Optimal Design Methods for Increasing Power Performance of Multiactuator Robotic Limbs

by

Nathan M Cahill

A Dissertation Presented in Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy

Approved July 2017 by the
Graduate Supervisory Committee:

Thomas Sugar, Co-Chair
Yi Ren, Co-Chair
Matthew Holgate
Panagiotis Artemiadis
Spring Berman

ARIZONA STATE UNIVERSITY

August 2017

ABSTRACT

In order for assistive mobile robots to operate in the same environment as humans, they must be able to navigate the same obstacles as humans do. Many elements are required to do this: a powerful controller which can understand the obstacle, and power-dense actuators which will be able to achieve the necessary limb accelerations and output energies. Rapid growth in information technology has made complex controllers, and the devices which run them considerably light and cheap. The energy density of batteries, motors, and engines has not grown nearly as fast. This is problematic because biological systems are more agile, and more efficient than robotic systems. This dissertation introduces design methods which may be used optimize a multiactuator robotic limb's natural dynamics in an effort to reduce energy waste. These energy savings decrease the robot's cost of transport, and the weight of the required fuel storage system. To achieve this, an optimal design method, which allows the specialization of robot geometry, is introduced. In addition to optimal geometry design, a gearing optimization is presented which selects a gear ratio which minimizes the electrical power at the motor while considering the constraints of the motor. Furthermore, an efficient algorithm for the optimization of parallel stiffness elements in the robot is introduced. In addition to the optimal design tools introduced, the KiTy SP robotic limb structure is also presented. Which is a novel hybrid parallel-serial actuation method. This novel leg structure has many desirable attributes such as: three dimensional end-effector positioning, low mobile mass, compact form-factor, and a large workspace. We also show that the KiTy SP structure outperforms the classical, biologically-inspired serial limb structure.

DEDICATION

To my wife.

ACKNOWLEDGMENTS

First, I want to thank Dr. Thomas Sugar, for his advice and direction. Also, SpringActive Inc. for allowing me the use of their office space and their resources to build the leg. I especially want to thank Dr. Matthew Holgate who originated the actuation method presented here and spent many hours arguing with me and helping me formulate my ideas. I want to acknowledge Dr. Yi Ren for his advice and contribution to the optimization portion of my dissertation, and Dr. Joseph Davidson, for his expertise in Screw Theory. In addition, I want to thank my wife Leslea, for being generally amazing and gracious. Finally, I want to thank God for making me, and for helping me through the endless dark hours of dissertation writing.

TABLE OF CONTENTS

Page

LIST OF TABLES

LIST OF FIGURES

Figure Page

Figure Page

Chapter 1

INTRODUCTION

Biology offers strong evidence that robust and efficient navigation of terrain using legged robots is possible. Legged vehicles have many attributes which make them ideal as an alternative to wheels. Wheeled vehicles are capable of efficient and fast navigation of flat terrain. However, wheels do not perform as well over rough terrain, or on steep inclines. Legs allow steady navigation over bumpy and discontinuous terrain. In contrast with wheels, legs are able to support a body on an incline without added joint loads. This is due to their ability to passively produce a forces in certain configurations.

Assistive robots must be able to navigate the same terrain obstacles that humans do. Stairs and steps as well as other obstacles are common in human environments, and pose a problem for wheeled vehicles. Legged systems may therefore be a better candidate for operating in spaces built for humans. Aware of this potential, scientists have progressed the state of the art in legged locomotion continuously over the last several decades. Many legged systems exist which are capable of various dynamic tasks such as walking, and running. Some of these systems can perform animal-like tasks including balancing in the presence of ice and running over bumpy terrain. Even in the presence of such advancements, much progress yet remains before the performance of animals is to be matched by robots. Every legged robot system has strengths and weaknesses. BigDog, by Boston Dynamics is capable of highly dynamic motions.[54] It is stable even in the presence of large disturbances like a swift kick to the side. Its glaring weakness is that it is very inefficient compared to animals(as documented in several papers by other groups [60, 61, 38]). The Cheetah robot developed at MIT is capable of efficient locomotion, including running and jumping.[61] However, the robot has not shown the kind of dynamic maneuverability required to reject large disturbances and stay balanced on slippery surfaces. This may be due to the fact that

the legs of the robot are limited by their planar mechanical structure. Motions such as side-stepping are made impossible by the leg mechanics. A third system, called Atrias, is a bipedal robot which is capable of efficient gait, and disturbance rejection. It is also efficient relative to systems like BigDog[22, 59, 58, 57, 55]. Even so, it has been shown that the robot wastes considerable energy due to antagonistic motor power [2]. These robotic systems will be described in detail in the Literature Review chapter of this document (see Chapter 2)

This thesis develops methods which allow the design of specialized robotic limbs which are capable of increasing system performance on many levels. In Chapter 3 and 4 the design of the Kinematically Tunable Hybrid Serial-Parallel robotic limb (KiTy SP, or KiTy for short), which has a highly tunable leg structure that allows the leg geometry to specialized to a given position-force-velocity trajectory. The leg outperforms the classical, biologically-inspired serial leg structure for both unperturbed and perturbed gait trajectories, even when the geometry of the serial limb is tuned by the same optimization algorithm which was used to design the KiTy SP geometry (See Chapter 4). The overarching goal is to advance the state of the art in multiactuator robotic limb design, making it possible to algorithmically design complex, counter-intuitive robotic limbs, increasing power density and efficiency. This dissertation also explores how the passive dynamics of the limb can result in Mechanical Antagonism, which can cause dramatic reductions in leg efficiency and performance. This gives an analytical explanation for why tuning limb geometry to a task-specific force-velocity trajectory can decrease even the mechanical power requirements of the joints, not to mention the electrical power requirements at the motors. This exploration also led to the development of a Power-Optimal pseudoinverses for overactuated systems: one for overactuated serial arms, and one for overactuated parallel manipulators.

This dissertation also explores optimal design of compliance in multiactuator robot arms. Chapter 6 explores adding an Exhaustive Parallel Compliance Matrix (EPCM) to a general multiactuator robotic manipulator. The EPCM is a set of all possible combinations of parallel springs, one across each joint and mulit-articulate springs across each joint combination. We then develop an algorithm which efficiently produces the optimal stiffness and offset values for the spring matrix, including a set of feasibility constraints. Implementing the spring matrix reduces the sum of square power exerted by the actuators by up to 47 percent, the peak power requirement by almost 40 percent, the sum of squared current by 55 percent, and the peak current by 55 percent. We also test the design robustness using a perturbation study which shows that the parallel springs are effective even in the presence of trajectory perturbation. This dissertation rigorously develops optimal design methods encompassing a broad area of robotic limb design. In so doing, we prove the potential of the task-specific paradigm for designing robotic limbs, which may lead to a dramatic increase in the power density of legged robot systems.

The motivation to focus on the increase of power density is justified by the fact that the performance of power storage and generating machines has not been growing at the same rate as other areas of technology, most importantly: information technology. This disparity is well-known, and has also been documented in literature. To give insight about the scale of this trend a small example is presented here (for a detailed study see [41, 19, 5, 6]). The example we give shows the growth of both information technology and power storage technology according to published literature from the year 1990 to 2003. In 1990 one dollar would buy 2 Megabits of hard drive storage, .00283 MIPS of processing power, and transfer 19 kilobytes per second over a one kilometer distance [41]. By 2003 a dollar could buy 5558 megabits of hard drive

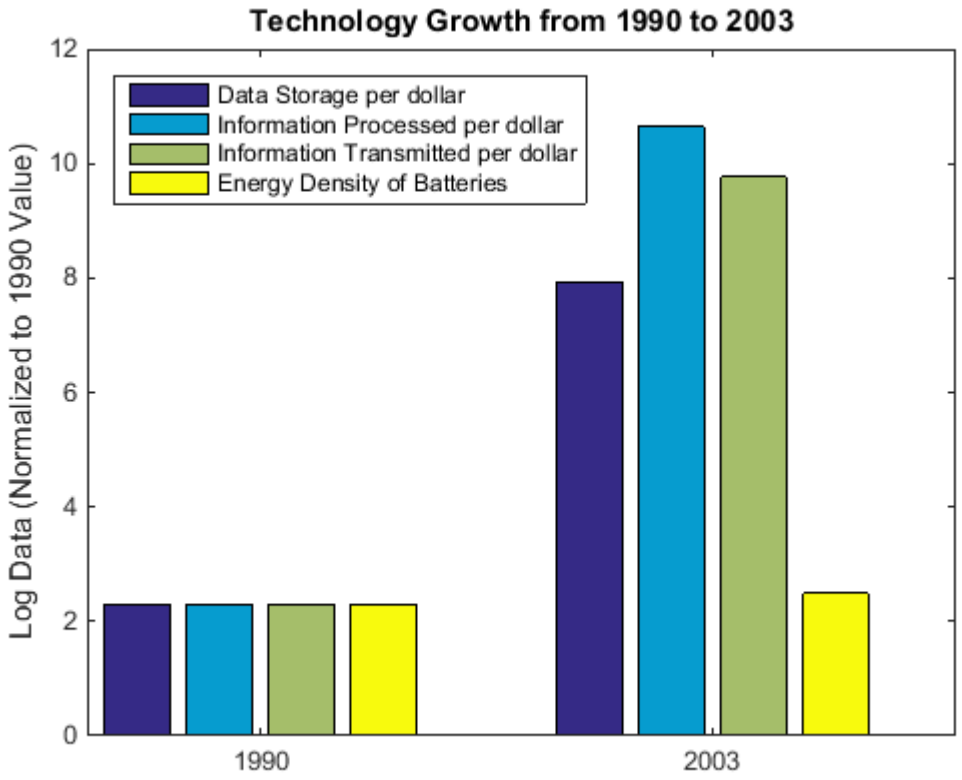

Figure 1. Growth in information technologies vs battery technologies, as documented in published literature, from 1990 to 2003.

storage, 11.8 MIPS of processing power and transfer more than 33,000 kilobytes per second over the distance of a kilometer. Today the information technology continues to get lighter, cheaper, and faster. Though it seems the literature has not reported new figures on the measures given above. Instead a rough, but insightful example will be given for perspective. Today, a bottom-shelf processor system (which is commonly used for robotic applications) would be the Rasberry Pi 2. It can achieve 4744 MIPS [37], for a price of $29.95. This calculates to 158 MIPS per dollar, but the Rasberry Pi has 1 gig of ram, a graphics core, and several other useful features. Even so 158 MIPS per dollar represents almost four doublings since the reported 2003 figure. This steep exponential growth is unmatched in most other areas of technology.

To show the contrast, we examine the growth of battery power density during the last 3 decades. From 1990 to 2004 power density increased from 100 watt hours per kilogram to about 120 watt hours per kilogram [5]. Again, it appears that literature fails to report values for this statistic after 2006 (even recent papers like Benson's 2015 paper analyze the trends of data which stops around 2006). but a very crude analysis of today's battery technology can show that the technology has not grown in leaps and bounds similar to information technology. Today a top of the line battery has an energy density of about 250 watt hours per kilogram [18]. This represents one doubling in the last 15 years, and just over two in the last 30 years. If improvements in battery technology had been trending with information technology since 1990, we would see batteries on the market today with power densities on the order of 800,000 watt-hours per kilogram. Similar to battery technology, the performance improvement of dc motors, hydraulic pumps, and gas/diesel engines have not been able to keep up with information technology (for more detailed analysis see [6, 19]).

It is clear that performance growth of energy storage and generating systems is far out-matched by the performance growth of information technology systems. This dissertation is an effort to stimulate an increase in the power densities of robotic limb independent of growth in batteries, or electric motors. This may be achieved by reducing losses that occur due to mechanical antagonism or by adding spring elements which store and amplify motor powers.

This document may be broken into three parts: the first part introduces the KiTy SP robot limb and describes its many performance related benefits. Next, optimization methods are introduced which are able to specialize the geometry of complex, multi-actuator robot limbs. This allows increases in performance and efficiency. Finally, an effort has been made to understand the force-velocity space of a robot limb and

how force velocity direction can have a dramatic affect on power usage. The first section was published at the 2016 ASME IDETC conference. These methods may help the designer intuitively understand the problem. These three topics represent the majority of the work herein. A brief introduction to each topic will be given next.

1.1 The Development of the Hybrid Parallel-Serial Limb

The robotic limb described here incorporates a novel spatial mechanism that combines many benefits of series and parallel actuation, while avoiding many of their typical weaknesses. These include the reduction of the mobile mass of the limb, and the ability to support loads in line with the leg in a nearly passive manner. The leg also has a large workspace with a compact form factor, which is traditionally a weakness of parallel manipulators. This thesis will detail the many phases of the development process of such a leg. Firstly the motivation for the design of the serial portion of the limb will be described. This portion is important because it defines the set of forces and torques which the leg may passively support. Secondly, the solution of the kinematic equations which govern the motion and the force interaction of the leg will be shown. Thirdly the several iterations of design which led to the final working prototype[10].

1.2 Screw Theory and Antagonistic Motor power

Ken Waldron published a paper in 1981 entitled "The Relationship Between Actuator Geometry and Mechanical Efficiency in Robots" [72], in which he introduced a phenomenon later called mechanical antagonism [2]. Section 2.3.1 describes the little

studied phenomenon which can result in very significant power losses. This dissertation contributes to the understanding of motor antagonism in two ways. Firstly, the force-velocity space of the end effector is defined and a visual method of understanding the phenomenon is introduced. Secondly over actuated systems are also considered and a new pseudoinverse for the Jacobian matrix is defined which minimizes motor antagonism instead of motor speed. These results have only partially been introduced in this document. They will be written up in full and included in this dissertation.

1.3 Optimization Methods

As a byproduct of the design effort summarized above, the author has discovered several useful methods for the optimization of multi-actuator limbs. Using optimization tools in the Matlab environment, useful geometries were discovered which minimized performance indices given task specific force and velocity data. On every iteration, the optimization algorithm runs a simulation of the leg moving through the design trajectory. It then sums the electrical power required to perform the task. This sum is the cost function by which the current geometry is judged. The author also used the condition number of the leg Jacobian as a constraint in the algorithm. Since the condition number of a matrix is a measure of nearness to singularity, the number has been used to restrict an algorithm from selecting geometries which cause near singularities in the leg as it moves through the goal trajectory. Optimization methods have also been developed for other aspects of the leg design.

The motors which drive the leg may be installed with a gear box. The gear ratio of this box will greatly impact the electrical power required to perform the task. Other constraints include the motor torque and speed limitations. Contained in this thesis is

a formulation of the KKT conditions for this optimization problem and then presented are the solutions for the particular design and gait trajectory used through the paper. The gear ratio optimization and the geometry optimization have been presented here (and a paper on the work has been accepted for publication at ICRA 2017).

1.3.1 Compliance Optimization

An efficient method for designing a limb with compliance has also been developed. Compliance may be added to the leg in parallel with the joints or in series. The author defines a matrix containing stiffness values for all possible joint springs, including multi-articular springs which cross more than one joint. A convex optimization problem has been formulated which efficiently finds an optimal solution for the this stiffness matrix. This analysis has been completed but has not been written up or implemented in the robot.

Chapter 2

LITERATURE REVIEW

Robotics is a science that covers many areas of expertise. Consequently there is a very broad area of work that forms the published background of this dissertation. This chapter will cover several varying bodies of research which include: existing legged robot platforms, parallel robots especially the Delta Robot, kinematics, optimization of geometry, optimal motor selection and gear box design, relative works documenting important aspects of biological limbs and animal gait, and compliant actuation. This chapter will be used as an index of the body of work referenced by the author.

2.1 Legged Robots

There have been several notable legged robot platforms developed recently which make up the state of the art of legged robots. Each platform has notable strengths, and weaknesses, which will be discussed in the following section. None of the platforms are yet competitive with biology.

2.1.1 BigDog

Big Dog (Figure 2) is a quadrupedal robot designed by a company called Boston Dynamics. This robot has a robust controller capable of complex overground terrain navigation. Little has been published in scientific journals or conferences about this robot. What is known about the system comes from a single conference paper from 2008[54] and the many videos that Boston Dynamics have published to the Internet. The mechanical structure of the limbs is similar to that of a biological dog's leg. Each leg has 5 actuated degrees of freedom: a three DOF hip joint, along with single DOF knee and ankle joints. In addition to these powered joints, the robot

also has one passive degree of freedom at the foot. BigDog's controller is capable of sensing and navigating complex terrain, avoiding falls - even in the presence of extreme disturbances such as a kick to the side - and carrying heavy loads [9]. A large drawback of this robotic system is that it is highly inefficient compared to biological animals. BigDog, has "a cost of energetic transport rating of over ten times that of a human" [22].

Figure 2. BigDog Robot traversing rugged terrain [9]

2.1.2 Atrias

Jonathan Hurst has designed an impressive legged robot which he calls Atrias. He has published several works detailing the design process of the robot. In them he details

his many contributions to the state of the art in robotics. One such contribution is the mechanical design of his leg. Aware of the extensive literature on the Spring Loaded Inverted Pendulum (SLIP) model, the team designed the robot's leg to approximate these dynamics [22, 32]. This approach simplified the robot's controller and paved the way for the efficient gait that the robot eventually achieved.

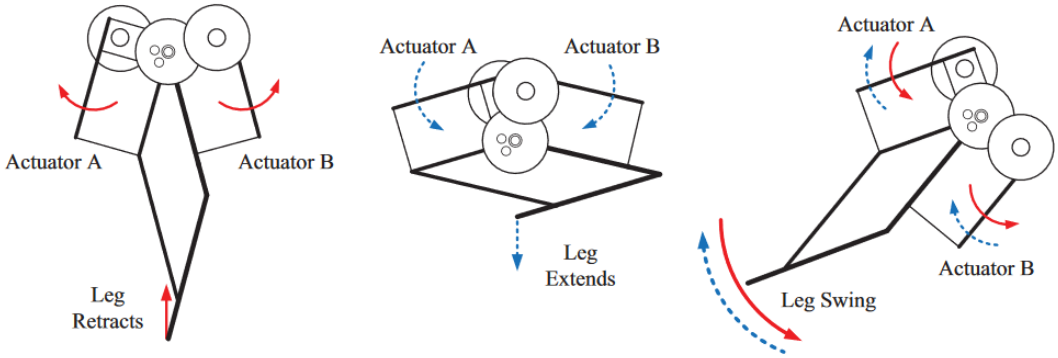

Figure 3. Atrias robot leg mechanism [22].

The robot's leg is controlled by three motors. The first motor controls the legs adduction-abduction motion. The other two motors are mounted as near as possible to the center of mass of the robot on the segment controlled by the first motor. Each motor actuates the hip joint of the robot via large leaf springs. The leg mechanism is a four-bar or pantograph mechanism which is constructed from lightweight carbon fiber tube. This approximates the weightless leg assumption in the SLIP model. The passive dynamics of the leg are not, however, a perfect match with the SLIP model. In later work, the deviations from the model were studied extensively and a "visually tactile" analysis gives insight into the discrepancies and how they affect the gait. [1].

The Atrias robot has a walking controller that has been well documented in scientific literature. The original published design consisted of a force controller [38, 40]. The intent of the controller was to provide the robot with a stable hopping gait

even over uneven surfaces. The controller attempted to match the force profile of an equivalent undisturbed spring-mass model. The force profile being the shape of the ground reaction force curve over time during the period when the robot's toe is in contact with the ground. Matching the force profile perfectly with the spring-mass model will give the center of mass of the robot the desired acceleration profile and therefore the desired trajectory in position and velocity space. This controller was robust to ground height and stiffness disturbances in simulation. However, under some conditions force control failed due to actuator limitations. To overcome this problem an impulse controller was developed, which controls the impulse profile instead of the force profile [39]. This controller proved to be more stable than the force controller, allowing the robot a more robust recovery from ground height and stiffness disturbances. The passive dynamics of the robot have been studied intensively [56, 1]. but in spite of this the system suffers from losses due to motor antagonism[2] (see section 2.3.1).

2.1.3 Cornell Ranger

Andy Ruina has designed another walking robot and made several contributions to the state of the art in walking robots. The robot is called Ranger(see Figure 4). It is a highly efficient walking robot that is based on the design of passive dynamic walkers. A paper published in 2012 by the group claimed that the robot had walked the furthest on a single battery charge, and obtained a lower cost of transport value than any other walking robot. It achieved this by leight weight mechanical design throughout, knee-less carbon fiber nearly planar legs, foot actuation that combines toe off and ground clearance, and a low power electronics system [7, 60].

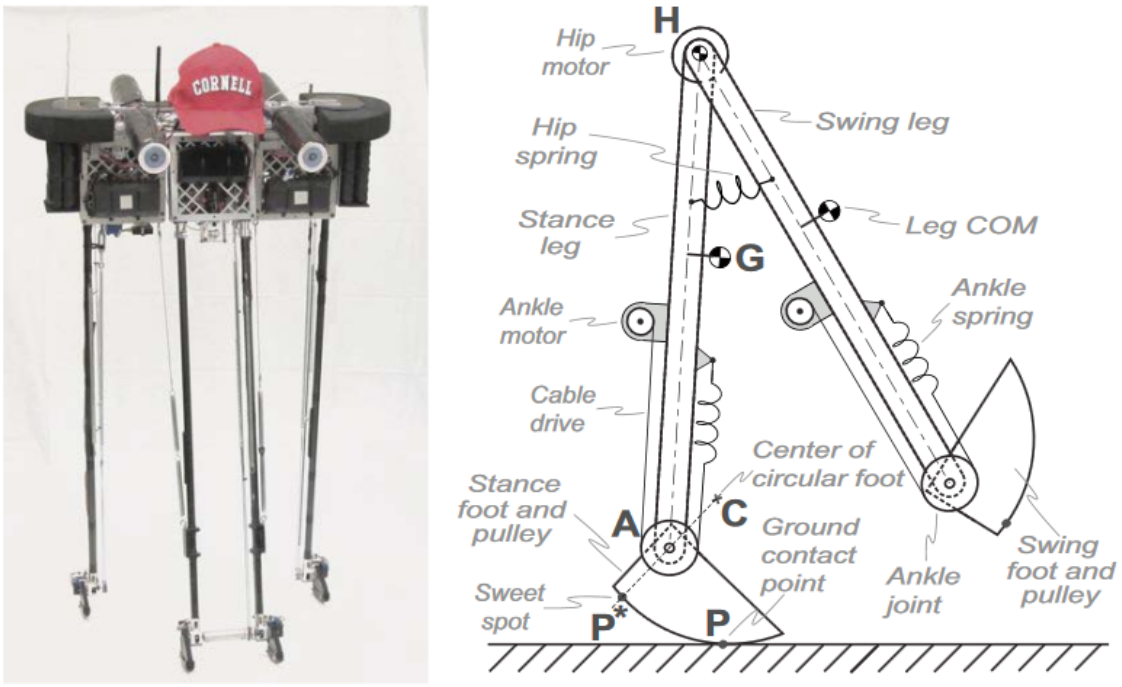

Figure 4. Ranger: efficient bipedal powered dynamic walker[60]

2.1.4 MIT Cheetah Robot

Sangbae Kim has developed a quadruped robot capable of robust, fast, and efficient movement over uneven terrain. In a 2013 paper, Kim outlines his design principles for efficient quadrupeds [61]. The paper emphasizes a regenerative approach which includes, custom designed high torque density motors, and a low impedance transmission. The cheetah robot uses a single stage gear box with a ratio of 5.8:1 to reduce the torque required at the motors. This is a relatively small ratio compared to typical gear ratio values in robotic systems. A smaller gear ratio reduces reflected inertia which is a major contributor to losses in legged locomotion. The mechanical design of the robot's limb is primarily concerned with the reduction of distal mass. The hip and knee motors are located co-axially at the hip joint. Torque from the

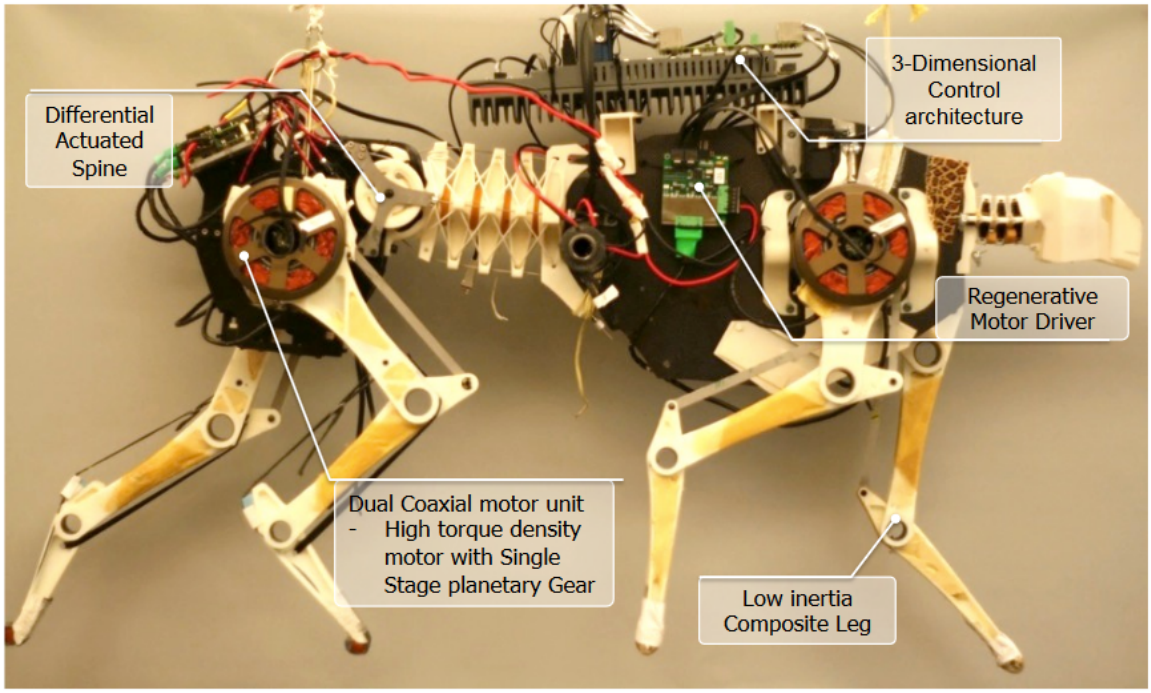

Figure 5. The MIT Cheetah, a quadruped with low inertia legs and transparent torque transmission[61]

motor is transferred to the knee joint through a mechanism. This moves the mass of the motor to a proximal location reducing the inertia of the leg [62]. The robot controller has also been well documented in scientific literature. The robot has a heirarchial controller. The low level controller is an impedance controller which is able to dynamically change the effective stiffness of the leg [25]. The higher level controllers include gait pattern generators and ground reaction force modulators. These controllers allow the robot to stably traverse rugged ground[36]. The details of this controller will be omitted as the primary focus of this dissertation is mechanical design of robot limbs.

2.1.5 HyQ

HyQ is a quadruped robot designed by a team led by Darwin Caldwell at the Instituto Italiano di Tecnologia. The robot has hyrdraulic actuators and stiff leg segments. This combination gives the designers the ability to dynamically adjust the effective stiffness of the leg as it interacts with the environment. The robot can walk trot and crawl. The power source for the robot is an off-board hydraulic pump. The robot has three active degrees of freedom per leg[8]. The group has explored many

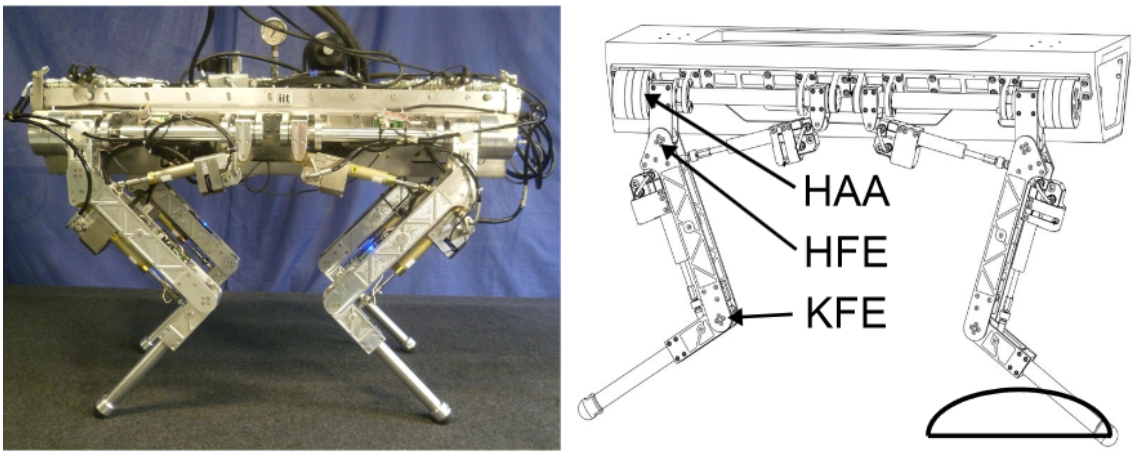

Figure 6. HyQ, a stiff quadruped robot actuated by hydraulics[8]

advanced topics in control and trajectory planning. Including deriving continuous equations that describe the nonlinear effect of the robot's foot colliding with the ground. [50].

2.1.6 Starleth

Starleth is a quadruped system which has been developed at the Swiss Federal Institute of Technology(see Figure 7). The robot utilizes DC motors, and harmonic

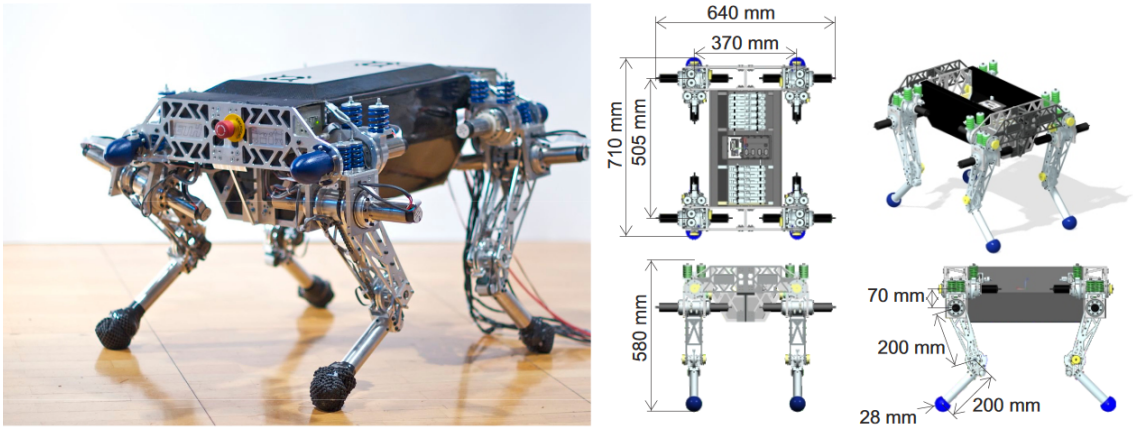

Figure 7. StarlETH, a compliant quadruped developed at the Autonomous Systems Lab (Photo: Francois Pomerleau) [33]

gear drives with a 1:100 ratio. The motors have been decoupled from the inertial load of the leg via series springs. These springs act as a mechanical low pass filter and a means for energy regeneration during gait. The weight of the motors has been moved to the hip joint, and actuates the leg via a chain/cable and pulley system. The springs are pre-compressed to avoid backlash. The leg design allows for both accurate position and force control. The robot is being used as a platform for on going controls research including a learning-by-practice method which enabled several high dynamic movements. [34, 35, 33, 20].

2.1.7 Rush

Rush is a quadruped robot with only one actuator per leg. Each leg has two springs which add compliance to the system. One of these springs acts across the knee joint, passively storing and releasing energy. The other spring is what the author calls a "direct acting spring device". It is a linear spring which acts along the robot's thigh, shortening and lengthening the distance between the hip and the knee joints.

The authors claim that it helps to absorb impact force from the robot's feet colliding with the ground during gait. The robot is capable of a bounding gait and pronking gait. The authors assume that steady-state running is governed by the spring-mass dynamics, and use a rhythm generator and delayed feedback controller to achieve this steady-state [74].

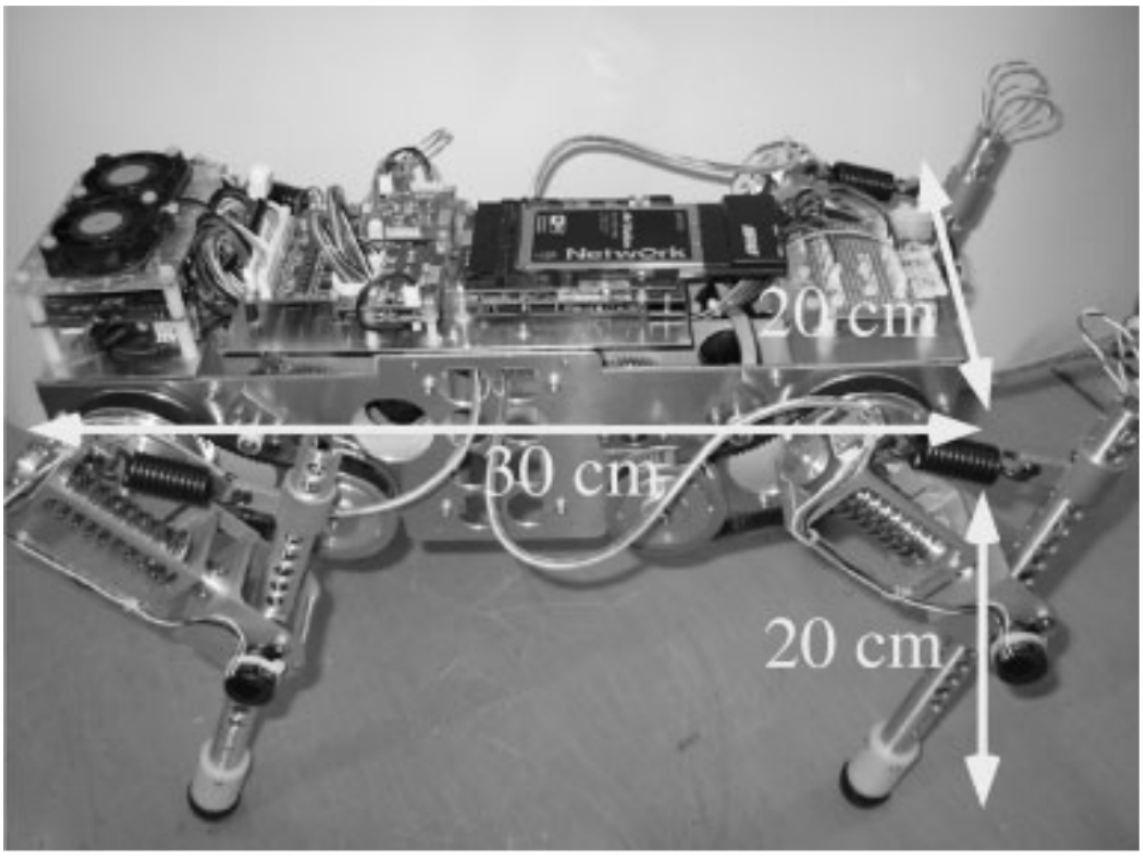

Figure 8. Rush, a quadruped running robot [74].

2.1.8 Ernie

Ernie is bipedal robotic platform. Each of his legs has two segments, thigh and shank, and two actuated revolute joints, hip and knee. This means the leg is

constrained to the plane normal to the hip and knee joint axes. The actuators are mounted to the torso and actuate the joints through wire cable and pulleys. The group used simulation data in an iterative process to select the lightest motor which would be capable of achieving gait. The knee is equipped with a parallel spring which

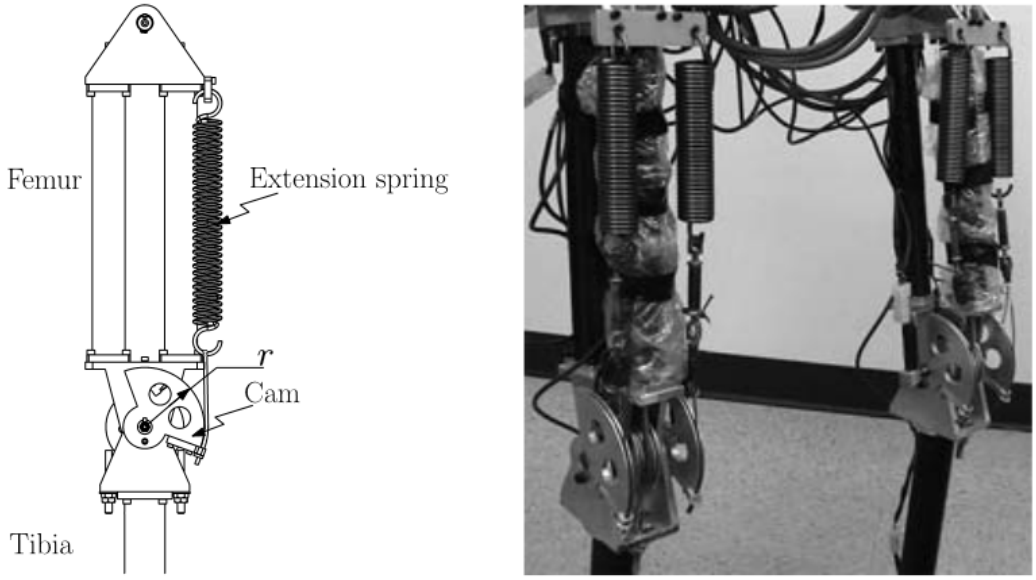

Figure 9. The parallel knee spring of Ernie, the bipedal walking robot with parallel stiffness [73].

actuates the joint passively along with the motor. The group performed an experiment to see how spring stiffness affected power use. Gaits of varying speed were designed especially for a number of knee spring stiffness. Each spring stiffness getting its own gait. It was shown that the average power required to walk could be decreased using a spring at speeds above .4 meters per second. Also, softer springs were more efficient at lower speeds but stiffer springs became more efficient at higher speeds [73]. It seems the optimal spring stiffness is positively correlated with walking speed.

2.2 Delta Robot and Delta Inspired Systems

The Delta Robot is a three DOF robot which has seen great success in industry. It is a parallel robot which is constrained to translate in three directions. It has three passive constraints which restrict the platform from rotating relative to the base. Analytic solutions to the forward and inverse kinematic equations have been derived as well as the inverse statics, and the inverse dynamics [51, 53]. This makes the robot excellent to use for applications such as pick in place. These robots have a very low mobile mass. The motors are fixed to the base, while the rest of the manipulator may be made from lightweight materials. Each of the three motors are attached symmetrically on the edges of an equilateral triangle at the base. The axes of the motors being coincident with each respective edge. Each motor applies torque about its axis to the first link of a 5 freedom kinematic chain which connects eventually to the platform. The three kinematic chains passively constrain the platform from rotating, while simultaneously transmitting the force generated by the motors to the platform. The motors control the position of the platform by controlling the angle of their respective joint. Again, the motors cannot change the orientation of the platform: it is fixed by the kinematics of the 3 parallel chains. This fixed platform orientation is useful for applications like pick and place where the object being manipulated does not need to be rotated.

A limitation of the delta robot, and indeed many parallel robots, is the robots workspace. The workspace of the robot is the space in which the end-effector can travel. Serial robots generally have a much larger workspace than parallel robots of similar dimension. Another limitation is that the robot cannot change the orientation

of the platform. Many variants of the Delta Robot have been created which address these limitations. Some of these systems will be addressed in the following subsections.

2.2.1 NUWAR Robot

Karol Miller at the University of Western Australia has done work to improve the workspace of the delta robot. Her robot, the New University of Western Austrailia Robot, has specialized geometry which maximizes the robot's workspace volume. It differs from the Delta Robot in that the motor axes are set at an angle from the sides of the equilateral triangle. The effect of varying these angles is studied exhaustively with respect to workspace volume. She uses AutoCAD to nuerically estimate the volume of the workspace [46].

Originally her work only considered the volume of the robot's workspace, but in a later paper she uses a manipulablility metric (η_1) as part the design fitness function. The metric, used here as the first of two performance indices, was adapted from Gosselin and Angeles [21]. It is the integral of the inverse condition number of the robot over the entire volume of the robot workspace:

$$\eta_1 = \frac{\int\limits_{V_w} \frac{1}{\kappa_J} dV_W}{V_W}, \tag{2.1}$$

where V_w is the workspace volume and κ_J is the condition number of the jacobian as a function of the end effector position. The function represents the quality of the workspace volume, as measured by the manipulability. This metric is bounded

between 0 and 1. It approaches 0 for a robot which is singular in all positions and unity for a robot which is perfectly kinematically isotropic throughout the workspace.

The second performance index is the space utilization perfromance index, and defined as:

$$\eta_2 = \frac{V_w}{Bounding\,Box\,Volume}. \tag{2.2}$$

She defines the bounding box volume as: "the smallest rectangular prism, whose sides are parallel to the global coordinate axes, which contains all fixed actuators and every point within the workspace." The utility function is a weighted sum of these two indices:

$$\eta = w_1\eta_1 + w_2\eta_2 \tag{2.3}$$

Defining the utility function this way helps to balance two competing goals: the workspace of the robot is large and that for every point in the workspace the robot has sufficient ability to manipulate objects.

The group maximized this function in a brute force method by performing an exhaustive search in a wide range of geometric parameter values. Using this method they created a linear delta robot This allowed them to find geometries which had a large workspace volume and good manipulability across the volume of the workspace [47, 45].

2.2.2 HEXA Robot

The HEXA is a 6 DOF robot which is fast and easy to model. The robot is inspired by the design of the Delta Robot, but it has 3 additional actuated degrees of freedom which allow the robot to control the orientation of the platform. The design of the robot consists of 6 linkages which connect from the base to the robot platform. These linkages are arranged in pairs (see Figure 11). If the two motors actuating each linkage pair are moved in unison the robot behaves exactly like a Delta robot, and the platform is constrained to a constant orientation. This endows the robot with the simple modeling of the Delta if only translation is desired. However, the orientation of the platform may also be controlled by decoupling the motion of the motors in each linkage pair. This makes it convenient to achieve tasks which may separated into subtasks consisting of pure translations (like pick and place applications) and rotations (like orienting for insertion) [52].

2.2.3 Delta Inspired Robot with an Off-Board Wrist

Another variation of the Delta robot is documented in a patent by Lung-Wen Tsai [66]. The patent describes a Delta robot which controls the translation of a first platform. A second platform is added which is fixed in position but its orientation is controlled by 3 additional motors. The interaction of the two platforms allows 6 total degrees of freedom. This type of robot could be used in applications like machining: the tool could be fixed to platform one while the part could be fixed to the second. This would allow highly complex parts to be machined with precision.

2.2.4 Delta-6

In their paper on the HEXA robot Pierrot et al describe another 6 degree of freedom Delta variant [52]. This variant adds a motor along each of the three links connecting to the base (this structure is shown in figure 13). This allows the platform to translate and rotate in any direction. A downside of this is that he mass of the three added motors move as the first three motors control the translation of the platform.

2.3 Passive Dynamics

2.3.1 Antagonistic power

In 1981 Waldron et al published a paper in which they describe a phenomenon which is an important source for potential energy loss in legged walking vehicles [72]. Thirty five years later Abate et al show that this is indeed a source of energy drain in Atrias, a state of the art legged robot. The phenomenon, which he calls mechanical antagonism, happens when one motor in a limb performs negative work or braking energy while another motor performs positive work [2]. If the actuator is doing net-positive work on the environment, this means the motor(S) doing positive work must produce the power needed at the end effector and compensate for the breaking energy produced at the antagonistic motors. The sum of the input work must equal the work done on the environment. In the cited paper the author shows that antagonistic work can play a significant role in the power required to perform a task. The mechanics of the Atrias robot were analyzed. It was shown that the power required to perform legged gait was increased by a factor of three due to motor

antagonism. In the paper a serial limb with a structure similar to the human leg was also analyzed. It was shown that the mechanics of the serial limb were well suited to gait. There were no power losses due to antagonism.

2.4 Animal Gait

In a review of a collection of work over several decades, Alexander et al. show that animals store elastic energy in their tendons to reduce the metabolic cost of locomotion [4]. Not only this but tendon recoil is faster than muscle contraction allowing animals to jump further than would be possible without elasticity. So animals make great use of tendons as biological springs. This shows that legged gait is aided by the ability to efficiently store and quickly release energy. Alexander et al. cite that tendons are 93 percent efficient at storing and releasing energy. Mechanical springs are even more efficient than tendons: approximately 95 percent[48]. Some have suggested using electrical motors to regenerate mechanical energy, but efficiencies for converting mechanical work into chemical potential energy in a battery are low, as many energy transfers are required. First the efficiency of a dc motor is in the range of 70 to 95 percent for high quality motors, at ideal speeds. Second, there must be a charging circiut which may require a dc/dc converter to increase the voltage to an acceptable value. There will be some heat loss involved in this process. Thirdly, the battery must be charged which also will involve heat loss. To make matters worse, more energy will be lost as heat when the stored chemical energy is converted back to mechanical work. Considering that the overall efficiency of the process is the product of the efficiencies of each step, it is likely that many times more energy will be lost to heat during this process as compared to using a mechanical spring. This suggests that using a

mechanical spring in a legged robot may be a much more effective energy regeneration strategy than using the motor itself.

2.4.1 Mechanical Parameter Tuning in Animals.

Animals including humans have many muscular-skeletal attributes that have been tuned to improve gait and stance stability. Wagner et. al show analytically that the negative slope of the muscle force-velocity contributes to stability in human walking [71, 42].

2.5 Compliance in Robotic Limbs

Adding compliance to robotic limbs has been shown to be a useful and energy efficient approach. Compliance may be added in parallel or in series. Springs in series with an actuator are called Series Elastic Actuators and has been used extensively in legged robots including Atrias and BigDog. This technology has also been used in wearable robotics. Dr. Thomas Sugar at Arizona State University has used the technology in many powered prosthetic and orthotic systems.

2.5.1 Robotic Tendon

In 2006 Sugar and Hollander published a paper on their development of a device they called the robotic tendon. The device is an actuator for a foot orthotic, but would later be developed into a high performance robotic prosthetic foot. The paper details their method for optimizing the stiffness of the spring to reduce motor power,

particularly peak motor power. The key metric for selecting a motor, in this case, was peak motor power. Reducing the peak power requirement meant reducing the weight of the motor. This allowed them to select a motor that was seven times lighter than the motor required by a rigid transmission [30].

2.5.2 SPARKy

Sugar et al continued their work in human worn robotics, developing SPARKy: a powered prosthetic ankle with tuned series elastic actuation. The team uses a compliant series elastic actuation to store braking energy, during one portion of gait, and return it powerfully during another. This technology allows the group to create lightweight devices with high power density. This is due to the fact that the spring, once loaded, is capable of producing large forces even during high velocity decompression. Energy savings come because prime conditions exist at the ankle during gait to take advantage of the passive spring element (and not coincidentally so, since the human ankle is also actuated by a biological series elastic actuator). The force velocity curve of the ankle during walking has a long, slow braking period as the mass of the body rolls over the joint during stance. This is followed by a powerful release of energy at toe off. During the braking portion SPARKy's robotic tendon is stretched. Some of the elastic energy comes from the kinetic energy of the body as the ankle damps the falling motion of the mass. At the same time, however, SPARKy's motor takes advantage of the slow loading period to store even more energy into the spring. Then, as the ankle joint's velocity switches direction, the spring and motor tug on SPARKy's heel, forcing the toe into the ground and pumping energy into the human's gait [24, 26].

2.5.3 Other Wearable Robots

The author of this document worked, as an undergraduate, with Sugar and his former students developing wearable robots intended for human assistance and augmentation. These devices extended the concept of the robotic tendon. Several devices were created including JTAR, the Joint torque augmenting robot, based closely off of the original robotic tendon [29].

Another device created by the team had a very different approach. The Orthotic Load Assistance Device is a robot was designed to transfer weight of a soldiers pack directly to the ground, bypassing the human's legs entirely[31]. This robot employed a variable stiffness actuator which allowed dynamic changes in both position and stiffness. The actuator, dubbed the Jack Spring, was another novel contribution of Sugar et al. It uses the coils of a spring as a lead screw giving it many desirable attributes [28, 27, 3].

2.6 Passive Dynamic Walkers

With his 1990 paper, Tad McGeer introduced a simple, but effective perspective on walking. Passive dynamic walkers are machines which are capable of stable planar bipedal motion powered only by gravity. These walkers have tuned dynamics which allow them to walk down a small incline. The idea is that when one semicircular foot is in contact with the ground it forms the equivalent of the rim of a wheel rolling on the ground. The leg makes the spoke. Meanwhile, the other leg is swinging as an undamped pendulum about the hip joint (see Figure 15). As the the swing foot reaches its maximum forward angle, the the body rolls off of the toe of the stance foot

and onto the heel of the swing foot. The roles of the feet swap and the cycle repeats itself. In this way, the legs and feet become the spokes and rim of a revolving wagon wheel [43]. In a second paper McGeer looks at the effect of adding knees to the legs to ensure ground clearance during swing (see Figure 16).

He starts with analytic equations of motion for the legs. The input variables of these equations are the initial values of the leg angle and rotational velocities of the leg segments. He integrates these over the entire step and looks for limit cycles with initial conditions which reproduce themselves at the end of the cycle, making a stable step. He does this by minimizing a difference equation from start to end of the step. Once a stable set of initial conditions are found, the stability of the limit cycle can be tested by finding the eigenvalues of the matrix from the step-to-step difference equation. This stability analysis will only be valid for small perturbations. For large perturbations the author integrates the perturbed equations of motion.

These papers have been cited thousands of times as other researchers have used the concepts of passive dynamic walkers and extended them to extreme efficiency powered walkers systems including, notably, the Cornell Ranger (see Section 2.1.3).

2.7 Optimal Motor and Gearbox Selection

The problem of selecting a motor and gearbox for an actuator is not a new one. Much is published in literature on the subject.

2.7.1 The Classical Method: Inertia Matching for a single DOF System

A classic approach, introduced in 1984 by Pasch and Seering, finds the gear ratio which maximizes output acceleration for a given amplitude of actuator effort[49]. The system in this paper has only a one degree of freedom. It consists of a dc motor, a ball screw, and a moving mass. The load is entirely inertial. The load on the motor is made up of two parts. One part is from the angular inertia of the motor shaft. The other is caused by the mass accelerating linearly along the screw. The equation for motor load follows:

$$T = \left(\frac{J}{r} + Mr\right)a \tag{2.4}$$

Where T is the motor torque, J is the rotational inertia, M is the mass, r is the gear ratio and a is the acceleration. The authors found an analytic solution to the optimization problem:

$$r' = \sqrt{\frac{J}{M}}, \tag{2.5}$$

where r' is the optimal gear ratio. This result gives what the authors call an "impedance matched system". Meaning the rotational and linear inertial loads match when r' is substituted into equation 2.4. The authors also show that r' is a good system transmission ratio from the perspective of minimizing power losses as well. For a given

velocity and acceleration, r' minimizes ohmic heating in the motor windings which is a key contributing factor to motor output power limitations.

2.7.2 Inertia Matching Principle for Multi-DOF Robotic Arms

Chen and Tsai expand the inertia match principle to multi-actuator robotic devices with a series of two papers [13, 12]. The authors use a fitness metric, the maximum acceleration capacity. This metric represents the spatial acceleration capacity of a multi degree of freedom geared robotic mechanism. It is proportional to the volume of the acceleration ellipsoid. The authors define the generalized principle of inertia match for multi degree of freedom, geared robotic systems. The principle states that the "mass inertia matrix of the input links reflected at the joint-space is equal to that of the major linksKiTy SP.

In the second paper the authors develop a method to both maximize the acceleration condition and produce kinematic isotropy at the end effector. To explain this further, the generalized velocity ratio must be defined. The generalized velocity ratio K_v is similar to a gear ratio in a single degree of freedom system. For a given end effector velocity vector $\dot{X}$, the corresponding actuator velocity vector $\dot{\Phi}$ required to produce $\dot{X}$ may be calculated via the well-known Jacobian matrix, or vice versa. The ratio of the magnitudes of these two vectors represents how much the leg magnifies the actuator velocities in that direction. This ratio is the generalized velocity ratio:

$$K_v = \frac{|\dot{X}|}{|\dot{Phi}|} \tag{2.6}$$

In general K_v varies with the direction of $\dot{X}$, some directions amplifying input velocities more than others. When the end effector is in a state of kinematic isotropy, K_v does not vary with regard to direction. This means that actuator velocity is amplified equally in every direction.

2.7.3 Optimal Design with Motor Selection

Van de Straete et al published a series of papers describing a method to select a feasible motor given a dynamic loading scenario. They first normalize the problem using the inertia of the motor. This means that many motors may be plotted meaningfully on a single graph by their normalized maximum torque and velocities. Then they develop a function they call the MLB load curve (standing for Maximum Lower Bound). The curve is generated based on the force and velocity requirement of the loading scenario. Any motor that falls under the MLB line is infeasible, it will be unable to perform the task regardless of transmission ratio. This gives a fast visual method for narrowing the design selection [69]. In a subsequent paper the group then proposes a two step process whereby feasible motors are selected and then an optimal motor/gear ratio combination is chosen which reduces the peak current. The reasoning behind using peak current as the optimization criterion is to reduce the inverter cost [68]. Finally the group looks into the benefit of adding a variable transmission and calculates a smooth motor trajectory with an optimal transmission ratio [67].

Another group that uses a very similar inertia normalized graphical method is Cusimano et al [14, 16, 15]. One notable difference between the two groups is that Cusimano makes his plot on a graph with normalized power as the axes (normalized kinetic energy on the horizontal and normalized acceleration power on the vertical).

Van de Strate uses normalized torque on the vertical axis and normalized velocity on the horizontal. Both normalize using the square root of motor inertia.

2.7.4 Motor Selection and Gear Ratio Optimization with Fully Dynamic Load

A recent paper Rezazadeh and Hurst describe their methods of selecting a motor and a corresponding optimal gear ratio [57]. Since the time of Pasch and Seering, dramatic decreases in computation price have made model simplifications more and more unnecessary. It seems fitting then that Rezazadeh et al use a full dynamic load, doing away with the commonly used assumption of a purely inertial load. They also include damping effects in their motor model which have previously been ignored. In their method they calculate the total energy required to perform a task by integrating the loading terms over a cycle of the gait data to which they are optimizing their actuator. The minimum of this function is found in the usual way: by setting the derivative of the energy function to zero. The equation that results is a general case, and the classical results (presented above) may be reproduced by plugging in the associated simplifications (such as pure inertial loading). In addition to this result, the authors present a optimal gear ratio for maximizing actuator bandwidth, which differs from the energy reduction result.

Next the authors present their method of motor selection. They plot Pareto surfaces of each of the candidate motors. The horizontal axis is motor bandwidth and the vertical axis is motor speed. They create a Pareto curve for each motor. Every point on the curves represents the power use and bandwidth of a given gear ratio. The best motors will have low energy use and high bandwidth, so motors will be selected with Pareto curves that cross the lower left most portion of the graph.

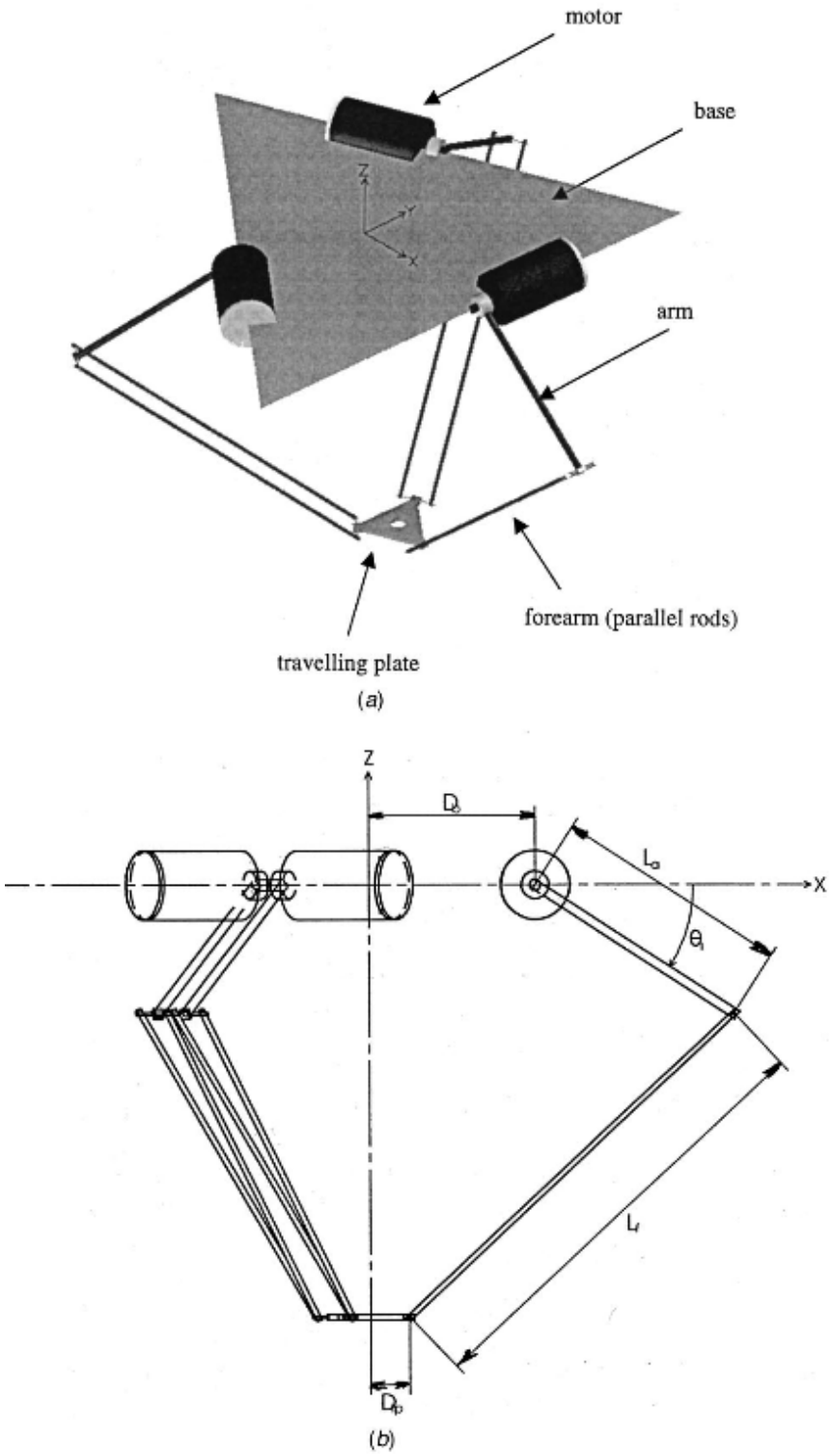

Figure 10. Delta Robot, a 3 degree of freedom parallel robot: (a) layout of the robot (b) dimensional parameters of the robot [46].

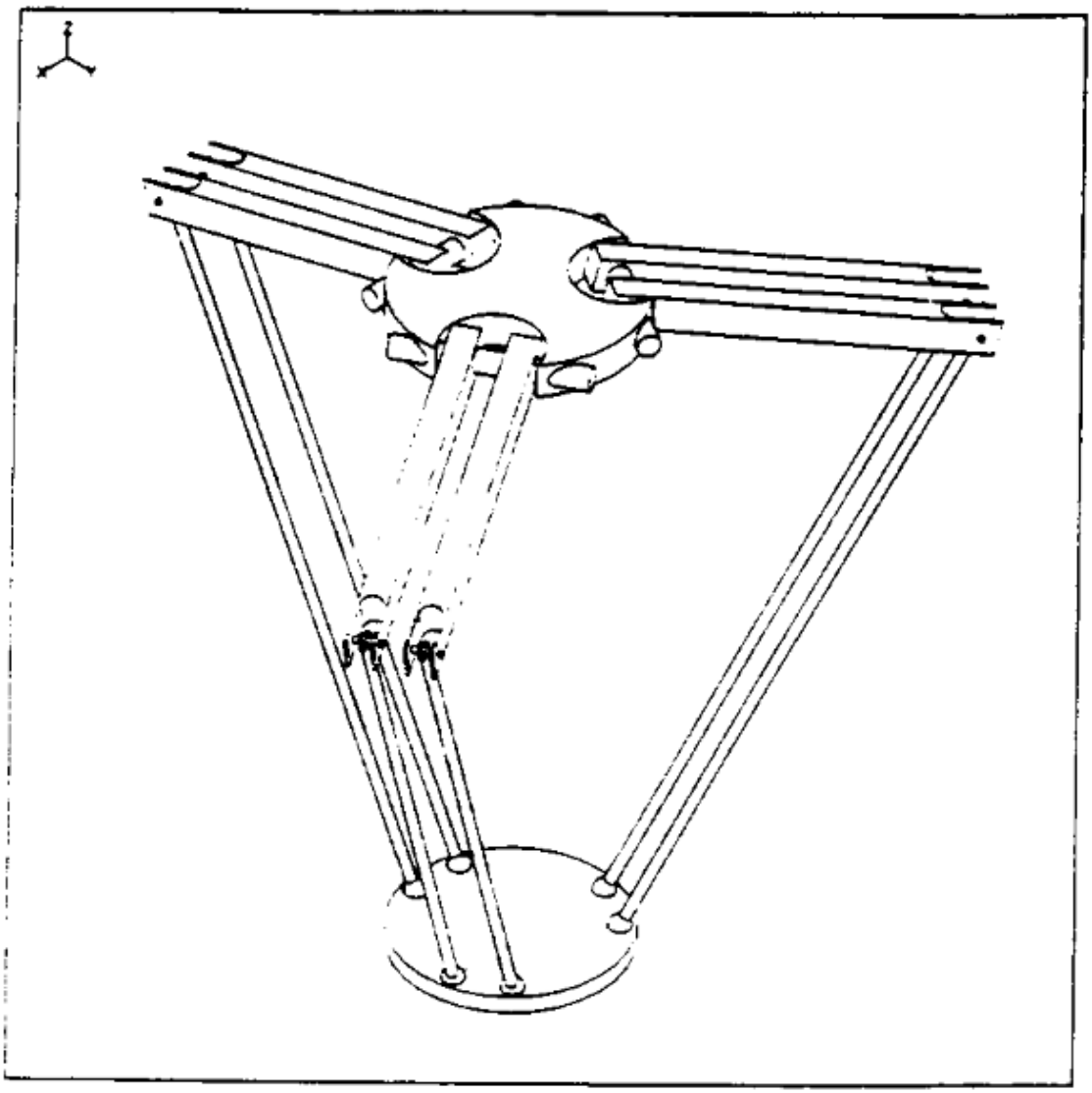

Figure 11. HEXA: a 6 DOF robot fully parallel robot[52]

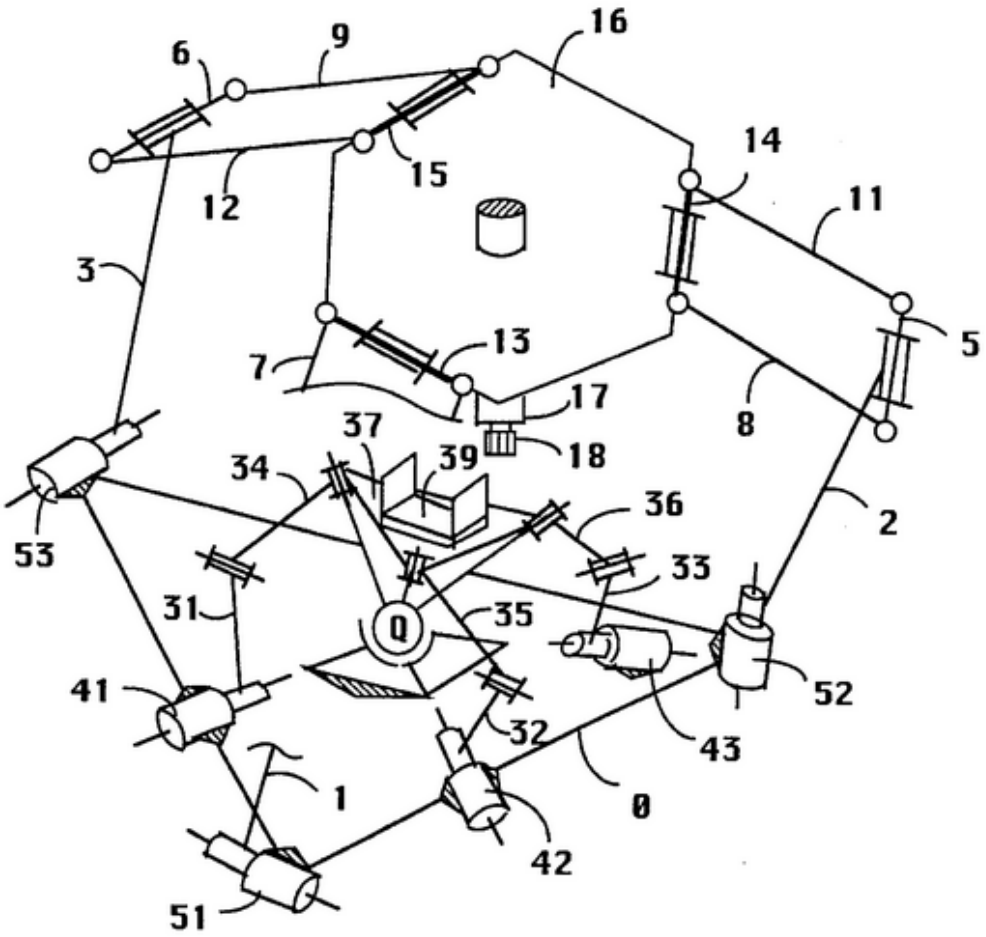

Figure 12. A Delta robot combined with a 3 DOF off-board wrist [66].

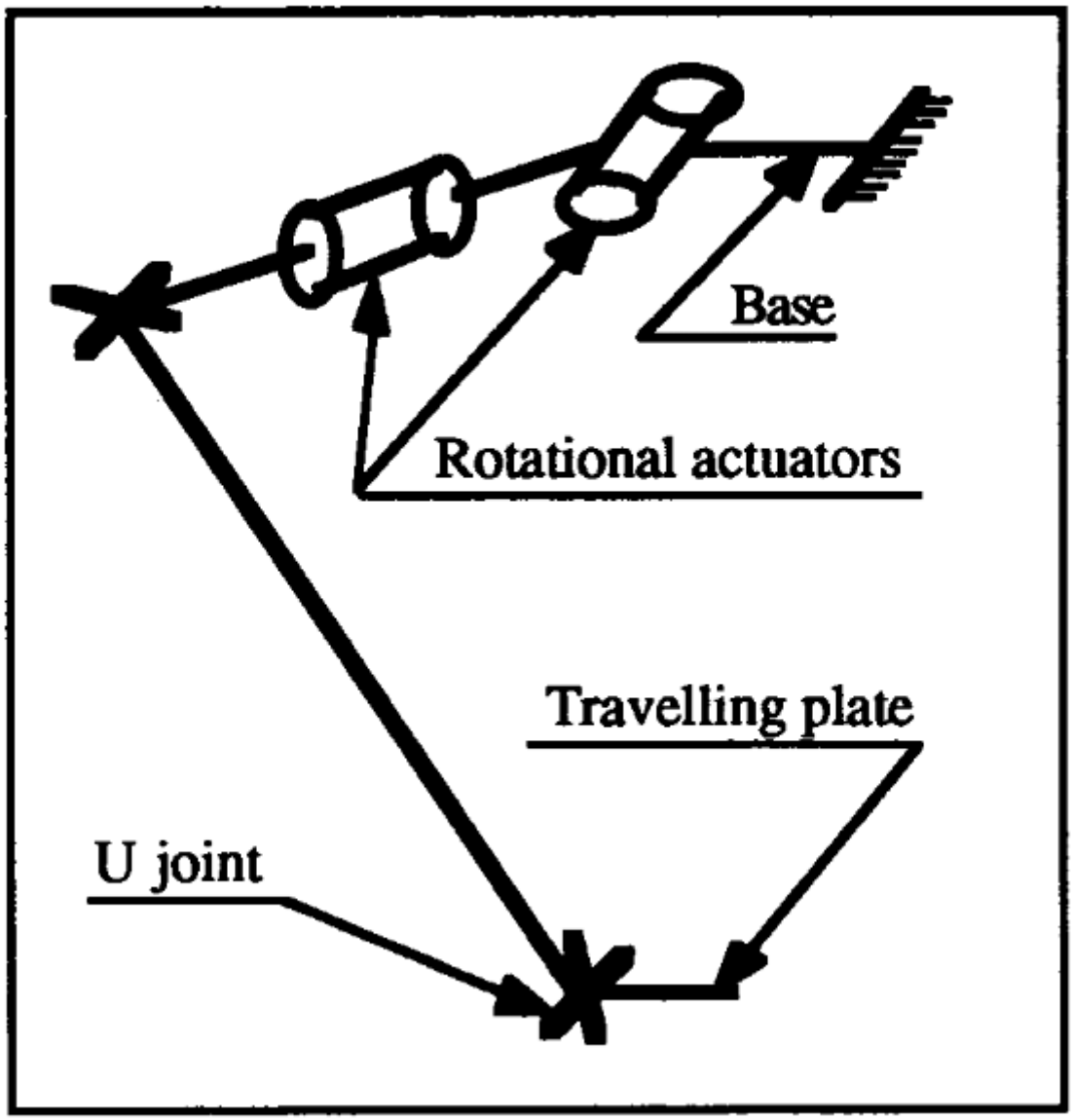

Figure 13. Delta-6: A Delta variant with three rotations added [52].

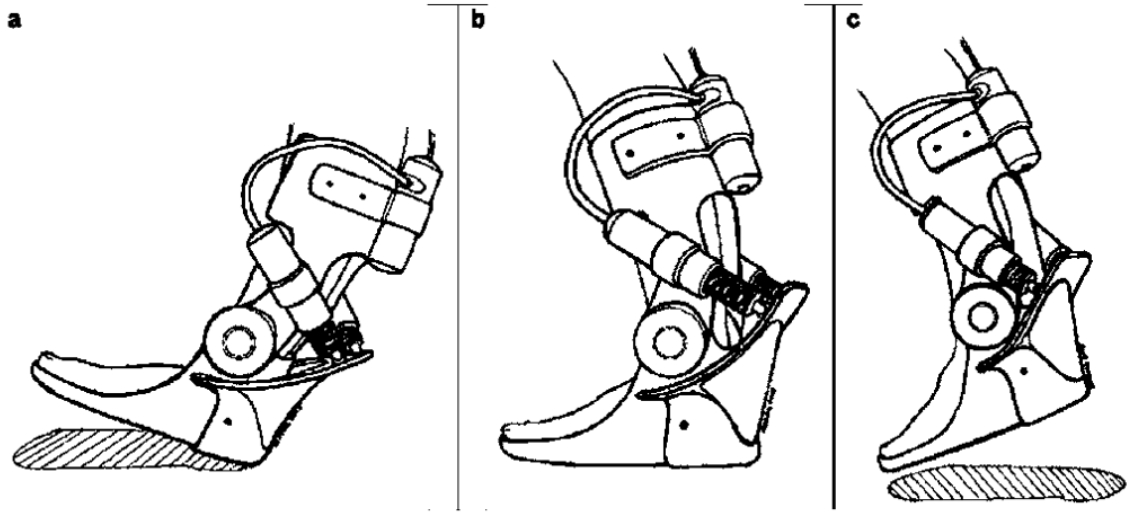

Figure 14. The Robotic Tendon: a tuned series elastic actuator by Sugar et. al.

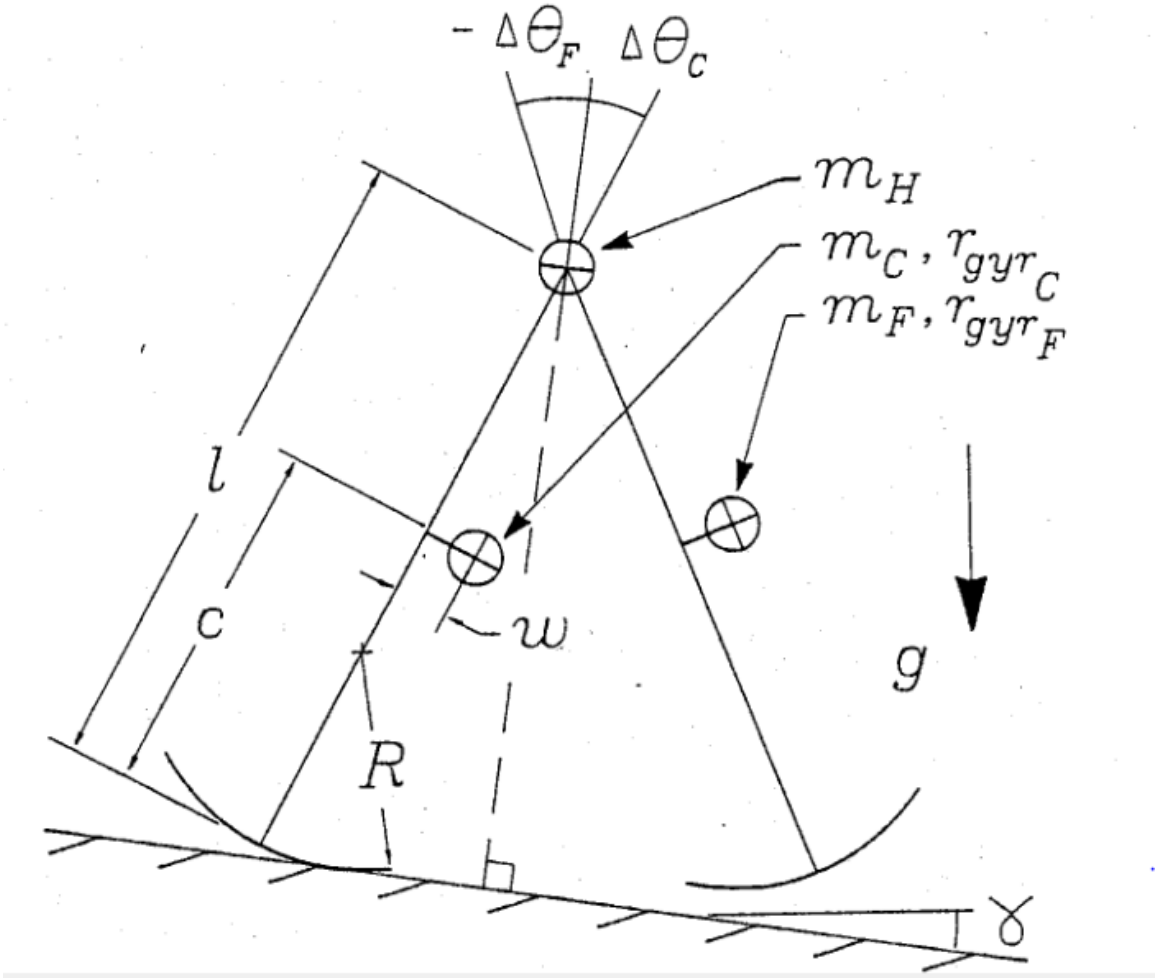

Figure 15. Sketch of passive dynamic walker. [43]

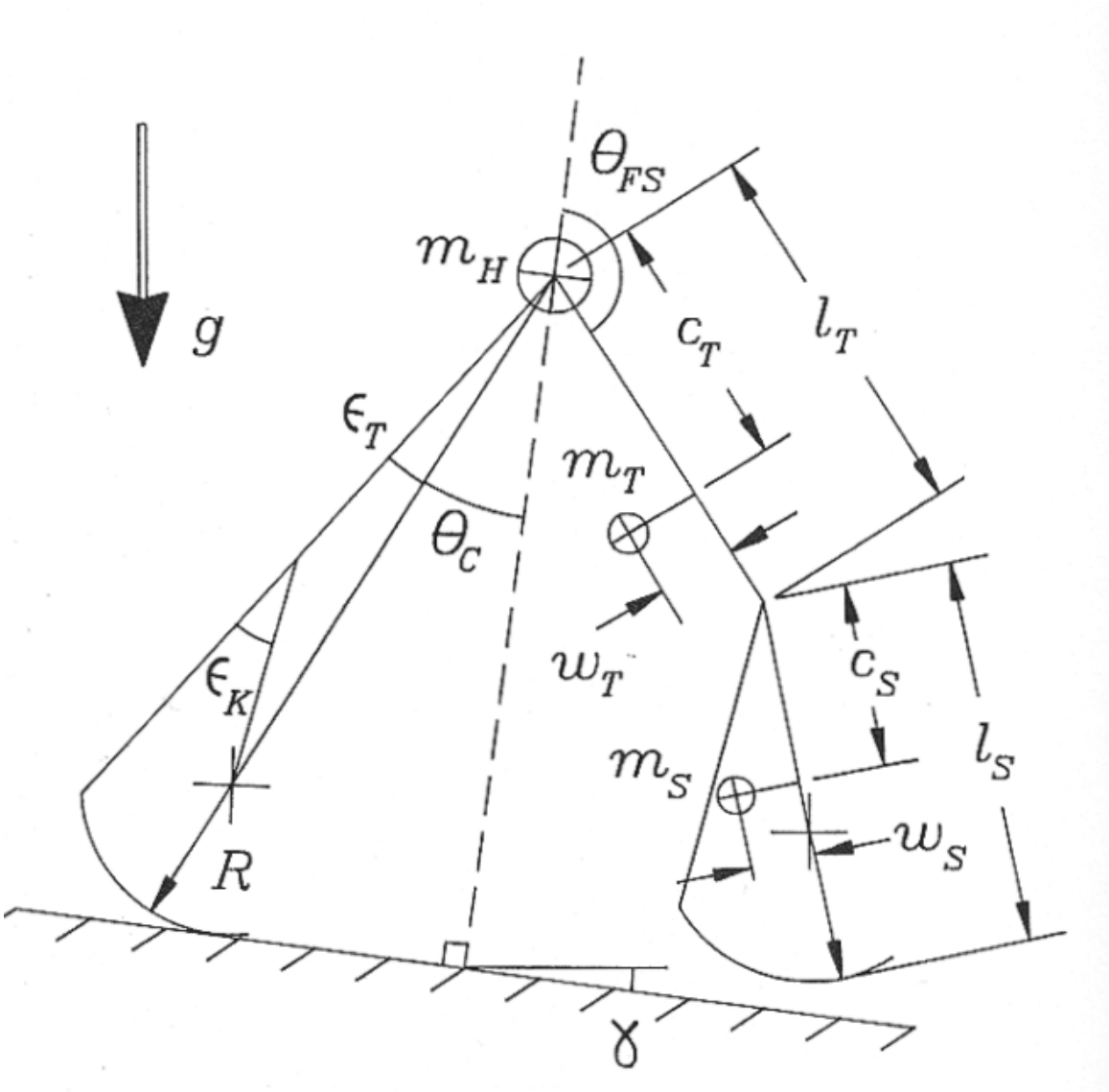

Figure 16. Sketch of passive dynamic walker with knees. [44]

Chapter 3

DESIGN OF THE KITY SP ROBOTIC LEG

3.1 Introduction

This chapter introduces the Kinematically Tunable Hybrid Serial-Parallel (KiTy SP) robotic leg. This leg structure is unique from other robotic limbs, among other things, because it employs a hybrid serial-parallel structure. Many of the current state of the art legged robot platforms use a biologically-inspired serial limb structure. BigDog (Section 2.1.1), the MIT Cheetah (Section 2.1.4), HyQ (Section 2.1.5), the Cornell Ranger (see Section 2.1.3) and others utilize a simple, serial leg structure. Serial limb structures have many benefits for legged walking. One such benefit is, using a straight-leg configuration, a serial structure supports vertical forces with little or no required joint torque. This allows animals and robots to stand with very little expended energy. Another benefit of the serial leg structure is that it is able to produce large end effector velocities parallel to the ground, a beneficial feature in legged gait. Finally serial legs have a relatively large work-space which is a necessary attribute in a legged robot.

In spite of the benefits of the biologically-inspired serial structure, many researchers have created new structures for robotic limbs. The Atrias system, for example, has a unique pantagraph mechanism embedded in its leg structure (See Section 2.1.2). The designers modeled the leg to emulate the Spring Loaded Inverted Pendulum (SLIP) model. Atrias has shown the capability of traversing uneven terrain while maintaining a relatively long battery life.

Other areas of robotics, such as industrial robotics, have enjoyed diverse approaches in limb structure. Parallel robots in particular have seen particular success in many applications. One of the most important of these robots is the Delta robot (see Section 2.2. The Delta robot is a parallel manipulator commonly used for applications such

as “pick and place” and food packaging. This is due, in part, to the fact that parallel manipulators are generally faster and more precise than their serial counterparts, because of their superior rigidity and smaller mobile mass [64]. The KiTy SP leg structure attempts to combine the benefits of serial and parallel manipulation.

This chapter will detail the design process of the KiTy SP robotic leg. First the leg structure will be detailed showing how the admittedly complex leg structure functions. Then the solution to the kinematic equations will be presented, and finally, as a proof of concept, a prototype of the leg will be presented along with the results of a static weight test.

3.2 Development Process

This section will cover all of the many steps of the development process for this leg. This process includes choosing a specific actuation approach, mathematically modeling the leg by solving the kinematic equations and finding the Jacobian of the leg, and finally, designing and building a physical prototype. The first of these steps is among the most important.

3.2.1 Actuating in Parallel

Choosing the structure and actuation method of a robotic limb is a part of the process that should never be underestimated. The general structure of a limb defines and inherently limits the design space of the project. We chose to start with a biologically inspired three segment leg with a hip, knee and ankle joint. Starting with a biological leg structure seemed like a good choice because the robot is intended to

perform many of the same tasks as biological animals. As stated above, animals are still significantly better at efficient navigation of rugged terrain than robots presently are. This does not, however, preclude the possibility that there exists some structure that is better suited for the task, or for the differing technology employed by robots (eg. mechanical actuators instead of muscles and tendons). A simplification has been made from the biological animal's leg which is a reduction of the degrees of freedom of the leg. Animals such as cats, dogs and horses typically have a 3 DOF hip joint, but the hip joint in this leg will only have only 2 DOF. These animals also have extra degrees of freedom in their foot, but this robot will have none. The knee joint and ankle joint of this robot are both 1 degree of freedom, which is similar to the animal leg structure (see Figure 18). The hope here is that most of the benefit of the 3 jointed structure will be retained without the extra complication of more jonits to control. Figure 17 is an annotated sketch of a side view of a simplified version of the leg. It also includes the angles used to define the position of the leg. This figure only shows a planar simplification of the system and does not show hip abduction/adduction. Figure 18 shows the spatial leg mechanism, but a planar version of the mechanism is evaluated first.

Many other robots have similar leg structures to the structure used here. These include Boston Dynamic's BigDog robot[54], MIT's Cheetah robot[61], and IIT's HyQ robot [8]. These robots, however, use series actuation methods. This means, in most cases, that the heavy actuators are fixed to the leg segments which they actuate. This adds to the inertial losses in the system. In addition this actuation method limits the number of parameters which can be tuned in an attempt to optimize the system. This is the reason that a more complex actuation approach was chosen for this robot: parallel actuation. Parallel actuation reduces inertial loads and introduces

many new parameters which can be optimized to increase the performance of the leg. We actuate the joints of the leg with motor driven two-link RSS mechanisms which are attached between the body and the corresponding leg segment. The R joints of these linkages are fixed to the body through small motors which are used to control the leg. The general actuation method is shown in Figure 17. This drawing is a planar simplification of the robot, which allows easier visualization of the effect of each motor. The red and blue arrows marked τ_1 and τ_2 represent torques applied by each motor. The dotted red and blue arrows represent the torque that each joint would have to produce to provide a force at the end effector equivalent to what would be produced by τ_1 and τ_2. A strength of this approach is that the geometry of the mechanism can significantly change each motor's contribution to the task. This allows the load to be spread across the motor's more evenly than in the serial case. Adding the second degree of freedom at the hip (see Figure18) allows the end effector of the robot (in this case, its foot) to move out of the plane. This creates a potential for a significant improvement in maneuverability for the final robot. It also adds another degree of complexity and with it the opportunity to specialize the limb even further. Since this will be an actuated degree of freedom, it comes with a new actuator to add to the system. Unlike the serial case, this actuator will not need to be connected directly across the joint. It will be placed in a more advantageous way that allow it to work together with all of the other leg motors.

3.2.2 Passive Spring Clutch Ankle Mechanism

As stated above, there are 4 degrees of freedom total on this leg, but only three of them will be actuated. This choice was made because three actuators are the minimum

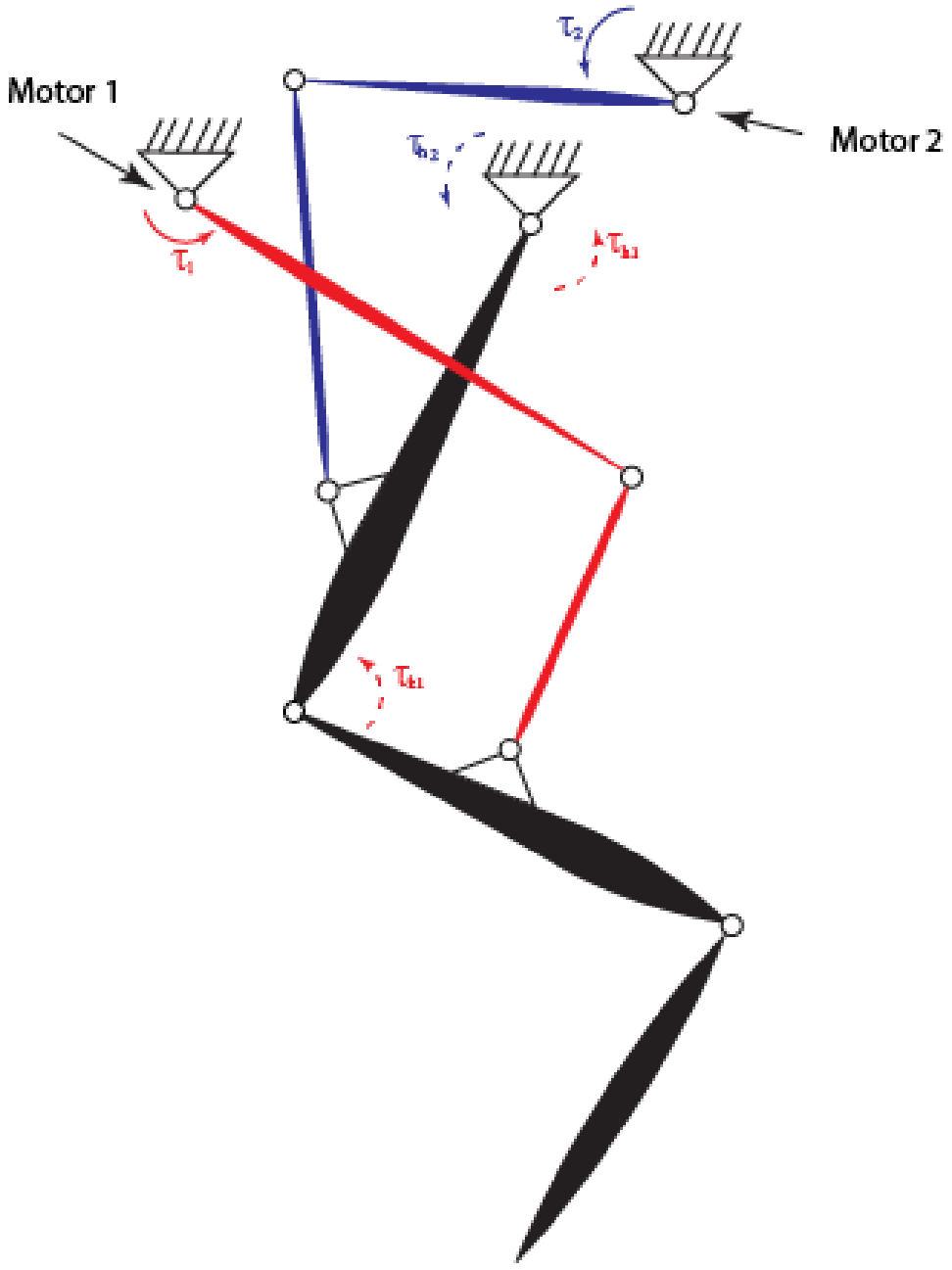

Figure 17. Annotated Drawing of the Planar Leg Structure with Parallel Actuators.

one needs to be able to control the end effector position fully in 3 dimensional space. This simplifies the system, creating unique solutions to the inverse kinematic solutions for example (see following sections). The problem this presents, however, is: how can

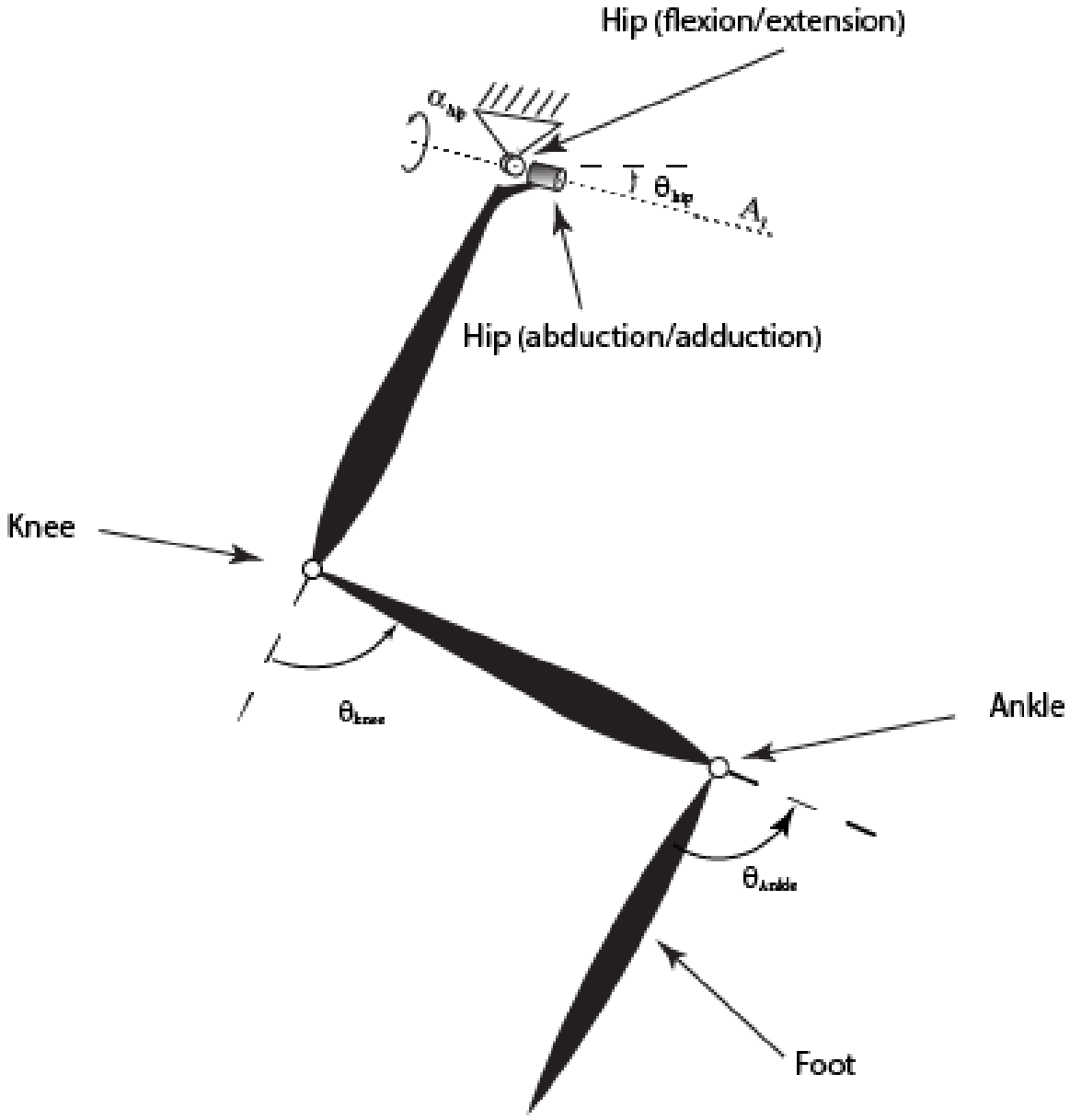

Figure 18. Annotated Drawing of the Serial Leg Structure with All Joints Included.

the ankle joint be controlled. A clever spring clutch mechanism was created to solve this problem(see Figure 19) The spring clutch mechanism allows the ankle to switch between a spring joint during stance (when load is applied to the foot, causing it to rotate clockwise) and a passively controlled four bar parallelogram mechanism during

swing. The mechanism was designed around digitized high speed data of a cheetah running. It approximates the way the cheetah lifts its foot during swing. When force is applied to the foot, however, an elastic member which plays a similar role to a human's Achilles tendon is engaged and makes the ankle a passive spring joint. This use of mechanism design allows a simple way to control the whole leg with only three motors.

3.2.3 Choosing a Configuration

Choosing a serial actuation approach posed an unexpectedly important question: which leg segment should each actuator be attached to? There are several possible configurations. These configurations are listed in Table 1. These various configurations will be outlined in the following text.

Table 1. List of All Possible Actuator Configurations

Configuration	Thigh	Shank
One	2 Actuators	1 Actuator
Two	1 Actuator	2 Actuators
Three	0 Actuators	3 Actuators

The authors did not want to create an over actuated system, as this adds unnecessary complication to the system at this stage of the project. For example, one can choose to connect zero, one, or two actuators to the thigh but not all three. This is because the hip joint only allows two DOF: flexion/extension and abduction/adduction. If three actuators are connected to this segment one of them will be dependent on the position of the other two. Additionally, the knee joint would be entirely unactuated

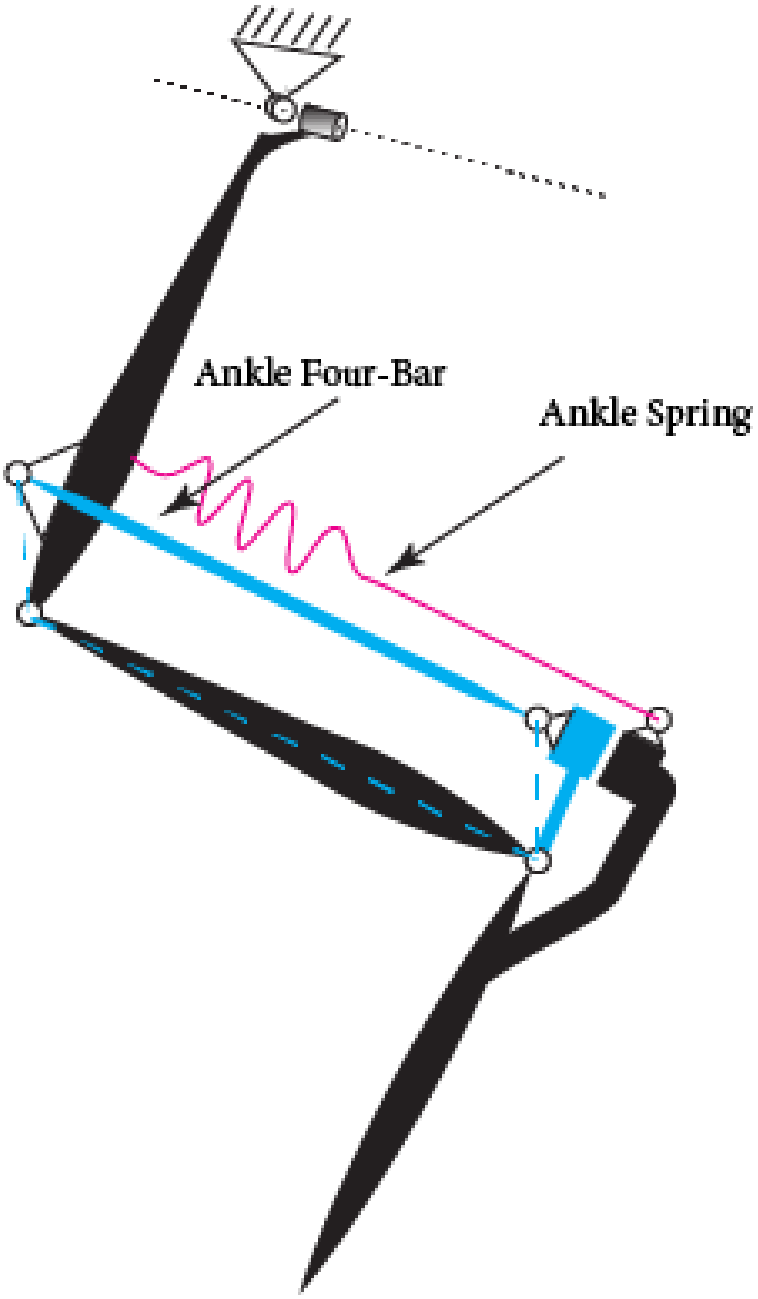

Figure 19. Annotated Sketch of the Ankle Mechanism.

(Or a fourth motor could be added to the system to actuate the knee, but this would create an over-actuated system: 3 degrees of freedom and 4 actuators).

In Configuration 1 (Shown in Figure 20), two actuators are connected to the thigh

of the robot and one is connected the shank. Table 2 describes how each motor affects the leg joints in this configuration, and Figure 6 is an annotated drawing of the configuration.

Table 2. Detailed Breakdown of Configuration One

Leg Joint	Actuator 1 Thigh	Actuator 2 Thigh	Actuator 3 Shank
Hip Ad/ Ab	Y	Y	Y
Hip Flex/ Ext	Y	Y	Y
Knee Flex/ Ext	N	Y	N
Ankle Flex/ Ext	N	Y	N

In Configuration 2 (Figure 21), one actuator is connected to the thigh and two actuators are connected to the shank. Table 3 describes how each motor affects the leg joints in this configuration, and Figure 21 is an annotated drawing of the configuration.

Table 3. Detailed Breakdown of Configuration Two

Leg Joint	Actuator 1 Thigh	Actuator 2 Shank	Actuator 3 Shank
Hip Ad/ Ab	Y	Y	Y
Hip Flex/ Ext	Y	Y	Y
Knee Flex/ Ext	N	Y	Y
Ankle Flex/ Ext	N	N	N

After some analysis Configuration 2 was shown to be significantly better than Configuration 1. It allowed more control of important parameters (such as the conditioning number of the Jacobian) which resulted in a vastly greater usable workspace.

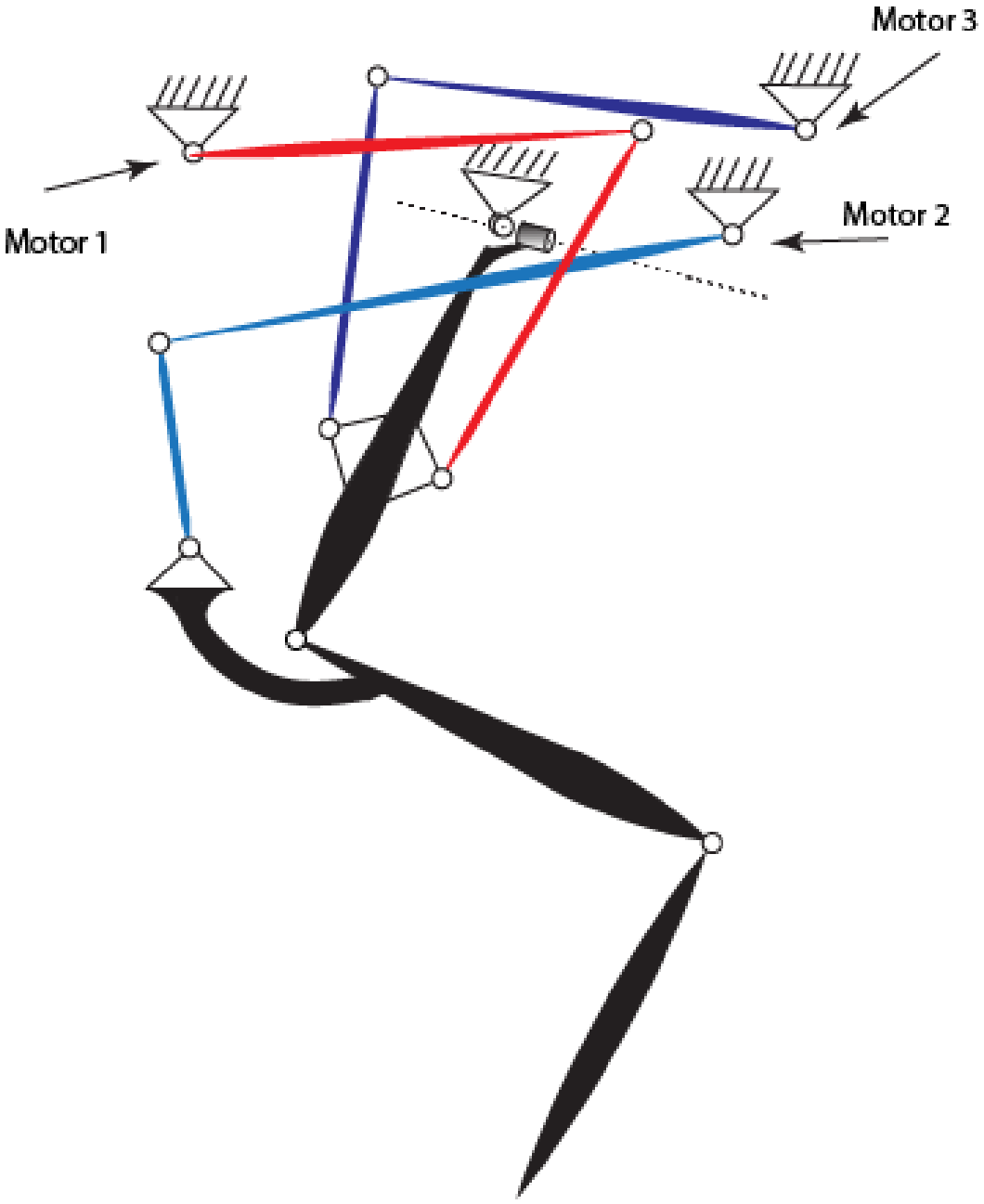

Figure 20. Annotated Sketch of Configuration 1 in three dimensions.

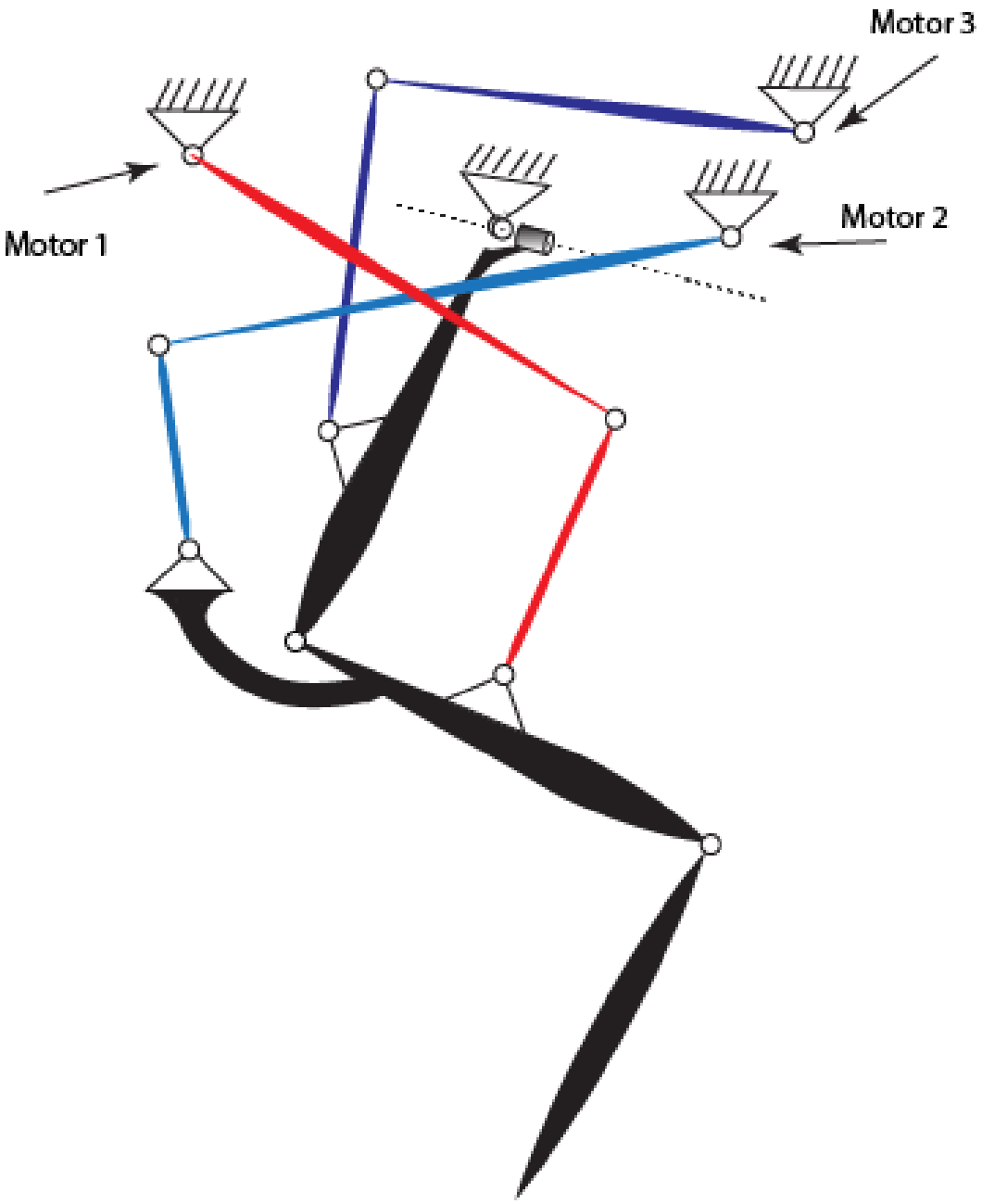

Figure 21. Annotated Sketch of Configuration 2 in three dimensions.

3.2.4 Serial Kinematics

Another engineering problem faced in the design and implementation process, was the selection of the geometric properties of the linkages. The kinematics of the leg

mechanism would have to be solved to gain a scientific understanding of the problem. In fully serial robot limbs the forward kinematics are commonly used and tend to be a simple problem to solve. The inverse kinematics for a serial mechanism are not as simple. However, in parallel robots the inverse kinematics problem tends to be simpler, and the forward kinematics more complex. Our design is a combination of the two; therefore, challenges arise when solving either forward or inverse kinematics. In the end the inverse kinematic equations were solved. A detailed description of the solution to the inverse kinematic equations is given. The first step in this method of solving the equations for this robot was to break the problem up into two parts: the serial portion of the leg and the parallel portion of the leg. Serial Kinematics. The serial portion of the robot limb is shown in Figure 8. Notice that the foot is left out of this diagram. This greatly simplifies the kinematic solution. In fact if the foot and ankle angle are left in the kinematics, there are infinite solutions to the inverse kinematic equations. This also would make the Jacobian matrix non-invertible. Recall that for this system, the ankle is controlled by a spring during stance and a mechanism during swing. For stance this means that with a desired force output the ankle joint will need to be at some known angle. For swing, the position of the foot will be defined by the thigh and shank leg segments. In both cases controlling the ankle joint's position will be enough to fully define the leg. Figure 8 shows the thigh and the shank, along with all of the angles needed to fully define their position in space.

From Figure 22it can be seen how the two degree of freedom hip connects the thigh to the body. The first rotation θ_{Hip} represents hip flexion and extension, the second rotation, α_{Hip}, represents hip abduction and adduction. The third rotation is θ_{Knee}, which represents knee flexion and extension. The goal is to determine these angles given a three dimensional ankle joint position and the geometric properties of

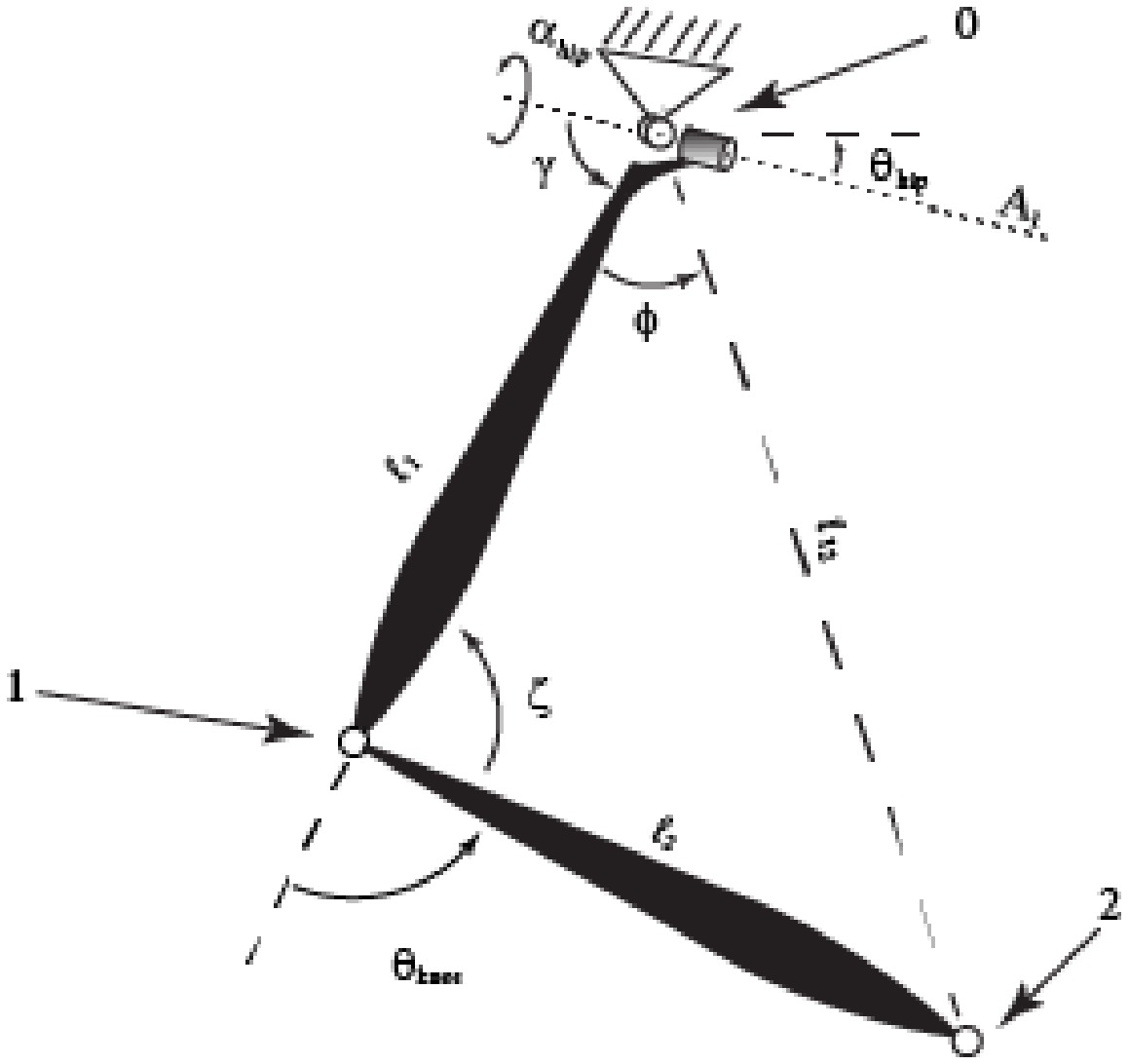

Figure 22. Serial Limb Annotated Sketch

the leg. The first step is to find the hip and knee angles. We know that the axis $\overline{A_1}$ lies in the plane that contains the thigh and the shank sections of the leg. We can

define axis one as a function of θ_{Hip}:

$$\overline{A_1} = \begin{bmatrix} cos(\theta_{Hip}) \\ -sin(theta_{Hip}) \\ 0 \end{bmatrix}. \tag{3.1}$$

The dot product of $\overline{A_1}$ and the vector between the hip and the ankle $\overline{O_2}$ is defined:

$$\overline{O_2} \cdot \overline{A_1} = \|\overline{O_2}\|\|\overline{A_1}\|cos(\gamma + \phi) \tag{3.2}$$

Note that γ and ϕ are angles shown in Figure 8. The angle γ is a constant geometric property of the leg and ϕ can be easily found using the Law of Cosines since the three lengths of the triangle that contain it are already known. Since $\overline{A_1}$ is a function of θ_{Hip}, this equation can be solved symbolically for θ_{Hip} (using symbolic math software such as Matlab). Finding θ_{Knee} is just a matter of solving the triangle formed by points 0, 1, and 2 using law of cosines to find the inner angle and subtracting that from pi to determine θ_{Knee}. The angle α_{hip} is the arctangent of the perpendicular distance from ($vecA_1$ to the ankle divided by the distance of the ankle coordinate in the y direction:

$$\alpha_{Hip} = atan2\left(\frac{-y_2}{x_2 sin(\theta_{Hip}) + z_2 cos(\theta_{Hip}}\right) \tag{3.3}$$

where, x_2,y_2, and z_2, are the x, y, and z coordinates of the ankle joint (joint 2 in Figure 8).

3.2.5 Parallel Kinematics

The next part of solving the inverse kinematics of the leg is solving the parallel portion of the leg. The solution to this portion will be used three times, once for each of the three motor linkages. The problem is defined as follows: given the leg angles (from the preceding section), find the motor angles (θ_m). To understand the solution to this problem, the two link mechanism needs to be defined carefully. It is an "RSS" linkage. The joint fixed to the leg segment is a spherical joint, (joint B in Figures 23, and 24). The link connected to this joint (which is called the connecting rod) constrains joint A_0 to be distance ρ away from joint B. This means for any leg state joint A_0 must exist on a sphere of radius ρ about joint B. This sphere is shown in grey in Figures 23-24. Joint A is the motor which is, in effect, an R-joint. The link connected to the motor (which is called the control arm) constrains A_0 to lie on a circle of radius r, about joint A, on a plane perpendicular to the motor axis U. The important take-away here is that the solution for this problem is the solution to the intersection points of a sphere of known radius about a known point (B), and a circle of known radius about a known point(A) which lies on a known plane.

The first step to solving this problem is defining carefully the location in space of both the sphere and the circle. As stated above the center of the sphere is Joint B which is fixed rigidly to its corresponding leg segment. The exact location of Joint B in space can be found using a rotation matrix method and the leg joint angles found in the serial kinematic portion. To define the circle, the location of Joint A and the

vector U associated with the motor axis are all that is required. The motors are fixed to the body in a known location, which does not change with the leg angles. Therefore the circle about A is fully defined. (See Figure 9)

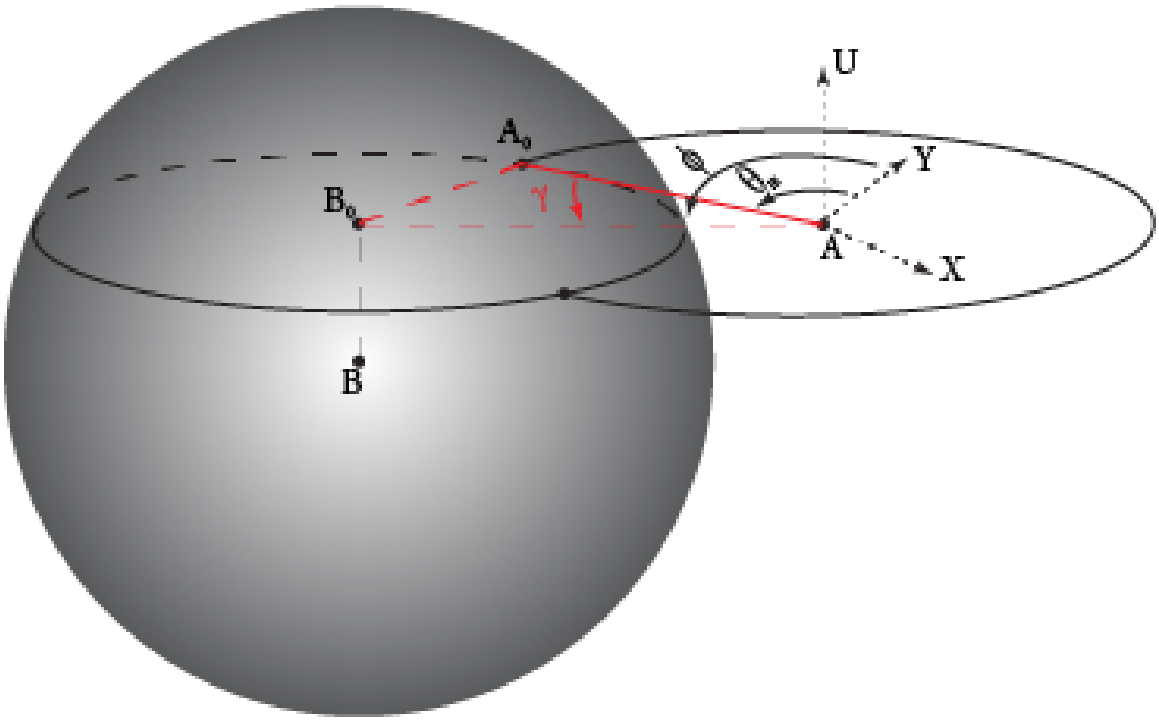

Figure 23. Defining the circle centered at B0. This is the intersection of the plane containing circle A and sphere B.

A_0 is confined to a known circle, and therefore, a known plane. The intersection of a plane and a sphere is a circle or a single point, or nonexistent. The second and third cases correspond with positions of the leg that are singular or outside of the workspace, respectively they will be ignored for this solution. (See Figure 23). If the intersecting circle can be described by known parameters this would simplify this solution from a spatial problem to a planar problem. As it turns out, this intersecting circle can be defined by known geometric information. The center of the intersecting circle (B_0) makes a vector with the center of the sphere $\overline{BB_0}$ This vector is parallel with the vector $\overline{U}$ because $\overline{BB_0}$ can be defined as the projection of vector $\overline{BA}$ onto

vector $\overline{U}$:

$$\overline{BB_0} = \overline{BA} \cdot \overline{U} \tag{3.4}$$

The center (B_0) of the intersecting circle is now known and finding the radius will fully define the circle. This is accomplished by solving for the third length of a right triangle, see Figure 24.

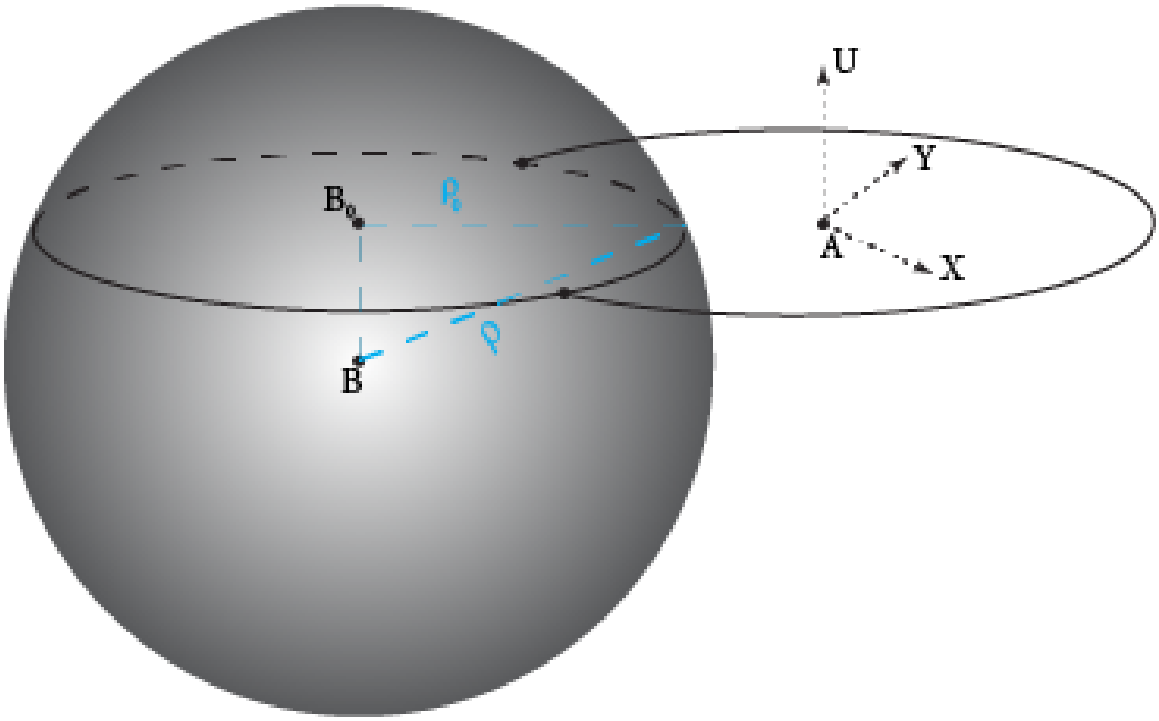

Figure 24. Finding the radius of the intersecting circle at B0.

The equation for the radius follows:

$$\rho_0 = \sqrt{\rho^2 - |(BB_0)|^2} \tag{3.5}$$

where ρ_0 is the radius of the intersecting circle centered at B_0. The goal is to find θ_m. We can write θ_m as a function of γ and ϕ (see Figure 24). The next step towards solving for θ_m involves solving the triangle formed by the points $B0$, A_0, and A. The three lengths of this triangle are known: $|\overline{(B_0 A_0)}|$ is ρ_0, and $|\overline{(A_0\, A)}|$ is the length of the control arm (r), and the last length is the distance between two known points. The inner angle of this triangle γ is necessary to find the motor angle. This angle can be found using law of cosines. Next, the angle ϕ needs to be determined. This is the angle between the vector $(\overline{AB_0})$ and the unit vector Y. This vector Y is an arbitrarily defined vector that represents the direction the control arm would be pointing when θ_m is zero. The angle ϕ can be found using the dot product of the unit vector and the vector $(\overline{AB_0})$. Finally, the equation for θ_m is:

$$\theta_m = \phi \pm \gamma \tag{3.6}$$

There are two solutions to this problem, "elbow in" and the other, "elbow out". As can be seen in figures 20-21, some of the linkages are set in one configuration and others are set in the other.

3.3 Building and Testing the Physical Prototype

A physical prototype has been developed that allows a small validation of the mathematical model of this actuation approach. The process of actualizing this approach presented many difficult challenges during the the construction process. These challenges included avoiding interference between the moving parts of the

3D models, and achieving geometric properties, in three dimensions, to match the mathematical model. It has taken several iterations of design to create a functional prototype that is controllable in 3 dimensions. The method chosen to create the parts for this leg was 3D printing. This method was chosen because it is both inexpensive and fast. Not only this, the plastic materials compatible with 3d printing were strong enough to handle the loads required for this test. The model is driven by 3 Dynamixel RX-24f actuator pack.

3.3.1 Validation

A small validation exercise was performed which shows that the model used to predict motor currents actually matches the physical system. In this exercise the leg holds a known weight in a known position and the motor currents are measured. The mass to be suspended by the leg was chosen to be 500g (see Figure 25) This mass was suspended from the leg such that the reaction force vector from gravity at the end effector would be:

$$F = \begin{bmatrix} 0 \\ 0 \\ -.5 \end{bmatrix} [kg]. \tag{3.7}$$

Using the Jacobian transpose this force can be transformed into motor coordinates:

$$\tau_m = J^T F[n \cdot m]. \tag{3.8}$$

J^T here is the transpose of the leg jacobian evaluated at given leg position. For this experiment, several leg positions were chosen to give a more complete validation of the kinematic and kinetic equations. Table 4 lists the various end effector positions chosen for the experiment.

Since the device does not include a sensor to directly measure the mechanical torque at the motors, motor current will be used to estimate the motor's output torque at each position. This can be accomplished using the motor torque constant:

$$I_m = K_t \, \tau_m \, [A]. \tag{3.9}$$

which can also be written:

$$I_m = K_t \, J^T \, F \, [A]. \tag{3.10}$$

This formula will be used in all of the following estimations for current. The sensors used to measure current for this test are Polulu ±5 Amp pololu current sensors (model ACS714). These have a ±1.5% total output error and 21 mV peak to peak noise.

A photograph of the experiment setup has also been included (see figure 25).

3.3.2 Experimental Results

Figures 26-28 display the results of the experiment. Figure 26 shows a stair step graph of the expected vs measured values, for all three motors, in each of the 14 tested

Table 4. Positions used in Model Validation

Trial number	Coordinates [m]		
	X	Y	Z
1	0.02	0	-0.1206
2	0.01	0	-0.1206
3	0	0	-0.1206
4	0	0.01	-0.1206
5	0	-0.01	-0.1206
6	-0.01	0	-0.1206
7	-0.02	0	-0.1206
8	0.02	0	-0.1106
9	0.01	0	-0.1106
10	0	0	-0.1106
11	0	0.01	-0.1106
12	0	-0.01	-0.1106
13	-0.01	0	-0.1106
14	-0.02	0	-0.1106

leg positions. Figure 27 shows a bar graph of the angular error in the results. Since the this angle comes from the arccosine of the dot product of the expected current vector and the measured current vector. The formula used to calcualate this angle was:

$$\theta_e = acos\left(\frac{I_{expected} \cdot I_{measured}}{|I_{expected}|\,|I_{measured}|}\right) \tag{3.11}$$

Figure 28 shows a bar graph of the magnitude error between the measured and expected current. The formula used to calculate this data follows:

$$M_e = \frac{I_{measured}}{I_{expected}} \tag{3.12}$$

Figure 25. Photograph of validation experiment (position 1 shown)

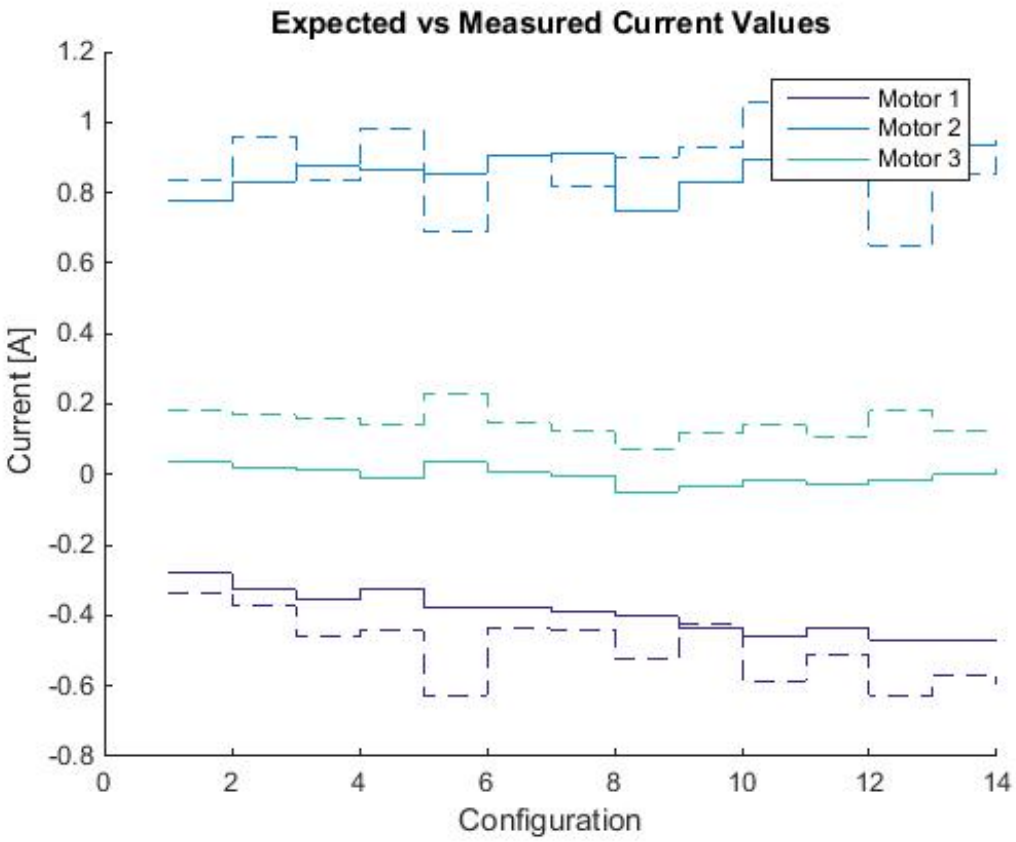

Figure 26. Results of the Validation Test

3.3.3 Conclusions

From these results it can be seen that the prediction was good in terms of direction (see Figure 27), and magnitude (see Figure 28). What error exists likely originates from sensor error, or error in the motor parameters (especially the armature resistance). Importantly, the robot limb is able to be position controlled and to exert force through the end effector on the environment. The next chapter will discuss optimal design methods and how the limb geometry may be tuned to decrease the power required by the motors to perform a given force-velocity trajectory.

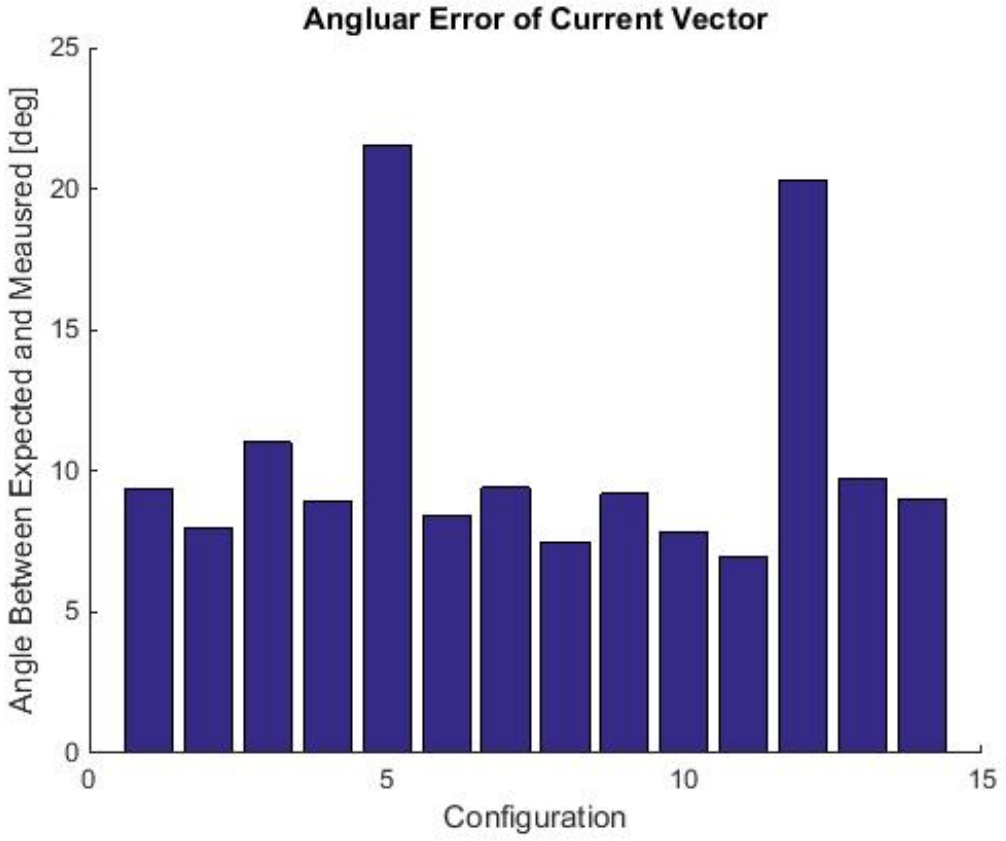

Figure 27. Angle between the expected and measured current values [Degrees]

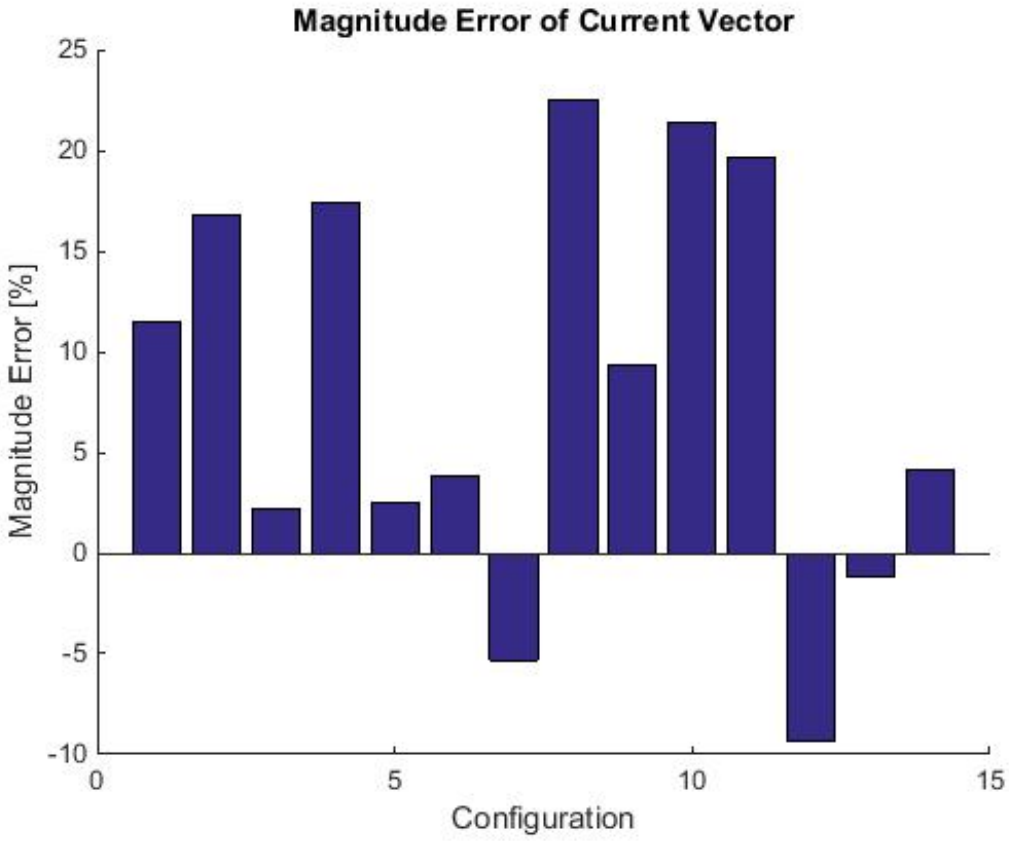

Figure 28. Magnitude difference between the expected and measured current values [Percent]

Chapter 4

SPECIALIZATION OF DESIGN GEOMETRY

4.0.1 Introduction

This chapter describes the optimization method we develop which can be used to design any holonomic robotic limb. This design optimization tool varies geometric properties of the limb structure, finding efficient, task-specific designs. The efficiency is a function of design geometry due to the passive dynamics. This is important because of the lagging progress in power storage and generation technology. Advancements like these may be critical to the successful development of legged robotic technology.

One source of energy waste in legged robot systems is due to the passive dynamics their limbs. The passive or natural dynamics of a system are those features of the robotic system which exist without drawing power from the actuators. For example a robotic limb with a spring has the capacity to store and release energy embedded in its passive dynamics. Animals take advantage of the passive dynamics of their limbs to increase energy efficiency and performance capability. One example is their use of compliant tendons[4]. Even the act of standing on a straight leg uses the passive dynamics of the limb to support the body weight without exertion from the muscles (actuators). This chapter outlines an optimization algorithm which tunes the kinematics (and therefore the passive dynamics) of a robotic limb to reduce energy waste.

An example of an exceptional tuning of the passive dynamics of an active robot is the Ranger by Ruina et al (See Section 2.1.3). To achieve this performance, Ranger's passive dynamics have been tuned to walk efficiently over level ground, while sacrificing its capability of robust terrain navigation.

Atrias, a biped robot developed by Hurst et al., is capable of efficient gait *and* robust terrain navigation capabilities. In this case, the robot's natural dynamics

are designed to mimic the Spring Loaded Inverted Pendulum (SLIP) Model, and a controller is then developed to utilize the natural dynamics of the robot in an efficient way (see Section 2.1.2). This approach has led to robust and efficient 3D bipedal locomotion [59]. A recent paper has shown, however, that there are significant energy losses in the Atrias system due to a phenomenon described by Abate et al. as Mehcanical Antagonism[2]. Seok et al. [61] developed a legged robot that is similar to a biological cheetah. To improve its energy efficiency for legged gaits, which require high torque and low speed, a motor with a high torque density is specially designed. As a result, the robot achieves an energy efficiency comparable to that of biological animals. A limitation of the system is its planar leg mechanism, which is incapable of lateral movements. This limits the robot's ability to balance and to reject lateral disturbances.

Chapter 3 introduces the KiTy SP robot limb structure. This chapter introduces the methods we use which allow the design of KiTy SP to be tuned to a given task trajectory. This will optimize the natural dynamics of the limb, significantly reducing the power required to perform the task.

The chapter also introduces an algorithm which produces the optimal gear ratio for a gear box in series with a DC motor. The gear ratio of a motor has a large effect on electrical power required by a task. This problem has been well studied over the years (See Section 2.7). A recent paper in the field Rezazadeh and Hurst review many subsequent results and derive a more general solution to the problem accounting for inertial loading as well as force loading at the end effector [57]. This chapter uses a model similar to Rezazadeh, but adds a torque dependent velocity constraint which forces the motor stay within its "continuous operating region". Also, this gear ratio

optimization routine is used as a subroutine in the aforementioned geometric tuning algorithm.

The contribution of this chapter is three-fold. We first propose a constrained optimization technique which searches design geometries, minimizing electrical power at the motor. Secondly we introduce an efficient gear ratio optimization algorithm which includes a full motor and loading model, and a method for accurately constraining the motor to torques and velocities within a typical "continuous operating region", given by motor manufacturers. Finally we optimize two limb structures: the KiTy SP structure, and a serial leg structure commonly employed in legged robot systems. We tune the geometric properties of both structures using the constrained optimization presented and compare the performance.

The rest of the chapter is structured as follows. We will first discuss in detail the optimal design procedure in Sec. 4.3. Results are presented in Sec. 4.5. We conclude in Sec. 4.7 with future work.

4.1 Robot Description

The last chapter introduced the KiTy SP robotic limb along with the details of the inverse kinematic solution to this mechanism. This chapter details the design optimization algorithm and compares the KiTy results with the standard biologically-inspired serial leg structure. This structure is the same as the serial portion of the KiTy SP mechanism. The inverse kinematic equations for which are described in Section 3.2.4.

Once written, the inverse kinematics for each structure can be differentiated to find the Jacobian matrix (J). This matrix relates the motor velocities with the end

effector velocity, denoted as V_{EE}:

$$V_m = J^{-1} V_{EE}. \tag{4.1}$$

The vector V_m is 3-by-1 which represents the instantaneous motor velocities at a given point in the data set. This vector will vary with the optimization state variables (in this case geometry) in order to achieve the desired end effector velocity vector (V_{EE}). The Jacobian also defines the relationship between the force produced at the end effector (F_{EE}) and the force at the motors (F_m):

$$F_m = J^T F_{EE}. \tag{4.2}$$

The vector F_m is a 3-by-1 vector similar to the velocity vector. It represents a force in motor coordinates. Each coordinate axis represents a torque produced by the motor. Like V_m, this vector varies with the optimization state variables in order to achieve the desired force at the end effector (F_{EE}) (note: V_{EE} and F_{EE} are given in the gait data set and do not vary with the optimization parameters).

4.2 Gait Data

We use gait data recorded from biological cats walking, converted from Suter et al. [65]. Figure 55a shows the three coordinates of the ground reaction force throughout the gait cycle. Figure 55b shows the three coordinates of the ankle position. The

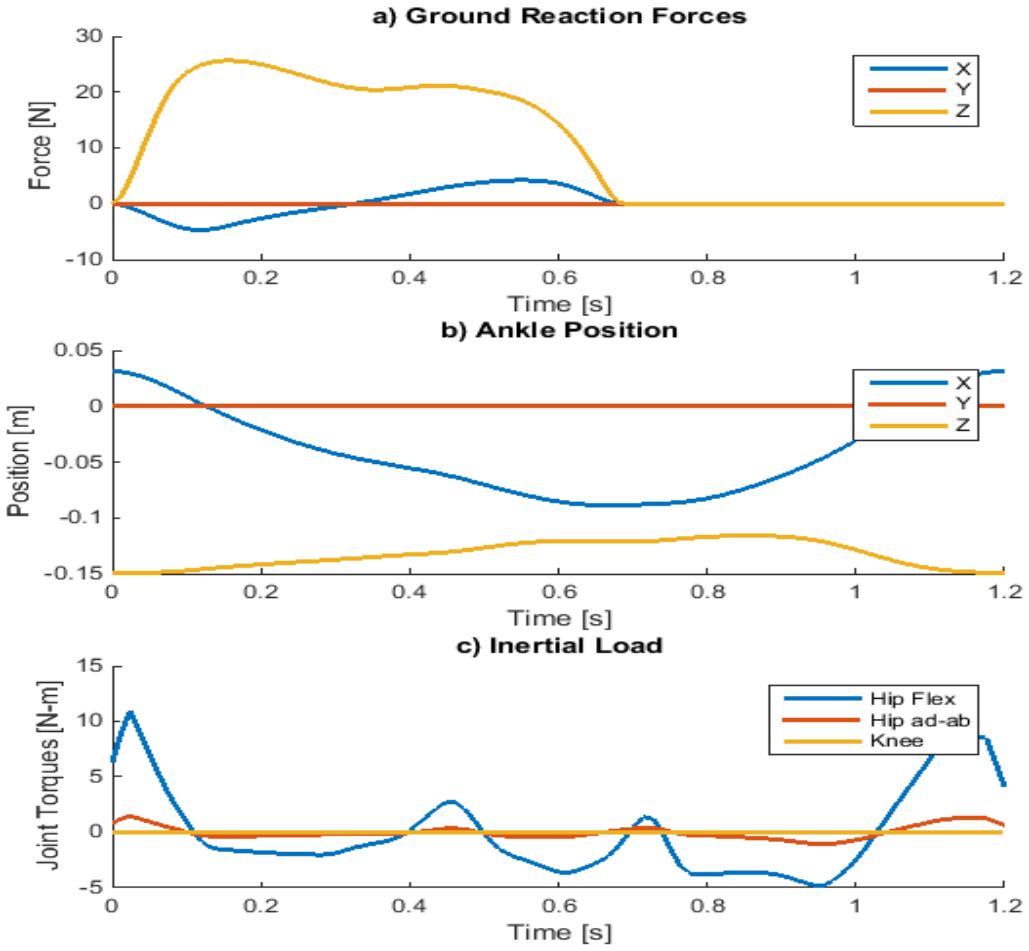

Figure 29. a)Ground reaction force on the end effector b)position of end effector c) inertial load on the end effector

inertia of each segment will be approximated as a point mass at its center. Using this approximation the moment of inertia (I) is calculated as follows:

$$I = m\,d^2, \tag{4.3}$$

where m is the mass of the leg segment, and d is the distance from the joint. The inertial load is calculated in the joint coordinates of the robot, and is plotted in Figure 55c.

4.2.1 Perturbing the Trajectory

We develop a method to perturb the design trajectory. The magnitude of the perturbation may be varied, affecting how much the design accounts for the presence of perturbations. To perturb the gait trajectories we first fit them with a seventh order Fourier series and add a small random variable to the coefficients:

$$T_\epsilon(x) = a_0 \sum_{n=1}^{7} (a_i + \epsilon_{(i,1)}) sin(n\,x) + (b_i + \epsilon_{(i,2)}) cos(n\,x) \tag{4.4}$$

Where a_i and b_i are the coefficients of the Fourier series. The variable ϵ is a randomly distributed number in a range which is centered at zero. It is produced by Matlab's rand function. The magnitude of the range of ϵ will be varied in the creation of the Pareto surface. If its range is zero the trajectory will be unperturbed. The effect of the perturbation will be relative to the scale of the perturbed data. For this reason, we set the force perturbation magnitudes to be three times as large as the position perturbation magnitudes, which produces a more even effect on the data.

Perturbing the coefficients in this way creates harmonically perturbed trajectory that is differentiable. The resulting perturbations are shown in Figures 32 - 33. We used a set of 50 of these perturbed trajectories to design the limbs, and another set of 50 to test the designs.

4.3 Design Optimization

Formulated here is an optimization routine which optimizes the geometry of the robot to minimize the electrical power required to follow the gait trajectory. The electrical power will be calculated by simulating the whole gait cycle and summing the squared electrical power required.

For this study the state variable for the optimization algorithm is a set of geometric properties which are the design variables of each respective structure. This set of geometric properties, for the serial leg structure, contains three variables. These variables are shown in Figure 22, they are: l_1(the length of the "thigh" section), l_2 (the length of the "shank"), and γ (the offset angle between the thigh and the abduction-adduction axis). These same three parameters are varied in the design optimization of the KiTy SP limb, along with nine parameters per parallel chain: 27 additional parameters in total. The three parallel chains of the KiTy SP limb apply force to the segments of the passive serial structure. The geometries of the three parallel chains influence the passive dynamics of the limb. This gives the optimization algorithm more degrees of freedom and a much larger solution space when designing the parameters of the KiTy SP structure, in comparison with the serial structure.

4.3.1 Problem Formulation

We start by formulating the electrical energy consumption:

$$P = U \cdot I, \tag{4.5}$$

where U are motor voltages, I are the corresponding motor currents. The motor voltage and current are dependent on both the geometry of the leg, and the gear ratio to the motors. We will consider the design of the motor gear ratio as a nested problem within the leg geometry problem (see Subsection 4.3.2).

4.3.2 Gear Ratio Sub-problem

A gear box is added between the electric motors and their associated joints, in order to increase the torque capacity of the motor and to reduce the electrical power consumption. In this section we discuss an efficient optimization routine for gear ratio design given the velocity and torque trajectories of a motor. Here we define gear ratio as the continuous ratio between the input and output speeds of the gearbox.

The first step to do this is to rigorously formulate a motor model which predicts the electrical power required by a set of velocities and torques, for given motor parameters, gear ratio, and desired torque and velocity. The motor current is approximated using the torque constant (K_t):

$$I = \frac{\tau_m}{K_t}. \tag{4.6}$$

The motor torque (τ_m) will be estimated from the torque load stipulated by the data set (τ_{out}), the rotational inertia of both the motor shaft (J_m) and the leg (J_l), and the

acceleration of the motor per the data (a_m):

$$\tau_m = \mathrm{J_m\, a_m} + \frac{\tau_{\mathrm{out}}}{g} + \frac{\mathrm{J_l\, a_m}}{g^2} \tag{4.7}$$

The motor current can now be obtained as a function of known parameters, via substitution:

$$I = \frac{\frac{\tau_{\mathrm{out}}}{g} + \mathrm{J_m\, a_{out}}\, g + \frac{\mathrm{J_l\, a_{out}}}{g}}{\mathrm{K_t}}. \tag{4.8}$$

Voltage in the motor will be estimated using the voltage constant of the motor (K_v), the required speed (V_m), and the armature resistance (R_a):

$$U = \frac{V_m}{K_v} + R_a\, I. \tag{4.9}$$

And after substitution:

$$U = \frac{g\, \mathrm{V_{out}}}{\mathrm{K_v}} + \frac{\mathrm{R_a}\left(\frac{\tau_{\mathrm{out}}}{g} + \mathrm{J_m\, a_{out}}\, g + \frac{\mathrm{J_l\, a_{out}}}{g}\right)}{\mathrm{K_t}} \tag{4.10}$$

Substituting Equations (4.10) and (6.6) into equation (4.5) gives a polynomial in g^4:

$$P = \frac{a_1\, g^4 + a_2\, g^2 + a_3}{a_4\, g^2} \tag{4.11}$$

$$a_1 = \mathrm{R_a}\, \mathrm{K_v}\, {\mathrm{J_m}}^2\, {\mathrm{a_{out}}}^2 + \mathrm{K_t}\, \mathrm{V_{out}}\, \mathrm{J_m}\, \mathrm{a_{out}} \tag{4.12}$$

$$\begin{aligned} a_2 = {} & \mathrm{K_t}\, \tau_{\mathrm{out}}\, \mathrm{V_{out}} + \mathrm{J_l}\, \mathrm{a_{out}}\, \mathrm{K_t}\, \mathrm{V_{out}} \\ & + 2\, \mathrm{J_m}\, \mathrm{R_a}\, \mathrm{a_{out}}\, \mathrm{K_v}(\tau_{\mathrm{out}} + \mathrm{J_l}\, \mathrm{a_{out}}) \end{aligned} \tag{4.13}$$

$$\begin{aligned} a_3 = {} & \mathrm{R_a}\, \mathrm{K_v}\, {\mathrm{J_l}}^2\, {\mathrm{a_{out}}}^2 + 2\, \mathrm{R_a}\, \mathrm{K_v}\, \mathrm{J_l}\, \mathrm{a_{out}}\, \tau_{\mathrm{out}} \\ & + \mathrm{R_a}\, \mathrm{K_v}\, {\tau_{\mathrm{out}}}^2 \end{aligned} \tag{4.14}$$

$$a_4 = {\mathrm{K_t}}^2\, \mathrm{K_v} \tag{4.15}$$

Each of these four coefficients is a vector of length n. They vary as a function of the leg geometry but are independent of the gear ratio. The formal fitness function is as follows:

$$f_{min} = \sum_{j=1}^{m} \sum_{i=1}^{n} P_{i,j}, \tag{4.16}$$

where m is the number of motors, and n is the length of the data vector. This one-dimensional sub-problem may be solved efficiently using a standard optimization tool.

The next section will look at how to properly constrain this problem.

4.3.3 Constraining the Optimization Routine with the Condition Number of the Jacobian

The optimization routine being described here optimizes the leg about a particular set of end effector positions, velocity, and forces. A concern is that the limb will be over-specialized for the task. This is concerning because, in most applications, the task may vary slightly due to disturbances or other such un-modeled phenomena. To be sure that the limb will be able to perform in such an environment, we will constrain the condition number of the Jacobian. The condition number of a matrix is the ratio of the largest eigenvalue to the smallest eigenvalue of the matrix. In the case of the leg Jacobian, the largest eigenvalue represents the maximum velocity amplification through the leg. The associated eigenvector represents the direction which the leg moves the fastest. The smallest eigenvalue and eigenvector represents the

magnitude and direction which the leg least amplifies the motor velocities. Because of the dual nature of the Jacobian, the eigenvalues of the Jacobian are the inverse of the eigenvalues of the transformation matrix relating motor torques to end effector forces. This means the eigenvector associated with the largest velocity amplification is also associated with the smallest force amplification, and the eigenvector associated with the smallest velocity amplification is also associated with the direction of largest force amplification. As the robot approaches a singular position the smallest velocity eigenvalue approaches zero and thus the large force eigenvalue grows to infinity. In the singular position the robot loses a freedom, and can carry loads in the associated direction structurally with zero torque from the motors.

While this may seem desirable in some cases, it also imposes undesirable limitations on the leg. If the desired trajectory of the end effector changes slightly in the direction of the lost freedom the leg will not be able to perform the task. For this reason the condition number of the leg Jacobian is constrained to be less than a threshold value for every point along the leg trajectory.

4.3.4 Modeling the Motor Constraints

This optimization problem is best constrained by the continuous operating region of the motor. The following section will show how this region may be approximated. A motor's continuous operating region can be approximated as a set of three linear equations. These are: velocity-independent maximum torque (τ_{max}), a torque-independent maximum velocity (V_{max}), and a torque-dependent max velocity line ($V_{max}(\tau)$). These three equations may be written as a vector equation:

$$A \begin{bmatrix} V_m \\ \tau \end{bmatrix} + B \leq 0. \tag{4.17}$$

Figure 30 shows the continuous operating region of the kind of high performance motors that are commonly used in robotic systems (photo courtesy of Maxon Motors). Figure 31 shows the linearized operating region used for the optimization problem. In

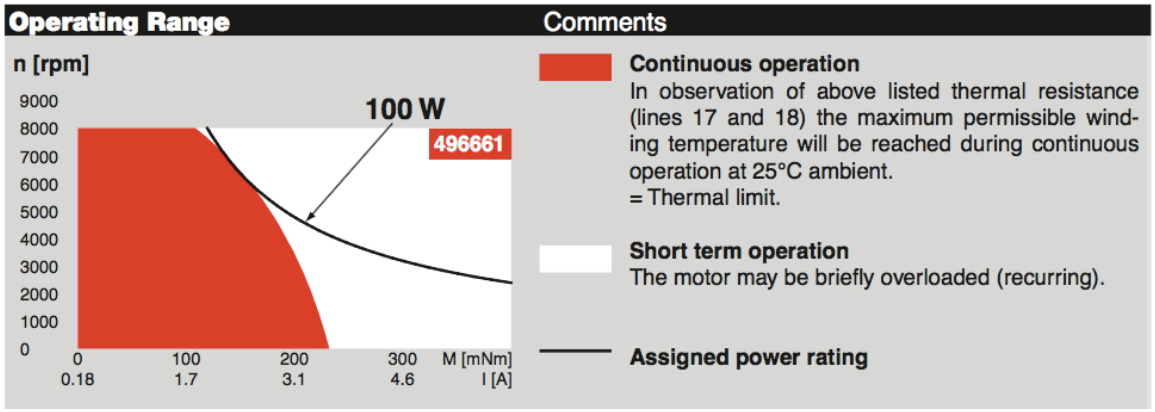

Figure 30. An example of a motor's continuous operating region. Credit: Maxon Motors.

order for a particular leg configuration to satisfy the constraints, every point in the data set must satisfy the linear constraints.

4.4 Pareto Optimization

Since there are two objectives that we are designing for (unperturbed efficiency and robustness), we chose to perform a multi-objective optimization algorithm which resulted in a Pareto frontier. In a multi-objective optimization routine the optimization

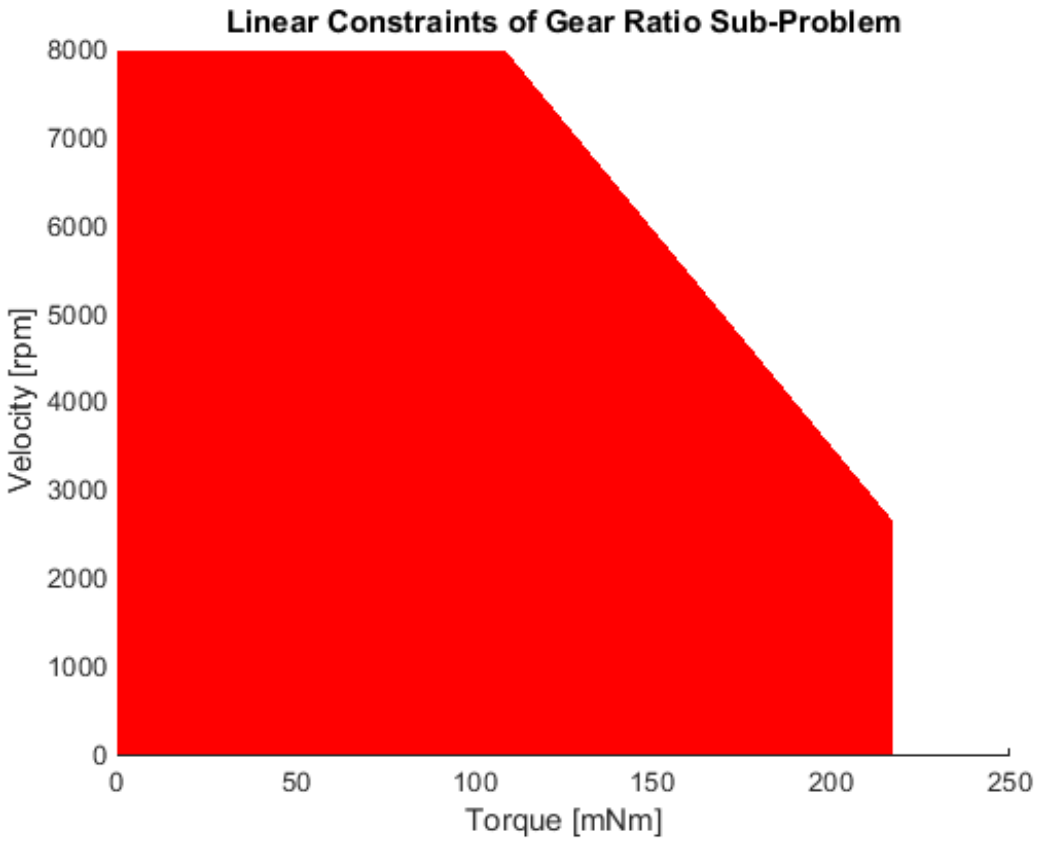

Figure 31. The linearized constraints of the gear ratio sub-problem

commonly run several times, each time with a slightly changing optimization metric. In our case we do just this. We begin by optimizing both leg structures to the unperturbed data set. Next, we re-optimize with a small-magnitude perturbation added to the design trajectory, and again with an incrementally larger trajectory. In total ten design optimizations were run per leg structure with a perturbation magnitude (ϵ) which varies incrementally from zero to .01. Figures 32 and 33 show the fully perturbed gait trajectories. It can be seen that at a magnitude of .01 the trajectories are sufficiently perturbed. It can be argued that this value for magnitude is somewhat arbitrary. That argument is the source of the motivation for varying the perturbation magnitude in the multi-objective optimization instead of using a

weighted objective function between a fully perturbed and an unperturbed model. Varying the magnitude of the perturbation allows the designer to see what aspects of the design change as perturbations increase. It also allows conclusions to be made about threshold perturbation magnitudes, which, in this case, do exist. For example, Section 4.5 shows that designing for a perturbation of larger than .04 does not result in significantly better perturbation robustness, but does result in poorer unperturbed efficiency values.

4.5 Results

In this section, we present the results of the optimization routines, and compare the KiTy SP results to the purely serial structure's. The KiTy SP robotic leg was more efficient and more robust than the serial structure. These results are presented in a series of figures (Figures 34-37).

Figure 34, shows the Pareto frontier associated with the KiTy SP structure in black, and the frontier associated with the serial leg in red. On the x-axis is plotted the value of f_{min} for the geometries performing unperturbed gait trajectory, a metric which is inversely related to the leg's unperturbed efficiency. On the y-axis f_{min} is plotted for each geometry for a set of 50 perturbed trajectories using the larges value for . This is a measure which is inversely related to the leg's robustness to perturbation. A low value on each axis is more desirable. As can be seen in Figure 34, the KiTy SP structure out-performs the serial structure on both metrics.

Figures 35 and 36 reinforce this result. This figures plot the relative increase in f_{min} of each design compared to the best value achieved by any design, for both the unperturbed and fully perturbed case. The yellow bars in the figures display the

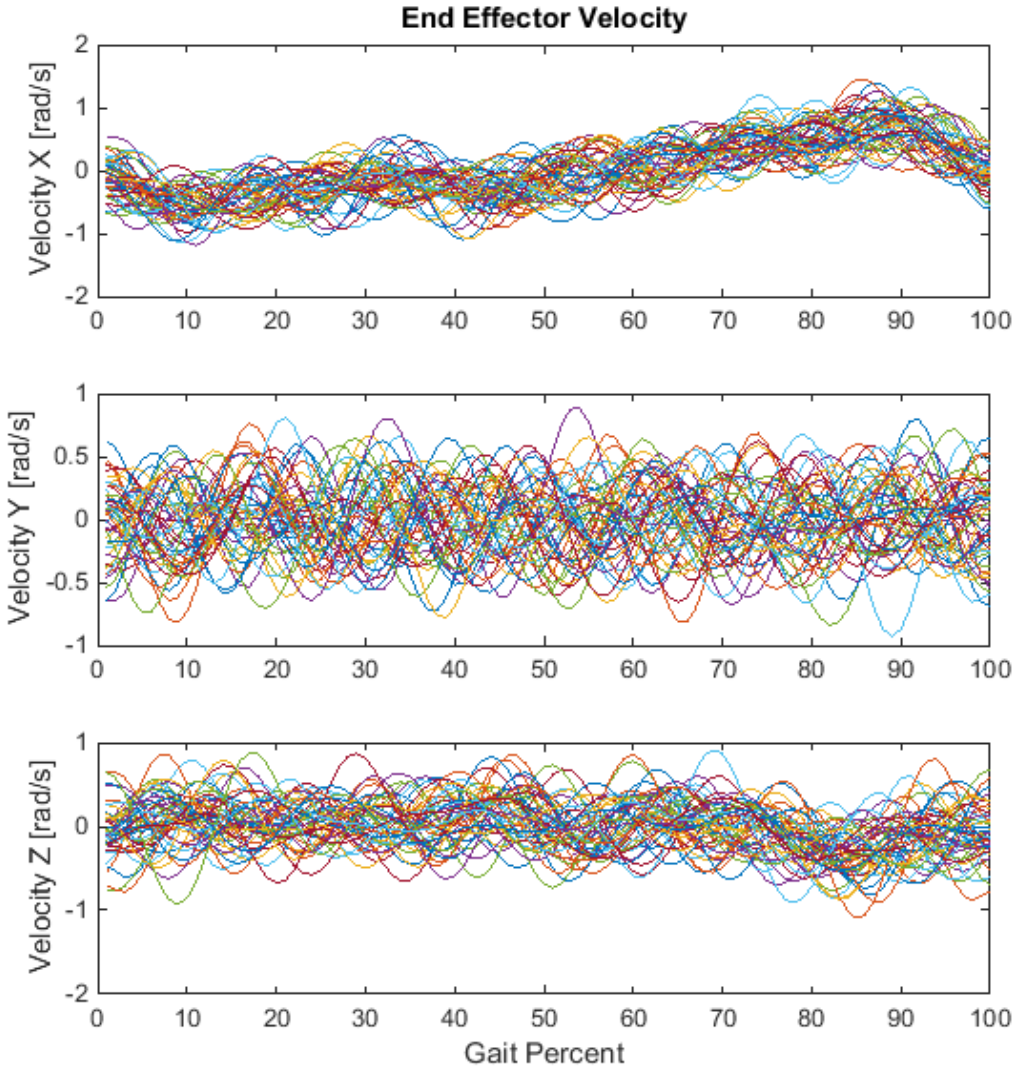

Figure 32. The perturbed end effector velocity trajectory.

percentage increase in f_{min} from the best design, which was the KiTy SP optimized with $\epsilon = .01$. The blue bars represent the percentage increase in f_{min} from the best f_{min} value achieved, which was achieved by the KiTy SP geometry optimized for $\epsilon = 0$.

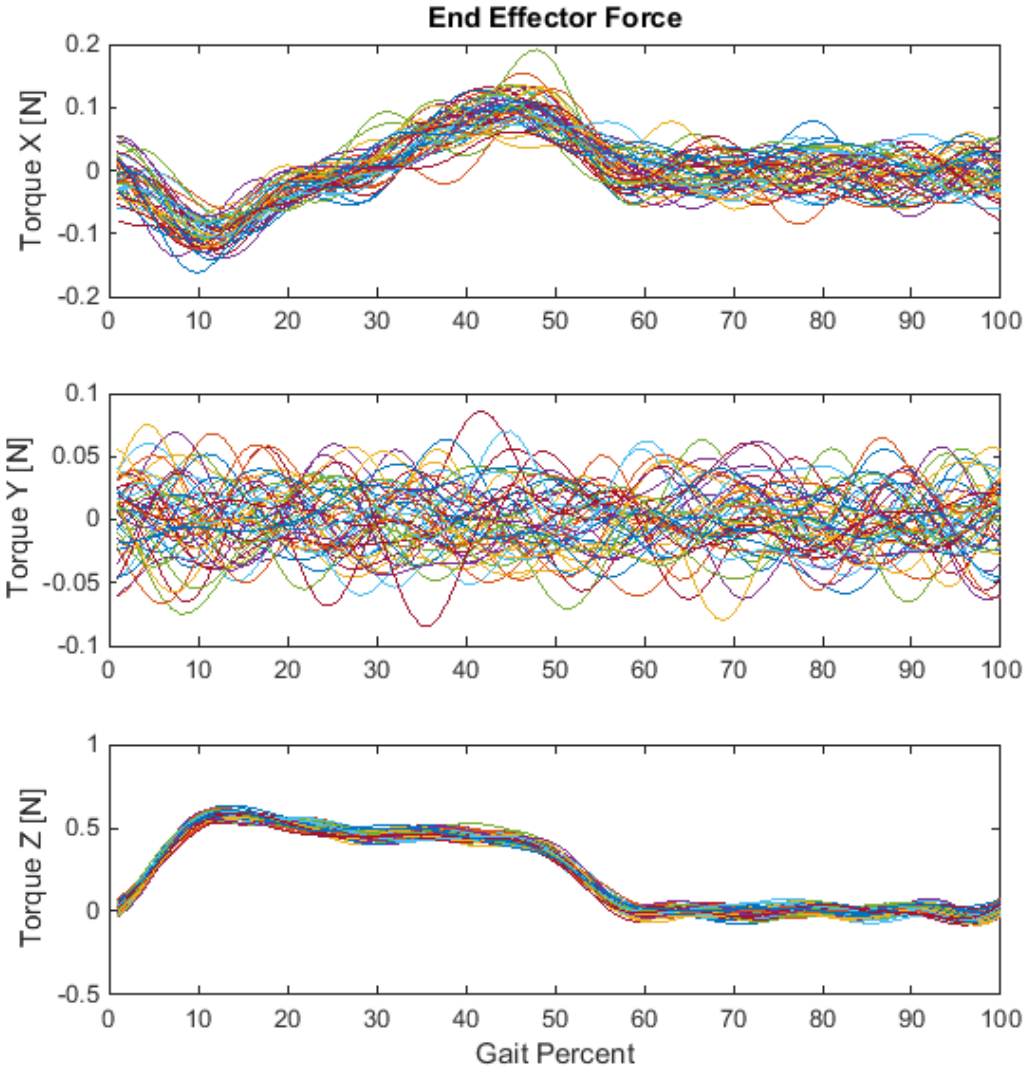

Figure 33. The perturbed end effector force trajectory.

The equation for this metric is as follows:

$$\Delta_1 = 100\frac{f(\hat{X}_\epsilon) - MV}{MV}, \tag{4.18}$$

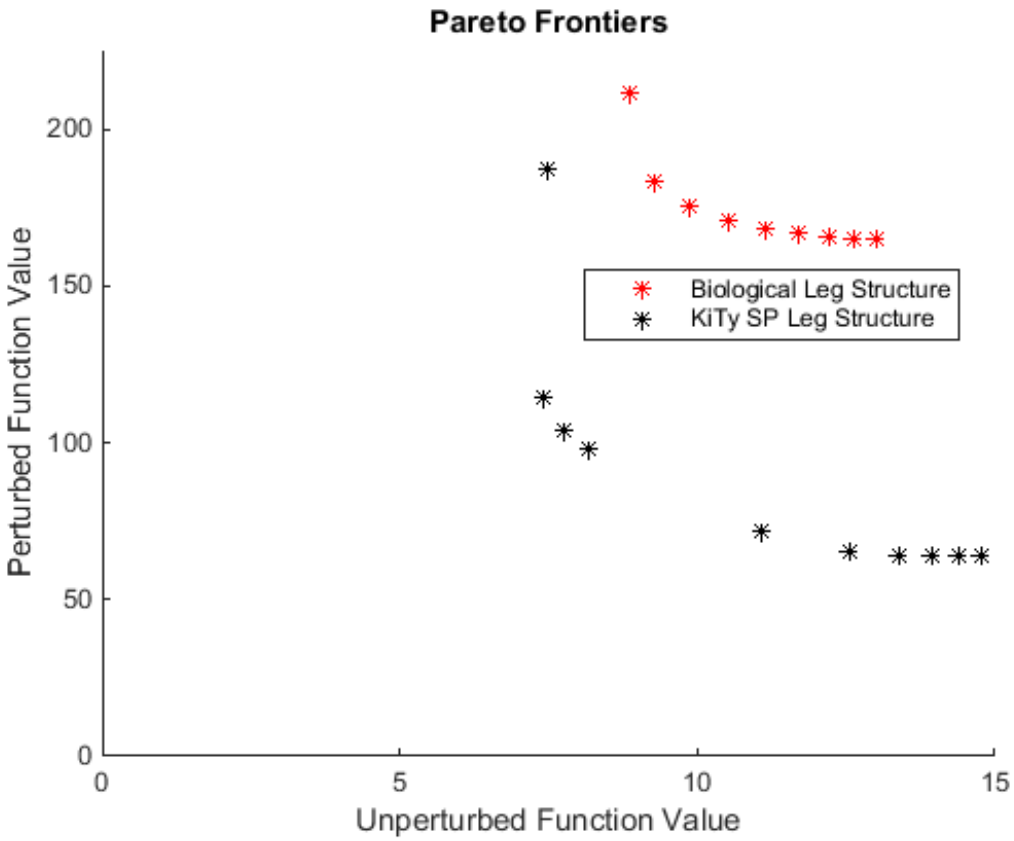

Figure 34. Pareto plot of comparison study: leg performance on unperturbed trajectory versus performance on set of randomly perturbed trajectories.

where MV is the minimum value achieved by any design.

As expected, the perturbed performance gets better and better (the height of the yellow bars decreases) as the design value ϵ increases. In other words, the larger the perturbation to the design trajectory is, the better the design will function under large perturbation. As mentioned earlier, a threshold ϵ value seems to exist for the KiTy SP structure. Above $\epsilon = .04$ the perturbed performance does not improve significantly, but the unperturbed performance continues to worsen. In contrast, the serial structure has a similar threshold at $\epsilon = .02$. This, however, is not the only difference between the structures. Nor is it the most significant: the KiTy SP structure achieves the lowest

values of f_{min} for both the unperturbed and perturbed performance metrics. As a result the blue bars in Figure 35 start at zero, and the yellow bar decreases to zero. In contrast, the serial limb never scores below 150 percent for the perturbed performance comparison. In unperturbed performance the serial leg has an f_{min} value which is 17 percent higher than the KiTy SP limb's best mark. Furthermore, all but one of the KiTy SP geometries are within 80 percent of the best case for perturbed performance, and half of them are within 3 percent of the best case. In summary the KiTy SP structure is dramatically better at handling perturbations in the task trajectory, and is more efficient in general than the biologically-inspired serial structure.

Figure 37 compares the performance of each KiTy SP geometry against the performance of the serial limb solution for a given ϵ value. The equation for the metric (Δ_2) is:

$$\Delta_2 = 100\frac{f_{KiTy}(\hat{X}_\epsilon, T_\epsilon) - f_{serial}(\hat{X}_\epsilon, T_\epsilon)}{f_{serial}(\hat{X}_\epsilon, T_\epsilon)}. \tag{4.19}$$

Where f_{KiTy} is the performance function for the KiTy SP Structure, and f_{serial} is the performance function for the serial structure. The value of Δ_2 for a given ϵ represents the percentage improvement associated with using the KiTy SP Structure. For the unperturbed case the KiTy SP is 11 percent more efficient than the serial structure. As the value of ϵ increases, the KiTy SP structure outperforms the serial structure even more. For $\epsilon = .01$, the maximum perturbation level, the KiTy SP structure is 45 percent more efficient than the Serial structure.

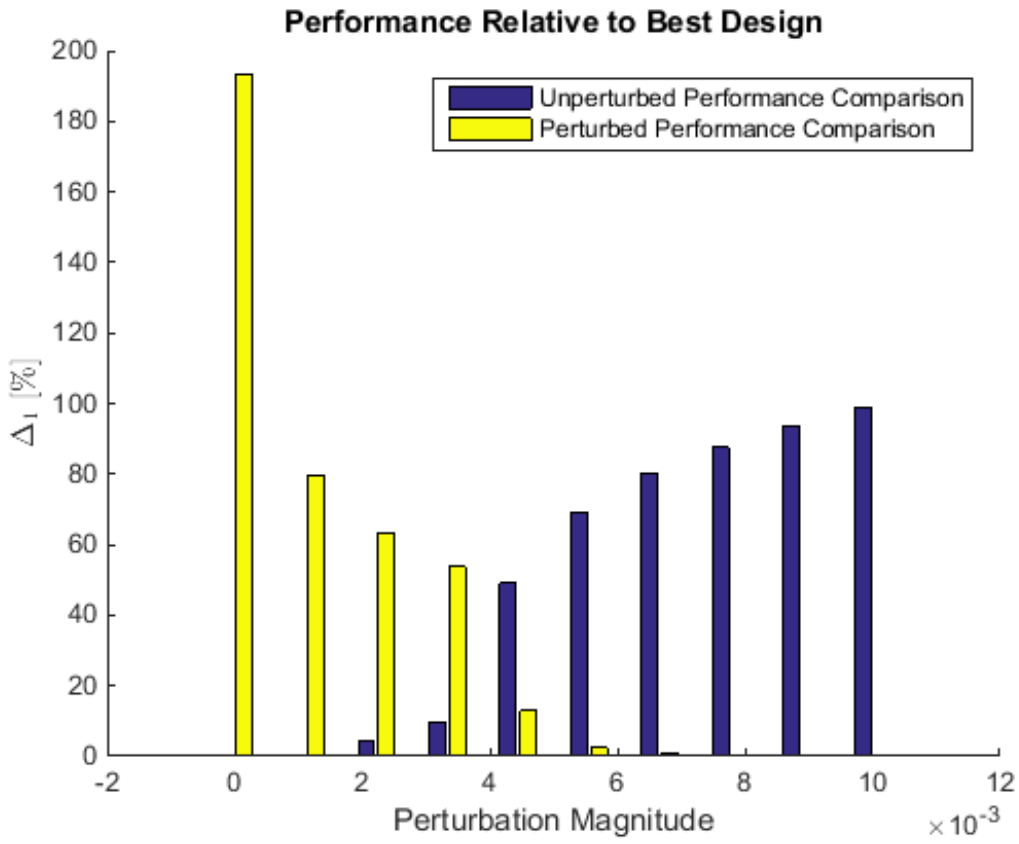

Figure 35. Performance comparison of each of the 10 KiTy geometries.

4.6 Conclusions

We show that the KiTy SP Leg structure may be tuned to significantly reduce the power required to perform a task. The tuned geometry of the leg structure is more efficient than the biological leg structure, even when both are tuned by the same process. Not only this, the structure was also able to perform randomly perturbed tasks more efficiently than a similarly-tuned biological structure, making it more efficient and more robust to trajectory changes. The KiTy SP structure's superiority may partially result from the fact that it has more design variables which may be tuned by the optimization algorithm. In addition, the KiTy SP structure has the

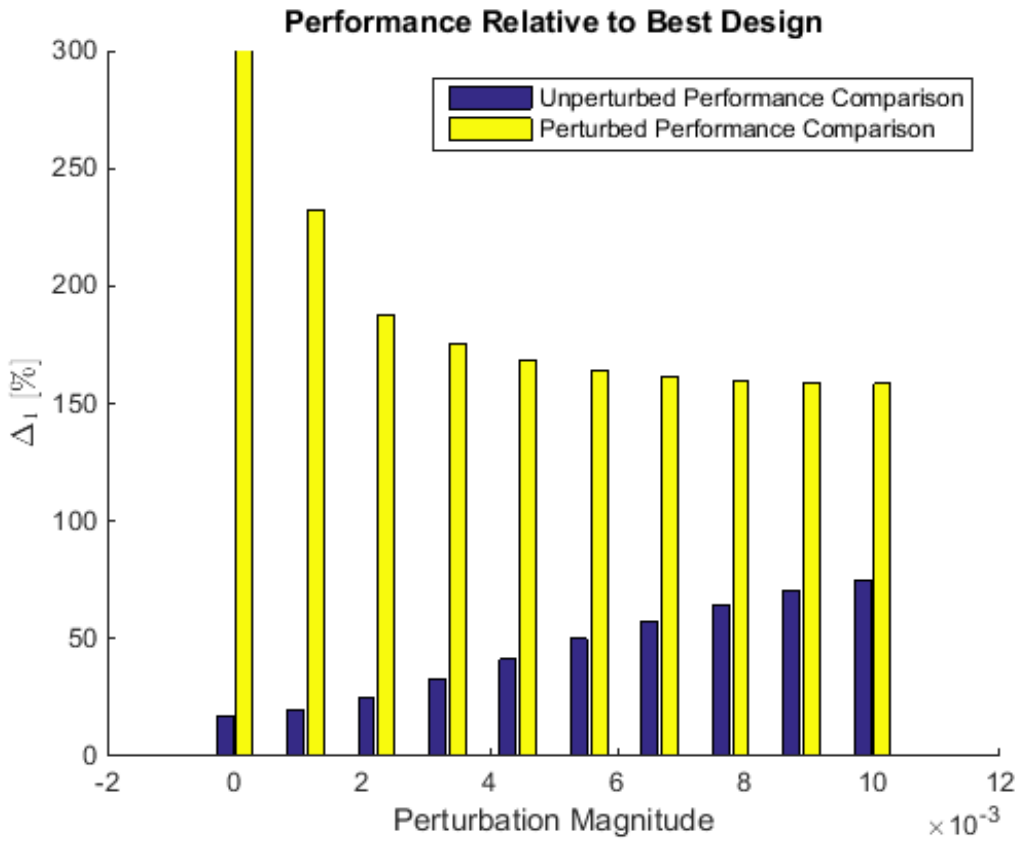

Figure 36. Design performance comparison of each of the 10 serial geometries.

natural dynamics of the biological structure built into the serial portion of its hybrid structure.

4.7 Future Work

The work presented in this chapter represents only the beginning of this line of research. In the future this kind of task-specific optimization of design parameters may be extended to other aspects of the design. One such attribute well-suited for this approach is the stiffness of mechanical springs in the design. These springs could

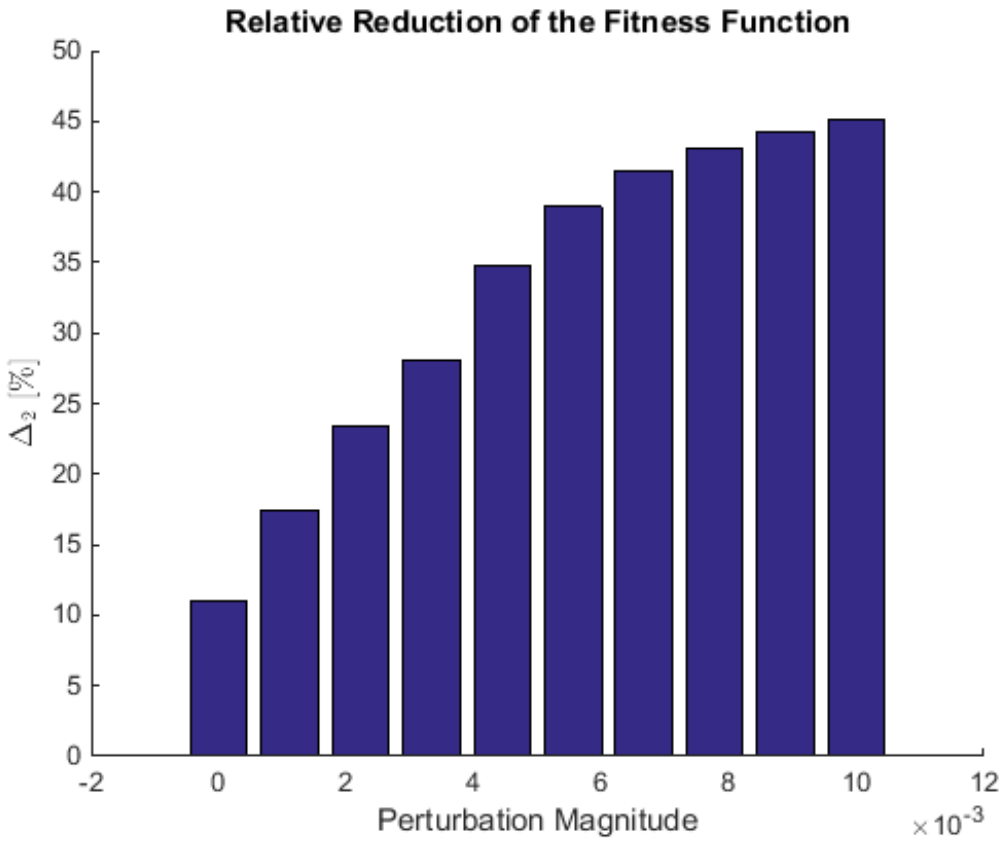

Figure 37.

be in-parallel or in-series. An optimization of this type could benefit a multi-actuator robot such as the one described in this chapter.

Another opportunity to advance the work presented here would be to improve the gait trajectory data set. To this end we could rigorously simulate a quadruped robot with a three-dimensional walking controller and realistic dynamics. This would allow the creation of a realistic trajectory of end effector forces and velocities.

An alternate future work route is to find new applications for this type of mechanical specialization. Robot arms are used in many environments including industrial applications. The design optimization method outlined in this chapter could be applied to one of these, potentially creating valuable energetic savings.

4.8 Appendix: Solution Families

The design optimization tool described in this chapter resulted in the discovery of three design families. These design families resulted from the unperturbed data trajectory, and were not meaningful once the Pareto optimization was performed. These results are presented here as an appendix.

The optimization routine was able to successfully discover several families of power-efficient limb geometries. These are referred to as families of solutions because they represent three different embodiments of the mechanism. Each of these embodiments has special characteristics that vary from the others. These families were discovered during the process of designing the robot and developing the optimization routine. It is important to evaluate different design optimization solutions because they may have other benefits or drawbacks not modeled by the fitness function. Three such families will be presented here.

4.8.1 Family One

The first of these families is shown in Figures 38 and 39. Figure 38 shows the geometry of the linkage. For each of the three plots the leg is near mid-stance in the gait trajectory. The links are colored in the same scheme as Figure 20. The red lines represent the two-link mechanism connected to motor one. The light blue lines represent the two-link mechanism connected to motor two. The dark blue lines represent the two-link mechanism connected to motor three. Finally, the black lines represent the serial portion of the leg. Figure 39 shows some performance characteristics of the geometry. The blue bars are sums of the positive power across

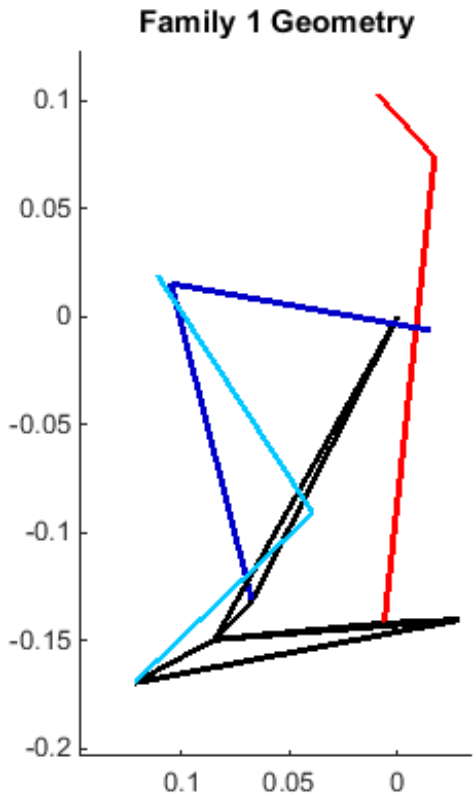

Figure 38. Plot of linkage geometry of family one mechanism

the gait trajectory. The pink bars are the sums of negative power across the gait trajectory. The first three sets of bars represent each motor's contribution. The fourth bar is the total combined power. This family of geometry solutions has several notable attributes. One notable feature is that the optimization has found a solution in which motor two is inactive. Another thing to note is that the black lines are long causing excessive knee flexion. In this case, the knee joint is almost horizontal with the ankle joint in the configuration shown. For this application this may be a negative feature of the mechanism as it may interfere with gait.

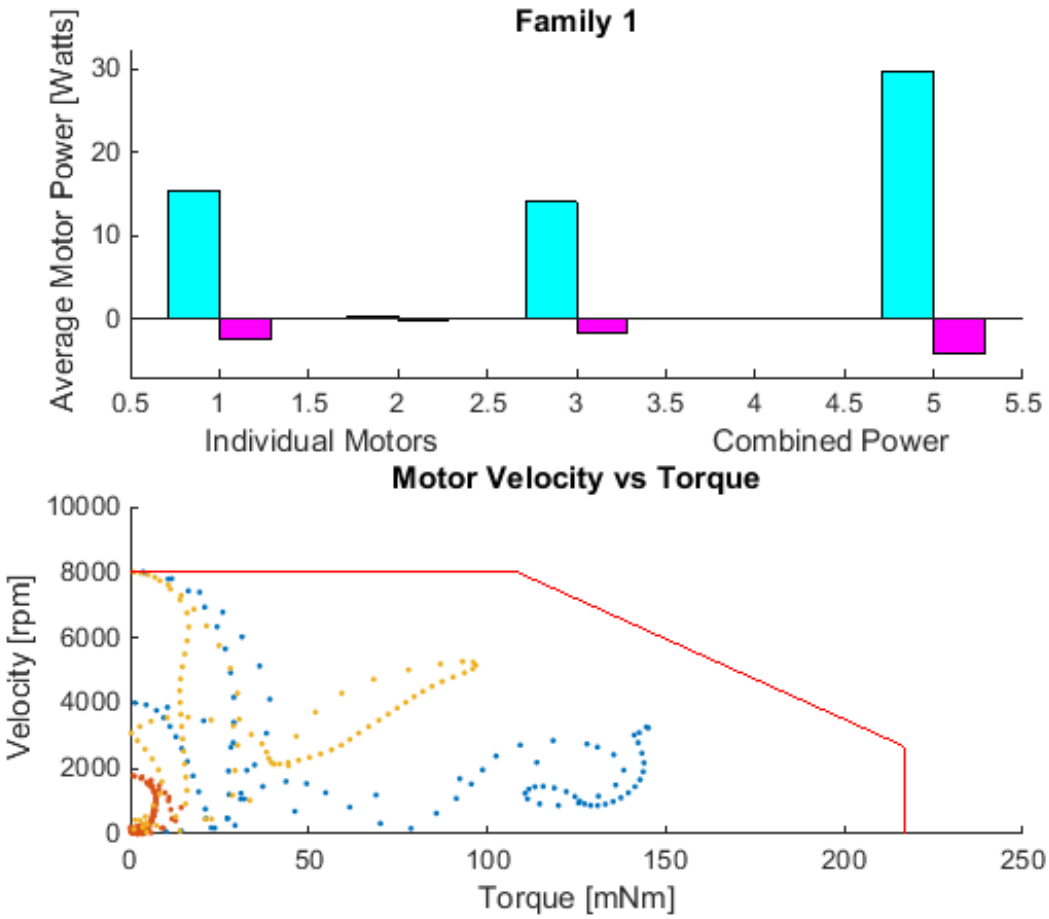

Figure 39. Plot Showing motor power sums and a torque velocity scatter plot for family one

4.8.2 Family Two

The second family of leg geometry is similar to the first in that the optimization found a geometry which reduced one motor power to almost zero. This can be observed in Figure 41 with the small size of the third set of motor power bars. The power is reduced on this motor in a very different manner than it was in family one. In family two, the optimization reproduced a freedom set which is similar to a serial leg. Motor one crosses only the hip flexion axis. Motor two crosses only the knee flexion axis.

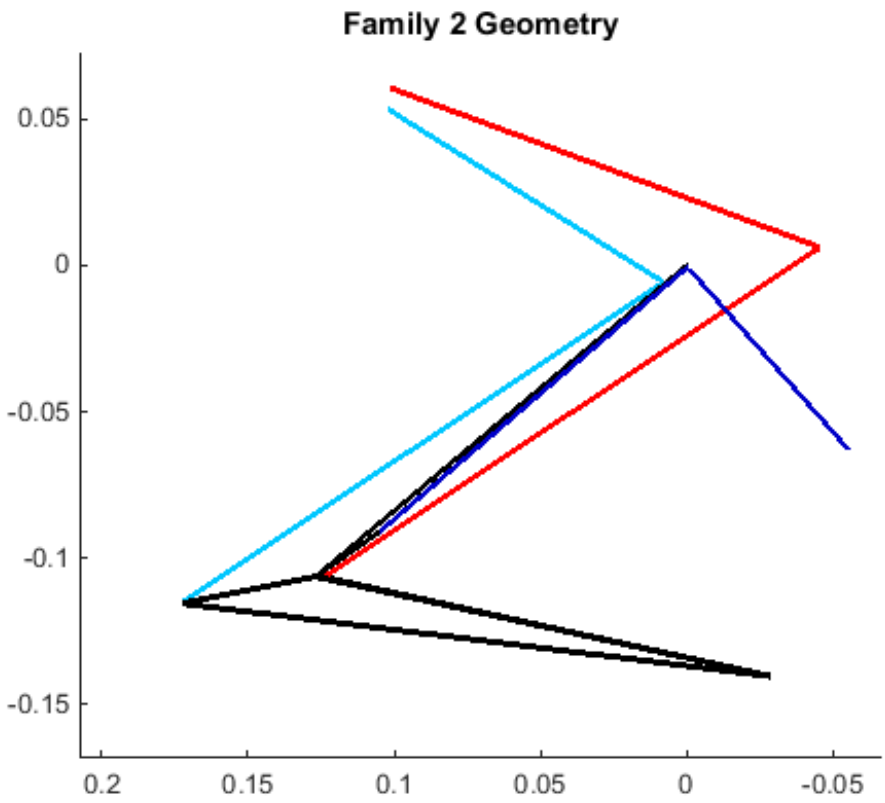

Figure 40. Plot of linkage geometry of family two mechanism

Motor three controls only the hip adduction axis. This is the reason that motor three is contributing zero power to the system. The gait trajectory has zero velocity into or out of the page. This family, like family one has long leg segments compared to the end effector position, resulting in a highly flexed knee angle, and an increase in the size of the leg envelope in the front-to-back dimension.

4.8.3 Family Three

Family three is unique from the other two families for several reasons. The first of these is that all three motors exert significant positive power. This can be noted

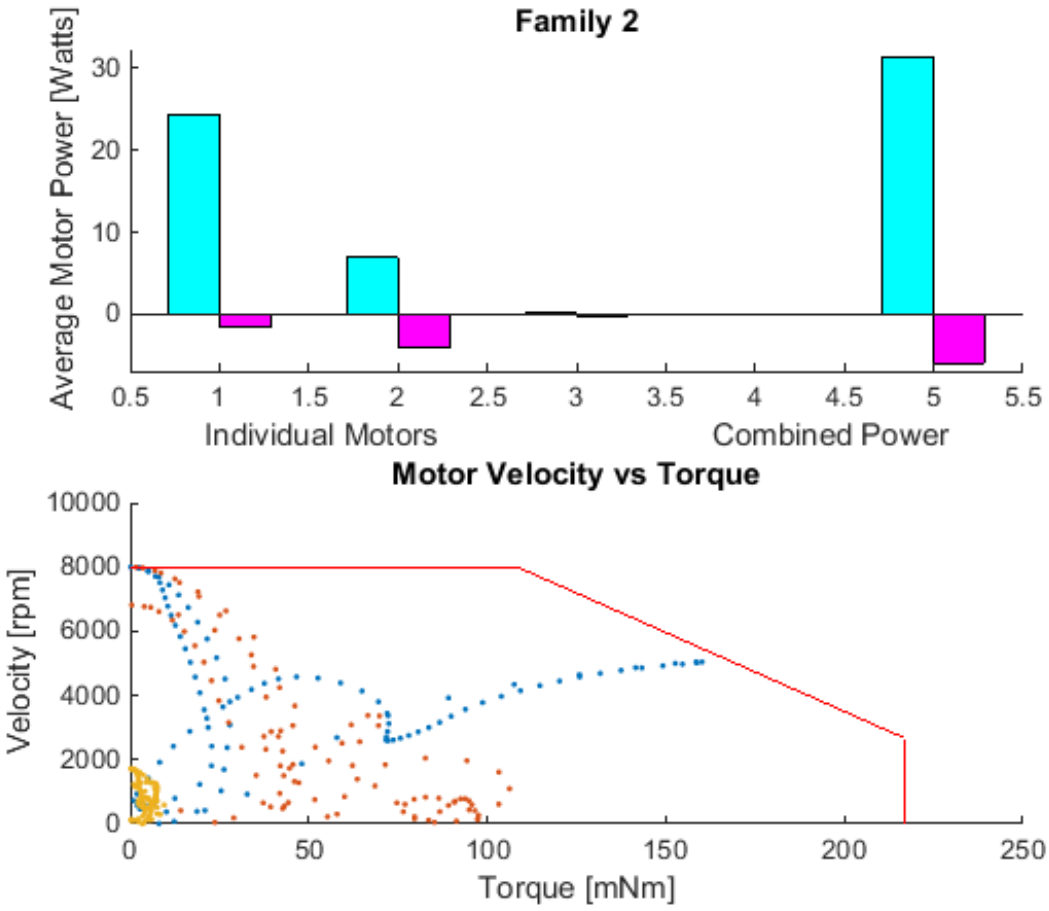

Figure 41. Plot showing motor power sums and a torque velocity scatter plot for family two

from Figure 43. Another beneficial feature of this family is that it has leg lengths which are similar in scale to the leg of a biological quadruped (note the more extended knee angle in Figure 42). Since the leg segments in this family are smaller than their counterparts in the other families, the envelope and the inertia of the leg are reduced.

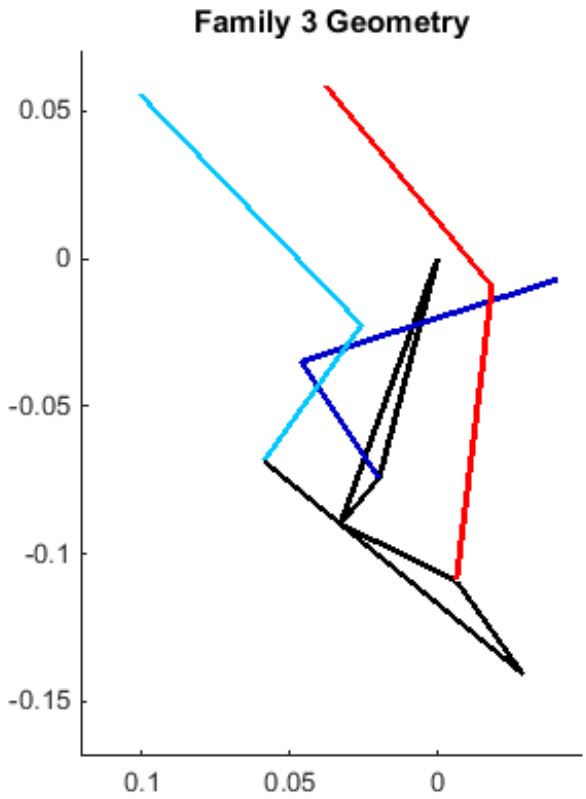

Figure 42. Plot of linkage geometry of family three mechanism

4.8.4 Comparing the Families

The previous section has described the various aspects of the families which make them unique, but a direct comparison of performance criteria is necessary to choose a result. First let us examine the average power output of the leg across the data set. These values are plotted in the bar graphs shown in Figures 39, 41, and 43.

These results are summarized in Table 5.

If the average power is the only design rubric considered, the obvious choice would be family one as this family has the highest efficiency value. However, other factors play a role choosing the best design.

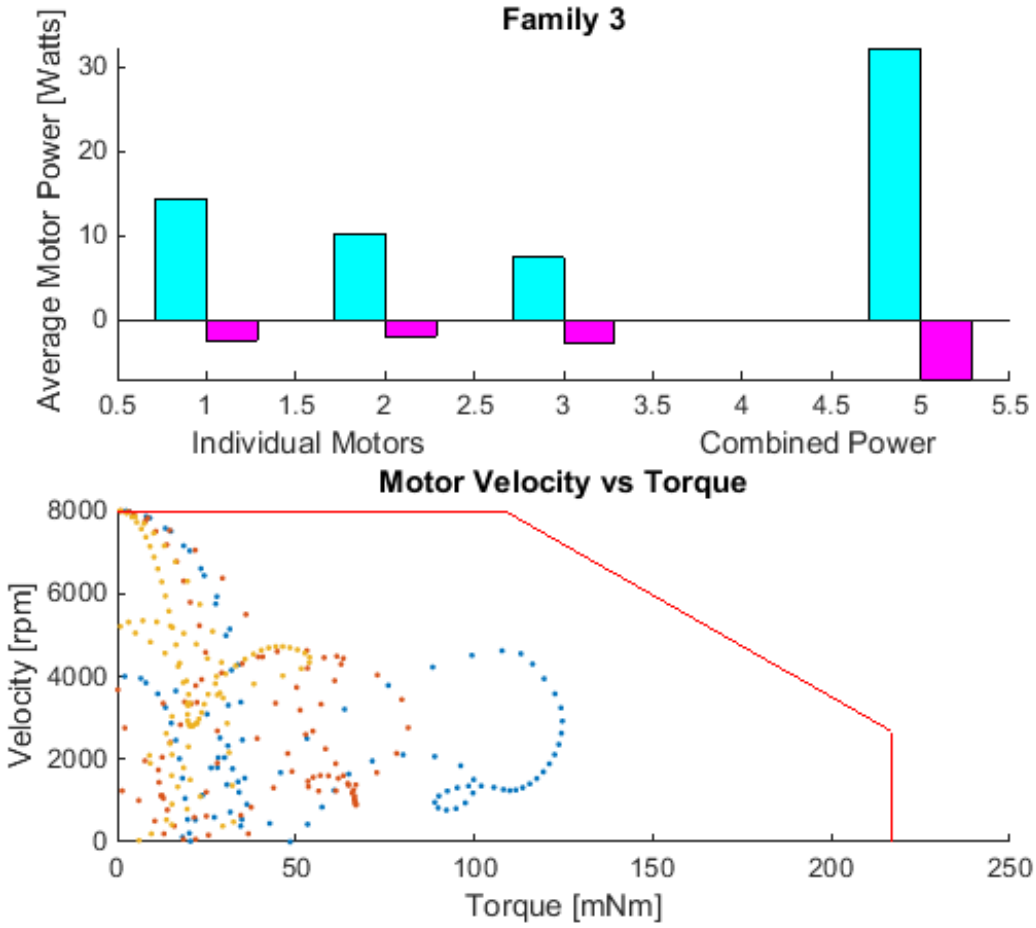

Figure 43. Plot showing motor power sums and a torque velocity scatter plot for family three

One of these factors is the peak motor power. Figure 44 shows the electrical power exerted by the motors throughout the gait cycle. Regions of time when all motors are exerting positive power are marked with vertical green lines. Regions of time when all three motors are being used in braking are marked with vertical red lines. Regions in which the motors are being used antagonistically are marked with vertical orange lines. Notice the peak value for each family of leg designs. This peak occurs at approximately .1 seconds, which corresponds to the first peak in ground reaction force (seen in Figure 55a). Table 6 lists these peak power values, the time that they occur

Table 5. Summary of Average power values for each family

Family	Average Power	
	Positive	Negative
1	29.7 w	-4.1 w
2	31.4 w	-5.9 w
3	32.2 w	-6.9 w

and for which motor they occurred. It can be seen that family two has a significantly

Table 6. Peak power values for each family

Family	Peak Time	Peak Motor	Peak Power
1	.13	3	53.8
2	.13	1	88.4
3	.1	1	55.8

higher value for peak electrical power than the others. This is due to the fact that the peak power motor is carrying a large portion of the load during the time when the peak power occurs. In family three at the moment of peak power output, all three motors are carrying a portion of the load, so much so that the peak time for an individual occurs a moment slightly before (.1 seconds) the peak power output for the system (.13 seconds). In spite of this family one has the lowest peak power value. This is an important result because the peak power in a gait cycle will often correspond to the point closest to the motor constraint. Reducing this value will mean that the limb will be able to perform even in situations which require more force or velocity than modeled in the gait trajectory.

Another factor that can be used to differentiate between the leg designs is the condition number of the Jacobian. The meaning of this number is described in Section 4.3.3. Figure 45 shows a plot of the condition number of the Jacobian of each leg family across the data set. A smaller condition number means the limb is further from

a singular position. All else equal, a smaller value is beneficial. Our initial hypothesis was that a higher value for the condition number may be beneficial to reduce the average power consumption of the leg across the data set. However, it can be seen by

Table 7. Condition Number vs Electrical Power

Family	Condition Number	Average Power	Peak Power
1	7.6	29.7 w	53.8 w
2	2.9	31.4 w	88.4 w
3	5.8	32.2 w	55.8 w

inspection from Table 7 that the results do not support this hypothesis. Family two has the best condition number, but not the lowest average power mark. However there does seem to be a negative correlation between peak power and condition number. It is not clear if the decrease in peak power is caused by the increase in condition number, or if the two variables happen to relate in this case. This is the extent of the comparisons that will be made between the three families of leg geometry. The next section will detail what conclusions can be made from these results.

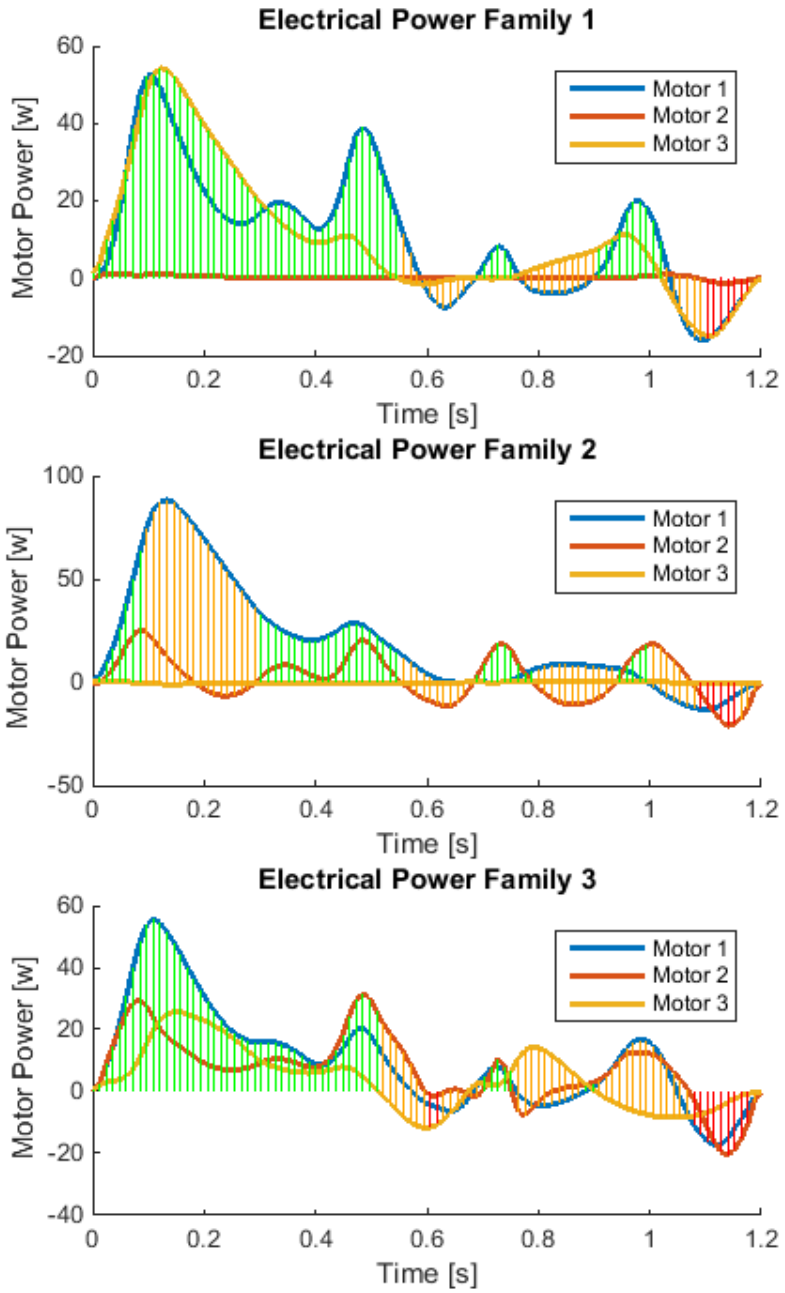

Figure 44. Electrical Power over the gait cycle for each of the families

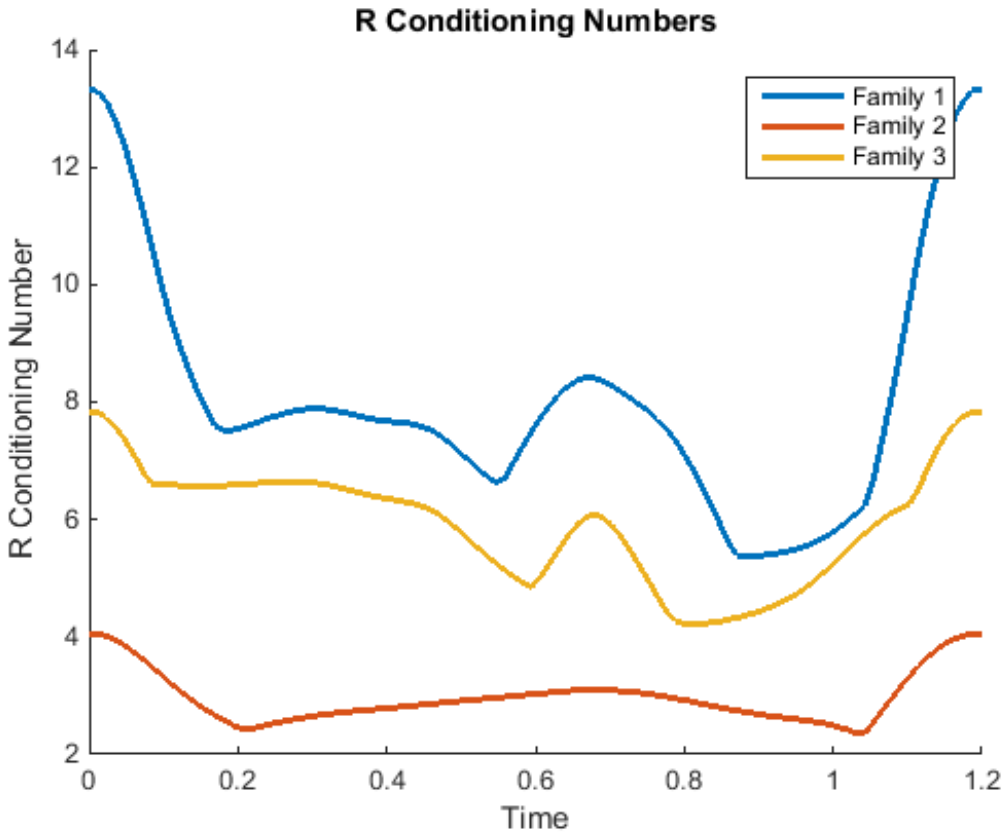

Figure 45. Condition number of the Jacobian over the gait cycle for each of the three families

Chapter 5

UNDERSTANDING POWER LOSS DUE TO MECHANICAL ANTAGONISM AND A NEW POWER-OPTIMAL PSEUDOINVERSE FOR REDUNDANT ACTUATORS

5.1 INTRODUCTION

State-of-the-art legged robot systems are less energy efficient and less powerful than their biological counterparts. Progress of these systems has been bottlenecked by relatively slow growth in power storage and generation technology. To counteract this slow growth, research must be focused on reducing power losses in legged systems. This chapter outlines a phenomenon, called mechanical antagonism, whereby multiactuator robot arms waste considerable power due to an effect first mentioned by Waldron and Kinzel's early work [72]. In [2], Abate et al. resurrected this work showing that even cutting edge robotic systems suffer significant losses due to this phenomenon. The contributions of this chapter are twofold. We first reveal the directional nature of antagonistic power losses in robotic arms, using a 2-DoF robot manipulator for demonstration. Specifically, we partition the end effector space into geometric regions using concepts from screw theory, and identify the antagonistic regions. We then present suggestions on how to avoid mechanical antagonism in designs for performance-driven multiactuator robotic limbs. In the second part of the chapter, we present a new pseudoinverse for over-actuated robot arms that minimizes losses due to mechanical antagonism.

We argue that energy efficiency and power density should be chief concerns for robot developers. As evidence of this, we turn to the growth rate of various technologies. The performance of power storage and generating machines has not been growing at the same rate as information technology. This is well known, and has also been documented in literature [41, 19, 5, 6]. Figure 1 compares the growth of information technology to that of battery technology, as an example of the stark difference. During the reported time-frame (from 1990 to 2003), battery power density improved by 20

percent. In contrast, hard drive storage per dollar increased by over two thousand percentage points. As can be seen in the figure, relative to many metrics of information technology growth, battery technology has not been growing at all. This is the case for seemingly all power storage and generation technologies [6, 19]. Rapid exponential growth in power technologies would allow for extremely useful legged robots. In absence of this, engineers must make effort to find alternative ways to increase the power density of the kind of multiactuator robotic limbs necessary for legged robotic systems.

The motivation to progress the state of the art in legged vehicles comes mainly from their impressive potential in biology. Assistive robots must be capable of navigating the same environments that people do- environments which are built to be traversed by legs. Wheeled systems are less capable of overcoming obstacles which do not present a problem for legs. Uneven terrain is among the foremost, even indoors, in the form of steps or debris. Many impressive robotic systems have been developed with capabilities to traverse uneven terrain, but none is as powerful and/or efficient as their biological counterparts. Robotic systems such as BigDog [54], and Honda's Asimo [23] are capable of dynamic balancing and traversing rugged terrain. These capabilities come at the cost of energy efficiency. Other robots such as Cornell University's Ranger [7] have extreme energy economy but are not designed for rugged terrain navigation.

Many groups are attempting to build robotic systems that have both high performance and energy efficiency. For example, Hurst et al. [38] has developed an efficient bipedal robot, Atrias, that is capable of navigating rough terrain. They designed the passive dynamics of their robot to approximate the Spring Loaded Inverted Pendulum model. Their control approach takes advantage of this well-studied dynamic model to

create stable and efficient robot gait [39, 70]. It also focuses on utilizing the natural dynamics of the robot in a way that will help them save energy step to step [1, 22]. Nonetheless, a weakness of the leg structure was recently discovered: the robot loses considerable power due to mechanical antagonism due to its pantograph leg mechanism [2].

The MIT cheetah robot created by Kim et al. has also made significant progress towards efficient navigation of rugged terrain [61, 62]. The mechanical design of the legs is a planar serial robot. The group designed the leg to be stiff, and to have low impedance. To achieve this they designed a high torque density motor and coupled it with a small gear reduction in comparison to traditional robotic actuators. They implemented a system to regenerate energy from braking through the motors. Efficiently regenerating power back into the batteries of a DC motor actuation system is difficult, at best. Kim et al., with custom designed motors, achieve the braking process with an efficiency of 63 percent. Other systems have used compliance as a means of storing and releasing energy during gait. For example, animals have been shown to use tendons in this manner [4]. While some amount of braking energy during gait may be unavoidable, the authors approach is to circumvent the problem by specialization of the design of the limb.

The author shows the viability of this approach in a series of published works [10, 11]. In the first paper, a hybrid serial-parallel actuation approach is introduced. In the second paper, Cahill et al. show that the geometry of spatial mechanism may be tuned to a given data set, sufficiently eliminating losses to antagonistic power [11]. Designing the kinematics to avoid actuator antagonism in this way avoids the inefficiencies inherent to energy regeneration, whether by DC generators or springs. Avoiding antagonism not only reduces the need for regeneration; in many cases it also reduces

the peak motor powers associated with a task. This is because, like any thermodynamic system, power-in equals power-out. Assuming for the moment that the limb is being used to produce positive work at the end-effector, when the mechanical antagonism phenomenon occurs, one or more actuators has become a power *sink*, instead of a power *source*. This means that the remaining power source actuators must provide the sum of the output energy demanded by the end effector *and* the energy being sunk into the antagonistic actuators. Depending on the task, this may dramatically increase the value of peak instantaneous power and current required. This, in turn, will lead to larger motors and more expensive power inverters.

This chapter has three parts. The first part, described in section 5.2, uses a visual method to introduce the phenomenon and to show how the kinematics of the limb, as well as the directionality of the task, can cause mechanical antagonism. Next, in section 5.3, we derive a power-optimal pseudoinverse for over-actuated systems which minimizes antagonistic motor power for both statically determinate and statically indeterminate systems. Finally, in section 5.4, we validate the power-optimal pseudoinverse using a model of a three DoF over-actuated planar serial manipulator.

5.2 Mechanical Antagonism Visualized at the End Effector

This section gives a series of steps which help to understand the mechanical antagonism phenomenon in an intuitive and visual way. We introduce a planar two DoF serial robotic limb. This planar limb can move its end effector linearly in any direction on the plane. Also, it is capable of applying a force in any direction on the plane but cannot apply a moment. This robot limb is shown in Figure 46. In order to visualize the phenomenon, the velocity space and the force space of the end

effector, and the screws/wrenches which form the basis for these vector subspaces, are shown. Finally, these two spaces combine to form the force-velocity vector space in IR^4. This vector subspace is split into contiguous motor power regions distinguished by the sign of the power generated at each motor. Only a small subset of these regions are non-antagonistic. The goal is to reduce the multiactuator robot limb's power consumption by avoiding the regions with poor efficiency, or by designing the robot to fit the desired task.

5.2.1 Visualizing the Velocity Space

The velocity space of a robot limb is the column space of the leg Jacobian. It is a vector space that contains all of the velocity vectors that the end effector is theoretically capable of achieving (it does not account for any actuator limitations). And each of the columns corresponds to a specific actuator (column one with motor one, column two with actuator two, etc.) This space can be visualized for the example we have chosen. The first column of the Jacobian is the two-dimensional screw associated with motor one (notated as $\$_{m_1}$). It represents the end effector velocity corresponding to a unit input from motor one and zero input from motor two (e.g., motor two fixed). This vector is perpendicular to a line drawn between joint one and the end effector (see Figure 46). Similarly the second column in the Jacobian (notated as $\$_{m_2}$) describes the velocity of the end effector when motor two is excited with a unit velocity and motor one is held fixed. This vector is perpendicular to a line between the end effector and joint two. These two vectors are basis vectors for the motor velocity space, meaning a velocity of any magnitude in any direction in the

plane can be formed via a linear combination of the two:

$$\$_{EE} = \omega_{m_1} \$_{m_1} + \omega_{m_2} \$_{m_2}, \tag{5.1}$$

where ω_{m_1} and ω_{m_2} are the velocity magnitudes of motors one and two respectively. Also, $\$_{EE}$ is the velocity of the end effector. Equation 5.1 is just a rewritten form of the well-known Jacobian equation:

$$\$_{EE} = J\,\omega. \tag{5.2}$$

where ω is a 2x1 vector of the motor velocities, and J is the Jacobian.

A "region" of the velocity space is a contiguous set of vectors in which the signs of the motor velocities in equation 5.1 do not change. Figure 47 shows the four important regions of the velocity space for this leg, and Table 8 details what each region represents. For example region one is the set of end effector directions that correspond to motor one and motor two moving in a positive direction (see row one of table 8). The edges of each region correspond to zero crossings of one motor or the other. For example the edge between region one and region two is along the vector $\$_{m_1}$. This vector represents a zero crossing for motor two. These velocity regions are an important component in determining which directions a given robot's antagonistic power regions will lie. To determine the sign of the motor coordinates in each region, one must look at the direction of the vectors $\$_{m_1}$ and $\$_{m_2}$. For example, a velocity parallel with the vector $\$_{m_1}$ will by definition require a positive velocity at motor one

(and a zero velocity from motor two). The sign of motor one's velocity is positive in any direction that is not separated from $\$_{m_1}$ by a motor one zero crossing (zero crossings for motor one happen at $\pm\,\$_{m_2}$). This means motor one has a positive velocity value in regions one and two, and a negative value in regions three and four. Following this same line of thought, motor two will have a positive velocity in regions one and four and a negative value in regions two and three.

Table 8. Velocity Regions

	Sign: ω_{m_1}	Sign: ω_{m_2}
Region One	+	+
Region Two	+	-
Region Three	-	-
Region Four	-	+

It may be noted that the four regions defined in this discussion are a mapping of the four quadrants of a two-dimensional Cartesian coordinate system having motor one velocity on one axis and motor two velocity on the other. The Jacobian maps these four coordinates onto the end effector space. This discussion is meant to give an intuitive and visual understanding of how that mapping works, because this mapping has a large impact on the efficiency of the system for a given task. In the next section, the visualization of the force space will be detailed.

5.2.2 Visualizing the Force Space

The force space of the leg can be visualized much in the same way as the velocity space. It is the column space of the inverse transpose of the leg Jacobian (assuming no energy loss). To begin visualizing this space, imagine a unit torque is applied through motor one but motor two is free. Since no moment can be transferred through

joint two or the end effector, the force contribution of motor one ($\$'_{m_1}$) can be drawn starting at the end effector and pointing away from joint two (represented by the red dotted line in Figure 46). The direction of this vector is determined by the assigned direction of positive torque about motor one. The force contribution of motor two can be drawn in the same way. Its line of action must lie on a line that contains the end effector and joint one (it is shown as a blue dotted line in Figure 46. (An interesting side note: $\$'_{m_2}$ is drawn perpendicular to $\$_{m_1}$ and $\$'_{m_1}$ is perpendicular to $\$_{m_2}$.) These are the basis vectors of the force space. Similar to equation 5.1, the end effector output force ($\$'_{EE}$) can be written as:

$$\$'_{EE} = \tau_{m_1} \$'_{m_1} + \tau_{m_2} \$'_{m_2}. \tag{5.3}$$

In this case, $\$'_{EE}$ is a 2x1 vector containing the components of the end effector force. As in the above discussion equation 5.3 is a rewritten form of the well-known equation:

$$\$'_{EE} = (J^T)^{-1} \tau. \tag{5.4}$$

It should be noted here that the Jacobian must be invertible to write this equation. In this case, the matrix is square, and as long as the robot is not in a singular position, the Jacobian will be invertible. In section 5.3 we discuss over-actuated systems and their implications on antagonistic power. Figure 48 shows the four regions of the velocity space for this leg, and Table 9 details what each region represents. A region in the force space, similar to those of the velocity space, corresponds to a contiguous set

Table 9. Force Regions

	Sign: τ_{m_1}	Sign: τ_{m_2}
Region One	+	+
Region Two	+	-
Region Three	-	-
Region Four	-	+

of end effector forces which have a particular combination of motor torque signs. For example, region one encompasses all the end effector force directions which correspond to both motors having positive torque values. The signs of the motor force change at zero crossings. This means that the edges of each region represent end effector force directions in which one motor contributes zero force. These zero crossing directions correspond to the vectors $\$'_{m_1}$ and $\$'_{m_2}$, shown in figure 46 - $\$'_{m_1}$ being a zero crossing for motor two and $\$'_{m_2}$ being a zero crossing for motor one. In this way the columns of the inverse transpose leg Jacobian create the edges of the four regions of the leg force space. Determining the sign of the motor forces in these regions can be accomplished in a similar method as described in section 5.2.1. By definition the sign of the torque value at motor one is positive when end effector force is pointing in the direction of $\$'_{m_1}$. This sign will not change as the angle between the end effector force and $\$'_{m_1}$ increases unless a zero crossing for motor one occurs. These zero crossings happen when the end effector force is aligned with $\pm\$'_{m_2}$. Using this method, it becomes apparent that regions one and two correspond to positive torque values for motor one. Therefore, regions three and four have negative torque values for motor one. In a similar manner, motor two has positive values in regions one and four and a negative value in regions two and three.

5.2.3 The Force-Velocity Space

Since actuators can apply forces and velocities independently, the force space and the velocity space can be combined into a four-dimensional force-velocity space. This force-velocity space is the space that is of interest. One point in this space specifies the velocity direction and magnitude, and the force direction and magnitude. It is useful to define this space for the discussion. There are two coordinate frames of interest. The global coordinate frame χ has the end effector forces and velocities:

$$\chi = \begin{bmatrix} \$_{EE} \\ \$'_{EE} \end{bmatrix}. \tag{5.5}$$

The other frame of interest is the local or motor coordinate frame ξ, which has ω and τ:

$$\xi = \begin{bmatrix} \omega \\ \\ \tau \end{bmatrix} = \begin{bmatrix} \omega_{m_1} \\ \omega_{m_2} \\ \tau_{m_1} \\ \tau_{m_2} \end{bmatrix} \tag{5.6}$$

The Jacobian can be used as a coordinate transformation between the two coordinate systems:

$$\chi = \begin{bmatrix} J\,\omega \\ \\ (J^T)^{-1}\,\tau \end{bmatrix} \tag{5.7}$$

One transformation matrix ($\mathcal{J}$) can be built from the Jacobians which maps the forces and velocities from actuator frame to the end effector frame.

$$\mathcal{J} = \begin{bmatrix} J & \underline{0} \\ \\ \underline{0} & (J^T)^{-1} \end{bmatrix}. \tag{5.8}$$

Which allows:

$$\chi = \mathcal{J}\,\xi. \tag{5.9}$$

The force-velocity space of the leg contains all the necessary information to know which regions are antagonistic and which regions are not. To define the antagonistic power regions, power must be formulated.

5.2.4 Power

Mechanical power (P) is a function of only velocity and force. Due to this, every point in the force-velocity space is associated with a specific overall power output. This power output can be formulated as:

$$P = \$_{EE}^{T} \; \$'_{EE} \tag{5.10}$$

Where P is the mechanical power output at the end effector. Since we are assuming no losses, power-in equals power-out, and the sum of the individual motor powers (P_{m_1} and P_{m_2}) is equal to P as well.

$$P = \tau^{T}\omega = P_{m_1} + P_{m_2} \tag{5.11}$$

Note that the sign of the motor powers will be determined by the product of the motor torques and velocity. We have just shown that these are a function of the direction of desired end effector torque and velocity. Imagine, for example, that the desired power at the end effector is positive and in a direction that causes the power at motor one to be negative. If this is the case, motor two will have to produce power equivalent to the sum of the magnitude of the power at motor one *and* the end effector power requirement. This is the phenomenon of interest - mechanical antagonism - and it is undesirable for more than one reason: firstly, because it will raise the peak power requirement of the task, resulting in the need for larger motors in the design; secondly,

because inevitably, energy will be lost to heat by the motor which is performing braking (not to mention the heat losses due to the added work on the positive power motor). One could argue that the energy could be regenerated through the motor and stored back into the battery for later use, but the regeneration process is currently very inefficient, and much of the energy will be wasted. Regardless of this, reducing the antagonistic power will increase the power density of the system, which is a very valuable result considering the slow growth of power technology. The next section will detail which regions in the torque velocity space are associated with mechanical antagonism.

5.2.5 Antagonistic Power Regions

Antagonistic power regions are those which the sign of the product of torque and velocity for one motor does not match the other(s). Since, as noted above, the torque and velocity of an actuator are generally independent, every combination of the regions from Table 8 and 9 is possible. This means there are 16 distinct regions in the force-velocity space. Since even for this simple example the space is four-dimensional, a figure displaying the regions directly is too complicated. Instead the 16 regions are displayed in Table 10. Each row corresponds to a region of Figure 47, and defines which of the regions in Figure 48 are antagonistic. For example, if one desired to move the end effector vertically downward, to avoid mechanical antagonism, a force may only be applied in the direction of region 2 of Figure 48.

It should be noted that only four of these regions are "positive power regions", or regions which correspond to positive power output for both actuators. Of the twelve other regions eight correspond with antagonistic power regions, and four correspond

Table 10. Power regions

Velocity	Force			
	Region 1	Region 2	Region 3	Region 4
Region 1				
P_{m_1}	+	+	-	-
P_{m_2}	+	-	-	+
Region 2				
P_{m_1}	+	+	-	-
P_{m_2}	-	+	+	-
Region 3				
P_{m_1}	-	-	+	+
P_{m_2}	-	+	+	-
Region 4				
P_{m_1}	-	-	+	+
P_{m_2}	+	-	-	+

with both motors braking. If the end effector is used to dampen energy out of the system then the braking regions would be desirable in this case. Otherwise the goal would be to keep the robot operating in the "positive power regions". This may be accomplished by changing the task to fit the robot or, by designing the kinematics of the robot especially for the intended task. The latter is the approach taken by the author and published in a 2016 paper [10].

This completes the detailed force-velocity analysis of this robot arm. Every robot limb has a force-velocity space. Some of these spaces are more complex than others. For every degree of freedom added to the robot two degrees of freedom are added to the force-velocity space. For example, imagine a third actuator was added to the end effector of the above example. This would allow the arm to control not only linear velocities and forces but also the rotational velocity and torque output. The

force-space would be six dimensional, with eight (2^3) force and eight velocity regions, making 64 distinct power regions, of which eight are non-antagonistic positive power regions. A robot arm that operates in six-dimensional space (three linear dimensions and three rotational dimensions) has a 12-dimensional force-velocity space with 64 (2^6) force and 64 velocity regions. This means there are 4096 distinct regions but *only 64 of these are non-antagonistic positive power regions.*

5.3 Power-Optimal Pseudoinverses for Redundant Manipulators

It is of interest to extend this analysis to over-actuated robotic systems. These systems have non-square Jacobian matrices. Two cases will be presented here. Case one is a system which is statically determinate and has one and only one solution for actuator torque given the desired end-effector torque. This case has infinite solutions for actuator velocities given desired end-effector velocities. This case is applicable to redundantly actuated serial manipulators. Case two is a system which is statically indeterminate, meaning there are infinite combinations of actuator forces which produce the desired motor torque but there is only one solution for actuator velocity. This case is applicable to redundantly actuated parallel robots. Both of these cases will be discussed here along with a power-optimal pseudoinverse

5.3.1 Case 1: Statically Determinate Manipulators

In the case of statically determinate manipulators, such as redundant serial limbs, the J will have dimension $n \times m$, where $m > n$. Equations 5.2 still holds, but now J is not invertible so equation 5.4 does not. There are infinite solutions to Equation 5.2

for ω given $\$_{EE}$. The classical solution to this problem is the so-called psuedoiverse of the Jacobian:

$$J^{\dagger} = J^T \left(J\, J^T\right)^{-1}. \tag{5.12}$$

This solution minimizes the sum of squares of the motor velocities. In light of mechanical antagonism, this solution does not seem ideal. Small motor velocities are not as important as the reduction of antagonistic motor power. The ideal solution therefore will minimize the sum of square motor powers. Fortunately, an analytical solution exists which accomplishes this goal. We formulate the problem as follows:

$$Minimize: \; P^T P$$

$$Such\ That: \; \$_{EE} = J\omega$$

Where P is a vector of motor powers,

$$P = diag(\tau)\omega. \tag{5.13}$$

The power vector is a product of a matrix, $diag(\tau)$ the diagonal of the motor torques and the motor velocity vector. We will give the diagonal torque matrix a more convenient convention, T, for the remainder of the analysis. The equation for sum of square power can be written as:

$$P^T P = \omega^T \mathrm{T}^2 \omega \tag{5.14}$$

Next we solve the optimization problem, starting by constructing the Lagrangian:

$$L = \frac{1}{2}\omega^T \mathrm{T}^2 \omega + \lambda^T (J\,\omega - \$_{EE}) \tag{5.15}$$

The minimum of equation 5.15 can be found by equating the gradient of the function to zero. The gradient of this function is as follows:

$$\nabla L = \begin{bmatrix} \\ \frac{\delta L}{\delta \omega} \\ \\ \\ \frac{\delta L}{\delta \lambda} \end{bmatrix} = \bar{0} \tag{5.16}$$

Where:

$$\frac{\delta L}{\delta \omega} = \mathrm{T}^2 \omega + J^T \lambda, \tag{5.17}$$

and

$$\frac{\delta L}{\delta \lambda} = J \omega - V. \tag{5.18}$$

Equation 5.17 may be solved for ω:

$$\omega = -\mathrm{T}^{-2} J^T \lambda. \tag{5.19}$$

Since the square of a diagonal matrix is a positive definite matrix T^2, which will always be invertible. Equation 5.19 may now be substituted into equation 5.18 and solved for λ giving:

$$\lambda = -(J \mathrm{T}^{-2} J^T)^{-1} \$_{EE}. \tag{5.20}$$

If J has full row rank, the quantity $(J \mathrm{T}^{-2} J^T)$ will be invertible. The power-optimal

motor joint velocity ($\hat{\omega}$) results from substituting equation 6.14 into 5.19:

$$\hat{\omega} = \mathrm{T}^{-2}\, J^T\, (J\, \mathrm{T}^{-2}\, J^T)^{-1}\, \$_{EE} \tag{5.21}$$

The power-optimal pseudoinverse ($\hat{J}$) results from Equation 5.21:

$$\hat{J} = \mathrm{T}^{-2}\, J^T\, (J\, \mathrm{T}^{-2}\, J^T)^{-1}. \tag{5.22}$$

Using this pseudoinverse of the nonsquare Jacobian matrix will guarantee that antagonistic power is minimized, given a desired end effector velocity and an expected motor torque matrix. Note that T^2 is the square of a diagonal matrix. This means the objective here is positive semi-definite and, since the constraints are linear, the problem is convex.

5.3.2 Case 2: Statically Indeterminate Manipulators

Case two differs from case one in that for statically indeterminate manipulators, such as redundant parallel robots, there are infinite combinations of actuator torques (τ) which produce the desired end effector torque ($\$'_{EE}$). The formulation of the

optimization problem becomes:

$$Minimize: \ P^T P$$

$$Such\ That: \ \$'_{EE} = J'\tau$$

Each column of the J' matrix is the non-normalized wrench associated with the corresponding actuator. Davidson and Hunt spend a great deal of time defining this matrix and its relation to the velocity relationship in the text, (see in particular table 8.2 of [17]). However, these relationships fail when redundant actuators are concerned. We propose here a pseudoinverse of J' which minimizes motor power. The process involved is nearly identical to the process shown in detail in section 5.3.1. The difference is that power is formulated as follows:

$$P = diag(\omega)\tau \tag{5.23}$$

Similar to section 5.3.1, we give the diagonal velocity matrix $diag(\omega)$ new convention: W. The Lagrangian of this problem then takes the same form as the one in the previous section:

$$L = \frac{1}{2}\tau^T \mathrm{W}^2 \tau + \lambda^T (J'\,\tau - \$'_{EE}). \tag{5.24}$$

Following the same steps as above, the power-optimal torque vector is:

$$\hat{\tau} = \mathrm{W}^{-2}\, J'^T\, (J'\,\mathrm{W}^{-2}\, J'^T)^{-1}\, \$'_{EE}. \tag{5.25}$$

It follows that the power-optimal pseudoinverse for redundant parallel manipulators is:

$$\hat{J}' = \mathrm{W}^{-2}\, J'^T\, (J'\,\mathrm{W}^{-2}\, J'^T)^{-1} \tag{5.26}$$

This solution is guaranteed to be the global solution to the problem because the problem is convex. This is due, as previously stated, to the fact that the weighting matrix - in this case W^2 - is a positive semi-definite matrix and the constraints are linear.

5.4 Validation Using Redundant Serial Limb

In effort to verify the optimal solutions presented above, we will consider three DoF planar serial robot arm (see Figure 49). Similar to section 5.2, the robot may move in two-dimensions and apply force in two-dimensions. Torque at the end effector is constrained to zero. This means the system has one redundant DoF.

The Jacobian of this manipulator has dimension two by three. Three actuators produce velocities in a three dimensional space which projects onto the two-dimensional end effector velocity space. Given a desired end effector velocity, infinite solutions exist in the three dimensional actuator velocity space which can create the desired velocity. Section 5.3.1 derives the solution within the infinite possibilities which minimizes the sum of square power (or SSP).

As a case study, we chose arbitrary desired end effector force and velocity directions. Then we reduced the system of linear equations (from equation 5.2) to two equations which define the velocity of actuators one and two as a function of actuator three velocity. Finally, we made a plot of the SSP vs a range of actuator three velocities which satisfy the conditions. Figure 50 shows this plot, including the location of the classical minimal velocity pseudoinverse and the power optimal pseudoinverse solutions. It should be noted that the power optimal solution lies at the minimum of the function while the classical solution lies at a point with smaller velocity, but larger SSP.

Since the actuator velocity space of this robot in three dimensional space, visualization of the solution is possible. The power optimal pseudoinverse is a three by two matrix. The column space of this matrix makes a plane in the actuator velocity space. This plane is shown in Figure 51. This plane represents the power-optimal,

two-dimensional mapping from the end effector velocity space to the actuator velocity space, given a desired end effector force. In other words, given any 2D desired end effector velocity vector, the 3D actuator velocity vector which requires minimum power will lie on the plane shown in Figure 51.

Using this mapping, mechanical antagonism may now be addressed. Since the system is statically determinate, there is only one solution for the actuator torques. If the robot arm is to do positive work on the environment, a non-antagonistic velocity vector must have the same sign for each coordinate as the force vector (if motor one is applying negative torque, it should be moving in the negative direction). This means that only one octant of the actuator velocity space is non-antagonistic: the same octant in which the actuator torque lies. For the example problem, the force vector chosen was $F_{EE} = [0 - 1]^T$, which requires a negative torque about each joint. Therefore, the velocity vector must lie in the octant with negative values for each coordinate. In Figure 51 the portion of the plane which satisfies this requirement is green in color.

A similar figure has been made for the plane which results from the classical pseudoinverse solution. It should be noted that this plane's intersection with the non-antagonistic velocity quadrant is smaller than that of the optimal power solution.

As a final comparison between the presented and the classical approaches, we plot the percent reduction in SSP as a function of desired velocity angle (see Figure 53). The desired torque vector is the same as before: $[0 \ -1]^T$. It can be seen from the figure that the maximum reduction in SSP is almost seventy percent. Also, there are two directions for which there is no change in SSP between the two solutions. These directions lie on the line of intersection of the planes plotted in Figures 51 and 52.

5.5 Conclusions

In this three-part chapter, we first rigorously developed the force-velocity space of a multiactuator robotic arm. We detail the process for a two DoF serial arm to help the reader build intuition for how antagonism develops in multiactuator systems. We emphasized the importance of the directions of the desired end effector force and velocity, and we showed how the kinematics define which directions are efficient and which are not. Antagonistic actuator work is a phenomenon which must not be ignored in the design process of a robotic limb. Designers should know that the directional aspect of a task will determine which force-velocity region it will fall into, and therefore how much energy is lost to mechanical antagonism in the joints. The kinematics of the robot arm have a direct effect on which directions are efficient and which are not. We suggest designing the kinematics of the arm to avoid motor antagonism. An optimization routine like the one in [11] may be implemented to do so.

In the second part, we looked at over-actuated systems, which have non-invertible Jacobians. We show how the added degrees of freedom may be utilized to reduce the effect of mechanical antagonism. This led to analytical solutions which minimize the sum of squares of the mechanical work exerted by the joints in both parallel and serial, redundant mulitactuator robotic arms. We argue that this pseudoinverse is superior to the classical velocity-optimal one. Minimizing actuator work will extend battery life of mobile robots and, if the actuator is a DC motor, reduce the risk of motor failure due to waste heat generated in the windings.

5.6 Future Work

While this chapter has focused on the phenomenon of mechanical antagonism and how to minimize it for redundant systems, there may be a more meaningful solution to the problem. Using an actuator model specific to the robot in question would allow the minimization of electrical power. This model could take into account losses due to inertia, actuator inefficiency, regenerative capabilities, and so on. For this chapter, a general solution has been presented which minimizes mechanical antagonism. More specific solutions could be attained minimizing, for example, electrical power consumption, wasted heat in motor windings, or the fuel consumption of the hydraulic pump. If future end effector trajectory is known, the future energy consumption may be considered. In this way, the optimal joint trajectory in the position and velocity space could be determined. Since an abundance of solutions may exist which could be considered the best for one reason or another, we chose to present a simple general solution here. We leave it to future work to further explore this problem and compare this general solution to more specific ones.

While this line of research would be interesting, redundant manipulators may not be the ideal solution to reducing mechanical antagonism. Redundant manipulators are not often used in applications requiring high power to weight ratios. Legged robots, for example, have limbs that are typically under-actuated. This may be due to the fact that redundant actuators add weight and cost to the system. For systems that require high power density, we suggest using optimal design methods to reduce mechanical antagonism and other actuator specific inefficiencies (see Chapter 3).

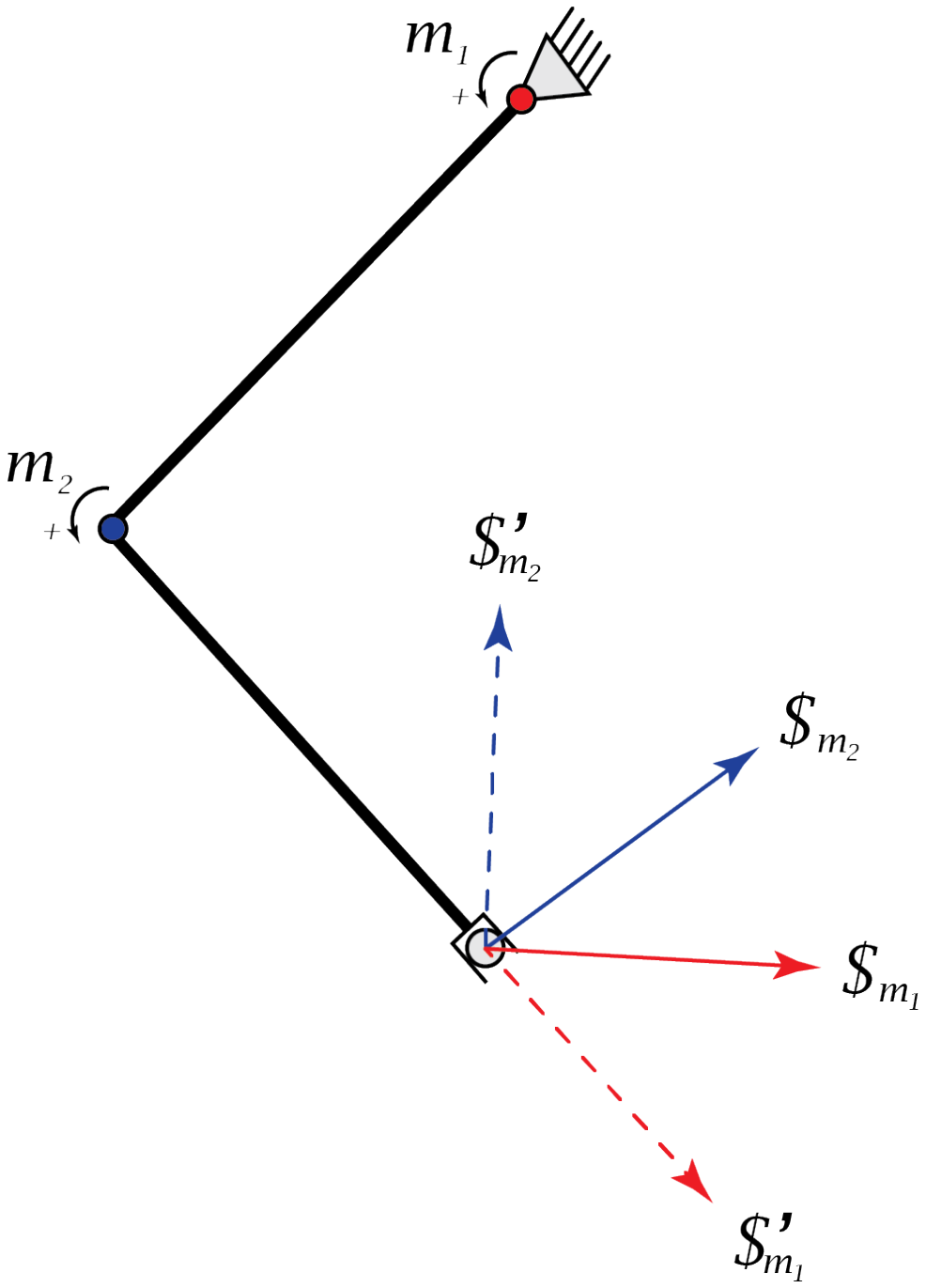

Figure 46. Diagram of the serial limb robot. Also shown are the basis vectors of the velocity and force space.

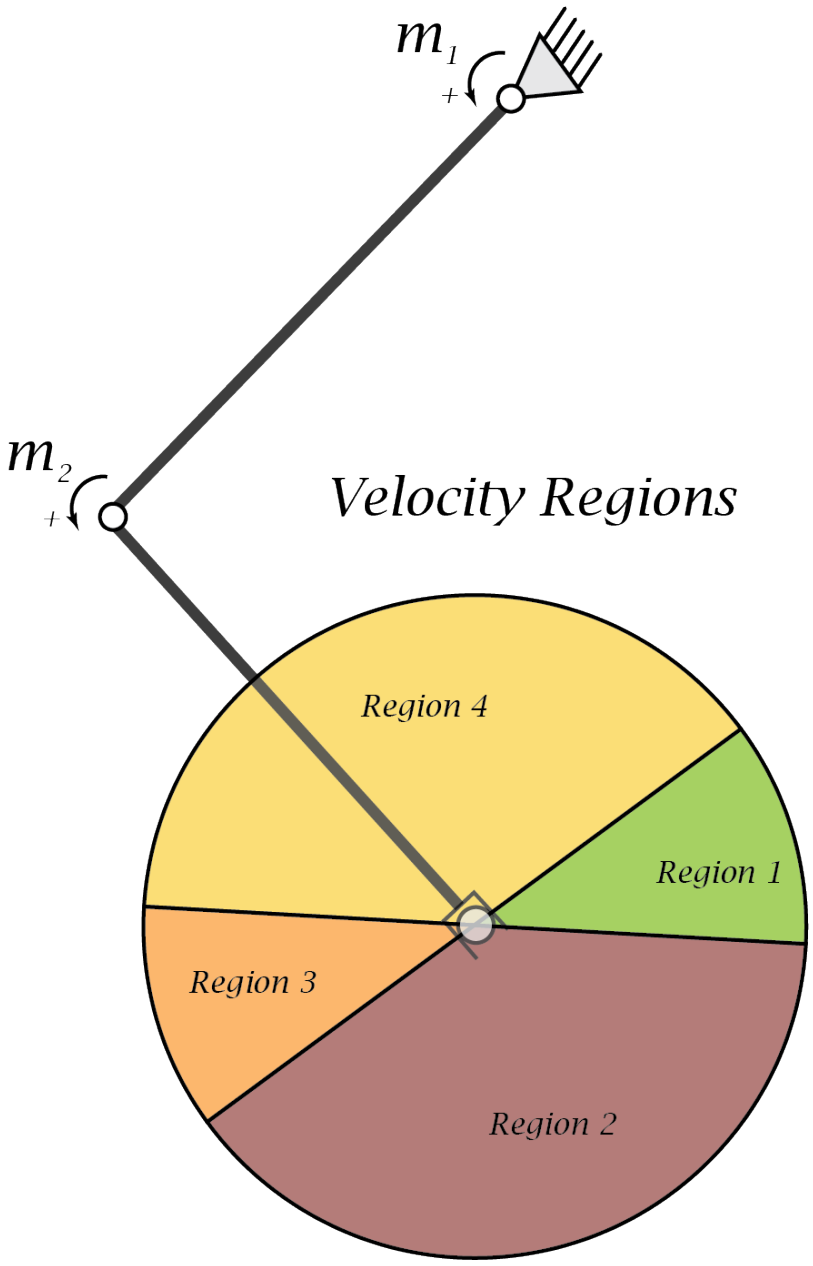

Figure 47. Visualizing the velocity space of the robot leg

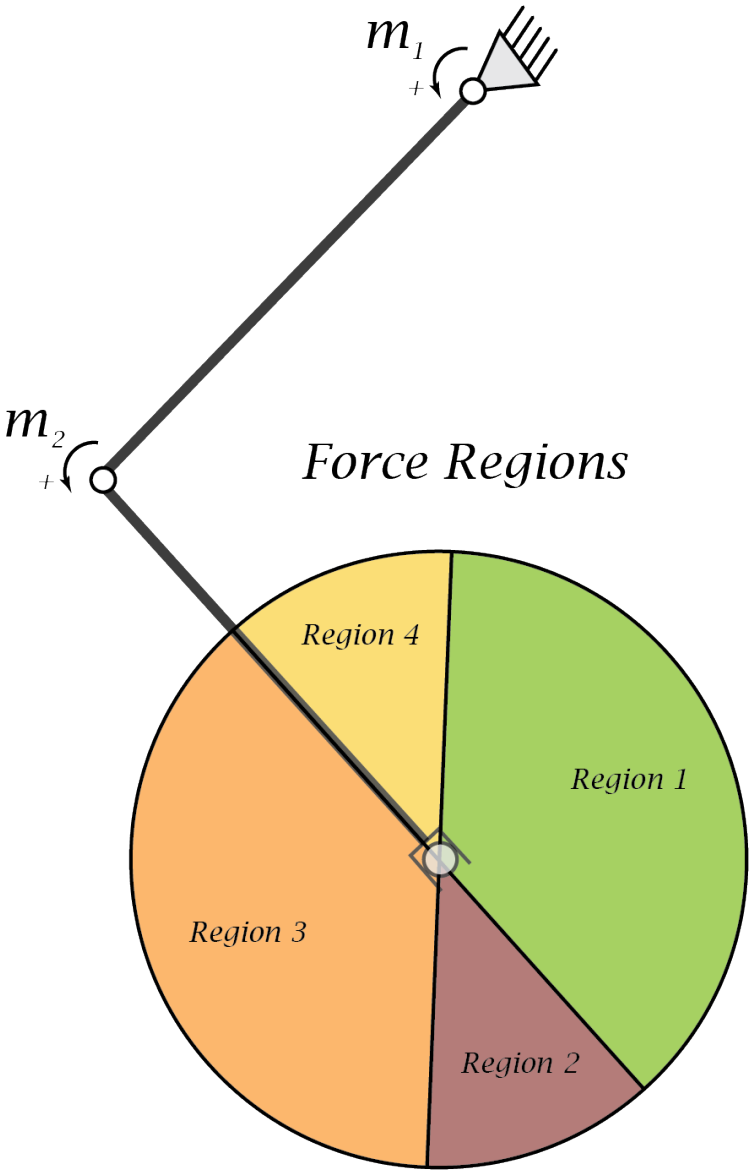

Figure 48. Visualizing the Force Space of the robot leg

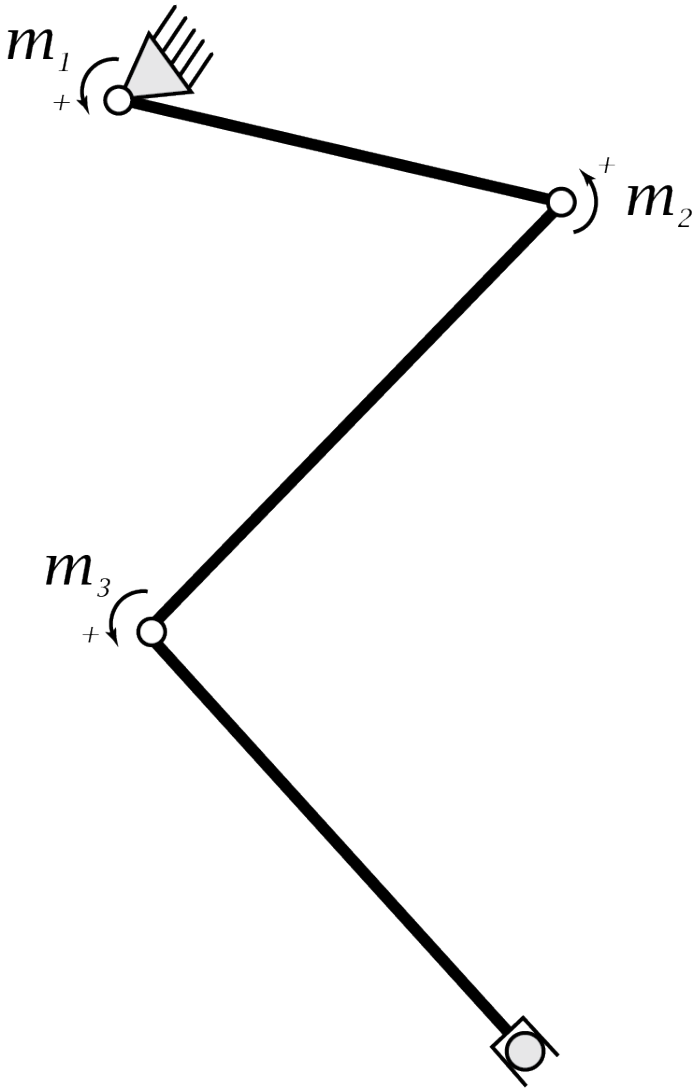

Figure 49. A visualization of the robot being modeled.

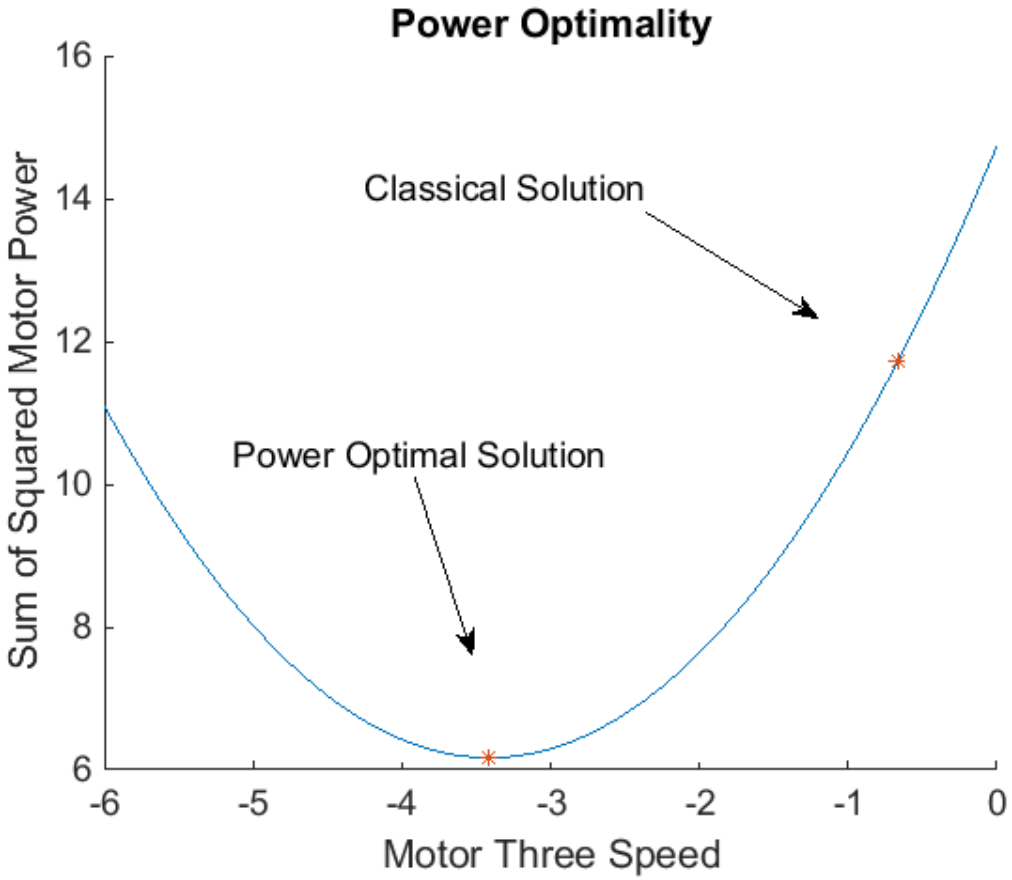

Figure 50. Plotting the cost function over the constrained problem space.

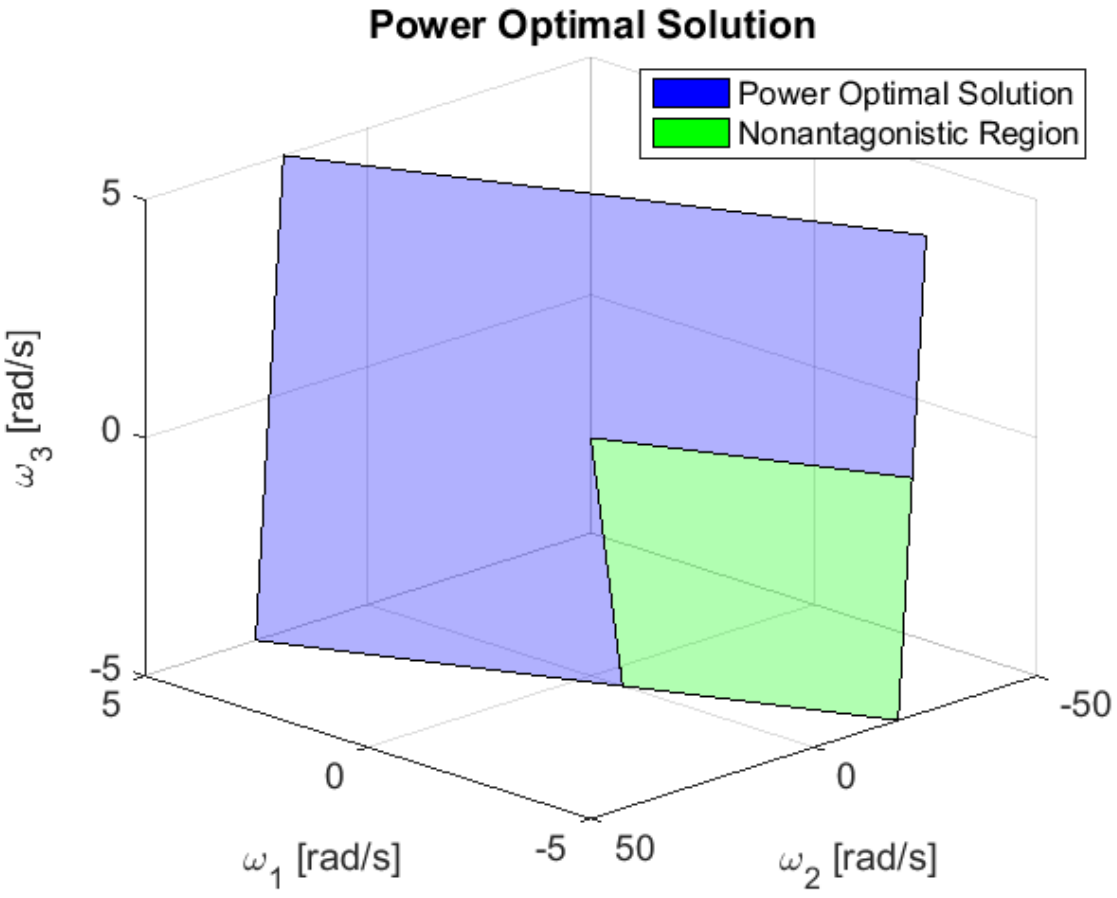

Figure 51. Optimal power solutions lie on a plane in the three-dimensional motor velocity space.

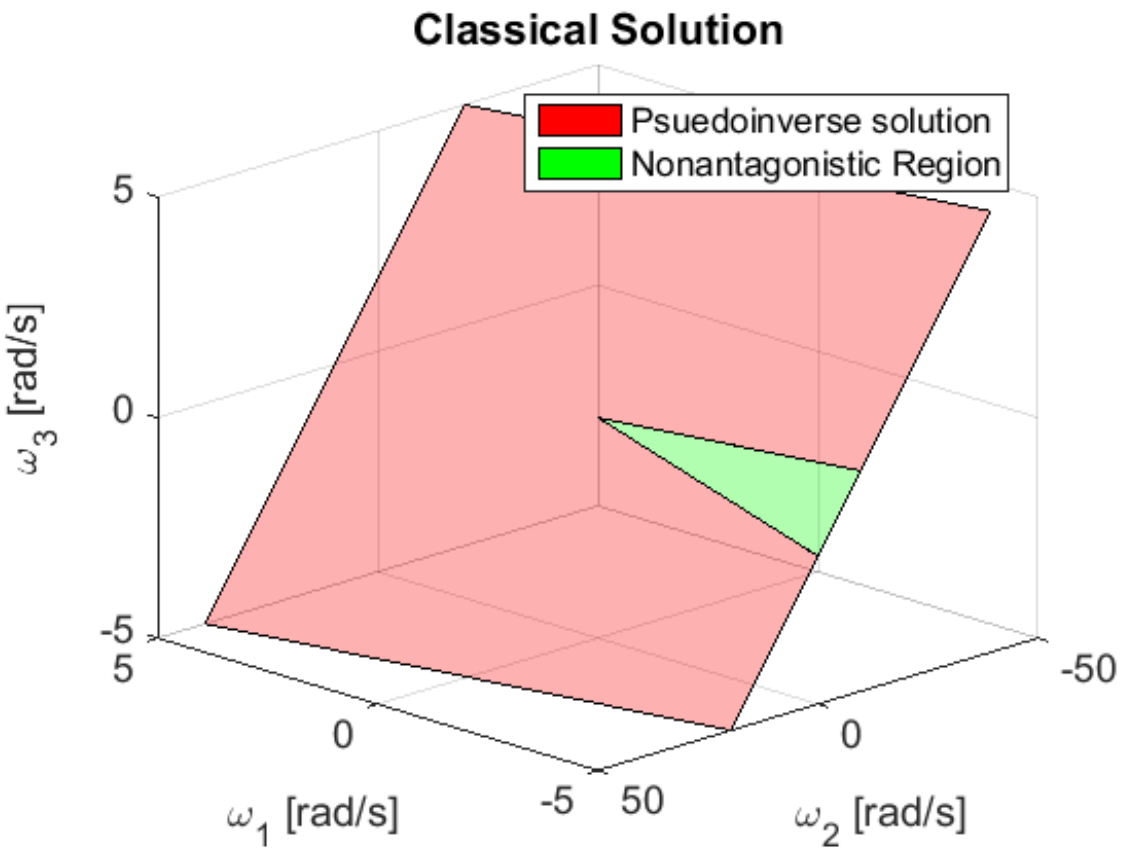

Figure 52. Optimal velocity solutions lie on a plane in the three-dimensional motor velocity space.

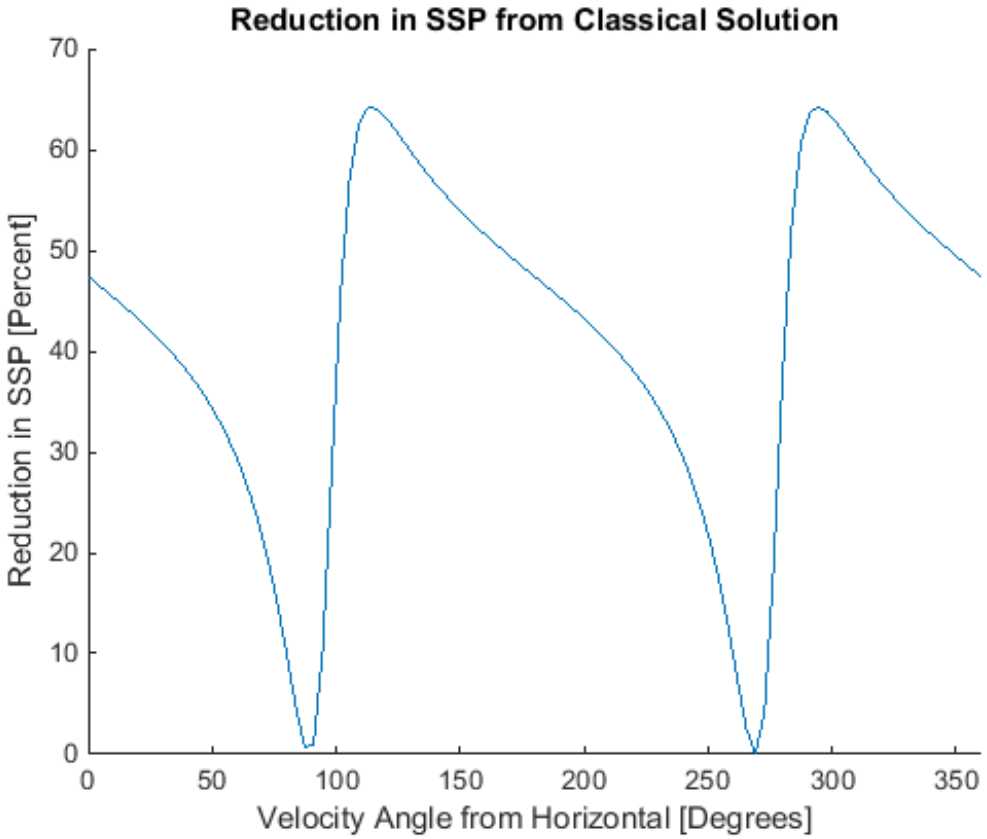

Figure 53. Reduction in SSP versus direction of desired end effector velocity.

Chapter 6

OPTIMAL STIFFNESS DESIGN FOR AN EXHAUSTIVE PARALLEL COMPLIANCE MATRIX (ECPM) IN MULTIACTUATOR ROBOTIC LIMBS

6.1 Introduction

The motivation of this chapter is to maximize power performance of robotic limbs. In order for the results to be meaningful we use a realistic loading scenario and an accurate motor model. For the loading scenario, we use a gait trajectory including both force and velocity which was measured empirically from a biological quadruped. This gives us a realistic loading scenario to test the merits of our optimal design tool. Using this gait trajectory, we formulate a fully-dynamic motor torque profile including both the inertial effects from the acceleration profile and the torque required by the ground reaction force profile of the gait trajectory. This motor torque profile varies as a function of the stiffness and offset values of the parallel spring matrix. The stiffness-dependent torque profile affects the motor power required for the task. After presenting the motor model, we carefully formulate three different cost functions, based on the motor torque and power profiles, which we use to optimize the design of the spring stiffness matrix. We then compare and contrast the results of these solutions. The springs prove to be very capable of reducing motor effort. Specifically, we show that including parallel springs in the design reduced the sum of squared power by up to 47 percent, along with up to 39 percent reduction in peak motor power, up to 55 percent reduction in sum of square current and up to 41 percent reduction in peak current. Each performance metric performed better in some aspects than the others, and the choice between them will depend on the designers' priorities.

While our exhaustive parallel spring matrix concept is novel, compliance in robotic actuators is not. There have been many instances where adding compliance to the design of an actuator has proven to significantly and positively affect a wide range of performance parameters. One example of this is the Series Elastic Actuator (SEA),

i.e. an actuator with a compliant member added in series. This technology has been used extensively in legged robots such as Atrias [22] and Big Dog [54], and in wearable robotic systems [24, 26]. Hollander et al. showed that tuning the spring stiffness of a SEA can significantly reduce the peak power required at the motor [30]. Parallel compliance is another type of compliant actuation, less commonly used than serial compliance in actuator designs. Yang et al. incorporated this method in the design of ERNIE, a bipedal walking robot, and performed an experiment to see how spring stiffness affected power use [73]. In their experiment, gaits of varying speed were designed for a number of knee spring stiffnesses. For each spring stiffness gaits were designed by simultaneously optimizing gait trajectory and spring offsets. The paper does not rigorously detail their optimization process, but they use the average of the absolute value of motor power as a fitness metric throughout the paper. This fitness metric is similar to the SSP metric which we formulate in Section 6.5.1. It was shown that the average power required to walk could be decreased using a spring at speeds above .4 meters per second, and that the best spring stiffness increased with walking speed. They do not, as opposed to the method introduced here, find the optimal spring stiffness for each walking speed. In this chapter we rigorously develop an optimization method which, given a fixed task (e.g. walking at 3 mph, or any arbitrary task which can be modeled by a set of force and velocity trajectories), our method will find an optimal parallel stiffness which minimizes one of three fitness metrics which we suggest.

One sub-purpose of this chapter is to test different fitness metrics against each other. This need arises from the fact that, over the years, many different fitness metrics have been developed for optimizing performance of robotic systems driven by a DC motor. Many of these metrics have been suggested in work on the optimization

of transmission ratio for actuators driven by geard DC motors. This chapter draws on several strategies from this body of research. Following is an overview of the major papers which are relevant to our work. The classical result was introduced by Pasch and Seering [49]. The performance metric they suggest is output acceleration for a given amplitude of actuator effort using an inertial load. The result is often referred to as the principle of inertia matching, because they find that the optimal gear ratio equivalates the inertial load of the motor with the inertial load of the task. Chen and Tsai expand the inertia match principle to multi-actuator robotic devices [13, 12]. The authors choose a fitness metric, the maximum acceleration capacity, which represents the spatial acceleration capacity of a geared robotic arm with multiple degrees of freedom. The maximum acceleration capacity is proportional to the volume of the acceleration ellipsoid. Van de Straete et al. described a method to select a feasible motor given a dynamic loading scenario [69, 68, 67]. They use a graphical approach to select the best performing motor (with feasibility constraints) for a dynamic loading scenario. They seek to minimize peak current which reduces the cost of the inverter. Rezazadeh and Hurst [57], continue the trend towards fully-dynamic, trajectory-based loading scenarios. Their performance metric of choice is the sum of waste energy (ohmic heating) in the motor windings. Overheating in the motor windings is a critical failure point for DC motors; significantly reducing ohmic heating will, in many cases, allow the designer to use a smaller motor. In this chapter we utilize a fully-dynamic trajectory-based loading scenario, but we test several different performance metrics. We formulate a metric very similar to Rezazadeh [57] in Section 6.5.3, one based roughly on Yang et al. [73] in Section 6.5.1, and a third which considers regenerative affects.

6.2 The Model: A Two DoF Planar Robot Arm with Parallel Springs

For this chapter we will design the optimal spring stiffness for a multiactuator robotic arm. For the sake of this chapter we implement our optimal design methods on a minimalist example: a two degree of freedom (DoF) planar serial manipulator. The robot limb has two actuated joints and two leg segments. Joints one and two will be the robot's hip and knee joints, respectively. And segments one and two will be the thigh and shank. The robot will not have a joint and segment corresponding to the ankle and foot. In the following subsections we introduce the robot's (well-known) kinematic and dynamic equations.

6.2.1 Kinematics

In order to design a robot limb the kinematic equations of the limb must be known. For this chapter we use a well-known robot model. If the reader is interested in seeing the specific kinematic equations for this robot arm, they are detailed in [63] for example. For this analysis, both the force and the velocity equations should be derived. A common and simple way to find these relationships is to differentiate the forward kinematics and formulate the Jacobian matrix (again, see [63] for a more detailed explanation). Once the force and velocity relationships have been derived, inertial effects must be modeled.

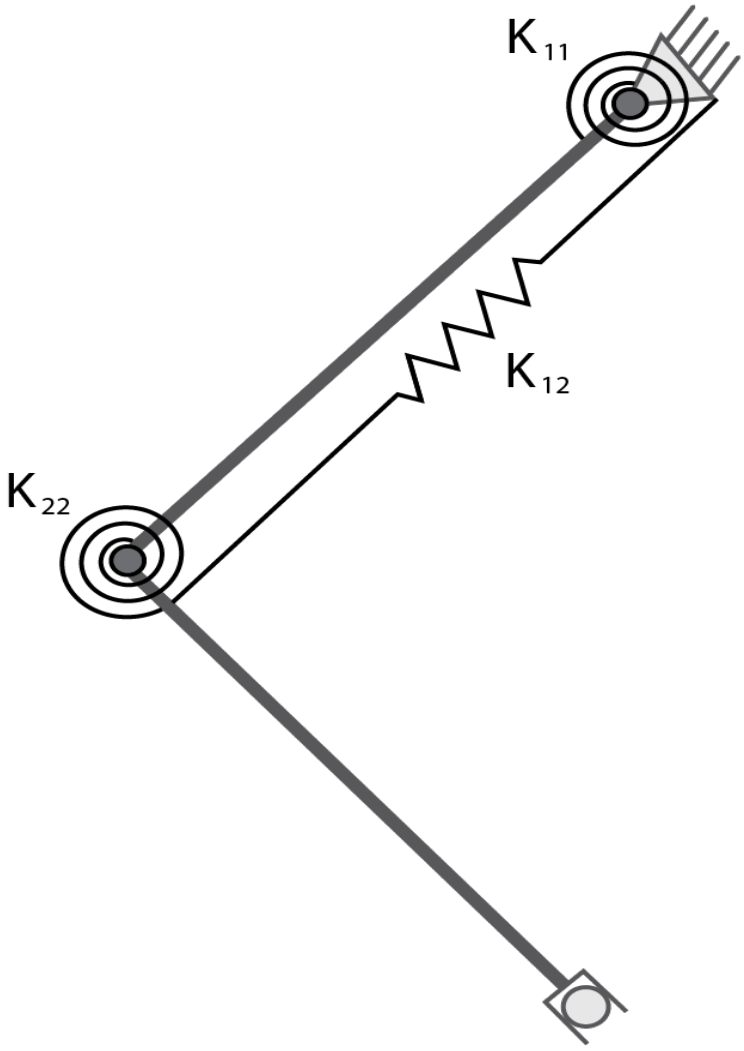

Figure 54. An illustration of the robot-spring model.

6.2.2 Dynamics

Inertial effects may play a significant role in the torques developed at the joints during legged gait. For the sake of this analysis, we choose to model each leg segment as a point mass located at its midpoint. The reader may refer to [63] for a more detailed description of how to model inertial effects in multiactuator robot limbs.

6.2.3 The Exhaustive Parallel Compliance Matrix

The key feature of this model is that we are incorporating a parallel spring across every joint and every possible joint combination. This thorough inclusion of every possible parallel spring stiffness is why we call the group of springs the Exhaustive Parallel Compliance Matrix (EPCM). Each of these springs has an associated stiffness (k_{ij}) and an offset (b_{ij}). Force is developed in each spring according to Hooks Law:

$$f_{ij} = -k_{ij}(\theta_{ij} - b_{ij}). \tag{6.1}$$

Where $\theta_{ij} = \theta_i + \theta_j$ when $i \neq j$. Otherwise, $\theta_{ii} = \theta_i$. In matrix form Equation 6.1 becomes:

$$\tau_s = -K\theta + Kb. \tag{6.2}$$

Replacing Kb with the vector β simplifies the equation to:

$$\tau_s = -K\theta + \beta. \tag{6.3}$$

For the planar case K is a 2x2 matrix, θ and β are 2x1 vectors. The values of matrix K and vector β will be the design variables for the optimization algorithm.

Figure 54 shows an actualization of the robot with an added parallel spring matrix. The spring stiffnesses in the figures are labeled with their respective index from the K matrix. It should be noted that a linear bi-articular spring, such as the one shown in Figure 54, would create the non-zero stiffness values in the off-diagonals of the stiffness matrix, but the values would not correspond to the spring's linear stiffness. The true effect of the spring design shown would need to be modeled, and then linearized, to fit the stiffness model formulated here. The linear bi-articular model is shown here because it is the most clearly displayed design. A rotational spring which, connected to both joints via a gear system, would fit the model we have formulated more precisely, but would be less clearly representable in a line drawing.

6.3 The Task: Walking Gait Data from a Biological Quadruped

We use gait data recorded from biological cats walking, converted from Suter et al. [65] as the target gait for the optimal design. The cat gait data represents a real task performed by a mulitactuator limb, biological or not. We expect that the results of this study give an indicator for the effectiveness of this approach in a robotic system. However, in practice, the trajectory should come from a detailed model which creates

an accurate use-case scenario. This task-specific data then can be easily integrated into the optimization routine we formulate here, and a task-specific stiffness matrix will result. Figure 55a shows the joint torques and velocities throughout the gait trajectory. The motivation for using biological walking data for this exercise is that it is indicative of the torques and velocities present for walking in a legged vehicle. An alternative method would be to simulate the data using an advanced dynamic simulator and a model of a legged robot.

6.4 Formulating Electrical Power in a Geared DC Motor

This section formulates the electrical power consumed by a DC motor for a given torque-velocity task. We first approximate the motor current(I) using a linear relationship with the motor torque (τ_m):

$$I = \frac{1}{K_t}\tau_m. \tag{6.4}$$

Where K_t is the torque constant associated with the motor. The motor torque (τ_m) is a $m \times n$ vector where m is the number of motors and n is the length of the data set. It is estimated from the given joint torque load in the data set (τ_j), the rotational inertia of the motor (J_m), and subtracting the torque contribution of the

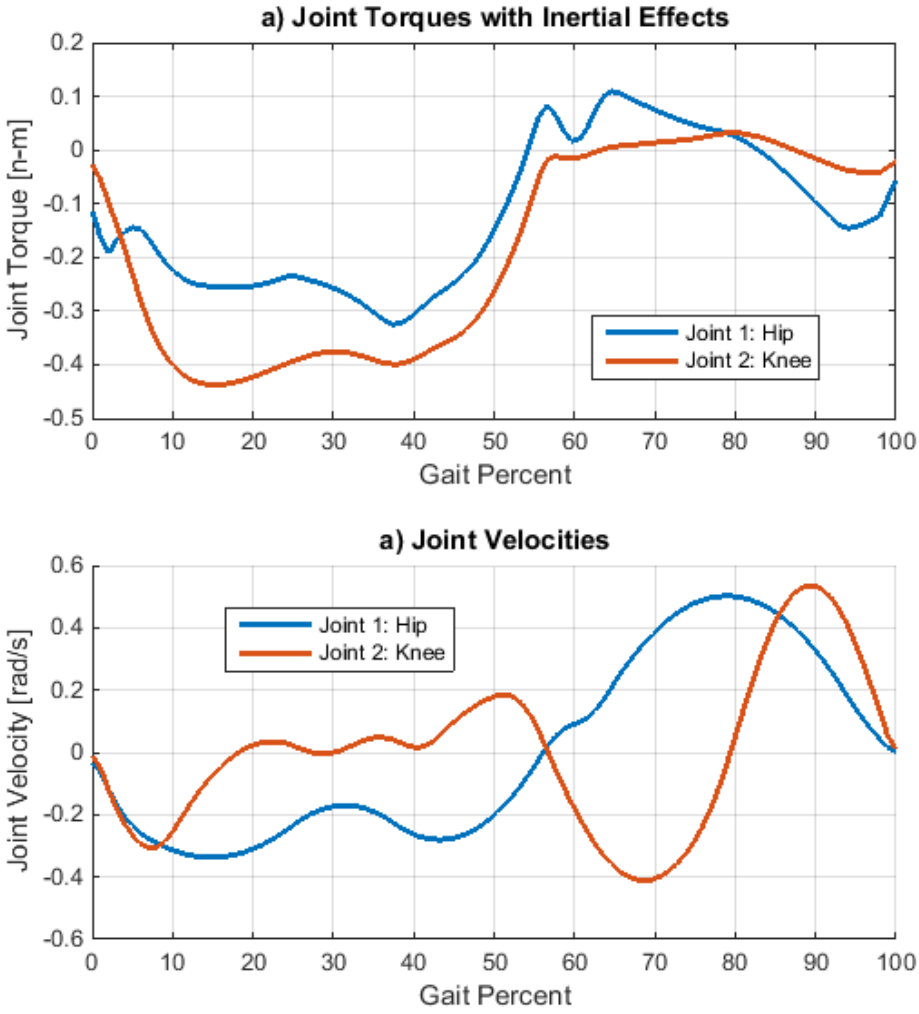

Figure 55. a) Ground reaction force on the end effector b) Position of end effector c) Inertial load on the end effector

springs:

$$\tau_m = g\,\mathrm{J_m}\,\mathrm{a_j} + \frac{1}{g}\tau_\mathrm{j} - \frac{1}{g}\tau_s. \tag{6.5}$$

Where a_j is the angular acceleration of the joint angle. Note that this is where

the spring torque from Equation 6.3 factors into the power formulation. Substituting Equation 6.5 into Equation 6.4 gives:

$$I = \frac{1}{g\mathrm{K_t}}\,\tau_{\mathrm{j}} \;+\; \frac{\mathrm{J_m\,g}}{\mathrm{K_t}}\,a_j \;-\; \frac{1}{g\mathrm{K_t}}\,\tau_s. \tag{6.6}$$

Voltage will be formulated using both a linear speed constant (K_v) and a current dependent effect resulting from the armature resistance(R_a):

$$U = \frac{1}{K_v}\,V_m \;+\; R_a\,I. \tag{6.7}$$

Where V_m is the velocity of the motor given in the data set. It should be noted that U, and I are $m \times n$ matrices, where m is the number of motors and n is the length of data set. Each column of these matrices is a vector containing the motors' instantaneous voltage or current levels. Similarly, the power vector (P) will be constructed as an $m \times n$ matrix containing instantaneous motor power values for each motor. It is the element wise product (notated as $\odot$) of the instantaneous voltage and current matrices.

$$P = U \odot I \tag{6.8}$$

This power matrix, however, cannot be used directly as the performance metric for the optimization problem. Optimization routines require a single scalar to be used as a metric for evaluating one design against another.

6.5 Cost Function Formulation

There are several methods which may be used to produce a performance metric for the optimization routine as was discussed in Section 6.1. In this chapter we will use three methods to formulate the fitness function and compare the results. The first method will be to minimize the sum of the squared power values (SSP). The second method will account for regenerative capabilities of the DC motor. The third method (used in [57]) is to minimize ohmic heating in the windings which is equivalent to minimizing the sum of square current (SSC). Another method would be to minimize the sum of square power at the motor leads. Finally, one could model the regenerative capabilities of the motor as Kim et al. [61] do and minimize power drain at the battery. Each of these formulations will be detailed in this section, and design results will be presented in Section 6.8.

6.5.1 SSP Method

The Sum of Square Power (SSP) Method sums the square of the motor power at each of the m motor across the length of the data set (n).

$$f = \sum_{i=1}^{n} \sum_{j=1}^{m} P_{ij}^2 \tag{6.9}$$

This method penalizes negative power as much as it does positive power. It also penalizes large power spikes more than small power draws due to the non-linearity of the squared function. Large power spikes are undesirable because the size of the motor is often determined by the peak power requirement of a task. Using this method will reduce braking energy in the force-velocity trajectory of the motor, and in all likeliness it will reduce the value of the peak power required by the motor. A small peak power requirement often means a smaller motor may be utilized.

6.5.2 PSP

The Penalized Sum of Power (PSP) Method produces a performance metric which allows for regenerative energy to be subtracted from the total energy but penalizes according to regenerative efficiency. DC motors may be used as generators during the braking portions of a gait cycle. Kim et al.[61] rely on this capability in their design of the cheetah robot.

To truly model the regenerative effect one must consider the efficiencies of the charging circuit and the motor controller. A model such as this is outside the scope of this dissertation. A simpler approach has been taken in the Penalized Sum of Power Method, one which allows regeneration but penalizes it based on its inefficiency.

We know that the process of converting mechanical work to chemical energy in a battery is less efficient than the reverse process. Kim et al. report that regenerative efficiency for their system was measured at 63 percent [61]. We use this figure directly

as the penalty for negative power before summing:

$$P_b = \begin{cases} P_b = P & P > 0 \\ P_b = .63P & P < 0 \end{cases}. \tag{6.10}$$

Where P_b is the power at the battery. The performance metric is then calculated by summing the elements of the matrix:

$$f = \sum_{i=1}^{n} \sum_{j=1}^{m} P_{b_{ij}}. \tag{6.11}$$

This summation produces the second metric.

6.5.3 SSC

The third method is called the Sum of Square Current (SSC) Method. It is a similar approach to that used by Rezazadeh et al. in their paper on optimal transmission ratio [57]. Minimizing SSC metric will also minimize ohmic heating (P_Ω) in the windings and, correspondingly, the size of the motor necessary to accomplish the task. In addition, decreasing the SSC will in this and most other cases, reduce the peak current which will decrease the size and cost of the inverter. This is because ohmic heating is

the product of the squared current and the armature resistance:

$$P_\Omega = I^2 \, R_a. \tag{6.12}$$

Where, R_a is the armature resistance, and I is the current in the windings. We know from calculus that multiplying a function by a scalar will not change the location of its minimum. Therefore, minimizing SSC is the same as minimizing the sum of Ohmic Heating.

The formulation for SSC is as follows:

$$f = \sum_{i=1}^{n} \sum_{j=1}^{m} I_{ij}^2. \tag{6.13}$$

Where, n is the length of the data set and m is the number of actuators.

6.6 Constraints

There are two constraints for this optimization. The first of these constraints is that the spring stiffnesses must be positive, for obvious reasons. The second constraint is that the stiffness matrix must be symmetric. Certain joint combinations may make it infeasible to design parallel springs around. In this case the stiffness associated with infeasible springs may be constrained to zero in the optimization.

6.7 Perturbing the Data

To test the robustness of the design, a set of 50 gait trajectories were produced which have been harmonically perturbed using the coefficients of a Fourier series fit to the unperturbed data set. Instead of adding random noise directly to the leg angles and forces, a seventh order Fourier series was fit to the gait trajectories and then used to create the perturbed trajectories. The coefficients of the Fourier series were perturbed randomly producing seventh order and lower harmonic perturbations around the original gait cycle. Figures 56 and 57 show the perturbed model.

These perturbed models will be used to test how well the spring stiffness solution performs in real world conditions. If slight variations in the gait trajectory cause significant decreases in performance, the solution may be over-specialized.

To create a numerical metric to measure this concept, we compare the value of the cost function under the unperturbed optimal solutions ($\hat{K}_{SSP}$, $\hat{K}_{PSP}$, and $\hat{K}_{SSC}$) to the value of the cost function under what we call the Specific Perturbed Optimal Solution (SPOS). The SPOS is the solution of spring stiffness which minimizes the given performance metric for a perturbed gait trajectory. For this study we had a set of 50 randomly perturbed gait trajectories. This means that there are 50 different SPOS values for each fitness function. We average the percent difference between the SPOS fitness values and the unperturbed optimal stiffness values. This value, $\bar{\Lambda}$ is defined here:

$$\bar{\Lambda} = \frac{100}{n} \sum_{i=1}^{n} \frac{f(\hat{K}, \hat{\beta}) - f(K_i^*, \beta_i^*)}{f(K_i^*, \beta_i^*)} \tag{6.14}$$

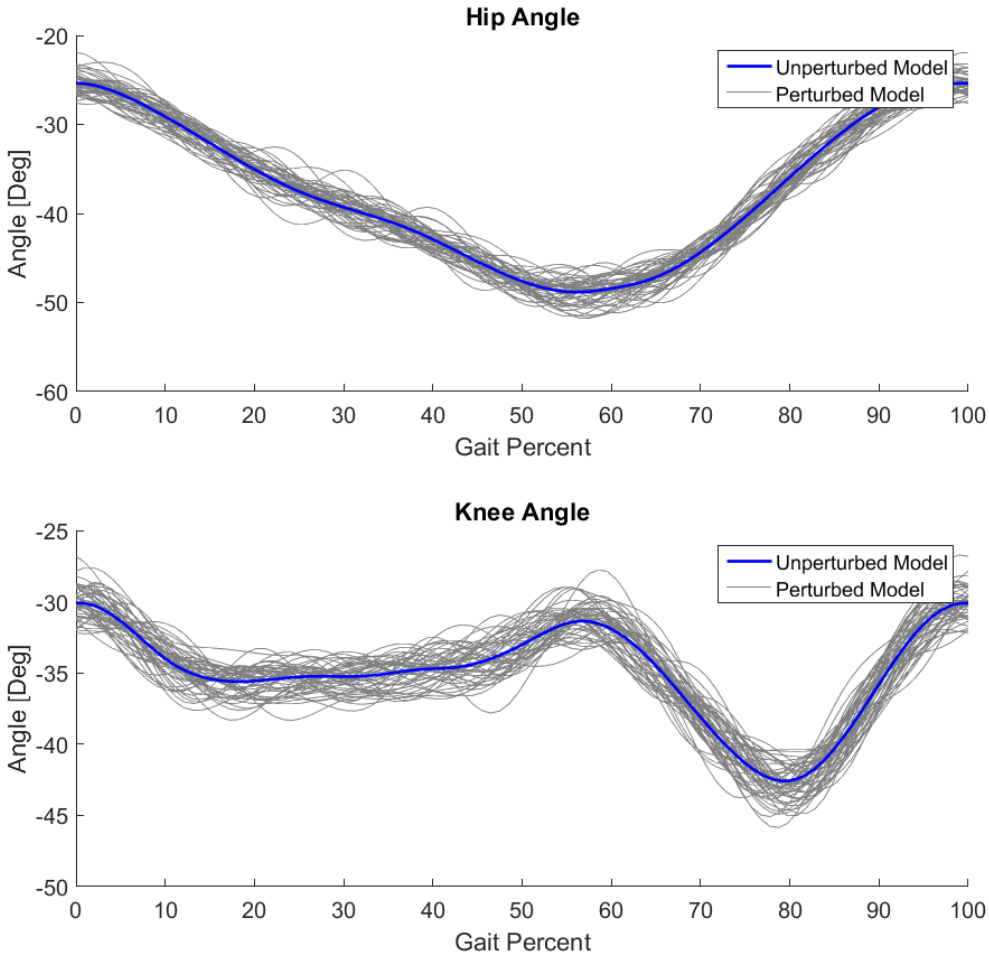

Figure 56. Perturbed Model

Where f, is the fitness function being evaluated, and $\hat{K}$, and $\hat{\beta}$ are the stiffness and offset solutions which minimize that fitness function for the unperturbed data. K_i^* and β_i^* are the SPOS values for the i$^{\text{th}}$ perturbation trajectory.

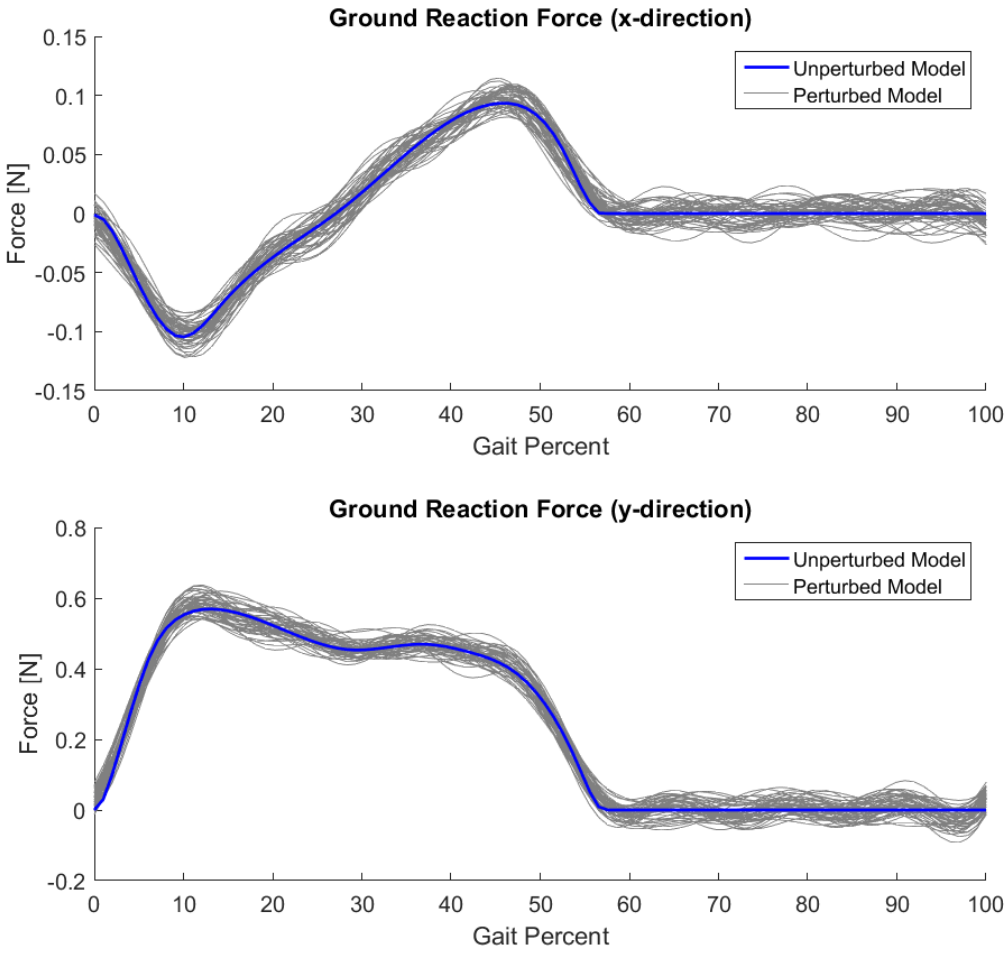

Figure 57. Perturbed Model

6.8 Results

The results of the optimizations using the three metrics will be presented here. Table 11 displays a comparison of the solutions of the optimizations. Each column in the table represents an optimal leg stiffness design, with the exception of the last column which represents a design without parallel compliance. The rows of the table represent the various performance metrics which the designs may be judged by: the

Table 11. Comparing Optimization Results

Performance Metric	Spring Stiffness Solution			
	SSP	PSP	SSC	No Springs
SSP $[Watts^2]$	0.18	0.19	0.29	0.34
PSP $[Watts]$	3.65	3.64	3.86	3.97
SSC $[Amps^2]$	0.092	0.090	0.064	0.141
Peak Power $[Watts]$	0.066	0.076	0.088	0.108
Peak Current $[Amps]$	0.050	0.048	0.033	0.056

first three rows being the methods described in the previous section. While, the last two rows are alternative performance metrics: peak power, and peak current. Peak power is the maximum instantaneous power exerted by any of the motors during any moment in the gait cycle, a determining factor in required motor size. Peak current is the maximum instantaneous motor current in either the positive direction or the negative direction.

Table 12 is related to Table 11 in that the three columns of Table 12 equal the percentage difference of first three columns of Table 11 and the fourth column of Table 11. In short, Table 12 shows by what percentage each metric was reduced compared to the unassisted robot. This gives an easier comparison between the three results.

Table 12. Comparing Optimization Results: Percent Reduction

Performance Metric	Spring Stiffness Solution		
	SSP	PSP	SSC
SSP [%]	47	43	13
PSP [%]	8	8	3
SSC [%]	35	36	55
Peak Power [%]	39	30	19
Peak Current [%]	10	14	41

Table 13 shows the results of the perturbation study. The first three rows of the

table represent the percentage each metric was reduced, on average, compared to the spring-free case.

Each column contains results using the solutions of the three optimization methods: SSP for column one, PSP for column two, and SSC in column three. The first three rows of the table represent the average reduction in each performance metric across the fifty different perturbed gait cycles. The fourth row contains the mean difference, in percentage, from the perturbation-specific best-case scenario, as defined in Equation 6.14. This value effectively represents how sensitive each solution is to perturbation, a lower value representing a more positive result.

Table 13. Effect of perturbation on optimization results

Performance Metric	Spring stiffness solution		
	SSP	PSP	SSC
SSP [%]	41.1	39.2	26.1
PSP [%]	9.0	9.4	6.0
SSC [%]	34.4	35.4	53.7
$\bar{\Lambda}$ [%]	11.0	1.0	1.2

6.8.1 SSP

Minimizing SSP resulted in a 47 percent reduction in SSP, an 8 percent reduction in PSP, 35 percent reduction in SSC, 39 percent reduction in peak power, and a 10 percent reduction in peak current. This means that adding parallel springs could significantly reduce the size of the motor required, the amount of heat produced by the motors during gait, the size of the batteries needed for a given duration of gait, and the size and cost of the power inverter necessary. Even in the presence of random perturbation the spring matrix (with $\hat{K}_{SSP}$ and $\hat{\beta}_{SSP}$ values unchanged) reduced the

SSP by an average of 41 percent, PSP by an average of 9 percent, and SSC by an average of 34 percent. All of these factors are large design considerations, and large contributers to overall system cost. This savings in dollars and mass would have to outweigh the cost of adding the springs to the legs in the first place, but in general mechanical springs have a very high power density and lower cost compared to DC motors.

Figures 58 - 60 compare the power and torque required by the motors with and without parallel springs (labeled: unassisted and assisted, respectively) for each optimization method. The green hashed areas in the figures are the portions of gait where the springs reduced the power/torque required by the motors. Similarly, the red hashed areas mark points where the motors do more with springs than without them. It can be noted from Figure 58 that the SSP method reduced the power required by the motors especially in the first portion of gait ("stance") during which time the motors are performing their peak effort. The power required by the motors during second half of gait (the "swing" portion) is increased marginally as a byproduct. It should be noted that the curves in the figures are actually the two-norms of the instantaneous power/torque vectors at each point along the gait cycle. This show the results of the optimization in a form which is much easier to understand than separately plotting the individual coordinates of the motor power vector.

$$\hat{K}_{SSP} = \begin{bmatrix} .1849 & .1978 \\ .1978 & .6236 \end{bmatrix} \tag{6.15}$$

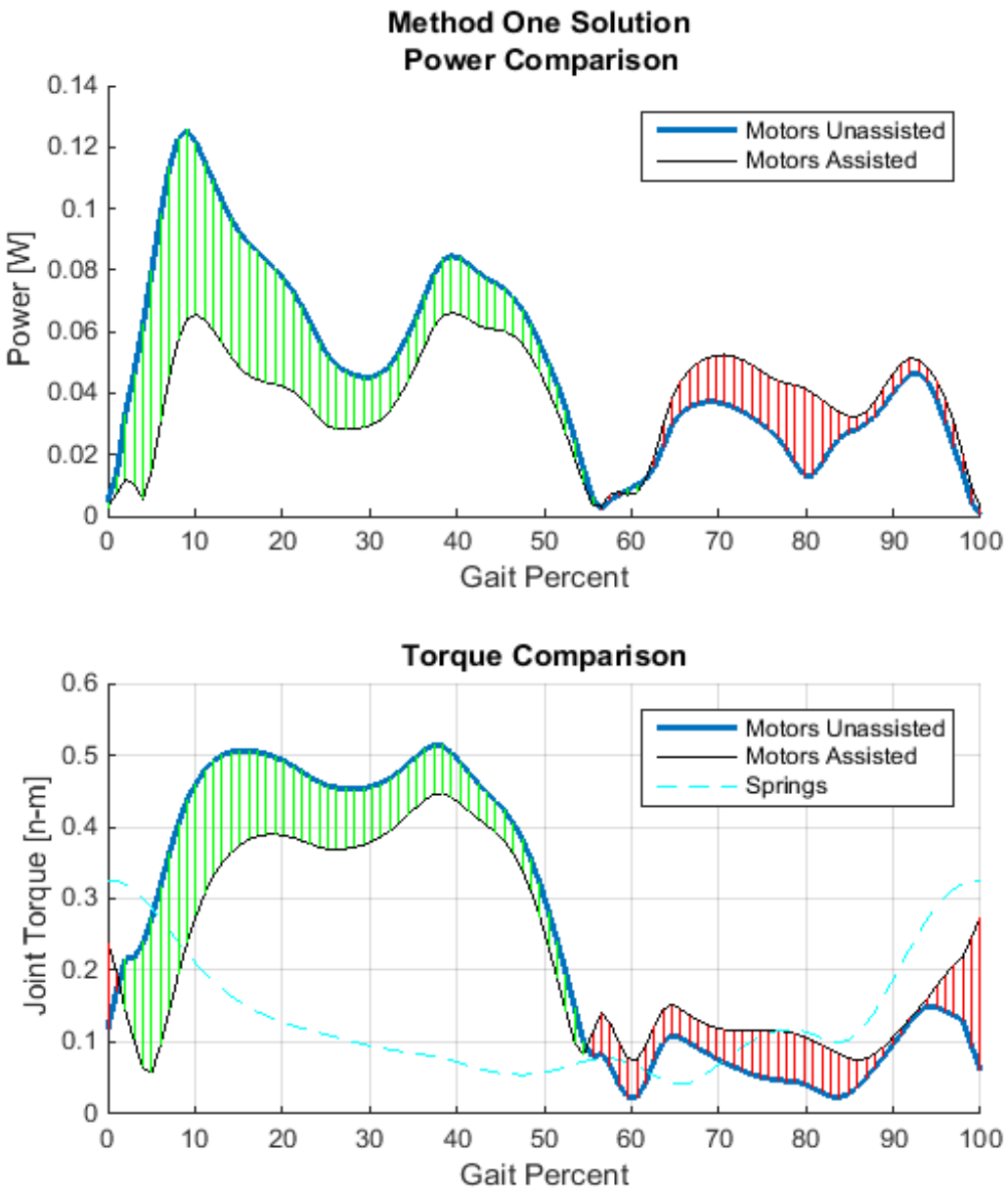

Figure 58. Results of SSP Minimization Problem

$$\hat{\beta}_{SSP} = \begin{bmatrix} -.5845 \\ -1.0769 \end{bmatrix} \tag{6.16}$$

6.8.2 PSP

Minimizing PSP resulted in a 43 percent reduction in SSP, an 8 percent reduction in PSP, 36 percent reduction in SSC, 30 percent reduction in peak power, and a 14 percent reduction in peak current. In the presence of random perturbation the spring matrix (with $\hat{K}_{PSP}$ and $\hat{\beta}_{PSP}$ values unchanged) reduced the SSP by an average of 39 percent, PSP by an average of 9 percent, and SSC by an average of 35 percent.

Comparing these results to the SSP method is an interesting study. Overall, the results are similar to each other. The SSP method scores higher than the PSP method in 2 out of 5 of the metrics in Tables 11 and 12. In the other three the PSP method did marginally better. As expected, the PSP method reduced the PSP metric to a lower value than the SSP method did, but the difference between the two results was less than half a percentage point. Another metric which the PSP scores higher than the SSP is the SSC metric. The Penalized Sum of Power method reduced the Sum of Square Current metric by 36 percent compared to the reduction of 35 percent by the SSP method. Finally the PSP reduced the Peak Current metric by 14 percent whereas the SSP method reduced it by only 10 percent. The SSP method out-performed the PSP method by 9 percentage points in reducing Peak Power, and by 4 percentage points in reduction of Sum of Square Power. The PSP method seems to be less sensitive to perturbation than the SSP, in that the ideal SPOS solution is on average only one percent better than the unperturbed solution. In contrast, the same metric ($\bar{\Lambda}$) is 11 percent for SSP case.

Figures 58 and 59 are not clearly distinguishable from each other. Both methods reduced the power and torque in the peak regions (during stance), at a cost of

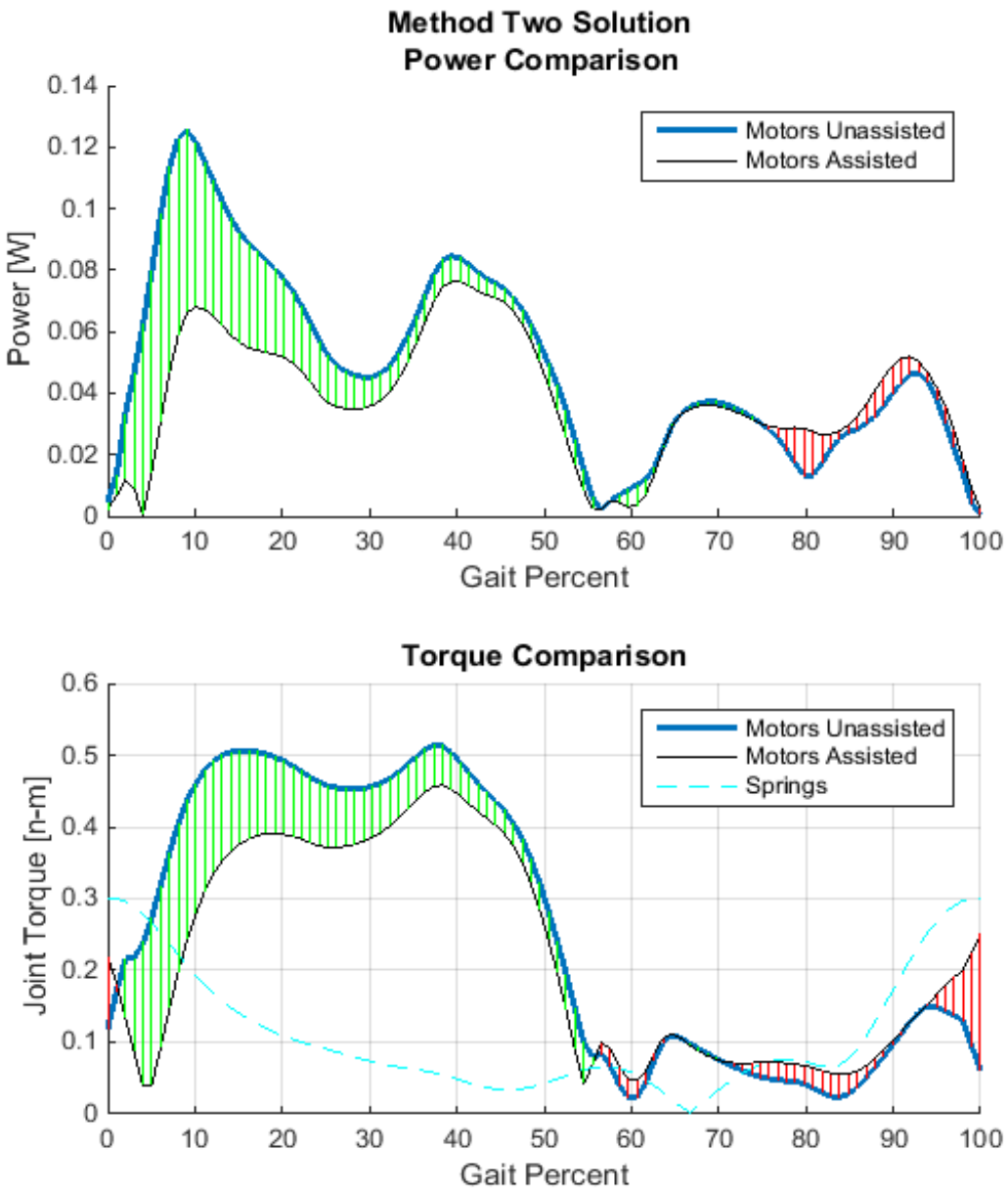

Figure 59. Results of Battery Power Minimization Problem

increasing the power and torque in regions of the gait where it was relatively small (the swing region).

$$\hat{K}_{PSP} = \begin{bmatrix} .2126 & .1841 \\ .1841 & .5172 \end{bmatrix} \tag{6.17}$$

$$\hat{\beta}_{PSP} = \begin{bmatrix} -.5702 \\ -.9418 \end{bmatrix} \tag{6.18}$$

The actual stiffness ($\hat{K}_{PSP}$) and offset solution ($\hat{\beta}_{PSP}$) vary from their counterparts in the SSP solution in value but not so much in their form. The stiffness matrices have off diagonals (biarticular spring stiffnesses) which are on a similar scale the hip stiffness (K_{11}). The knee stiffness (K_{22}), on the other hand, is several times larger than the others. It should be noted that, though the .63 penalty factor comes from literature, the most accurate value for the penalty will change from system to system, and it is likely that the optimization result will be sensitive to this change.

6.8.3 SSC

The results of the SSC minimization were distinctive from the other two minimizations. Several of the metrics were reduced, or increased, with double-digit percentage differences compared to the SSP and PSP methods (see Table 12). Notably, the method reduced the Sum of Square Current metric by 55 percentage points, whereas

the SSP and PSP methods reduced the figure by a mere 36 and 35 percentage points, respectively. On the other hand, the SSC minimization reduced sum of square power by only thirteen percent compared to the SSP and PSP methods' 47 and 43 respective percentage point reductions. The peak current was reduced by 41 percent, compared to 10 and 14 percent reductions from the other two methods, while the Peak Power metric was reduced by only 19 percent by the Sum of Square Current Method compared with a 39 and 30 percent reduction achieved by the others. The results of the perturbation study follow the same trend: the SSC method scores much lower on the SSP and PSP metrics, but much higher on the SSC metric. According to it's low $\bar{\Lambda}$ score the optimal solution for the SSC metric does not change much with perturbation, the SPOS best-possible solutions are on average only 1.2 percent better than the unperturbed solution.

Figure 60 is also clearly distinguishable from the other two methods. In Figure 60 (a), notice the relatively large increase in power during the second portion of gait, and decrease of power in the first. In Figure 60 (b) notice that torque is more evenly distributed throughout the gait cycle, as opposed to the larger peaks seen in Figures 58 and 59.

$$\hat{K}_{SSC} = \begin{bmatrix} .1762 & .0339 \\ .0339 & .5172 \end{bmatrix} \tag{6.19}$$

$$\hat{\beta}_{SSC} = \begin{bmatrix} -.3948 \\ -.9078 \end{bmatrix} \tag{6.20}$$

The differences in the actual stiffness and offset solution of this method show up in the off-diagonal terms of the $\hat{K}_{SSC}$ matrix. The biarticular spring stiffness has been reduced almost a full order of magnitude from the previous results. This, along with small differences in the offset terms, may be responsible for the noticeable performance differences.

6.9 Conclusions

This introductory work shows that the EPCM may dramatically reduce the power required for a legged vehicle to ambulate if designed carefully. We optimize the spring design according to three separate fitness functions. Each fitness function produces a spring stiffness matrix which significantly reduces the performance requirements of the actuators. Each performance metric reduces some metrics more than the other. It is difficult to make a decision about which method is the best without more context about the problem and the associated goals. The penalized sum of power metric has the potential of reducing the size of the battery or increasing the range of the system. The sum of square power will reduce the peak power value, along with extending the battery life by reducing the PSP nearly as much as the PSP method itself. This means a reduction in the size of the motor *and* the battery size. Finally, the SSC metric minimized the ohmic heating and dramatically reduced the peak current value, at the cost of comparatively low savings in SSP and PSP. This implies that the cost of the inverter will decrease along with the temperature of the windings, and potentially

the size of the motor. No matter which of these three advantage sets seems most attractive to the designer, any one of them may dramatically change the capability of the robotic system being designed. Double digit percentage point decreases in many key performance metrics is a significant finding. Coupling this with the fact that the perturbation study shows that these performance improvements remain in the presence of small changes in the task trajectory, makes the results even more significant. This means that if the system and task are well defined, the performance gains modeled in this chapter should be applicable to real-world scenarios which will inevitably include perturbations. In conclusion, the results of this study show that parallel springs should be considered in the design robotic limbs required to efficiently perform a specific task.

6.10 Future Work

In the future an algorithm could be developed which accounts for more realistic design constraints. Since linear torsional springs are difficult to implement, future work could take into account the non-linearities which would most likely be present in real designs. Also, with a realistic design the mass of the springs would be known and should be taken into consideration. Furthermore using an exhaustive spring matrix may be infeasible.

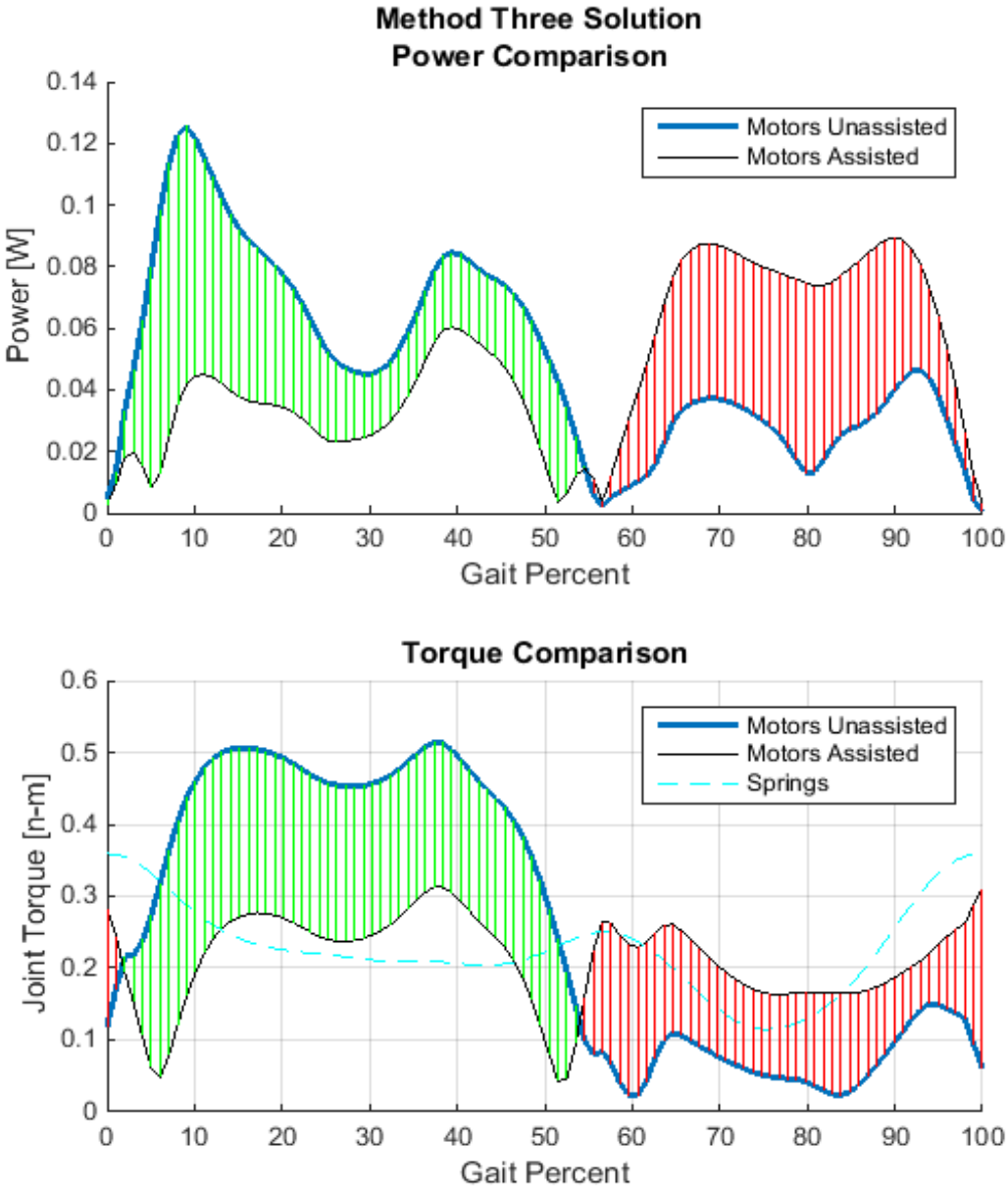

Figure 60. Results of Waste Heat Minimization Problem: (a) comparison of power required by the unassisted limb versus the spring-assisted system (b) comparison of torque required by the unassisted limb versus the spring assisted system

Chapter 7

CONNECTIONS

This dissertation introduces three optimization routines. A geometry optimization is introduced in Chapter 4, a compliance optimization in Chapter 6, and a velocity/force optimization in Chapter 5. This chapter attempts to make connections by comparing the different optimization algorithms.

One of the three algorithms cannot be compared to the others. The velocity/force optimization solution in Chapter 5 is only applicable to overactuated systems. Whereas, the other two algorithms apply only to holonomic systems. In spite of this, even comparing the compliance optimization with the kinematics optimization is complicated. For example, performance of the optimization is highly dependent on the task trajectory associated with the design. In some scenarios one method may be far superior and in other scenarios it may be ineffective. Springs are generally better than motors at storing and quickly releasing energy, so task trajectories such as hopping may cater towards using the compliance optimization. On the other hand the kinematics of a leg may be used to create near singularities which dramatically amplify joint torques at the cost of end-effector mobility. For this reason, the geometric optimization algorithm may be expected to perform well when the task trajectory involves large loads with little velocity in the loading direction. For example, tasks like walking or standing require a large vertical force, but relatively small vertical end effector velocity.

Another factor that complicates the question is that it is difficult, in some cases, to measure what "better" means. It could be measured in terms of percent improvement but, in the case of the geometry optimization, it is difficult to define the nominal case. Which geometry should be used to compare the performance improvement? In the case of the compliance optimization, the nominal case is a zero stiffness spring.

Finally, it is possible that there may be cases where combining the two optimizations produces synergistic effects. In other words, an optimization routine may be developed

which tunes the geometry and the compliance of a design simultaneously. This would make it possible for the geometry of the limb to take advantage of the compliant elements of the design, and vise versa. This idea, however, has not been further explored in this document.

Chapter 8

CONTRIBUTIONS

This chapter summarizes the various contributions of this work. There are four major areas of contribution: the development of the KiTy SP robotic leg, the formulation of an optimization routine capable of tuning the geometry of complex multiactuator robotic limbs, the Exhaustive Parallel Compliance Matrix and the methods to tune its associated design variables, and finally the detailed analysis of the Mechanical Antagonism phenomenon.

8.1 Development of Novel Robotic Leg

A large part of the contribution of my work is in the development of the KiTy SP Robotic Limb, a complex multi-actuator robot leg meant for high performance legged gait. After an extensive review of published literature it seems that this design is not a recreation of any other published work. In the development of this novel actuation method including a working prototype many obstacles needed to be overcome.

8.1.1 Solved Kinematic Equations

One such obstacle is solving the kinematic equations of the leg. This has been accomplished and was not a trivial task. The inverse kinematics of the legs have been solved such that the exact motor positions can be calculated in one step given a desired end effector position.

8.2 Systematic Optimal Design Methods for Tuning the Design of Multiactuator Robotic limbs

8.2.1 Geometry Optimization

The geometry of the limb has been optimized to task specific data in order to minimize motor antagonism and other sources of energy loss in the system. This optimization considers and takes advantage of the passive dynamics of the robotic limb, including nonlinear effects of the linkage closure equations. Complexities like these become an advantage, reducing the over-all power required to perform a task compared to more simple leg structures. Not only has the geometry of the limb has been optimized to allow robust movement, but the power required to perform a specific task has been minimized such that the power required by the task for the KiTy SP Structure, is significantly less than a similarly-tuned serial limb structure inspired by biology.

8.2.2 Leg is capable of robust three-dimensional control

After developing the optimization tools needed to produce a functioning geometry of the complex KiTy SP structure, the question was whether or not it the design would be feasible given the unmodeled spatial design constraints especially interference between the many moving parts of the design. Building a working model and moving it through the designed gait cycle showed that this design approach is capable of producing truly feasible designs to be used in real-world systems. Future iterations of

of the optimization tool could work within a CAD software and self-check for model interference.

8.2.3 Workspace analysis using Condition Number

Using the conditioning number of the Jacobian as a constraint on a power minimization algorithm was a critical aspect to the creation of a working design optimization algorithm. Its use in this context is a novel contribution of the author

8.2.4 Gear Ratio Optimization

There has been a long list of contributors to this field, but my approach contains a full model optimization with motor constraints included. I think my approach still contributes to the field.

8.2.5 Compliance Optimization

I explore adding an Exhaustive Parallel Compliance Matrix (ECPM) to a multi-actuator robotic limb, and develop a constrained optimization routine to tune the stiffnesses and offset values of these springs. The ECPM is a matrix of monarticular (crossing one joint) parallel springs which cross every joint and multiarticular (crossing more than one joint) parallel springs which cross every joint combination. The optimization routine finds the ideal stiffnesses of these springs given a task-trajectory. I explore the idea using a 2 DoF planar serial arm and show that, under gait-like loading conditions, there is dramatic potential for energy reduction. The method can

also be constrained to include as little as one spring in the design, and as a method to determine which joint or joint combination would be ideal for parallel springs.

8.2.6 Antagonistic Power regions and Power Optimal Pseudoinverses

I contribute a detailed explanation of how the freedoms of a multiactuator robotic leg create directional regions in the force-velocity space which are inherently undesirable for power performance of robotic systems. I also present a modified pseudoinverse of the Jacobian for over-actuated systems, which minimizes mechanical antagonism.

REFERENCES

[1] A. Abate, R.L. Hatton, and J. Hurst. "Passive-Dynamic Leg Design for Agile Robots". In: *2015 IEEE International Conference on Robotics and Automation (ICRA)*. 2015 IEEE International Conference on Robotics and Automation (ICRA). May 2015, pp. 4519–4524.

[2] Andy Abate, Jonathan W. Hurst, and Ross L. Hatton. "Mechanical Antagonism in Legged Robots". In: Robotics: Science and Systems XII. 2016.

[3] "Adjustable Stiffness Jack Spring Actuator". Thomas Sugar and Kevin Hollander. Aug. 2011.

[4] R. McNeill Alexander. "Tendon Elasticity and Muscle Function". In: *Comparative Biochemistry and Physiology Part A: Molecular & Integrative Physiology* 133.4 (2002), pp. 1001–1011.

[5] Christopher L. Benson and Christopher L. Magee. "Quantitative Determination of Technological Improvement from Patent Data". In: *PLOS ONE* 10.4 (Apr. 15, 2015), e0121635.

[6] Christopher Benson and Christopher L. Magee. "On Improvement Rates for Renewable Energy Technologies: Solar PV, Wind Turbines, Capacitors, and Batteries". In: *Renewable Energy* 68 (2014).

[7] JCPA Bhounsule, Jason Cortell, and Andy Ruina. "Design and Control of Ranger: An Energy-Efficient, Dynamic Walking Robot". In: *Proc. CLAWAR*. 2012, pp. 441–448.

[8] T. Boaventura et al. "Dynamic Torque Control of a Hydraulic Quadruped Robot". In: *2012 IEEE International Conference on Robotics and Automation (ICRA)*. 2012 IEEE International Conference on Robotics and Automation (ICRA). May 2012, pp. 1889–1894.

[9] *Boston Dynamics: Dedicated to the Science and Art of How Things Move.* URL: http://www.bostondynamics.com/robot_bigdog.html (visited on 05/16/2016).

[10] Nathan Cahill. "Advanced Parallel Actuation of a Serial Robotic Leg". In: ASME IDETC 2016. Charolette, NC, 2016.

[11] Nathan Cahill, Yi Ren, and Thomas Sugar. “Mechanical Specialization of Robotic Limbs”. In: International Conference on Robotics and Automation. Singapore, May 29, 2017.

[12] Dar-Zen Chen and Lung-Wen Tsai. “Kinematic and Dynamic Synthesis of Geared Robotic Mechanisms”. In: *Journal of Mechanical Design* 115.2 (1993), pp. 241–246.

[13] D.-Z. Chen and L.-W. Tsai. “The Generalized Principle of Inertia Match for Geared Robotic Mechanisms”. In: *Robotics and Automation, 1991. Proceedings., 1991 IEEE International Conference On*. IEEE, 1991, pp. 1282–1287.

[14] Giancarlo Cusimano. “A Procedure for a Suitable Selection of Laws of Motion and Electric Drive Systems under Inertial Loads”. In: *Mechanism and machine theory* 38.6 (2003), pp. 519–533.

[15] Giancarlo Cusimano. “Choice of Electrical Motor and Transmission in Mechatronic Applications: The Torque Peak”. In: *Mechanism and Machine Theory* 46.9 (Sept. 2011), pp. 1207–1235.

[16] Giancarlo Cusimano. “Generalization of a Method for the Selection of Drive Systems and Transmissions under Dynamic Loads”. In: *Mechanism and machine theory* 40.5 (2005), pp. 530–558.

[17] Joseph K. Davidson, Kenneth H. Hunt, and G. R. Pennock. *Robots and Screw Theory: Applications of Kinematics and Statics to Robotics, 2004*. Oxford University Press Inc., Nova Iorque, 2004.

[18] *Extreme High Energy Density - Amicell*. URL: http://www.amicell.co.il/batteries/rechargeable-batteries/our-extreme-high-energy-density-lithium-polymer-series/ (visited on 01/12/2017).

[19] J. Doyne Farmer and François Lafond. “How Predictable Is Technological Progress?” In: *Research Policy* 45.3 (Apr. 2016), pp. 647–665.

[20] C. Gehring et al. “Practice Makes Perfect: An Optimization-Based Approach to Controlling Agile Motions for a Quadruped Robot”. In: *IEEE Robotics Automation Magazine* 23.1 (Mar. 2016), pp. 34–43.

[21] C. Gosselin and Jorge Angeles. “The Optimum Kinematic Design of a Spherical Three-Degree-of-Freedom Parallel Manipulator”. In: *Journal of Mechanical Design* 111.2 (1989), pp. 202–207.

[22] Jesse A. Grimes and Jonathan W. Hurst. "The Design of ATRIAS 1.0 a Unique Monoped, Hopping Robot". In: *Proceedings of the 2012 International Conference on Climbing and Walking Robots and the Support Technologies for Mobile Machines*. 2012, pp. 548–554.

[23] Masato Hirose and Kenichi Ogawa. "Honda Humanoid Robots Development". In: *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences* 365.1850 (Jan. 15, 2007), pp. 11–19.

[24] Joseph K. Hitt et al. "The Sparky (Spring Ankle with Regenerative Kinetics) Project: Design and Analysis of a Robotic Transtibial Prosthesis with Regenerative Kinetics". In: *ASME 2007 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*. American Society of Mechanical Engineers, 2007, pp. 1587–1596.

[25] N. Hogan. "Impedance Control: An Approach to Manipulation". In: *American Control Conference, 1984*. American Control Conference, 1984. June 1984, pp. 304–313.

[26] Matthew A. Holgate et al. "The SPARKy (Spring Ankle with Regenerative Kinetics) Project: Choosing a DC Motor Based Actuation Method". In: *2008 2nd IEEE RAS & EMBS International Conference on Biomedical Robotics and Biomechatronics*. IEEE, 2008, pp. 163–168.

[27] Kevin W. Hollander, Thomas G. Sugar, and Donald E. Herring. "A Robotic "Jack Spring"TM for Ankle Gait Assistance". In: *ASME 2005 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*. American Society of Mechanical Engineers, 2005, pp. 25–34.

[28] Kevin W. Hollander, Thomas G. Sugar, and Donald E. Herring. "Adjustable Robotic Tendon Using a'jack Spring'TM". In: *9th International Conference on Rehabilitation Robotics, 2005. ICORR 2005.* IEEE, 2005, pp. 113–118.

[29] Kevin W. Hollander et al. "A Passive and Active Joint Torque Augmentation Robot (JTAR) for Hip Gait Assistance". In: *ASME 2014 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*. American Society of Mechanical Engineers, 2014, V05AT08A079–V05AT08A079.

[30] Kevin W. Hollander et al. "An Efficient Robotic Tendon for Gait Assistance". In: *Journal of biomechanical engineering* 128.5 (2006), pp. 788–791.

[31] Kevin W. Hollander et al. “Design of the Orthotic Load Assistance Device (OLAD)”. In: *7th Annual Dynamic Walking Conference, IHMC*. 2012, pp. 172–173.

[32] J. W. Hurst and A. A. Rizzi. “Series Compliance for an Efficient Running Gait”. In: *IEEE Robotics Automation Magazine* 15.3 (Sept. 2008), pp. 42–51.

[33] Marco Hutter et al. “StarlETH: A Compliant Quadrupedal Robot for Fast, Efficient, and Versatile Locomotion”. In: *15th International Conference on Climbing and Walking Robot-CLAWAR 2012*. 2012.

[34] M. Hutter et al. “High Compliant Series Elastic Actuation for the Robotic Leg ScarlETH”. In: *Proc. of the International Conference on Climbing and Walking Robots (CLAWAR)* (2011).

[35] M. Hutter et al. “ScarlETH: Design and Control of a Planar Running Robot”. In: *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*. 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems. Sept. 2011, pp. 562–567.

[36] Dong Jin Hyun et al. “High Speed Trot-Running: Implementation of a Hierarchical Controller Using Proprioceptive Impedance Control on the MIT Cheetah”. In: *The International Journal of Robotics Research* 33.11 (2014), pp. 1417–1445.

[37] *Instructions per Second*. In: *Wikipedia*. Dec. 27, 2016.

[38] Devin Koepl and Jonathan Hurst. “Force Control for Planar Spring-Mass Running”. In: *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Sept. 2011, pp. 3758–3763.

[39] Devin Koepl and Jonathan Hurst. “Impulse Control for Planar Spring-Mass Running”. In: *Journal of Intelligent & Robotic Systems* 74 (3-4 June 1, 2014), pp. 589–603.

[40] Devin Koepl, Kevin Kemper, and Jonathan Hurst. “Force Control for Spring-Mass Walking and Running”. In: *2010 IEEE/ASME International Conference on Advanced Intelligent Mechatronics*. IEEE, 2010, pp. 639–644.

[41] Heebyung Koh and Christopher L. Magee. “A Functional Approach for Studying Technological Progress: Application to Information Technology”. In: *Technological Forecasting and Social Change* 73.9 (Nov. 2006), pp. 1061–1083.

[42] Marjolein M. van der Krogt et al. “Robust Passive Dynamics of the Musculoskeletal System Compensate for Unexpected Surface Changes during Human Hopping”. In: *Journal of Applied Physiology* 107.3 (Sept. 1, 2009), pp. 801–808. pmid: 19589956.

[43] Tad McGeer. “Passive Dynamic Walking”. In: *The international journal of robotics research* 9.2 (1990), pp. 62–82.

[44] Tad McGeer. “Passive Walking with Knees”. In: *Robotics and Automation, 1990. Proceedings., 1990 IEEE International Conference On*. IEEE, 1990, pp. 1640–1645.

[45] K. Miller. “Experimental Verification of Modeling of DELTA Robot Dynamics by Direct Application of Hamilton’s Principle”. In: *, 1995 IEEE International Conference on Robotics and Automation, 1995. Proceedings.* , 1995 IEEE International Conference on Robotics and Automation, 1995. Proceedings. Vol. 1. May 1995, 532–537 vol.1.

[46] Karol Miller. “Maximization of Workspace Volume of 3-DOF Spatial Parallel Manipulators”. In: *Journal of Mechanical Design* 124.2 (2002), pp. 347–350.

[47] Karol Miller. “Optimal Design and Modeling of Spatial Parallel Manipulators”. In: *The International Journal of Robotics Research* 23.2 (Jan. 2, 2004), pp. 127–140.

[48] J. A. Paradiso and T. Starner. “Energy Scavenging for Mobile and Wireless Electronics”. In: *IEEE Pervasive Computing* 4.1 (Jan. 2005), pp. 18–27.

[49] Kenneth A. Pasch and W. P. Seering. “On the Drive Systems for High-Performance Machines”. In: *Journal of mechanisms, transmissions, and automation in design* 106.1 (1984), pp. 102–108.

[50] N. Perrin et al. “Continuous Legged Locomotion Planning”. In: *IEEE Transactions on Robotics* PP.99 (2016), pp. 1–6.

[51] F. Pierrot, P. Dauchez, and A. Fournier. “Fast Parallel Robots”. In: *Journal of Robotic Systems* 8.6 (Dec. 1, 1991), pp. 829–840.

[52] F. Pierrot, P. Dauchez, and A. Fournier. “HEXA: A Fast Six-DOF Fully-Parallel Robot”. In: *, Fifth International Conference on Advanced Robotics, 1991. ’Robots in Unstructured Environments’, 91 ICAR.* , Fifth International Conference on Advanced Robotics, 1991. ’Robots in Unstructured Environments’, 91 ICAR. June 1991, 1158–1163 vol.2.

[53] F. Pierrot, C. Reynaud, and A. Fournier. “DELTA: A Simple and Efficient Parallel Robot”. In: *Robotica* 8 (02 Apr. 1990), pp. 105–109.

[54] Marc Raibert et al. “Bigdog, the Rough-Terrain Quadruped Robot”. In: *Proceedings of the 17th World Congress*. 2008, pp. 10823–10825.

[55] Alireza Ramezani et al. “Performance Analysis and Feedback Control of ATRIAS, A Three-Dimensional Bipedal Robot”. In: *Journal of Dynamic Systems, Measurement, and Control* 136.2 (Dec. 9, 2013), pp. 021012–021012.

[56] Daniel Renjewski et al. “Exciting Engineered Passive Dynamics in a Bipedal Robot”. In: *IEEE Transactions on Robotics* 31.5 (2015), pp. 1244–1251.

[57] Siavash Rezazadeh and Jonathan W. Hurst. “On the Optimal Selection of Motors and Transmissions for Electromechanical and Robotic Systems”. In: *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2014, pp. 4605–4611.

[58] Siavash Rezazadeh and Jonathan W. Hurst. “Toward Step-by-Step Synthesis of Stable Gaits for Underactuated Compliant Legged Robots”. In: *2015 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2015, pp. 4532–4538.

[59] Siavash Rezazadeh et al. “Spring-Mass Walking with ATRIAS in 3D: Robust Gait Control Spanning Zero to 4.3 Kph on a Heavily Underactuated Bipedal Robot”. In: *ASME 2015 Dynamic Systems and Control Conference*. American Society of Mechanical Engineers, 2015, V001T04A003–V001T04A003.

[60] A. Ruina. “Cornell Ranger, 2011 4-Legged Bipedal Robot”. In: (2012).

[61] Sangok Seok et al. “Design Principles for Highly Efficient Quadrupeds and Implementation on the MIT Cheetah Robot”. In: *2013 IEEE International Conference on Robotics and Automation (ICRA)*. 2013 IEEE International Conference on Robotics and Automation (ICRA). May 2013, pp. 3307–3312.

[62] S. Seok et al. “Actuator Design for High Force Proprioceptive Control in Fast Legged Locomotion”. In: *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*. 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems. Oct. 2012, pp. 1970–1975.

[63] Bruno Siciliano et al. *Robotics: Modelling, Planning and Control*. Springer Science & Business Media, Aug. 20, 2010. 644 pp.

[64] Richard E. Stamper, Lung-Wen Tsai, and Gregory C. Walsh. "Optimization of a Three DOF Translational Platform for Well-Conditioned Workspace". In: *Robotics and Automation, 1997. Proceedings., 1997 IEEE International Conference On*. Vol. 4. IEEE, 1997, pp. 3250–3255.

[65] E. Suter et al. "One-Year Changes in Hind Limb Kinematics, Ground Reaction Forces and Knee Stability in an Experimental Model of Osteoarthritis". In: *Journal of Biomechanics* 31.6 (June 1, 1998), pp. 511–517.

[66] Lung-Wen Tsai and Tsai; Lung-Wen. *Multi-Degree-of-Freedom Mechanisms for Machine Tools and the Like*. Apr. 3, 1995.

[67] H. J. Van de Straete, Joris De Schutter, and K. U. Leuven. "Optimal Variable Transmission Ratio and Trajectory for an Inertial Load with Respect to Servo Motor Size". In: *Journal of Mechanical Design* 121.4 (1999), pp. 544–551.

[68] Herman J. Van de Straete, Joris De Schutter, and Ronnie Belmans. "An Efficient Procedure for Checking Performance Limits in Servo Drive Selection and Optimization". In: *IEEE/ASME transactions on mechatronics* 4.4 (1999), pp. 378–386.

[69] Herman J. Van de Straete et al. "Servo Motor Selection Criterion for Mechatronic Applications". In: *IEEE/ASME Transactions on mechatronics* 3.1 (1998), pp. 43–50.

[70] H. R. Vejdani et al. "Touch-down Angle Control for Spring-Mass Walking". In: *2015 IEEE International Conference on Robotics and Automation (ICRA)*. 2015 IEEE International Conference on Robotics and Automation (ICRA). May 2015, pp. 5101–5106.

[71] H. Wagner and R. Blickhan. "Stabilizing Function of Skeletal Muscles: An Analytical Investigation". In: *Journal of Theoretical Biology* 199.2 (July 21, 1999), pp. 163–179.

[72] K. J. Waldron and G. L. Kinzel. "The Relationship between Actuator Geometry and Mechanical Efficiency in Robots". In: *Fourth, _ 'mposium on Theory and Practice of Robots and Manipulators. Poland* (1981).

[73] T. Yang et al. "Design and Control of a Planar Bipedal Robot ERNIE with Parallel Knee Compliance". In: *Autonomous Robots* 25.4 (Nov. 1, 2008), p. 317.

[74] Z. G. Zhang and H. Kimura. “Rush: A Simple and Autonomous Quadruped Running Robot”. In: *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering* 223.3 (2009), pp. 323–336.