ISTANBUL TECHNICAL UNIVERSITY ★ GRADUATE SCHOOL OF SCIENCE ENGINEERING AND TECHNOLOGY

SIMULATION AND DESIGN OF ONE LEGGED THREE DIMENSIONAL HOPPING ROBOT

M.Sc. THESIS

Ercüment BAŞ

Department of Mechatronics Engineering

Mechatronics Engineering Programme

JUNE 2012

İSTANBUL TEKNİK ÜNİVERSİTESİ ★ FEN BİLİMLERİ ENSTİTÜSÜ

ÜÇ BOYUTTA HAREKET EDEN TEK BACAKLI ROBOTUN TASARIMI VE SİMÜLASYONU

YÜKSEK LİSANS TEZİ

Ercüment BAŞ
518081008

Mekatronik Mühendisliği Anabilim Dalı

Mekatronik Mühendisliği Programı

Tez Danışmanı: Yrd. Doç. Dr. S. Murat YEŞİLOĞLU

HAZİRAN 2012

ISTANBUL TECHNICAL UNIVERSITY ★ GRADUATE SCHOOL OF SCIENCE ENGINEERING AND TECHNOLOGY

SIMULATION AND DESIGN OF ONE LEGGED THREE DIMENSIONAL HOPPING ROBOT

M.Sc. THESIS

Ercüment BAŞ
(518081008)

Department of Mechatronics Engineering

Mechatronics Engineering Programme

Thesis Advisor: Asst. Prof. S. Murat YESİLOĞLU

JUNE 2012

Ercüment Baş, a **M.Sc.** student of ITU **Graduate School of Science Engineering and Technology** student ID **518081008**, successfully defended the **thesis** entitled "**SIMULATION AND DESIGN OF ONE LEGGED THREE DIMENSIONAL HOPPING ROBOT**", which he prepared after fulfilling the requirements specified in the associated legislations, before the jury whose signatures are below.

Thesis Advisor :	**Assist. Prof. Dr. Murat YESILOGLU** İstanbul Technical University	....
Jury Members :	**Prof. Dr. Hakan TEMELTAS** İstanbul Technical University	....
	Assoc. Prof. Dr. E. Faruk KECECI İstanbul Technical University	....

Date of Submission : 03 May 2012
Date of Defense :05 June 2012

PREFACE

It is a big reason of happiness for me that I had a chance to study on this subject because I see and believe that robots are the key to the future.

I would like to thank my family and of course Assistant Professor Sıddık Murat YEŞİLOĞLU for their endless help and support to make this happen.

JUNE 2012 Ercüment BAŞ

INDEX

<u>Page</u>

ABBREVIATIONS

SLIP **:** Spring Loaded Inverted Pendulum
LQR **:** Linear Quadratic Regulator
EPM : Energy Pumping Mechanism
DOF : Degree Of Freedom

FIGURE LIST

SIMULATION AND DESIGN OF ONE LEGGED THREE DIMENSIONAL HOPPING ROBOT

SUMMARY

As a result of improving in technology, we can see many machines around us. And for specifically, robots started new era in this field. Now we use robots places where human is not able to work or can be dangerous to work there.
But we should not think that all robots work all places. For example, we cannot use wheeled robots in every area. Because of this, there are many type of robots which used in many places. So we see the advantage of legged locomotion after knowing this fact.

Legged locomotion has been widely used because of its easy mobility in rough terrain and it is the least constrained walking. The planar one-legged robots have attracted many researchers due to the simplicity of its mechanical design. It was thought that the analysis and experiments of the one-legged robots would enlighten the designing of biped, quadruped, and multi-legged robots.

The field of dynamically stable legged locomotion has made great strides in 1980s, led primarily by Marc Raibert. The study of one-legged hopping problem has fascinated many scholars and researchers since then.

The aim of this thesis , after looking to past researches, is to create a prototype of one legged hopping robot which can be able to walk in three dimensions. It used the energy stored in the spring when the time of touchdown and stance . It moves in three dimensions by its electrical motors placed in 120 degrees around the base.

Before this , we need model of the system. Because dynamics of the system is different in stance and flight, it is hard to control it properly. So we should make sum assumptions.

Another important job of the system is to walk. After jumping, system should walk in dimension we want to move. It can be done by controlling the length and angle of the electrical motors.

In this study, CATIA software is used to create the prototype of the system, MATLAB and SIMULINK is used for simulations.

ÜÇ BOYUTTA HAREKET EDEN TEK BACAKLI ROBOTUN SİMULASYONU VE TASARIMI

ÖZET

Hepimizin bildiği gibi teknoloji günümüzde oldukça gelişmiştir ve gelişmeye devam etmektedir. Bu gelişmede en büyük pay hiç şüphesiz robotlarındır. Robotlar insan hayatını kolaylaştırmanın yanı sıra insanın çalışmasının mümkün olmadığı veya çok zor olduğu şartlarda da çalışabildiği için oldukça popülerdir. Bu da bizi pek çok türde robot olduğunu gösterir.

Bu tezde tek bacaklı zıplayan robot ele alınmıştır. Bu konuyu seçmemin sebebi yürüyen robotların temelini teşkil etmesidir. Yürüyen robotların tasarımında temel teşkil ettiği için zıplayan robotlar iyi analiz edildiği takdirde değişik türde yürüyen robot tasarımı daha kolay yapılabilir.

Tekerlekli vb. diğer türde robotlardan farklı olarak ayaklı robotlar hemen her türde hareket edebilme ve minimum enerji ilkesiyle çalışma gibi avantajlara sahip olduğu için sıklıkla tercih edilmiştir.

Tek bacaklı zıplayan robotlar konusunda ilk çalışmalar 1980 yıllarında Raibert tarafindan yapılmıştır. Bu tarihlerden itibaren pek çok araştırmacı bu konuda çalışmıştır.

İlk olarak yatay ve düşey hareketin modellenmesi yapılmıştır. Hız denklemeleri yatay ve düşey yer değiştirme denklemlerinin iki kere türev alınmasıyla elde edilmiştir. Daha sonra bu hareket sırasında meydana gelen kuvvetler tespit edilmeye çalışılmıştır. Bu kuvvetler robotun yere temas ettiği noktada ve eklem noktasında yatay ve düşey kuvvetler olarak tespit edilmiştir. Ardından robotun hareket çevrimi ele alınmıştır. Havadaki ve yerdeki anlar olmak üzere iki kısımda incelenebileceği görülmüştür. Sistemin hareketi için önemli olduğundan bir turda geçirilen zaman öğrenilmeye çalışılmıştır.

Bütün bu bilgilerden sonra zıplama hareketinin matematik modellenmesine çalışılmıştır. Bu modelleme Lagrange dinamiği yöntemi ile yapılmıştır. Sistemin ilk temas, temas, kopma ve havada olmak üzere dört hali için ayrı ayrı denklemler çıkarılmıştır. Bunlardan hareketin temelin oluşturan yerde ve havada olduğu durumların durum-uzay denklemleri çıkarılmıştır.

Daha sonra sistemin sadece düşey hareketi incelenmiş ve bu hareketin dinamik denklemleri elde edilmiştir. Bu sistemin kontrol edilmesi gerektiğinden bu işin nasıl yapılması gerektiği önem kazanmıştır. Bunun için Lineer Kuadratik Regülatör (LQR) yönteminin uygun olabileceği görülmüştür. Bu yöntem minimum enerji prensibine göre çalıştığından sistemin kontrolü için yeterlidir. Bu yöntemin nasıl elde edildiği ve denklemlerin nasıl çıkarıldığı araştırılmıştır. Önceden elde ettiğimiz düşey harekete ait dinamik denklemler bu yönteme uygulanmıştır. Durum-uzay formatına getirilen sistemin kontrol edilebildiği ve gözlemlenebildiği tespit edilmiştir.

Zıplama hareketi incelendikten sonra sistemden beklediğimiz diğer hareket olan yürüme hareketine geçilmiştir. Sistemin havada ve yerdeki durumlarında nasıl hareket etmesi gerektiği incelenmiştir. Yürüme anında sistemin davranışının nasıl olduğu ve yürümenin kararlı bir şekilde olması için simetri kavramı ortaya çıkmıştır. Sistemin yürümesinin hızlı veya yavaş bir şekilde nasıl olabileceği nört nokta ile açıklanmıştır. Bütün bu analizlerden görülmüştür ki sistemin yürümesi pek çok parametrenin uygun bir şekilde bir araya gelmesi sonucu olmaktadır.

Şimdiye kadar elde edilen bilgiler ışığında üç boyutta hareket eden robot tasarımı yapılmıştır. Tasarım yapılırken CATIA programından faydalanılmıştır.

Sistemin çalışma prensibi temel olarak yay yüklü ters sarkaç (SLIP) tasarımı üzerine geliştirilmişir. Çalışma prensibi belli bir yükseklikte bırakılan sistemin içinde bulunan yayı kullanarak sıkışması ve sıkışma sonucu depoladığı enerjiyi yeniden aynı yüksekliğe çıkmak için kullanması şeklinde açıklanabilir. Sistemin yürümesi ise içinde bulunan elektrik motorlarının uzunluk ve açılarının kontrol edilmesi sonucu mümkündür. Elektrik motorlarının üçü birden aynı anda hareket ettirilirse sisteme yerdeyken yay sıkışması sonucu oluşan kuvvet dışında ekstra kuvvet sağlar.

Sistemin tasarımında belli parametrelere dikkat edilmiştir. Bunun başında sistemin toplam ağırlığı gelir.Sistem ne kadar hafif olursa yerda kalma süresi o kadar kısa olur ve havada daha fazla kalır. Bu da yürümenin daha kolay kontrol edilmesi anlamına gelir.

Sistemi oluşturan parçaları daha detaylı incelersek, temel parça olarak bacak, yay, mesnet, elektrik motorları ve tabandır.

Sistemin toplam serbestlik derecesine baktığımız zaman motorların alt ve üst bağlantı noktalarındaki üniversal mesnetten dolayı 6, toplamda 18; ayağın sıkışması ve üç yöne hareketinden oluşan 4 olmak üzere toplam 22 serbestlik derecesi mevcuttur. Fakat sistem motorların ve ayağın birbirine bağlı olmasından dolayı paralel yapıdadır ve son olarak sistemin serbestlik derecesi 4 olarak görülür.

Sistemi oluşturan temel parçaları incelediğimiz zaman ayak yapısının en önemli parçası olduğunu görürüz. En uçta yerle teması sağlayan birinci parçadır. Bu parçanın enerji kayıplarını en aza indirmek için ince ve boru şeklinde olması düşünülmüştür. Birinci ve ikinci parça arasındaki yay temas anında sıkışmayı sağlar. Üçüncü parça tabana bağlıdır ve dönme hareketini gerçekleştirir. Ayak yapısı tasarlanırken uzun ve ağır olmamasına dikkate dilmiştir. Çünkü ağır sistem havada kalma süresini kısaltır ve uzun olursa sistemin kendi ve diğer ekipmanların ağırlığından dolayı eğilme problemi görülebilir. Bu nedenlerden ötürü ayak yapısının imal edilmesi aşamasında alüminyumun daha uygun olabileceği öngörülmüştür.

Sistemdeki diğer önemli parça elektrik motorlarıdır. Sistemin istenilen yönde hareketini sağlar ve sıkışma sırasında sisteme eksta kuvvet verir. Dairesel hereketi doğrusal harekete çevirir. Sistemin temas anı kısa olduğu için seçilen motorlar hızlı olmalıdır. Bunun yanında motor seçiminde dikkate edilmesi gereken diğer parametreler uzun çalışma aralığı, hafiflik ve fiyattır. Bu ihtiyaçları karşılayan SKF-CARE33H modeli elektrik motorun uygun olacağı düşünülmektedir.

Motor ve ayakların üç boyutta hareketi mesnetler tarafından sağlanır. Üniversal olarak tasarlanan mesnetler ayağa ve tabana bağlıdır. Tabana bağlı mesnet ayağın dönmesine yardımcı olurken ayağa bağlı mesnetler motorun dönme hareketini öteleme hareketine çevirmesine yardımcı olur. Mesnetlerin tasarımı CATIA programı kullanılarak yapılmıştır.

Sisteme ait taban olarak adlandırdığımız parça ayak, motorlar ve diğer ekipmanları (batarya, sensör vb.) taşıyan kısmıdr. Mümkün olduğu kadar hafif olması gerektiği için et kalınlığı ince dairesel halka şeklinde tasarlanmıştır. Tasarım yapılırken zıplama ve yürüme esnasında sistemin devrilebileceği öngörüldüğü için çemberin çapı büyük düşünülmemiştir.

Tasarımdan sonra sistemin simülasyonları yapılmıştır. Simülasyonlar yapılırken MATLAB ve SIMULINK programları kullanılmıştır.

Öncelikle sistemin sadece kütle ve yaydan ibaret olduğu temel durum ele alınmıştır. Bu durumda sistemde herhangi bir kayıp yoktur ve sistem aynı yüksekliğe tekrar yükselir. İkinci durumda sisteme damper eklenir. Bu durumda sistemin enerjisi sönümleneceği için hareket belli bir süre sonra durur. Sistemin normal hareketi bu olduğu için aynı yüksekliğe ulaşmak içi dışarıdan bir kuvvet eklenmesi gerketiği görülür. Bu kuvvetin nasıl ve ne zaman ekleneceği araştırılmıştır. Sisteme gereken bu kuvvetin yere temas ettiği anda elektrik motorlarına rampa fonksiyonu şeklinde girişinin uygun olacağı düşünülmüş ve simülasyon sonuçları ile görülmüştür.

Sonuç olarak bu çalışma neticesinde elde edilen bilgiler ışığında tek ayaklı robotun zıplama ve daha sonrasında yürümesinin pek çok parametreye bağlı olduğu görülmüştür. Bu parametrelerin uygun şekilde seçilmesi neticesinde daha kararlı bir hareket elde edilmesi mümkün olacaktır. Bu hareketin sağlanmasından sonra üç boyutta hareketin kontrollü bir şekilde yapılması daha kolay olacaktır.

1. INTRODUCTION

In this thesis a special type of robot named one-legged hopping robot is studied. The most notable results obtained on this subject was achieved in the 1980's by a group of researchers led by Prof. Dr. Marc Raibert at MIT. In this chapter the motivation and the literature review regarding to one legged hopping machines are presented.

1.1 Background

There are several reasons to conduct this type of research.

The first reason is to study the mobility on rough terrain. When considering displacements on rough terrains, perturbations resulting from the system's interaction with the ground are intermittent and discretized in space in the case of legged vehicles, while their effects on the vehicle are permanent in the case of wheeled locomotion. It is commonly acknowledged that legged locomotion is superior to wheeled locomotion when the terrain is soft. It is also considered that legged vehicle can easily overcome obstacles by utilizing the flight period.

The second reason to study legged machines is to understand biological legged locomotion. The principle of control which is used in human and animal locomotion is still not understood. Humans and animal enjoy high mobility and efficiency of locomotion due to their naturally designed legged system. It is of great interest to the researchers to build mechanical machines which replicates human and animal motions.

The planar one-legged robots have attracted many researchers due to the simplicity of its mechanical design. It was thought that the analysis and experiments of the one-legged robots would enlighten the designing of biped, quadruped, and multi-legged robots.

The first one-legged robot was built by Raibert. It had a pneumatic cylinder installed in its leg and hence moved as a springy inverted pendulum while on the ground. Raibert decomposed the control problem into body attitude control, hopping height control and horizontal velocity control and showed that separate designs of the three controllers may be robust enough to allow decoupled operation. After Raibert's works, one legged running robots filled with a leg spring have been widely studied both experimentally and theoretically. Most of the one-legged robots proposed during the past two decades were similar to Raibert's design since they had some parts functioning like a spring to restore energy and provide force for take-off. These robots were also called one legged hoppers since their locomotion were series of cyclic hops.

1.2 Literature Review

1.2.1 Types of legged robots

Legged robots fall into two classes, statically stable robots and dynamically stable robots.

The static stability is a simple concept. There is a contact polygon formed by connecting all the neighboring footholds. It is called statically stable when the projection of the center of mass of the body lies in the convex hull of the contact polygon.

A statically stable robot can stand still without falling over. Static stability is a useful feature and it can be achieved by requiring enough legs on the robot to provide sufficient static points of support.

On the other hand, dynamic stability allows a robot to be stable while moving. For example, one-legged hopping robots are dynamically stable and they can hop in place or to various places without falling over. It is enough to say a robot is dynamically stable when it maintains balance in the overall locomotion cycle even if the robot is not balanced statically at any time.

A statically stable system could be dynamically unstable and a statically unstable system could be dynamically stable.

Statically stable machines are only simple successful solutions for low-speed locomotion, where dynamic forces are small compared to static ones. As a result, statically stable legged machines suffer not only from a large number of legs but also from a low cruising speed. On the contrary, dynamically stable robots enjoy less design complexity, achieve higher speeds and can be more energy efficient. The main disadvantage of dynamic balance is the lack of a general control methodology.

1.2.2 Research on legged machines

The scientific study of legged locomotion began over a century ago when Muybridge tried to find out whether or not a trotting horse left the ground with all four feet at the same time. After that Muybridge went on to document the walking and running behavior over 40 animals including humans. His photographic data are still of considerable value and survive as a landmark in locomotion research.

The studies of walking machines also have its origin in Muybridge's time. An early walking model appeared in about 1870 .It used a linkage originally designed by the famous Russian mathematician Chebyshev some years earlier to move the body along a straight horizontal path while the feet moved up and down to exchange support.

During the 90 years that followed, people viewed that building walking machines as the task of designing kinematic linkages that would generate suitable stepping motion. Many designs were proposed but the performance of such machines was limited by their fixed patterns of motion since they could not adjust to the terrain's variations. By 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 harness human. Ralph Mosher used this approach in building four legged walking truck at General Electric in mid 1960s Despite its dependence on a well-trained human for control, this walking machine was a landmark in legged technology.

Computer control became alternative to human control of legged vehicles in 1970s. McGhee's group was the first to use this approach successfully They built an insect-like hexapod that could walk with a number of standard gaits and negotiate simple obstacles. The computer's major task is to solve kinematic equations in order to coordinate the 18 electric motors driving the legs. Gurfinkel and his group build a machine with quite similar performances to McGhee's at about the same time. It used a hybrid computer for control Hirose realized the linkage design and computer control are not exclusive and his experience with clever and unusual mechanisms led to a simplified control of locomotion and improved their efficiency McGhee, Gurfinkel, and Hirose's walking machine groups represent a class called Static Crawlers. Each differs in the details of construction and computing technology used for control. Several other machines that fall into this class have been studied in the intervening years.

Another class of legged systems are dynamic machines that balance actively. The legged machines fall into this class operate in a regime where the velocity and kinetic

energies are important determinants of behavior. The exchange of energy among its various forms is also important in the dynamics of legged locomotion. Shannon was probably the first to build a machine that balanced an inverted pendulum atop of a small powered truck. The truck drove back and forth in response to the tipping

movements of the pendulum. This study was forwarded by his students to demonstrate controllers for two pendulums at once, and finally the case that one pendulum were mounted on top of the other. Later, they extended these techniques to provide balance for a flexible inverted pendulum. Miura and Shimoyama built the first walking machine that really balanced actively. The control of their biped relied on an inverted pendulum model. Each time a foot was placed on the floor, its position was chosen according to the tipping behavior expected from an inverted pendulum. Matsuoka was the first to build a machine that was able to run, where running is defined by the presence of intervals of ballistic flight when all feet are off the ground. Matsuoka's goal was to model repetitive hopping in humans. He formulated a model with a body and one massless leg and also simplified the problem by assuming that the duration of the support phase was short compared with the flight phase.

The field of dynamically stable legged locomotion has made great strides in 1980s, led primarily by Marc Raibert. He build a variety of running robots, starting with a planner one-legged machine, followed by a 3D one-legged, a two-legged planar robot and a four-legged quadruped. Among his works, Raibert introduced a scheme for exciting the leg spring from rest in order to regulate the energy in the spring-mass.

Raibert has also proposed a tabular control algorithm which uses a large table of pre-computed data and calculates the control signal by interpolation. This algorithm was applied experimentally to 2-D and 3-D physical prototypes.

After Raibert's work, many researchers were attracted to study and conduct research on one-legged running robots either experimentally or theoretically.

Based on their established approach to analyze the intermittent dynamics of a juggling robot, Koditschek and Buehler derived an analytical discrete map for a simplified vertical hopper, modeled after Raibert's robots. Lapshin proved the asymptotic global stability of the vertical motion of a one-legged robot with linear leg spring. Vakakis and Burdick observed chaotic behavior when they changed the parameters of a model similar to that used by Koditschek and Buehler. Later M'Closky and Burdick used a planar hopper model with a body placed on top of a massless and an inertia-less leg similar to Raibert's, They showed by perturbation and numerical integration methods that even the planner hopper can show chaotic behavior in a certain set of parameters. There exist very few related results that use the continuous-time framework for stability analysis of hopping robots due to their intermittent dynamics. Li and He used a perturbation approach to study the existence and stability of limit cycles for a vertical hopper. Sznair and Damborg (1989) used an adaptive control algorithm for both vertical and horizontal motion of a two dimensional hopping robot, deriving a rather simple analytical solution for vertical motion control. Prosser and Kam suggested an approach which involved a near inverse of the system dynamics (based on off-line synthesis and inverse model) on an electrically actuated hopper. The preliminary control strategy in their electrically actuated hopping robot was obtained by a least-square fit of data to a multivariable polynomial. This method was later enhanced by applying on-line estimation of Controllers parameters (Lebaudy, 1993) Rad et al (1993) employed an open loop control approach on an electrically actuated hopper. A high gain PD controller was used to return the actuator.

2. THE MODEL FOR VERTICAL AND HORIZONTAL MOTION

In this chapter, a one-legged hopping machine is modeled by a springy leg with nonzero mass, a simple body, and an actuated hinge-type hip. We can discuss both vertical and horizontal motion in this model. The most important characteristic of the body is that it has a mass that must be balanced at the top of the leg because of the structure; it applies torque to the leg. Legs change length and orientation with respect to the body. Figure 2.1 shows the model used for analysis and simulation.

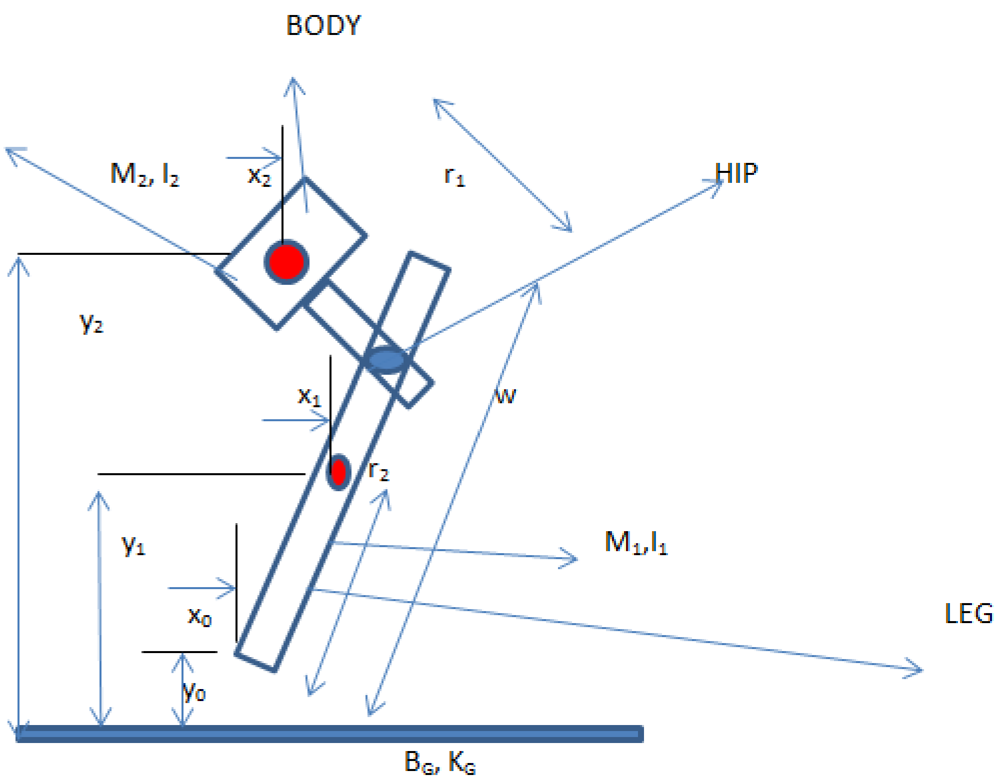

Figure 2.1: Planar one legged model.

List of parameters and notations is presented as follows:

Symbol	Parameter
M_1	Leg mass

M_2	Body mass
r_1	Leg center of mass
r_2	Body center of mass
I_1	Leg moment of inertia
I_2	Body moment of inertia
k_0	Leg spring rest length
K_L	Leg spring stiffness
K_{L2}	Mechanical stop stiffness
B_{L2}	Mechanical stop damping
K_G	Ground stiffness
B_G	Ground damping
X	Position actuator length

The basic features of the model are a body of mass M_2, a compliant leg of mass M_1, and the ground. The overall length of the leg is influenced by a spring, a position actuator in series with the spring, and a mechanical stop.

The position actuator, length x, is arranged in series with the spring, acting between the spring and the hip. When the actuator changes length, it does work on the spring to increase or decrease the energy stored in the spring.

The spring, which has unsprung length k_0 and stiffness K_L, is modeled as though one end is rigidly connected to the foot, with the other end fastened to one side of the position actuator. The mechanical stop, modeled as a very stiff spring with damping, acts to prevent the spring from expanding beyond its rest length. The spring and mechanical stop are arranged so that only the spring generates forces when $(\omega - x) < k_0$, and only the mechanical stop generates forces when $(\omega - x) > k_0$ .The stiffness and damping of the mechanical stop, K_{L2} and B_{L2}, are chosen so that vibrations between the body and leg at lift-off decay quickly. The ground is modeled as a stiff, damped spring which has stiffness K_G and damping B_G. The damping coefficient is chosen to keep the foot from bouncing on the ground during touchdown and lift-off. This compliance in the ground represents the compliances of both the ground and the foot. We assume that the stiffness of the ground is much greater than the stiffness of the leg, $K_G >> K_L$.

2.1 Body and Leg Dynamics

The following equations are derived using geometric relationships shown in Figure 2.1. We differentiate twice with respect to time. First, x_0 is the horizontal displacement of the foot and x_1 is the horizontal displacement of the leg:

$$\mathrm{x}_1 = \mathrm{x}_0 + \mathrm{r}_1 \sin\theta_1 \tag{2.1}$$

$$\dot{\mathrm{x}}_1 = \dot{\mathrm{x}}_0 + \mathrm{r}_1 \cos\theta_1 \dot{\theta}_1 \tag{2.2}$$

$$\ddot{\mathrm{x}}_1 = \ddot{\mathrm{x}}_0 + \mathrm{r}_1 \cos\theta_1 \ddot{\theta}_1 - \mathrm{r}_1 \sin\theta_1 (\dot{\theta}_1)^2 \tag{2.3}$$

Next, y_0 is the vertical displacement of the foot and y_1 is the vertical displacement of the leg:

$$y_1 = y_0 + r_1 \cos\theta_1 \tag{2.4}$$

$$\dot{y}_1 = \dot{y}_0 + r_1(-\sin\theta_1)\dot{\theta}_1 \tag{2.5}$$

$$\ddot{y}_1 = \ddot{y}_0 - r_1(\ddot{\theta}_1 \sin\theta_1 + \cos\theta_1 (\dot{\theta}_1)^2) \tag{2.6}$$

Third, x_2 is the horizontal displacement of the body:

$$x_2 = x_0 + \omega \sin\theta_1 - r_2 \sin(-\theta_2) \tag{2.7}$$

$$\dot{x}_2 = \dot{x}_0 + \dot{\omega} \sin\theta_1 + \omega \dot{\theta}_1 \cos\theta_1 + r_2 \dot{\theta}_2 \cos\theta_2 \tag{2.8}$$

$$\begin{aligned}\ddot{x}_2 = \ddot{x}_0 + \ddot{\omega} \sin\theta_1 + \omega \ddot{\theta}_1 \cos\theta_1 - \omega(\dot{\theta}_1)^2 \sin\theta_1 \\ + r_2(\ddot{\theta}_2 \cos\theta_2 - (\theta_2)^2 \sin\theta_2) \\ + 2\dot{\omega}\dot{\theta}_1 \cos\theta_1\end{aligned} \tag{2.9}$$

Finally, y_2 is the vertical displacement of the body:

$$y_2 = y_0 + \omega\cos\theta_1 + r_2\cos(-\theta_2) \tag{2.10}$$

$$\dot{y}_2 = \dot{y}_0 + \dot{\omega}\cos\theta_1 + \omega\dot{\theta}_1(-\sin\theta_1) - r_2\dot{\theta}_2\sin\theta_2 \tag{2.11}$$

$$\begin{aligned} \ddot{y}_2 &= \ddot{y}_0 + \ddot{\omega}\cos\theta_1 + \omega(\dot{\theta}_1)^2(-\sin\theta_1) \\ &+ r_2(\ddot{\theta}_2\cos\theta_2 - (\theta_2)^2\sin\theta_2) + 2\dot{\omega}\dot{\theta}_1\cos\theta_1 \end{aligned} \tag{2.12}$$

2.2 Force and Torque Analysis of Leg and Body

There are external and internal forces act on the one-legged robot. The forces are decomposed and studied into details in this section. Figure 2.2 shows a force diagram of the external and internal forces in the model.

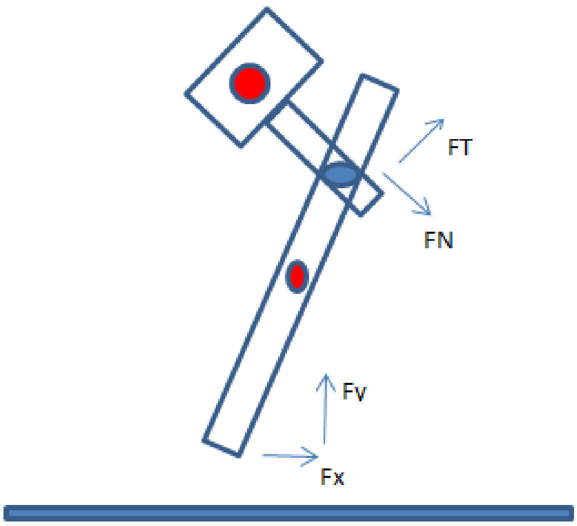

Figure 2.2: Force illustration for full model.

The first two equations are derived by summing the forces in the x and y directions for the leg. The third equation sums the moments of inertia for the leg:

$$M_1\ddot{x}_1 = F_x - F_t\sin\theta_1 - F_N\cos\theta_1 \tag{2.13}$$

$$M_1\ddot{y}_1 = F_y - F_T\cos\theta_1 - F_N\sin\theta_1 - M_1 g \tag{2.14}$$

$$I_1 \ddot{\theta}_1 = -F_x r_1 \cos\theta_1 + F_y r_1 \sin\theta_1 - F_N(\omega - r_1) - \tau(t) \qquad \textbf{(2.15)}$$

The next three equations are similarly derived for the body;

$$M_2 \ddot{y}_2 = F_T \cos\theta_1 - F_N \sin\theta_1 - M_2 g \qquad \textbf{(2.16)}$$

$$M_2 \ddot{x}_2 = F_T \sin\theta_1 + F_N \cos\theta_1 \qquad \textbf{(2.17)}$$

$$I_2 \ddot{\theta}_2 = F_T r_2 \sin(\theta_2 - \theta_1) - F_N r_2 \cos(\theta_2 - \theta_1) + \tau(t) \qquad \textbf{(2.18)}$$

Where ;

F_x, F_y horizontal and vertical forces on the foot

F_T, F_N forces acting at the hip between the leg and body. F_T acts tangent to the leg and F_N acts perpendicular to the leg.

2.3 The Hop Cycle

2.3.1 Phases and events

Hopping is a cycle which has two phases. We call flight when the foot is not touching the ground. During flight, the trajectory of the center of gravity of the system is ballistic. The other phase when the foot is touching the ground is called stance. During stance, the leg helps the system to behave like that of an inverted pendulum. there are four events in the hopping cycle :

Lift-off: The moment at which the foot loses contact with the ground

Top: The moment in flight when the body has peak altitude and vertical motion changes from upward to downward

Touchdown: The moment the foot makes contact with the ground

Bottom: The moment in stance when the body has minimum altitude and vertical motion of the body changes from downward to upward. These events can each be detected from the behavior of the state variables.

2.3.2 Total time of a hopping cycle

For the case of repetitive hopping in which periods of support alternate with periods of flight, the following values can be calculated. During the stance phase, the system can be viewed as a spring-mass oscillator with natural frequency ω_n**:**

$$\omega_n = \sqrt{\frac{K_L}{M_2}} \qquad \textbf{(2.19)}$$

During repetitive hopping, each stance interval has duration equal to one half of the period associated with ω_n:

$$T_{STANCE} = \frac{1}{2}.\frac{2\pi}{\omega_n} = \pi\sqrt{\frac{M_2}{K_L}} \qquad \textbf{(2.20)}$$

During flight the system moves along a parabolic trajectory determined by the acceleration of gravity. The period of flight is:

$$T_{FLIGHT} = \sqrt{\frac{8H}{g}} \qquad \textbf{(2.21)}$$

where

g is gravity and

H is the hopping height measured at the foot.

The period of a full hopping cycle is just the sum of $T_{s\tan ce}$ and T_{flight}:

$$T = \sqrt{\frac{8H}{g}} + \pi\sqrt{\frac{M_2}{K_L}} \qquad \textbf{(2.22)}$$

3. MODELLING AND CONTROL OF THE SYSTEM

3.1 Mathematical Model of Jumping

We need to obtain the mathematical model of the system to decide when we will give energy to the system, to make simulation of the system and to observe the behaviour of the system.

In the previous chapter, we see that dynamics of the system is different both in stance phase and in flight phase. So we should need two models for these phases. In addition, we need to add extra energy losses to these models. Figure 3.1 shows the hopper model.

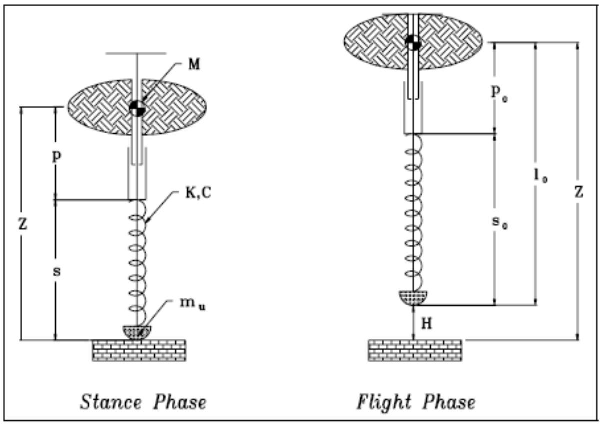

Figure 3.1: Hopper model.

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 modeled 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.

3.2 Dynamics

The motion is divided into two phases, flight and stance phases and two transient, touchdown and lift-of. In modeling , 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 τ Moreover, we suppose the motor and ball screw transmission have an overall efficiency η . For the modeling we use the Lagrange approach. The Lagrange formulation can be written as;

$$\frac{\partial}{\partial t}\left(\frac{\partial L}{\partial \dot{q}_j}\right) - \frac{\partial L}{\partial q_j} = Q_j \tag{3.1}$$

Where Lagrangian L is subtraction of potential energy from kinetic energy namely $L = T - V$, q_j is the generalized coordinate, which for our system is z and p, and the Q_j is the generalized non-conservative force, which could be derived using virtual work principle.

3.2.1 Flight

During flight phase we can write the kinetic and potential energies as follows

$$T = \frac{1}{2}J(\dot{p}/r)^2 + \frac{1}{2}(m_u + M)\dot{z}^2 + \frac{1}{2}m_s(\dot{z} - \dot{p})^2 \tag{3.2}$$

And;

$$V = Mgz + m_s g(z - p) + \frac{1}{2}k(s_0 - l_0 + p)^2 + m_u g(z - l_0) \tag{3.3}$$

where p is the downward linear motion of the actuator, and is related to the DC motor angular position θ with;

$$\theta = \frac{p}{r} \quad \textbf{(3.4)}$$

Therefore, Lagrangian L follows;

$$L = \frac{1}{2}\alpha J \dot{p}^2 + \frac{1}{2} m_t \dot{z}^2 + \frac{1}{2} m_s \dot{z}\dot{p} - m_t g z + m_s g p + \frac{1}{2} k (s_0 - l_0 + p)^2 + m_u g l_0 \quad \textbf{(3.5)}$$

$$m_t = M + m_s + m_u \quad \textbf{(3.6)}$$

$$\alpha = \frac{J}{r^2} + m_s \quad \textbf{(3.7)}$$

Before taking derivatives, we need to 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 η 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, Q_z and 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;

$$dW_z = Q_z dz = F_{fr,p} sign(-\dot{z}) dz \quad \textbf{(3.9)}$$

And the generalized force turn out to be ;

$$Q_z = -F_{fr,p} sign(\dot{z}) dz \quad \textbf{(3.10)}$$

Now suppose a virtual displacement dp and no displacement dz the virtual work will be;

$$dW_p = Q_p dp = -c \dot{s}\, ds + \tau_s d\theta \quad \textbf{(3.11)}$$

where τ_s is the torque acting on the sliding mass and is equal to $\eta\tau$ where τ is the motor torque. Moreover ;

$$s = s_0 - p \qquad \dot{s} = -\dot{p} \qquad ds = -dp \tag{3.12}$$

By using these equations we obtain;

$$Q_p = -c\dot{p} + \frac{\eta\tau}{r} \tag{3.13}$$

Using Lagrangian equation we end up with;

$$m_t \ddot{z} - m_s \ddot{p} + m_t g = -F_{fr,p} sign(\dot{z}) \tag{3.14}$$

$$\alpha \ddot{p} - m_s \ddot{z} - m_s g = \frac{\eta\tau}{r} - k(s_0 - l_0 + p) - c\dot{p} \tag{3.15}$$

Final state equation will be;

$$\begin{bmatrix} \ddot{z} \\ \ddot{p} \end{bmatrix} = \frac{1}{\gamma}\begin{bmatrix} \alpha & m_s \\ m_s & m_t \end{bmatrix}\begin{Bmatrix} -m_t g - F_{fr,p} sign(\dot{z}) \\ m_s g + \dfrac{\eta\tau}{r} - F_s \end{Bmatrix} \tag{3.16}$$

Where F_s is the spring force

$$\gamma = \alpha m_t - m_s^2 \tag{3.17}$$

$$F_s = k(s_0 - l_0 + p) + c\dot{p}$$
(3.18)

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

$$\dot{x} = Ax + B\tau + E \tag{3.19}$$

Where ;

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & -\dfrac{m_s c}{\gamma} \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & -\dfrac{m_t c}{\gamma} \end{bmatrix} \quad ; \quad B = \frac{\eta}{\gamma r}\begin{bmatrix} 0 \\ m_s \\ 0 \\ m_t \end{bmatrix} \quad ; \quad x = \begin{bmatrix} z \\ \dot{z} \\ p \\ \dot{p} \end{bmatrix}$$
(3.20)

$$E = \frac{1}{\gamma}\begin{bmatrix} 0 \\ \alpha(m_t g - F_{fr,p} sign(\dot{z})) + m_s(m_s g - k(s_0 - l_0 + p)) \\ 0 \\ m_s(m_t g - F_{fr,p} sign(\dot{z})) + m_t(m_s g - k(s_0 - l_0 + p)) \end{bmatrix}$$

(3.21)

3.2.2 Touchdown

It occurs when the leg has its free length. At his time the velocity is not changed but the kinetic energy of the unsprung mass m_t (the mass of the toe, the lower leg tube, the fraction of the spring mass) is dissipated.

$$E_k = \frac{1}{2} m_t \dot{z}_{t_d}^2 \qquad \textbf{(3.22)}$$

t_d = moment of touchdown

3.2.3 Stance

In this phase, the kinetic energy is stored as potential energy in the spring. The model is derived using the Lagrangian, based on general coordinates z and p. Total kinetic energy ;

$$T = \frac{1}{2} J(\dot{p}/r)^2 + \frac{1}{2} M \dot{z}^2 + \frac{1}{2} m_s(\dot{z} - \dot{p})^2 \qquad \textbf{(3.23)}$$

Where $p = r\theta$

The potential energy is given by ;

$$V = Mgz + m_s g(z-p) + \frac{1}{2} k(s_0 - l_0 + p)^2 \qquad \textbf{(3.24)}$$

Therefore;

$$L = \frac{1}{2}\alpha J \dot{p}^2 + \frac{1}{2}(M + m_s)\dot{z}^2 - m_s \dot{z}\dot{p} - Mgz + m_s g(z-p) - \frac{1}{2} k(s_0 - z + p)^2 \qquad \textbf{(3.25)}$$

$$\alpha = \frac{J}{r^2} + m_s \qquad \textbf{(3.26)}$$

The friction forces are modeled 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;

$$dW_z = Q_z dz = -c\dot{s}\,ds + (F_{fr,p} + F_{fr,s})sign(-\dot{z})dz \tag{3.27}$$

But in this case,

$$s = z - p, \quad \dot{s} = \dot{z} - \dot{p} \tag{3.28}$$

In general, $ds = dz - dp$, but now we do not have dp. Therefore,

$$ds = dz \tag{3.29}$$

Thus, the generalized force Q_z will derived as;

$$Q_z = -c(\dot{z} - \dot{p}) - (F_{fr,p} + F_{fr,s})sign(\dot{z}) \tag{3.30}$$

Now consider a virtual displacement dp and no dz,

$$dW_p = Q_p dp = -c\dot{s}\,ds + \tau_s d\theta \tag{3.31}$$

$$\tau_s = \eta\tau \tag{3.32}$$

In this case,

$$ds = -dp \tag{3.33}$$

Finally, generalized force will be simplified to;

$$Q_p = c(\dot{z} - \dot{p}) = \frac{\eta\tau}{r} \tag{3.34}$$

By differentiation, we end up with ;

$$\begin{gathered}(M + m_s)\ddot{z} - m_s\ddot{p} + (M + m_s)g = k(s_0 - z + p) + c(\dot{p} - \dot{z}) - (F_{fr,s} + F_{fr,p})sign(\dot{z}) \\ \alpha\ddot{p} - m_s\ddot{z} - m_s g = \frac{\eta\tau}{r} - k(s_0 - z + p) - c(\dot{p} - \dot{z})\end{gathered} \tag{3.35}$$

Final state equation will be;

$$\begin{bmatrix} \ddot{z} \\ \ddot{p} \end{bmatrix} = \frac{1}{\beta}\begin{bmatrix} \alpha & m_s \\ m_s & M+m_s \end{bmatrix}\begin{Bmatrix} -(M+m_s)g + F_s - (F_{fr,s} + F_{fr,p} sign(\dot{z})) \\ m_s g + \dfrac{\eta\tau}{r} - F_s \end{Bmatrix} \tag{3.36}$$

$$\beta = \left(M + \frac{J}{r^2}\right)m_s + \frac{MJ}{r^2} \tag{3.37}$$

$$F_s = k(s_0 - z + p) - c(\dot{p} - \dot{z}) \tag{3.38}$$

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

$$\dot{x} = Ax + B\tau + E$$

Where ;

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -\dfrac{Jc}{r\beta} & 0 & \dfrac{Jc}{r\beta} \\ 0 & 0 & 0 & 1 \\ 0 & \dfrac{Mc}{\beta} & 0 & -\dfrac{Mc}{\beta} \end{bmatrix} \quad ; B = \frac{\eta}{\beta r}\begin{bmatrix} 0 \\ m_s \\ 0 \\ M+m_s \end{bmatrix} \quad ; x = \begin{bmatrix} z \\ \dot{z} \\ p \\ \dot{p} \end{bmatrix}$$

(3.37)

$$E = \frac{1}{\gamma}\begin{bmatrix} 0 \\ \alpha(-(M+m_s)g - (F_{fr,s} + F_{fr,p})sign(\dot{z})) + m_s(m_s g - k(s_0 - z + p)) + k(s_0 - z + p)) \\ 0 \\ m_s(-(M+m_s)g + k(s_0 - z + p) - (F_{fr,s} + F_{fr,p})sign(\dot{z})) + (M+m_s)(m_s g - k(s_0 - z + p)) \end{bmatrix}$$

3.2.4 Lift-Off

It is the time when the leg is fully extended. At this time, due to conservation of linear momentum,

$$(M+m_s)\dot{z}_{l0-} - m_s\dot{p} = (M+m_u)\dot{z}_{l0+} + m_s(\dot{z}_{l0+} - \dot{p}) \tag{3.39}$$

Then,

$$\mu \dot{z}^{-} = \dot{z}^{+} \tag{3.40}$$

Where μ is defined as ;

$$\mu = \frac{m_t - m_u}{m_t} \tag{3.41}$$

4. VERTICAL HOPPING AND CONTROL STRATEGIES

Figure 4.1 shows the simplified vertical model and based on this model new vertical controllers are designed and simulations are performed. Equations can be reduced to the following for vertical motion:

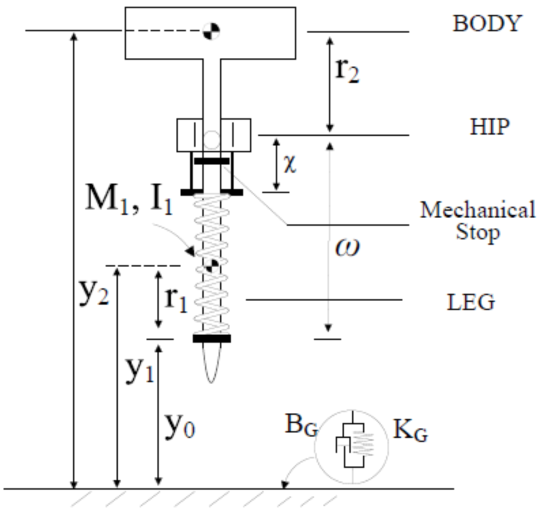

Figure 4:1. One legged hopping robot (vertical only).

$$M_1 \ddot{y}_1 = F_y - F_T - M_1 g \tag{4.1}$$

$$M_2 \ddot{y}_2 = F_T - M_2 g \tag{4.2}$$

$$y_0 = y_1 - r_1 \tag{4.3}$$

$$\omega = y_2 - y_1 + r_1 - r_2 \tag{4.4}$$

Where

$$F_T = K_L(k_0 - \omega + x) \quad if\,(k_0 - \omega + x) > 0 \tag{4.5}$$

Otherwise;

$$F_T = K_{L2}(k_0 - \omega + x) - B_{L2}\dot{\omega} \tag{4.6}$$

$$F_y = -K_{Gy_0} - B_G \dot{y}_0 \qquad if \ y_0 < 0 \qquad \textbf{(4.7)}$$

Otherwise;

$$F_y = 0 \qquad \textbf{(4.8)}$$

4.1 Linear Quadratic Regulator

The optimal Linear Quadratic Regulator (LQR) method is a powerful technique for designing controllers for complex systems that have stringent performance requirements. For most realistic applications, the LQR problem must be solved by using a Computer-Aided-Design package. A simulation is conducted by using MATLAB.

Consider the system;

$$\dot{x} = A(t)x + B(t)u, \quad x(0) = x_0 \qquad \textbf{(4.9)}$$

where $x \in R^n, u \in R^m$ with associated quadratic performance index,

$$J = \frac{1}{2}x^T(T)Sx(T) + \frac{1}{2}\int_{t_0}^{T}(x^T Q(t)x + u^T R(t)u)dt \qquad \textbf{(4.10)}$$

$[t_0;T]$ is the time interval over which we are interested in the behavior of the plant. We want to determine the control $u^*(t)$ on $[t_0;T]$ that minimizes J for the case where the final state is fixed. In this case u^* will turn out to be an open-loop control.

We assume that the final time T is fixed and known, and that no function of the final state $\psi(x(T))$ is specified. The initial plant state $x(t_0)$ is given. Weighting matrices S(T) and Q(T) are symmetric and positive semi definite, and $R(t)$ is symmetric and positive definite, for all $t \in [t_0;T]$ [6]

4.1.1 The state and costate equations

The Hamiltonian is;

$$H(t) = \frac{1}{2}(x^T Qx + u^T Ru) + \lambda^T(Ax + Bu) \qquad \textbf{(4.11)}$$

where $\lambda^T \in R^n$ is an undetermined multiplier, commonly referred to as the costate variables. The state and costate equations are

$$\dot{x} = \frac{\partial H}{\partial \lambda} = Ax + Bu \quad \textbf{(4.12)}$$

$$\dot{\lambda} = \frac{\partial H}{\partial x} = Qx + A^T \lambda \quad \textbf{(4.13)}$$

and the stationary condition is ;

$$0 = \frac{\partial H}{\partial u} = Ru + B^T \lambda \quad \textbf{(4.15)}$$

Solving **(4.15)** yields the optimal control in terms of the costate;

$$u(t) = -R^{-1} B^T \lambda(t) \quad \textbf{(4.16)}$$

Using **(4.16)** in the state equation yields the homogeneous Hamiltonian system.

$$\begin{bmatrix} \dot{x} \\ \dot{\lambda} \end{bmatrix} = \begin{bmatrix} A & -BR^{-1}B^T \\ -Q & -A^T \end{bmatrix} \begin{bmatrix} x \\ \lambda \end{bmatrix} \quad \textbf{(4.17)}$$

4.1.2 Fixed final state and open loop control

Suppose that the initial state is known to be $x(t_0)$ and that the control objective is to drive the state exactly to the given fixed reference value $r(T)$ at the final time T. Since $x(T)$ is fixed at $r(T)$, it is redundant to include a final state weighting in the cost index. Let $S(T) = 0$. Also, for our LQR design, we are only interested in minimizing the control effort, so let $Q = 0$ so that the cost function reduces to; [6]

$$J = \frac{1}{2} \int_{t_0}^{T} u^T R u dt \quad \textbf{(4.18)}$$

The state and costate equation are now,

$$\dot{x} = Ax - BR^{-1}B^T \lambda \quad \textbf{(4.19)}$$

$$\dot{\lambda} = -A^T \lambda \quad \textbf{(4.20)}$$

Setting $Q = 0$ decoupled the costate equation from the state equation, so its solution is just

$$\lambda(t) = e^{A^T(T-t)}\lambda(t) \quad \textbf{(4.21)}$$

where $\lambda(t)$ is still unknown. Using the expression in the state equation yields

$$\dot{x} = Ax - BR^{-1}B^T e^{A^T(T-t)}\lambda(t) \quad \textbf{(4.22)}$$

whose solution is

$$x(t) = e^{A(t-t_0)}x(t_0) - \int_{t0}^{t} e^{A(t-\tau)}BR^{-1}Be^{A^T(T-\tau)}\lambda(T)d\tau \quad \textbf{(4.23)}$$

4.1.3 Dynamics of the hopper

For the vertical case, the hopper can be modeled as follows:

$$M_2\ddot{y}_2 = F_T - M_2 g \quad \textbf{(4.24)}$$

$$y_0 = y_1 - r_1 \quad \textbf{(4.25)}$$

$$\omega = y_2 - y_1 + r_1 - r_2 \quad \textbf{(4.26)}$$

Where

$$F_T = K_L(k_0 - \omega + x) \quad if\,(k_0 - \omega + x) > 0 \quad \textbf{(4.27)}$$

Otherwise ;

$$F_T = K_{L2}(k_0 - \omega + x) - B_{L2}\dot{\omega} \quad \textbf{(4.28)}$$

$$F_y = -K_{Gy_0} - B_G\dot{y}_0 \qquad if\; y_0 < 0 \quad \textbf{(4.29)}$$

Otherwise ;

$$F_y = 0 \quad \textbf{(4.30)}$$

During stance period, $F_y = -K_{Gy_0} - B_G\dot{y}_0$ and $F_T = K_L(k_0 - \omega + x)$ =0. As a result, the reduced state-space model;

$$\dot{x}_1 = x_2 \quad \textbf{(4.31)}$$

$$\dot{x}_2 = \frac{F_y - F_T}{M_1} - g \tag{4.32}$$

$$\dot{x}_3 = x_4 \tag{4.33}$$

$$\dot{x}_4 = \frac{F_T}{M_2} - g \tag{4.34}$$

by defining the transformation;

$$x_1 = y_2$$

$$x_2 = \dot{y}_1$$

$$x_3 = y_2$$

$$x_4 = \dot{y}_2$$

The state space model can be represented as;

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -\frac{K_G + K_L}{M_1} & -\frac{B_G}{M_1} & \frac{K_L}{M_1} & 0 \\ 0 & 0 & 0 & 1 \\ \frac{K_L}{M_2} & 0 & -\frac{K_L}{M_2} & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{K_G r_1 - K_L(k_0 - r)}{M_1} \\ 0 \\ \frac{K_L(k_0 - r)}{M_2} - g \end{bmatrix} + \begin{bmatrix} 0 \\ -\frac{K_L}{M_1} \\ 0 \\ \frac{K_L}{M_2} \end{bmatrix} \tag{4.35}$$

where $u = x$

Since during flight phase, there is no control applied to the system and the control can only be applied during the stance phase. Thus the problem can be formulated as how to transfer the states from ;

$$\begin{bmatrix} r_1 \\ -\sqrt{2gH_{i-1}} \\ k_0 + r_2 + x_0 \\ -\sqrt{2gH_{i-1}} \end{bmatrix} \text{to} \begin{bmatrix} r_1 \\ \frac{M_1 + M_2}{M_2}\sqrt{2gH_i} \\ k_0 + r_2 + x_0 \\ \frac{M_1 + M_2}{M_2}\sqrt{2gH_i} \end{bmatrix}$$

within time $T = \pi\sqrt{\frac{M_2}{K_L}}$. H_{i-1} is the previous hopping height and H_i is the desired hopping height for this cycle.

After the coordinate transformation, the state space model is transformed into;

$$\begin{bmatrix} \dot{\bar{x}}_1 \\ \dot{\bar{x}}_2 \\ \dot{\bar{x}}_3 \\ \dot{\bar{x}}_4 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -\frac{K_G+K_L}{M_1} & -\frac{B_G}{M_1} & \frac{K_L}{M_1} & 0 \\ 0 & 0 & 0 & 1 \\ \frac{K_L}{M_2} & 0 & -\frac{K_L}{M_2} & 0 \end{bmatrix} \begin{bmatrix} \bar{x}_1 \\ \bar{x}_2 \\ \bar{x}_3 \\ \bar{x}_4 \end{bmatrix} + \begin{bmatrix} 0 \\ -\frac{K_L}{M_1} \\ 0 \\ \frac{K_L}{M_2} \end{bmatrix} x \tag{4.36}$$

Where;

$$\bar{x}_1 = x_1 + \frac{(M_1+M_2)}{K_G} - r_1 \tag{4.37}$$

$$\bar{x}_2 = x_2 \tag{4.38}$$

$$\bar{x}_3 = x_3 + \frac{(M_1+M_2)}{K_G} - r_1 - k_0 + r + \frac{M_2}{K_L} g \tag{4.39}$$

$$\bar{x}_4 = x_4 \tag{4.40}$$

Next, we examine the controllability of the linear system.

$$A : \begin{bmatrix} 0 & 1 & 0 & 0 \\ -\frac{K_G+K_L}{M_1} & -\frac{B_G}{M_1} & \frac{K_L}{M_1} & 0 \\ 0 & 0 & 0 & 1 \\ \frac{K_L}{M_2} & 0 & -\frac{K_L}{M_2} & 0 \end{bmatrix}, \; B : \begin{bmatrix} 0 \\ -\frac{K_L}{M_1} \\ 0 \\ \frac{K_L}{M_2} \end{bmatrix}$$

$$[\lambda I - A] = \begin{bmatrix} 0 & -1 & 0 & 0 & 0 \\ \lambda(\lambda + \frac{B_G}{M_1}) + \frac{K_G+K_1}{M_1} & 0 & -\frac{K_L}{M_1} & 0 & -\frac{K_L}{M_1} \\ 0 & 0 & 0 & -1 & 0 \\ \lambda^2 + \frac{B_G}{M_1}\lambda + \frac{K_G}{M_1} & 0 & \frac{M_2}{M_1}\lambda^2 & 0 & 0 \end{bmatrix} \tag{4.41}$$

for ¸ $\lambda = 0$ Full Rank

for ¸ $\lambda = 0$ Full Rank

So the system is controllable.

5. MATHEMATICAL MODEL OF WALKING

The main point of this way of modeling is when jumping in stance phase, the use of its pushing force not only for jump but also for walking with and angle and in flight phase moving towards the leg to prepare landing. The model can be derived as a SLIP model with Lagrange method.

5.1 Algortihm For Flight Phase

In flight phase, there is no disturbance affects to the system. So we can think the movement of the system same as the control of inverted pendulum. During this time, the signal of the motor should calculate the point of landing and the try to keep the angle of leg between rotation angle compared to the reference angle. [3]

5.2 The Behavior of The Leg In Stance and Symmetry

The angle when the leg is on the ground describes the orbit of center of gravity of the system. In this orbit, if we think the ahead speed is constant, total movement of the system can be described with speed and the stance period (T_s).

$$\Delta x = T_s \dot{x} \tag{5.1}$$

In this orbit, to keep the speed constant, the sum of the horizontal forces act to the system should be 0. This means that there is symmetry in the stance.

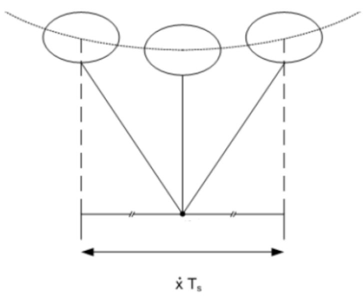

Figure 5.1: The desired orbit in stance.

Figure 5.1 shows the ideal movement of the leg. If we assume in the middle of this orbit t= 0 and x(0)=0; because of the symmetry, ϕ is the angle between leg and the rotation axis and τ is the motor torque, the parameters act the center of gravity in stance ;

$$z(-t) = z(t)$$
$$x(-t) = -x(t)$$
$$\phi(-t) = -\phi(t)$$
$$\tau(-t) = -\tau(t)$$

These expressions are true only in symmetry.

When we think about symmetry in flight, we ask ourselves the point of landing. Symmetry is available when the leg lands on the middle of orbit. because the orbit describes the movement of center of gravity, we can keep the symmetry by putting the toe to the point between the calculated orbit and the length of the leg. These data tell the controller how to calculate the reference.

The middle point of the orbit in stance can be called as neutral point. If the leg lands before that point leg slows down ; after that leg fastens. So that we can change the speed of the leg. Figure 5.2 shows the importance of neutral point in stance phase.

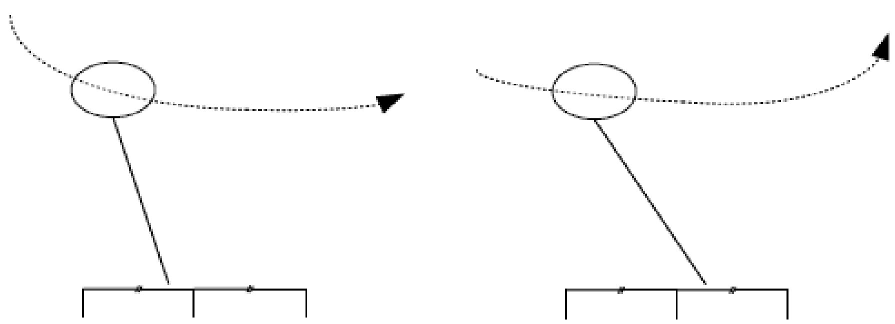

Figure 5.2: The effect of neutral point to acceleration.

In flight, the job of algorithm is to calculate the orbit in stance decide the angle of leg between rotational axis. [3]

Controller needs to do when in flight, to maintain the angle of the leg to the reference angle. xso is the length between the neutral point and the center of gravity ; $\dot{x}d$ is the desired towards speed and $xs\Delta$ is the length between the neutral point to reach keep the desired speed. The equations are ;

$$xso = \frac{T_s}{2}\dot{x},\ \ xs\Delta = k_{\dot{x}}(\dot{x} - \dot{x}_d) \qquad \textbf{(5.2)}$$

With these equations, we can derive the equation that gives the reference angle effects rotational axis,

$$\phi d = \arcsin\left(\frac{T_s \dot{x}}{2R} + \frac{k_{\dot{x}}(\dot{x} - \dot{x}_d)}{R}\right) \qquad \textbf{(5.3)}$$

Where R is the distance of the leg to the rotational axis. Because stance is only related to the mass of the leg and the spring constant, when the system is working this equation can be applied by the feedback of $\dot{x}$.[3]

5.3 Algorithm For Stance Phase

When we examine the behavior of the system in this phase, the thing that should be done is after toe is on the ground, to rotate the body around this pivot point until flight. This movement should be suitable to the symmetry rules. This can be possible by giving position reference to the motors which system's center of gravity is on the orbit.

As writing the algorithms that give these position reference, we assume that the center of the mass is on the rotational axis (y_r). The distance between the mass center and the neutral point, the distance to the neutral point to reach the desired speed are needed to define the orbit in this phase.

$$xso = \frac{T_s}{2}\dot{x},\ \ xs\Delta = k_{\dot{x}}(\dot{x} - \dot{x}_d) \qquad \textbf{(5.4)}$$

The sum of these equations give us the distance between the mass center and the toe when stance.

$$xs = xso + xs\Delta \qquad \textbf{(5.5)}$$

In this phase, the speed of leg is constant according to the mass center. By thinking this, orbit of the mass center according to the toe which is on the x_r, t_k is the time of stance; the equations are ; [3]

$$xs(t) = xso + xs\Delta - \dot{x}_d (t - t_k) \quad \textbf{(5.6)}$$

$$\phi d = \arcsin(xso + xs\Delta - \dot{x}_d (t - t_k)) \quad \textbf{(5.7)}$$

6. DESIGN OF ONE LEGGED 3D HOPPING ROBOT

6.1 The Method

System is designed based on the spring loaded inverted pendulum (SLIP) model. In this model, energy of the system is stored by springs and then given to the system again. It jumps in a vertical direction (z-axis) and gives its initial potential energy to the spring. By using this stored energy, system jumps.

Because we aim to design the system moving in 3 dimensions, we do not need any rotating axis. We need to rotate the leg to move forward. So we have 3 degree of freedom in the system.

We must examine the moves of jumping and walking before starting the design of the leg.

6.2 Jumping

This is the essential movement of the robot. To be succeed in walking, robot should jump. As a result of spring- mass system described by Hooke's Law; potential energy is stored in the spring. After that this energy is used to provide the initial potential energy. Because of the frictions and damping, some of this energy turns into heat that has negative effect in jumping. To prevent this effect, we need actuators to give the lost energy in the system. We can make this by using pneumatic cylinder or electrical motor. In this design, we choose electrical motor because of its easy application. Figure 6.1 shows the phases of hopping.

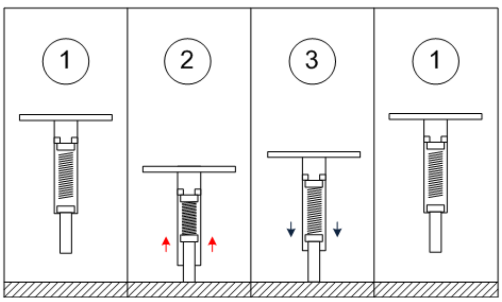

Figure 6.1: Phases of hopping.

These are ;

1 : leg is in the air, spring is free

2: touchdown phase, spring compression

3 : lift-off begins, spring gives stored energy inside

1: robot reached initial height

This electrical motor converts the rotational motion to linear motion. Because of the height control is directly related to spring compression, we can measure the length of the leg with length sensor and we stop the motor when we reach the lenght we want.

6.3 Walking

After jumping, robot can walk by its hip ankle. We can control walking with actuators too. For doing this, we need to measure the angle between the leg and the ground. If the angle is too big, robot can fall to the ground. So we should not have too much angle.

Another important point we need to discuss is the choose of electrical motor. So we should know the time of touchdown to decide the speed of motor.

$$T_s = \frac{\pi}{\omega_n} = \pi\sqrt{\frac{M}{k}} \qquad \textbf{(6.1)}$$

T_s : time of touchdown

If we put random mass and spring coefficient in this equation, we see that time is so short, like 0.120 ms. And if we think we have electrical linear actuator which has 5 mm pitch, we will add the energy to the system by the formula ;

$$\frac{1}{2}kx^2 = \frac{1}{2}.3500.(0,005)^2 = 0,0437 \text{ joules} \qquad \textbf{(6.2)}$$

This energy is not enough to compensate the energy loss in the system. To prevent this, we need to have motors which have big pitch. Normally, this affects the price of the motor.

After this discussion, we can see that optimization of the system is too complicated. Total mass and the spring coefficient effect the time of touchdown which is too short and at the same time we need fast electrical motor to jump.

6.4 The Design of The Robot

Figure 6.2 shows the overall design of one legged 3d hopping robot. The parts of it will be listed below.

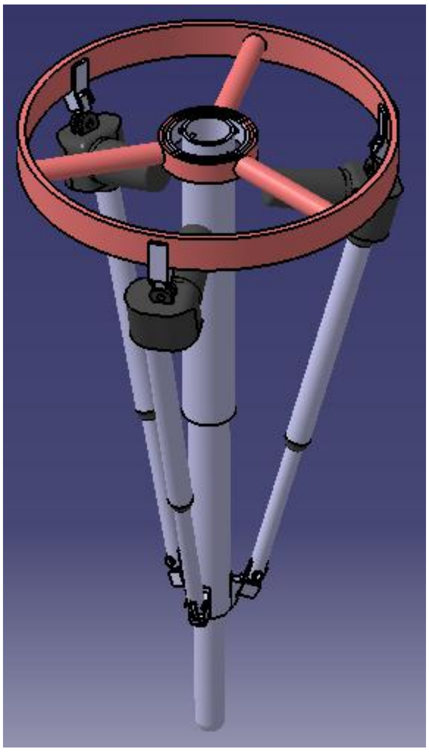

Figure 6.2: Overall view of the robot.

6.4.1 How it works ?

As it is said previous chapter, we drop the system from initial height. We have leg that have 3 parts and it has a spring with them. At touchdown, legs will compress but after that, as a reason of losses; it will not reach same height. To prevent this, we have 3 linear actuators. They have two effects to the system;

- If we apply three of them at the same time in touchdown phase, we will have strong extra force to reach initial height.
- If we could control the legs one by one or together, we would able to walk in direction we want to move.

Each linear actuator has 2 DOF. There is 2 DOF universal joint to hold the actuator to the base and similar joints to hold it to the leg. In total, there are 22 DOF in the system but 19 is dependent on each other due to parallel structure. So, system has 4 independent DOF .

6.4.2 Parts of The System

6.4.2.1 Leg

Figure 6.3 shows the structure of leg.

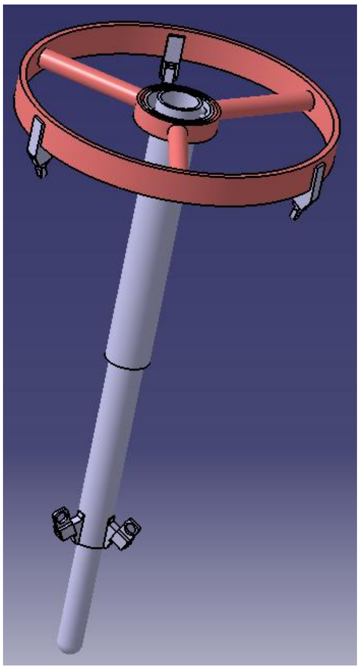

Figure 6.3: Structure of leg and its connection.

We decided to design leg with 3 parts. Part 1 is touching to the ground in the stance phase. Part 2 guides the leg and part 3 is the connecting part of the leg to the base . Hopping of the system is the result of compression of spring between part 1 and part 2. During stance phase, spring is compressed by actuators to pump extra energy to the system.

We need to discuss two parameters before design. First one is the mass of the legs. In previous chapters, we saw that mass of the legs effects the time of touchdown. Because of this, it should be better to use aluminum as material rather than steel to make it light.

Other concern is the length of the keg. If it is too long, there can be problem called flambage because of the mass and the impact at touchdown. So we do not design it long to prevent the flambage.

6.4.2.2 Linear actuators

Figure 6.4 is the electrical motor used in the system.

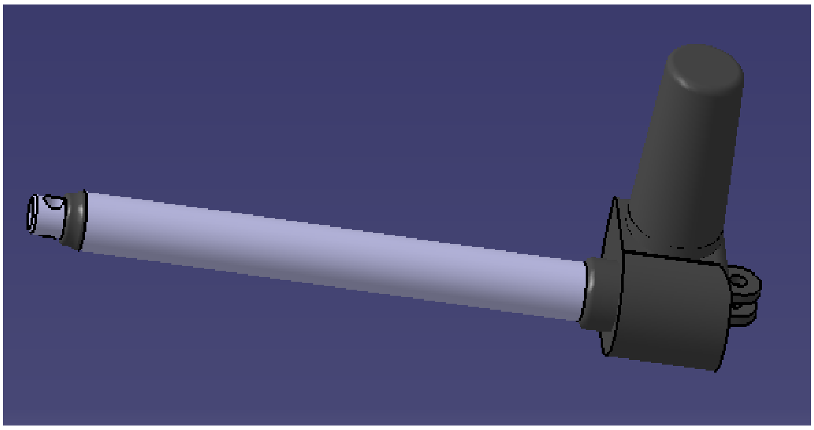

Figure 6.4: Electrical motor.

We want to move in three dimensions, so we use three linear actuators. We move by controlling the angle and the length of the actuators. When we are choosing them, we need to check some parameter like weight, speed, stroke and load.

Below we can see some examples of linear actuators. They have long stroke, less weight, high speed and big load. Figure 6.5 shows the technical data of the motor.

Technical data

	Unit	CARE 33A	CARE 33M	CARE 33H
Push load	N	2 000	1 400	800
Pull load	N	2 000	1 400	800
Speed (full load to no load)	mm/s	8 to 12[1)]	16 to 22[1)]	32 to 45[1)]
Stroke	mm	50 to 300	50 to 500	50 to 500
Retracted length	mm	S+150/162/193[2)]	S+150/162/193[2)]	S+150/162/193[2)]
Voltage	V DC	24	24	24
Power consumption	W	N/A	N/A	N/A
Current consumption	A	3,5	3,5	3,5
Duty cycle	%	15	20	30
Ambient temperature	°C	–10 to +50	–10 to +50	–10 to +50
Type of protection	IP	44/65	44/65	44/65
Weight	kg	1,5 to 2,0	1,5 to 2,0	1,5 to 2,0
Color	–	Black	Black	Black

Figure 6.5: Technical specifications of the motor.

It is benefit for us to choose CARE33H when we look at the table because even if it has not high load ; it has more speed compared to other models.

6.4.2.3 Joints

Figure 6.6 and 6.7 are the joints used in the system.

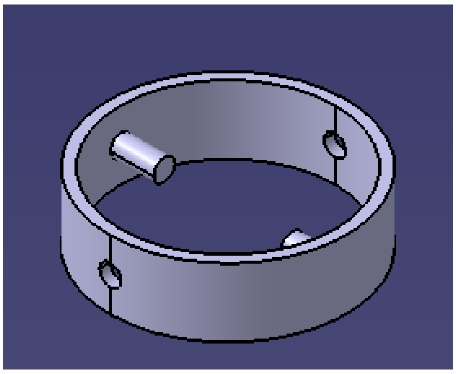

Figure 6.6: Upper joint.

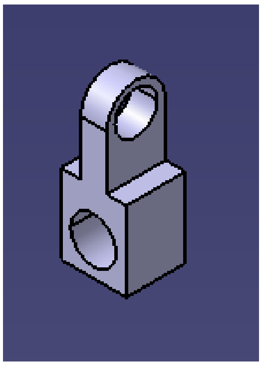

Figure 6.7: Lower joint.

These are another essential parts because they give degree of freedom to the system.

Upper joint helps the system to move up and down; lower joints help to rotate the leg when linear actuators are extracting.

6.4.2.4 Base

Figure 6.8 is the base of the system.

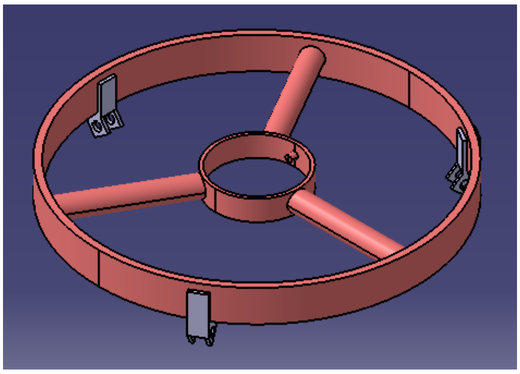

Figure 6.8: The base.

Leg and the actuators connected to this part. And also it carries other parts like battery. So this part should be strong. We need to decide this part's diameter related to the length of the leg because if we take the diameter too long when we have short leg, system could fall to the ground when jumping. And also we have to make it light to have long period of lift-off.

7. SIMULATION OF THE SYSTEM

7.1 Spring-Mass Without Damper

In this part of simulation, system is considered only with spring but without damper. System is thrown from initial height H and when touchdown occurs, legs begin to distract because of spring. When lift-off begins, legs begin to extract and system reaches to the same height because it has no damper.

Figure 7.1 shows the free body diagram of the system ;

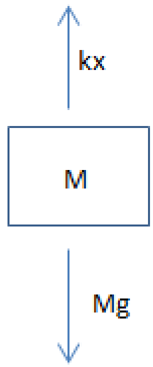

Figure 7.1: Free body diagram of spring- mass system without damper.

Equations of motion are ;

$$\Sigma F = M\ddot{x} \tag{7.1}$$

$$Mg - kx = M\ddot{x} \tag{7.2}$$

$$\ddot{x} = g - \left(\frac{k}{M}\right)x \tag{7.3}$$

M : total mass of the leg and body

g: gravity

k: spring coefficient

Figure 7.2 shows the simulink diagram of the spring mass system without damper.

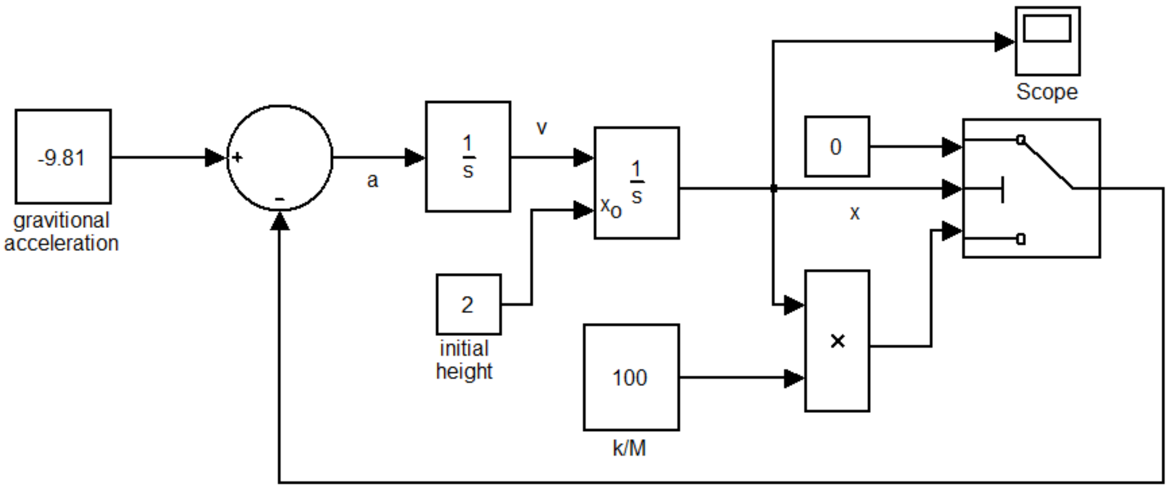

Figure 7.2: Simulation of spring mass system without damper .

We begin this simulation when we drop the system from initial height we choose. Until touchdown phase, there is no effect of integrator to the system. We need to add this effect by comparing the height of system (x). When $x \le 0$, spring begins to compress and until take-off phase ($x = 0$) we should add this effect to the system. Figure 7.3 plot of spring mass system without damper.

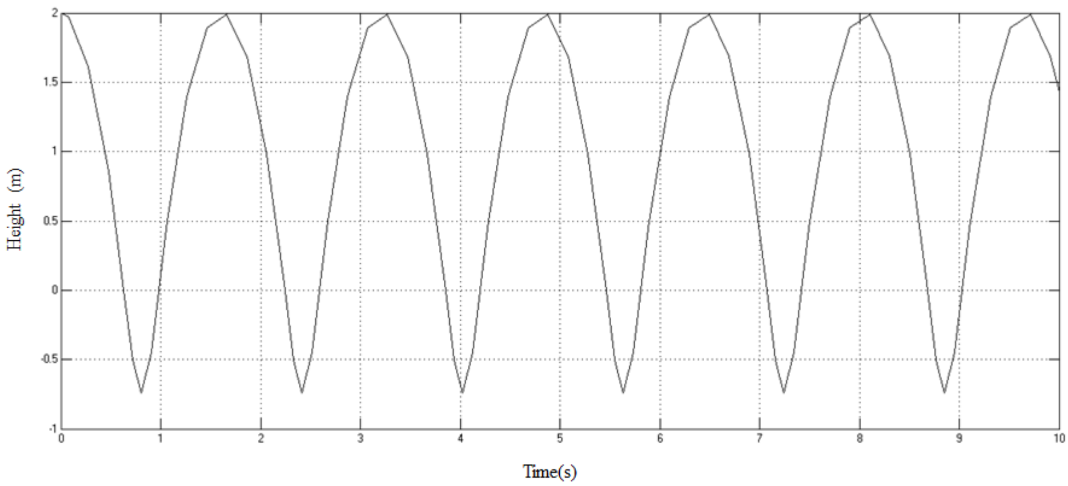

Figure 7.3 : Plot of spring mass system without damper.

Figure 7.3 shows that system reaches same initial height in every hop because there is no damper. System has sinusoidal movement and because of friction neglecting, system does not stop.

7.2 Spring-Mass System With Damper

In this part of simulation, system is considered only with spring and damper. System is thrown again from initial height H and when touchdown occurs, legs begin to shorten because of spring. When liftoff begins, legs begin to extract and because of the damper, system could not reach its initial height and in every hop loses its energy; then stops at the height where system is balanced at its own weight. Figure 7.4 is the free body diagram of the system with damper.

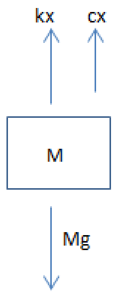

Figure 7.4 : Free body diagram of spring mass system with damper.

Equations of motion are :

$$\sum F = M\ddot{x} \tag{7.4}$$

$$Mg - kx - c\dot{x} = M\ddot{x} \tag{7.5}$$

$$\ddot{x} = g - \left(\frac{k}{M}\right)x - \left(\frac{c}{M}\right) \tag{7.6}$$

C: damping coefficient

Figure 7.5 is the simulink of spring mass system with damper.

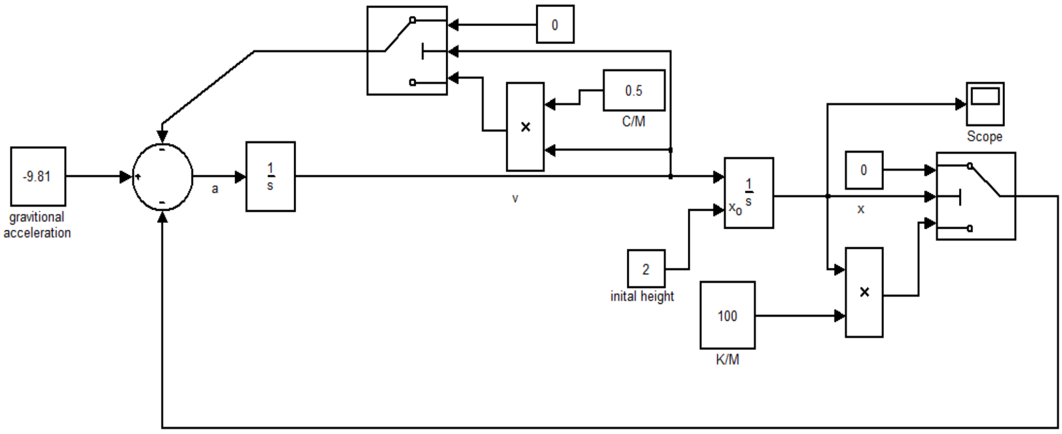

Figure 7.5 : Simulink of spring mass system with damper.

Similar to previous simulation, we drop the system from initial height. We have extra damper effect to the system. When $x \leq 0$, spring begins to compress and damper effects to the system related to its velocity. These two effects will continue until take-off phase again. Figure 7.6 is the plot of spring mass system with damper.

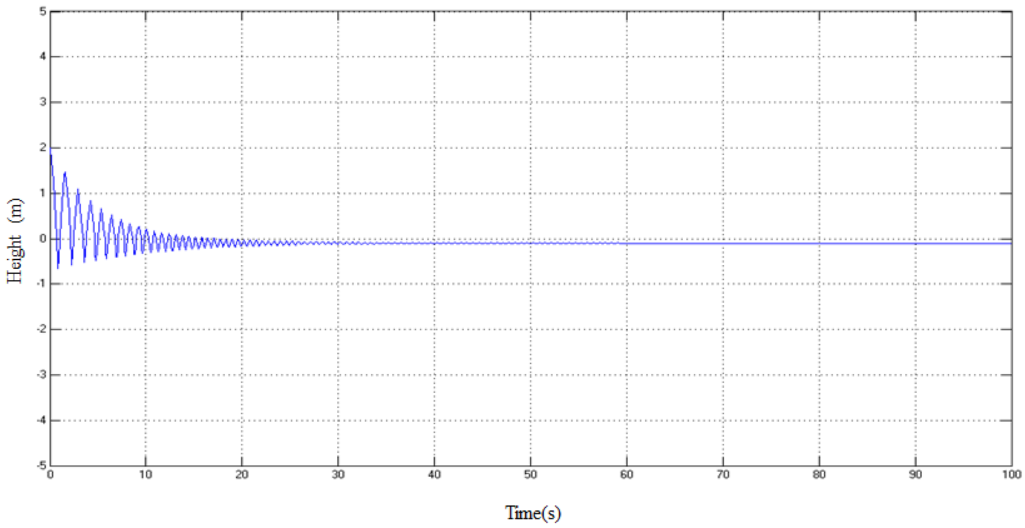

Figure 7.6 : Plot of spring mass system with damper.

Figure 7.6 shows us the effect of damper to the system. In every hop, system jumps to a lower height. And the result of oscillations, system stops in a period of time. The reason of the steady state in negative height is the total mass of leg and body.

7.2.1 Bode plot of the system

If we think of system with damper, we derive the transfer function ;

$$H(s) = \frac{1}{Ms^2 + cs + k} \quad \textbf{(7.7)}$$

When we put for M= 50kg, c=500 Nm/s ,and for k=1000 N/m we have ;

$$H(s) = \frac{1}{50s^2 + 500s + 1000} \quad \textbf{(7.8)}$$

By this transfer function using MATLAB, we obtain Figure 7.7 is the Bode plot of the system.

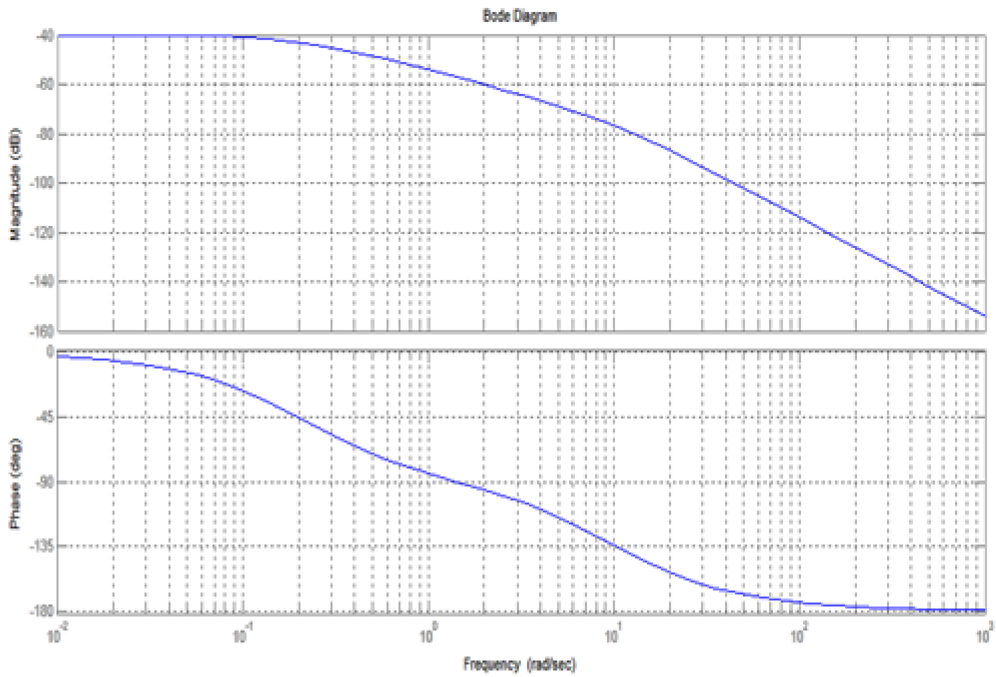

Figure 7.7 : Bode plot of the system.

Our system jumps too fast and because it has periodic motion, we need to know the frequency of it if we have resonance. We can use this plot as a guide to pick the electrical motor.

7.3 Spring-Mass System With Damper and Force

In the previous part, we see that we cannot have the same hopping height in every jump. In order prevent this, we need to add force to the system in every touchdown phase. Next part, it will be discussed the parameters effect energy that we need to add. Figure 7.8 is the free body diagram of the system with damper and force.

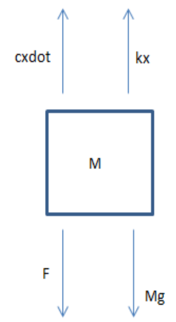

Figure 7.8: Free body diagram of spring-mass system with damper and force.

$$\Sigma F = M \ddot{x} \tag{7.7}$$

$$Mg - kx - c\dot{x} + F = M\ddot{x} \tag{7.8}$$

$$\ddot{x} = g - \left(\frac{k}{M}\right)x - \left(\frac{c}{M}\right) - F \tag{7.9}$$

Figure 7.9 is the simulink diagram of the system with damper and force.

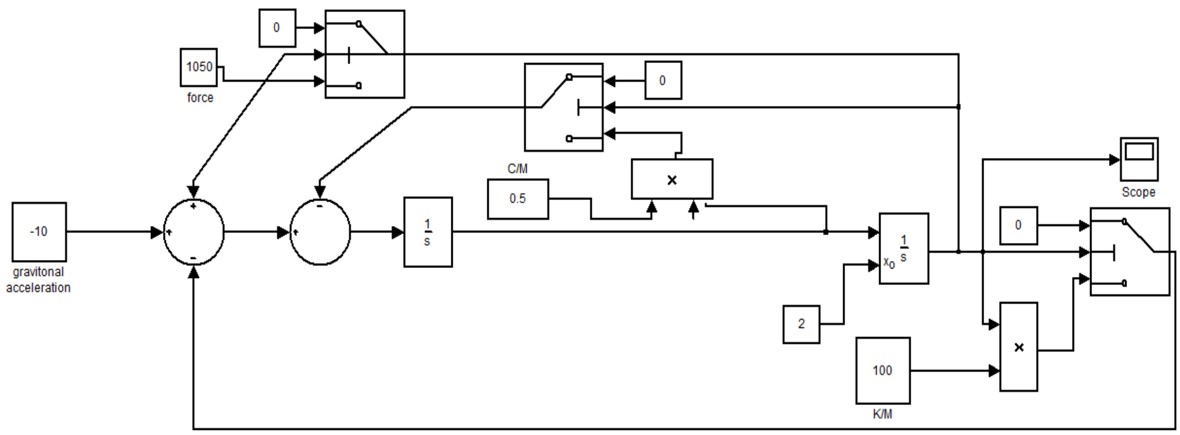

Figure 7.9: Simulink of spring mass system with damper and force.

As we see in the previous simulation, we drop the system from initial height. We have both damper and spring in the system. In order to obtain height at start, we add force block to the simulation. When $x \leq 0$, we add force related to the height of the system (x) until take-off phase. Next chapter, we will try to see what effects the height of the system. Figure 7.10 is the plot of spring mass system with damper and force.

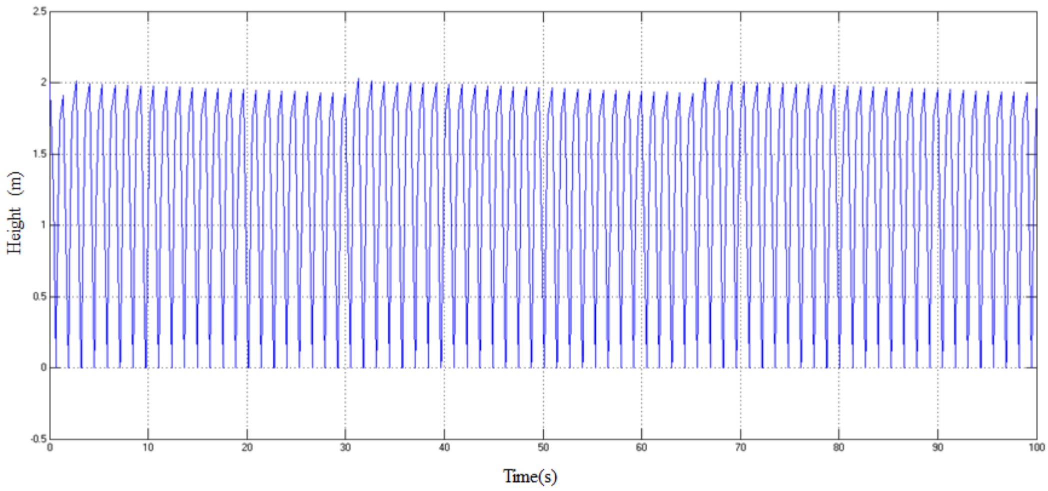

Figure 7.10: Plot of spring mass system with damper and force.

Figure 7.10 shows us directly that if we want to keep hopping height constant, we need to add extra energy when touchdown to gain energy lost by collusion and friction.

7.4 Energy Pumping Mechanism (EPM)

In the previous part, simulation is showed us that we need add extra energy to have same hopping height in every jump. An efficient energy pumping mechanism is an indispensable element of a self-sustaining hopping robot. This chapter will illustrate with a simple simulation result as to why the EPM is so critical for a hopping robot and hence set forth the objectives of the project in relation to the EPM.

7.4.1 Why we need it ?

Figure 7.11 shows a simplistic model of a hopping robot where the block M represents the body of the hopper while the block m represents the leg. The friction that is present in the actual mechanism is ignored in the analysis.

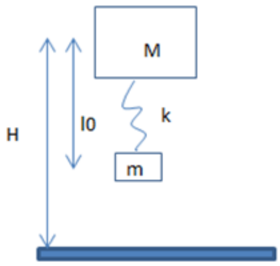

Figure 7.11: Simplified model of hopping robot.

In the Figure 7.11, M is the mass of the body and m is the mass of the leg. k is the spring constant of the spring used while l_0 is its free length. H is the height from which the hopper is dropped. Let us define an event, touchdown as the instant when the leg of the hopper touches the ground. Let us define another event, lift-off when the leg just leaves the ground. Let us first consider the case when there is no interim energy input to the system and for the sake of simplicity, the system is constrained to move along a fixed vertical axis. Due to the impact during touchdown, it can be assumed that the collision of the leg and the ground is perfectly inelastic. No further energy loss occurs in the system till the next touchdown. Let E_i be the total energy of the system just before touchdown and E_f be the total energy of the system just after touchdown. As per our assumption that there is no other source of energy loss in the system, the total energy of the system just after lift-off is also E_f .

$$E_i = MgH + mg(h - l_0) \quad \textbf{(7.10)}$$

Due to a perfectly inelastic collision ,

$$E_f = MgH \quad \textbf{(7.11)}$$

Now, let the hopper now hop to a height of h_1. Thus,

$$MgH = Mgh_1 + mg(h - l_0) \quad \textbf{(7.12)}$$

Which yields,

$$h_1 = \frac{MH + ml_0}{M + m} \quad \textbf{(7.13)}$$

After n hops,

$$h_n = \frac{Mh_{n-1} + ml_0}{M + m} \quad \textbf{(7.14)}$$

This equation shows that for a constant hopping height for the robot the m/M ratio should go to 0. In other words, a hopping robot without an giving energy can exhibit sustained hopping only if its leg is massless or its body is too heavy. Either of the two cases are not practically realizable. Hence for sustained hopping, we need a periodic input of energy equal to E_i- E_f to be pumped into the system. Now if we consider the contribution of friction to the energy of the system, the energy lost due to friction will be required to be pumped into the system along with the energy loss due to collision. Thus the periodic energy input will now equal E_i- E_f + E_{fr} where E_{fr} is the energy loss of the system due to friction present at all the joints. It may be of interest to note that if we increase or decrease the energy input about this required value then we get a control of the hopping height of the robot.

7.5 Height Control With Linear Displacement Actuation

Figure 7.12 is the simulink diagram of system with linear displacement.

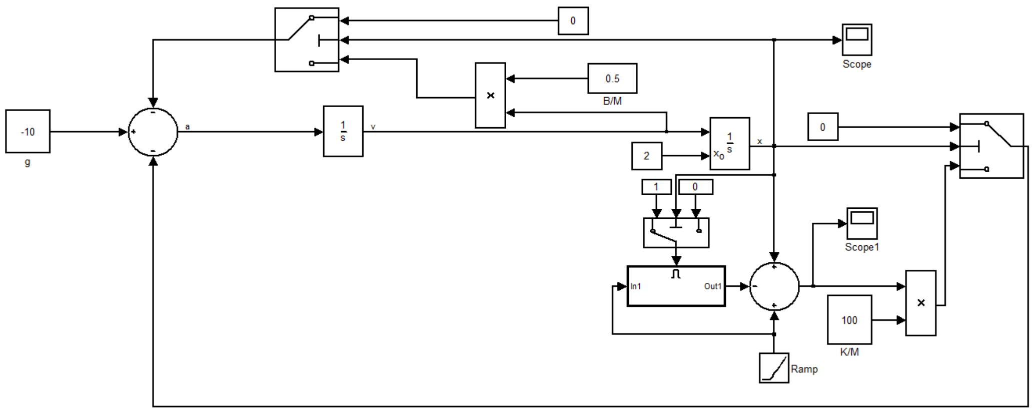

Figure 7.12: Simulink of spring-mass system with linear displacement.

In the previous simulations, we saw that we cannot control hopping height. We have to add extra force for jumping to the initial height. This chapter, a way of adding force to the system will be discussed.

Because of the system has spring to jump, easy way to have force is distracting the spring by electrical motor. By using this, we obtain spring force adding to the mass of the system and the leg.

We have nearly same simulation as we see in the previous part. The difference is the new part that we add for hopping initial height.

We use ramp signal for input because we need to trigger this input only in touchdown phase for a short time. To obtain this, we hold ramp input 0 in lift-off phase. In touchdown phase, we send ramp signal; because of the property of ramp input, we have sudden increase in spring displacement. As a reason of this, we have more force to jump. Figure 7.13 is the plot of spring-mass system with linear displacement.

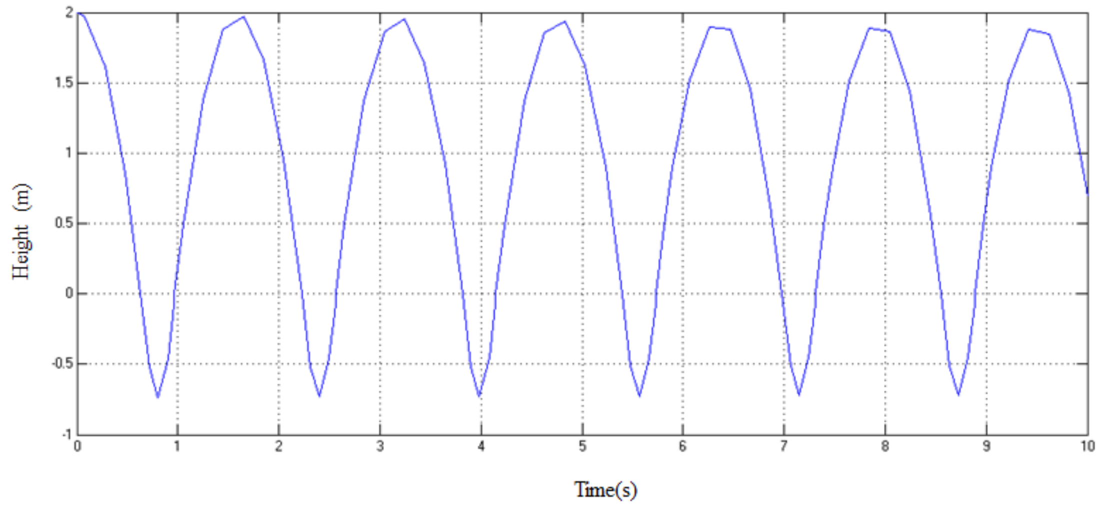

Figure 7.13: Plot of spring-mass system with linear displacement.

In the Figure 7.13, we see we have oscillation motion in the system as it is in the previous simulation. The reason of wide jumps is the delay of the ramp signal that we hold it in lift-off phase.

8. CONCLUSION

In this thesis, the simulation and the design of hopping robot is studied. Design criteria were chosen to keep the total weight of the system at minimum, to have an energy efficient system, to be able to manufacture a prototype using reasonable priced actuators available in the market. To control the system, three actuators perform the orientation control of the leg as well as external force control to the spring. 3D design was done using CATIA software. Simulation results are obtained using MATLAB SIMULINK. The future works of this study includes manufacturing of the prototype of the proposed system in this thesis and obtain experimental results.

REFERENCES

[1] **A. Sayyad, B. Seth, and P. Seshu**, *Single-Legged Hopping Robotics Research A Review*, Robotica, vol. 25, 587-613, Cambridge University Press, April 2007.

[2] **H. Rad, P. Gregorio, and M. Buehler**, *Design, Modeling and Control of a Hopping Robot*, IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS), 1778-1785, Yokohama, Japan, July 1993.

[3] **Raibert, M. H**., *Legged Robots That Balance,* MIT Press, Cambridge, MA, 1986.

[4] **Raibert, M. H..** *Symmetry in running*. Science, 231:1292—1294,1986

[5] **M. Ahmadi and M. Buehler**, *Stable Control of a Simulated One-Legged Running Robot with Hip and Leg Compliance,* IEEE Transactions on Robotics and Automation, 13(1):96-104, February. 1997.

[6] **Ogata, P**, *Modern Control Engineering*, Prentice Hall, Fourth Edition, 2002.

[7] **P. Gregorio, M. Ahmadi, and M. Buehler**, *Design, Control, and Energetics of an Electrically Actuated Legged Robot*, IEEE Transactions on Systems, Man, and Cybernetics, vol. 27B, no. 4, pp. 626-634, August 1997.

CV

Name Surname: Ercüment BAŞ

Place and Date of Birth : Çanakkale-20.02.1983

Adress: Güzeltepe Mh. Finanskent Konutları B2 Blok No: 61 Eyüp/İSTANBUL

E-Mail: ercubas@gmail.com

B. Sc.: Yıldız Technical University- Mechanical Engineering-2006

M.Sc.: İstanbul Technical University-Mechatronics Engineering- 2012