison. The only parameters that could be adjusted are the ball nut mass and the constant torque input for the control law, since they are not measurable. We know that the spring is not mass less and a portion of its mass should be added to the ball-nut weight which is 250 gr. On the other hand since the motor is accelerating from rest during the stance phase the stall torque of the motor is not achievable in low speeds. By some trials we found that these amount are suitable for our case.

$$\tau = 0.62 \text{ N.m} \quad ; \quad m_s = 300 \text{ gr} \quad (4.20)$$

Figure (4.12) illustrates the overall comparison of the simulation and experiment results for this case. Again we observe a very nice compatibility between the result. Notice the way we are losing energy during the flight phase, which is similar in the experiment and simulation.

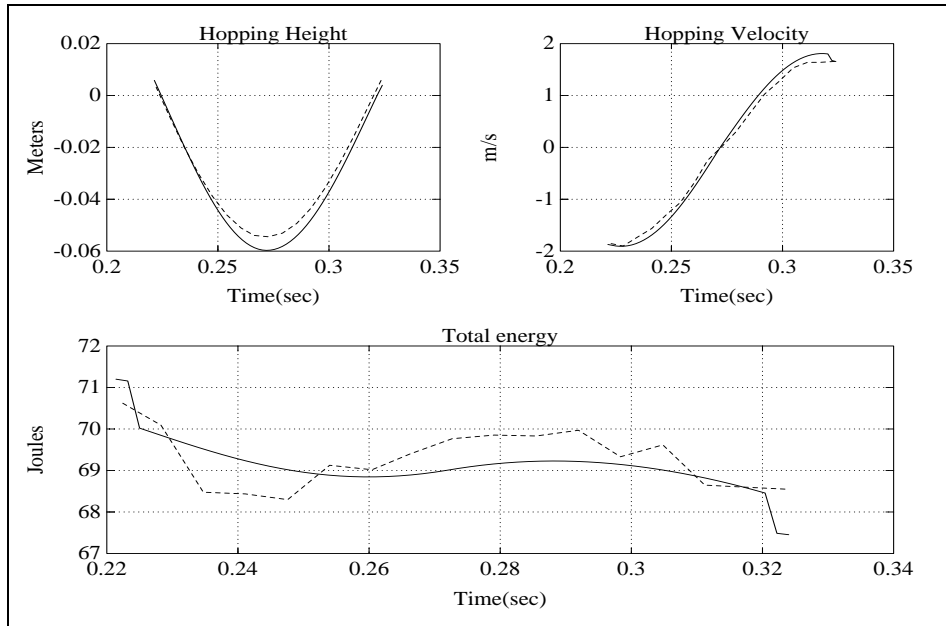


Figure 4.13: *Simulation and Experiment result during the first stance; Actuated case*

Since the flight period is more than stance period the friction forces during the flight will be more important to reduce than the stance friction forces. The hopping height in this case is even better matched to the simulation than the passive case. Figure (4.13) illustrates the comparison in the first stance time. If we regard a smooth curve for the experiment then the shape of the total energy during the stance were very close to the simulation.

This results verifies that the simplified model used in the simulation is good enough to analyse the system and predict its trajectory. For the modified torque control law implemented in the experiment, namely Section (4.4.3), the amount of the motor torque increases to 1 N.m which is better compared to Equation (4.20). This confirms again the validity of the simplified model.

## 4.6 Design Modification

Refer to the experimental result given in Section (4.5), we concluded that it is necessary to reduce the friction, to achieve a stable limit cycle and reentrant behaviour. Also we observed that the friction losses during flight phase are unexceptionally high and of the same

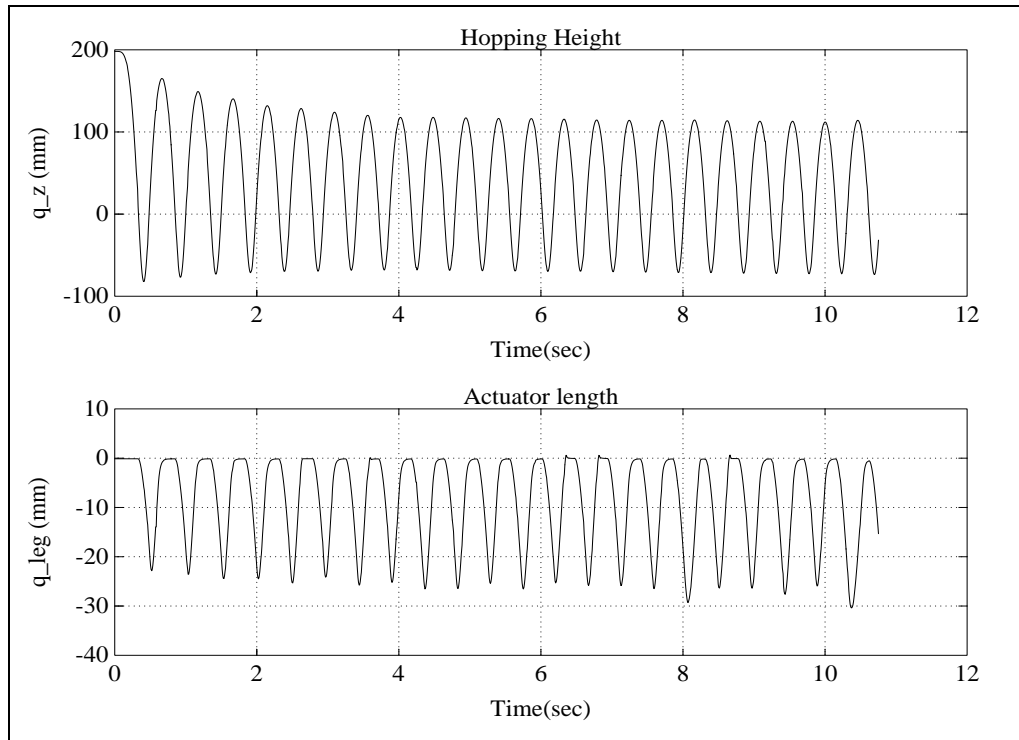


Figure 4.14: *Stable hopping experiment, starting from large hopping height*

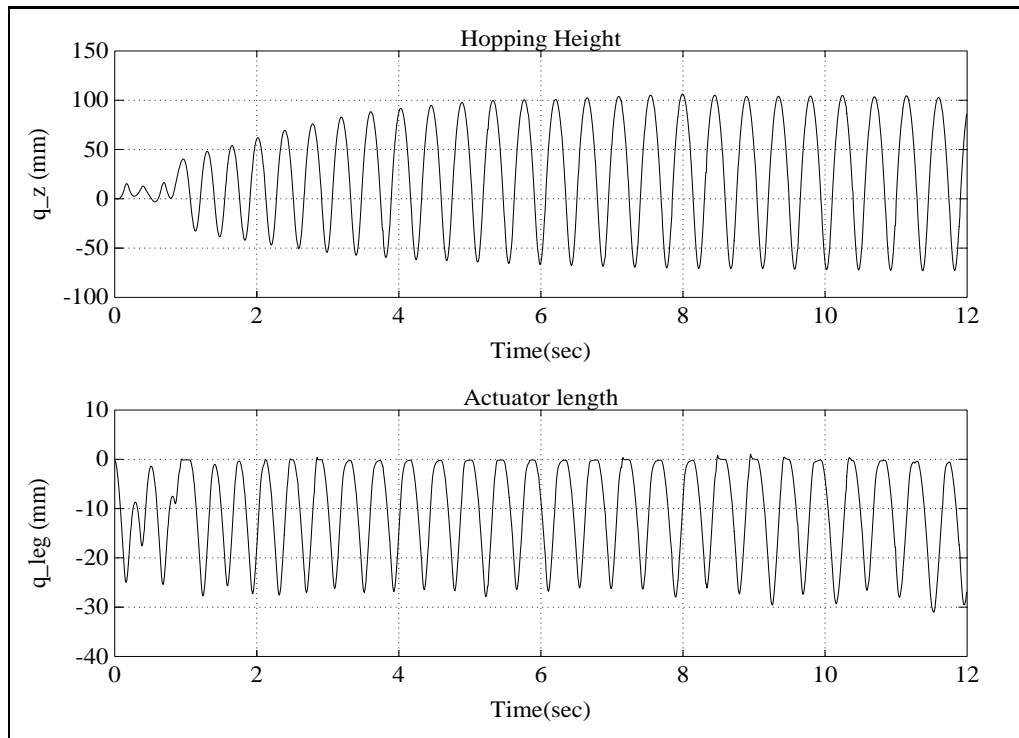


Figure 4.15: *Stable hopping experiment, starting from small hopping height*

importance of the stance phase friction. My fellow graduate student, Pedro Grigorio, who took over the project, was able to reduce the amount of friction by a series of modification in design. The problem was caused mainly by excessive energy dissipation in a pulley arrangement for the vertical height sensor. The alignment of the ball bushing shafts, paying attention to the roundness of the lower leg tube, and some electronic debugging of the I/O board were the other modification done on the system.

We bring some of the successful experimental results here. In this experiments, open loop torque control is used in the stance phase, and PD control is used in the flight phase. Figure (4.14) shows the stable hopping experiment for the robot. Hopper starts at the larger hopping height and stabilized itself at the desired hopping height. To examine the power of the motor and the control law, we started the hopping at a very small hopping height. Figure (4.15) shows how the hopper gain its desired hopping height. This experiments shows how the simple control algorithm can stabilize the system, and motivates the implementation of other purposed control strategies. This is the current research on the hopper.

# Chapter 5

## Conclusion and Future Work

### 5.1 Conclusion

There is a theoretical-practical research done on the one-legged hopping robot. The theoretical part is divided into the modelling and simulation of vertical hopper model including actuator dynamics, and one-legged passive hopping model. Moreover, different controllers presented and examined in simulation for the vertical hopper. For the experimental part the first step was the design and construction of the hopper, which was based on the results obtained from different simulations. The development of computer control hardware was done regarding the nature of the application. Different measurement units, data acquisition, input output board, and memory units accompanying the transputer and a Sun work station were assembled for the experiment. In the experimental part, the computer control of a DC motor was studied and the necessary modification to the theoretical results, were developed. Then hopping experiments were studied, with a simple open loop torque control law during stance phase, and a PD control law during flight. By improving the control law maximum possible motor power during the short stance time was gained.

There are some differences between the theoretical and experimental results. Gaining reentrant hopping trajectory for the hopper was not possible in the experiment, although we achieved it in the simulation. This motivated for the simulation verification, and parameter

identification, which was done by two steps. The unknown parameter of the system, which were friction forces, were identified from passive hopping experiment, and the power characteristics of the motor was identified by actuated experiments.

The unexpected friction, especially during flight phase, cause the fast decay of hopping height of the robot. Even the maximum power injected to the system couldn't overcome this amount of losses. This calls some modification in the design.

However, there are some important aspects studied by the experiments.

- The Simulation results are valid for passive and actuated hopping, and this can be used for further design modification or system analysis. This means that the simplification made for the simulation was reasonable and the simulation results could closely predict the real behaviour of the system.
- Regarding the simulation results if we reduce the friction forces to the half of their present values, the reentrant hopping is guaranteed.
- The validity of the simulation predicts the success in the different model-based control laws. Moreover, simulation could be used as a yardstick to compare different control algorithms more quickly than experiment.
- The complex nature of the system can be analysed by looking through the simulation results. Different compromises for detail design of the system and further modification can be studied in this way.
- The design of the leg is examined by almost 250 experiments, and still the leg is in good condition. Therefore, it could be used with small modification for more-legged robots.
- The software and hardware are ready, and reliable for further hopping experiments.

## 5.2 Future Work

To continue this work, which was essentially the starting point of a long term research, the further suggestion may help the new researchers in this field. For starting this subject, some reference materials especially [21], [27], and [22] could be studied. Then Chapter 2 especially

Section (2.1) will help to understand the main idea of the hopper control. Chapter 4 will help the reader to start the experiments faster. For the hopping experiments we suggest the following items:

**•Reducing Friction**

This can be done by a precise alignment of the planariser shafts and ball bushings. Moreover, it could be some mechanism designed to reduce the amount of moment exerted to the ball bushings. For the flight phase friction regular lubrication and checking is important.

**•DC Motor Control**

For first hardware experiments we suggest to implement simple DC motor control algorithms and compare the result with theory (Section (4.3)). In this respect, LQR controller and computer controlled analysis of PID control and LQR control could be studied.

**•Hardware and Software Adjustment**

Although the hardware and software are working good enough to implement some control laws, they need some adjustment and modification. There are two main points in this respect that should be considered. Time synchronisation, and constant time step for the software should be obtained, and the electronic problem of the encoder readings should be fixed.

**•Further Experiments**

After overcoming the friction problem two steps could be followed for vertical hopping. Implementing the control algorithms developed in Section (2.1), and developing new control algorithms and examining them in practice. Moreover, this study could then completed by designing the trunk actuation system and studying the stability and reentrant behaviour of the planar hopper toward the model-based controls similar to the vertical hopper.

# Bibliography

- [1] Interelectric AG. *Maxon Motor*. Form 001-92, Sachseln, Switzerland, 91/92.
- [2] Barns Group Inc. Associated Spring. *Design Handbook*. Form No. 515 10M-MM-12/86, Bristol, USA, 1991.
- [3] Barns Group Inc. Associated Spring. *SPEC; Stock Springs, Spring Washers, and Retaining Rings*. Form No. 700C 4000 SP11/91 30581, Bristol, USA, 1991.
- [4] V.S. Gurfinkel. Walking robot with supervisory control. *J. Mechanism and Machine Theory*, (16):31,36, 1981.
- [5] S. Hirose. A study of design and control of quadruped walking machine. *International J. Robotics Research*, (3):113,133, 1984.
- [6] J. M. Hollerbach, I. W. Hunter, and J. Ballantyne. A comparative analysis of actuator technologies for robotics. In O. Khatib, J. J. Craig, and T. Lozano-Perez, editors, *The Robotics Review 2*. MIT Press, Cambridge, MA, 1991.
- [7] I. W. Hunter and S. Lafontaine. A comparison of muscle with artificial actuators. *Proc. Sensors and Actuators*, 1992.
- [8] Thomson Industries. *Advanced Linear Motion Systems*. Form TI-001A, New York, USA, 1991.
- [9] D. E. Koditschek and M. Buehler. Analysis of a Simplified Hopping Robot. *Int. J. Robotics Research*, 10(6), Dec 1991.

- [10] V. V. Lapshin. Vertical and horizontal motion control of a one-legged hopping machine. *Int. J. Robotics Research*, 11(5), 1992.
- [11] R. B. McGhee. Vehicular legged locomotion. *Advances in Automation and Robotics*, 1985.
- [12] T. A. McMahon. *Muscles, Reflexes, and Locomotion*. Princeton University Press, Princeton, 1984.
- [13] SGS-Thomson Microelectronics. *TCPLink Hardware Transputer Architecture*. Form 72 TRN 229 01, USA, 1991.
- [14] D.I. Miller. Resultant lower extremity joint moments in below-knee amputees during running stance. *J. Biomechanics*, 20(5):529,541, 1987.
- [15] H. Miura and I. Shimoyama. Computer control of an unstable mechanism. *J. Fac. Eng.*, (17):12,13, 1980.
- [16] H. Miura and I. Shimoyama. Dynamic walk of a biped. *International J. Robotic Research*, (30):60,74, 1984.
- [17] R.S. Mosher. Test and evaluation of a versatile walking track. In *Off Road Mobility Research*, pages 359,379, Washington, 1968. International Society for Terrain Vehicle Systems.
- [18] E. Muybridge. *Animals in Motion*. New Dover Edition, Dover Publications, Inc., New York, (first published in 1899), 1957.
- [19] K. V. Papantoniou. Electromechanical design for an electrically powered, actively balanced one leg planar robot. In *IROS*, Osaka, Japan, 1991.
- [20] M. H. Raibert. Dynamic stability and resonance in a one-legged hopping machine. *4th Symp. Theory and Practice of Robots and Manipulators*, 1981.
- [21] M. H. Raibert. *Legged Robots That Balance*. MIT Press, Cambridge, MA, 1986.

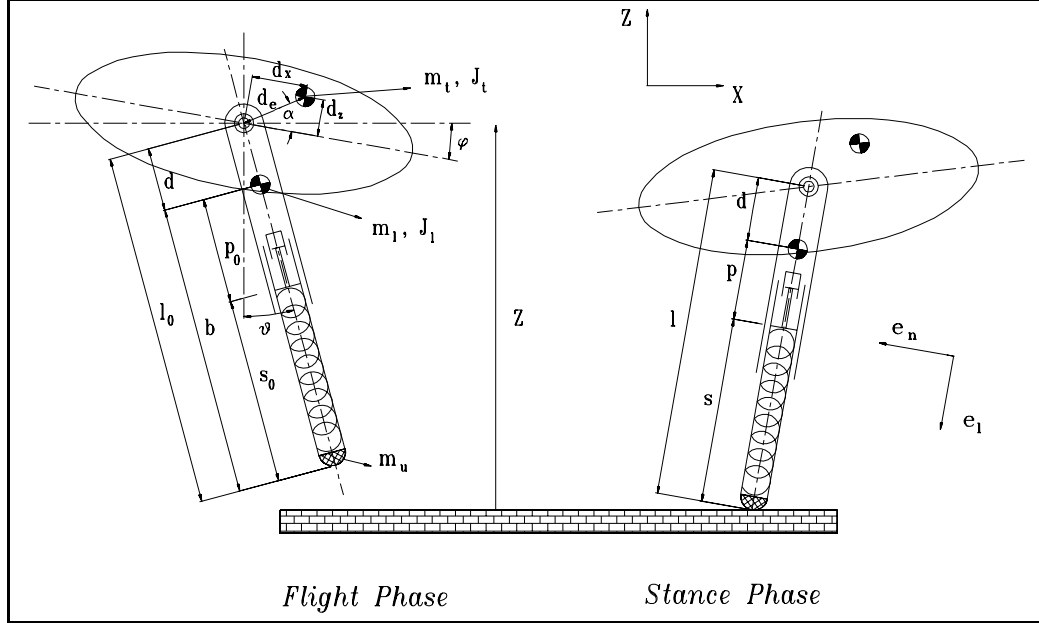
- [22] M. H. Raibert, et al. Dynamically stable legged locomotion. Technical Report CMU-LL-4-1985, Carnegie-Mellon University, Pennsylvania, USA, 1985.
- [23] M. H. Raibert, et al. Dynamically stable legged locomotion. Technical Report 1179 (LL-6), MIT AI Laboratory, 1989.
- [24] J.F. Schaefer and R.H. Cannon. On the control of some unstable mechanical systems. *International Federation of Automatic Control*, (6c):1,13, 1966.
- [25] Nippon Seiko. *Precision Machine Parts; Linear Motion Products*. Pr. No. A1371b, Japan, 1991.
- [26] J. E. Shigley and C. R. Mischke. *Mechanical Engineering Design*. McGraw-Hill Company Inc., New York, 1989.
- [27] C.M Thompson and M.H. Raibert. Passive dynamics running. In V. Hayward and O. Khatib, editors, *International Symposium of Experimental Robotics*, New York, 1990. Springer-Verlag.

# Appendix A

## General Planar Dynamics

Considering the models given in Chapter 2, there are a lot of simplifications made to reach to a compact formulation. This is desired in practice since we want to discover some new methods for control. However, it seems necessary to have a more complete model and compare its complexity to the previous models. In this section we introduce a planar one-legged running robot, where the unsprung mass is not neglected. Moreover the center of masses are not at the hip joint, and the friction force is considered in this case as dry and viscous frictions. A schematic of the model is illustrates in Figure (A.1). The system consist of three masses. Trunk has a mass of  $m_t$  with an offset of  $d_x$  and  $d_z$  from the hip joint, and a total moment of inertia  $J_t$  around the center of mass. The leg has sprung mass  $m_l$  and unsprung mass  $m_u$  where the latter is considered as a point mass in the toe. The total length of the leg is fixed; however, the initial preload of spring could be adjusted. The actuator operates in series with the spring. The total movement of the actuator is represented by  $p$ , where  $p_0$  is the initial value of it. The actuator and spring act together to lengthen and shorten the leg, in order to exert forces between the foot and the body. Mechanical stops keeps the stretched length of leg constant.

Two springs acting on the system. The spring  $k_h$ , lies at a distance  $r_h$  from the hip joint, and store the swinging motion energy. A compression spring connects the actuator to the unsprung mass. The rest length of spring is  $s_0$  and its length in the stance phase will shorten

Figure A.1: *The general planar running robot model*

to  $s$ . the leg length at rest is  $l_0$  and the center of mass of sprung mass lies at distance  $b$  from the toe.

## Dynamics

In this case we use Lagrange method to derive the equations of motion. The independent states during stance are  $x, z, \theta, \phi$  and their derivatives; however, in the flight they reduce to  $l, \theta, \phi$  and their derivatives. Therefore, we divide this section into stance and flight sections. The approaches are the same; we should derive the linear and angular velocities of the masses and their elevation from a reference level. Then finding Lagrangian from  $\mathbf{L} = \mathbf{T} - \mathbf{V}$ , where  $\mathbf{T}, \mathbf{V}$  are kinetic and potential energies of the system. Then using the Lagrange, namely

$$\frac{d}{dt} \left( \frac{\partial \mathbf{L}}{\partial \dot{q}_j} \right) - \frac{\partial \mathbf{L}}{\partial q_j} = \mathbf{Q}_j \quad (\text{A.1})$$

where  $\mathbf{Q}_j$  is the generalized force and is obtained using virtual work method, the state equations will be obtained.

**Flight Phase**

Refer to Figure (A.1) the velocities of three masses are given by:

$$\begin{aligned}\mathbf{v}_t &= (\dot{x} + (d_x \sin \phi + d_z \cos \phi)\dot{\phi})\hat{i} + (\dot{z} + (d_z \sin \phi - d_x \cos \phi)\dot{\phi})\hat{k} \\ \mathbf{v}_l &= (\dot{x} + d \cos \theta \dot{\theta})\hat{i} + (\dot{z} + d \sin \theta \dot{\theta})\hat{k} \\ \mathbf{v}_u &= (\dot{x} + l_0 \cos \theta \dot{\theta})\hat{i} + (\dot{z} + l_0 \sin \theta \dot{\theta})\hat{k}\end{aligned}\tag{A.2}$$

Therefore total kinetic energy is:

$$\begin{aligned}\mathbf{T} &= \frac{1}{2}m_t\{(\dot{x} + (d_x \sin \phi + d_z \cos \phi)\dot{\phi})^2 + (\dot{z} + (d_z \sin \phi - d_x \cos \phi)\dot{\phi})^2\} + \\ &\frac{1}{2}(J_t\dot{\phi}^2 + J_l\dot{\theta}^2) + \frac{1}{2}m_l\{(\dot{x} + d \cos \theta \dot{\theta})^2 + (\dot{z} + d \sin \theta \dot{\theta})^2\} + \\ &\frac{1}{2}m_u\{(\dot{x} + l_0 \cos \theta \dot{\theta})^2 + (\dot{z} + l_0 \sin \theta \dot{\theta})^2\}\end{aligned}\tag{A.3}$$

The potential energy is given by:

$$\mathbf{V} = m_t g(z + d_x \sin \phi + d_z \cos \phi) + m_l g(z - d \cos \theta) + m_u(z - l_0 \cos \theta)\tag{A.4}$$

Regarding  $\mathbf{L} = \mathbf{T} - \mathbf{V}$ ; and using Lagrangian, namely Equation (A.1) for four states we reach to :

$$\begin{aligned}m\ddot{x} + m_t\ddot{\phi}d_e \sin(\phi + \alpha) + m_t\dot{\phi}^2 d_e \cos(\phi + \alpha) + \\ (m_l d + m_u l_0)(\cos \theta \ddot{\theta} - \dot{\theta}^2 \sin \theta) = F_{fr,x} \text{sign}(-\dot{x})\end{aligned}\tag{A.5}$$

$$\begin{aligned}m\ddot{z} - m_t\ddot{\phi}d_e \cos(\phi + \alpha) + m_t d_e \dot{\phi}^2 \sin(\phi + \alpha) + (m_l d + m_u l_0) \times \\ (\sin \theta \ddot{\theta} + \dot{\theta}^2 \cos \theta) + mg = F_{fr,z} \text{sign}(-\dot{z})\end{aligned}\tag{A.6}$$

$$\begin{aligned}(m_t d_e^2 + J_t)\ddot{\phi} + m_t(d_e \ddot{x} \sin(\phi + \alpha) + d_e \dot{x} \dot{\phi} \cos(\phi + \alpha) - d_e \ddot{z} \cos(\phi + \alpha) + \\ d_e \dot{z} \dot{\phi} \sin(\phi + \alpha)) - m_t \dot{\phi}(\dot{x} d_e \cos(\phi + \alpha) + \dot{z} d_e \sin(\phi + \alpha)) + m_t g d_e \cos(\phi + \alpha) + \\ c_h r_h(\dot{p}_h - r_h(\dot{\theta} - \dot{\phi})) + k_h r_h(p_h - p_{h0} - r_h(\theta - \phi)) = T_{fr} \text{sign}(-\dot{\phi})\end{aligned}\tag{A.7}$$

$$\begin{aligned}(m_l d^2 + m_u l_0^2 + J_l)\ddot{\theta} + (m_l d + m_u l_0)((\ddot{x} + \dot{x}\dot{\theta}) \cos \theta + (\ddot{z} - \dot{z}\dot{\theta}) \sin \theta) + \\ (\dot{x} \sin \theta - \dot{z} \cos \theta)(m_l d + m_u l_0)\dot{\theta} - c_h r_h(\dot{p}_h - r_h(\dot{\theta} - \dot{\phi})) + \\ m_l g d \sin \theta + m_u g l_0 \sin \theta - k_h r_h(p_h - p_{h0} - r_h(\theta - \phi)) = T_{fr} \text{sign}(-\dot{\theta})\end{aligned}\tag{A.8}$$

The final state equation are given by the following equations:

$$\ddot{\theta} = \frac{-A_3 a_2 + A_4 a_1}{-a_2^2 + a_1 a_3}\tag{A.9}$$

$$\ddot{\phi} = \frac{A_3 a_3 - A_4 a_2}{-a_2^2 + a_1 a_3} \quad (\text{A.10})$$

$$\ddot{x} = -\frac{m_t}{m} d_e \sin(\phi + \alpha) \ddot{\phi} + \left( \frac{m_l d + m_u l_0}{m} \right) \cos \theta \ddot{\theta} + A_1 \quad (\text{A.11})$$

$$\ddot{z} = \frac{m_t}{m} d_e \cos(\phi + \alpha) \ddot{\phi} + \left( \frac{m_l d + m_u l_0}{m} \right) \sin \theta \ddot{\theta} + A_2 \quad (\text{A.12})$$

where  $m$  is total mass and the auxiliary parameters are:

$$\begin{aligned} A_1 &= \frac{F_{fr,x}}{m} \text{sign}(-\dot{x}) - \left\{ \frac{m_t}{m} d_e \dot{\phi}^2 \cos(\phi + \alpha) + \left( \frac{m_l d + m_u l_0}{m} \right) \dot{\theta}^2 \sin \theta \right\} \\ A_2 &= \frac{F_{fr,z}}{m} \text{sign}(-\dot{z}) - \left\{ \frac{m_t}{m} d_e \dot{\phi}^2 \cos(\phi + \alpha) + \left( \frac{m_l d + m_u l_0}{m} \right) \dot{\theta}^2 \cos \theta + g \right\} \\ A_3 &= T_{fr} \text{sign}(-\dot{\phi}) - m_t d_e (A_1 \sin(\phi + \alpha) - A_2 \cos(\phi + \alpha)) - m_t g d_e \cos(\phi + \alpha) - \\ &\quad k_h r_h (p_h - p_{h0} - r_h(\theta - \phi)) - c_h r_h (\dot{p}_h - r_h(\dot{\theta} - \dot{\phi})) \\ A_4 &= T_{fr} \text{sign}(-\dot{\theta}) - (m_l d + m_u l_0) (A_1 \cos \theta + A_2 \sin \theta) - m_l g d \sin \theta - m_u g l \sin \theta + \\ &\quad k_h r_h (p_h - p_{h0} - r_h(\theta - \phi)) + c_h r_h (\dot{p}_h - r_h(\dot{\theta} - \dot{\phi})) \\ a_1 &= m_t d_e^2 + J_t - 2 \frac{m_t^2}{m} d_e^2 \sin(\phi + \alpha) \cos(\phi + \alpha) \\ a_2 &= -\frac{m_t}{m} (m_l d + m_u l_0) d_e \sin(\phi + \alpha - \theta) \\ a_3 &= m_l d^2 m_u l_0^2 + J_l - \frac{(m_l d + m_u l_0)^2}{m} \end{aligned} \quad (\text{A.13})$$

## Stance Phase

Considering Figure (A.1) the unsprung mass is stationary and the velocities of two other masses are as follows:

$$\mathbf{v}_t = (\dot{l} + d_e \dot{\phi} \cos(\phi + \alpha - \theta)) \hat{e}_l + (l \dot{\theta} + d_e \dot{\phi} \sin(\phi + \alpha - \theta)) \hat{e}_n \quad (\text{A.14})$$

$$\mathbf{v}_l = -\dot{l} \hat{e}_l - (l - d) \dot{\theta} \hat{e}_n \quad (\text{A.15})$$

Therefore kinetic and potential energies are obtained by:

$$\begin{aligned} \mathbf{T} &= \frac{1}{2} m_t \left\{ \dot{l}^2 + (l \dot{\theta})^2 + (d_e \dot{\phi})^2 - 2 d_e \dot{\phi} (\dot{l} \cos(\phi + \alpha - \theta) + l \dot{\theta} \sin(\phi + \alpha - \theta)) \right\} + \\ &\quad \frac{1}{2} J_t \dot{\phi}^2 + \frac{1}{2} J_l \dot{\theta}^2 + \frac{1}{2} m_l \left\{ \dot{l}^2 + (l - d)^2 \dot{\theta}^2 \right\} \end{aligned} \quad (\text{A.16})$$

$$\mathbf{V} = m_t g(l \cos \theta + d_e \sin(\phi + \alpha)) + m_l g(l - d) \cos \theta + \frac{1}{2} k_l (s_0 + p + d - l)^2 + \frac{1}{2} k_h (p_h - p_0 - r_h(\theta - \phi))^2 \quad (\text{A.17})$$

Considering  $\mathbf{L} = \mathbf{T} - \mathbf{V}$  and Equation (A.1) we end up with the following state equations:

$$(m_t + m_l) \ddot{l} - m_t d_e \{ \ddot{\phi} \cos(\phi + \alpha - \theta) + \dot{\phi} \sin(\phi + \alpha - \theta) (\dot{\theta} - \dot{\phi}) \} - m_l (l - d) \dot{\theta}^2 - m_t (l \dot{\theta}^2 - d_e \dot{\phi} \dot{\theta} \sin(\phi + \alpha - \theta)) + m_t g \cos \theta + m_l g \cos \theta - c_l (\dot{p} - \dot{l}) - k_l (s_0 + p + d - l) = F_{fr} \text{sign}(-\dot{l}) \quad (\text{A.18})$$

$$(m_t l^2 + m_l (l - d)^2 + J_l) \ddot{\theta} + m_t d_e l \{ \ddot{\phi} \sin(\phi + \alpha - \theta) + \dot{\phi} \cos(\phi + \alpha - \theta) (\dot{\theta} - \dot{\phi}) \} - 2 \dot{\theta} \dot{l} (m_t l + m_l (l - d)) - m_t d_e \sin(\phi + \alpha - \theta) \dot{l} \dot{\phi} + m_t d_e \dot{\phi} \dot{l} \sin(\phi + \alpha - \theta) - m_t d_e \dot{\phi} \dot{\theta} \cos(\phi + \alpha - \theta) - m_t g l \sin \theta - m_l g (l - d) \sin \theta - k_h r_h (p_h - p_0 - r_h(\theta - \phi)) - c_h r_h (\dot{p}_h - r_h(\dot{\theta} - \dot{\phi})) = T_{fr} \text{sign}(-\dot{\theta}) \quad (\text{A.19})$$

$$(J_t + m_t d_e^2) \ddot{\phi} - m_t \ddot{l} d_e \cos(\phi + \alpha - \theta) - m_t d_e l \sin(\phi + \alpha - \theta) \ddot{\theta} - 2 m_t d_e \sin(\phi + \alpha - \theta) \dot{\theta} \dot{l} + m_t d_e l \cos(\phi + \alpha - \theta) \dot{\theta}^2 + m_t d_e \sin(\phi + \alpha - \theta) \dot{l} \dot{\phi} - m_t d_e \cos(\phi + \alpha - \theta) \dot{\phi} \dot{\theta} - m_t (d_e \dot{\phi} (\dot{l} \sin(\phi + \alpha - \theta) + l \dot{\theta} \cos(\phi + \alpha - \theta)) + m_t g d_e \cos(\phi + \alpha) + k_h r_h (p_h - p_0 - r_h(\theta - \phi)) + c_h r_h (\dot{p}_h - r_h(\dot{\theta} - \dot{\phi}))) = T_{fr} \text{sign}(-\dot{\phi}) \quad (\text{A.20})$$

The final simplified state equations are:

$$\ddot{l} = A_1 \ddot{\phi} + B_1 \quad (\text{A.21})$$

$$\ddot{\theta} = A_2 \ddot{\phi} + B_2 \quad (\text{A.22})$$

$$\ddot{\phi} = \frac{B_3 - a_2 B_1 - a_3 B_2}{a_1 + a_2 A_1 + a_3 A_2} \quad (\text{A.23})$$

Where the auxiliary parameters are defined as:

$$A_1 = \frac{-1}{m_t + m_l} m_t d_e \cos(\phi + \alpha - \theta) \quad (\text{A.24})$$

$$A_2 = \frac{-1}{m_t l^2 + m_l (l - d)^2 + J_l} m_t d_e l \sin(\phi + \alpha - \theta)$$

$$\begin{aligned}
B_1 &= \frac{1}{m_t+m_l}F_{fr}\text{sign}(-\dot{l}) + m_t d_e \dot{\phi} \sin(\phi + \alpha - \theta)(\dot{\theta} - \dot{\phi}) + \\
&\quad m_t(l\dot{\theta}^2 - d_e \dot{\phi} \dot{\theta} \sin(\phi + \alpha - \theta)) + m_l(l-d)\dot{\theta}^2 + m_t g \cos \theta - \\
&\quad m_l g \cos \theta + k_l(s_0 + p + d - l) + c_l(\dot{p} - \dot{l}) \\
B_2 &= \frac{1}{m_t l^2 + m_l(l-d)^2 + J_l} - m_t d_e l \dot{\phi} \cos(\phi + \alpha - \theta)(\dot{\theta} - \dot{\phi}) + \\
&\quad 2\dot{\theta} \dot{l}(m_t l + m_l(l-d)) + m_t d_e \sin(\phi + \alpha - \theta) \dot{l} \dot{\phi} - m_t d_e \dot{\phi} \dot{l} \sin(\phi + \alpha - \theta) + \\
&\quad m_t d_e \dot{\phi} \dot{\theta} \cos(\phi + \alpha - \theta) + m_t g l \sin \theta + m_l g(l-d) \sin \theta + \\
&\quad k_h r_h(p_h - p_0 - r_h(\theta - \phi)) + c_h r_h(\dot{p}_h - r_h(\dot{\theta} - \dot{\phi})) + T_{fr} \text{sign}(-\dot{\theta}) \\
B_3 &= 2m_t d_e \sin(\phi + \alpha - \theta) \dot{l} \dot{\theta} - m_t d_e l \cos(\phi + \alpha - \theta) \dot{\theta}^2 + m_t d_e \sin(\phi + \alpha - \theta) \dot{l} \dot{\phi} + \\
&\quad m_t d_e \cos(\phi + \alpha - \theta) \dot{\phi} \dot{\theta} + m_t(d_e \dot{\phi}(\dot{l} \sin(\phi + \alpha - \theta) + \\
&\quad l \dot{\theta} \cos(\phi + \alpha - \theta) + \dot{l} \dot{\theta} \cos(\phi + \alpha - \theta)) - m_t g d_e \cos(\phi + \alpha) - \\
&\quad k_h r_h(p_h - p_0 - r_h(\theta - \phi)) - c_h r_h(\dot{p}_h - r_h(\dot{\theta} - \dot{\phi})) + T_{fr} \text{sign}(-\dot{\phi}) \\
a_1 &= (J_t + m_t d_e^2) \\
a_2 &= -m_t d_e \cos(\phi + \alpha - \theta) \\
a_3 &= -m_t d_e l \sin(\phi + \alpha - \theta)
\end{aligned} \tag{A.25}$$

Notice that the parameters with subscript  $h$  related to the hip joint, i.e.  $p_h$  represents the hip actuator position. In this derivation we assume to have linear actuator and spring connected to the hip through an arm with radius  $r_h$ ; therefore, the hip actuation phenomena get similar to the leg actuation. The discussion of energy losses due to unsprung mass and mechanical stops elaborated in Section (2.1) is also valid here.

By comparison the results obtained here to the results obtained in Section (2.2) we can conclude that by just slightly changing the assumption the dynamics of the system became excessively complex. We will show how acceptable the simplified model is by using experimental results. This could be generalized to the system with more complexities and we could expect that the simple model-based controller, could accomplish the goals for more complex systems.