DEVELOPMENT AND TESTING OF AN UNPOWERED ANKLE EXOSKELETON FOR WALKING ASSIST

Justin Leclair

A thesis submitted to the
Faculty of Graduate and Postdoctoral Studies
In partial fulfillment of the requirement for the degree of
Masters of Applied Science in Biomedical Engineering

Ottawa-Carleton Institute for Biomedical Engineering
Department of Mechanical Engineering
Faculty of Engineering
University of Ottawa

ABSTRACT

Assistive technologies traditionally rely on either strong actuation or passive structures to provide users with increased strength, support or the ability to perform lost functions. At one end of the spectrum are powered exoskeletons, which significantly increase a user's strength, but require strong actuators, complex control systems, and heavy power sources. At the other end are orthoses, which are generally unpowered and lightweight devices that rely on their structure's mechanical behaviour to enhance user's support and stability. Ideally, assistive technologies should achieve both systems' characteristics by enhancing human motion abilities while remaining lightweight and efficient. This can be achieved by using distinctive actuators to harness gait energy, towards enhancing human mobility and performance.

Pneumatic Artificial Muscles (PAMs), compliant and flexible, yet powerful and lightweight, present a unique set of characteristics compared to other mechanical actuators in human mobility applications. However, given the need of a compressor and power source, PAMs present a significant challenge, limiting their application. In contrast, PAMs can be implemented as unpowered actuators that act as non-linear elastic elements.

This thesis aims to develop a wearable lightweight unpowered ankle exoskeleton, which relies on the PAM to harness gait energy and compliment the human ankle biomechanical abilities at the push off movement, thusly assisting the user in propelling the body forward during walking. Presently, limited PAM models have been developed to analyse PAM passive behaviour and to assist in designing and selecting the appropriate PAM for unpowered application. Thus, this thesis aims to develop a passive model for the PAM.

To mechanically validate the proposed exoskeleton design, a prototype is fabricated, and tested within an Instron tensile machine setup. The unpowered exoskeleton has shown its ability to provide significant contribution to the ankle timed precisely to release at the push off phase of the gait cycle. Furthermore, the proposed PAM stiffness model is validated experimentally, and accounts for muscle pressure, geometry, material and stretching velocity. This enables the evaluation of the impact of various parameters on the muscle behaviour and designs the PAM accordingly for the unpowered ankle exoskeleton.

ACKNOWLEDGMENTS

My master’s thesis is the result of years of education, and hard work. It would not have been possible without the support and encouragement of many.

First, I would like to express a special thanks to my thesis supervisor Dr. Marc Doumit, whose guidance and mentorship was instrumental in the accomplishment of my master’s thesis.

I want to thank the faculty members and staff members at the University of Ottawa and Carleton University that are part of the Ottawa-Carleton Institute for Biomedical Engineering. Moreover, I would like to thank the personnel with the University of Ottawa, Department of Mechanical Engineering and the University of Ottawa’s machine shop staff that contributed directly or indirectly to my achievement.

A special thanks to my family and friends who supported me in writing, gave moral and emotional support and incented me to strive towards my goals. I would like to express great appreciation to my wife Caroline who was always there to provide support and encouragement through the duration of my studies.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1 - INTRODUCTION

1.1 Introduction

When considering the complexity of the human body, walking is often overlooked as a simple process, expending little energy and requiring limited resources. Whereas walking is a relatively low energy process for healthy individuals, the amount of energy spent on walking in daily life is larger than any other activity [1] due to its continual use. Estimates indicate the average person takes just under 10 000 steps a day [2]. Therefore a reduction of energy used during walking can present a significant advantage for both healthy and reduced mobility individuals. Considering that walking is already very efficient, due to various internal energy saving and optimization strategies developed over time [3], improving the energy consumption of walking for healthy individuals presents a difficult challenge.

1.2 Rationale

Current assistive technologies for able-bodied individuals vary from powered exoskeletons to passive orthoses. Generally, exoskeletons aim to enhance human physical abilities, usually with powerful actuation and complex control systems. Orthoses, for their part, are habitually passive in nature and serve a single function, such as correction, support or protection. Assistive technologies that aim to reduce energy consumption should ensure that the energy reduction is larger than the energy expenditure caused by the added weight, while preventing discomfort, which has, historically, been a challenge [4]. Unpowered assistive devices are light and comfortable; however, it is difficult to imagine that optimizing gait is possible through the addition of passive components, when human locomotion already has been optimized over millennials [5].

That being said, unpowered assistive devices have been developed and have been shown to capture and release energy through the gait cycle. However, limited devices have shown the ability to also achieve a reduction in energy consumption [5]–[8]. The main challenges that have plagued passive devices can be summarized as the choice of the actuators, and the method and timing of the addition of energy into the gait cycle.

The highest amount of energy consumed during the walking pattern, also known as the gait cycle, is at the ankle and during push off [3], [9]–[11]; the instant the front foot is solidly planted on the ground and the back foot pushes on the ground. This large release of energy is done, mostly, by the elastic release of the Achilles tendon, which was charged during the gait cycle through various mechanisms. That is why most current unpowered assistive devices position a passive component in parallel with the Achilles tendon, and try to coordinate the release of that component's accumulated energy with the release of the Achilles tendon [5], [12]–[17].

The large amount of energy required at push off establishes the need for devices able to deliver a high amount of energy in a relatively short time. However, increases in weights at the ankle has been shown to increase the metabolic rate of walking [18], meaning that the device used must be high powered and lightweight. Continuing, placing a passive component in parallel with the Achilles tendon requires this passive component to behave in a similar fashion: muscle-tendon systems have a non-linear force-length behaviour [19], [20], which proves complicated to duplicate.

Pneumatic Artificial Muscles (PAMs) are pneumatic actuators composed of an internal elastic bladder around which an inelastic braided mesh is wrapped, as shown in Figure 1-1. End fixtures are used to seal the bladder and secure the braided mesh. Upon inflation, the bladder expands radially causing an increase in braid angle, resulting in the overall muscle to shorten.

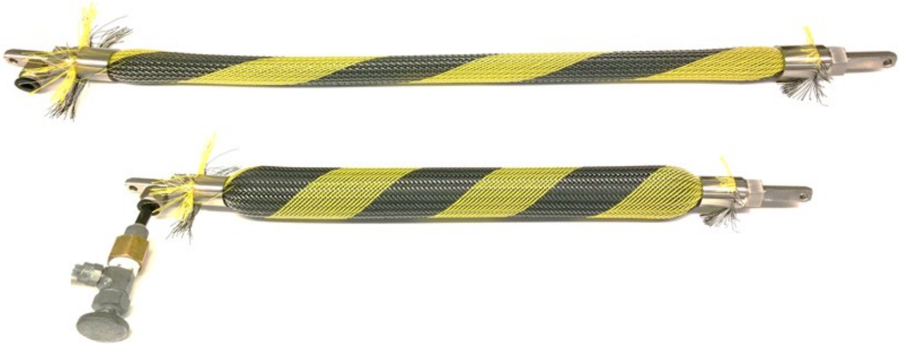


Figure 1-1: Pressurized PAM (top) and deflated PAM (bottom)

This actuator, when compared to biological muscles, presents similar mechanical characteristics. These are lightweight, approximately 0.1 to 0.2 kg [21], have the ability to create significant longitudinal forces [22], and present a non-linear force-length behaviour. These characteristics make it a great candidate for passive assistive devices.

The second main issue for walking assistive devices is the absence of a timing mechanism able to provide a propulsive force at the late stance of the walking cycle, the instant where the largest power is required during walking. This issue is the main reason most walking devices rely on complex electronic sensors and control systems to time the release of their devices, significantly increasing requirements for energy storage and on-board processing. The presence of a purely mechanical control system relying on the user's movement could present a significant advantage, in terms of weight, range of independence, maintenance, and complexity, compared to powered systems.

1.3 Objectives

The main objective of this thesis is to develop and mechanically validate an unpowered wearable ankle exoskeleton to reduce the energy consumption of human ambulation. The proposed device shall combine the passive nature of orthoses with the enhancing ability of exoskeleton to create a lightweight efficient walking assist device. Recognizing that the passive components selected and the timing of the release of the harvested energy are both essential considerations, the exoskeleton shall harness PAM as a passive actuator, and proposes a novel mechanical triggering device, relying on the leg's motion during gait to ensure proper timing of the released energy.

1.4 Thesis Contributions

This work has led to the following main research contributions:

1. A comprehensive biomechanical analysis of human gait has been achieved to characterize the mechanical behaviour requirements of a walking assist device
2. An unpowered ankle exoskeleton has been designed and fabricated to show the feasibility to reduce the power requirements at push-off. Figure 1-2 shows the conceptualized passive ankle exoskeleton.
3. The biomechanical contribution of the proposed ankle exoskeleton was analytically simulated and mechanically tested throughout the gait cycle
4. A novel mechanical locking and indexing mechanism has been designed, fabricated and mechanically tested to charge the PAM during the stance phase and release it at push-off.
5. A new stiffness model that accounts for PAM hysteresis and predicts more accurately the stiffness than previous models was developed. This model was validated experimentally

where the effect of various muscle parameters was evaluated for their impact on the muscle force-length behaviour

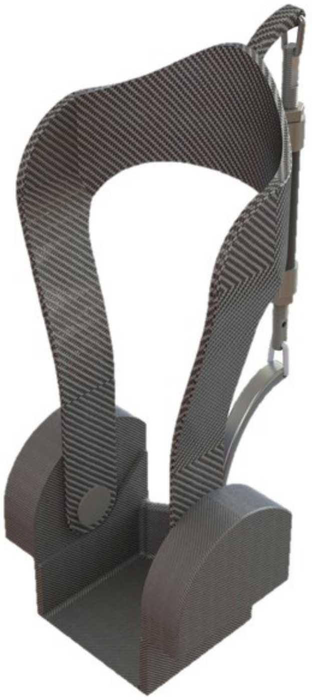


Figure 1-2: Proposed passive ankle exoskeleton

1.5 Thesis Outline

This thesis is organized into six chapters. Chapter 1 introduces the challenges currently facing the assistive technologies and provides the rational for the development of an unpowered ankle exoskeleton harnessing a PAM. Furthermore, a list of objectives and contributions is provided.

Chapter 2 presents results of a literature survey. This survey is organized into a review of human locomotion, current assistive technologies including exoskeletons, prostheses and orthoses, and PAM modeling and challenges.

Chapter 3 presents the development and validation of a PAM stiffness model, which considers various muscle parameters, such as its physical composition, geometry, and viscoelastic behaviour.

Chapter 4 presents a three-step design methodology used to develop the unpowered ankle exoskeleton, as well as its modeling, simulation and optimization in terms of material and dimensions. Chapter 5 presents the fabrication of an ankle exoskeleton prototype, experimental setups, experimental testing and the mechanical validation of the proposed exoskeleton to provide biomechanical contribution to the ankle joint at the push off phase.

Chapter 6 presents the conclusions for the thesis and recommendations toward future research for the passive ankle exoskeleton and PAM modeling.

Chapter 2 – LITERATURE REVIEW

In this chapter, a literature review of human locomotion, current motion assisting technologies and PAMs are presented.

In terms of human locomotion, a brief overview of the anatomy and physiology of the human legs is presented, followed by a review on the kinematics and kinetics of the ankle, knee and hip joints for walking, which is considered the main daily activity. Next, a brief literature survey on the current state of motion assistive technologies providing assistance at the ankle joint is presented. The technologies presented include prosthesis, orthoses, and exoskeletons, focusing on applications relating to rehabilitation, power augmentation, metabolic rate reduction, and movement generation. Lastly, although many different PAMs configuration exists, this review focuses on PAMs composed of a cylindrical elastic bladder, surrounded by an inelastic braided meshing and filled with air. This review introduces PAM operation, the existing force and stiffness models, and their limitations in terms of application to human mobility technologies.

2.1 Human Locomotion

The successful development of an ankle assistive device requires thorough understanding of human locomotion and biomechanical behaviour of lower limbs. This knowledge and analysis will assist in defining the design requirements allowing the walk assist device to contribute and work flawlessly with human mobility. This section first reviews the physiology and anatomy involved in human locomotion, and then presents current standard to quantify gait analysis, and anthropometric data.

2.1.1 Walking Physiology and Anatomy

The scope of the current anatomical and physiological review is limited to the basic reference position, joints and large muscle structures involved in ambulation. A complete review of the structure of the various tissues involved in ambulation is presented in [19]. Refer to the Glossary for anatomical and physiological nomenclature. The anatomical position and the three anatomical planes are shown in Figure 2-1.

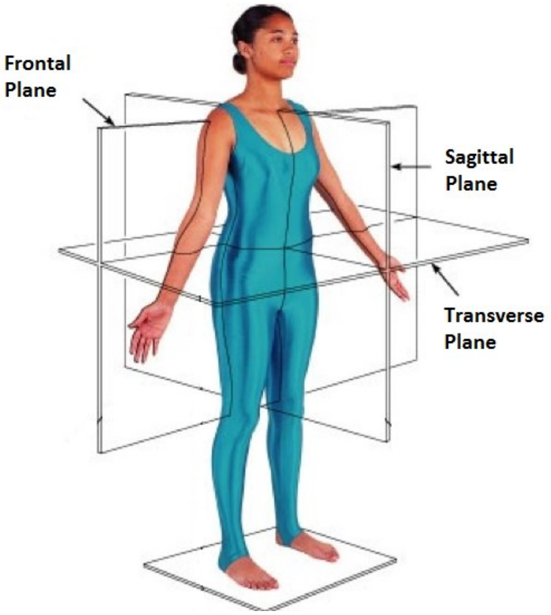


Figure 2-1: Anatomical planes **[19]** (modified)

While the whole body is involved in ambulation, it can be simplified as motions of the pelvis, thighs, shanks and feet, while mostly ignoring the motions of the head, torso and arms. Each thigh articulates on either side of the pelvis to form the hip joints. The shank articulates with the thigh to form the knee joint. The foot articulates with the shank to form the ankle joint. The structure of the bones in the foot creates another articulation, called the metatarsophalangeal joint, which arcs the foot and bends the toes.

The hip joint is similar to a ball and socket joint, allowing flexion and extension, abduction and adduction, and internal and external rotations. It is stabilized by a deep articulating cavity, the acetabulum, by a number of ligaments, and by considerable amount of muscles and tendons overlaying the articulation.

The knee joint can be considered as a polycentric hinge-like joint caused by the shape of the articulating surfaces of the femur, tibia and patella, which can perform flexion and extension, some internal rotation and very limited translation. The knee depends mostly on ligaments, tendons and muscles for its stability, making it prone to injuries.

The ankle joint presents loading dependent behaviour, due to the presence of irregular bones. In general, the ankle motions are summarized to flexion and extension, eversion and inversion, and internal and external rotation. Motion occurs at the high ankle (talocrural joint) when weight is applied and at the lower ankle (subtalar joint) without weight applied. The stability of the ankle is due to a combination of the bone articulating relatively well and soft tissue surround the joint. Although the ankle is prone to injuries, these usually occur under with the eversion or inversion movement. Finally, the metatarsophalangeal joint is akin to multiple misaligned hinge joints acting in unison. The movements and association range of motion for each joint are shown in Table 2-1.

Table 2-1: Normal range of motion for the joints in ambulation **[23]–[25]**

Joint	Movement	Range of Motion (Degrees)
Hip	Flexion/Extension	120/30
	Abduction/Adduction	45/40
	Internal/External Rotation	40/45
Knee	Flexion/Extension	130/15
	Internal Rotation	10
Ankle	Plantar flexion/Dorsiflexion	50/20
	Eversion/Inversion	30/20
	Internal/External Rotation	5/10
Metatarsophalangeal	Flexion/Extension	45/70

2.1.2 Gait Analysis

Simple in appearance, human ambulation is a very complex mechanism. Therefore, prior to designing the exoskeleton, a thorough biomechanical analysis of walking, or gait, is presented. This motion represents the most common and recurrent movements a person has to face during daily activities.

In all ambulation scenarios, certain parameters can be measured to obtain a better understanding of an individual's gait. The kinematics of ambulation can be divided into spatial and temporal parameters. The former reflect physical dimensions of ambulation, whereas the latter accounts for time. As defined in Table 2-2 and shown in Figure 2-2, each parameter can be greatly affected by the terrain, incline, and individual height and weight.

Table 2-2: Temporal parameters of gait analysis

Temporal Parameters	Definition
Step Time	Time required for one step
Stride Time	Time required for one stride
Stance Duration	Time of the stance phase for one foot
Swing Duration	Time of the swing phase for one foot
Speed	Horizontal distance divided by time
Cadence	Number of steps per minute

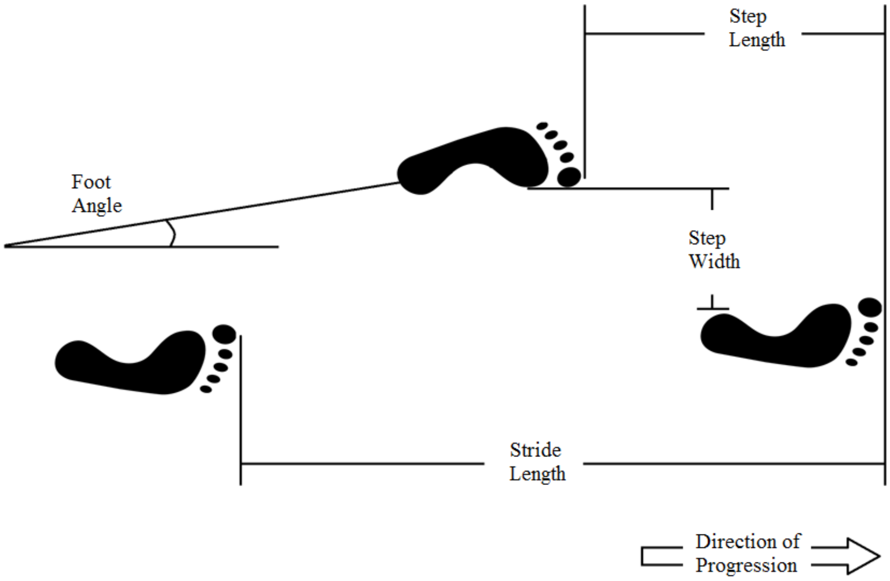


Figure 2-2: Spatial parameters of gait analysis

Although walking occurs in multiple planes, the present analyses only consider sagittal plane motions because the largest motions, forces and moments occur in that plane. Figure 2-3 indicates the sign convention for the angle measurements of each joint in the sagittal plane, which will be adopted in this work. The positive angle is counter-clockwise and is measured as the movement of the distal segment relative to the proximal segment.

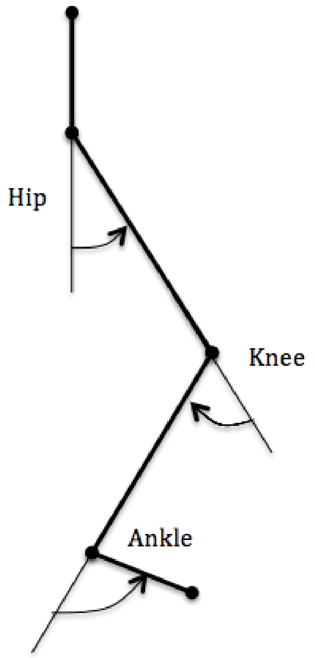


Figure 2-3: Sign convention for gait

The two principal biomechanical analysis are musculoskeletal model based approach and an inverse dynamics approach [27]. The first approach relies on an optimized muscle mathematical model for an overall body movement, generating muscle loadings, joint loadings, kinetics and kinematics. The second approach determines joint loadings based on ground reaction forces, kinematic measurements and an anthropomorphic rigid link model. The simplicity and ability to parameterize the inverse dynamics approach makes it better suited to exoskeleton development. This technique is used in the development of the ankle exoskeleton device.

2.1.2.1Walking

Walking is the most common type of locomotion for human, characterized by a swing phase and a stance phase, corresponding to about 40% and 60% of the cycle, respectively, as shown in Figure 2-4. At any point during the walking cycle, there are one or two legs in contact with the ground, called single and double support, respectively. The walking cycle timing of each event may vary for each individual depending on speed, weight, ground surface and so on.

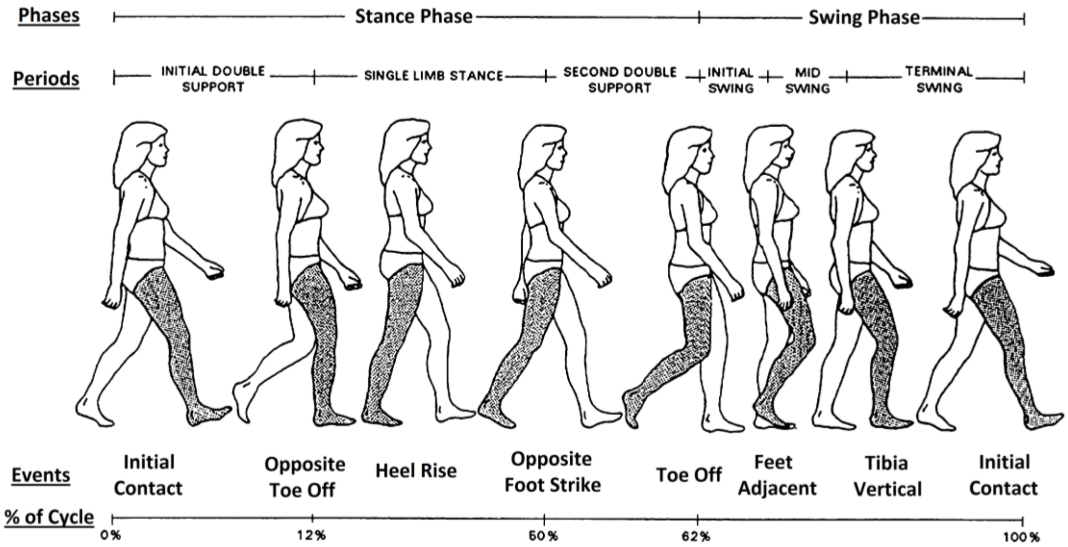
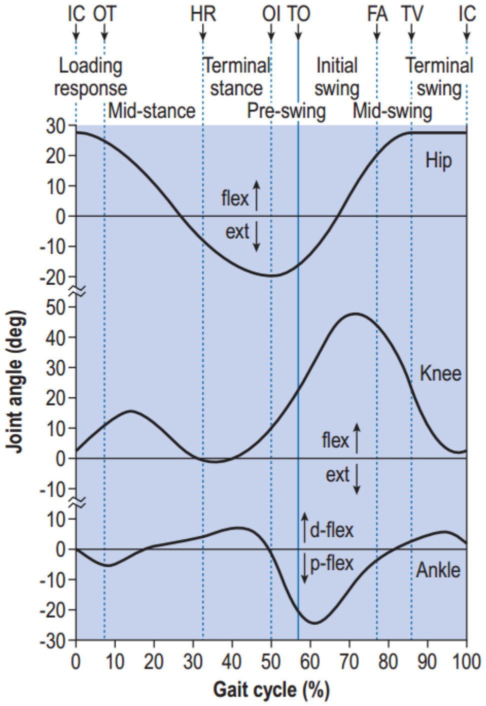


Figure 2-4: Human walking cycle **[28]** (modified)

By analysing the walking cycle in more detail, it can be seen that the center of gravity shifts vertically along a sinusoidal curve and horizontally along a second smaller sinusoidal. As such, walking has been modeled as an inverted pendulum [27], [29], where 60-70% of the energy is conserved. However, this model grossly misestimates the ground reaction forces due to a lack of compliance in the model [27].

One of the most common methods of analyzing gait is the use of gait graphs. These are visual representations of angle, moment and power distribution at the hip, knee and ankle joint, normalized for bodyweight, and cycle duration [3], [11], as shown in Figure 2-5 and Figure 2-6. These graphs allow quick comparison between a user's gait and that of the average population, to identify muscles weakness, injury risks, or other walking problem.



IC: initial contact

OT: opposite toe off

HR: heel rise

OI: opposite foot strike

TO: toe off

FA: feet adjacent

TV: tibia vertical.

Figure 2-5: Angle gait graphs **[11]**

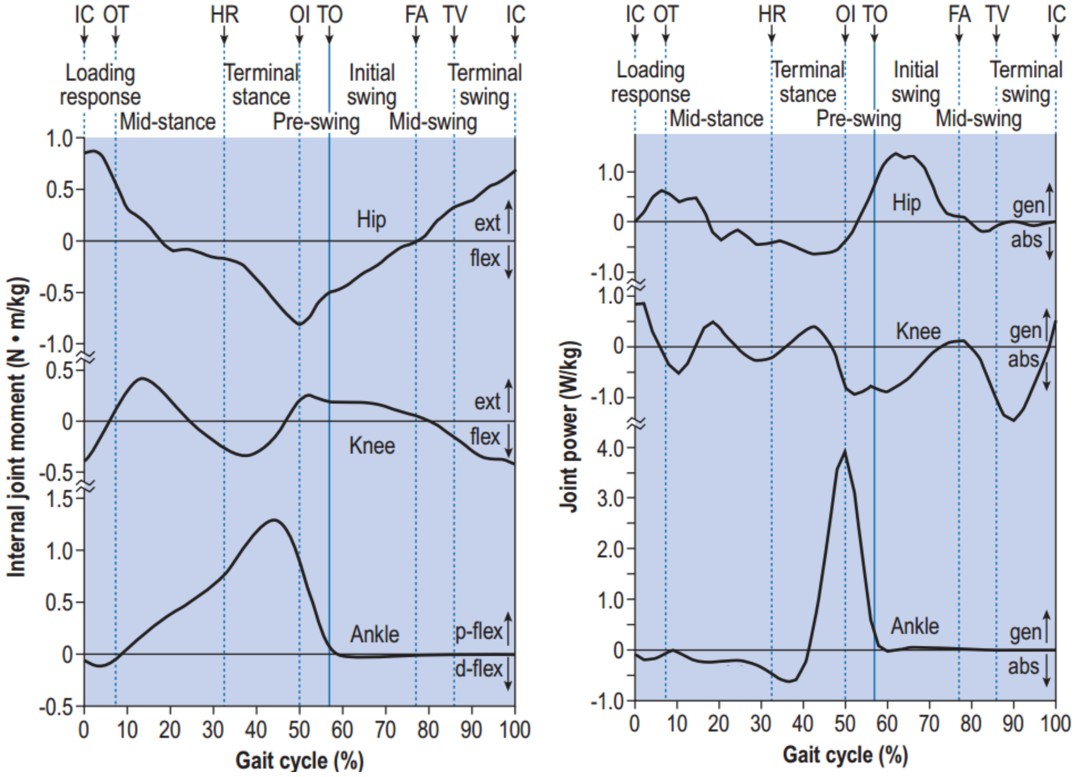


Figure 2-6: Moment and power gait graphs **[11]**

Upon closer investigation of the joint angles in Figure 2-5, the ranges of motion of each joint during walking do not approach the maximum joint range of motion, shown in

Table 2-3, as expected. In terms of angular motion, the hip goes through one flexion and extension motion through the gait cycle. The knee goes through two flexion motions during the same cycle: once during the early stance phase and one during the swing phase. The ankle movements are small during the stance phase, followed by a large plantar flexion motion during at toe-off.

In terms of joint power, the hip and knee have both energy absorption and generation periods whereas the ankle has one large peak at the toe off to provide the energy required to propel the individual forward and swing the leg. This power peak corresponds to the largest power consumption during the gait cycle of any joint. The absence of power and moment in the ankle during the swing phase suggests natural passive elements are involved in creating ankle motion during the swing phase [27], [29].

When designing the exoskeleton, the maximum joint range, moments and power must be taken into considerations. Reviewing the gait cycle, Table 2-3 illustrates the maximum angle, moment and power of each joint for each movement.

Table 2-3: Gait maximum joint angle, moment and power **[11], [30]–[32]**

Movement	Characteristic	Hip	Knee	Ankle
Walking	Angle (º)	28/-20	47/0	10/-25
	Moment (Nm/kg)	0.8/-0.8	0.5/-0.4	1.25/-0.1
	Power (W/kg)	1.25/-0.5	0.8/-1.5	4/-0.8

2.1.3 Metabolic Rate and Cost of Transport

Metabolism is the sum of all the chemical reactions in the body, including degradation, synthesis and energy transformation [19]. Since metabolism cannot be directly measured, the metabolic rate is most commonly estimated by measuring the oxygen and carbon dioxide volumetric rate $\dot{V}O_2$ and $\dot{V}C_2$ respectively, and applying the empirical relationship (2-1) [33]. The Cost Of Transport COT is the energy cost required to move. It is defined in (2-2) as the ratio between the metabolic rate and the work of moving the bodyweight BW over a certain distance d.

$$\dot{E}_{metabolic} = 16.58\,\dot{V}O_2 + 4.51\,\dot{V}CO_2 \tag{2-1}$$

$$COT = \frac{\dot{E}_{metabolic}}{BW \cdot d} \tag{2-2}$$

Many researchers have tried to relate the metabolic cost of propulsion to different forces applied on the body, namely added weight, removed weight, forward assisting forces and backward holding forces. In 2003, Gottschall and Kram [34], using the set up shown in Figure 2-7, tested the effect of impeding and assisting horizontal forces, applied at the waist, on the oxygen consumption. Both situations were tested with forces of 5, 10 and 15% of the participant's bodyweight.

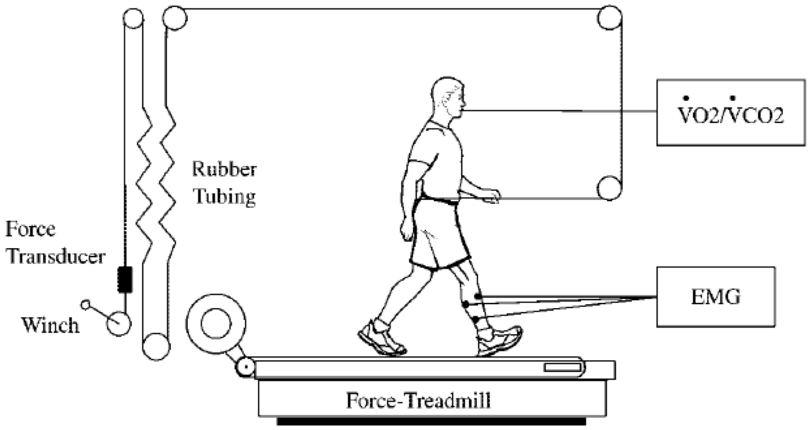


Figure 2-7: Horizontal impeding and assisting force set-up **[34]**

The results of the experiment, shown in Figure 2-8, presented a linear increase in metabolic rate for increasing horizontal impeding forces, and a non-linear decrease in metabolic rate for decreasing horizontal forces. The maximum decrease occurred with a 10% bodyweight assisting force, resulting in a 53% reduction in metabolic rate.

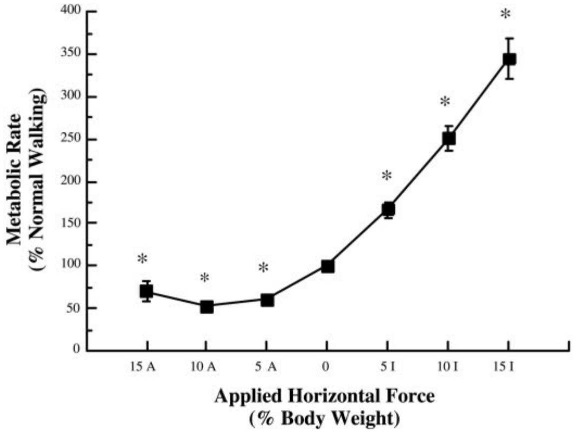


Figure 2-8: Normalized metabolic rates against horizontal forces (*P≤0.05) **[34]**

In 1992, Farley and McMahon [35] evaluated the link between the gravity and the metabolic cost of walking, using the apparatus shown in Figure 2-9, which applies vertical forces to reduce the effect of the individuals bodyweight, simulating the effect of reduced gravity. The metabolic rate was evaluated for simulated gravities of 0.25g, 0.50g, 0.75g and 1.0g. This study found that the metabolic rate decreases almost linearly with the removal of sensed bodyweight when walking. Specifically, a 75% felt bodyweight reduced the metabolic rate by 11%. The reduction was of 25% for a felt bodyweight of 50% and 33% for a felt bodyweight of 25%.

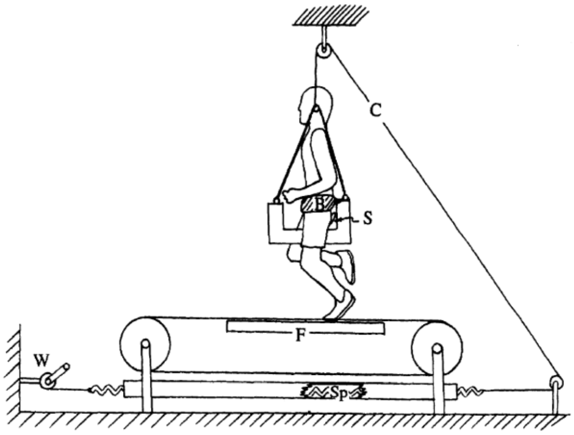


Figure 2-9: Vertical impeding and assisting force set up **[35]**

In 2003, Griffin *et al* [18] studied the effect of increased loading on the metabolic rate, at various walking speed. These researchers found that the metabolic rate increases approximately linearly with increase in added mass, placed symmetrically around the waist, while walking irrespective of speed. Quantitatively, a 30% increase in body mass created a 47±17% increase in metabolic rate.

Other studies have shown that adding weights on an individual will cause variations of gait. For example, someone carrying a backpack has a tendency to lean forward [36], and distributed loads over the leg length increase the hip gait angles and the angular velocity of the ankle [36].

Synthetizing these results suggest that to reduce the metabolic rate of individuals walking, an assistive propulsion force applied equivalent to 10% of the bodyweight must be applied and the bodyweight felt by the user should be reduced as much as possible. Furthermore, any added weight can significantly impact gait patterns and the weight distribution of any apparatus on the legs should be carefully considered to minimize gait deviations.

2.1.4 Anthropometric Data

For the design of the exoskeleton, normalized dimensions of the human body segments in terms of a user's height and weight are valuable. Although each person has slightly different limb dimensions, population averages are available, as shown in Figure 2-10, Table 2-4 and Table 2-5.

The intention is to use the anthropometric data to create a parameterized conceptual passive ankle exoskeleton model so as to optimize it before fabricating the prototype. The parameterized model will be able to provide exoskeletons dimensions for the 5th percentile women to the 95th percentile ground troops. Even though the targeted population is not ground troops, the population's measurements are assumed to fall within the same anthropometric range.

Table 2-4: Anthropometric data **[53] (**Modified)

Segment	**Definition**	**Segment Weight/Total Body Weight**	**Center of Mass/ Segment Length**		**Radius of Gyration/ Segment Length**			**Density**
			Prox.	**Distal**	**C of G**	**Prox.**	**Distal**	
Total arm	Glenohumeral axis/ ulnar styloid	0.05	0.53	0.47	0.368	0.645	0.596	1.11
Foot	Lateral malleolus/ head metatarsal II	0.0145	0.5	0.5	0.475	0.69	0.69	1.1
Leg	Femoral condyle/ medial malleolus	0.0465	0.433	0.567	0.302	0.528	0.643	1.09
Thigh	Greater trochanter/ femoral condyle	0.1	0.433	0.567	0.323	0.54	0.653	1.05
Foot and leg	Femoral condyle/ medial malleolus	0.061	0.606	0.394	0.416	0.735	0.572	1.09
Total leg	Greater trochanter/ medial malleolus	0.161	0.447	0.553	0.326	0.56	0.65	1.06
Pelvis	L4-L5/ greater trochanter*	0.142	0.105	0.895	-	-	-	-
Trunk	Greater trochanter/ glenohumeral joint*	0.497	0.5	0.5	-	-	-	1.03
HAT	Greater trochanter/ glenohumeral joint*	0.678	0.626	0.374	0.496	0.798	0.621	-
***These segments are presented relative to the length between the great trochanter and the glenohumeral joint**								

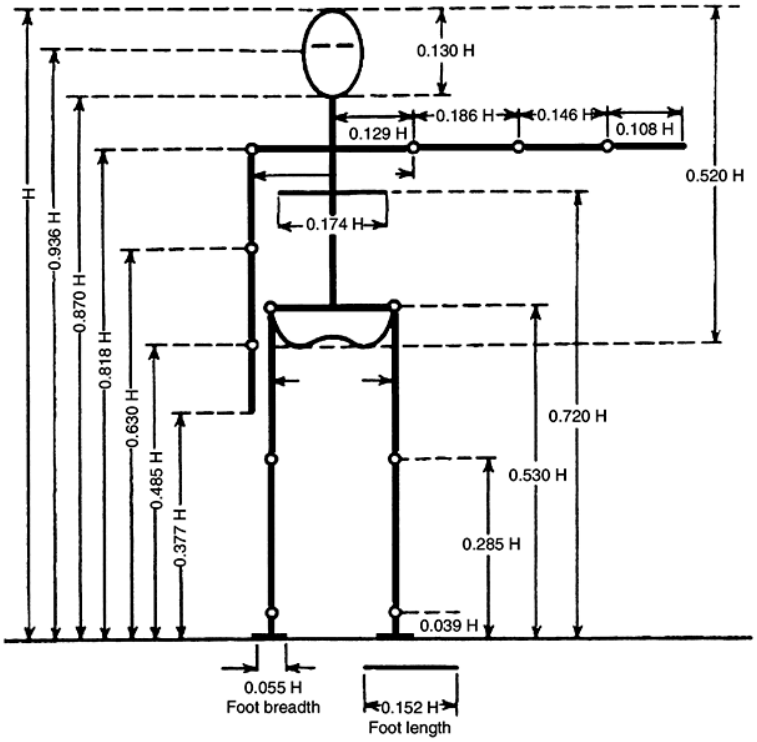


Figure 2-10: Anthropometric dimensions based on total height H **[52]**

Table 2-5: Military personnel percentile distribution **[23]** (modified)

Percentile values, cm						
	5th percentile			95th percentile		
	Ground troops	Aviators	Women	Ground troops	Aviators	Women
Weight, kg	65.5	60.4	46.6	91.6	96	74.5
Standing body dimensions						
Stature	162.8	164.2	152.4	185.6	187.7	174.1
Eye height (standing)	151.1	152.1	140.9	173.3	175.2	162.2
Shoulder (acromial) height	133.6	133.3	123	154.2	154.8	143.7
Chest (nipple) height	117.9	120.8	109.3	136.5	138.5	127.6
Elbow (radiate) height	101	104.8	94.9	117.8	120	110.7
Fingertip (dactylion) height	-	61.5	-	-	73.2	-
Waist height	96.6	97.6	93.1	115.2	115.1	110.3
Crotch height	76.3	74.7	68.1	91.8	92	83.9
Gluteal furrow height	73.3	74.6	66.4	87.7	88.1	81
Kneecap height	47.5	46.8	43.8	58.6	57.8	52.5
Calf height	31.1	30.9	29	40.6	39.3	36.6
Functional reach	72.6	73.1	64	90.9	87	80.4
Functional reach, extended	84.2	82.3	73.5	101.2	97.3	92.7

2.1.5 Joint Stiffness

When considering a passive walking assist device, it is important to respect the natural joint passive properties, such as the joint stiffness. The joint stiffness k_t has been defined passively and dynamically. Both are defined as being variation of joint moment M_θ with regards to the joint angle θ.

$$K_t = \frac{\Delta M_\theta}{\Delta \theta} \tag{2-3}$$

Passive joint stiffness is calculated using external apparatuses that apply moments to the desired joint, all while muscles are kept relaxed [37], [38], thereby measuring the passive properties of soft tissues. The dynamic joint stiffness with experimental joint moment and angle, measured in a biomechanical laboratory [39]–[49]. Consequently, the dynamic joint stiffness takes into account the passive properties of the various tissues, and inertial and gravity forces. For the design of an exoskeleton, the dynamic joint stiffness is more relevant because it is related to the actual efforts involved in ambulation.

Using data available in literature, the ankle stiffness was calculated by curve-fitting the linear portions of the ankle moment-angle graph. Granting this method yields unique results for each individual, it can provide an estimate of the ankle joint stiffness. The presented analysis was done using biomechanical data published by Dr. D.A. Winter [9], digitized and posted online [10]. The data represents a female volunteer (56.7 kg) walking at low speed.

The ankle has non-zero moments only during the stance phase of the gait cycle. As shown in Figure 2-11, there are 3 linear sections, which occur during Loading Response to Mid-Stance, during Terminal Stance and during Pre-Swing. The ankle has an approximate dynamic stiffness of 0.0576 Nm/kg/° during the Loading Response to Mid-Stance portion and an approximate dynamic stiffness of 0.0687 Nm/kg/° during the Pre-Swing portion.

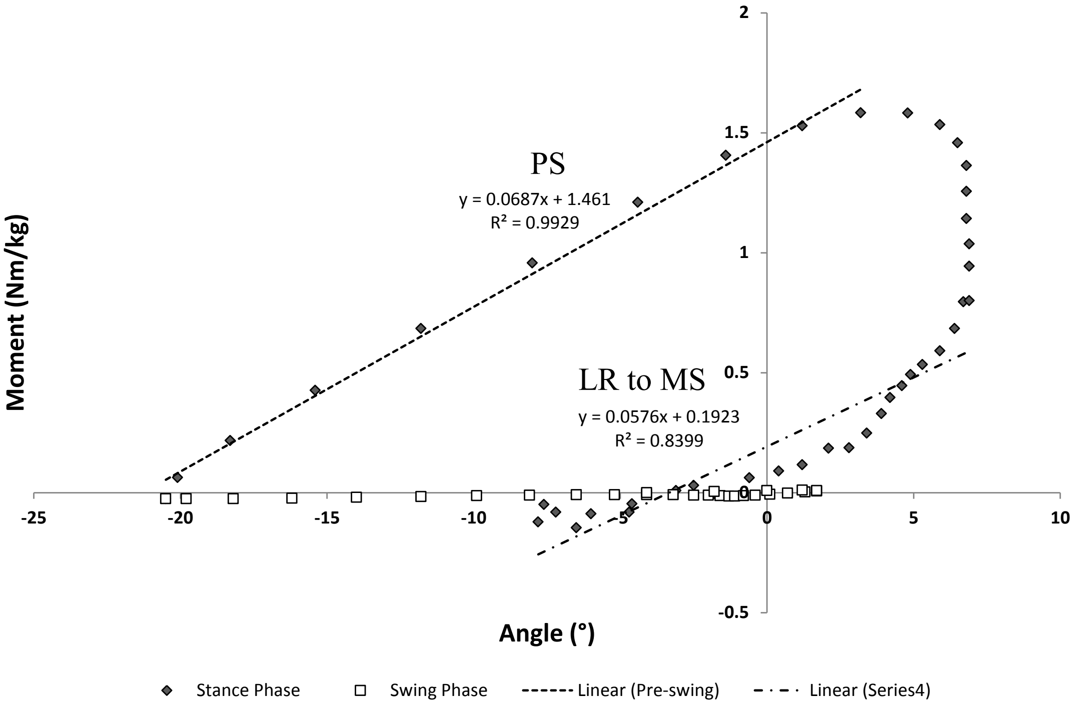


Figure 2-11: Ankle stiffness throughout gait cycle with pre-swing (PS) and loading response (LR to MS)

Previous researchers have found similar results (see Table 2-6), thereby validating the results of the current analysis. Therefore, a dynamic stiffness of 0.058 Nm/kg/º will be used as a benchmark to evaluate the proposed PAM actuator for the exoskeleton design.

Table 2-6: Ankle dynamic stiffness literature comparison

Group [Source]	Ankle Dynamic Stiffness (Nm/kg/º) Loading Response to Mid-Stance	Pre-Swing
Women [9], [10]	0.0576	0.0687
Children [40]	0.0598	-
Men/Women [42]	0.0596/0.0511	0.0844/0.0691
Children [47]	0.0384	0.0668

2.2 Human Locomotion Assist Devices

Recent findings have found strong links between the ability to move independently and quality of life [50]. Since human ambulation is closely linked to individual autonomy, many technological solutions have been devised to increase independence for physically impaired individuals. Therefore, devices that allow biological-like walking locomotion are desirable, namely prostheses, orthoses, and exoskeletons.

While the scope of this thesis consists of unpowered ankle exoskeletons, limited such devices are found in literature. Therefore, the review will cover prostheses, orthoses and exoskeletons, with a specific focus on the ankle joint and reduction of metabolic consumption, two parameters important to the development of the proposed exoskeleton.

2.2.1 Prostheses

Conventional prostheses are aimed at restoring basic functions for amputees and they are usually categorized by the amputation location. For this review, transtibial and transfemoral prostheses are considered and presented in chronological order. Quasi-passive prostheses are mostly passive but rely on actuators for damping purposes, whereas active prostheses rely on actuators to perform motions.

2.2.1.1Quasi-Passive Prostheses

In 2010, Collins and Kuo [51] designed the foot component of a quasi-passive transtibial prosthesis, with the intention of decreasing the walking energy consumption to non-amputee levels. To achieve their objective, an energy capture and storage system based series-elastic actuators was developed, a schematic of which is shown in Figure 2-12. At heel strike, the rear-foot compresses the back spring until a clutch system locks the rear-foot and back spring in place. When a large load is detected on the forefoot, the clutch releases the back spring, providing toe-off energy for the swing phase, during which a small spring resets the position of the forefoot. Experimental results have shown that the prosthesis performs better than conventional prosthesis but still 14% worse than non-amputees, in terms of metabolic consumption. Although using motors for its timing, the simple clutch locking mechanism suggests that a simple mechanical system can produce the desired gait motions.

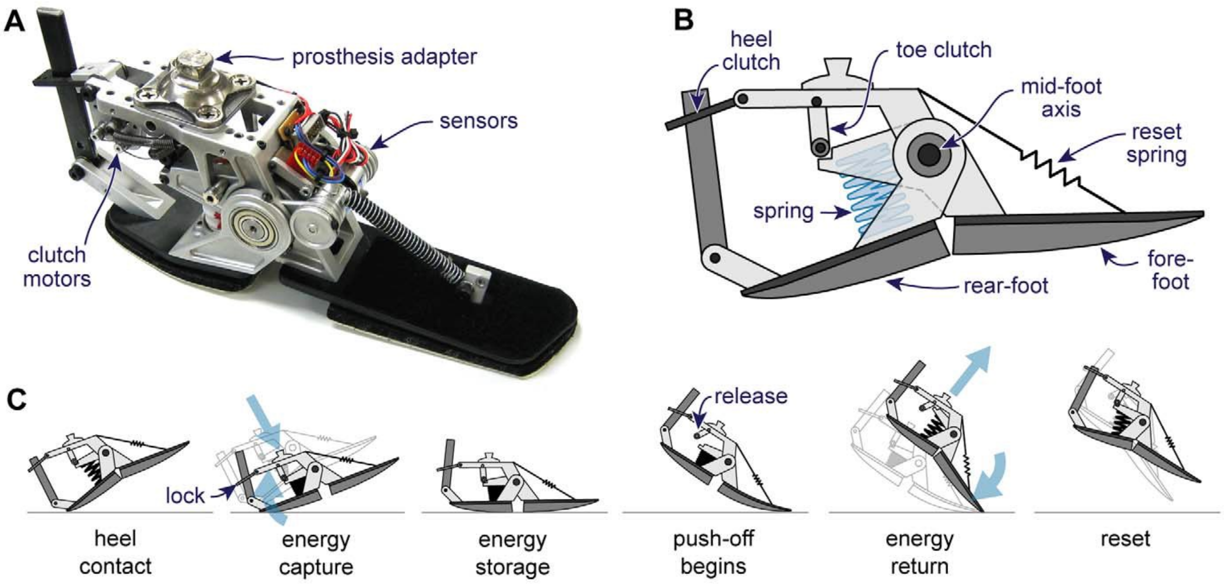


Figure 2-12: Recycling energy prosthesis **[51]**

In 2012, Cherell *et al* [12], [52], developed a minimally actuated transtibial prosthesis, the AMP-Foot 2.0, as shown in Figure 2-13. The goal of this prosthesis was to mimic the power of a normal ankle during the walking cycle so as to reduce the metabolic rate of walking while minimizing the amount of external power needed. The AMP-Foot 2.0 uses a double spring system to harvest energy throughout the gait cycle and redistribute it at toe-off. The first spring is associated with a series-elastic actuator, where the plantar flexor spring accumulates energy at heel-strike, and throughout the stance phase with an electric motor. The second spring is a toe-off spring used with a four-bar mechanism, which accumulates energy from the plantar flexor spring and releases it at the toe-off. Experimental results indicate that the prosthesis is able to duplicate the natural motion of the ankle and of the toes, while providing sufficient energy through the gait cycle. These results need to be confirmed with further clinical studies.

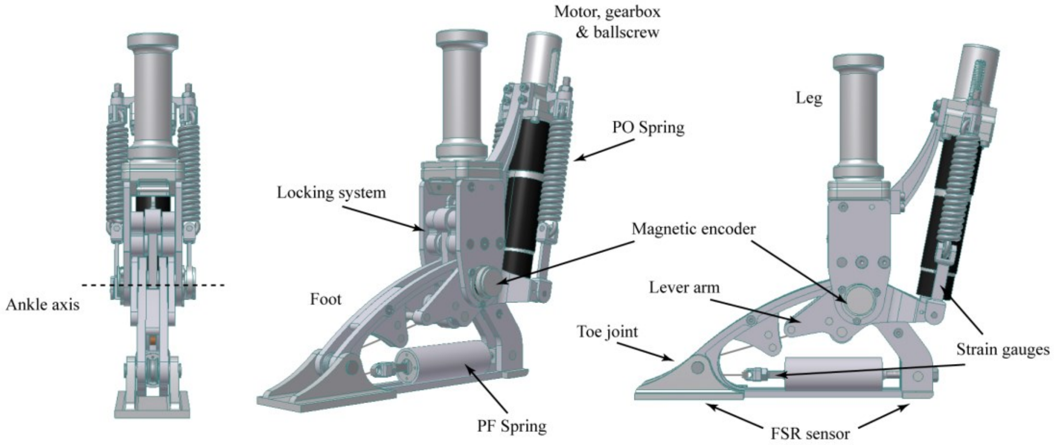


Figure 2-13: AMP-foot 2.0 design **[12]**

2.2.1.2Active Prostheses

In 2009, Au *et al* [53] developed a powered active transtibial prosthesis that attempts to lower the metabolic consumption of walking compared to other transtibial prostheses. The MIT prosthesis, based on series-elastic actuators, provides power at toe-off comparable to the normal ankle. It also uses a leaf spring to provide shock absorption at heel strike and release energy throughout the stance phase. Experimental testing has shown that this prosthesis reduces the metabolic cost of walking by 14% when compared to conventional prosthesis. This suggests that reducing the amount of energy required by the ankle during toe-off leads to a reduction in metabolic consumption.

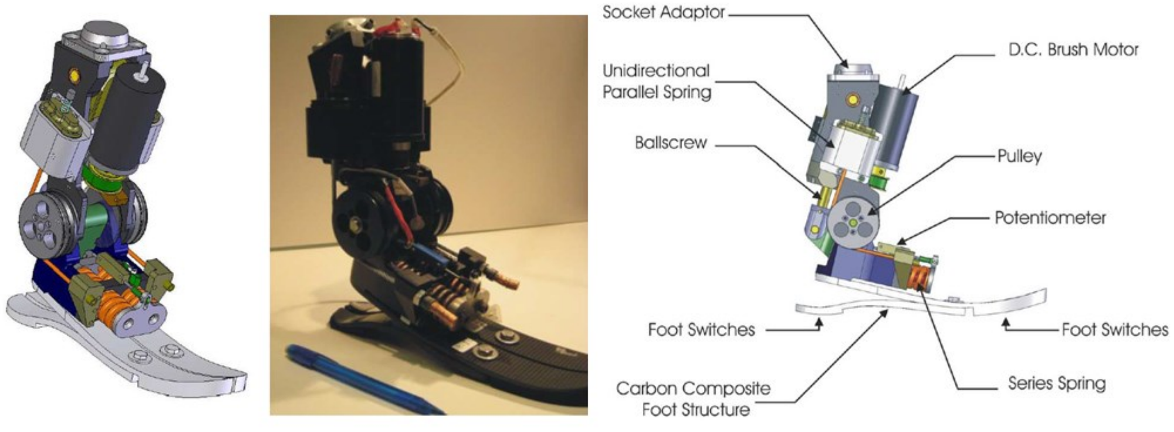


Figure 2-14: MIT prosthesis **[53]**

In 2009, Versluys *et al* developed a transtibial prosthesis, using antagonistically placed pleated pneumatic artificial muscles (PPAMs) to generate dorsiflexion and plantar flexion. Also, the use of two PPAMs at the back of the device, as shown in Figure 2-15, allows some ankle inversion/eversion motions. The device was designed without an onboard power system or local pressure source, thereby limiting this device to use in rehabilitation.

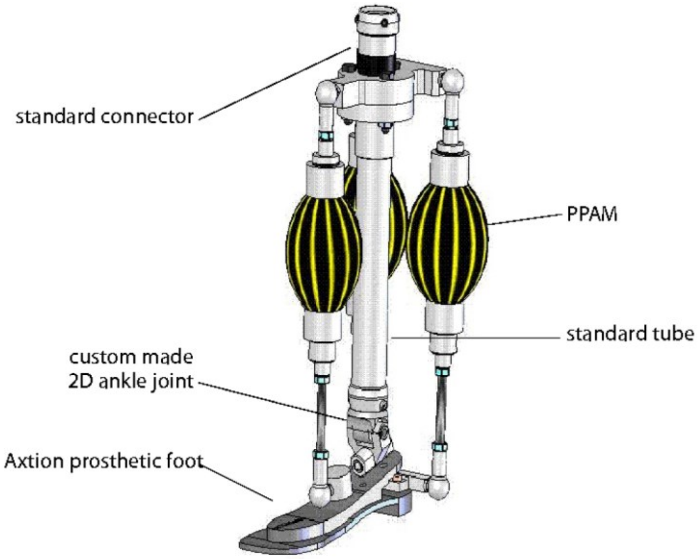


Figure 2-15: PPAM transtibial prosthesis **[13]**

2.2.2 Orthoses and Single-Joint Exoskeletons

In general, orthoses are passive systems aimed at protecting and restoring some level of gait motion by providing support through a molded shell, made of carbon fiber, thermoplastics or other [54]. However, orthoses have started to move toward actuation to restore normal gait patterns. The actuation of orthoses is still at its early stages and the gold standard remains, on the most part, passive devices. At the moment, orthoses are custom made by orthotists making their quality and functionality in direct link with his/her proficiency. For this review, a selection of commercial orthoses will be presented.

2.2.2.1Passive Orthoses

The most common orthoses are structural support for joints to alleviate pain or correct position, in the aftermath of knee injuries, chronic pain or other condition. Figure 2-16 shows a typical ankle-foot orthosis. It fits around the shank and in the shoes so as to prevent foot-drop, a condition where the dorsiflexors are weakened or incapacitated.

Figure 2-16: Typical ankle-foot orthosis from Ottobock **[55]**

In 2006, Hirai *et al* [56] developed a passive ankle-foot orthosis that uses a novel pneumatic element. The novelty of this device is the use of pneumatic elements to provide a constraining force to lock the ankle in a neutral position. The pneumatic element is composed of laminated sheets enclosed within an airtight chamber, with a rotation axis in the middle of the chamber. The device is pressurized when the air chamber located under the sole of the foot is compressed during stance phase. Experimental results have concluded that the orthosis behaves similarly to traditional orthosis. However, the constraining force can be modulated by changing the size and location of the air chamber under the sole of the user.

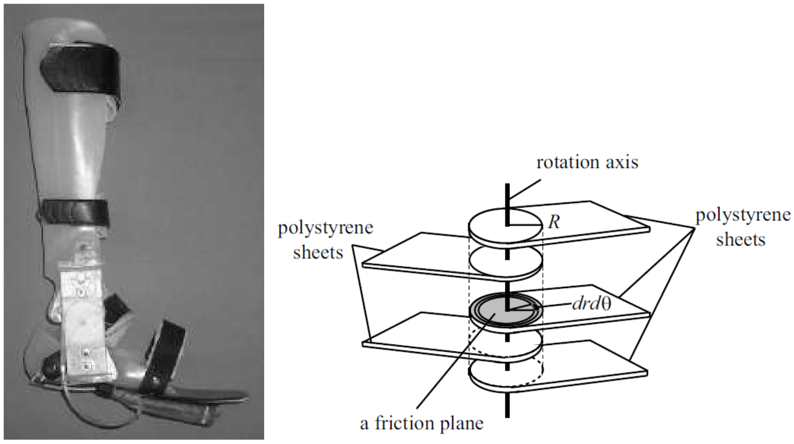


Figure 2-17: Passive pneumatic element (left) and AFO device (right) **[56]**

2.2.2.2Active Orthoses and Partial Exoskeletons

In 2005 and 2009, Ferris *et al* [14], [57], [58] developed a powered ankle-foot orthosis that can be used for rehabilitation. The orthosis is pneumatically powered by 2 PAMs, positioned at the back and front of the lower leg, aligned with the Achilles tendon. Experimental results suggest the ankle-foot orthosis is able to generate 60% to 70% of the required plantar flexion for paraplegics when using sensors underneath the sole of the user to activate the PAMs.

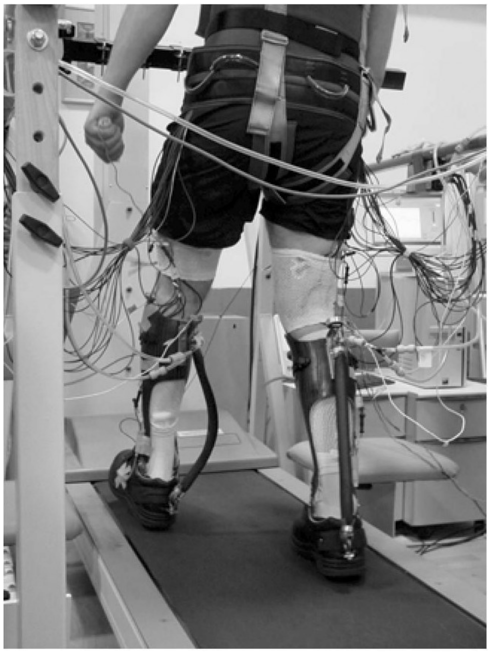


Figure 2-18: Pneumatic AFO and knee orthosis used in rehabilitation **[14], [57], [58]**

In 2005, A. Agrawal *et al* [59] designed an electrically actuated ankle-foot orthoses. The proposed 2.5kg orthoses provides actuation to the plantar flexion/dorsiflexion and eversion/inversion, along their respective natural axes of rotation, by measuring each user to find the axes experimentally. In theory, providing movement along the natural axis of rotation could be beneficial to promote normal gait patterns and limit forces due to the interaction between user and orthosis. However, this results in a bulky device, requiring significant adjustment time.

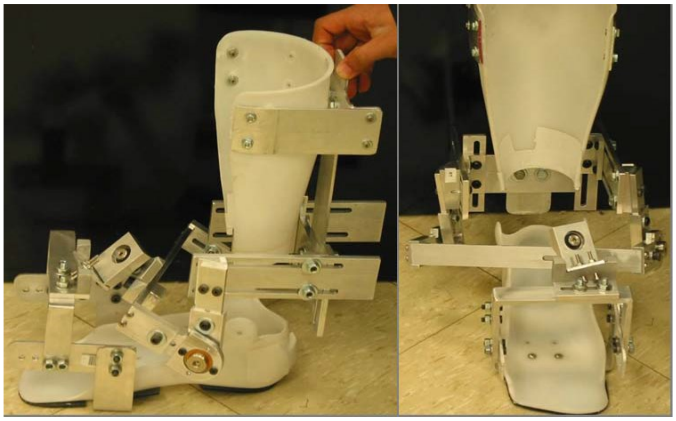


Figure 2-19: Two degree of freedom actuated AFO **[59]**

In 2007, Hitt *et al* [15] developed an actuated ankle-foot orthosis with dynamic control, as shown in Figure 2-20. A series-elastic actuator provides electric actuation to the ankle plantar flexion and dorsiflexion movements. The actuator was dubbed the "Robotic Tendon" due to its position at the back of the leg, in parallel with the Achilles tendon. The spring is compressed at heel strike by the user's weight and by the motor through the stance phase, and then releases its energy from late stance phase to toe-off. Experimental results have shown reduction in energy consumption on the order of 2.4 to 2.7 times the energy consumed by an ankle-foot orthosis with purely electrical actuator.

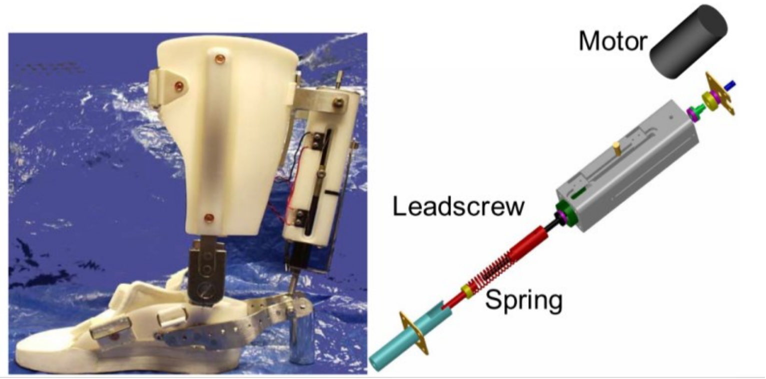


Figure 2-20: Hitt's robotic tendon orthosis **[15]**

In 2011, Carberry *et al* [16] built and tested a pneumatically actuated ankle-foot orthosis for rehabilitation purposes providing two degrees of freedom, plantar/dorsiflexion and inversion/eversion. As shown in Figure 2-21, dorsiflexion is actuated through a "diamond linkage" system that transforms a pulling force into a compressive force applied to the heel. When the upper portion of the diamond is pulled, it causes the diamond linkage to reduce its width and increase its height, applying a downward force on the ground, causing dorsiflexion motion and increasing toe-off force. Experimental testing showed that the orthosis caused minor reductions in stride length and stride velocity, which the researchers suggested was linked to the devices weight.

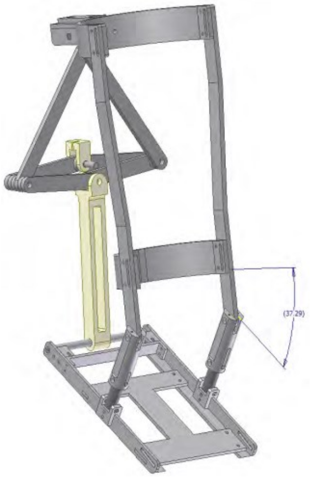


Figure 2-21: Pneumatic two degrees of freedom orthosis **[16]**

In 2011, Zhu *et al* developed the EXO-PANTOE 1, shown in Figure 2-22, an ankle exoskeleton that enhances ankle motions with series-elastic actuators. The device has a segmented sole to provide independent movements of the ankle and toes, allowing the full actuation of the plantar flexion and dorsiflexion movements. One of the actuators is located at the back of the ankle, parallel to the Achilles tendon, while the other is located within a specially designed cavity inside the exoskeleton sole to actuate the toes. Experimental results for level ground walking indicate that the use of the toe segment significantly reduces device and user energy consumption compared to a device without toe flexibility.

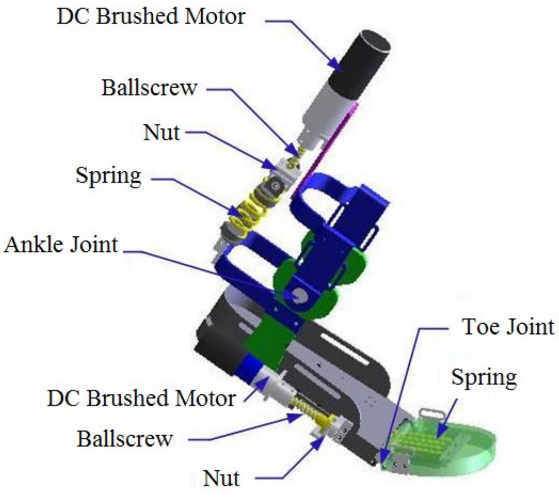


Figure 2-22: EXO-PANTOE 1 **[17]** (modified)

In 2011 and 2015, Wiggin, Collins *et al* [5], [60] developed a passive ankle exoskeleton aimed at reducing the metabolic cost of walking. This device proposes to harvest energy from the heel strike and early stance phase and redistribute it at toe-off, using a spring and a passive clutch system. The clutch system, shown in Figure 2-24, locks and unlocks the spring at specific intervals, thereby providing a feasible mechanical alternative to electronic control systems. Results indicate that the reduction of the metabolic rate of 7.2 ± 2.6% was achieved. This suggests not only that metabolic reductions are possible with passive exoskeletons, but also that the activation of the clutch system and the spring stiffness are critical factors to consider when designing and fabricating such devices.

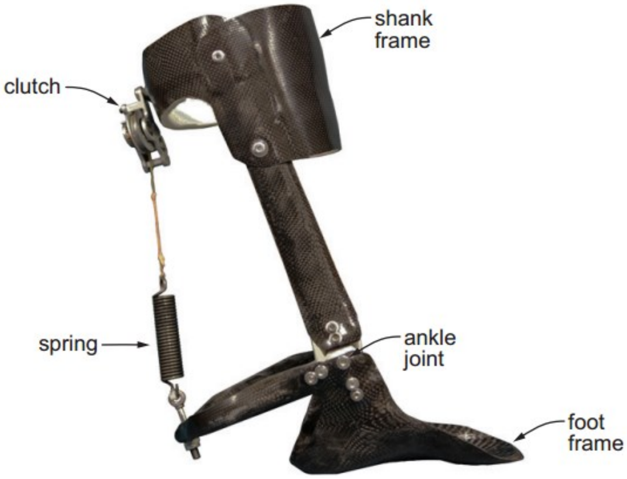


Figure 2-23: Proposed ankle exoskeleton with clutch mechanism **[5]**

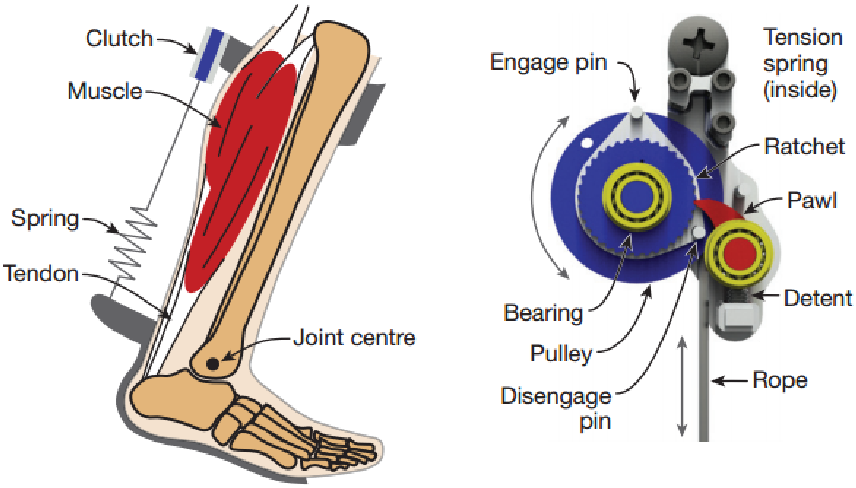


Figure 2-24: Clutch mechanism **[5]**

2.2.3 Exoskeletons

Exoskeletons are devices that have a structure running parallel to the human body, enhance human abilities, and protect and support the user. Since the first exoskeleton patent in 1890 [61], there has been numerous wave of exoskeleton research. Most notably are 1970’s development with GE’s Hardiman project [62], Seireg’s exoskeleton for paraplegics

[63], and the Pupin Institute's exoskeleton [64]. Problems with energy storage, the use of heavy materials and computing speed prevented further development.

Over the last decade, there has been a significant increase in exoskeleton development due to technological advances combined with increase funding, furthered gait understanding, and a shift of focus from movement restoration to power augmentation. Currently, the main challenge facing the widespread adoption of this technology is the availability of lightweight energy storage. The current review covers a selection of lower limb exoskeletons, presented in terms of their actuation.

2.2.3.1Passive and Quasi-Passive Exoskeletons

The most recognized quasi-passive exoskeleton was developed by Walsh *et al* at MIT [6], [65]–[67]. This device was designed for load carrying augmentation and reducing the metabolic rate of walking with a load. Optimizing a theoretical leg model for lowest moment, where each joint is modeled as a combination of springs, dampers and clutches [68]–[70], this methodology was used to size the mechanical components accurately.

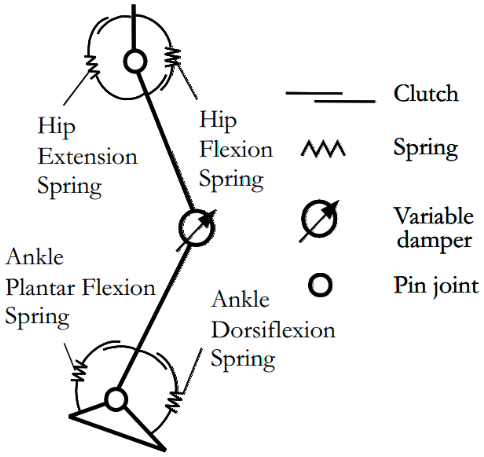


Figure 2-25: MIT quasi-passive leg model **[68]–[70]**

The exoskeleton consists of two legs, allowing three degrees of freedom at the hip, one at the knee, and two at the ankle. Specifically for the ankle joint, the device consists of a heel spring to absorb heel strike energy and a series-elastic actuator at the back of the leg to create dorsiflexion motions. Using underpowered actuators, the device's power consumption is about 2W, 1% of the consumption of most powered exoskeletons. Experimental results suggest that it is feasible to transfer most of the added load through the exoskeleton and reduce the metabolic rate of the user almost to the point of normal walking.

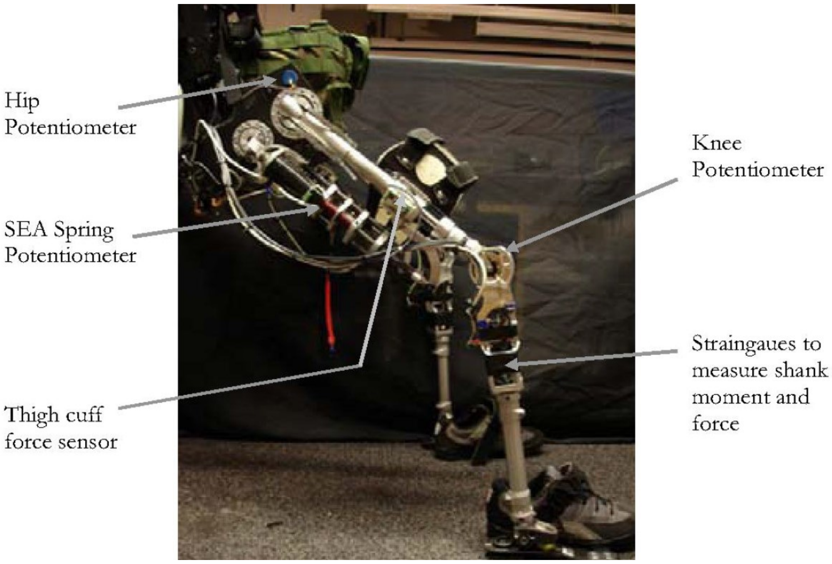


Figure 2-26: MIT quasi-passive exoskeleton **[6], [65]–[67]**

In 2006 and 2007, Banala *et al* [71], [72] designed a gravity-balancing leg orthosis to aid treadmill rehabilitation. The intent of the device was to reduce the felt gravity when walking by keeping the system's potential energy constant, using springs linking the centers of mass of the torso, thigh and lower leg. Experimental results with normal and injured individuals have shown a significant reduction of the hip torques. Although the system does not include the ankle joint, the gravity-balancing concept could be applied to the ankle joint.

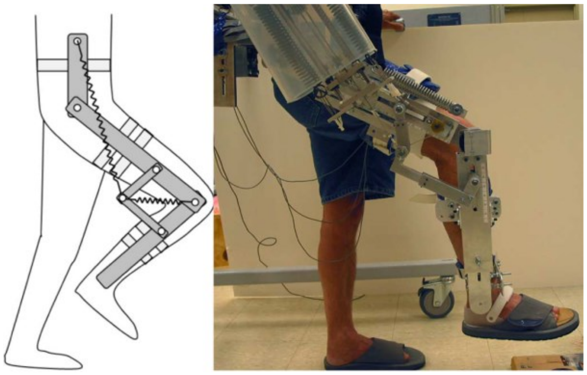


Figure 2-27: Gravity balancing leg orthosis **[71], [72]**

2.2.3.2Active Exoskeletons for Power Augmentation

Most powered exoskeletons allow at least the seven degrees of freedom per leg (3 at hip, 1 at knee and 3 at ankle) and use similar joint conformity. Therefore, the review of selected exoskeletons will focus on the particularities of each exoskeleton, specifically for the ankle joints, if information is available.

The Berkeley Lower Extremity Exoskeleton (BLEEX) [26], [28], [73], [74] is a powered exoskeleton to increase the payload, up to 75 kg, carried by soldiers for extended periods. Initially, the device was hydraulically actuated, but was later converted to electric actuation, making the device lighter. Energetically, BLEEX uses approximately 2.27kW of energy to walk at 1.3m/s, generated onboard via a small 1-piston internal combustion engine.

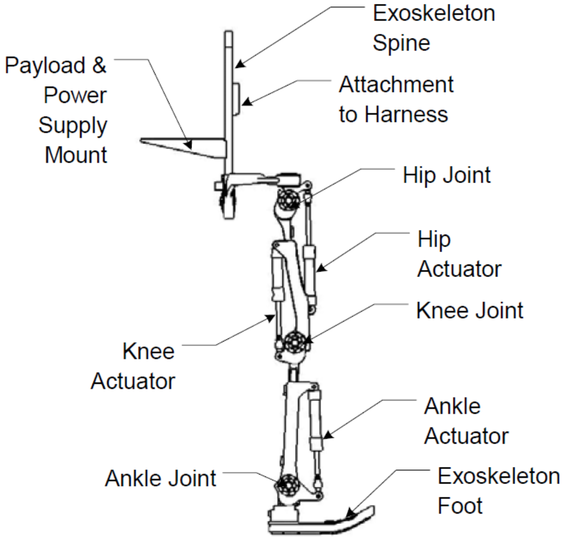


Figure 2-28: BLEEX structure [40]

As shown in Figure 2-28, the ankle actuator is located at the front of the lower leg, a feature unique to the BLEEX. Electronically, the device contains an array of sensors that permit the control system to predict and follow the user's movement, allowing it to kinematically match the leg movements. The ankle joint has two degree of freedom and allow the toe flexion with a flexible toe. The user's foot is placed on a hard plate on which a number of sensors are located.

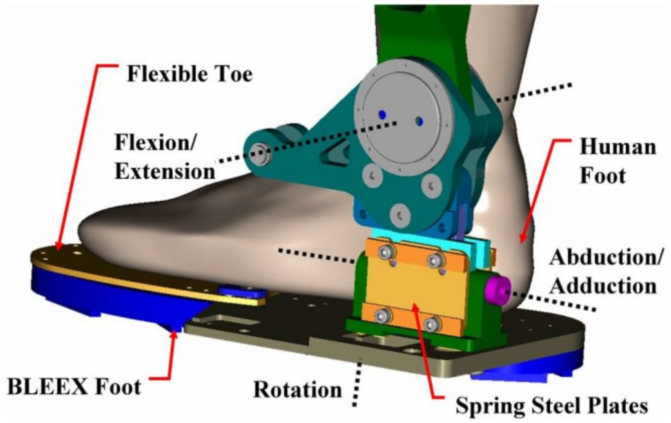


Figure 2-29: BLEEX foot

Emerging from the development of the BLEEX, the company now known as Ekso Bionics [75], developed several other exoskeletons, such as the exoHiker, exoClimber, HULC, the eLEGS, and the EKSO, which are devised for specific applications, such as hiking, climbing, large load carrier, and gait generation.

Following the development of the ASIMO project, Honda developed two exoskeletons, called the Bodyweight Support Assist [76]–[78] and the Stride Management Assist [79], [80]. The first is a seat-based exoskeleton, where the user sits on the device, and the structure is found between the user's legs, as opposed to outside the legs. This electric device actuating the knees intends to reduce fatigue on factory workers by reducing the user's felt weight. The second device provides added force at the hip joint to reduce fatigue over extended walking period. Both systems are qualified as walking assistive device do not provide any ankle power contribution. These devices have shown great promise but they are limited in the range of daily activities they can perform.

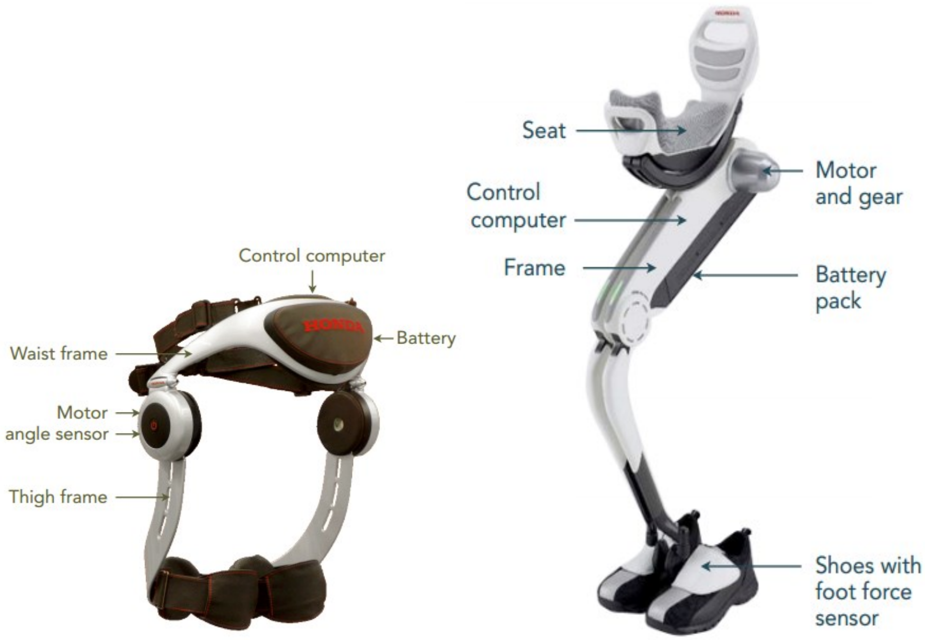


Figure 2-30: Honda stride assist (left) **[79], [80]** and body weight assist (right) **[76]–[78]**

In 1999, Tsagarakis [81] developed an upper body exoskeleton actuated with PAM positioned in an antagonistic configuration, marking one of the first applications of PAMs to exoskeletons. In 2006, Costa *et al* [82], developed a "Soft 10DOF Exoskeleton" following the earlier progress of Tsagarakis. This leg exoskeleton is actuated by pairs of antagonistic PAMs linked via steel cables around the actuated joint. However, like most pneumatically actuated assist devices, the two main problems are local high-pressure gas source and control difficulties.

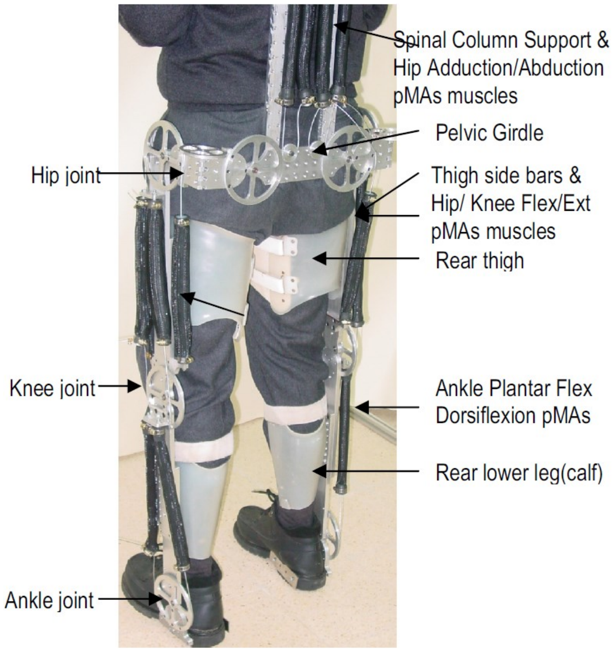


Figure 2-31: Soft 10DOF exoskeleton **[82]**

In 2011, Ikehara *et al* [83], at the Tokyo Metropolitan College of Industrial Technology, developed an electrically actuated exoskeleton, small enough to fit under clothing. This is done by having the power source, control system and actuators in a backpack, connected to the actuated joints via flexible shafts. There is no mention as to the shaft material or the presence of rubbing between the shafts and clothing.

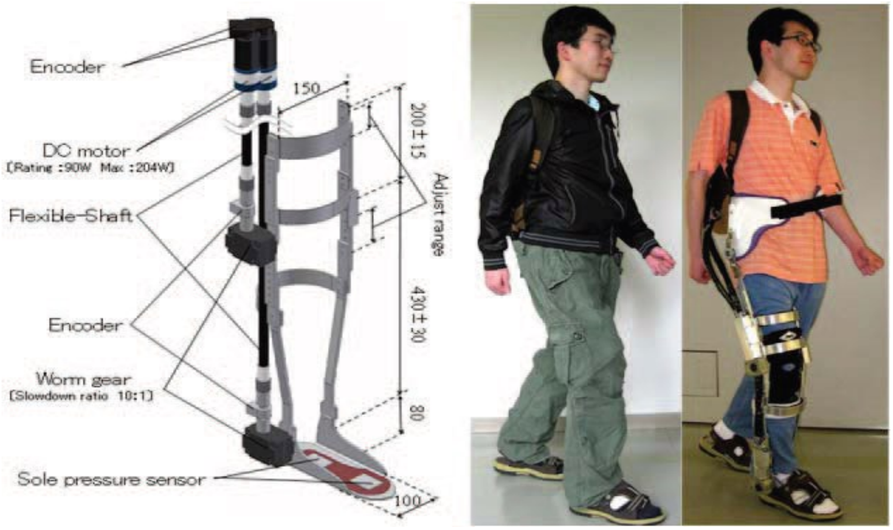


Figure 2-32: Close-fitting exoskeleton **[83]**

2.2.4 Human Locomotion Assist Devices Summary

Circling back to the intent of the current thesis, the goal is to develop an unpowered ankle exoskeleton using a PAM passively. As shown in the devices review, there has been limited development in unpowered exoskeletons in general and in devices using PAMs as actuators. Instead, the walk assist technology development is centered on powered devices using, most notably, series-elastic actuators. Although these might be more versatile, significant technological and financial barriers persist to their widespread adoption.

In the cases where PAMs were used, they were exclusively used as actuators, making the devices dependent on a local high-pressure gas source and complex control algorithms, limiting the devices to short range or tethered uses, such as rehabilitation. Using a PAM as a non-linear spring would permit a much larger user autonomy.

Focusing on the ankle joint, most devices proposed to allow the flexion and extension motion, while limiting the other ankle motions via rigid structures. For devices providing assistance to the ankle flexion and extension, the consensus was to power the ankle at toe-off by applying an upwards force at the back of the foot, in parallel with the calf muscles. Evidence from devices with this configuration suggests it would be the preferable position.

One challenge when using such a configuration is timing the force release to optimize the toe-off power, which has proven to be difficult to solve. For active devices, large amounts of sensors requiring significant processing and real-time control systems have shown varying degrees of success at the cost of larger devices. Passive devices, on the other hand, relied on the devices architecture and/or material for proper timing. For instance, Hirai [56] proposed a pneumatic lock using the user's weight for timing. Still, Carberry [16] suggested the use of a diamond linkage mechanism. However, the most favorable solution was the Smart-Clutch AFO by Wiggin [60], where a system of springs, pins and ratchets recharges a large spring at the back of the shank. This device suggests that simple mechanical systems can be used to control the timing of a PAM used passively.

When considering the overall device structure, most devices prefer to use structures molded to the intended user, usually made of carbon fiber, thermoplastic material or other. Although more expensive, molded structures are usually better as they provide a better fit to the user, transfer loads more easily and generally lighter. In terms of foot connections, the use of insoles or flat plates under the shoe is dependent on the desired outcome, where orthoses generally use insoles and exoskeletons generally use flat plates under the shoe. However, devices allowing a degree of freedom at the toe, whether through compliant structures or hinges, have showed increase walking efficiency, compared to devices that prevent toe flexion.

There are a number of devices currently on the market or in development, each with its own objective and varying degrees of success. Table 2-7, Table 2-8 and Table 2-9 highlight the prosthesis, orthosis and exoskeletons reviewed for the current thesis.

Table 2-7: Prosthesis summary

Device Name	Source	Joints	Power Type	Actuation	Special Feature	Objective
HEKTA	[84]	Knee, Ankle	Passive	Springs	Energy Harvesting	Gait Generation
Recycling Energy Prosthesis	[51]	Ankle	Quasi-Passive	Electric	Series-Elastic Actuator	
AMP-Foot 2.0	[12], [52]	Ankle		Electric	Series-Elastic Actuator	
Flowers Prosthesis	[85]	Knee, Ankle	Active	Hydro-electric	Hydro-Electric Actuators	
MIT Prosthesis	[53]	Ankle, toes		Electric	Series-Elastic Actuator	Gait Generation and Reduce COT
Biologically-inspired BK prosthesis	[13]	Ankle		Pneumatic	Antagonistic muscle (PPAM)	Gait Generation

Table 2-8: Orthosis and single joint exoskeleton summary

Device Name	Source	Joints	Power Type	Actuation	Special Feature	Objective
Pneumatic AFO	[56]	Ankle	Passive	Pneumatic	Ankle Locking mechanism	Foot-drop Prevention
Smart-Clutch AFO	[60]	Ankle		Spring	Smart Clutch	Reduce COT
Xeleton and Dorsiflexion assist	[55]	Knee, Ankle		Material	Typical	Weakened or injured individuals
Natural Axes AFO	[59]	Ankle	Active	Electric	Exact ankle joint line of action	Rehabilitation
Powered AFO	[14], [57], [58]	Ankle		Pneumatic	antagonistic muscle (PAM)	
DCO with Robotic Tendon	[15]	Ankle		Electric	Series-Elastic Actuator	
Active AFO with passive compliance	[16]	Ankle		Pneumatic	Diamond Push-Pull Mechanism	
Joint-Coupled Orthosis	[86]	Hip, Knee, Ankle		FES	Hip-Knee biasing via cord	Spinal Cord Injuries
Stride Management Assist (Honda)	[79], [80]	Hip		Electric	Hip assist	Reduce COT
EXO-PANTOE 1	[17]	Ankle, toes		Electric	Series-Elastic Actuator	Rehabilitation
PPAM Knee-Exoskeleton	[87]	Knee		Pneumatic	PPAM	Feasibility
PAM Knee-Exoskeleton	[88]	Knee		Pneumatic	PAM	Power Augmentation

Table 2-9: Exoskeleton summary

Device Name	Source	Joints	Power Type	Actuation	Special Feature	Objective
GBO	[71], [72]	Hip, Knee	Passive	Springs	Gravity balancing	Rehabilitation
Artificial Tendon Exoskeleton	[89]	Hip, Knee, Ankle		Springs	Artificial Tendon	Reduce COT
Apparatus to aid running	[61]	Hip, Knee, Ankle		Springs	Springs	Power Augmentation
MIT Exoskeleton	[6], [65]–[67]	Hip, Knee, Ankle	Quasi-Passive	Electric	Series-Elastic Actuator	
HARDIMAN	[62]	Entire body	Active	Electric	Giant proportions	
HAL	[90], [91]	Hip, Knee, Ankle		Electric	EMG sensors	
NTU Exoskeleton	[92], [93]	Hip, Knee, Ankle		Electric	Zero Moment Point	
BLEEX	[26], [28], [73], [74]	Hip, Knee, Ankle		Hydraulic, Electric	Shadow Movement	
WPAL	[94], [95]	Hip, Knee		Electric	Low-cost	
Walking Assistance Apparatus	[96], [97]	Knee, Ankle		Electric	Spatial Parallel Links	
XOS	[98]	Entire body		Hydraulic	Military Exoskeleton	
Close-Fitting Exoskeleton	[83]	Knee, Ankle		Electric	Flexible shafts	
EXPOS	[99]	Hip, Knee		Electric	Caster Walker	
Pupin Institute Exoskeleton	[64]	Hip, Knee, Ankle		Electric	N/A	Gait Generation
Seireg Exoskeleton	[63]	Hip, Knee, Ankle		Electric	N/A	
10 DOF Exoskeleton	[82]	Hip, Knee, Ankle		Pneumatic	PAM	
X1-MINA Exoskeleton	[100], [101]	Hip, Knee, Ankle		Electric	Series-Elastic Actuator	
Prime-Walk WPAL	[102], [103]	Hip, Knee, Ankle		Electric	Motors between legs	
Vanderbilt Exoskeleton	[104]	Hip, Knee		Electric	Distributed Electronics	
ReWalk	[105], [106]	Hip, Knee, Ankle		Electric	Commercially available	
LokoMat	[107]–[110]	Hip, Knee		Electric	Treadmill	Rehabilitation
Mechanized Gait Trainer	[111]	Hip, Knee, Ankle		Electric	Step Ladder	
Legs Rehabilitation Exercise System	[112]	Hip, Knee, Ankle		Pneumatic	PAM	
WalkTrainer	[113]	Hip, Knee, Ankle		Electric	Wheels	
PIGRO	[114]	Hip, Knee, Ankle		Pneumatic	Pneumatic Cylinder	
University of Tunku Abdul Rahman Exoskeleton	[115]	Hip, Knee, Ankle		Pneumatic	PAM	
7 DOF Exoskeleton	[81]	Arms		Pneumatic	PAM	Haptic Feedback
Bodyweight Support Assist	[76]–[78]	Knee		Electric	Seat	Reduce COT

2.3 Pneumatic Artificial Muscle

Mobile powered technologies, such as assistive technologies, all face one similar challenge: lack of appropriate actuation system. Specifically for mobility assist devices, actuators need to have high power-to-weight ratios, high reliability, low injury risks, and good energy efficiency. Distinctively different from other forms of actuators, PAMs present many of the properties that may be a key to the challenge of actuation for mobility assist devices. PAMs consist of a mostly cylindrical elastic bladder pressurized with a fluid, usually air, and present a non-linear force-length behaviour [116]. This behaviour is especially interesting for human mobility assistance technologies given its similarities to the human muscles [117].

Whereas PAMs can be used actively or passively, they are mostly used in active applications, most notably the Shadow Robot Company Dextrous Hand [118]. Consequently, static and dynamic force models have been developed describing the force and contraction distance behaviour fairly accurately. The main limitations of using PAMs as actuators in walking assistive technologies are the need for a high-pressure gas source, power source and associated accessories. This limits the use of PAMs as actuators to tethered applications where movements are limited, such as rehabilitation.

Using PAMs as passive components, instead of actuators, removes the need for a local high-pressure gas source, resulting in a wider range of motion assist applications, other than rehabilitation. However, the implementation of PAMs in unpowered assistive technologies requires customization to meet the mechanical behaviour requirements, and thus modeling of passive PAM behaviour is imperative. Unfortunately, current passive models have been limited with results varying from acceptable to poor in terms of accuracy.

2.3.1 Physical Characteristics

With regards to its construction, the PAM is made from an elastic internal bladder surrounded by a braided mesh and two fixtures, one at either end. Although the system can be filled with any fluid, in most cases PAMs are filled with air. The elastic internal bladder can be made of different elastic material, such as butyl rubber, silicone tubing or latex. The braided mesh fibres are made from polyethylene terephthalate (PET), a commercial product most commonly used in electrical systems [119]. The end fixtures allow the internal bladder to create a seal and provide an attachment for the braided mesh. These custom made fixtures are made from stainless steel [120], as shown in Figure 2-33.

With reference to Figure 2-33, θ is the fibre angle, β is the mesh inclination angle at muscle ends, ΔL is the muscle contraction distance, F is the longitudinal muscle force, and L_0 is the deflated, or un-pressurized, length. Typically, the PAM total weight ranges between 0.1 and 0.2 kg [1].

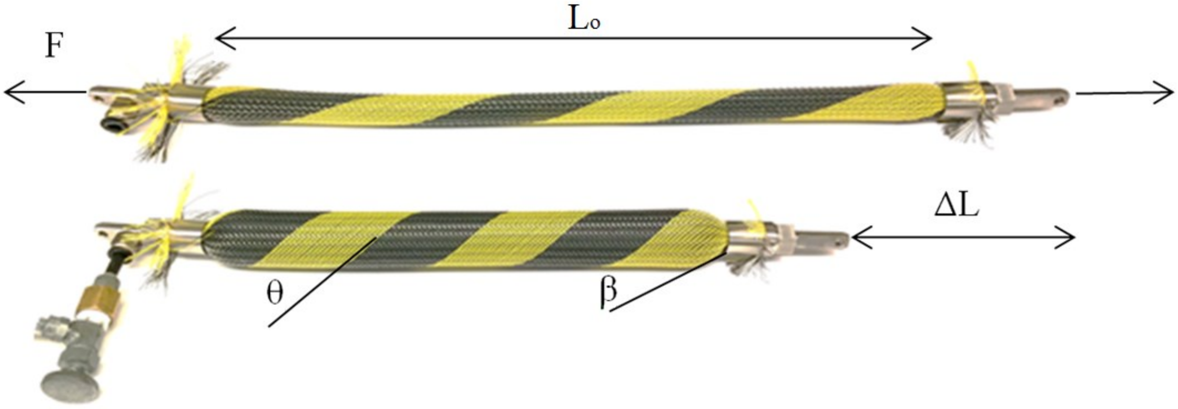


Figure 2-33: Deflated (top) and inflated (bottom) PAM with stripped braided mesh

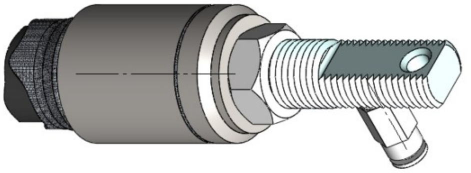


Figure 2-34: Modeled PAM end caps

When pressurized, the internal bladder diameter increases, causing an increase in the fibre angle, forcing an overall muscle length reduction. The mechanical force behaviour is non-linear and presents a hysteresis through a contraction-relaxation cycle. Keeping the PAM's length constant over a pressurization cycle (isometric contraction) will produce a substantial longitudinal contraction force. For example, it was found that a PAM with a 1.9 cm diameter, pressurized at 200 kPa can produce a force of 1200 N. Since the braided mesh is axisymmetric, the pressurization does not cause any muscle torsion.

2.3.2 PAM Geometry

The PAM model development is based on the PAM geometry, which is presented in Figure 2-35 and Figure 2-36. The PAM overall length L is split in three sections: a long cylindrical section of length L_m, and 2 end length L_{end}. The ends are considered as cone frustums [21], [121] with diameters D and d, and a cone angle β. The larger diameter also corresponds to the diameter of the cylinder formed in the middle of the muscle. Figure 3-2 shows a single fiber of the braided mesh within the cylindrical section and the relationship that exists between the fibre length L_y, the fibre angle θ, the muscle diameter and the PAM cylindrical length.

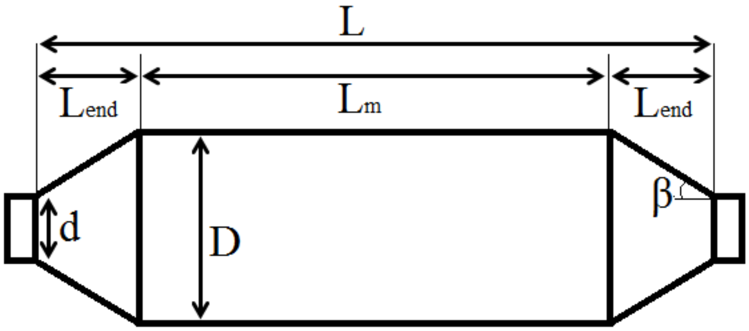


Figure 2-35: PAM geometry

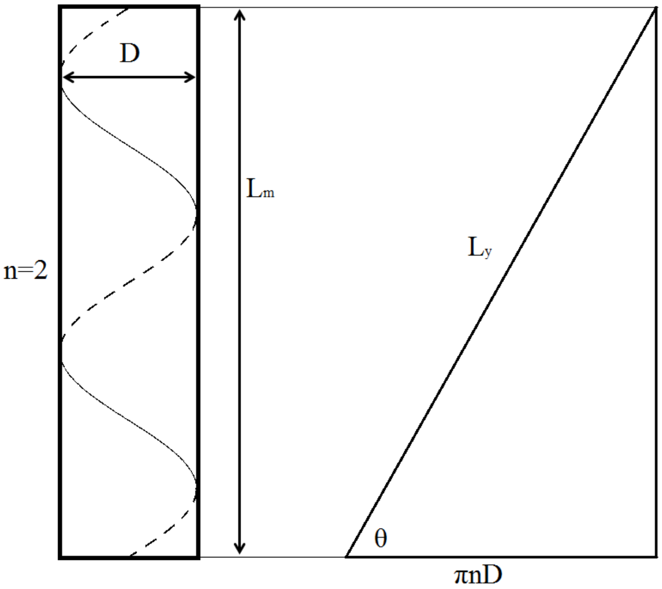


Figure 2-36: PAM braided mesh geometry for single fibre

Consequently, the PAM geometric relationships for the cylindrical section can be obtained.

$$L_m = L_y \, cos\, \theta \qquad \textbf{(2-4)}$$

$$D = \frac{L_y \, sin\, \theta}{\pi n} \qquad \textbf{(2-5)}$$

$$\tan\theta = \frac{L_m}{\pi n D} \qquad \textbf{(2-6)}$$

$$L_m = \sqrt{L_y^2 - (\pi n D)^2} \quad \textbf{(2-7)}$$

It has been proposed to model the end caps behaviour by cone frustums, but when considering PAMs for passive applications, they are usually pre-inflated and rather long. In such cases, the length of the cone frustums is much smaller than the length of the cylindrical section. Consequently, most of the length variation occurs in the cylindrical sections. Therefore, the end lengths are neglected, transforming (2-4) and (2-7) to:

$$L = L_y \, cos \, \theta \quad \textbf{(2-8)}$$

$$L = \sqrt{L_y^2 - (\pi n D)^2} \quad \textbf{(2-9)}$$

The PAM volume V is another important variable to consider when developing the stiffness model. Considering the muscle as a cylinder, its volume can be expressed in terms of muscle length.

$$V = \frac{L \cdot L_y^2 - L^3}{4\pi n^2} \quad \textbf{(2-10)}$$

2.3.3 Force Model Developments

Using different approaches, many static and dynamic force models have been developed to model the PAM non-linear behaviour, with or without the hysteresis effect. The PAM was first introduced in 1941 in a patent by Johnson and Piece, for the mining industry as a tool to dislodge coal [122]. The patent describes the PAM as an "expansible cover"; an improvement to mining cartridges, which are inflatable bags without a surrounding braided mesh. The PAM was patented a second time in 1949 by De Haven for use in a self-contained seat belt tensioning device, activated with gun powder [123]. These two patents present early PAM developments and describe its behaviour, without presenting any mathematical model.

The first published PAM force-displacement model was by Gaylord in 1955 [124]. However, no theoretical development was provided with the model.

$$F = \frac{P\pi D_{90^o}^2}{4}(3cos^2\theta - 1) \quad \textbf{(2-11)}$$

where F is the force, P is the internal pressure, D_{90^o} is the muscle diameter and θ is the braid angle. In 1957, McKibben developed a Wrist-Driven Wrist Hand Orthosis using a PAM, which is where PAMs gained their alternative name: the McKibben muscle. In 1961, Schulte proposed a fully analytical model incorporating forces, internal pressure, and geometry, based on Newtonian mechanics [125]. After providing a series of simplifications, he obtained the model proposed by Gaylord in (2-11).

Although presenting interesting properties, PAMs were soon abandoned due to their control complexity, their need for large supply tanks and portable high-pressure gas source. The next developments came with the commercialization of PAMs by Bridgestone Co. in the 1980's [126], and its continued commercialization by Festo [127]. The commercialization of the PAM was associated with a renewed interest in this technology, when it was found that its force-length behaviour resembles that of biological muscles.

Various methods were used to model the PAM behaviour, such as Newtonian Mechanics [121], Thermodynamics [128], Virtual Work [129]–[132] and Continuum Mechanics [133]–[136]. The current thesis omits the continuum mechanics approach, as it differs considerably in terms of underlying assumptions and methodology. The most common variation of the force model, shown in (2-12), represents the force as a function of length, as opposed to braid angle like (2-11), due to the simplicity in measuring the muscle length. For simplification purposes, this model is termed the traditional force model herein, where *F* is the force, *P* is the gauge pressure, *L* is the muscle length, *b* is the fibre length, and *n* is the fibre rotations.

$$F = \frac{P(3L^2 - b^2)}{4\pi n^2} \tag{2-12}$$

In 1994 and 1996, Chou and Hannaford [129], [130] developed a static PAM force model using the virtual work approach, obtaining the traditional force model. They also included a term to account for the PAM wall thickness t_k which yielded the following:

$$F = \frac{P\pi D_0^2}{4}(3cos^2\theta - 1) + \pi P\left[D_0 t_k\left(2\sin\theta - \frac{1}{\sin\theta}\right) - {t_k}^2\right] \tag{2-13}$$

This model was validated by testing PAMs in a static and dynamic environment, and presented fairly accurate results. Meanwhile, they suggested that the main friction present is a Coulomb friction but did not quantify it.

In 1997 and 2000, Tondu *et al*. [131], [132] proposed a static force model identical to the traditional force model, using virtual work. They then modified it to include the end effects, with a contraction parameter *k*:

$$F = P\pi r_0^2(a(1 - k\varepsilon)^2 - b) \tag{2-14}$$

where a and b are geometric parameters, r_o is the muscle radius, and ε is an experimental contraction ratio. This model was validated for static and dynamic conditions, with acceptable results. They also proposed a friction model that accounts for the friction between the fibers.

$$F_f = \frac{2\pi r_0 l_0 \sin\theta_0}{1-\varepsilon\sqrt{1-(\cos\theta_0)^2(1-\varepsilon)^2}}\frac{\mu}{n}P \tag{2-15}$$

In 2000, Tsagarakis and Caldwell [121] developed the traditional force model, and incorporated the thickness effect proposed by Chou and Hannaford, as shown in (2-13). They further expanded this model by incorporating end-effects, and radial expansion effects, as shown in (2-16) and (2-17), respectfully.

$$F_{end} = \frac{Pb\pi D_0^2(\cos\theta)^2\sin\theta + \frac{2}{3}L_1\pi D_0^2[(\cos\theta)^2-(\cos\theta_{min})^2]}{2b\sin\theta + 4b[\cos\theta - \cos\theta_{min}]} \tag{2-16}$$

$$F_{radial} = k\big(D_0\sin\theta - D_{flat}\big) \tag{2-17}$$

where L_1 is the length of the cone frustums and k is an experimentally determined factor that relates the change in diameter to the muscle pressure. After comparing experimental results with their proposed model and the Chou and Hannaford model, their model appeared to have better results than the Chou and Hannaford model.

In 2006, Davis and Caldwell [137] furthered the model development done by Chou and Hannaford [130] and by Tondu et al. [132] by incorporating the Hertz Contact Stress Theory to determine the contact area between the fibers in the friction model calculations. This proved to reduce the error of the previous model to within a range of 10%.

In 2009, Doumit *et al.* [21] developed a force model that takes into account the deformation of the bladder close to the end fixtures, modeled as frustum cones. The force model is based on thin pressure vessel theory combined with tension forces in the braided mesh fibres, and considers friction forces, internal bladder thickness and end effects.

$$F = NT_f\, cos\,\theta_e\, cos\,\beta - f_{fs} \tag{2-18}$$

where N is the number of fibres, T_f is the tension in the fibres, θ_e is the muscle-end braid angle, β is the frustum cone angle and f_{fs} is the friction force. By neglecting friction, material properties and end effects, it was shown that the model did contain the traditional force model, as shown in (2-19). The derivation used by Doumit *et al.* [21] is shown in 0, for reference.

$$F_m = -P\left[\frac{L_y^2 - 3L^2}{4\pi n^2}\right] \tag{2-19}$$

Vo-Minh *et al.* [138] and Godage *et al.* [139] have proposed adding numerical models, such as the Maxwell-Slip Model or the Bouc-Wen Model, to the usual force-length relationship to account for the PAM hysteresis. These proposed models have been shown to be accurate but rely on experimentally determined parameters.

In summary, the traditional force model is, to this day, the basis for all other force models applied to the PAM, either for static or dynamic applications. The model was modified by incorporating terms accounting for end-effects, fiber friction, radial expansion effects, and wall thickness effects, which usually translated into more accurate results. However, the present objective is to obtain a stiffness model for the PAM. Force models were initially developed, as opposed to stiffness model, due to the intended applications of PAMs as actuators. In comparison, the use of PAMs as passive components is rather recent.

2.3.4 Stiffness Model Developments

Stiffness is defined as the infinitesimal force variation with regards to length. For PAMs, one of two approaches is usually taken. The first is to apply the stiffness definition to a previously developed force model. The second is to assume a relationship based on empirical evidence and validate it with experimental results. These two approaches were developed by considering the PAM as a gas spring [140].

$$K \equiv \frac{dF}{dL} \tag{2-20}$$

In 1994 and 1996, Chou and Hannaford [129], [130] were the first to propose to model the PAM as a gas spring. They proposed using two spring constants, where one uses the stiffness definition K_g, and the other relates to the stiffness per unit pressure K_p.

$$F = K_g(P^I - P_{th})(L - L_{min}) + K_p(L - L_0) + nl(L) \tag{2-21}$$

where F is the force, P^I is the internal pressure, P_{th} is the pressure threshold to overcome the radial elasticity of the internal bladder, L is the total length of the actuator at desired pressure, L_0 is the resting length, L_{min} is the minimum length of the actuator and $nl(L)$ is an adjustment factor. The PAM model constants were determined with experimental analyses, by varying the muscle length at various constant pressures. Although this model considers internal bladder properties, the use of experimental constants limits its generalization.

In 1999, Tsagarakis *et al* [81] proposed a PAM force model based on surface pressures, incorporating a variable stiffness model based on pure springs with constant stiffness', in an antagonistic PAM pair that causes rotation:

$$F = K(a \pm r\theta) \tag{2-22}$$

$$K = K_p P + K_e \tag{2-23}$$

where F is the force, a is the dilatation distance from the original length to the full contraction length, r is the radius of the pulley used, θ is the rotational angle, K is the overall spring constant, K_p is the stiffness proportional to the internal pressure and K_e is the spring stiffness. Although novel in approach, these authors did not provide a method of obtaining the stiffness' nor did they provide theoretical or experimental values.

In 2000, Colbrunn *et al.* [141] proposed a stiffness model based on applying the stiffness definition to the traditional force model. As such, the obtained stiffness model was:

$$k = \frac{b^2}{4\pi n^2}\left(\frac{3L^2}{b^2} - 1\right)\frac{dP_g}{dL} + \frac{3P_g L}{2\pi n^2} \tag{2-24}$$

where P_g corresponds to the internal gauge pressure, n corresponds to the number of fibre revolutions per braid length L, and b is the length of the uncoiled fibre. The derivative of P_g with respect to the muscle length was considered negligible, assuming that the actuator external volume is equal or larger than the internal actuator volume, making the internal pressure approximately constant through the contraction cycle. This assumption simplifies the stiffness model, resulting in a linear relationship with the length.

$$P_g \approx constant \rightarrow \frac{dP_g}{dL} \approx 0 \tag{2-25}$$

$$k = \frac{3P_g L}{2\pi n^2} = \frac{6F}{(3L - \frac{b^2}{L})} \tag{2-26}$$

Since this is the only stiffness model without experimental factors, it is used in this thesis as the current standard.

In 2003 and 2006, Davis *et al* [137], [142] proposed a model for the stiffness of the fibers in the braided mesh, obtained by combining the traditional force model with an analytical elastic modulus:

$$K = \frac{4\pi n^2 EA}{PLb_{min}} \tag{2-27}$$

$$E = \frac{FA(b - b_{min})}{b_{min}\, cos\, \theta} \tag{2-28}$$

where b_{min} is the minimum fibre length, b is the fibre length, A is the fibre cross-sectional area, F is the applied force and θ is the fibre angle. Although the results for the proposed force model were good, accounted for fibre stresses, frictional force and the hysteresis, no overall PAM stiffness model was proposed.

While stiffness models have been developed for pneumatic drives [143]–[145], they are not applicable to PAMs because they are fundamentally different from pneumatic drive in terms of mechanical properties and geometry. Currently, most stiffness models developed consider PAMs as perfect cylinders and incorporate frictional forces and material properties through experimentally determined coefficients. Furthermore, they are often a second thought as most PAMs are currently used as actuators, with the drawback of needing a local high-pressure gas source. The development of a theoretical PAM stiffness model, incorporating mechanical and geometrical properties, as well as friction, would be beneficial towards the design and application of PAMs to unpowered devices, specifically to human mobility devices.

Chapter 3– PNEUMATIC ARTIFICIAL MUSCLE PASSIVE MODELS

The PAM presents a higher power-to-weight ratio compared to other mechanical actuators and a non-linear force-length behaviour, making it very attractive for human mobility devices. Its main drawback as an actuator is the need for a local high-pressure gas source, which is cumbersome for mobile applications. The use of a PAM as a passive component not only avoids this need for a local high-pressure gas source, but also mimics the behaviour of the Achilles tendon, which is passive and non-linear by nature. Thus, the PAM would be pre-inflated and then used as an air spring, without the need for a portable high-pressure gas source.

However, PAM stiffness models in open literature are sparse and do not predict the PAM stiffness well. An accurate stiffness model that accounts for various muscle characteristics is crucial as a design and selection tool to assist in the development of an ankle exoskeleton. This chapter presents the development and validation of a PAM stiffness model that accounts for geometric interactions and friction effects.

The stiffness is defined as the variation of force with regards to a variation in length. The traditional force model accounts for the motions in the cylindrical part of the muscle only and was obtained by various researchers using a variety of methods. Although the traditional force model ignores many parameters due to its underlying assumptions, its linear structure permits the linear inclusion of terms that account for additional PAM parameters, as demonstrated by [129], [130], [135]. The development of the traditional force model, using Doumit's development [21], is included in 0 as a reference.

3.1 Passive Model Development

The stiffness k is defined as the variation of force *dF* with regards to a variation in length *dL*.

$$k \equiv \frac{dF}{dL} \tag{3-1}$$

A generic PAM stiffness model can be obtained by applying the stiffness definition (3-1) to the traditional force model (3-2).

$$F_m = -P\left[\frac{L_y^2 - 3L^2}{4\pi n^2}\right] \tag{3-2}$$

$$k = \frac{dF_m}{dL} = -\frac{dP}{dL}\left[\frac{L_y^2 - 3L^2}{4\pi n^2}\right] + \frac{3PL}{2\pi n^2} \tag{3-3}$$

where *P* is the internal gauge pressure, *D* is the external diameter, *n* is the number of fiber revolution, *L* is the muscle length, and L_y is the fibre length.

As shown in the (3-3), the PAM stiffness is defined as a parabolic function of length. Colbrunn *et al* [141] obtained this result but assumed the pressure remained constant in the muscle, making the pressure derivative term dP/dL equal to zero. This assumption is reasonable for a situation when a dead volume, in the form of a large supply tank, intermediary tank or long tubing, significantly larger than the internal PAM volume is present. However, the intent of the current analysis is to develop a stiffness model for a PAM used as an air spring, without any or with minimal dead volume. Consequently, the pressure derivative term cannot be neglected and an analytical expression for the pressure derivative must be derived. Considering that the fluid used is air at a temperature between -10ºC and 40ºC, air can be assumed to behave as a perfect gas. The PAM air mass *M* can then be expressed as:

$$M = \frac{PV}{RT} \tag{3-4}$$

where R is the gas constant for air, V is the muscle volume, and T is the internal temperature. Deriving (3-4), expanding it using binomial theorem and ignoring higher order terms produces a linearized perfect gas equation, where the upper case symbols represent steady-state values and the lower case symbols represent their variations.

$$\frac{m}{M_{ss}} = \frac{p}{P_{ss}} + \frac{v}{V_{ss}} - \frac{t}{T_{ss}} \tag{3-5}$$

Assuming a polytropic relationship as shown in (3-6), where n_g is the polytropic constant varying from 1 for an isothermal process and 1.4 for an adiabatic process, the temperature can be replaced to obtain (3-7). This type of assumption is based on empirical results and not on the thermodynamic process that occurs over a contraction cycle.

$$PV^{n_g} = constant \tag{3-6}$$

$$\frac{p}{P_{ss}} = n_g \frac{m}{M_{ss}} - n_g \frac{v}{V_{ss}} \tag{3-7}$$

As stated previously, the use of a PAM as an air spring assumes that there is no dead volume. Consequently, the internal mass is constant, reducing the linearized perfect gas equation to:

$$p = -P_{ss} n_g \frac{v}{V_{ss}} \tag{3-8}$$

Rewriting the variations p and v as infinitesimal variations dP and dV, and substituting V with (2-10), a relationship describing the pressure derivative with regards to the muscle length is obtained:

$$\frac{dP}{dL} = -\frac{P_{ss}}{V_{ss}} n_g \frac{dV}{dL} \tag{3-9}$$

$$\frac{dP}{dL} = -P_{ss} n_g \left(\frac{{L_y}^2 - 3L^2}{L_{ss}{L_y}^2 - {L_{ss}}^3} \right) \tag{3-10}$$

where L_{ss} is the steady state length. When developing the pressure derivative, one of the assumptions stated that the internal pressure is described as the steady state pressure plus the variation from steady state ($P = P_{ss} + dP$). Hence, the internal pressure can be expressed in terms of steady state pressure by combining relationship for PAM volume (2-10) and the polytropic relationship (3-6):

$$P = P_{ss}\left(\frac{L_{ss}{L_y}^2 - {L_{ss}}^3}{L{L_y}^2 - L^3}\right)^{n_g} \tag{3-11}$$

Finally, the complete expression for the stiffness based on the traditional model is defined.

$$k_g = -\frac{P_{ss}n_g}{4\pi n^2}\left[\frac{\left({L_y}^2 - 3L^2\right)^2}{L_{ss}{L_y}^2 - {L_{ss}}^3}\right] + \frac{3LP_{ss}}{2\pi n^2}\left(\frac{L_{ss}{L_y}^2 - {L_{ss}}^3}{L{L_y}^2 - L^3}\right)^{n_g} \tag{3-12}$$

The PAM stiffness model is developed based on the traditional force model and provided an explicit solution for the derivative of pressure over length, which was neglected in previous studies. Thus, this model accounts for muscle geometry, idealized internal pressure and muscle force, while ignoring end effects, friction, and wall thickness.

3.2 Friction Effects

Experimental results indicated that the PAM presents a hysteresis over an eccentric contraction cycle, as shown in Figure 3-1.

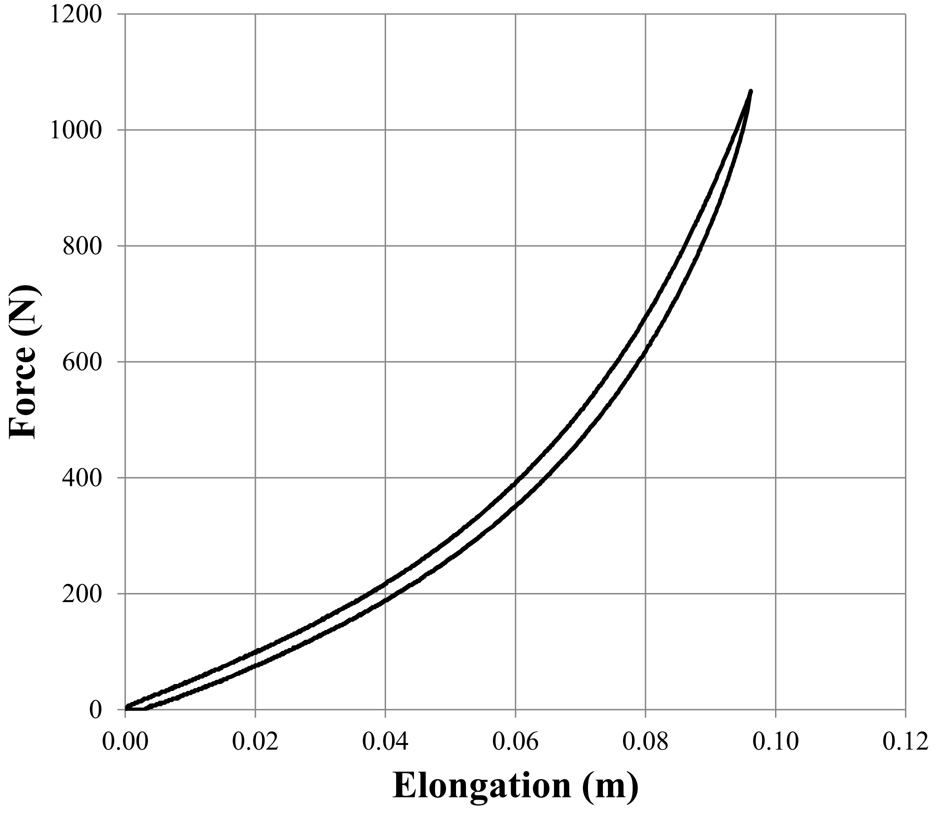


Figure 3-1: Eccentric contraction cycle of a PAM inflated at 311 kPa, with a rubber internal bladder and a FlexoPet braided mesh

In general, a hysteresis is associated with energy losses. Through experimental observations of PAMs, three main energy loss mechanisms became apparent: leaks, plastic deformations and friction. Limited energy losses with regards to the PAM have been explored in open literature. The purpose of this section is to model the PAM friction losses following friction force developed by Doumit *et al.* [21], and find the related analytical stiffness variations.

By definition, friction is the resistance that is encountered when two solid surfaces slide or tend to slide over each other [23]. For the PAM, two friction mechanisms are considered: fibre on fibre friction, and fibre on bladder friction.

Customarily in classic mechanics, friction force F_f is quantified in terms of static or dynamic friction coefficient μ_s or μ_k and a force perpendicular to the surface F_n, as shown in (3-13). For the PAM, the only perpendicular force acting on the surface is the result of the internal pressure. Thus, the friction force is defined in terms of internal pressure P and friction area A_f, as shown in (3-14).

$$F_f = F_n\mu_x \tag{3-13}$$

$$F_f = PA_f\mu_x \tag{3-14}$$

As stated previously, the internal pressure of the bladder is theoretically approximated from the polytropic relationship (3-11). The main difficulty in calculating the friction force becomes identifying and developing a relationship for the friction area.

3.2.1 Fibre-Fibre Friction

The braided mesh is made of two interweaved fiber families, where a fiber family consists of all parallel fibres. Considering the fibers to be inextensible over the contraction cycle range of forces, they tend to rotate and change braid angle to accommodate for internal bladder inflation, creating friction between the overlapping fibers. A corollary of the inextensibility of the fibers is that the fibers are un-deformable, thereby removing the possibility of using the Hertz Contact Stress Theory to quantify the deformation of the fiber contact surface.

Figure 3-7 indicates the resulting parallelogram of side z created by the crossing of two fibers of diameter d_f. The side length can be described either in terms of fibre width and fibre angle θ or fibre width d_f and muscle length L:

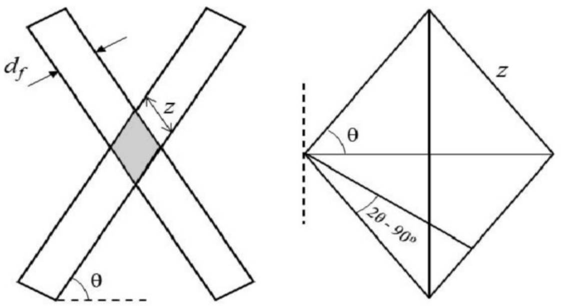


Figure 3-2: Overlapping fibre parallelogram **[21]**

$$z = \frac{d_f}{cos(2\theta - 90)} = \frac{{L_y}^2 d_f}{2L\sqrt{{L_y}^2 - L^2}} \tag{3-15}$$

The total friction area for the fibre-fibre friction consists of the sum of all the parallelograms formed by the overlapping fibres in the braided mesh. Figure 3-3 indicates that each fiber crosses twice the fibers from the other family during each revolution.

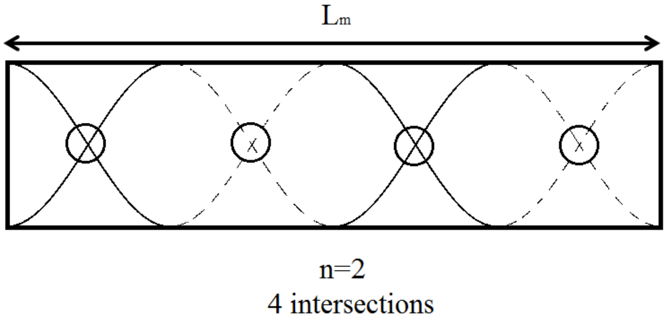


Figure 3-3: Pair of intersecting fibres in the PAM over two revolutions

In addition, there is a third parallelogram formed between fibres that start in the same location. Therefore, the number of parallelograms *qff* formed by the overlapping fibres can be defined as:

$$q_{ff} = \frac{N}{2}(nN + 1) \tag{3-16}$$

The overall friction area *Aff*, fibre-fibre friction *Fff* and resulting stiffness *kff* can be obtained:

$$A_{ff} = zd_f q_{ff} = \left(\frac{{L_y}^2 {d_f}^2}{2L\sqrt{{L_y}^2 - L^2}}\right)\left(\frac{N}{2}(nN + 1)\right) \tag{3-17}$$

$$F_{ff} = P\mu_{ff}\left(\frac{{L_y}^2{d_f}^2}{2L\sqrt{{L_y}^2 - L^2}}\right)\left(\frac{N}{2}(nN+1)\right) \tag{3-18}$$

$$k_{ff} = \mu_{ff}\frac{{d_f}^2{L_y}^2}{\sqrt{{L_y}^2 - L^2}}\frac{N}{4}(nN+1)\left[\frac{1}{L}\frac{dP}{dL} - \frac{P}{L^2} + \frac{P}{{L_y}^2 - L^2}\right] \tag{3-19}$$

where μ_{ff} is the fiber on fiber friction coefficient. Equation (3-19) requires technical information on the braided mesh, current muscle length and pressure. The properties of the braided mesh used for experimental analysis are tabulated in Appendix C. Relationships for the internal pressure and its derivative have been developed previously in terms of steady state parameters in (3-10) and (3-11), respectively.

3.2.2 Fibre-Bladder Friction

The fibre-bladder friction represents the friction between the bladder and the innermost layer of fibres in the braided mesh. This friction mechanism has never been explored because previous researchers assumed that relative motion between the bladder and the fibres is non-existent. However, motion is not a sufficient requirement to justify the absence of friction. Therefore, it is herein proposed that friction between the bladder and the innermost fibres is present, and that it can be considered static due to limited relative motion between the two surfaces.

Each fibre is in contact with the bladder along its whole length, minus every second intersecting parallelogram in a single family, assuming the fibres are weaved in an "over-under" pattern. Therefore, the number of parallelograms q_{fb} corresponds to half the parallelograms of the fibre-fibre friction.

$$q_{fb} = \frac{q_{ff}}{2} = \frac{N}{4}(nN+1) \tag{3-20}$$

The overall friction area A_{fb}, fibre-fibre friction F_{fb} and resulting stiffness k_{fb} can be obtained:

$$A_{fb} = d_f(NL_y - zq_{fb}) = d_fL_y\left(N - \frac{L_yd_f}{2L\sqrt{{L_y}^2 - L^2}}\frac{N}{4}(nN+1)\right) \tag{3-21}$$

$$F_{fb} = P\mu_{fb}d_fL_y\left(N - \frac{L_yd_f}{2L\sqrt{{L_y}^2 - L^2}}\frac{N}{4}(nN+1)\right) \tag{3-22}$$

$$k_{fb} = \frac{dP}{dL}\mu_{fb}d_fL_yN - \frac{\mu_{fb}{d_f}^2{L_y}^2}{\sqrt{{L_y}^2 - L^2}}\frac{N}{8}(nN+1)\left[\frac{1}{L}\frac{dP}{dL} - \frac{P}{L^2} + \frac{P}{{L_y}^2 - L^2}\right] \tag{3-23}$$

where μ_{fb} is the fiber on fiber friction coefficient. Noticing that the pressure and its length derivative are present, these can be replaced by (3-10) and (3-11) respectively, to obtain the overall stiffness in terms of steady state properties.

3.3 Overall Stiffness Model

To summarize, the stiffness due to friction takes into account two friction mechanisms: friction between fibres in the braided mesh, and friction between fibres and the internal bladder. Since the traditional force model, used as a basis for the development of the friction components, has been shown to be a linear model in terms of PAM characteristics, the overall stiffness model k_f combines the stiffness based on the traditional force model and both stiffnesses based on the friction relationships.

$$k_f = \begin{cases} k_g + k_{ff} + k_{fb}, & Lengthening \\ k_g - k_{ff} - k_{fb}, & Shortening \end{cases} \tag{3-24}$$

To properly visualize how the friction models are added to the geometric model, consider a pre-inflated PAM undergoing cyclic loading. Initially, the muscle is being lengthened, causing the force calculated using the traditional force model and both friction forces to oppose the external force. When the muscle is shortened, the traditional force still opposes the external force but the friction forces have changed direction.

3.3.1 Review of Friction Models

Since the developed friction models rely on accurate depiction of the area, it may be useful to quantify the surface areas involved during an experimental cycle. Figure 3-4 compares the surface areas involved for the two proposed friction models.

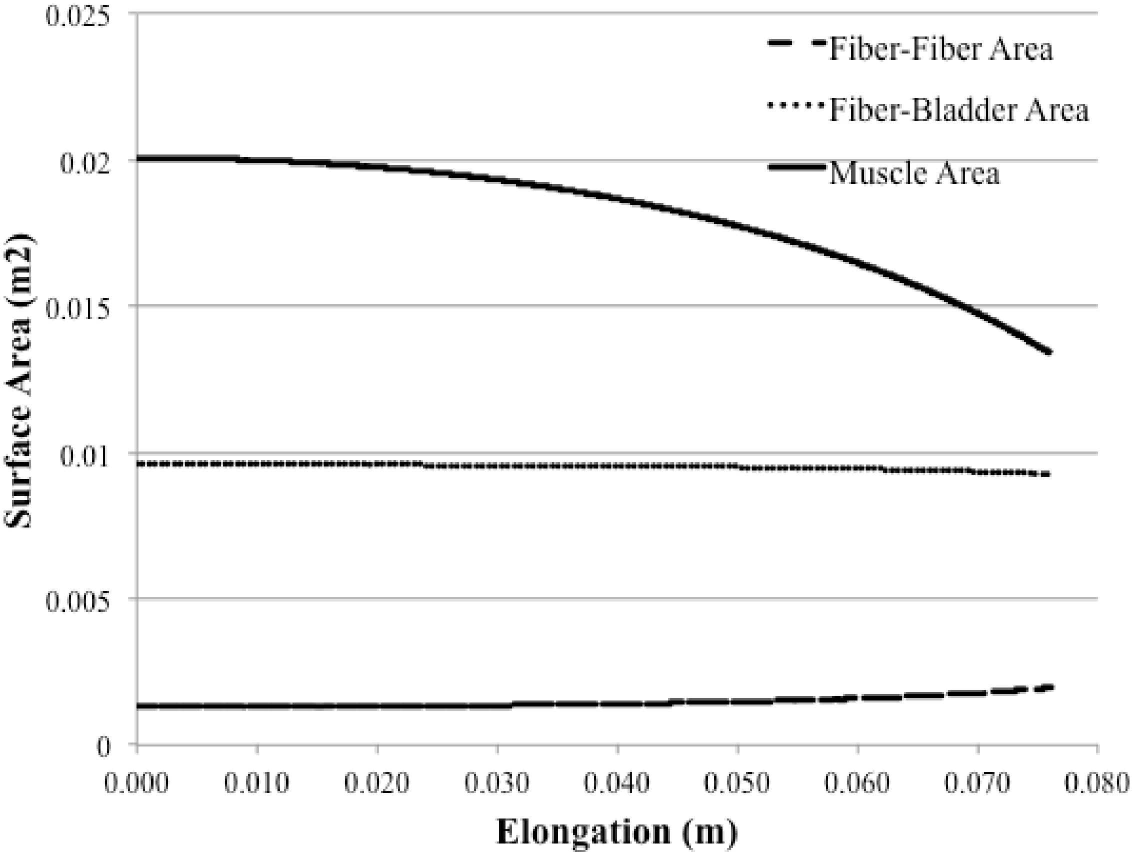


Figure 3-4: Surface Area over an Eccentric Contraction for PAM Prototype n2.82_D13_L316_P276_MNB_BBR_V1.00

As can be seen, the area for the friction between the fibers and the bladder (upper curve) is much larger than the area for the fiber friction (lower curve) and the area for the muscle (middle curve), suggesting that a large static friction would be created.

However, the experimental results, shown later in section 3.4.2, indicate a relatively small hysteresis, suggesting that the overall friction is small compared to the actual muscle force. Furthermore, going back to the friction definition, friction requires the interplay of two forces: one to push the surfaces against each other, and a second to move the objects. For the PAM, the internal muscle pressure causes the bladder to push on the fibers, and the longitudinal force rotates the fibers against themselves, but does not act between the fiber and the bladder. Therefore, it is suggested that the friction between the bladder and the fibers k_{fb} is negligible and may by ignored from further analysis. Thus, the stiffness model is simplified and becomes:

$$k_f = \begin{cases} k_g + k_{ff}, & Lengthening \\ k_g - k_{ff}, & Shortening \end{cases} \tag{3-25}$$

In conclusion, this model accounts for the PAM internal pressure, muscle dimensions, fiber dimensions and friction between fibers but excludes end effects, bladder properties, braided mesh properties and thermodynamic effects.

3.4 PAM Stiffness Validation

To validate the proposed stiffness model, a series of experiments were performed, where PAM prototypes were subjected to series of eccentric contractions. The parameters extracted from the experiments were the contraction force, the muscle elongation and internal pressure. The present section details the methodology used and presents results comparing the experimental results to the developed stiffness model as well as the current standard stiffness model.

3.4.1 Method and Set-up

3.4.1.1Methodology

An experimental methodology was developed to acquire the experimental force, experimental elongation and experimental pressure of various PAM prototype undergoing eccentric contractions. The experimental methodology consisted of the following steps:

1. Measure the PAM geometric parameters at atmospheric pressure (length, diameter, braid cycle, braid angle)
2. Inflate the PAM to desired pressure
3. Close the valve connecting the high-pressure gas source to the PAM, thus closing off the pressurized PAM from the external environment.
4. Verify if air leaks are present. If so, restart with new muscle
5. Measure the pressurized PAM geometric parameters.
6. Place PAM in the tensile testing machine to perform at least four cycles.
7. Set and activate the data acquisition systems for data collection at 10Hz
8. Perform non-destructive cyclic tensile test for an extensions corresponding to 5% smaller than the difference between the unpressurized length and the pressurized length.

The parameters include the initial pressure, the internal bladder material, the elongation rates and the braided mesh type, as detailed in Table 3-1.

Table 3-1: Experiment design parameters

Parameters	Values Tested
Initial internal pressure	210 kPa (30 psi) and 415 kPa (60 psi)
Elongation rates	1mm/s, 6.72 mm/s, and 10 mm/s
Internal bladder material	Butyl Rubber and Silicone
Braided mesh	Yellow-Black FlexoPet and Neon-Blue Over-Expanded Braided Mesh

The choice of the parameters was done to account for the fullest range of behaviour possible. The lowest internal pressure is chosen as the minimum required pressure to inflate the muscle, while the highest internal pressure corresponds to the pressure tranducer range. The internal bladders were chosen based on their compliancy and availability, where butyl rubber is very compliant and silicone is not as compliant. The different braided meshes were chosen based on their number of fibre and coverage of the internal bladder, where the Yellow-Black FlexoPet meshing contains 216 fibers, grouped in threes, and the Neon-Blue Over-Expanded meshing contains 120 fibers, not grouped. The different velocities chosen correspond to the PAM loading rates when applied to human motions, specifically when walking. These were determined by using gait data publicly available, shown in Appendix D. Combining this data with the anthropometric information detailed in section 2.1.4 and performing a kinematic study of the motion, it was possible to estimate the elongation rate at which a PAM would be subjected, if it were located in parallel with the Achilles tendon.

3.4.1.2Experimental Set-up

The PAM internal pressure was measured using a pressure transducer (Texas Instrument C-61CP-022), connected to LabVIEW. The eccentric force and the elongation were measured with a tensile testing machine (Instron 4482) and also recorded by the data-acquisition software, permitting side-by-side comparison of the elongation, pressure, and force. The experimental set-up is shown in Figure 3-5.

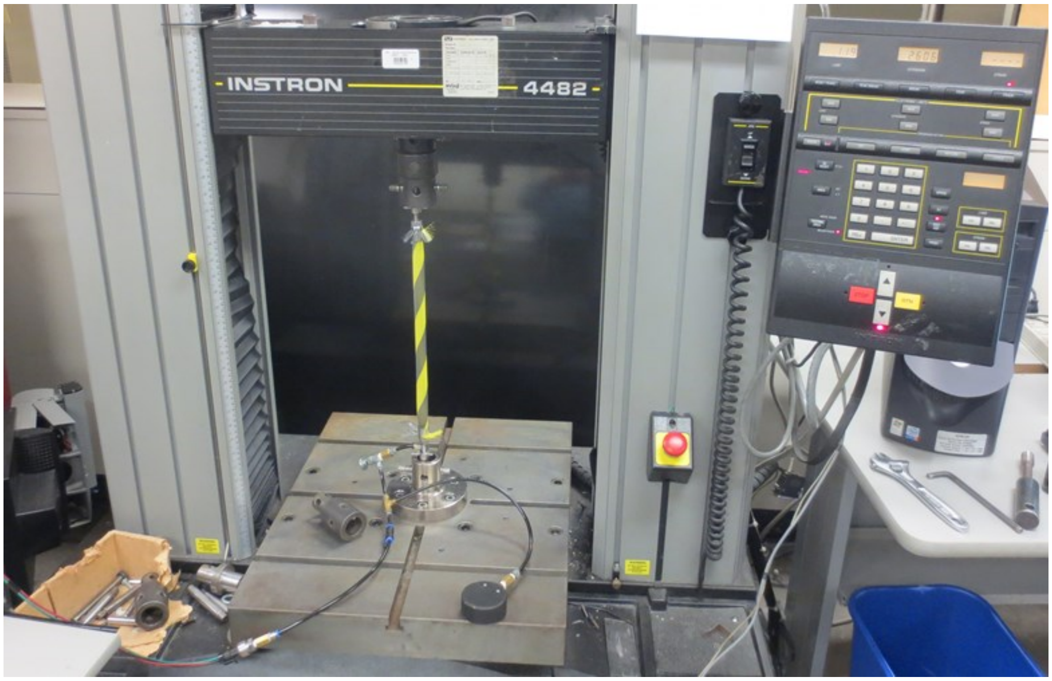


Figure 3-5: PAM cyclic loading set-up

3.4.1.3PAM Labeling

For the experiments, the muscles were labeled according to the tests performed and their geometric properties. The labeling system is nxx_Dxx_Lxx_Pxx_Mxx_Bxx_Vxx, where *nxx* is the number of braid cycle, *Dxx* is the diameter when un-inflated in millimeters, *Lxx* is the length when un-inflated in millimeters, *Pxx* is the initial inflated pressure in kilopascals, *Mxx* is the braid (YB or NB), *Bxx* is the bladder material (BR or SR) and *Vxx* is the velocity in millimeters per second. A sample prototype label is given in Table 3-2.

Table 3-2: Prototype labeling example

Properties	Values
Name	n2.05_D19_L337_P276_MYB_BBR_V1.00
Number of braid Cycle	2.05
Deflated diameter (mm)	19
Deflated length (mm)	337
Initial Pressure (kPa)	276
Braid type	Yellow and Black (YB)
Bladder material	Rubber
Velocity (mm)	1.00

The list of muscles tested is shown in the Appendix C, along with other general properties used in the development of the theoretical and experimental models.

3.4.1.4Experimental Constants

As part of the stiffness model development, the friction coefficients for both friction mechanisms were assumed to be constants. Specific friction coefficients, were not readily available, and were assumed to be 0.30, based on the known ranges for this material [146]–[148].

Finally, the model development used a polytropic process to define the behaviour of the gas inside the PAM. For air, the polytropic exponent varies between 1 for an isothermal process, to 1.4 for an adiabatic process. Since the experimental contractions occur relatively quickly and the bladder wall is thin, it was assumed that the muscle remains at a constant temperature. Therefore, the PAM contraction is assumed isothermal resulting in a polytropic exponent of 1.

3.4.1.5Experimental Stiffness Calculations

The objective of these experiments was to validate the theoretical stiffness model. However, stiffness is not a variable that can be measured directly; rather, it must be calculated from experimental data. The experiment was able to yield pressure, force and elongation data. Consequently, these had to be manipulated to determine the PAM's experimental stiffness.

Going back to the stiffness definition (3-1), it can be interpreted as the instantaneous force variation over the instantaneous length variation. Whereas an instantaneous variation cannot be evaluated directly, considering a change over a small period can approximate this variation.

$$k_{exp} = \frac{F_{inst.}}{L_{inst.}} = \frac{\Delta F}{\Delta L} \tag{3-26}$$

While the difference between consecutive collected points would provide the best approximation for this variation, the data was collected at a frequency of 10 Hz, resulting in consecutive collected points presenting variations smaller than the equipment's rated resolution. As such, the variation was defined as the difference between points over a specified interval. This resulted in the average stiffness over a small period.

$$k_{exp} = k_{i-\frac{x}{2}} = \frac{F_i - F_{i-x}}{L_i - L_{i-x}} \tag{3-27}$$

The remaining factor to consider was to determine the data interval x used to calculate the experimental stiffness. Since tests were done at selected velocities with a constant data acquisition frequency, the data interval was chosen to correspond to the following empirical relationship:

$$x = \frac{Frequency}{Velocity\ in\ mm/s} \tag{3-28}$$

Concretely, this meant that for a velocity of 1 mm/s, the experimental stiffness was calculated with data collected at an interval of 10 points. This loss of sensitivity was considered acceptable due to the amount of data accumulated for each tests performed: a single test acquired between 2 000 and 10 000 data points.

3.4.2 PAM Stiffness Behaviour

Considering that the experiments were performed with various PAM parameters, it was possible to analyze the collected data for each individual parameter by controlling for all the others. This enabled a detailed analysis informing the relevance of each parameter during the validation of the proposed stiffness model. As mentioned previously, the experimental stiffness was obtained through a numerical derivation of the experimental force with regards to the length, as shown in section 3.4.1.5

3.4.2.1 General PAM Behaviour

Each cyclic test performed presented a similar behaviour profile: the PAM stiffness tended to be larger during lengthening compared to shortening, as shown in Figure 3-6. Moreover, for most experiments, the divergence became most significant after having reached 50 to 80% of lengthening, meaning that the loading and unloading stiffness patterns were very similar for small changes of length.

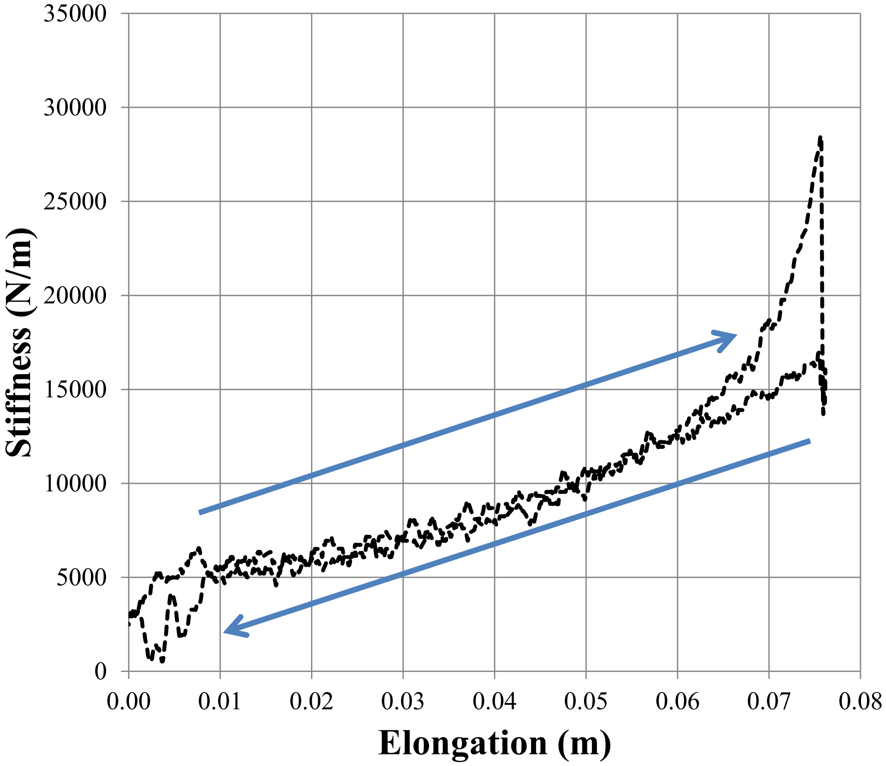


Figure 3-6: PAM profile for prototype n2.82_D13_L316_P276_MNB_BBR_V1.00

This behaviour warrants a more in depth investigation into PAM's behaviour with regards to identify which parameter significantly affect and impact the muscle's behaviour. This analysis is required, as it will dictate the muscle selection, and the design of the exoskeleton in later sections, because the parameters evaluated fall within the range of values needed for the design of the exoskeleton.

3.4.2.2PAM Pressure

The muscle pressure tends to play a significant role in the forces produced by the muscles, as detailed theoretically in the force models previously developed. Then, recognizing that the muscle pressure, along with its derivative, is present in the proposed stiffness model, it is conceivable that it would have a significant impact on the stiffness. In terms of experimental testing, two different initial pressures were tested for each PAM, as shown in Figure 3-7 and Figure 3-8.

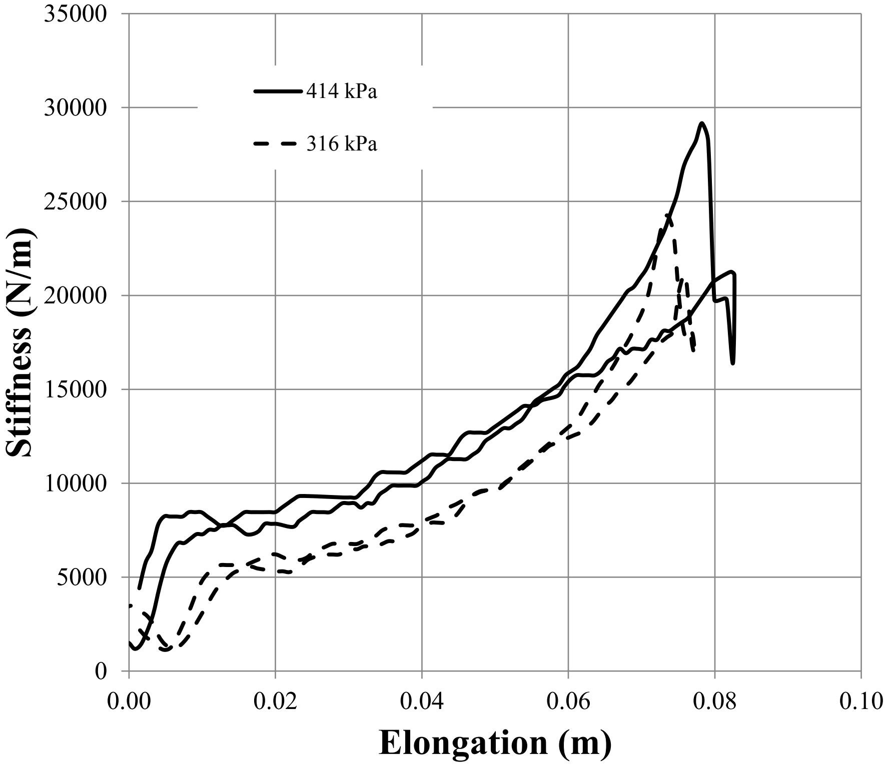


Figure 3-7: Experimental stiffness results for PAM prototype n2.82_D13_L316_MNB_BBR_V10 for two initial pressure of 316 kPa and 414 kPa

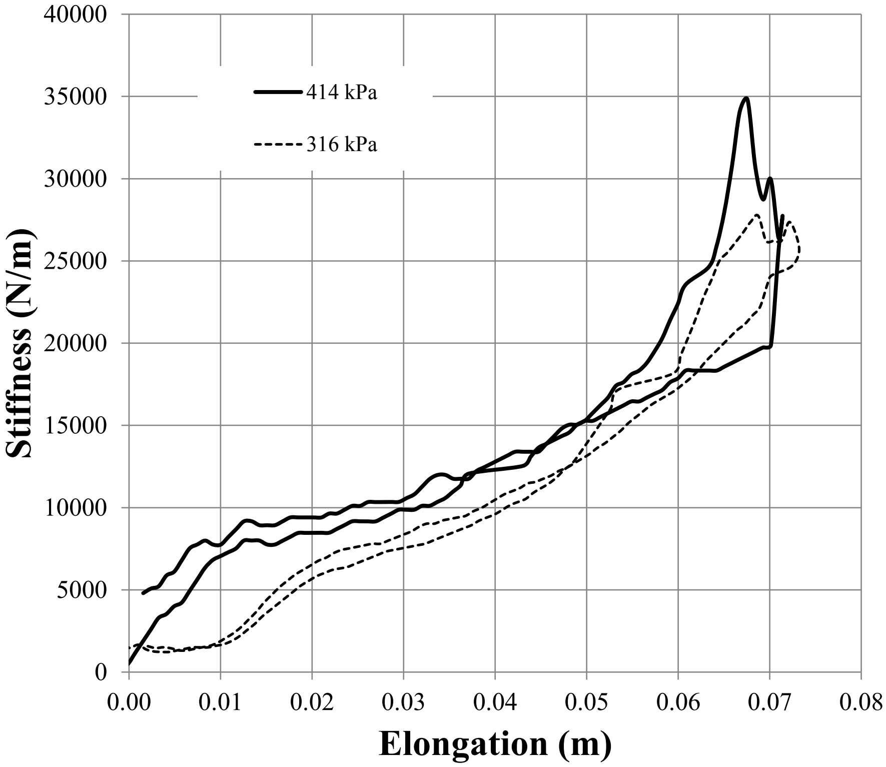


Figure 3-8: Experimental stiffness results for PAM prototype n2.83_D13_L316_MNB_BSR_V10 for two initial pressure of 316 kPa and 414 kPa

For each muscle tested, larger initial pressures were usually associated with larger forces, as expected from the traditional force model (2-19), but did not appear to be linked to larger stiffness's. Instead, the experimental results suggest the initial pressure impacts mostly the initial PAM stiffness, where a higher initial pressure was related to a higher initial stiffness, when controlling for all other parameters.

Such a behaviour was expected, as the proposed stiffness model, detailed in (3-12) and (3-19), is driven not only by the muscle pressure, but also by the pressure variation. This behaviour suggests that the selection of the proper muscle should be done based on the desired minimal stiffness required, which, is strongly linked to the initial muscle pressure.

Another crucial consideration with regards to the pressure is to verify if the pressure within the muscle is well predicted by the proposed model. During the model development, it was assumed that the thermodynamic processes could be simplified by using the polytropic process, as shown in (3-6), which empirically relates the pressure to the volume via a polytropic exponent. Figure 3-9 shows the experimental pressure and the polytropic pressure for an isothermal evolution (n_g=1) and for an adiabatic evolution (n_g=1.4).

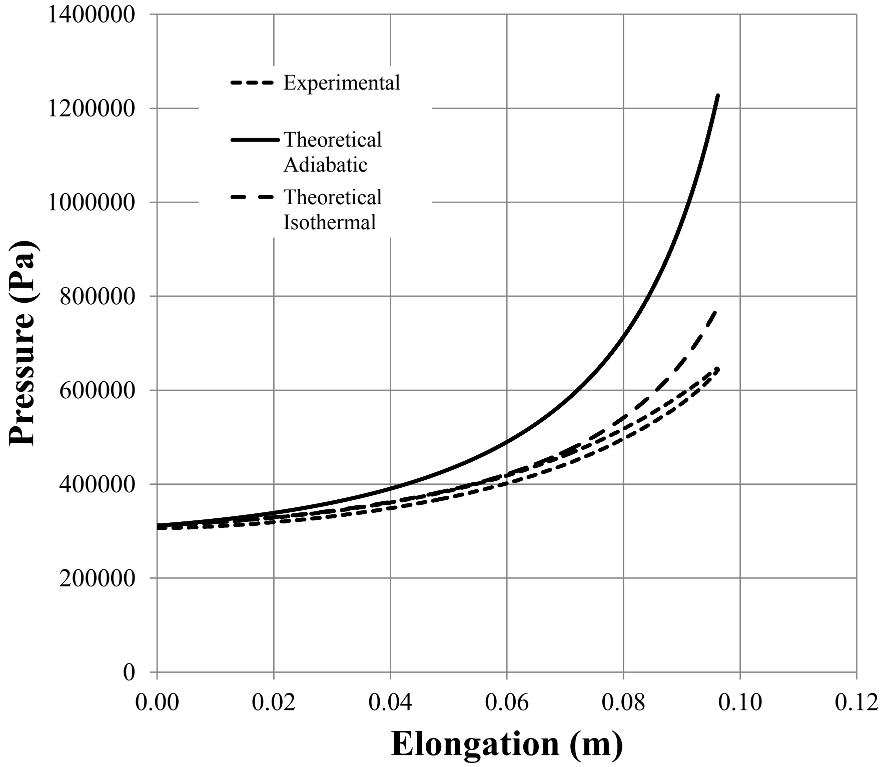


Figure 3-9: PAM n2_D19_L328_P414_MYB_BBR_V1 pressure validation over an eccentric contraction

From Figure 3-9, it can be seen that the isothermal polytropic evolution provides a closer approximation to the experimental pressure, compared to the adiabatic evolution. Furthermore, the experimental pressure presents a hysteresis during the eccentric contraction cycle elongation, which was not modeled by the polytropic process. This suggests that the pressure is not an independent parameter and may depend on other parameters, which might be the bladder material elasticity.

Whereas further research is certainly needed to describe the pressure hysteresis of the PAM, the isothermal polytropic evolution provides a good approximation of the pressure inside the PAM over a contraction cycle, making it suitable for inclusion within the exoskeleton design process.

3.4.2.3Elongation Rate

The motion of the ankle during the gait cycle varies considerably, causing any component influenced by its motion to encounter various velocities during gait cycle. Since the PAM is to be used on the ankle, it is crucial to determine the impact of a range of elongation rate on the PAM stiffness behaviour to account for it during the design process. The different elongation tested were the 1mm/s, 6.72 mm/s and 10 mm/s representing the range of normal velocity for a PAM placed in parallel to the Achilles tendon and subject to normal human motion. Figure 3-10 and Figure 3-11 present the experimental results for PAMs undergoing eccentric contraction at the different velocities, with all other parameters controlled.

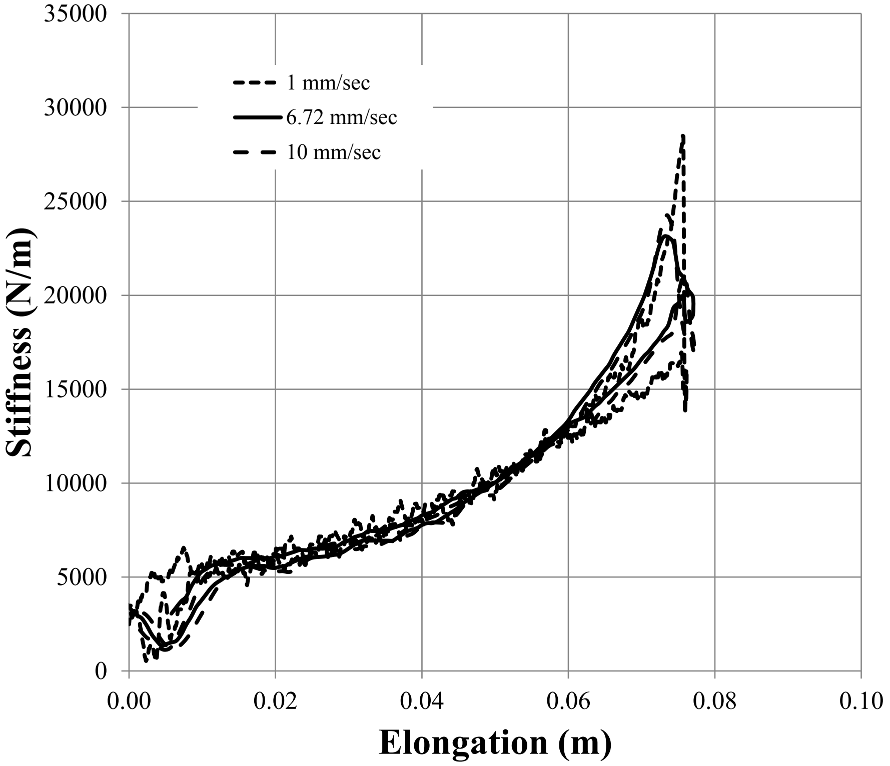


Figure 3-10: Stiffness for multiple elongation rates for PAM prototype n2.82_D13_L316_P276_MNB_BBR

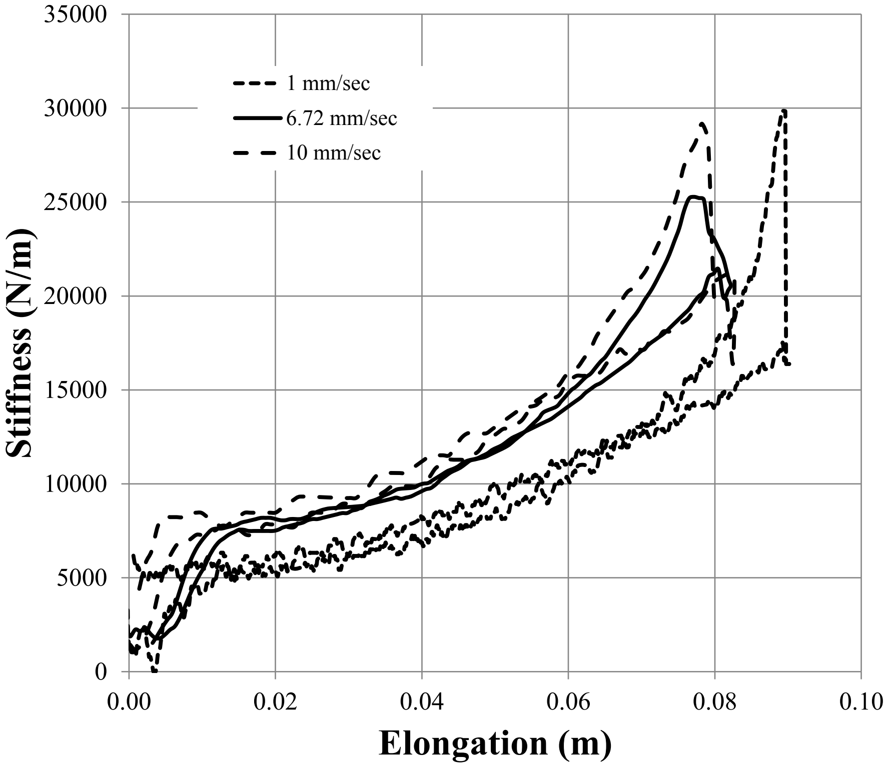


Figure 3-11: Stiffness for multiple elongation rates for PAM prototype n2.82_D13_L316_P414_MNB_BBR

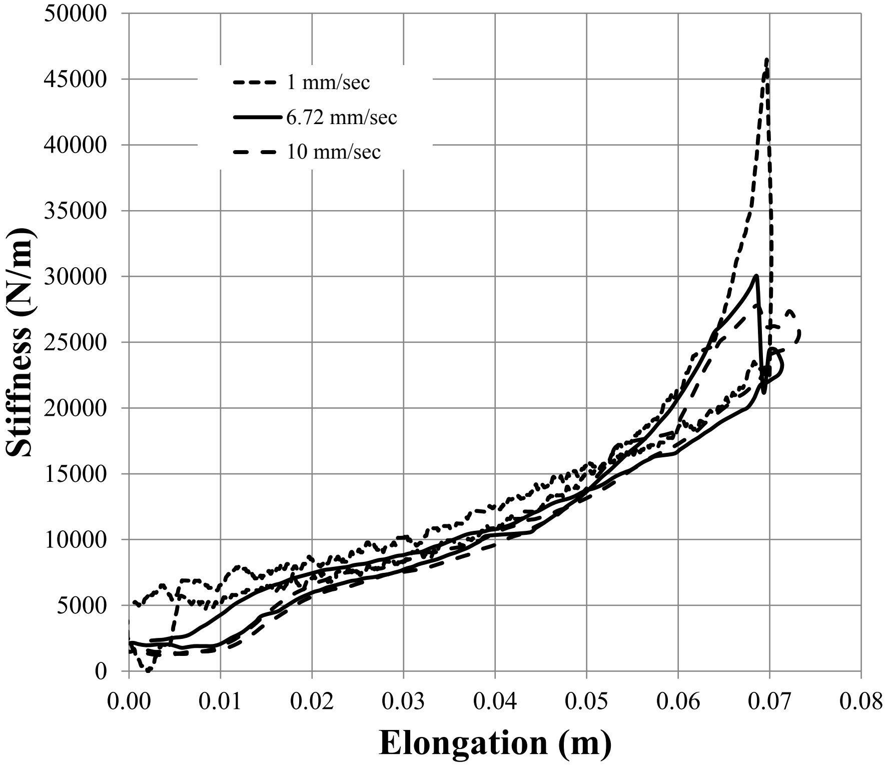


Figure 3-12: Stiffness for multiple elongation rates for PAM prototype n2.83_D13_L317_P207_MNB_BSR

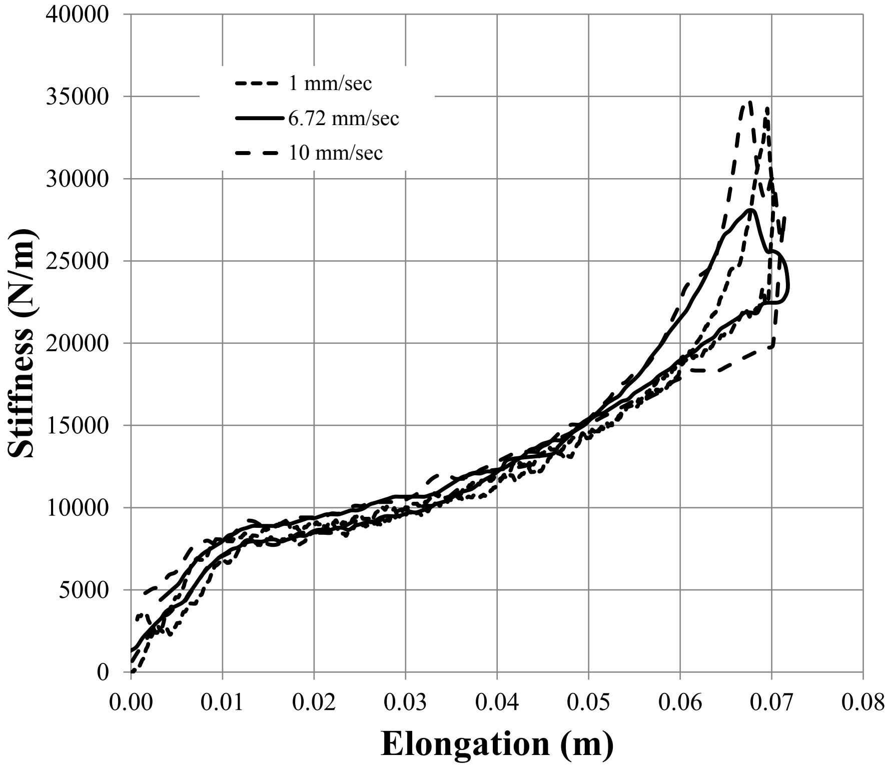


Figure 3-13: Stiffness for multiple elongation rates for PAM prototype n2.83_D13_L317_P414_MNB_BSR

The different elongation rates tested did not seem to have significant impact on the maximum stiffness achieved nor on the rate of change of the stiffness over the elongation range, when controlling for other factors. In general, soft materials present viscoelastic behaviour dependent on the velocity. In this case, the absence of an elongation rate dependent behaviour over the tested range suggests that the elongation velocity is not a critical factor to consider, currently. The main effect appeared to be a reduction in noise in the experimental data, which was expected, as the sampling rate was kept constant.

3.4.2.4PAM Construction Materials

PAMs are composed on an elastic internal bladder, surrounded by an inextensible wire meshing. As described previously, the internal bladder may consist of butyl rubber or silicon rubber, materials commonly used in bicycle tubing and biomedical applications respectively. Two possibilities are also available for the wire meshing, where both are made of Polyethylene Terephthalate (PET) but vary in terms of the number of fibers, and hence, their coverage of the internal bladder. Figure 3-17 and Figure 3-18 present the experimental results for similar PAMs, with different internal bladders (BSR and BBR) and wire meshing (MYB or MNB).

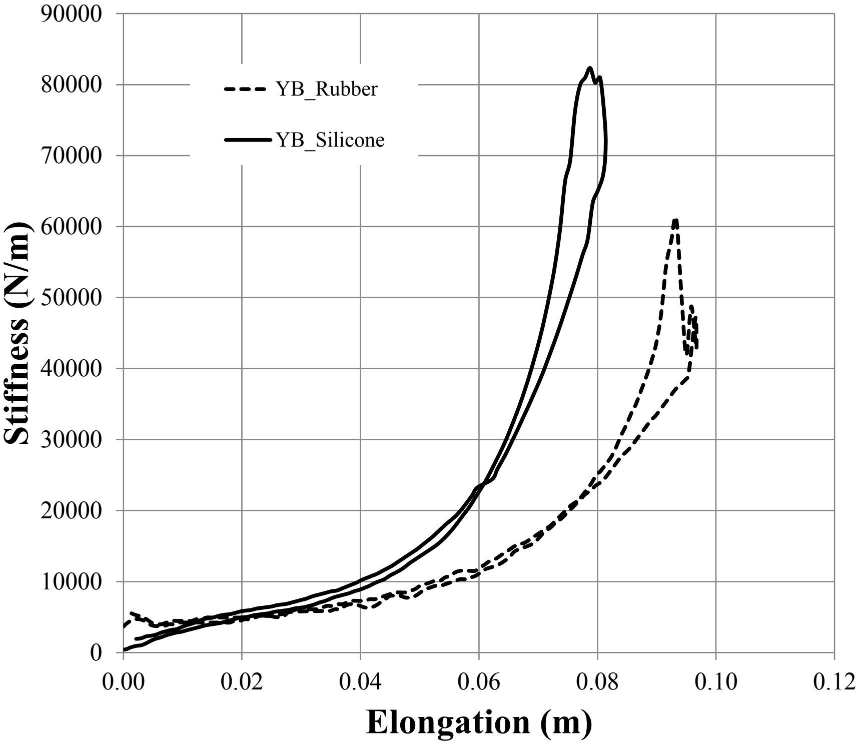


Figure 3-14: Experimental stiffness for PAM prototypes n2.05_D19_L337_P276_MYB_BXX_V10 using internal bladders made of butyl rubber and silicone

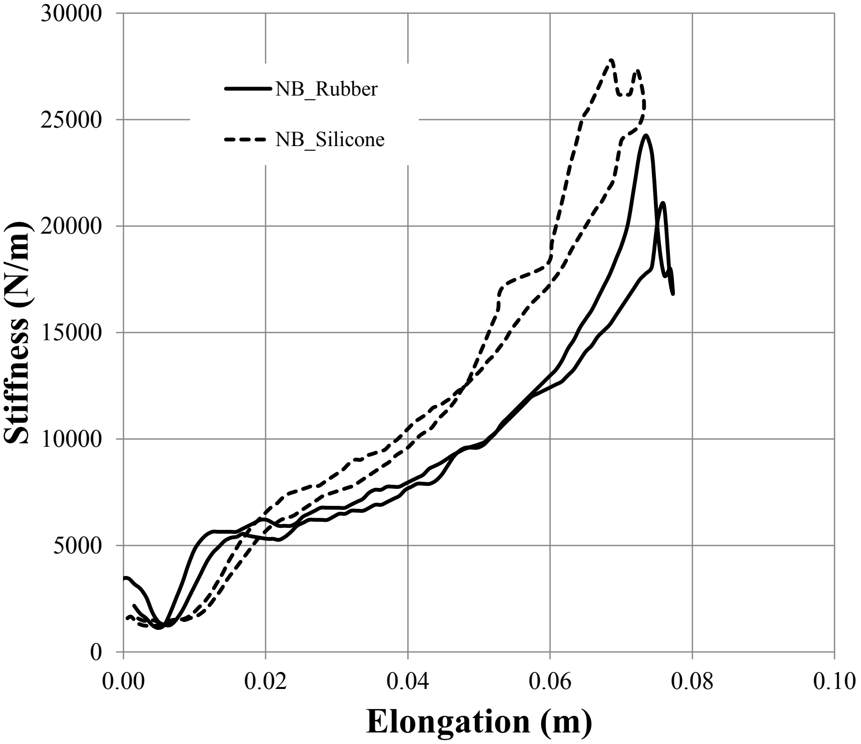


Figure 3-15: Experimental stiffness for PAM prototypes n2.82_D13_L316_P276_MNB_SXX_V10 using internal bladders made of butyl rubber and silicone

The first distinction noticed in Figure 3-14 and Figure 3-15 is that the PAM with internal bladder made of silicon rubber presents steeper stiffness slopes and achieves higher stiffness, irrespective of the wire meshing type, and controlling for all other parameters. This suggests that a more compliant bladder provides smaller stiffness variation over the elongation cycle. An explanation for this behaviour may lie in the minimum initial pressure required to inflate the PAM observed experimentally. A larger initial pressure was needed before radial expansion started occurring for the internal bladder made of silicon compared to that made of butyl rubber. This may influence the capacity of the material to move radially during the elongation cycle.

Presenting the curves for both internal bladder material and both wire meshing permits an analysis of the different meshing used, as shown in Figure 3-16.

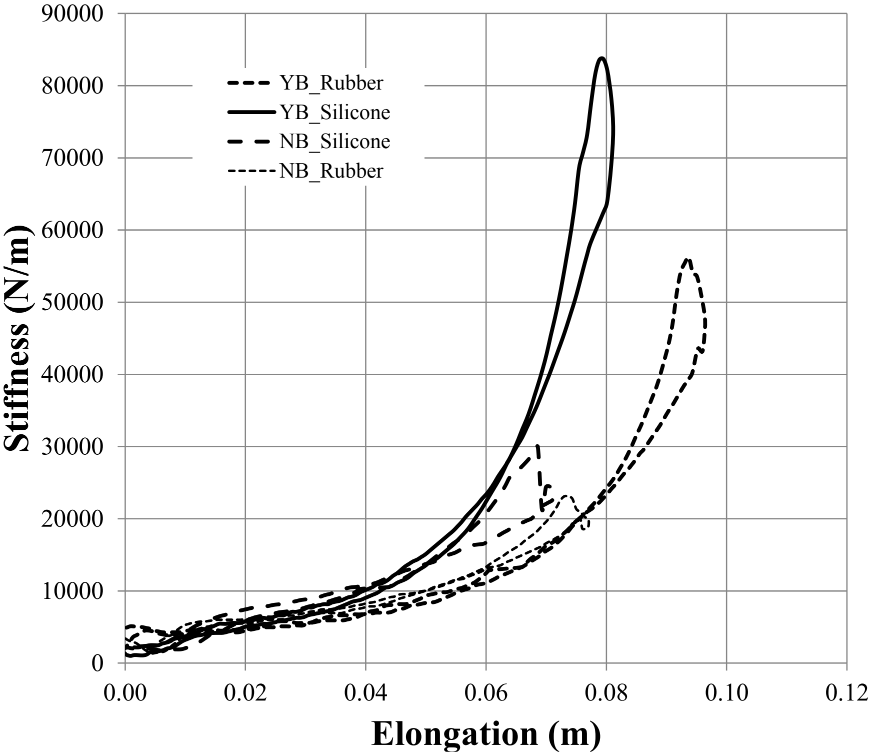


Figure 3-16: Experimental stiffness for PAM prototypes n2.82_D13_L316_P276_MNB_SXX_V6.72 and n2.05_D19_L337_P276_MYB_BXX_V6.72 using internal bladders made of butyl rubber and silicone and different meshings

On Figure 3-16, it is clear that the FlexoPet (YB) wire meshing presents a higher stiffness and elongation compared to the Overexpanded Pet (NB) wire meshing, which are theorized to be linked the meshing geometry and coverage of the internal bladder. The YB meshing presents a larger diameter (19 mm vs. 13 mm), smaller number of turns (2.05 vs. 2.82) and a larger number of fibers (216 vs. 120) compared to the NB braid.

Furthermore, the PAMs with a NB meshing appear to have a hysteresis of similar scale to the PAMs with YB meshing. This is contrary to the theory, because a smaller number of fibers should result in less friction, reducing the scale of the hysteresis. Analyzing the construction of the meshing might provide explanations to this phenomenon. Both meshing have two overlapping family of fibers, but the YB mesh is composed of fibers bundled in groups of three, while the NB mesh does not have bundles. The bundling of the YB mesh was assumed to have negligible effect. However, it appeared as though this structure solidified the overall PAM by increasing the internal bladder covered.

3.4.3 Stiffness Models Validation

Having evaluated the different PAM parameters essential for the development of the ankle exoskeleton, it is important to validate the proposed theoretical model with the experimental data to assure that the model adequately predicts the muscle behaviour. This section presents experimental results along with the proposed theoretical model for validation purposes. Also included in the validation is the Colbrunn stiffness model (2-26) for comparison to the current standard stiffness model. A selection of results is shown herein.

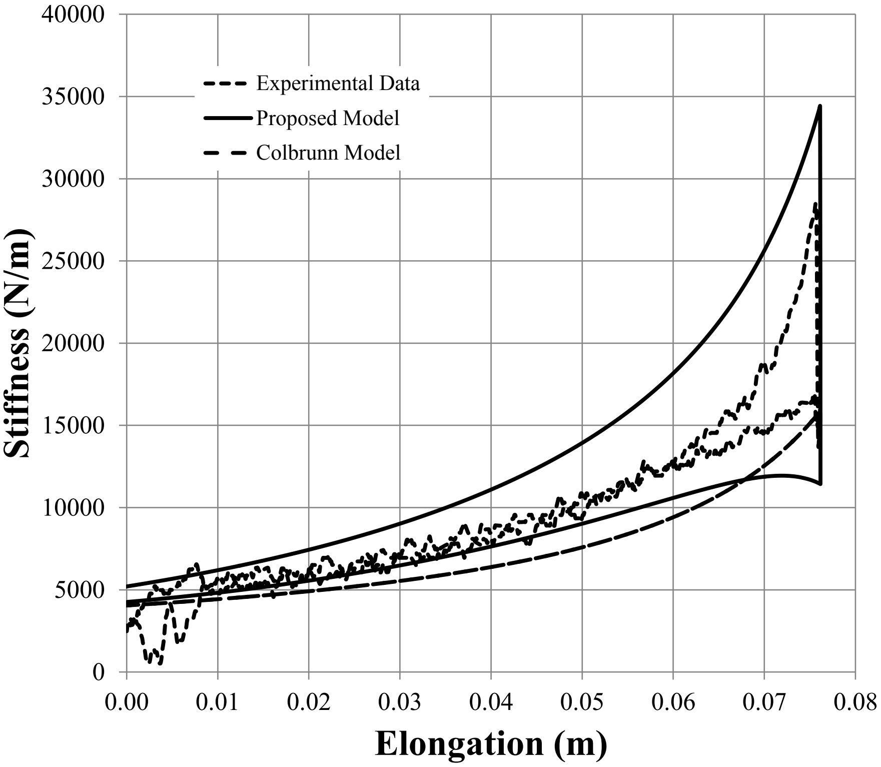


Figure 3-17: Stiffness model validation for PAM prototype n2.82_D13_L316_P276_MNB_BBR_V1.00

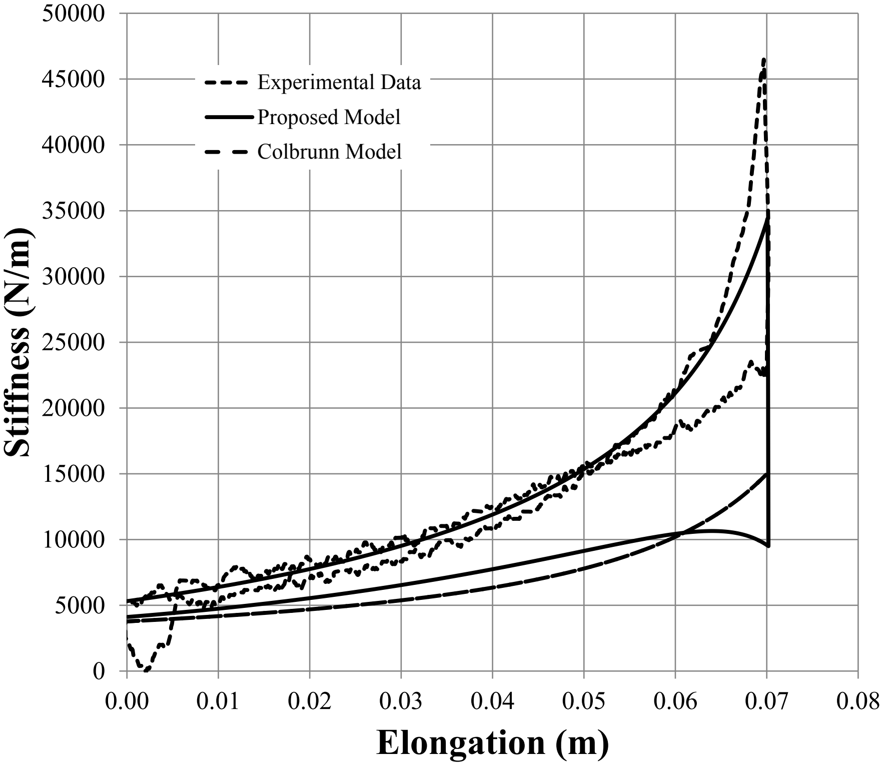


Figure 3-18: Stiffness model validation for PAM prototype n2.83_D13_L317_P207_MNB_BSR_V1.00

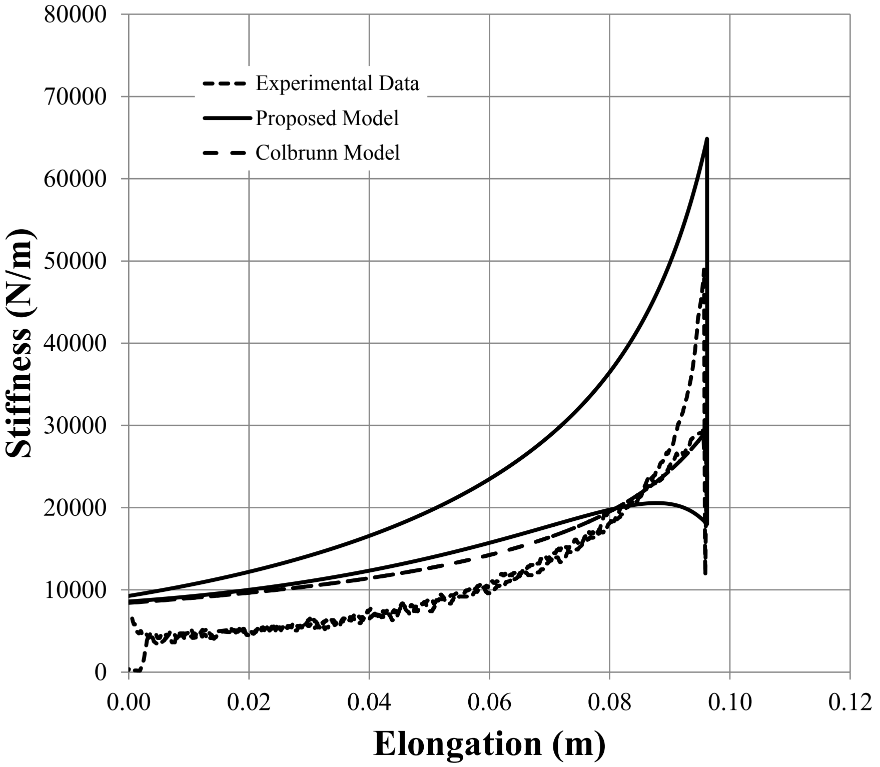


Figure 3-19: Stiffness model validation for PAM prototype n2.05_D19_L337_P276_MYB_BBR_V1.00

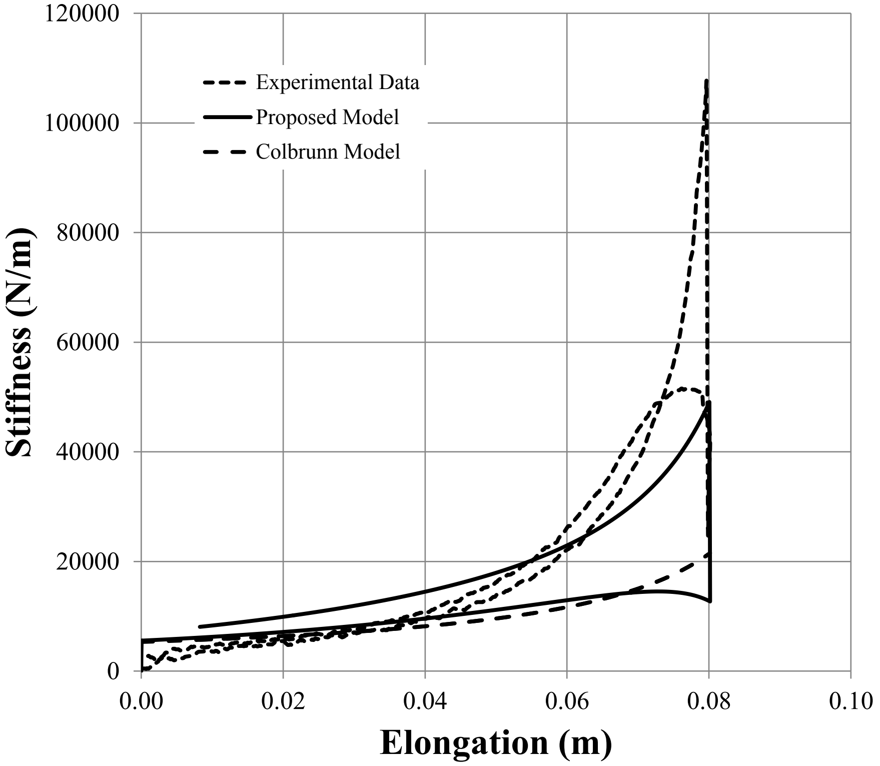


Figure 3-20: Stiffness model validation for PAM prototype n2.05_D19_L337_P207_MYB_BSR_V1.00

Analysis of Figure 3-17 to Figure 3-20 indicates that the proposed model consistently under-evaluates the experimental results, by 5 to 25%, for PAMs using a silicone bladder, and provides bounds for PAMs using a butyl rubber bladder. This behaviour can be explained by restating the underlying assumption of bladder compliancy: it was assumed, during thin-pressure vessel analysis, that the internal bladder was thin, such that internal wall forces did not change amplitude over its thickness. This assumption is directly linked to the bladder compliancy, as a non-compliant material will not provide any opposition to external forces, which will allow internal wall forces to be constant.

Furthermore, the proposed model over-predicts experimental results for PAMs using YB mesh, over the entire contraction cycle. This, as explained previously, is thought to be linked to the bundling of the YB mesh in groups of three, resulting in a surface area different from what was modeled in the proposed model.

Therefore, the proposed model is better suited to predict the experimental behaviour of PAMs using more compliant internal bladder (butyl rubber) and single strand mesh (NB). These results relate to the assumption adopted for the development of the proposed model, namely, neglecting bladder mechanical properties and assuming that the mesh is inextensible.

In comparison, the Colbrunn model [141], the current modeling standard consistently predicted conservative values compared to the experimental data, by 10 to 50%, which fell within a similar range as the shortening portion of the proposed model

The proposed theoretical model was able to account for the PAM hysteresis and predicted the experimental stiffness more accurately that the Colbrunn model.

3.5 Conclusion

The present chapter consisted in the development and validation of a theoretical stiffness model for PAM that incorporates factors accounting for internal pressure, muscle geometry and friction between fibers. Validation of the developed model consisted of a series of eccentric contractions, performed on a tensile testing machine, where force, elongation and internal pressure were measured. The experimental stiffness was determined by deriving the force-elongation data.

The parameters tested included elongation velocity, initial internal pressure, braided mesh, and internal bladder. According to the results, the proposed frictional stiffness model presents fairly accurate results, especially for PAMs with butyl rubber internal bladders and NB braided mesh, as they reflect the adopted assumption for model development. Moreover, the experimental data suggests that the stiffness is larger when the PAM is being pulled as opposed to being released, as predicted by the proposed model.

The internal muscle pressure was assumed to follow an isothermal polytropic evolution. This assumption was confirmed by showing that the experimental pressure followed this behaviour very closely. The experimental pressure presented a hysteresis, thought to be linked to the internal bladder's mechanical behaviour. Since the isothermal polytropic process modeled the experimental reasonably well, this process was considered satisfactory to be included in the design of exoskeleton.

The impact of the initial internal pressure was minimal over the elongation range but seem to only impact the initial PAM stiffness. This can be explained by noticing that a stiffer bladder required a larger initial pressure before radially expanding. Similarly, the impact of the elongation rate was minimal when looking at the stiffness rate of change.

The proposed model better predicted the behaviour of the most compliant internal bladder, made from butyl rubber, as internal bladder compliancy was one of the implicit assumptions of the model.

These findings will be incorporated, as shown in section 4.2.3, in the development of a selection tool that automatically identifies the ideal set of PAM characteristics for the unpowered ankle exoskeleton, based on the ankle power and torque requirement through the gait cycle.

In terms of future direction, a stiffness model that incorporates the viscoelastic behaviour of the internal bladder and that of the mesh might provide a better estimate of the PAMs stiffness over an eccentric contraction cycle. Furthermore, temperature changes over a contraction cycle should be verified to ascertain that a polytropic process is an appropriate assumption.

Chapter 4 – EXOSKELETON DESIGN AND MODELING

This chapter presents the design of an unpowered ankle exoskeleton that aims to harvest gait energy through the stance phase and release the accumulated energy at push off, in a controlled manner, thusly assisting the plantar flexion moment and reducing the ankle power.

The exoskeleton is composed of two structural pieces around the feet and shank, which articulate at the ankle, allowing ankle flexion and extension. The device has a PAM located at the back of the leg, parallel to the Achilles tendon that stores and release the energy. A sequencing mechanism, composed of cams, pins, springs and ratchet, controls the timing of PAM release. The device is worn over the shoe and attaches to the user via straps at the mid-shank and on the feet.

The design of any complex device requires a thorough design process justifying all design decisions made over its development. The design of the unpowered ankle exoskeleton followed a three-step methodology: the Initial Proposal, the Conceptual Design and the Detailed Design. During the Initial Proposal, the objectives and requirements are identified. During the Conceptual Design, using an iterative process, a global solution respecting the objectives and the requirements is developed. During the Detailed Design, the global solution is refined by dimensioning all parts and finding the correct materials using engineering analyses.

4.1 Initial Proposal

The design objectives of the exoskeleton can be summarized as follows:

- Assist: The device should provide assistance at the ankle during push off to 25% of the maximum ankle moment (See Table 2-3).
- Motion: The device should allow 45º of ankle plantar flexion and 20º of ankle dorsiflexion, such that the ankle movements are not hindered.
- Energy: Using no external or internal electrical energy, the device should reduce the energy consumed by the user compared to walking without the device
- Technology: Use of mechanical passive behaviour of PAM and timing mechanism to store energy during the stance phase, release it at toe-off and let the ankle free motion during the swing phase of gait.

To complement the objectives, a series of desirable characteristics were developed, allowing a better identification of the challenges and barriers facing the development of the exoskeleton. The device should:

- Weigh a maximum of 2.5 kg to reduce any pendulum effect
- Low maintenance and operate efficiently in harsh weather (sand and snow)
- The device should be passive and avoid the use of a on-board or external power source and electronic devices (sensors).
- The device should not cause injury to the user by pinching, twisting, cutting, or puncturing, causing excessive joint motions, breaking in normal daily motions, or any other behaviour that could cause injuries to the user.

4.2 Conceptual Design

The conceptual design is an iterative process where different problems pertaining to specific aspects of the exoskeleton are identified and solved, while respecting the objectives and requirements outlined previously. Since multiple solutions can exist for each problem, the final solutions are identified using the General Morphological Analysis [143].

The ankle can move in the three anatomical planes. The most significant motions occur in the sagittal plane and consist of the dorsiflexion and plantar flexion. The selection process is used to determine how many movements are permitted at the ankle joint. All other selection processes are displayed in 0.

Eight selection criteria (e.g.: device size) were developed based on the objectives and the requirements for the ankle motions. Each criterion was weighed against one another to determine their relative importance with regards to the ankle motion (e.g.: size has a weight of 12). For each criterion, the different combinations of ankle motions are scored (e.g.: dorsiflexion and plantar flexion got a score of 28 in the size criterion). The solution with the highest weighted-score is the winning solution, as shown in Table 4-1, corresponds to only allowing the dorsiflexion and plantar flexion motion. Multiple iterations of each decision analysis matrix were iteratively developed to assure compatible results.

Table 4-1: Ankle motion decision analysis matrix

Criteria	Weight	Dorsi/Plantar		Dorsi/Plantar + Ev./Inv.		Dorsi/Plantar + Rot.		All		Total
		Score	W.S.	Score	W.S.	Score	W.S.	Score	W.S.	
Size	12	28	336	26	312	24	288	22	264	100
Weight	15	27	405	25	375	25	375	23	345	100
Effectiveness	14	21	294	24	336	26	364	29	406	100
Complexity	14	30	420	24	336	25	350	21	294	100
Energy Consumption	11	29	319	24	264	24	264	23	253	100
Ease-of-use	12	22	264	25	300	26	312	27	324	100
Safety	11	25	275	22	242	26	286	27	297	100
Comfort	11	23	253	24	264	26	286	27	297	100
Total	100		2566		2429		2525		2480	

For the ankle motion, the solution with the highest score is the dorsiflexion and plantar flexion only. It can be noted that the solution that also includes ankle rotation is very close in score and might provide a better alternative if a simple technology that does both motion is simple an effective. The criteria selected by using the decision analysis matrix are shown in the table below:

Table 4-2: Winning criteria selection

Selection Criteria	Winning Solution
Ankle Motion	Dorsiflexion and Plantar flexion
Ankle Dorsiflexion and Plantar flexion	Ratchet
Foot Flexion and Extension Method	Free Motion
Foot Permitted Motion	Flexion and Extension
Attachment	Moulded Orthoses

4.2.1 Overall Design

The design for the exoskeleton is shown in Figure 4-1 and Figure 4-2. Its goal is to reduce the effort at the ankle by loading a PAM during the stance phase and releasing it at the toe-off to provide a forward momentum to the user. The design is entirely passive, using a combination of a ratchet and a sequencing mechanism to uni-directionally lock the ankle during stance and unlock it during the swing phase of gait. The exoskeleton is attached to the user via straps on the shank and foot.

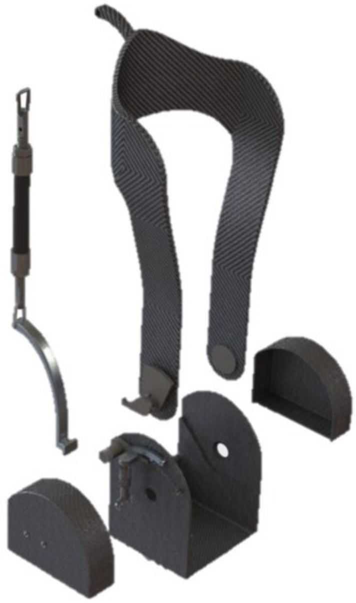


Figure 4-1: Exploded view of major component of the exoskeleton

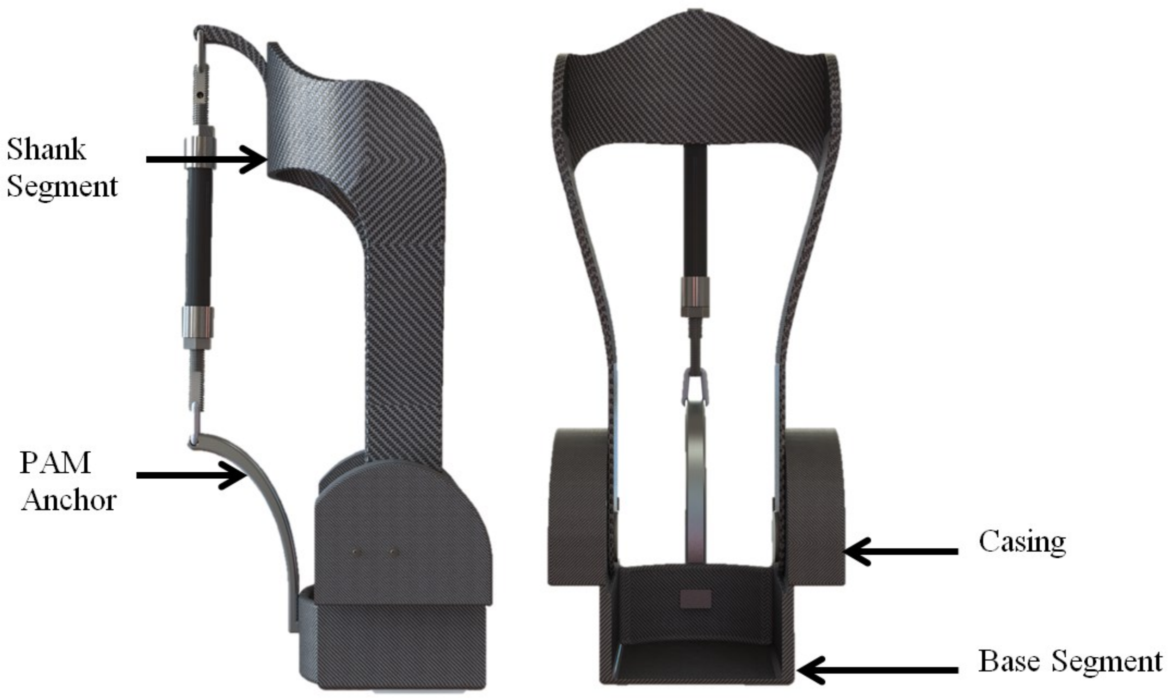


Figure 4-2: Annotated exoskeleton side (left) and front views (right)

4.2.2 Structure

4.2.2.1Shank Segment

The shank segment, shown in Figure 4-3, spans about half of the shank and attaches to the user with a strap. It consists of a two bars parallel to the shank, which merge a curved section conforming to the user's calf, reducing the motion of the device relative to the user. The parallel bars articulate with the Base Segment at the ankle, via a splined shaft. The structure is made of a composite material to enhance the user's comfort. A proximal anchor point for the PAM is positioned at the back of the curved sections.

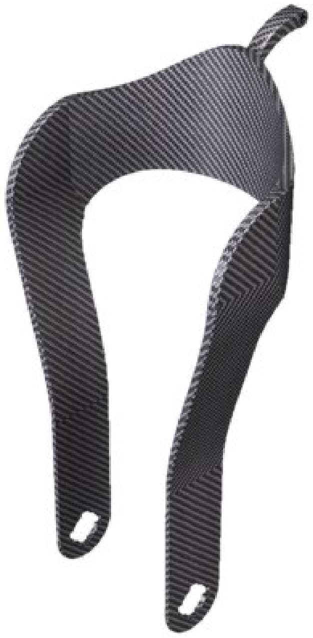


Figure 4-3: Shank segment isometric view

4.2.2.2Base Segment

Structurally, the Base Segment is the most important piece of the exoskeleton as it attaches to the user's foot via a conforming back support, a strap and a plate underneath the foot that comes into contact with the ground. It provides a distal anchor point to the PAM, through a lever for that purpose. Finally, the Base Segment is the main structure supporting the timing mechanism and articulates with the Shank Segment to form the exoskeleton ankle joint. Its bottom plate spans about half of the foot length to allow the user's toes to freely flex. This structure must be very light, to limit excessive loads at the hip, but also strong, to support stresses caused by ground contact, the PAM, and the user interaction.

As shown in Figure 4-4, both side sections have a step, reducing their thickness below the hole for the ankle shaft. These steps serve as mechanical stops to prevent ankle motion beyond 45 degrees of plantar flexion, in line with normal range of motion shown in Table 2-1.

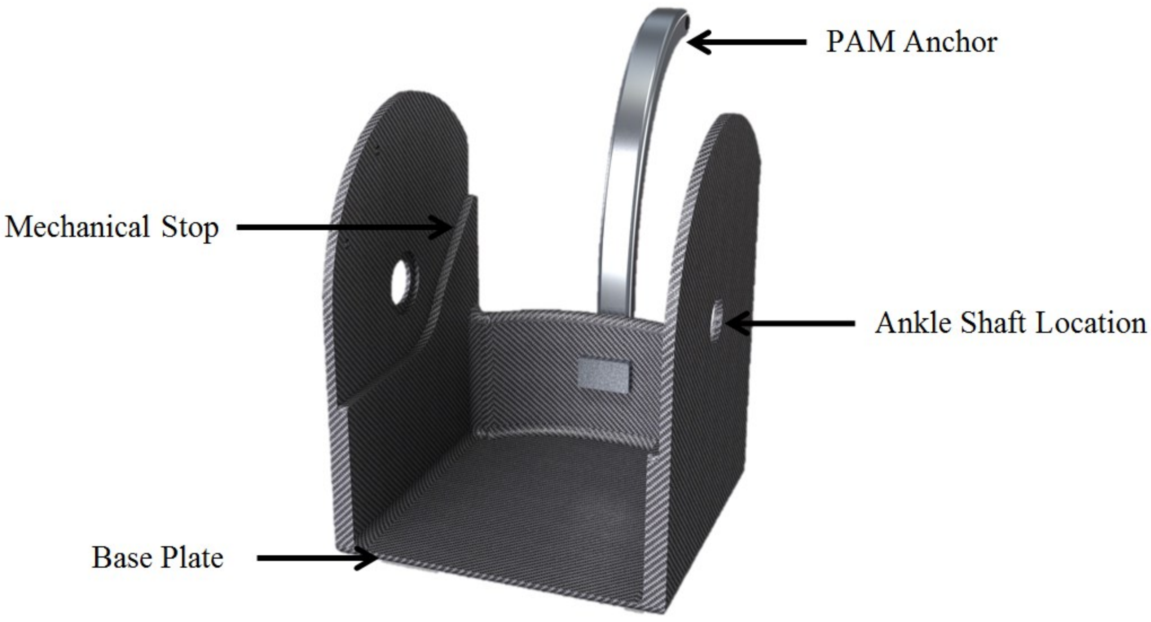


Figure 4-4: Annotated ankle segment dimetric view

4.2.2.3 Casing

As its name suggests, the Casing, shown in Figure 4-5, encloses the timing mechanism, protecting it from dust, debris, and protecting the user from accidental injury. The Casing conforms to the rounded shape of the base segment and attaches to it via small, embedded screws. Moreover, the Casing, in combination with the Base Segment, provides support to timing mechanism, which will be detailed later. Several components of the timing mechanism are attached to the Casing using screws.

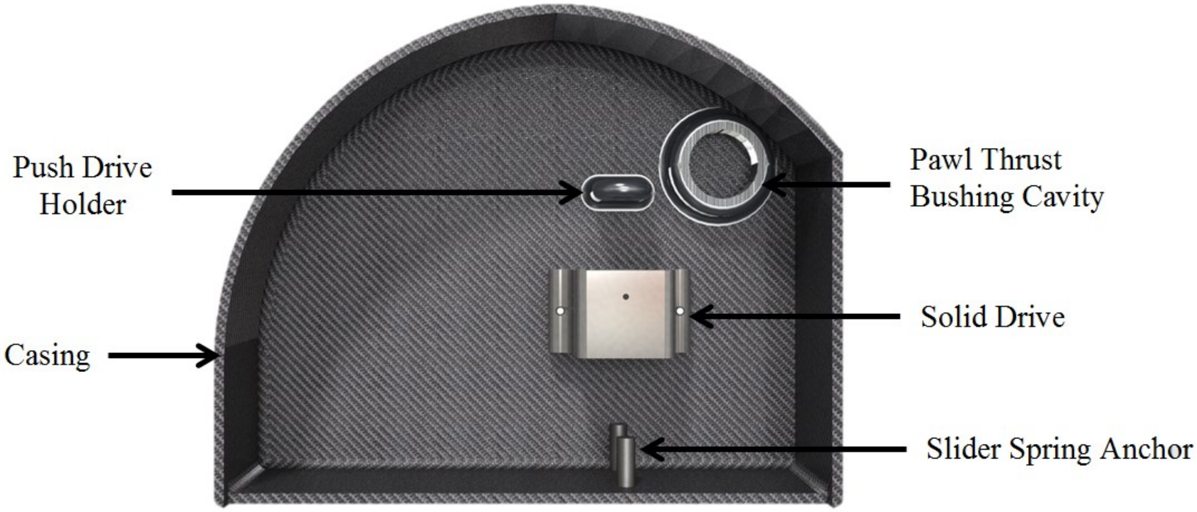


Figure 4-5: Annotated casing assembly side view

4.2.3 Pneumatic artificial muscle

The objective of the exoskeleton is to harness the non-linear passive behaviour of the PAM to provide energy at toe-off. The PAM is located at the back of the user's leg, anchored on the Base Segment distally, and on the Shank Segment proximally. In both cases, the PAM takes advantage of curved beams to create significant lever effects and achieve higher moments about the ankle.

To attach the PAM end fixtures to the anchors, it is proposed to use simple mechanical attachments. These attachments must be easy to attach and detach, while being able to withstand large forces, similar to climbing hooks.

In terms of PAM material to be used by the device, it is proposed to use the butyl rubber internal bladder with the Overexpanded PET braided mesh because the PAM stiffness model showed better predictive capacity with that configuration, as detailed in Section 3.3.2.

4.2.4 Timing Mechanism

As explained in the section 2.1.2 Gait Analysis, a thorough understanding of the ankle gait graphs is crucial to determine the energy recovery opportunity during gait. The ankle creates maximum power and moment at the toe-off, intended to push the entire body forward and give the leg enough energy to swing passively. Inspecting the gait graph for the ankle angle, the ankle performs a flexion-extension cycle in the stance phase (0-60%) and in the swing phase (60-100%).

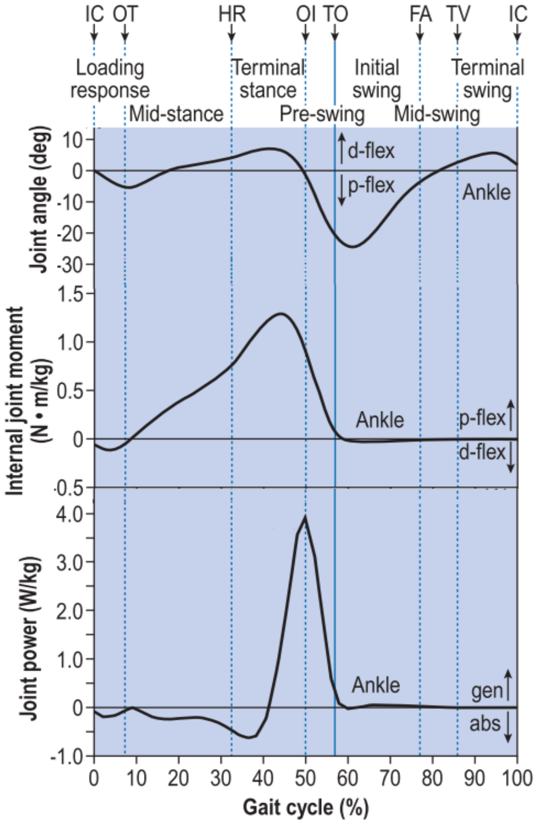


Figure 4-6: Ankle gait graphs **[11]**

Taking into account these factors, charging of the PAM must occur during the stance phase only, taking advantage of the body's momentum about the ankle. Charging during the swing phase is not possible as there is no power or moment at the ankle joint. Furthermore, PAM discharging must occur immediately before toe-off so as to release a maximum of energy from the PAM at toe-off and limit the ankle power required from the user. If this timing is incorrect, the user will have to produce larger ankle power than necessary.

The novelty of the exoskeleton lies in the development of a simple passive mechanism that charges the PAM during the stance phase, releases the PAM at the moment where most power is required, and allows free ankle motion during the swing phase. Therefore, a first mechanism should charge and release the PAM, while a second mechanism should sequentially activate the PAM charging mechanism based on the gait cycle. Figure 4-7 shows a cut view of the different components making up the timing mechanism, located inside the casing. As can be seen, the timing mechanism is composed of springs, ratchet, cams and pins.

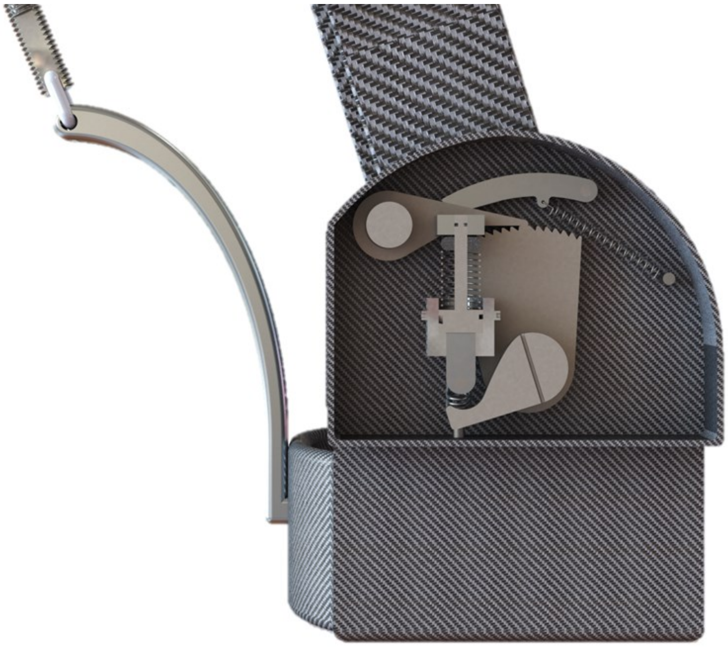


Figure 4-7: Cut-view of timing mechanism

4.2.4.1PAM Charging Mechanism

The objective of the PAM charging mechanism is to uni-directionally lock the Base Segment to the Shank Segment, thereby charging the PAM when the ankle undergoes dorsiflexion motion. In trying to keep the system as a simple as possible, the charging mechanism consists of a ratchet, pawl and toggle arm, as shown in Figure 4-8. The ratchet is mounted on the Ankle Shaft, which is the shaft that articulates the Base Segment and the Shank Segment. Then the motions of the ratchet correspond to the actual motion of the user's shank with regards to the user's foot.

The self-engaging ratchet and pawl were designed following the methodology laid out in [149]. During the design process, the ratchet was designed to have 100 teeth if it were circular, for an accuracy of 3.6 degrees. However, to reduce the amount of space required, only a section of the ratchet containing 14 straight teeth was selected. This corresponds to a range of about 50 degrees, which is larger than the ankle motion during a typical gait.

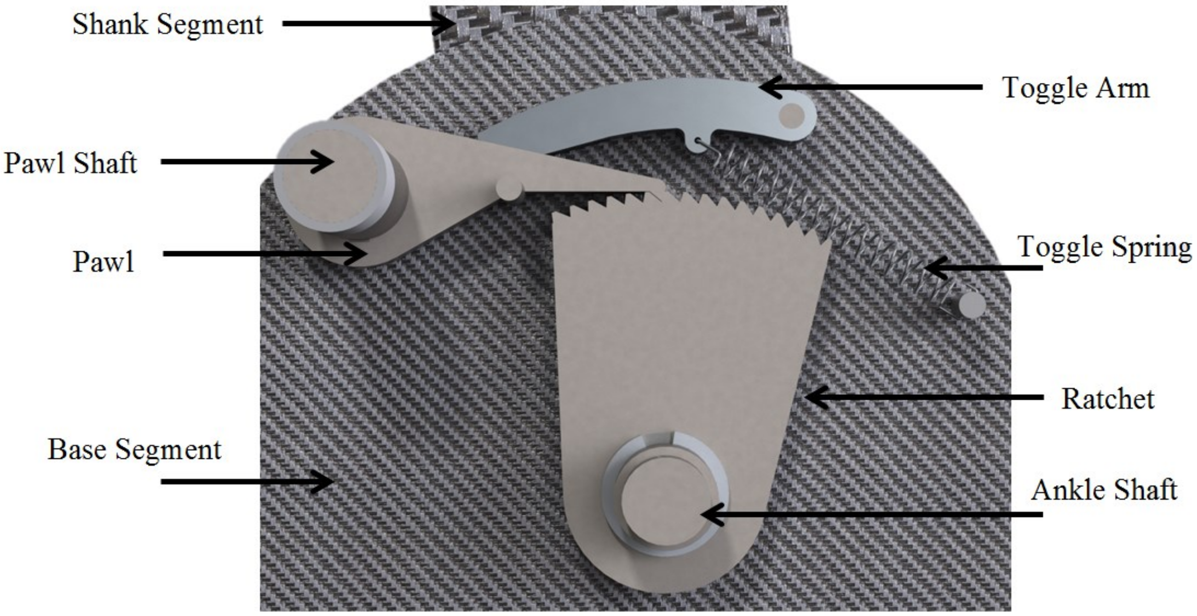


Figure 4-8: Annotated PAM charging mechanism

In terms of operation, the engaged pawl prevents the ratchet from moving in a counter-clockwise direction while allowing it to move clockwise. When the pawl is disengaged, the ratchet is free to rotate in any direction.

The engagement of the pawl to the ratchet is governed by two components: the toggle arm and the sequencing mechanism, which the latter will be discussed in the following section. The toggle arm consists of a curved piece, which presses on the superior part of the pawl at its midpoint, pushing the pawl to engage with the ratchet. This force is created via a tension spring pulling on the toggle arm, forcing it downwards.

The main drawback of using a ratchet system is its disengaging behaviour. The self-engaging nature of the ratchet system means that the forces to disengage the pawl are considerable. As such, it is common practice to minimally counter-rotate the ratchet to remove the forces on the pawl, before disengaging it. However, this is not a viable solution for the current system, as it would either modify considerably the user’s gait, or require the presence of actuators.

Instead, the proposed PAM charging system uses a hinge joint in the middle of the pawl to disengage it from the ratchet. The pawl hinges at its bottom surface, allowing it to fold upwards, as shown in Figure 4-9. The pawl also presents mechanical blocks to prevent motion in the opposite direction, as shown in Figure 4-10

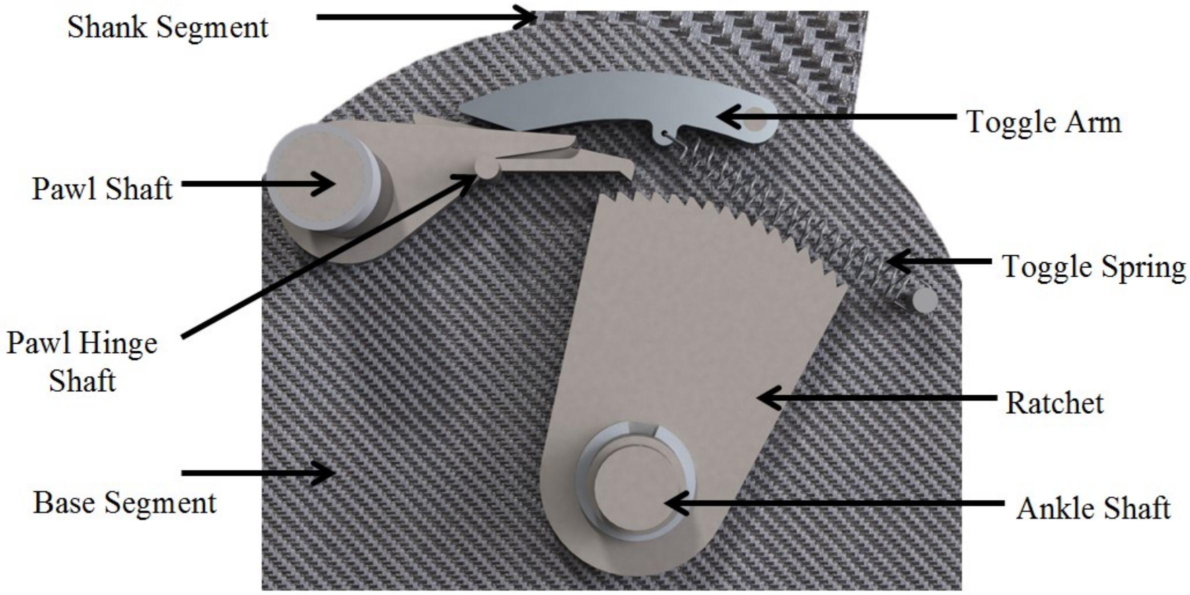


Figure 4-9: Disengaged PAM charging mechanism

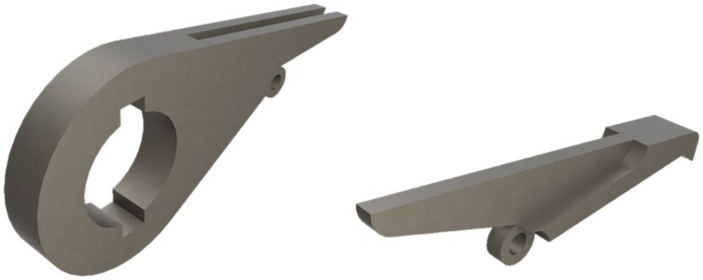


Figure 4-10: Pawl back piece (left) and pawl front piece (right)

In common mechanisms, the pawl is generally forced downward via a torsion spring, as opposed to a toggle arm. In this case, the toggle arm has a dual purpose: it serves to force the pawl to engage, but also prevents the pawl from moving at its hinge when the pawl is engaged. Thusly, the toggle arm must be positioned directly above the pawl hinge joint.

In summary, the PAM charging mechanism uses a ratchet and pawl system to uni-directionally lock the user's shank with regards to the user's foot, by placing the ratchet on the same shaft that articulates the Base Segment to the Shank Segment. The pawl has a hinge along its length to disengage the ratchet system at the correct timing. The toggle arm engages the pawl to the ratchet and prevents the pawl from buckling when engaged.

4.2.4.2Sequencing Mechanism

The sequencing mechanism is responsible to set the activation of the PAM charging mechanism according to the gait cycle. As mentioned previously, the PAM charging mechanism must be able to lock and unlock during specific phased of the gait cycle, to promote proper charging and discharging of energy in the PAM.

Analyzing the ankle angle through the gait cycle, as shown in Figure 4-11, it can be noticed that the maximum dorsiflexion is reached once in the swing phase and once in the stance phase. Considering that the dorsiflexion in the swing phase usually is smaller than in the swing phase, the point maximum dorsiflexion can be used as an activation point for the sequencing mechanism. The first time the mechanism reaches this dorsiflexion, the mechanism activates the PAM charging mechanism. The second time the mechanism reaches this same position, the PAM charging mechanism is deactivated.

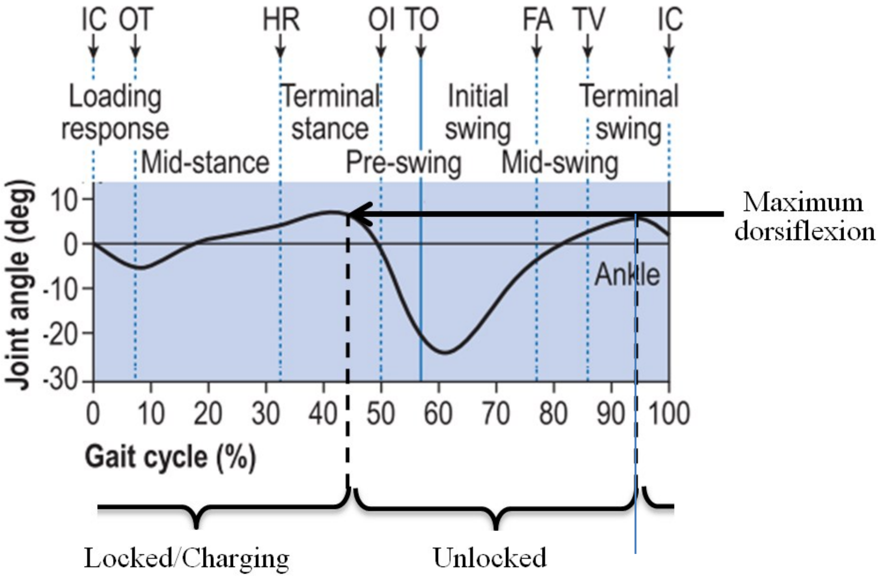


Figure 4-11: Ankle angle during gait

Noticing that this sequencing behaviour is similar to that of a click-pen, the click-pen mechanism was adapted to the current situation. Figure 4-12 shows the layout of the sequencing mechanism inside the casing

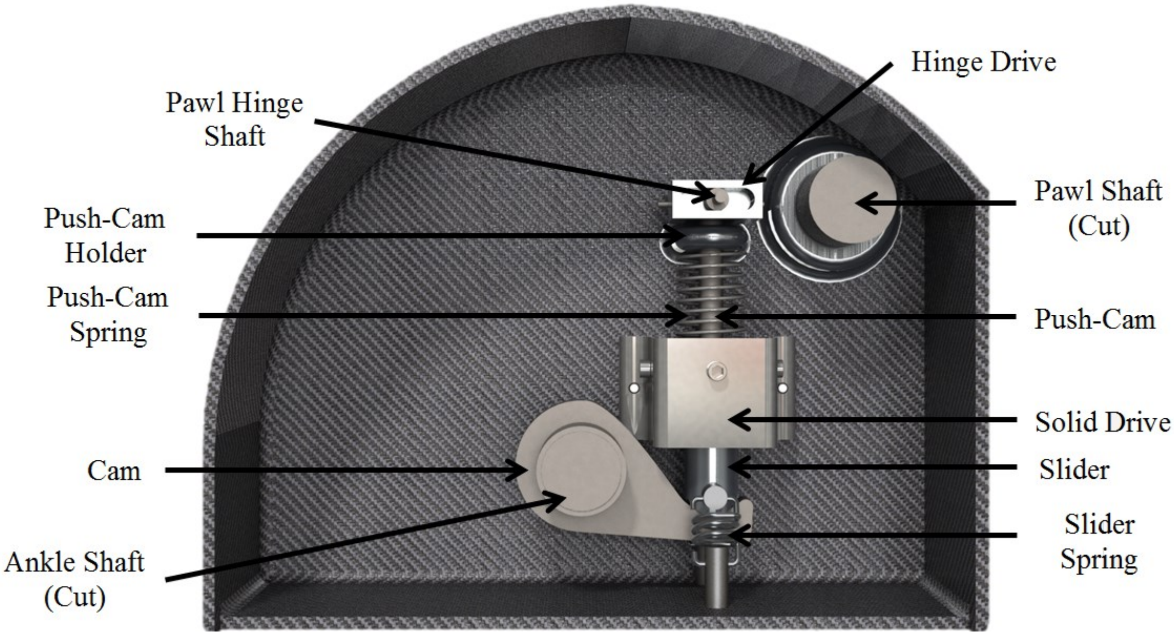


Figure 4-12: Annotated sequencing mechanism

The sequencing mechanism has a cam, a slider, a push-cam, a solid drive and a hinge drive. The cam is attached to the ankle shaft with a shaft pin, so as to use the user's actual motion. When the user reaches 5 to 8 degrees of dorsiflexion, the cam pushes upwards on the slider. Since each individual's gait is slightly different, the 5 to 8 degree range is provided to ensure proper engagement for more individuals. To modify this value, a new cam would have to be fabricated.

The slider, contained in the solid drive, moves vertically and pushes on the push-cam. This component is contained in the solid drive and the push-cam holder, and has a specially designed groove on its surface into which pins slide and guide its vertical and rotating motion. The push-cam, attached to the hinge drive, pushes the hinge shaft vertically. The system has a spring between the push-cam holder and the push-cam pushing the latter downwards, and springs between the slider and the pins at the bottom of the casing to keep the slider and the cam in contact.

The critical part of this mechanism is the groove in the push-cam. This groove dictates the engagement timing of the PAM charging mechanism by allowing the hinge shaft to move along the vertical axis and rotating around this same axis. The geometry of the groove, as shown in Figure 4-13, is repeated four times around the push-cam to promote continual motion.

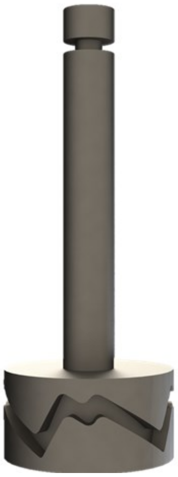
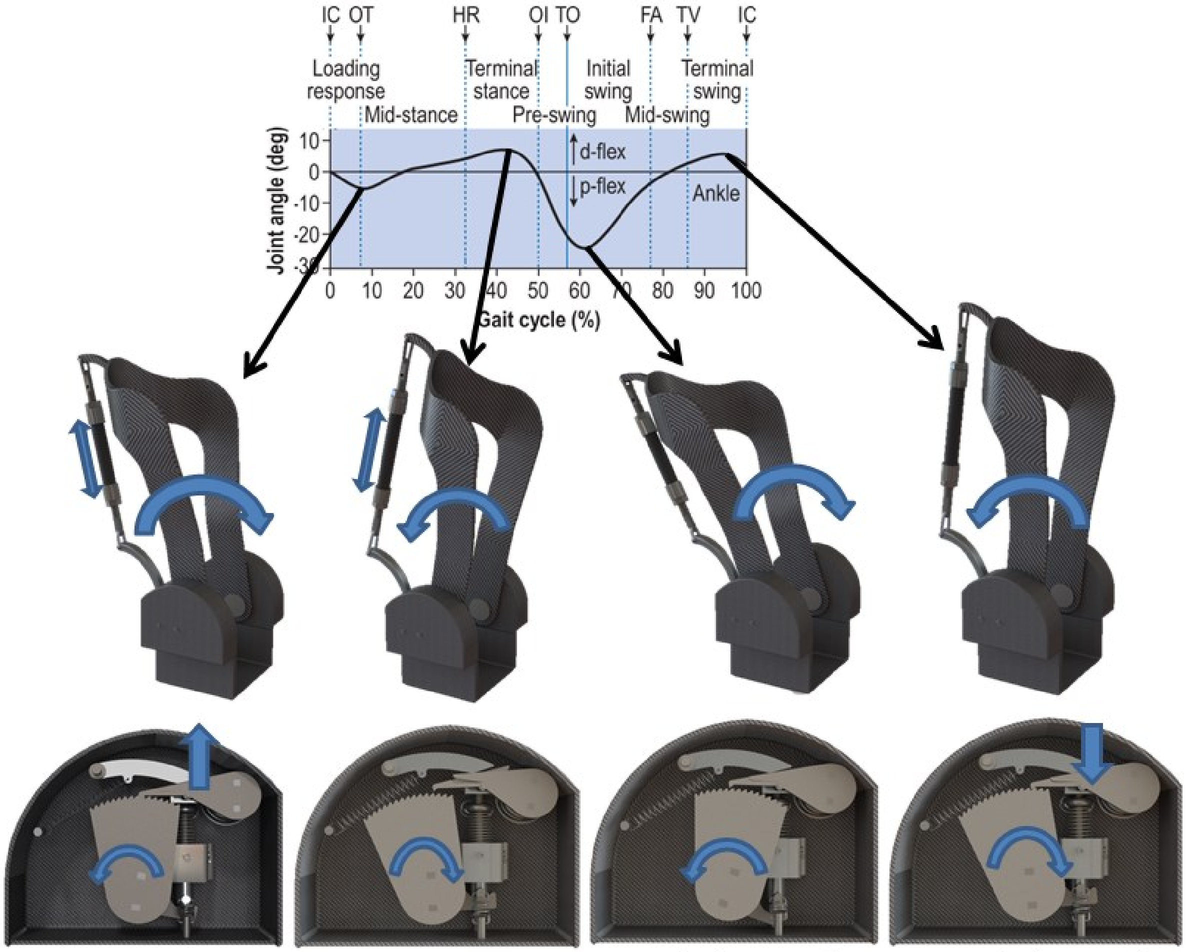


Figure 4-13: Push-cam groove geometry

The operation of the sequencing mechanism over a walking cycle is presented in Figure 4-14. At the Initial Contact, the slider and push-cam are at their lowest positions, the cam is rotated away from the slider and the pawl is engaged with the ratchet. Over the stance phase, the pawl stays engaged with the ratchet, charging the PAM, while the cam rotates and lifts the slider. The slider pushes on the push-cam, which is forced to rotate on is main axis due to the grooves and the solid drive pins. The push-cam nears its maximum vertical position when the user's ankle reaches 5 to 8 degrees of dorsiflexion. At this moment, the force of the push-cam spring overpowers the toggle arm force, making the hinge shaft moves upwards, rapidly disengaging the pawl and ratchet, releasing the charged PAM, and propelling the individual forward at toe-off.

At toe-off, the cam is again disengaged from the slider, which itself is at its lowest position. For its part, the push-cam lowered and rotated slightly, staying high enough to prevent the pawl from engaging to the ratchet during the swing phase, allowing free ankle rotation. Towards the end of the swing phase, the foot dorsiflexes to 5 to 8 degrees. At that point, the cam and slider are back at their highest point, forcing the push-cam to lift and rotate slightly, preparing for the Initial Contact. At Initial Contact, the cam, slider and push cam are all back to their lowest position, and the pawl and ratchet are engaged, allowing the cycle to start over.



Pawl Engaged
PAM Charging

Pawl Disengaged
PAM Releasing

Pawl Disengaged
Free PAM

Pawl Disengaged
Free PAM

Figure 4-14: Timing mechanism cyclic behaviour through a gait cycle

4.2.5 Other Components

The novelty of the exoskeleton is the passive timing mechanism and the use of a newly defined PAM stiffness model. However, to be effective, these must be used in conjunction with a lightweight structure, shafts and accessories, and straps that attach to the user. These are defined briefly to assure completeness.

4.2.5.1Ankle Shaft

The Ankle Shaft is the main connection between the Shank Segment and the Base Segment, and also supports the Ratchet and the Cam. The Shank Segment is mounted on it via splines, while the Ratchet is mounted with rectangular keys, and the Cam, with shaft pins. Capitalizing on the shaft geometry and the Base Segment to limit axial motions, only a retaining ring is used on the Ankle Shaft to prevent the Ratchet from moving axially.

Although common mechanical engineering practice indicates that a shaft should be ideally double supported to limit unwanted motions and/or loading, it was chosen to only support the Ankle shaft only at one section. Ignoring bending for an instant, single supported shafts tend to have some motion perpendicular to the main axis. This feature is meant to give the user more flexibility in terms of ankle motion in the frontal and coronal planes through the angling of the ankle shafts. Whereas long shafts are prone to bending, the proposed design incorporates short shafts to minimize bending. Figure 4-15 present an exploded view of the Ankle Shaft and its accessories.

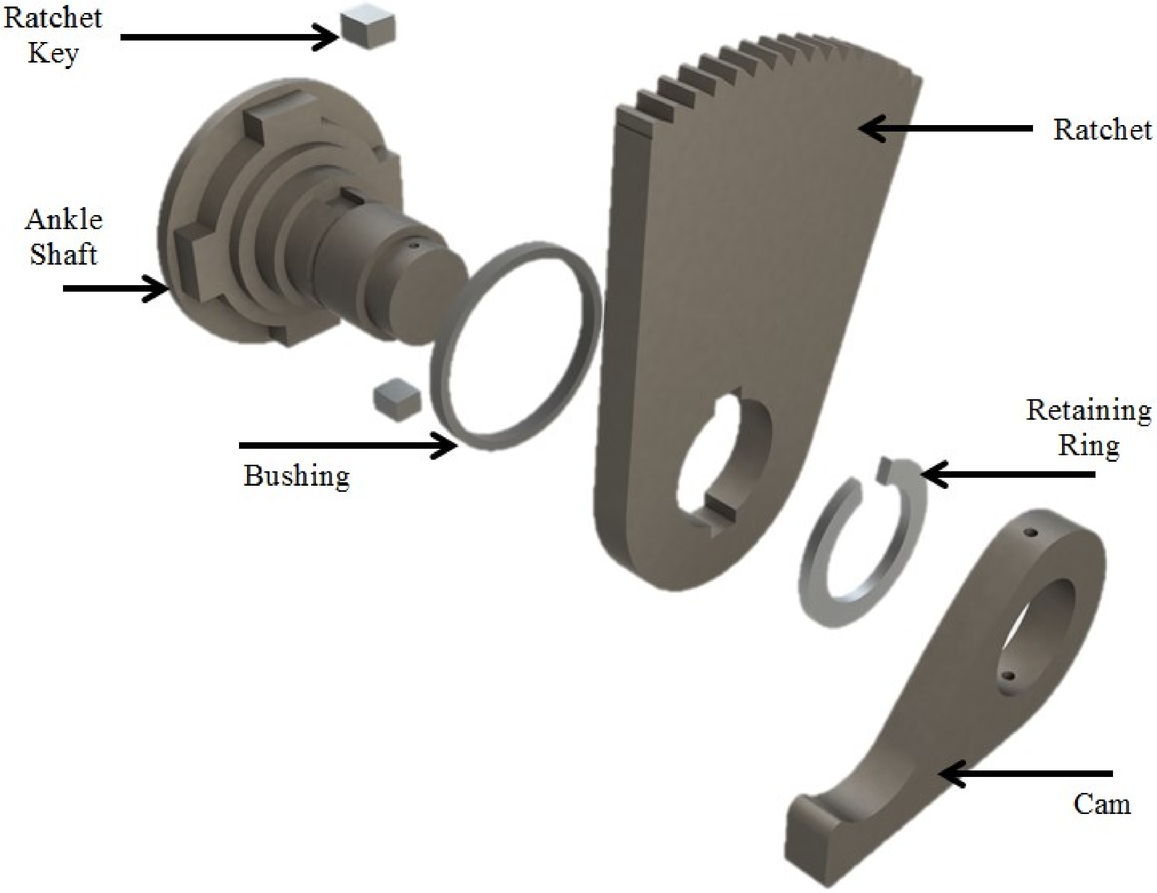


Figure 4-15: Exploded view of ankle shaft and accessories

4.2.5.2Pawl Shaft

The Pawl Shaft, for its part, provides support to the back portion of the Pawl, which is attached to the shaft via rectangular keys and held in place with the shaft geometry. The Pawl Shaft is supported at each end, in the Base Segment and in the Casing. An exploded view of the Pawl Shaft is shown in Figure 4-16.

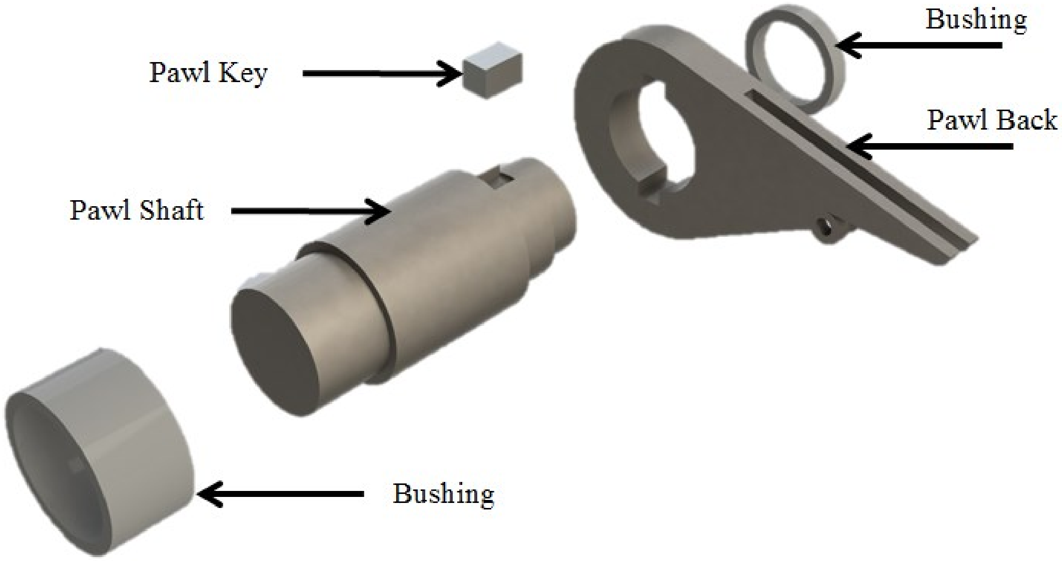


Figure 4-16: Exploded view of pawl shaft and accessories

4.2.5.3Straps

In order to maximize the device's effect, it should be fixed to the user's mid-shank and foot. The best method would be to have the entire structure molded to the intended user's shank and foot. However, the human body tends to change continuously, based on the activity level, diet and other factors, thereby requiring a new molded structure relatively frequently. Instead, the proposed structure, as explained previously, only molds approximately half of the user shank circumference. Therefore, the device relies on a strap at the shank and at the foot to be solidly attached to the user.

The proposed straps are rather simple: thick cushioning material with guides for straps made of Nylon, attached with Velcro. This is very similar to straps used in orthoses and hiking backpacks which have been shown to be effective at load carrying and preventing pressure sores [150]. Figure 4-17 shows an example of an orthosis using common strap with internal padding.

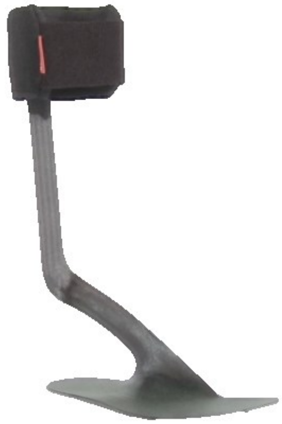


Figure 4-17: Strap example on Ottobock's Walkon orthosis **[151]**

4.3 Detailed Design

To achieve the design objectives, each exoskeleton component must be designed to withstand loads related to the device's everyday use. As such, the exoskeleton dimensions and materials were optimized based on desired specifications, using MATLAB, and modeled in three dimensions using SolidWorks. The following sections describe the modeling and optimization process used to size the different components of the exoskeleton.

4.3.1 Overall Modeling Process

The automated modeling process, programmed in MATLAB, follows the global procedure outlined in Figure 4-18. The preliminary component sizing, based on user inputs and initial material selection, approximates the component volumes, masses and centroids, represented by the "Component Sizing and Material Selection" block. Then the PAM is sized using the algorithm detailed in section 4.2.3 Pneumatic artificial muscle

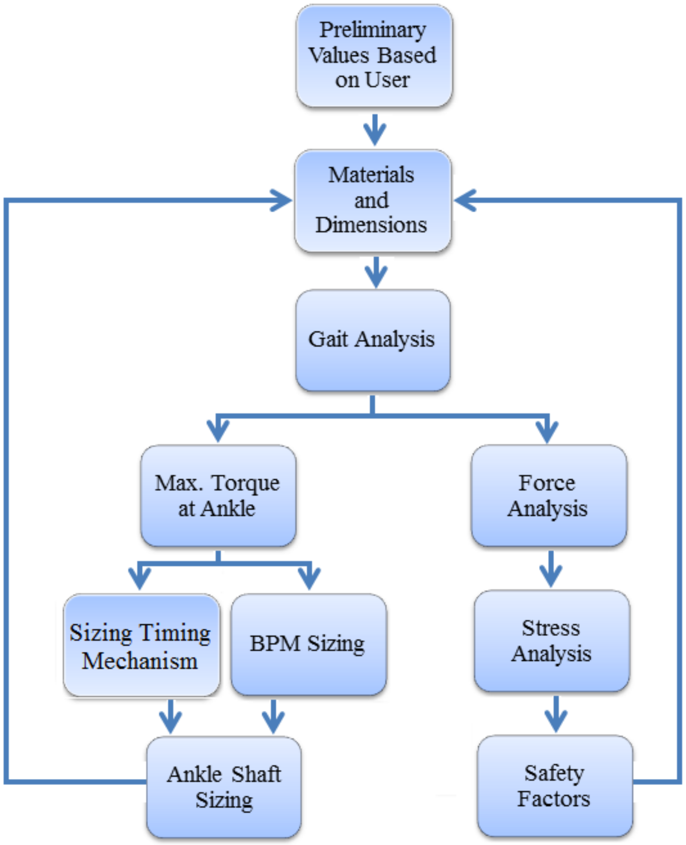


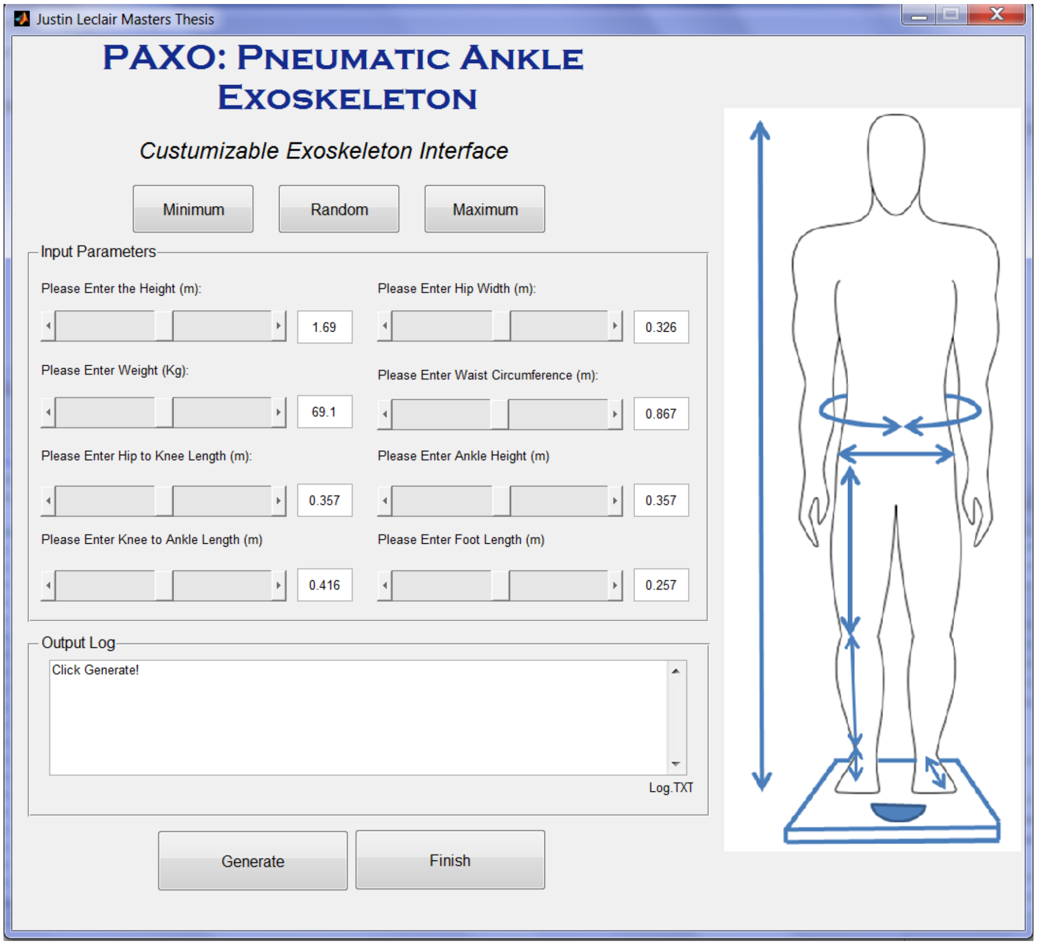
Figure 4-18: Exoskeleton optimizing process flowchart

Combining the PAM forces with kinematic data, the forces and stresses for critical device locations are determined. With these critical forces, mechanical components, like shafts, keys and ratchets, are sized. With the critical stresses, the material and dimensions for each component are found by evaluating the safety factors. This iterative process is repeated until the proper materials and dimensions are found. The MATLAB code is provided in Appendix E.

Once the optimization completed, the final dimensions and materials are transferred from MATLAB to SolidWorks, where a three-dimensional model is sized accordingly. This model can then be used for construction purposes, renderings or motion studies.

4.3.2 User Inputs

As previously stated, the optimization process requires certain user specifications. As part of the MATLAB program, a user interface, shown in Figure 4-19, asks the user to enter his/her height, weight, hip width, waist circumference, hip to knee length, knee to ankle length, ankle height and foot length. These specifications represent the first step of the optimization process presented in Figure 4-18.



PAXO: PNEUMATIC ANKLE EXOSKELETON

Custumizable Exoskeleton Interface

Input Parameters

Please Enter the Height (m): 1.69

Please Enter Hip Width (m): 0.326

Please Enter Weight (Kg): 69.1

Please Enter Waist Circumference (m): 0.867

Please Enter Hip to Knee Length (m): 0.357

Please Enter Ankle Height (m) 0.357

Please Enter Knee to Ankle Length (m) 0.416

Please Enter Foot Length (m) 0.257

Output Log

```
Click Generate!
```

Log.TXT

Figure 4-19: GUI for MATLAB optimization program

As stated in the Chapter 2, the potential user's range of specification spans from the 5th percentile female ground troops to the 95th percentile male ground troops.

4.3.3 Analyses

The optimization process contains three major analyses. The analyses performed are the kinematic analysis, the force analysis and the stress analysis. The kinematic, force and stress analyses will be presented herein.

4.3.3.1Kinematic Analysis

The objective of the kinematic analysis is to determine the position, velocity and acceleration of all exoskeleton segments during common movements. Recognizing that each individual has slight variations in the way they walk, gait graphs indicate normalized walking pattern by controlling for height, weight and walking speed. Therefore, taking a dataset containing this normalized data and applying the user specifications to it can give an estimation of the user kinematic for various motions.

Since the introduction of gait graphs, normalized gait datasets have usually been tallied in biomechanical laboratories by averaging the gait pattern for large number of individuals. Due to time limitations, a normalized dataset was extracted from the Gait Profile Score (GPS) [152], [153]. This score provides a functional comparison between an individual's gait pattern, captured in a biomechanical laboratory, and a normalized gait dataset of joint angle for the foot, ankle, knee, hip and pelvis. Since such datasets are only available for walking, only this motion is considered for the analysis.

Although the GPS considers motion in the three anatomical planes, walking can be approximated to be entirely in the sagittal plane. Therefore, only the hip flexion and extension, the knee flexion and extension, and the ankle dorsiflexion and plantarflexion in terms of gait percentage (*%Gait*) are extracted from the GPS dataset. A table containing the normalized dataset is presented in Appendix D.

Knowing the joint angles in terms of gait progress, it is possible to determine the angular velocity and acceleration, by estimating a relative time interval corresponding to a gait percentage point from the walking velocity and the stride length. Empirical relationship, linking an individual's height to the ideal walking velocity and to stride length, have been established with moderate levels of accuracy [154], as shown below.

$$\text{Stride Length} \approx 0.818 * \text{Height} \tag{4-1}$$

$$\text{Walking Velocity} \approx 0.820 * \text{Height} \tag{4-2}$$

Then, the time interval for each gait percentage point can be found as follows:

$$\Delta t = (\%\text{Gait})\frac{\text{Stride Length}}{\text{Walking Velocity}} \approx (\%\text{Gait})\frac{0.818\text{Height}}{0.820\text{Height}} \approx \%\text{Gait} \tag{4-3}$$

Recognizing the dataset contains only of joint angles in the sagittal plane, the angular velocities and the angular accelerations in that same plane can be evaluated.

$$\omega_i = \frac{\theta_i - \theta_{i-1}}{\Delta t} \tag{4-4}$$

$$\alpha_i = \frac{\omega_i - \omega_{i-1}}{\Delta t} \tag{4-5}$$

These definitions of angular velocity and acceleration are only valid for planar motion, but the remainder of the analysis is done in three dimensions. Considering only sagittal plane joint angle were extracted from the GPS dataset, the angular velocity and angular accelerations can be rewritten in three dimensions as:

$$\vec{\omega} = \omega\vec{\imath} + 0\vec{\jmath} + 0\vec{k} \tag{4-6}$$

$$\vec{\alpha} = \alpha\vec{\imath} + 0\vec{\jmath} + 0\vec{k} \tag{4-7}$$

Then, the position, velocity, and acceleration of any point on the legs throughout the gait cycle can be defined using the joint angles, angular velocity, and angular acceleration in combination with the user and device geometry. To simplify the system, the torso is assumed to always be vertical and moving at a constant velocity, causing the hips to have a null acceleration, relative to the torso.

$$\vec{a}_{hip} = 0\vec{\imath} + 0\vec{\jmath} + 0\vec{k} \tag{4-8}$$

$$\vec{a}_{knee} = \vec{a}_{hip} + \vec{a}_{knee/hip} \tag{4-9}$$

$$\vec{a}_{ankle} = \vec{a}_{knee} + \vec{a}_{ankle/knee} \tag{4-10}$$

$$\vec{a}_{foot} = \vec{a}_{ankle} + \vec{a}_{foot/ankle} \tag{4-11}$$

To simplify the overall analysis, every human body segment and exoskeleton component was assumed to act as a rigid body, where the entire weight for each component was applied at its center of mass, a common practice in biomechanical analyses. The mass, center of mass and radius of gyration for the human body segment were obtained from published tables [20], and are shown in Table 4-3.

Table 4-3: Human Body Segment Properties **[20]**

Segment	Mass (kg)	Proximal COM / Segment Length	Radius of Gyration/ Segment Length
HAT (Head, Arms, Torso)	0.678 BW	0.626	0.503
Thigh	0.1 BW	0.433	0.323
Lower Leg	0.0465 BW	0.433	0.302
Foot (Height)	0.0145 BW	0.50	0.475
Foot (Length)		0.50	

For the exoskeleton components, the mass was calculated by determining the volume of each piece and assuming a uniform material density

$$Volume = \sum V_i \tag{4-12}$$

$$Mass = Volume * Density \tag{4-13}$$

Then, the three dimensional center of mass was evaluated with the following relationships:

$$x = \frac{\sum x_i V_i}{\sum V_i}, y = \frac{\sum y_i V_i}{\sum V_i}, z = \frac{\sum z_i V_i}{\sum V_i} \tag{4-14}$$

Finally, each component's inertia was determined with the parallel axis theorem, assuming that each component is a point mass acting at the center of mass

$$I_{xx} = \rho \sum V_i r\text{-}_{xxi}{}^2 \tag{4-15}$$

$$I_{yy} = \rho \sum V_i r\text{-}_{yyi}{}^2 \tag{4-16}$$

$$I_{zz} = \rho \sum V_i r\text{-}_{zzi}{}^2 \tag{4-17}$$

$$I_{xy} = \rho \sum V_i (x - x\text{-}_i)(y - y\text{-}_i) \tag{4-18}$$

$$I_{xz} = \rho \sum V_i (x - x\text{-}_i)(z - z\text{-}_i) \tag{4-19}$$

$$I_{yz} = \rho \sum V_i (y - y_i)(z - z\text{-}_i) \tag{4-20}$$

$$r\text{-}_{xxi} = \sqrt{(y - y_i)^2 + (z - z\text{-}_i)^2} \tag{4-21}$$

$$r\text{-}_{yyi} = \sqrt{(x - x_i)^2 + (z - z\text{-}_i)^2} \tag{4-22}$$

$$r\text{-}_{zzi} = \sqrt{(x - x_i)^2 + (y - y\text{-}_i)^2} \tag{4-23}$$

4.3.3.2General Force Analysis

Using generalized equations, applicable to every configuration, the dynamic force balance and the dynamic moment balance can be evaluated:

$$\sum \vec{F} = \sum m\vec{a}$$

$$\sum \vec{M} = I \times \vec{\alpha} + \sum \vec{r} \times m\vec{a}$$

The three-dimensional inertia vector, represented by $I \times \vec{\alpha}$, for each limb and device segment is determined from the definition of inertia in three dimensions, combining the different results from the kinematic analysis:

$$\begin{aligned}(I \times \alpha)_x = {} & I_{xx}\alpha_x - \left(I_{yy} - I_{zz}\right)\omega_y\omega_z - I_{xy}\left(\alpha_y - \omega_z\omega_x\right) - I_{yz}\left({\omega_y}^2 - {\omega_z}^2\right) \\ & - I_{xz}\left(\alpha_z + \omega_x\omega_y\right)\end{aligned} \tag{4-24}$$

$$\begin{aligned}(I \times \alpha)_y = {} & I_{yy}\alpha_y - (I_{zz} - I_{xx})\omega_z\omega_x - I_{yz}\big(\alpha_z - \omega_x\omega_{xy}\big) - I_{xz}({\omega_z}^2 - {\omega_x}^2) \\ & - \; I_{xy}(\alpha_x + \omega_y\omega_z)\end{aligned} \tag{4-25}$$

$$\begin{aligned}(I \times \alpha)_z = {} & I_{zz}\alpha_z - \big(I_{xx} - I_{yy}\big)\omega_x\omega_y - I_{xz}\big(\alpha_x - \omega_y\omega_z\big) - I_{xy}\big({\omega_x}^2 - {\omega_y}^2\big) \\ & - \; I_{yz}(\alpha_y + \omega_z\omega_z)\end{aligned} \tag{4-26}$$

The proposed dynamic force and moment analysis results in an unsolvable system consisting of 6 equations and 20 unknowns, the latter of which are shown below:

- Ground force reaction magnitude, position and direction under left and right foot
- Ground force reaction magnitude, position and direction under left and right device
- Interactions between device and user on shank and on foot for each leg

Therefore, simplifications are required before results can be obtained. First, the analysis is done for only one leg and the results are assumed to be identical for the other leg but shifted by half a gait cycle. Then, the interactions between the device and the user are assumed to be exclusively caused by the pressure applied by the straps, making this interaction independent of the user’s motion. Furthermore, since the device has a plate going underneath the user’s foot and is solidly attached to the user, the ground force reactions for the foot and device are assumed to be identical. Lastly, the PAM force is estimated by assuming an ankle moment corresponding to the average stance phase ankle stiffness, as shown in Figure 2-11.

Applying these simplifications, the system is simplified to 6 unknowns, which are the ground force reaction magnitude, position and direction for each foot. Therefore, the system becomes solvable and a segment-by-segment force and stress analysis can be performed.

4.3.3.3General Stress Analysis

Having considered the general force analysis, a general stress analysis is also presented to allow the automation of the analysis into MATLAB. The state of stress at critical location in each component must be evaluated in order to properly size and define the right material for each component. By analyzing the internal stresses at various cross sections and determining the resulting safety factors, it is possible to determine whether the material and dimensions of the components are adequate and optimize the design of each component.

First, critical locations on each component are identified. These must present possible high stress concentrations, such as corners, or where the large external forces are present.

Cross-sectional analysis is done at these locations. The shear forces and moments acting on the cross-section are obtained through free body diagram analysis. The majority of the cross-sections present faces that are circular or rectangular, which simplifies the analysis significantly. For circular cross-sections, the points of interest correspond to the location maximum bending and shear stresses occur, which is at the top, on the side and at a 45° angle. For rectangular sections, the points of interests are similarly chosen. The points of interest for both shapes are shown in Figure 4-20. Note that the point of maximum stress will, in most cases, be on one of the indicated points, but may be at a different location is certain situations.

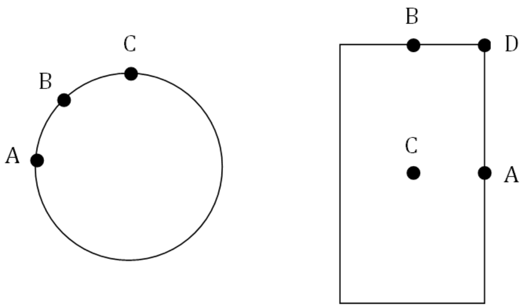


Figure 4-20: Points of interest on circular and rectangular cross-sections

The state of stress at each of these points is found using the distortion energy theorem, which is a conservative theorem. The general formula for the state of normal stresses is:

$$\sigma_z = \frac{-M_x y}{I_x} + \frac{-M_y x}{I_y} + \frac{P}{A} \qquad \textbf{4-27}$$

Where the z axis is normal to the face of the section. Due to gravity and the weight of the proposed exoskeleton, it is assumed that the stress in compression is always greater than that in tension. Thusly, the maximum stress in compression is determined. The yield strengths of the materials considered are assumed to be equal in tension and compression, which differs from actual material behaviour but must be assumed to use the distortion energy theorem. Continuing the previously established coordinate system, the transverse shear stress can be found as follows:

$$\tau_{zx} = \frac{M_z c}{J} \qquad \textbf{4-28}$$

$$\tau_{xy} = \frac{V_y Q_y}{I_X b_x} \qquad \textbf{4-29}$$

$$\tau_{zy} = \frac{V_x Q_x}{I_y b_y} \qquad \textbf{4-30}$$

Where the distance *c* is either the radius of the circle or half the width of the rectangle; *J* is the polar moment of inertia; *Q* is the first moment of inertia; *b* is the thickness of the section at the desired point in the direction perpendicular to the applied transverse force *V*.

The variable *Q* is identical for both *A* and *C* on the circular cross section but is a bit more complicated for *B* since it is located at a 45° angle. Both formulas are shown below:

$$Q_A = \ Q_C = \frac{4\left(\frac{Do}{2}\right)}{3\pi}\left(\frac{\pi}{8}(Do^2 - Di^2)\right) \qquad \textbf{4-31}$$

$$Q_B = \left(Do * sin\left(\frac{\pi}{4}\right) + \left[\frac{4\left(\frac{Do}{2}\right) sin^3\left(\frac{\pi}{4}\right)}{3(\frac{\pi}{2} - sin\frac{\pi}{2})}\right]\right)\left(\left(\frac{Do}{2}\right)^2 cos\left(\frac{\frac{Do}{2} - h}{\frac{Do}{2}}\right) - \left(\frac{Do}{2} - h\right)\sqrt{\frac{2Do}{2}h - h^2}\right) \qquad \textbf{4-32}$$

$$\text{Where } h = Do - Dosin\frac{\pi}{4}$$

Rectangular cross-sections have much simpler analysis, and can be determined with the basic formulas shown herein:

$$Ix = \frac{bh^3}{12} \qquad \textbf{4-33}$$

$$Iy = \frac{hb^3}{12} \qquad \textbf{4-34}$$

$$A = bh \qquad \textbf{4-35}$$

The overall stress is calculated by using the Von Mises stress (σ') formula:

$$\sigma' = \frac{1}{\sqrt{2}}\sqrt{2{\sigma_z}^2 + 6({\tau_{zx}}^2 + {\tau_{xy}}^2 + {\tau_{zy}}^2)} \qquad \textbf{4-36}$$

Then, the safety factor is calculated by comparing the Von Mises stress to the yield strength (S_y) of a proposed material.

$$n = \frac{S_y}{\sigma'} \qquad \textbf{4-37}$$

To be a suitable solution, the safety factors for every point had to be at least greater than 1.5 to account for manufacturing defects, material defects and any other problem that may arise. This analysis is easily programmable into MATLAB and allows the iterating of dimensions and material selections.

4.3.3.4Strap Analysis

One of the underlying assumptions made previously was that the forces between the device and the user are entirely caused by the pressure applied in the straps. Although it would be possible to obtain the strap forces on the shank from the ankle moment during the stance phase, it is not determinable during the swing phase as the ankle has no moment. Therefore, a new method, based on the weight support capacity for backpacks [150], is used to determine the interaction forces between the user and the device through the gait cycle, on the shank and on the foot.

The analysis uses the thin-walled pressure vessel theory to determine the forces based on standard belt sizes. The maximum internal pressure has to be smaller than the diastolic pressure in order to allow free blood circulation. The American Heart Association lists normal diastolic blood pressures as approximately 8000 to 10 600 Pascal [155]. For this model, the strap pressure is assumed to be 50% of the maximum diastolic blood pressure.

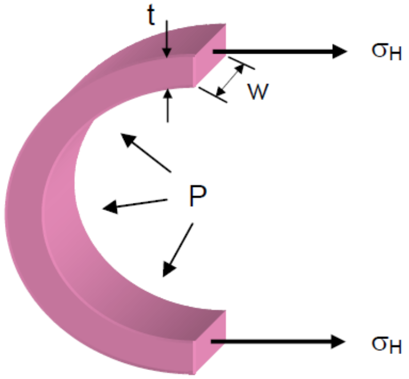


Figure 4-21: Straps forces using thin-walled pressure vessel theory **[150]**

Considering that the hoop stresses can be modeled in terms of internal pressure and in terms of tension in the walls, a relationship between the strap tension, the internal pressure and the strap geometry is obtained.

$$\sigma_h = \frac{T}{tw} = \frac{Pr}{t} \tag{4-38}$$

$$T = Prw \tag{4-39}$$

Where *T* is the belt tension, *σ_H* is the hoop stress, *t* is the thickness, *w* is the height, *P* is the internal pressure, and *r* is the radius. Then, observing that the tension in the strap presses the user against the device, creating a normal force as follows:

$$F_y = 2\int_0^{\frac{\pi}{2}} Pwr\sin\theta\, d\theta = 2Pwr \tag{4-40}$$

$$F_x = \int_0^{\pi} Pwr\cos\theta\, d\theta = 0 \tag{4-41}$$

The only remaining force caused by the strap is the vertical force. Using the backpack analysis, it has been shown that a typical strap on the waist with a tension of 100 N can support about 20 kg. Scaling it to the shank geometry, the vertical force is approximated as:

$$F_z = \left(\frac{20kg * 9.81}{100N}\right)(Prw) \tag{4-42}$$

4.3.3.5PAM and Ratchet Design

The entire timing mechanism relies on the user's movement to move the ratchet and to create effective PAM elongation. Both the ratchet design and the PAM design rely on the maximum achievable ankle moment during stance phase. Only the ankle moment during the stance phase is considered because it corresponds to the instances where the timing mechanism is locked and the PAM is charging.

As shown in section 2.1.5, the maximum moment at the ankle during stance phase is about 1.461 Nm/kg. Since the device's objective is to supplement the ankle effort by 25%, and not replace it completely, the ankle moment used for the design of the PAM and of the ratchet is:

$$M_{ankle\ effort} = 25\%(1.461) = 0.365\ Nm/kg \tag{4-43}$$

Then, the ratchet and pawl can be fully designed using the methodology laid out in [149], and optimized for a small face width and a ratchet diameter smaller than the ankle height.

For the PAM, an algorithm had to be created to size the PAM to create the ideal ankle moment, all the while respecting the available space based on the device geometry, and the change in length based on the user motion. This algorithm takes into account the PAM pressure, PAM diameter, PAM materials, and the user's height and weight. A flowchart detailing the PAM sizing algorithm is displayed in Figure 4-22.

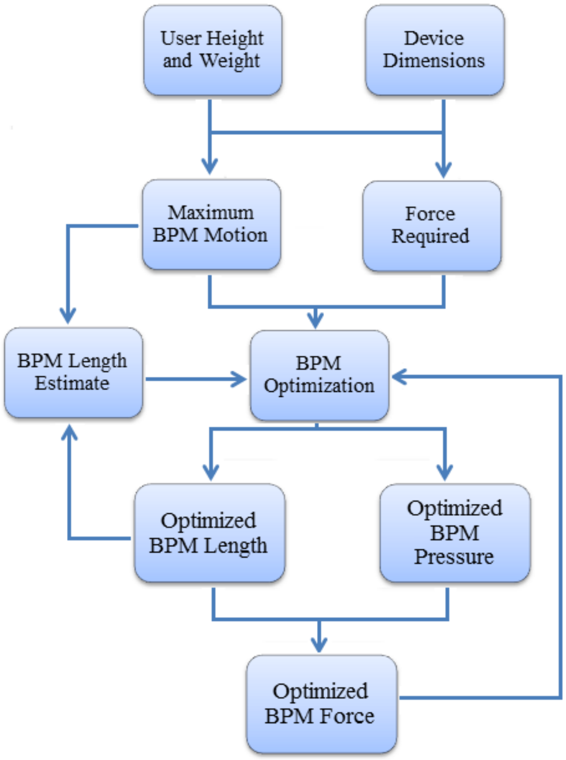


Figure 4-22: PAM design algorithm logic flow

Using the optimizations method, the exoskeleton moment contribution over the gait cycle can be calculated for each point through, as shown in Figure 4-23. This graph clearly shows that the effect of the exoskeleton is intended to be about a quarter of the normal ankle moment, as outlined in the initial proposal of the design proposal. It also suggests that the PAM will quickly release its energy, completing its energy transfer immediately after the push off is completed.

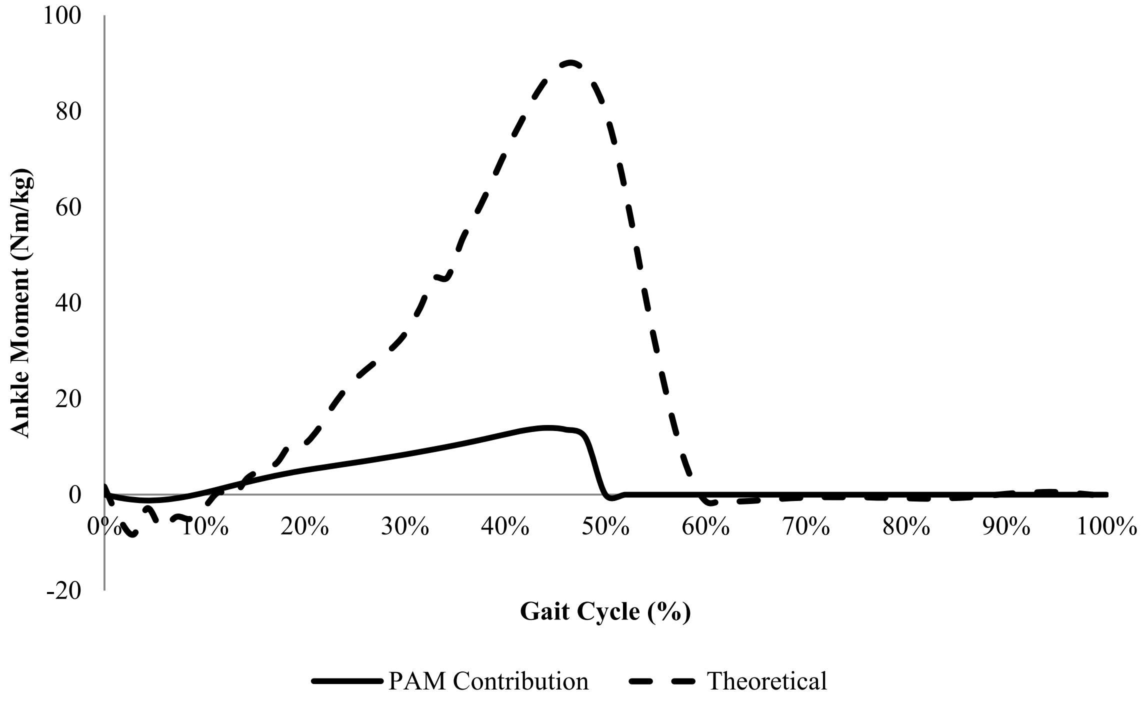


Figure 4-23: Exoskeleton moment contribution compared to ankle moment

4.4 Conclusions

The design of the unpowered ankle exoskeleton was done in three steps: the initial proposal, the conceptual design and the detailed design. The initial proposal outlined the design specification and selected some solutions using a weight score approach. The conceptual design outlined the design of the exoskeleton and provided information with regards to how each component behaved and interacted. The detailed design analyzed each component, one at a time, to determine their dimensions and material. This last process was automated using MATLAB. This three-dimensional model was proposed and parameterized to fit a large segment of the population using SolidWorks and MATLAB interfaces.

The main challenges of this chapter had to do with the timing mechanism. This passive mechanism was designed, composed of ratchet, pawl, cams and followers to sequentially charge and release a PAM located at the back of the ankle. With the design completed, what remains is to build and test a prototype.

Chapter 5– EXOSKELETON FABRICATION AND TESTING

The objective of the construction and testing is to show the feasibility of the timing and sequencing mechanism to lock and unlock the PAM through the gait cycle, at the right instances, such as to capture and the release energy through the gait cycle.

This chapter presents the fabrication, testing and mechanical validation of an unpowered ankle exoskeleton prototype designed in the previous chapter, as well as the fabrication of the mechanical jigs required to test the unpowered ankle exoskeleton prototype.

.

5.1 Prototype Fabrication

Construction of the exoskeleton was done once the final model had been developed. Prior to the construction, several components were developed using cardboard and 3D printers, to evaluate the effectiveness and potential challenges of certain geometries. These rapid manufacturing processes lead the iterative design process towards simpler solutions, and avoided the need for machining multiple prototypes for testing.

The proposed design presented several manufacturing challenges in terms of time required, cost, and materials. The design was modified to be made entirely of steel and aluminum, which are easier to machine, are more readily available, and fall within the machining abilities of the machinist at the University of Ottawa Machine Shop.

As such, the modifications consisted in changing the structure components, namely the shank segment and the base segment. The shank segment was split into four pieces: two vertical bars, a C-shape and a triangular anchor. The vertical bars were lengthened slightly to accommodate the testing apparatus that will be detailed later.

The base segment was split into five pieces: a thin flat plate for the bottom, machined thick plates to form the left and right pieces, a thick plate for the back and a triangular anchor. For both the shank segment and base segment, the pieces were bolted together to form the overall structure. Furthermore, this type of construction made it easier to assemble and disassemble the prototype and to troubleshoot it during the construction and the testing phases. The prototype used for testing is shown in Figure 5-1.

In terms of dimensions, the prototype was built to fit an individual 183 cm in height and weighing 80 kg. The prototype dimensions were roughly 40 cm tall and 10 cm wide. The final prototype weighed in at 1.35 kg, with an inflated bladder, thereby being under the target weight. This weight is considerably lower than the maximum weight outlined in the specifications, which suggests that a lightweight molded structure might be even lighter and cause very minimal inertial impact.

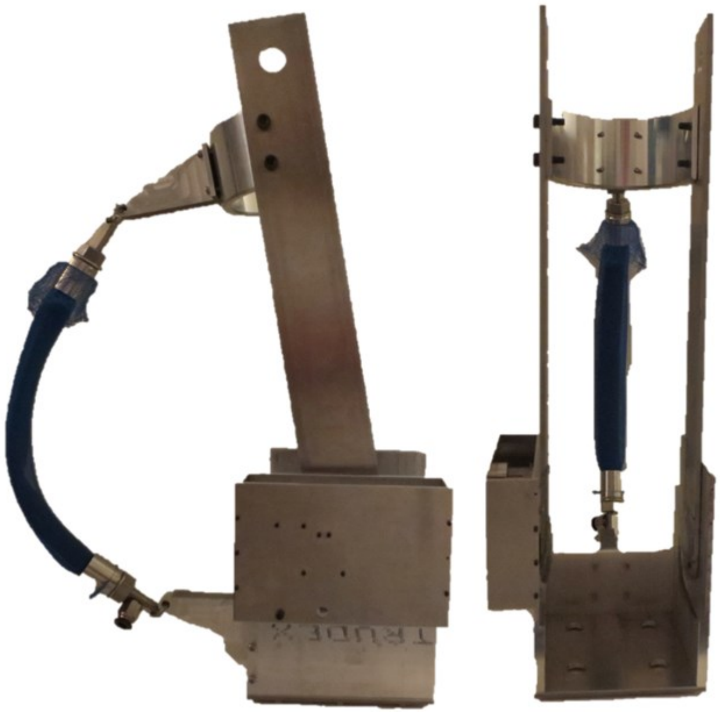


Figure 5-1: Prototype side and front view with deflated PAM

Most of the parts were initialy roughly cut using common saws, and then machined to precision using combinations of Wire EDM, lathes, and drill presses. Some components, especially the ones in the timing mechanism due to their small size and significant complexity, had to be machined using CNCs and fourth axis lathes, which was initially beyond the capability of the University of Ottawa Machine Shop and took more time than initially planned.

The PAMs end fixtures had been built for a previous project [120] and were readily available with minor modifications. A selection of muscles was built based on the specifications proposed by the parameterized algorithm detailed in Chapter 4. According to the algorithm, for a user of 183 cm in height and 80 kg in weight, the PAM using a butyl rubber bladder and a Neon-Blue Overexpanded fiber mesh, required a pressure of 206 kPa, a diameter of 1.3 cm and a deflated length of 19.2 cm, resulting in a stiffness varying between 3.5 and 6.5 kN/m.

5.2 Experimental Set-up and Jig Fabrication

In order to show the feasibility of the prototype, it has to be tested. Since it is not safe or practical to test a novel exoskeleton on human without prior testing, the prototype was tested manually and mechanically. The experimental set-up used for the mechanical testing is shown in Figure 5-2 and Figure 5-3.

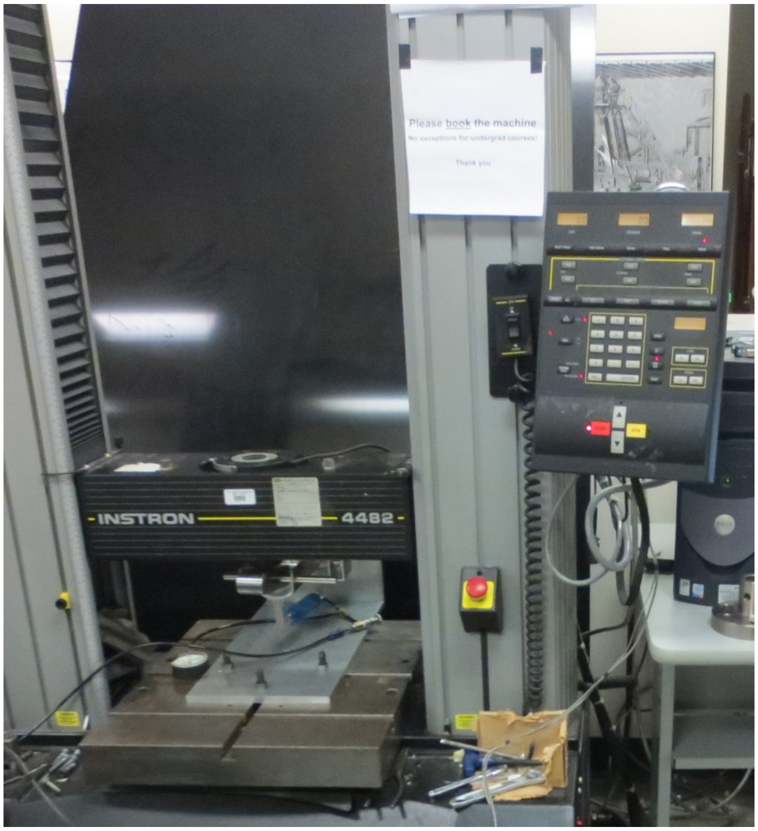


Figure 5-2: Unpowered Ankle exoskeleton prototype feasibility experimental set-up. Jig is attached to to the Instron machine with 3 large bolts and the prototype is attached to the jig via large bolts

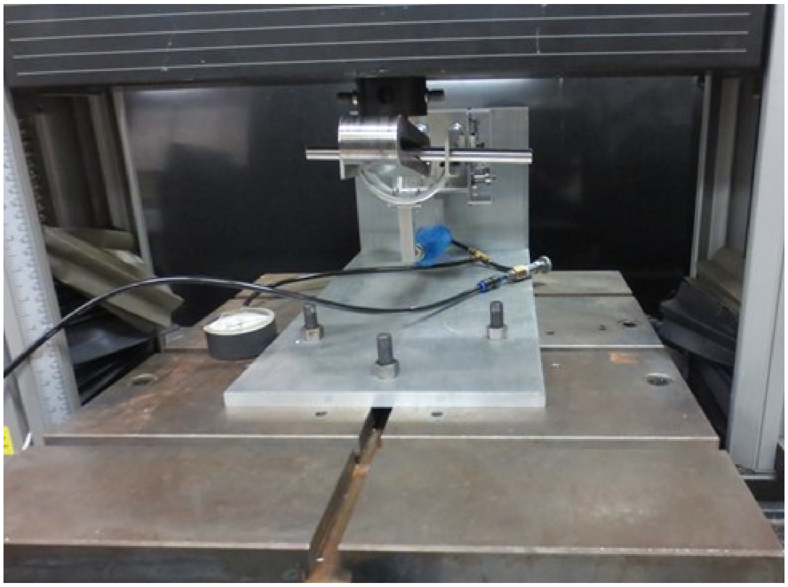


Figure 5-3: Unpowered ankle exoskeleton feasibility experimental set-up close-up. Jig is attached to the Instron machine via larg bolts, held solidly in place with nuts. Prototype is attached to the jig with bolts and to the Instron via tempered steel bar.

This type of machine, most recognizably used to test material behaviour, is able to move along a vertical axis only. This vertical motion could replace the ankle rotation by solidly attaching the prototype's ankle shaft far from the vertical axis of motion and the guiding shaft to slide freely horizontally. In other words, when the ankle rotates around a fixed axis, the shank, which has a fixed length, rotates around the ankle. An intermediate piece, called the Horizontal motion attachment, transforms that rotating motion into a horizontal sliding motion and a vertical motion aligned with the machine's main axis.

Figure 5-4 illustrates the Horizontal motion attachement attached to the Instron and to the exoskeleton prototype. It also illustrates the horizontal position in which the exoskeleton is positioned.

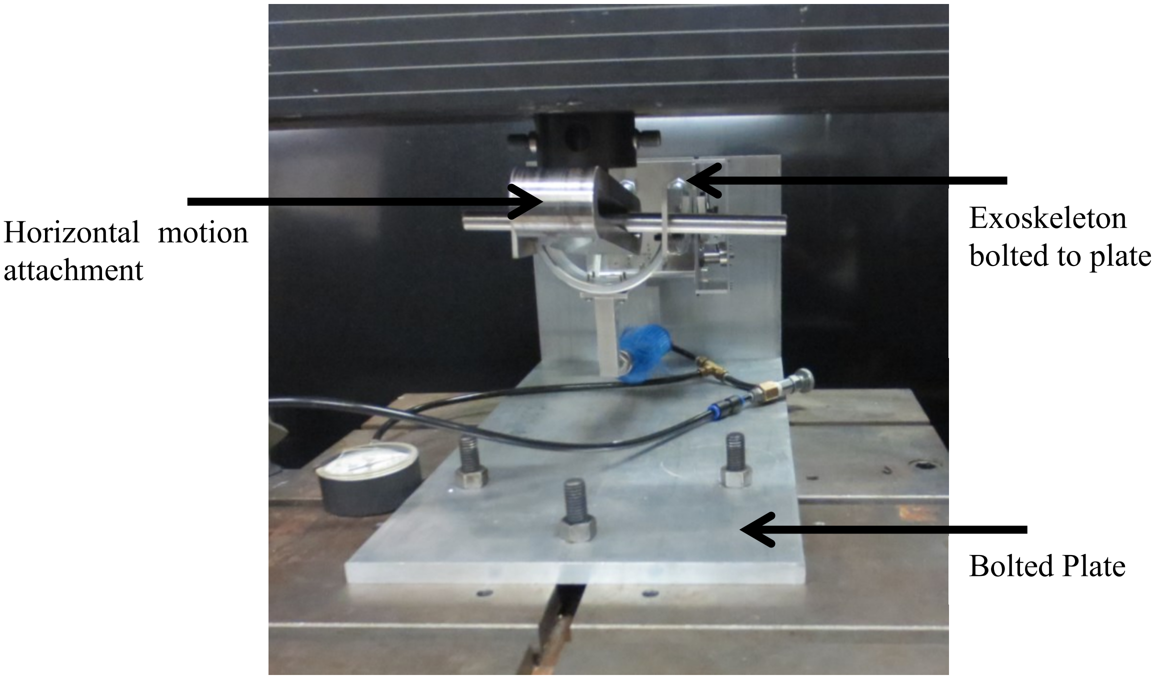


Figure 5-4: Exoskeleton feasibility experimental set-up – angle view

A simplified schematic of this configuration is shown in Figure 5-5, where the angle θ is the angle of rotation of the Shank segment, which corresponds to the ankle angle in the gait cycle. L is the distance between the rotation axis and the contact point of the tensile testing machine, and y is the vertical displacement of the contact point caused by the rotation angle. The other dimensions indicated are relevant to the location of the PAM attachment point.

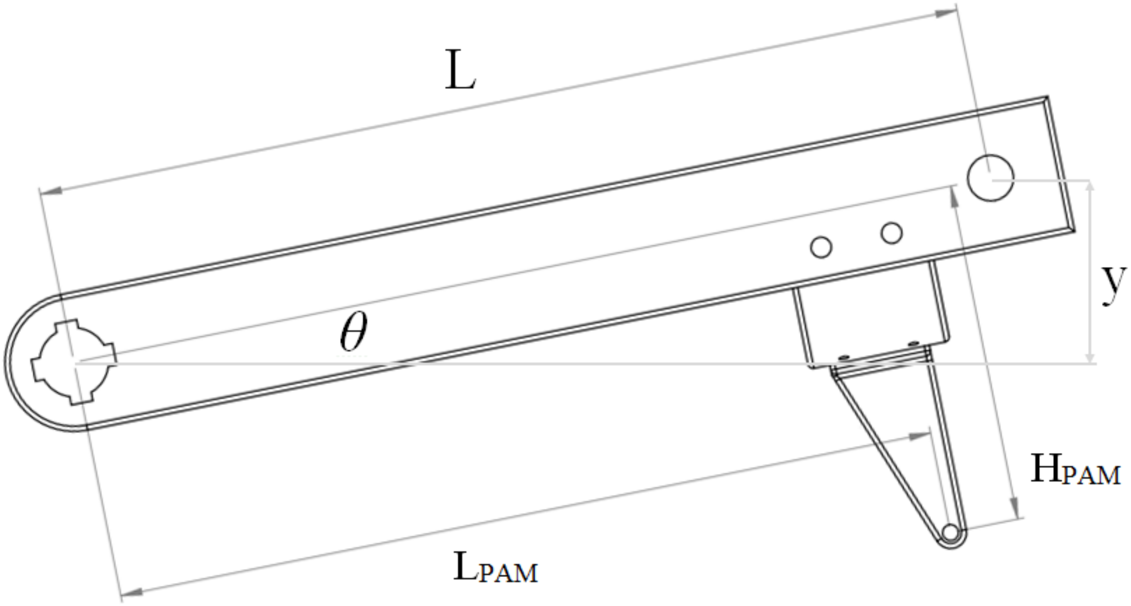


Figure 5-5: Simplified schematic of the exoskeleton on the tensile testing machine

With reference to Figure 5-5, this simple geometry shows that, knowing the values of L and θ, it is possible to determine the vertical displacement y using the formula (5-1).

$$y = L\sin\theta \tag{5-1}$$

The value of L is known from the prototype's geometry and a series of values of ankle angles θ are known for the entire gait cycle, as shown in Appendix D. Figure 5-6 indicates a top view of the prototype affixed to the Instron and to the Bolted Plate.

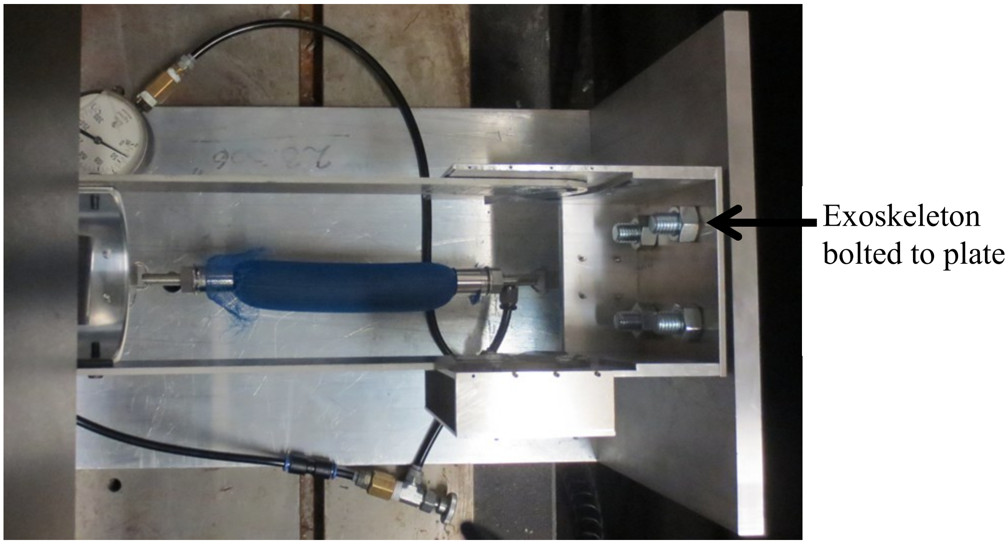

Exoskeleton bolted to plate

Figure 5-6: Exoskeleton feasibility experimental set-up – top view

5.2.1 Bolted Plate

The bolted plate used was made of an aluminum plate to provide solid anchor points to the prototype and to the tensile testing machine. This component consists of two pieces that are bolted together to form an L piece, as shown in Figure 5-7.

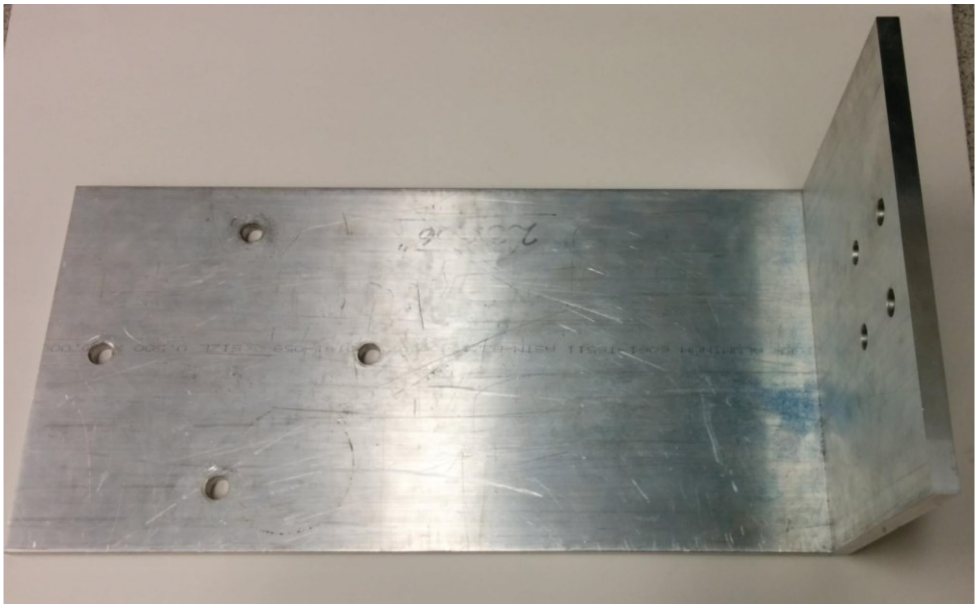


Figure 5-7: Bolted plate configuration

5.2.2 Horizontal Motion Attachment

The Horizontal Motion Attachment component was made of steel. A cylindrical piece with holes to attach to the tensile testing machine was welded to an ellipsoid component that allows the shank to move horizontal while the tensile testing machine moves vertically.

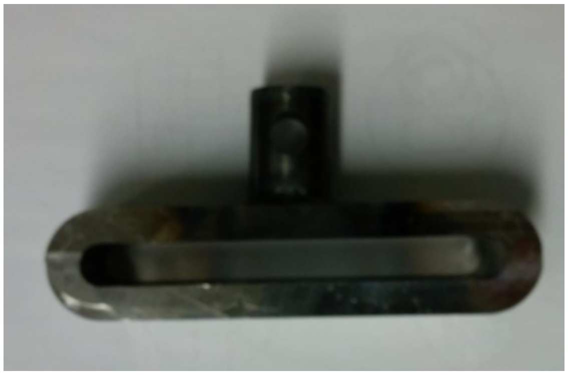


Figure 5-8: Horizontal motion attachment configuration

5.2.3 PAM

The PAMs were selected and fabricated to give the ideal ankle moment at the toe off while allowing the desired motion, based on the PAM optimization algorithm shown previously, in Figure 4-18. A comparison of the theoretical and experimental values for the

PAM properties is shown in Table 5-1. According to the algorithm, the resulting PAM is meant to have very precise initial pressure and muscle length. As the exact theoretical were near impossible to obtain, rounding of the values was necessary. Further, rounding was needed considering that wound steel wire was used to attach the PAM's end-caps to the exoskeleton's muscle attachments, for their simplicity, availability and ease of installation. The goal was to make the attachment distance as small as possible, while allowing some freedom of rotation.

Table 5-1: Selected PAM properties

Properties	Theoretical Values	Experimental Values
Length (inflated)	12.048 cm	12.1
Initial Pressure	206.84 kPa (30.00 psi)	187.5 kPa (27.2 psi)
Internal Bladder	Butyl Rubber	Butyl Rubber
Meshing Type	Over-expanded Meshing	Over-expanded Meshing

As can be seen in Table 5-1, it was possible to obtain an experimental inflated length very close to the theoretical length. This was done by building multiple PAMs and testing each one at various pressures, within a small range of the theoretical pressure, until the desired length was obtained.

The butyl rubber bladder and the over-expanded meshing were chosen due to the results of the Chapter 3, which indicated that the stiffness model provided the best prediction of the force-length behaviour for this combination of material. PAMs with different characteristics were not tested due to the scope of this testing, which was to evaluate the feasibility of the unpowered ankle exoskeleton with a PAM.

5.2.4 Cycling Testing Pattern

Recognizing that the tensile testing machine is able to do cyclic loading, it was determined that the best way to duplicate the human motion would be to adjust a four cycle testing pattern to mimic a gait cycle. Since the ankle is locked in placed, it is possible to do this through vertical motion only due to the presence of the Horizontal Motion Component that allows horizontal motion.

An approximation of the linear velocity was determined through differentiation. As such, the vertical displacement and velocity are known at each point through the gait cycle. These values were entered in the control program of the tensile testing machine to create a predefined gait pattern.

5.3 Testing and Mechanical Validation

With the objective to demonstrate of the device's feasibility, testing was done in two ways. Firstly, it was tested manually to verify the operation of the timing and the locking mechanism without loads. Secondly, it was tested on a tensile testing machine to determine the feasibility of the device while it was loaded.

5.3.1 Manual Testing

With the prototype fabricated, the device was manually tested to troubleshoot the various components and mechanisms. Initially manual testing showed that the device had trouble locking into place. It was found this was caused by friction between the components of the timing mechanism, which were gliding on one another. With mechanical oil added, the lubricated components moved as they were intended. Initial testing also determined that the ankle shaft needed an additional component to fix them axially, to prevent them from falling toward the inside of the device. Considering the size of the parts, it was determined that machining was not possible. The shafts were fixed to the shank members using industrial glue and tape, thereby providing a quick solution without any need for additional machining or a complex redesign. Once dried, these connections proved very solid and prevented any axial motion.

Manual testing was done without any muscle attached to the prototype. The base was held firmly on a flat platform while the shank segment was moved back and forth, within the range of motion of the normal ankle. This was done to see if the locking mechanism would activate and deactivate as intended.

This cycling was done for 10 to 50 cycles by three different individuals to remove the designer's bias. The device was able to lock and unlock successfully. It was observed that additional lubrication was needed after about 1 week of rest, which tended to result in harder to move components.

5.3.2 Methodology and Testing

With the prototype and the mechanical jig fabricated, the experimental set-up ready and the PAM selected, a sound testing methodology was developed to promote a uniform and repeatable experiment. The experimental methodology consisted of the following steps:

1) Attaching the PAM to the prototype
    a) Attach the deflated selected PAM to the unpowered ankle exoskeleton
    b) Using the previously selected PAM, inflate it to desired pressure.
    c) Close the valve connecting the high-pressure gas source to the PAM, thus closing off the pressurized PAM from the external environment.
    d) Verify if air leaks are present. If so, restart with new muscle
    e) Measure the pressurized PAM geometric parameters.
2) Tensile Testing Set up
    a) Activate the tensile testing machine and select the cycling loading pattern defined previously
    b) Attach the Bolted Plate to the bottom of the tensile testing machine using bolts and nuts such that it is fixed and cannot move
    c) Link the Horizontal Motion Component to the tensile testing machine by sliding a smooth bar for that purpose through it.
    d) Solidly attach, using bolts and nuts, the bottom plate of the prototype to the vertical portion of the Bolted Plate. This should result with the exoskeleton on its back, where the inflated PAM is at the bottom of the assembly and the shank segment is in line with the Horizontal Motion Component.
    e) Slide a second smooth bar through the holes in the Shank bars and through the Horizontal Motion Component
    f) Reset the tensile testing machine so the PAM is in neutral position and no force is applied to the device
3) Data Acquisition Set-up
    a) Connect the pressure transducer, and the tensile testing machine to a data-acquisition software, set at 10Hz.
    b) Connect the pressure transducer to the muscle and open the valve.
4) Run the data acquisition software and the predefined cyclic testing at least 3 times for each experiment.

This methodology was followed for five different experimental conditions where the initial pressure was varied slightly to attempt to obtain the best possible match to the theoretical initial pressure, indicated in Table 5-1.

5.3.3 Analysis and Results

The objective of these tests is to show that the device does follow the gait cycle and creates a moment about the ankle at the correct instances of the gait cycle. Following the methodology detailed previously, the output of the testing was the PAM internal pressure, the force at the shank extensions and the vertical displacement of the tensile testing machine. Consequently, moment about the ankle can be calculated for each data point.

Similarly to the conversion of ankle angles to the vertical displacement of the tensile testing machine, it is possible to calculate the moment at the ankle by using the geometry relationships shown in Figure 5-9. In this figure, the angle γ is the angular offset of the PAM from the ankle angle caused by the geometric alignment of the attachments between the PAM and the prototype.

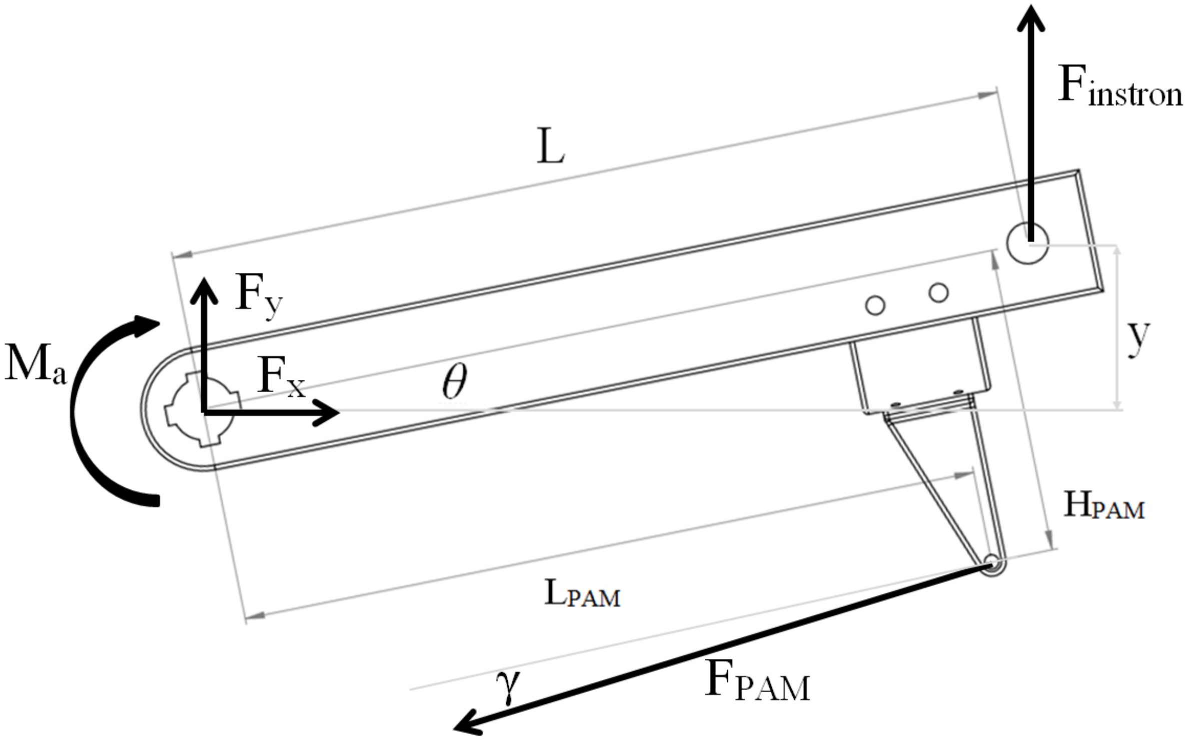


Figure 5-9: Geometry and force diagram of experimental testing setup

Then, it was possible to obtain a dynamic relationship between the ankle moment *M_a,* the muscle force *F_{PAM}* and the testing force *$F_{instron}$*.

$$M_a = F_{instron} L \cos\theta - F_{PAM} H_{PAM} \cos\gamma - F_{PAM} L_{PAM} \sin\gamma + L_{com} m a_x \sin\theta + L_{com} m a_y \cos\theta \tag{5-2}$$

where L_{com} is the position of the center of gravity of the structure, calculated using MatLAB and SolidWorks and verified manually, m is the mass of the moving components, and a_x and a_y are the linear accelerations of the center of mass calculated through a kinematic analysis.

This formula has the moment and the muscle force as unknowns. Considering that the muscle used had a butyl rubber internal bladder with an over-expanded meshing, the muscle force can be estimated by using the stiffness model developed in Chapter 3, where the muscle length L_{muscle} is defined in terms of gait cycle ankle angle and device geometry, as shown in (5-3) and (5-4).

$$F_{PAM} = k_{muscle} L_{muscle} \tag{5-3}$$

$$L_{muscle} = \sqrt{(L_{PAM} \cos\theta + H_{PAM} \sin\theta)^2 + (H_{Base} + L_{PAM} \sin\theta - H_{PAM} \cos\theta)^2} - L_{static} \tag{5-4}$$

where H_{Base} is the vertical distance between the ankle axis of rotation and the lower PAM anchor point, and L_{static} is the length of muscle end-caps and connectors.

5.3.4 Mechanical Validation

The mechanical validation of the exoskeleton consists in verifying the results of the experiment by comparing the theoretical and experimental ankle moment. This is done with regards to both the gait cycle progress and to the ankle angle.

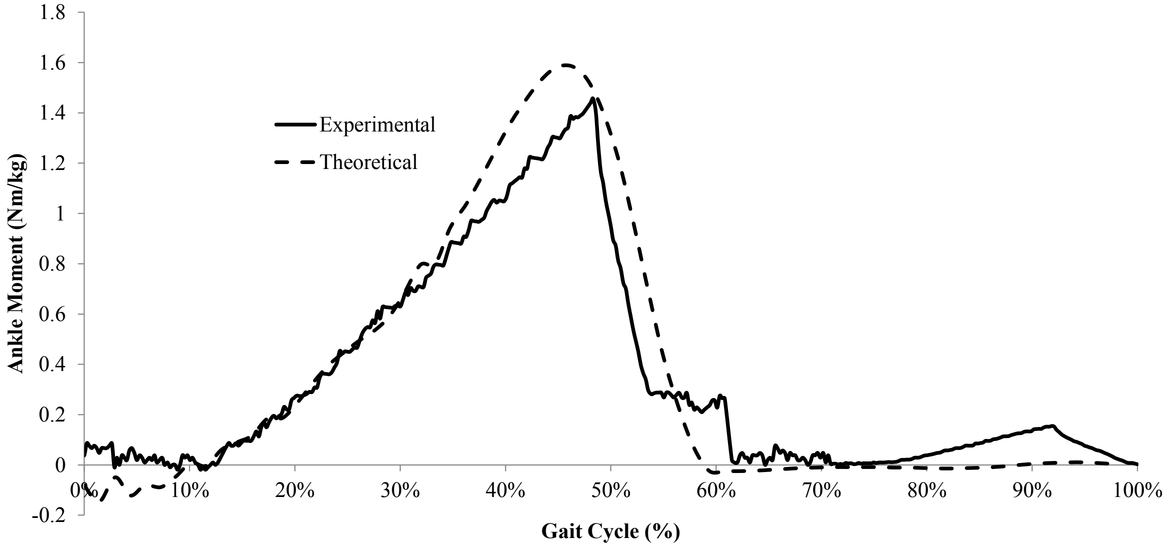


Figure 5-10: Ankle moment through the gait cycle

Figure 5-10 compares the experimental and theoretical ankle moment thoughout a gait cycle. This graph shows that the exoskeleton was able to create an ankle moment similar to what occurs during normal gait motion, with a similar slope during stance phase. One noticeable difference is the release of the moment, which does not converge to zero at push off and the presence of a second small peak during the swing phase.

The second small peak is attributed to the timing mechanism, which requires a small push to lock at the end of the swing phase in order to start harvesting energy. It is also created by the motion of the free ankle during swing phase which still creates some tension in the PAM.

The drop to a non-zero value at push-off is associated with the incomplete release of the muscle energy. This incomplete release is thought to be associated to the inertial acceleration of the device, which acts opposite to the muscle force until the ankle starts moving in the opposite direction during the swing phase.

Figure 5-11 compares the experimental and theoretical ankle moment with the ankle angle throughout the gait cycle, informing the amount of angular stiffness generated. The ankle exoskeleton was able to mimic the behaviour and stiffness of the ankle over a gait cycle.

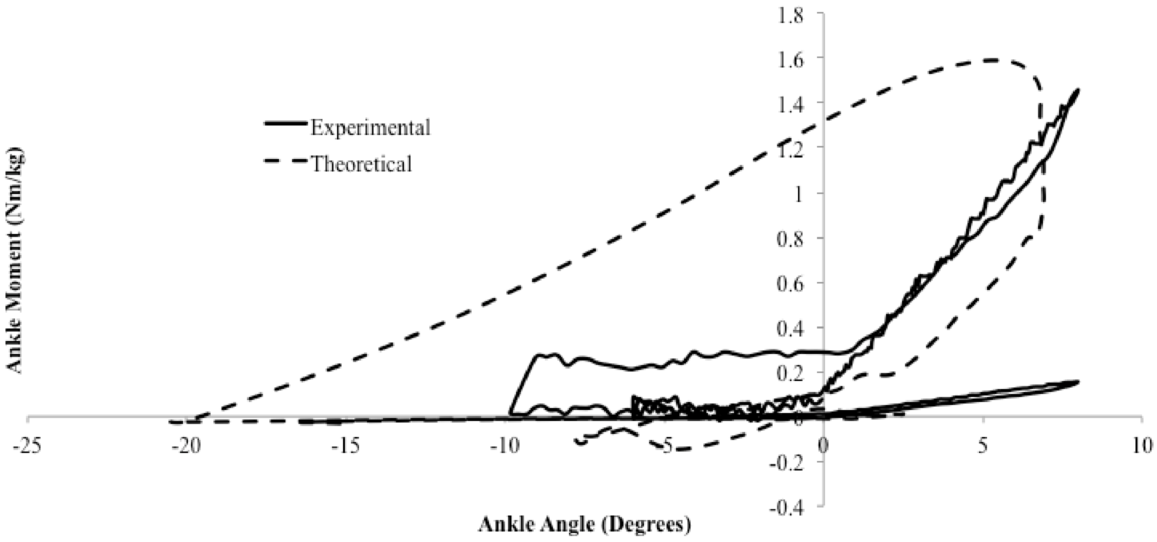


Figure 5-11: Ankle moment with regards to the ankle angle through a gait cycle

The device was not able to create the same range of ankle stiffness as the normal ankle over a gait cycle. Most notably is the presence of a constant moment of 0.3 Nm/kg between the angles of -10 to 5 degrees. This was referenced as the drop to non-zero in the previous graph. Whereas the loading occurs at a much larger rate in the experimental curve compared to the theoretical curve, they reach a similar maximum and are followed by an unloading at a comparable rate. Finally, the experimental data suggests there is limited loading over the swing phase of the gait cycle, concurrent the theoretical data.

5.4 Conclusion

The present chapter presented the fabrication of the exoskeleton prototype and of the mechanical jig, the experimental set-up and methodology, as well as the experimental results of the tests to show the feasibility of the device.

The fabrication of the device required significant modification to the proposed design to allow the use of simpler manufacturing processes. Whereas the design was initially conceptualized to be a lightweight conforming material, the prototype was made of aluminum and steel, thus permitting the use of readily available resources and reduced manufacturing time. The prototype weighed approximately 1.35 kg, suggesting a much lighter weight is achievable when using lightweight mouldable materials, like carbon-fiber.

The experiment was achieved on an Instron tensile testing machine using a predefined cycle mimicking the gait cycle, which required a mechanical jig to be fabricated to accommodate the unique motion of the exoskeleton.

Testing was first done manually, without an inflated PAM, to verify the locking and unlocking ability of the device. The device was shown to be effective at locking and unlocking, but required some lubrication and modification to prevent axial motion of the ankle shafts. Then, mechanical testing was completed by attaching an inflated PAM to the exoskeleton. During this testing, force, pressure and elongation data was acquired. From this data, the experimental ankle moment was calculated and compared to the theoretical ankle data over a gait cycle.

The experimental results indicate that the device was successful at creating moments of similar scale to the theoretical ankle with a timing that reflects the natural ankle. The device was not able to release all its harvested energy in one instance at push off and presented a small moment during the swing phase. Overall the exoskeleton prototype has shown its feasibility to mechanically follow the human motion while being loaded and unloaded through the gait cycle. Moreover, the device was able to obtain similar ankle moment-degree slopes during the stance phase of the gait cycle.

Chapter 6– CONCLUSIONS AND RECOMMENDATIONS

The main objective of this thesis was to design, fabricate and evaluate the feasibility of an unpowered ankle exoskeleton. This device aimed to assist individuals in walking by capturing and releasing energy throughout the gait cycle using a PAM as a source of actuator, and a mechanical locking mechanism. A newly developed PAM stiffness model was achieved given the limited stiffness model available in open literature and its requirement for the design of the unpowered ankle exoskeleton.

The present chapter presents a conclusion of the PAM stiffness model development and validation, of the passive ankle exoskeleton development and validation, as well as general conclusions and recommendations for future development.

6.1 PAM Stiffness Model

6.1.1 Conclusion

Following the development of the PAM stiffness model that took into consideration the geometry and friction parameters, the passive muscle behaviour was experimentally evaluated by performing eccentric contraction on a tensile testing machine. The parameters tested include the internal bladder material, the braided mesh type, the elongation velocity and the initial internal pressure. These tests enabled a full analysis of the PAM non-linear behaviour, and were used to validate the developed PAM stiffness model. The proposed PAM frictional stiffness model presented fairly accurate results for the entire range of parameters tested, where the better results were obtained for PAMs with butyl rubber internal bladders and NB braided mesh, at any initial pressure and elongation velocity.

In terms of results, the more compliant internal bladder tended to decrease the overall stiffness, and the initial pressure and elongation velocity had limited impact on the PAM stiffness over the range evaluated. Although friction was considered, the hysteresis appeared to be largest for meshing with fewer fibers, contrary to what the theory had proposed. This result suggests that other energy loss mechanisms are present to cause the hysteresis over a contraction cycle

6.2 Ankle Exoskeleton Design

The unpowered ankle exoskeleton consists of a solid structure that surrounds a user's calf and ankle, and allows ankle extension and flexion. The device uses a sequencing mechanism to charge and release a passive PAM located at the back of the ankle, in parallel with the user's Achilles tendon. The device was modeled using MATLAB interfacing with a computer-aided design software, SolidWorks. Technicians at the University of Ottawa machine shop used steel and aluminum to manufacture a prototype.

The sequencing mechanism consists of a ratchet and pawl combined with a cam-slider mechanism. The slider's motions are controlled by a cam, which moves with the user's shank. The slider, which is attached to the pawl, moves vertically to mesh and un-mesh the pawl to the ratchet. The sequencing mechanism causes the pawl and ratchet to be meshed during the stance phase, thereby charging the PAM, and un-meshed during the swing phase, releasing the PAM.

The exoskeleton prototype was tested to show the feasibility of the device to provide the required ankle moment through the gait cycle, using a tensile testing machine with cyclic loadings, mimicking the gait cycle. The exoskeleton prototype has shown that it is capable of producing ankle-moment behaviour similar to that of a human ankle, in terms of scale and in terms of timing. This shows that a purely mechanical system combined with passive PAM can duplicate human ankle motion and create a moment about the ankle that could provide assistance during gait.

6.3 Conclusion

Mobility assist device that promote human gait have the potential to increase an individual's quality of life by increasing their independence. Although these technologies are widespread, they present significant challenges in terms of weight, dimensions and cyclic behaviour. Present technologies use either strong, bulky motors that require significant power, or soft elastic material that store and release energy.

The present thesis presented a novel passive ankle exoskeleton that harnesses the passive use of PAM, combined with a purely mechanical timing mechanism. A comprehensive stiffness model was developed and validated for the PAM that incorporates the muscle geometry and friction. A multi-design testing of PAMs indicated that friction between the fibers and the bladder is not relevant, and was dismissed in the final model.

The final PAM stiffness model was incorporated in the design of the exoskeleton. The design process of the device was presented, as well as a comparison of the ankle moment against the ankle angle over a gait cycle. This showed that the exoskeleton was able to duplicate human ankle motion and moment through the gait cycle, thereby showing the feasibility of the design.

6.4 Future Work

6.4.1 PAM Stiffness Model

Although many parameters were tested and evaluated during the development of the stiffness model, many other factors were assumed to be negligible.

One energy loss mechanism that warrants further investigation is the behaviour between the internal bladder and the braided mesh. The present model assumed that the interaction between these components was limited to surface-to-surface behaviour. However, it was noticed experimentally that the inflated bladder tends to fill the voids in the braided mesh when it is inflated. This would increase the friction area between the internal bladder and the braided mesh, and could entirely stop the PAMs motion.

Other parameters that could merit further investigations with regards to the overall PAM passive behaviour are the internal bladder force-length behaviour, the thermodynamics process involved in the PAM contraction, the braided mesh extensibility, the end-effects, and the presence of slippage in the end-caps.

6.4.2 Ankle Exoskeleton Design

In terms of future directions, the exoskeleton could be tested on humans in a biomechanical laboratory to verify its effectiveness. The device could also be streamlined and built using a carbon fiber structure for a lighter prototype. Other improvements include expanding to the hip and knee joints, use gears in the sequencing mechanism to obtain a smaller resolution, and improve the device's aesthetics.

One of the main issues encountered was the need for lubrication for proper functioning. A simple improvement would be to increase the clearances of the components and reduce the angles of the timing mechanism to allow for smoother motion.

REFERENCES

[1] K. R. Westerterp, “Physical activity and physical activity induced energy expenditure in humans: measurement, determinants, and effects,” *Front. Physiol.*, vol. 4, p. 90, 2013.

[2] C. Tudor-Locke, W. D. Johnson, and P. T. Katzmarzyk, “Accelerometer-determined steps per day in US adults,” *Med. Sci. Sports Exerc.*, vol. 41, no. 7, pp. 1384–1391, Jul. 2009.

[3] C. Kirtley, *Clinical gait analysis theory and practice*, First. Edinburgh ; New York: Elsevier, 2006.

[4] H. Herr, “Exoskeletons and orthoses: classification, design challenges and future directions,” *J. NeuroEngineering Rehabil.*, vol. 6, no. 1, p. 21, 2009.

[5] S. H. Collins, M. B. Wiggin, and G. S. Sawicki, “Reducing the energy cost of human walking using an unpowered exoskeleton,” *Nature*, vol. 522, no. 7555, pp. 212–215, Jun. 2015.

[6] L. M. Mooney, E. J. Rouse, and H. M. Herr, “Autonomous exoskeleton reduces metabolic cost of human walking during load carriage,” *J. NeuroEngineering Rehabil.*, vol. 11, no. 1, p. 80, May 2014.

[7] L. C. Rome, L. Flynn, and T. D. Yoo, “Biomechanics: Rubber bands reduce the cost of carrying loads,” *Nature*, vol. 444, no. 7122, pp. 1023–1024, Dec. 2006.

[8] P. Malcolm, W. Derave, S. Galle, and D. De Clercq, “A simple exoskeleton that assists plantarflexion can reduce the metabolic cost of human walking,” *PloS One*, vol. 8, no. 2, p. e56137, 2013.

[9] D. A. Winter, *Biomechanics and Motor Control of Human Movement, Fourth Edition*. John Wiley & Sons, 2009.

[10] D. Roberts, “Winter’s gait data in Excel form | Dustyn Robots,” 27-Mar-2012. .

[11] M. Whittle, *Gait analysis an introduction*, 4th ed.. Edinburgh ; New York: Butterworth-Heinemann, 2007.

[12] P. Cherelle, A. Matthys, V. Grosu, B. Vanderborght, and D. Lefeber, “The AMP-Foot 2.0: Mimicking intact ankle behavior with a powered transtibial prosthesis,” in *2012 4th IEEE RAS EMBS International Conference on Biomedical Robotics and Biomechatronics (BioRob)*, 2012, pp. 544–549.

[13] R. Versluys, R. Van Ham, I. Vanderniepen, and D. Lefeber, “Successful walking with a biologically-inspired below-knee prosthesis,” in *IEEE International Conference on Rehabilitation Robotics, 2009. ICORR 2009*, 2009, pp. 652–657.

[14] D. P. Ferris, G. S. Sawicki, and A. Domingo, “Powered lower limb orthoses for gait rehabilitation,” *Top. Spinal Cord Inj. Rehabil.*, vol. 11, no. 2, pp. 34–49, 2005.

[15] J. Hitt, A. M. Oymagil, T. Sugar, K. Hollander, A. Boehler, and J. Fleeger, “Dynamically Controlled Ankle-Foot Orthosis (DCO) with Regenerative Kinetics: Incrementally Attaining User Portability,” in *2007 IEEE International Conference on Robotics and Automation*, 2007, pp. 1541–1546.

[16] J. Carberry, G. Hinchly, J. Buckerfield, E. Tayler, T. Burton, S. Madgwick, and R. Vaidyanathan, “Parametric design of an active ankle foot orthosis with passive compliance,” in *2011 24th International Symposium on Computer-Based Medical Systems (CBMS)*, 2011, pp. 1–6.

[17] J. Zhu, Q. Wang, Y. Huang, and L. Wang, “Adding compliant joints and segmented foot to bio-inspired below-knee exoskeleton,” in *2011 IEEE International Conference on Robotics and Automation (ICRA)*, 2011, pp. 605–610.

[18] T. M. Griffin, T. J. Roberts, and R. Kram, “Metabolic cost of generating muscular force in human walking: insights from load-carrying and speed experiments,” *J. Appl. Physiol.*, vol. 95, no. 1, pp. 172–183, Jul. 2003.

[19] E. N. Marieb and K. Hoehn, *Human anatomy & physiology*. Boston: Pearson, 2013.

[20] D. L. Bartel, D. T. Davy, and T. M. Keaveny, *Orthopaedic Biomechanics: Mechanics and Design in Musculoskeletal Systems*, 1st ed. Prentice Hall, 2006.

[21] M. Doumit, A. Fahim, and M. Munro, “Analytical Modeling and Experimental Validation of the Braided Pneumatic Muscle,” *IEEE Trans. Robot.*, vol. 25, no. 6, pp. 1282–1291, Dec. 2009.

[22] J. Leclair, M. Doumit, and G. McAllister, “Analytical Stiffness Modeling and Experimental Validation for a Pneumatic Artificial Muscle,” p. V009T12A089, Nov. 2014.

[23] E. Avallone, T. Baumeister, and A. Sadegh, *Marks’ Standard Handbook for Mechanical Engineers*, 11th ed. McGraw-Hill Professional, 2006.

[24] “Physical Examimation,” *OrthopaedicsOne Articles. In: OrthopaedicsOne - The Orthopaedic Knowledge Network*, 08-Apr-2012. [Online]. Available: http://www.orthopaedicsone.com/x/IYFF. [Accessed: 03-Dec-2013].

[25] B. Hintermann, *Total Ankle Arthroplasty: Historical Overview, Current Concepts and Future Perspectives*. Springer, 2005.

[26] A. Chu, H. Kazerooni, and A. Zoss, “On the Biomimetic Design of the Berkeley Lower Extremity Exoskeleton (BLEEX),” in *Proceedings of the 2005 IEEE International Conference on Robotics and Automation, 2005. ICRA 2005*, 2005, pp. 4345 – 4352.

[27] C. T. Farley and D. P. Ferris, “Biomechanics of walking and running: center of mass movements to muscle action,” *Exerc. Sport Sci. Rev.*, vol. 26, pp. 253–285, 1998.

[28] A. B. Zoss, H. Kazerooni, and A. Chu, “Biomechanical design of the Berkeley lower extremity exoskeleton (BLEEX),” *IEEEASME Trans. Mechatron.*, vol. 11, no. 2, pp. 128 –138, Apr. 2006.

[29] S. Ounpuu, “Biomechanics of Walking and Running,” *Clin. Sports Med.*, vol. 13, no. 4, pp. 843 – 863, Oct. 1994.

[30] R. Riener, M. Rabuffetti, and C. Frigo, “Stair ascent and descent at different inclinations,” *Gait Posture*, vol. 15, no. 1, pp. 32–44, Feb. 2002.

[31] S. A. Dugan and K. P. Bhat, “Biomechanics and analysis of running gait,” *Phys. Med. Rehabil. Clin. N. Am.*, vol. 16, no. 3, pp. 603–621, Aug. 2005.

[32] A. Protopapadaki, W. I. Drechsler, M. C. Cramp, F. J. Coutts, and O. M. Scott, “Hip, knee, ankle kinematics and kinetics during stair ascent and descent in healthy young individuals,” *Clin. Biomech.*, vol. 22, no. 2, pp. 203–210, Feb. 2007.

[33] J. M. Brockway, “Derivation of formulae used to calculate energy expenditure in man,” *Hum. Nutr. Clin. Nutr.*, vol. 41, no. 6, pp. 463–471, Nov. 1987.

[34] J. S. Gottschall and R. Kram, “Energy cost and muscular activity required for propulsion during walking,” *J. Appl. Physiol.*, vol. 94, no. 5, pp. 1766–1772, May 2003.

[35] C. T. Farley and T. A. McMahon, “Energetics of walking and running: insights from simulated reduced-gravity experiments,” *J. Appl. Physiol.*, vol. 73, no. 6, pp. 2709–2712, Dec. 1992.

[36] B. Smith, M. Roan, and M. Lee, “The effect of evenly distributed load carrying on lower body gait dynamics for normal weight and overweight subjects,” *Gait Posture*, vol. 32, no. 2, pp. 176–180, Jun. 2010.

[37] R. Y. W. Lee and J. Munn, “Passive moment about the hip in straight leg raising,” *Clin. Biomech.*, vol. 15, no. 5, pp. 330–334, Jun. 2000.

[38] A. Silder, B. Whittington, B. Heiderscheit, and D. G. Thelen, “IDENTIFICATION OF PASSIVE ELASTIC JOINT MOMENT-ANGLE RELATIONSHIPS IN THE LOWER EXTREMITY,” *J. Biomech.*, vol. 40, no. 12, pp. 2628–2635, 2007.

[39] C. Frigo, P. Crenna, and L. M. Jensen, “Moment-angle relationship at lower limb joints during human walking at different velocities,” *J. Electromyogr. Kinesiol.*, vol. 6, no. 3, pp. 177–190, Sep. 1996.

[40] R. B. Davis and P. A. DeLuca, “Gait characterization via dynamic joint stiffness,” *Gait Posture*, vol. 4, no. 3, pp. 224–231, May 1996.

[41] R. J. Butler, H. P. Crowell III, and I. M. Davis, “Lower extremity stiffness: implications for performance and injury,” *Clin. Biomech.*, vol. 18, no. 6, pp. 511–517, Jul. 2003.

[42] R. C. Gabriel, J. Abrantes, K. Granata, J. Bulas-Cruz, P. Melo-Pinto, and V. Filipe, “Dynamic joint stiffness of the ankle during walking: Gender-related differences,” *Phys. Ther. Sport*, vol. 9, no. 1, pp. 16–24, Feb. 2008.

[43] J. Folland and B. Morris, “Variable-cam resistance training machines: Do they match the angle – torque relationship in humans?,” *J. Sports Sci.*, vol. 26, no. 2, pp. 163–169, 2008.

[44] J. A. Zeni Jr. and J. S. Higginson, “Dynamic knee joint stiffness in subjects with a progressive increase in severity of knee osteoarthritis,” *Clin. Biomech.*, vol. 24, no. 4, pp. 366–371, May 2009.

[45] H. Tateuchi, R. Tsukagoshi, Y. Fukumoto, S. Oda, and N. Ichihashi, “Dynamic hip joint stiffness in individuals with total hip arthroplasty: Relationships between hip impairments and dynamics of the other joints,” *Clin. Biomech.*, vol. 26, no. 6, pp. 598–604, Jul. 2011.

[46] D. Hahn, M. Olvermann, J. Richtberg, W. Seiberl, and A. Schwirtz, “Knee and ankle joint torque–angle relationships of multi-joint leg extension,” *J. Biomech.*, vol. 44, no. 11, pp. 2059–2065, Jul. 2011.

[47] R. Wang, E. W. Broström, A.-C. Esbjörnsson, and E. M. Gutierrez-Farewik, “Analytical decomposition can help to interpret ankle joint moment–angle relationship,” *J. Electromyogr. Kinesiol.*, vol. 22, no. 4, pp. 566–574, Aug. 2012.

[48] H. S. Longpré, J. R. Potvin, and M. R. Maly, “Biomechanical changes at the knee after lower limb fatigue in healthy young women,” *Clin. Biomech.*, vol. 28, no. 4, pp. 441–447, Apr. 2013.

[49] K. McGinnis, L. Snyder-Mackler, P. Flowers, and J. Zeni, “Dynamic joint stiffness and co-contraction in subjects after total knee arthroplasty,” *Clin. Biomech.*, vol. 28, no. 2, pp. 205–210, Feb. 2013.

[50] R. Dodel and A. Schrag, “Health-Related Quality of Life in Movement Disorders,” in *Handbook of Disease Burdens and Quality of Life Measures*, V. R. Preedy and R. R. Watson, Eds. Springer New York, 2010, pp. 4013–4034.

[51] S. H. Collins and A. D. Kuo, “Recycling Energy to Restore Impaired Ankle Function during Human Walking,” *PLoS ONE*, vol. 5, no. 2, Feb. 2010.

[52] P. Cherelle, V. Grosu, A. Matthys, B. Vanderborght, and D. Lefeber, “Design and Validation of the Ankle Mimicking Prosthetic (AMP-) Foot 2.0,” *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 22, no. 1, pp. 138–148, Jan. 2014.

[53] S. K. Au, J. Weber, and H. Herr, “Powered Ankle–Foot Prosthesis Improves Walking Metabolic Economy,” *IEEE Trans. Robot.*, vol. 25, no. 1, pp. 51–66, 2009.

[54] D. Bishop, A. Moore, and N. Chandrashekar, “A New Ankle Foot Orthosis for Running,” *Prosthet. Orthot. Int.*, vol. 33, no. 3, pp. 192–197, Sep. 2009.

[55] “50K30 | Xeleton Knee Brace,” *Ottobock*, 2015. [Online]. Available: http://professionals.ottobockus.com/cps/rde/xchg/ob_us_en/hs.xsl/43764.html. [Accessed: 14-Jan-2015].

[56] H. Hirai, R. Ozawa, S. Goto, H. Fujigaya, S. Yamasaki, Y. Hatanaka, and S. Kawamura, “Development of an ankle-foot orthosis with a pneumatic passive element,” in *The 15th IEEE International Symposium on Robot and Human Interactive Communication, 2006. ROMAN 2006*, 2006, pp. 220–225.

[57] D. P. Ferris, J. M. Czerniecki, and B. Hannaford, “An Ankle-Foot Orthosis Powered by Artificial Pneumatic Muscles,” *J. Appl. Biomech.*, vol. 21, no. 2, pp. 189–197, May 2005.

[58] G. S. Sawicki and D. P. Ferris, “Powered ankle exoskeletons reveal the metabolic cost of plantar flexor mechanical work during walking with longer steps at constant step frequency,” *J. Exp. Biol.*, vol. 212, no. 1, pp. 21–31, Jan. 2009.

[59] A. Agrawal, S. K. Banala, S. K. Agrawal, and S. A. Binder-Macleod, “Design of a two degree-of-freedom ankle-foot orthosis for robotic rehabilitation,” in *9th International Conference on Rehabilitation Robotics, 2005. ICORR 2005*, 2005, pp. 41–44.

[60] M. B. Wiggin, G. S. Sawicki, and S. H. Collins, “An exoskeleton using controlled energy storage and release to aid ankle propulsion,” in *2011 IEEE International Conference on Rehabilitation Robotics (ICORR)*, 2011, pp. 1–5.

[61] N. Yagn, “Apparatus for facilitating walking,” US420179 A, 28-Jan-1890.

[62] B. J. Makinson, “Research and Development Prototype for Machine Augmentation of Human Strength and Endurance. Hardiman I Project,” p. 33, 1971.

[63] A. Seireg and J. G. Grundmann, “Design of a multitask exoskeletal walking device for paraplegics,” *Biomech. Med. Devices*, pp. 569–644, 1981.

[64] M. Vukobratovic, “Development of active anthropomorphic exoskeletons,” in *Medical and Biological Engineering*, 1974, pp. 66–80.

[65] C. J. Walsh, K. Pasch, and H. Herr, “An autonomous, underactuated exoskeleton for load-carrying augmentation,” in *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2006, pp. 1410–1415.

[66] C. J. Walsh, D. Paluska, K. Pasch, W. Grand, A. Valiente, and H. Herr, “Development of a lightweight, underactuated exoskeleton for load-carrying augmentation,” in *Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006*, 2006, pp. 3485–3491.

[67] C. J. Walsh, K. Endo, and H. Herr, “A QUASI-PASSIVE LEG EXOSKELETON FOR LOAD-CARRYING AUGMENTATION,” *Int. J. Humanoid Robot.*, vol. 04, no. 03, pp. 487–506, Sep. 2007.

[68] K. Endo and H. Herr, "A model of muscle-tendon function in human walking," in *IEEE International Conference on Robotics and Automation, 2009. ICRA '09*, 2009, pp. 1909–1915.

[69] K. Endo, D. Paluska, and H. Herr, "A quasi-passive model of human leg function in level-ground walking," in *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2006, pp. 4935–4939.

[70] Walsh, C.J., "Biomimetic Design of an Under-Actuated Leg Exoskeleton For Load-Carrying Augmentation," Massachusetts Institute of Technology, 2006.

[71] S. K. Banala, S. K. Agrawal, A. Fattah, V. Krishnamoorthy, H. Wei-Li, J. Scholz, and K. Rudolph, "Gravity-Balancing Leg Orthosis and Its Performance Evaluation," *IEEE Trans. Robot.*, vol. 22, no. 6, pp. 1228–1239, 2006.

[72] S. K. Agrawal, S. K. Banala, A. Fattah, V. Sangwan, V. Krishnamoorthy, J. P. Scholz, and H. Wei-Li, "Assessment of Motion of a Swing Leg and Gait Rehabilitation With a Gravity Balancing Exoskeleton," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 15, no. 3, pp. 410–420, 2007.

[73] A. Zoss, H. Kazerooni, and A. Chu, "On the mechanical design of the Berkeley lower extremity exoskeleton (BLEEX)," in *Intelligent Robots and Systems, 2005.(IROS 2005). 2005 IEEE/RSJ International Conference on*, 2005, pp. 3465–3472.

[74] A. Zoss and H. Kazerooni, "Design of an electrically actuated lower extremity exoskeleton," *Adv. Robot.*, vol. 20, no. 9, pp. 967–988, 2006.

[75] Ekso Bionics, "Ekso Bionics for the Human Endeavour," *Ekso Bionics*, 2013. [Online]. Available: http://intl.eksobionics.com/. [Accessed: 15-Jan-2015].

[76] Honda, "Honda Walking Assist Device with Bodyweight Support Assist." Honda, Sep-2008.

[77] Y. Ikeuchi, J. Ashihara, Y. Hiki, H. Kudoh, and T. Noda, "Walking assist device with bodyweight support system," in *IEEE/RSJ International Conference on Intelligent Robots and Systems, 2009. IROS 2009*, 2009, pp. 4073 –4079.

[78] J. Ashihara, Y. Matsuoka, H. Matsuda, Y. Hiki, and M. Shishido, "Walking Assist Device," 0306907, 15-Dec-2011.

[79] Honda, "Walking Assist Device with Stride Management Assist." Honda, 2009-2008.

[80] Y. Endo, K. Kikuchi, and R. Hara, "Assist Device," 0049333, 25-Feb-2010.

[81] N. Tsagarakis, D. G. Caldwell, and G. A. Medrano-Cerda, "A 7 DOF pneumatic muscle actuator (pMA) powered exoskeleton," in *8th IEEE International Workshop on Robot and Human Interaction, 1999. RO-MAN '99*, 1999, pp. 327–333.

[82] N. Costa and D. G. Caldwell, "Control of a Biomimetic 'Soft-actuated' 10DoF Lower Body Exoskeleton," in *The First IEEE/RAS-EMBS International Conference on Biomedical Robotics and Biomechatronics, 2006. BioRob 2006*, 2006, pp. 495–501.

[83] T. Ikehara, K. Nagamura, T. Ushida, E. Tanaka, S. Saegusa, S. Kojima, and L. Yuge, "Development of closed-fitting-type walking assistance device for legs and evaluation of muscle activity," in *2011 IEEE International Conference on Rehabilitation Robotics (ICORR)*, 2011, pp. 1 –7.

[84] M. Arnout, C. Pierre, V. D. Michael, V. Bram, and L. Dirk, "Concept and design of the HEKTA (Harvest Energy from the Knee and Transfer it to the Ankle) transfemoral prosthesis," in *2012 4th IEEE RAS EMBS International Conference on Biomedical Robotics and Biomechatronics (BioRob)*, 2012, pp. 550–555.

[85] W. C. Flowers, "A man-interactive simulator system for above-knee prosthetics studies.," Thesis, Massachusetts Institute of Technology, 1973.

[86] R. J. Farris, H. A. Quintero, T. J. Withrow, and M. Goldfarb, “Design of a joint-coupled orthosis for FES-aided gait,” in *IEEE International Conference on Rehabilitation Robotics, 2009. ICORR 2009*, 2009, pp. 246 –252.

[87] P. Beyl, M. Van Damme, R. Van Ham, B. Vanderborght, and D. Lefeber, “Pleated Pneumatic Artificial Muscle-Based Actuator System as a Torque Source for Compliant Lower Limb Exoskeletons,” *IEEEASME Trans. Mechatron.*, vol. Early Access Online, 2013.

[88] A. Sergeyev, N. Alaraje, C. Seidel, Z. Carlson, and B. Breda, “Design of a pneumatically powered wearable exoskeleton with biomimetic support and actuation,” in *2013 IEEE Aerospace Conference*, 2013, pp. 1–8.

[89] W. van Dijk, H. van der Kooij, and E. Hekman, “A passive exoskeleton with artificial tendons: Design and experimental evaluation,” in *2011 IEEE International Conference on Rehabilitation Robotics (ICORR)*, 2011, pp. 1–6.

[90] H. Kawamoto and Y. Sankai, “Comfortable power assist control method for walking aid by HAL-3,” in *2002 IEEE International Conference on Systems, Man and Cybernetics*, 2002, vol. 4, p. 6 pp. vol.4.

[91] Cyberdyne Inc., “‘Robot Suit HAL’ for Well-being,” *Cyberdyne*. [Online]. Available: http://www.cyberdyne.jp/english/customer/index.html. [Accessed: 11-Sep-2012].

[92] K. H. Low, X. Liu, H. Y. Yu, and H. S. Kasim, “Development of a lower extremity exoskeleton - preliminary study for dynamic walking,” in *Control, Automation, Robotics and Vision Conference, 2004. ICARCV 2004 8th*, 2004, vol. 3, pp. 2088–2093 Vol. 3.

[93] K. H. Low, X. Liu, and H. Yu, “Development of NTU wearable exoskeleton system for assistive technologies,” in *Mechatronics and Automation, 2005 IEEE International Conference*, 2005, vol. 2, pp. 1099–1106 Vol. 2.

[94] F. Chen, Y. Yu, Y. Ge, J. Sun, and X. Deng, “WPAL for Enhancing Human Strength and Endurance during Walking,” in *International Conference on Information Acquisition, 2007. ICIA ’07*, 2007, pp. 487 –491.

[95] F. Chen, Y. Yu, Y. Ge, and Y. Fang, “WPAL for human power assist during walking using dynamic equation,” in *International Conference on Mechatronics and Automation, 2009. ICMA 2009*, 2009, pp. 1039 –1043.

[96] H. Yusa, E. Tanaka, T. Ikehara, K. Ito, S. Saegusa, K. Hashimoto, Y. Sato, and L. Yuge, “Development of a walking assistance apparatus using a spatial parallel link mechanism and evaluation of muscle activity,” in *2010 IEEE RO-MAN*, 2010, pp. 151 –158.

[97] E. Tanaka, T. Ikehara, Y. Sato, H. Yusa, T. Sakurai, S. Saegusa, K. Ito, and L. Yuge, “Walking assistance apparatus using a spatial parallel link mechanism and a weight bearing lift,” in *2011 IEEE International Conference on Rehabilitation Robotics (ICORR)*, 2011, pp. 1 –7.

[98] S. Karlin, “Raiding Iron Man’s closet [Geek Life],” *IEEE Spectr.*, vol. 48, no. 8, pp. 25–25, 2011.

[99] K. Kong and D. Jeon, “Design and control of an exoskeleton for the elderly and patients,” *IEEEASME Trans. Mechatron.*, vol. 11, no. 4, pp. 428–432, Aug. 2006.

[100] H. K. Kwa, J. H. Noorden, M. Missel, T. Craig, J. E. Pratt, and P. D. Neuhaus, “Development of the IHMC Mobility Assist Exoskeleton,” in *IEEE International Conference on Robotics and Automation, 2009. ICRA ’09*, 2009, pp. 2556–2562.

[101] P. D. Neuhaus, J. H. Noorden, T. J. Craig, T. Torres, J. Kirschbaum, and J. E. Pratt, "Design and evaluation of Mina: A robotic orthosis for paraplegics," in *2011 IEEE International Conference on Rehabilitation Robotics (ICORR)*, 2011, pp. 1–8.

[102] T. Kagawa and Y. Uno, "Gait pattern generation for a power-assist device of paraplegic gait," in *Robot and Human Interactive Communication, 2009. RO-MAN 2009. The 18th IEEE International Symposium on*, 2009, pp. 633–638.

[103] S. Tanabe, E. Saitoh, S. Hirano, M. Katoh, T. Takemitsu, A. Uno, Y. Shimizu, Y. Muraoka, and T. Suzuki, "Design of the Wearable Power-Assist Locomotor (WPAL) for paraplegic gait reconstruction," *Disabil. Rehabil. Assist. Technol.*, vol. 8, no. 1, pp. 84–91, Jan. 2013.

[104] H. A. Quintero, R. J. Farris, and M. Goldfarb, "Control and implementation of a powered lower limb orthosis to aid walking in paraplegic individuals," in *2011 IEEE International Conference on Rehabilitation Robotics (ICORR)*, 2011, pp. 1 –6.

[105] A. Esquenazi, M. Talaty, A. P. Packel, and M. Saulino, "The ReWalk Powered Exoskeleton to Restore Ambulatory Function to Individuals with Thoracic-Level Motor-Complete Spinal Cord Injury," *J. Phys. Med.*, vol. 91, no. 11, pp. 911–921, 2012.

[106] Rewalk Bionic Research, "Rewalk - Bionic Suit," *Rewalk - Bionic Suit*, 2013. [Online]. Available: http://rewalk.us/. [Accessed: 16-Oct-2013].

[107] G. Colombo, M. Joerg, R. Schreier, and V. Dietz, "Treadmill training of paraplegic patients using a robotic orthosis," *J. Rehabil. Res. Dev.*, vol. 37, no. 6, pp. 693–700, Dec. 2000.

[108] S. Jezernik, A. Pfister, H. Frueh, G. Colombo, and M. Morari, "Robotic orthosis Lokomat: its use in the rehabilitation of locomotion and in the development of the biology-based neural controller," in *Conference of the International Functional Electrical Stimulation Society*, 2002.

[109] S. Jezernik, G. Colombo, T. Keller, H. Frueh, and M. Morari, "Robotic Orthosis Lokomat: A Rehabilitation and Research Tool," *Neuromodulation Technol. Neural Interface*, vol. 6, no. 2, pp. 108–115, 2003.

[110] Hocoma GmbH, "LokomatPro - Enhanced Functional Locomotion Therapy with Augmented Performance Feedback," *Hocoma*, 2013. [Online]. Available: http://www.hocoma.com/products/lokomat/lokomatpro/. [Accessed: 21-Nov-2013].

[111] S. Hesse and D. Uhlenbrock, "A mechanized gait trainer for restoration of gait," *J. Rehabil. Res. Dev.*, vol. 37, no. 6, p. 701, Dec. 2000.

[112] X. Li, H. Xia, and T. Guan, "Development of Legs Rehabilitation Exercise System Driven by Pneumatic Muscle Actuator," in *The 2nd International Conference on Bioinformatics and Biomedical Engineering, 2008. ICBBE 2008*, 2008, pp. 1309–1311.

[113] Y. Allemand, Y. Stauffer, R. Clavel, and R. Brodard, "Design of a new lower extremity orthosis for overground gait training with the WalkTrainer," in *IEEE International Conference on Rehabilitation Robotics, 2009. ICORR 2009*, 2009, pp. 550–555.

[114] G. Belforte, G. Eula, S. Appendino, and S. Sirolli, "Pneumatic Interactive Gait Rehabilitation Orthosis: Design and Preliminary Testing," *Proc. Inst. Mech. Eng. [H]*, vol. 225, no. 2, pp. 158–169, Feb. 2011.

[115] C. Teng, Z. Wong, W. Teh, and Y. Z. Chong, "Design and development of inexpensive pneumatically-powered assisted knee-ankle-foot orthosis for gait

rehabilitation-preliminary finding," in *2012 International Conference on Biomedical Engineering (ICoBE)*, 2012, pp. 28–32.

[116] R. Ham, T. G. Sugar, B. Vanderborght, K. W. Hollander, and D. Lefeber, "Compliant actuator designs," *IEEE Robot. Autom. Mag.*, vol. 16, no. 3, pp. 81–94, 2009.

[117] B. Tondu and S. D. Zagal, "McKibben artificial muscle can be in accordance with the Hill skeletal muscle model," in *The First IEEE/RAS-EMBS International Conference on Biomedical Robotics and Biomechatronics, 2006. BioRob 2006*, 2006, pp. 714–720.

[118] "Shadow Dexterous Hand E1 Series (E1M3R, E1M3L, E1P1R, E1P1L)," Shadow Robot Company, Technical Specifications, Jan. 2013.

[119] Techflex Inc., "TechFlex Solutions, Braided Sleeving Products," *A Versatile Bundling and Protection Solution*, 2013. [Online]. Available: http://www.techflex.com/prod_pet.asp. [Accessed: 28-Oct-2013].

[120] J. Murillo, "Design of a Pneumatic Artificial Muscle for Powered Lower Limb Prostheses," 2013.

[121] N. Tsagarakis and D. G. Caldwell, "Improved modelling and assessment of pneumatic muscle actuators," in *IEEE International Conference on Robotics and Automation, 2000. Proceedings. ICRA '00*, 2000, vol. 4, pp. 3641–3646 vol.4.

[122] C. R. Johnson and R. C. Pierce, "Expansible cover," US2238058 A, 15-Apr-1941.

[123] H. De Haven, "Tensioning device for producing a linear pull," US2483088 A, 27-Sep-1949.

[124] R. H. Gaylord, "Fluid actuated motor system and stroking device," US2844126 A, 22-Jul-1958.

[125] H. F. Schulte, *Characteristics of the braided fluid actuator*. [Ann Arbor], 1961.

[126] T. Takagi and Y. Sakaguchi, "Pneumatic actuator for manipulator," 4615260, 07-Oct-1986.

[127] Festo Inc., "FESTO Co. catalogue main page," *Catalogue - Fluidic Muscle*. [Online]. Available: http://www.festo.com/cat/en-ca_ca/products_010400. [Accessed: 27-Sep-2013].

[128] H. M. Paynter and J. M. Juarez, "Thermodynamic treatment of tug- amp;-twist technology. 2. Thermodynamic twistor design," in *1999 IEEE/ASME International Conference on Advanced Intelligent Mechatronics, 1999. Proceedings*, 1999, pp. 826–829.

[129] C.-P. Chou and B. Hannaford, "Static and dynamic characteristics of McKibben pneumatic artificial muscles," in *, 1994 IEEE International Conference on Robotics and Automation, 1994. Proceedings*, 1994, pp. 281–286 vol.1.

[130] C.-P. Chou and B. Hannaford, "Measurement and modeling of McKibben pneumatic artificial muscles," *IEEE Trans. Robot. Autom.*, vol. 12, no. 1, pp. 90–102, 1996.

[131] B. Tondu, "A dynamic model of the McKibben artificial muscle contraction," in *, IEEE/ASME International Conference on Advanced Intelligent Mechatronics '97. Final Program and Abstracts*, 1997, p. 64–.

[132] B. Tondu and P. Lopez, "Modeling and control of McKibben artificial muscle robot actuators," *IEEE Control Syst.*, vol. 20, no. 2, pp. 15–38, 2000.

[133] A. D. Kydoniefs, "Finite axisymmetric deformations of an initially cylindrical membrane reinforced with inextensible cords," *Q. J. Mech. Appl. Math.*, vol. 23, no. 4, pp. 481–488, 1970.

[134] D. M. Matsikoudi-Iliopoulou and G. Lianis, “Non linear elastic axisymmetric deformation of membranes with torsion,” *Acta Mech.*, vol. 42, no. 3–4, pp. 153–170, Sep. 1982.

[135] G. K. Klute and B. Hannaford, *Accounting for Elastic Energy Storage in McKibben Artificial Muscle Actuators*. 2000.

[136] T. C. Gasser and G. A. Holzapfel, “A rate-independent elastoplastic constitutive model for biological fiber-reinforced composites at finite strains: continuum basis, algorithmic formulation and finite element implementation,” *Comput. Mech.*, vol. 29, no. 4–5, pp. 340–360, Oct. 2002.

[137] S. Davis and D. G. Caldwell, “Braid Effects on Contractile Range and Friction Modeling in Pneumatic Muscle Actuators,” *Int. J. Robot. Res.*, vol. 25, no. 4, pp. 359–369, Apr. 2006.

[138] T. Vo-Minh, T. Tjahjowidodo, H. Ramon, and H. Van Brussel, “A New Approach to Modeling Hysteresis in a Pneumatic Artificial Muscle Using The Maxwell-Slip Model,” *IEEEASME Trans. Mechatron.*, vol. 16, no. 1, pp. 177–186, Feb. 2011.

[139] I. S. Godage, D. T. Branson, E. Guglielmino, and D. G. Caldwell, “Pneumatic muscle actuated continuum arms: Modelling and experimental assessment,” in *2012 IEEE International Conference on Robotics and Automation (ICRA)*, 2012, pp. 4980–4985.

[140] P. B. Petrovic, “Modeling and control of an artificial pneumatic muscle, Part one: model building,” presented at the The 10th Conference on Mechanical Vibrations, Timisoara, 2002, vol. 3.

[141] R. W. Colbrunn, G. M. Nelson, and R. D. Quinn, “Modeling of braided pneumatic actuators for robotic control,” in *2001 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2001. Proceedings*, 2001, vol. 4, pp. 1964–1970 vol.4.

[142] S. Davis, N. Tsagarakis, J. Canderle, and D. G. Caldwell, “Enhanced Modelling and Performance in Braided Pneumatic Muscle Actuators,” *Int. J. Robot. Res.*, vol. 22, no. 3–4, pp. 213–227, Mar. 2003.

[143] X. Shen and M. Goldfarb, “Energy Saving in Pneumatic Servo Control Utilizing Interchamber Cross-Flow,” *J. Dyn. Syst. Meas. Control*, vol. 129, no. 3, pp. 303–310, Oct. 2006.

[144] J.-H. Lee and K.-J. Kim, “Modeling of nonlinear complex stiffness of dual-chamber pneumatic spring for precision vibration isolations,” *J. Sound Vib.*, vol. 301, no. 3–5, pp. 909–926, Apr. 2007.

[145] M. Sorli and L. Gastaldi, “Thermic Influence on the Dynamics of Pneumatic Servosystems,” *J. Dyn. Syst. Meas. Control*, vol. 131, no. 2, pp. 024501–024501, Feb. 2009.

[146] “Overview of materials for Silicone Rubber,” *MatWeb Material Property Data*, 2015. [Online]. Available: http://www.matweb.com/search/DataSheet.aspx?MatGUID=cbe7a469897a47eda563816c86a73520. [Accessed: 19-Jan-2015].

[147] “Butyl Rubber (IIR, CIIR, BIIR),” *MatWeb Material Property Data*, 2015. [Online]. Available: http://www.matweb.com/search/DataSheet.aspx?MatGUID=754a0c3d7c194965a0b369792855bfe4. [Accessed: 19-Jan-2015].

[148] Plastic Products Inc., “PET(polyethylene terephalate),” *Plastic Products Inc. Just About Anything Plastic*, 2014. [Online]. Available: http://www.plastic-products.com/part12.htm. [Accessed: 07-May-2014].

[149] Quality Transmission Components, *Handbook of Metric Gears Q420 Product Guide & Technical Data*. 125 Railroad Avenue, Garden City Park, New York: Quality Transmission Gears, 2007.

[150] L. J. Hadcock, "Development of a Dynamic Biomechanical Model for Load Carriage: Phase III Part C1: Pressure and Force Distribution Measurement for the Design of Waist Belts in Personal Load Carriage Systems." Ergonomics Research Group Queen's University, 2005.

[151] "WalkOn AFO — Ottobock." [Online]. Available: http://www.ottobock.ca/orthotics/solution-overview/ankle-brace-walkon/. [Accessed: 30-Apr-2015].

[152] R. Baker, J. L. McGinley, M. H. Schwartz, S. Beynon, A. Rozumalski, H. K. Graham, and O. Tirosh, "The Gait Profile Score and Movement Analysis Profile," *Gait Posture*, vol. 30, no. 3, pp. 265–269, Oct. 2009.

[153] R. Baker, J. L. McGinley, M. Schwartz, P. Thomason, J. Rodda, and H. K. Graham, "The minimal clinically important difference for the Gait Profile Score," *Gait Posture*, vol. 35, no. 4, pp. 612–615, Apr. 2012.

[154] R. W. Bohannon, "Comfortable and maximum walking speed of adults aged 20—79 years: reference values and determinants," *Age Ageing*, vol. 26, no. 1, pp. 15–19, Jan. 1997.

[155] American Heart Association, "Understanding Blood Pressure Readings," *American Heart Association*. [Online]. Available: http://www.heart.org/HEARTORG/Conditions/HighBloodPressure/AboutHighBloodPressure/Understanding-Blood-Pressure-Readings_UCM_301764_Article.jsp.

GLOSSARY

Anatomical Position	Reference position of the body in anatomy. The body is standing up; feet shoulder width apart, with palms facing forward.
Abduction	Motion away from the body midline
Adduction	Motion toward the body midline
Anterior	Toward the front of the body
Concentric Contractions	Process where a muscle undergoes a reduction in length caused by the external force being smaller than the internal force.
Distal	Away from the center of the body
Eccentric Contractions	Process where a muscle undergoes an elongation caused by the external force being larger than the internal force.
Extension	The opposite of flexion
Flexion	A folding movement in which the anterior angle between two bones is decreased. It generally means that you are moving a bone closer to the body with respect to its anatomical position.
Frontal or Coronal Plane	Vertical plane crossing the body from the front. Equivalent to front view plane in engineering drawings
Inferior	Farther away from the head
Isometric Contractions	Process where a muscle does not change length due to the external force being equal to the internal force.
Lateral	Away from the midline of the body
Ligaments	Biological tissue linking bones and limit movements beyond a certain range
Medial	Toward the midline of the body
Posterior	Toward the back of the body
Proximal	Closer to the center of the body
Range of motion	Range of translation and rotation of a joint for each of its six degrees of freedom
Sagittal Plane	Vertical plane crossing the body from the side. Equivalent to side view plane in engineering drawings
Superior	Closer to the head
Tendons	Biological tissue linking muscles to bones that transfer contractile energy from the muscle to the bone.
Transverse Plane	Horizontal plane crossing the body. Equivalent to top view plane in engineering drawings

Appendix A. FORCE MODEL DERIVATION

*T*he traditional force model is derived following the methodology outlined by Doumit *et al* [21]. This method uses the thin-walled pressure vessel theory to relate the wall stresses fiber tension. This methodology assumes the wall thickness is at least a tenth of the diameter; the internal wall stresses are uniform through the wall thickness; and the material is isotropic.

Starting with the hoop stresses, Figure 0-1 shows the free body diagram for an infinitesimal cylindrical section, where *P* is the internal gauge pressure, *D* is the external diameter, and σ_{h1} is the hoop stress.

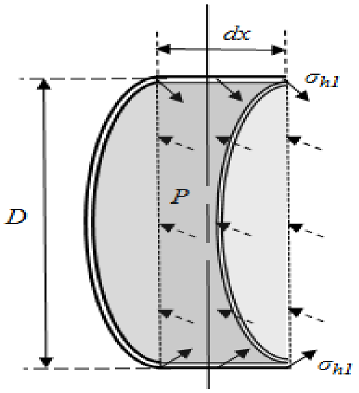


Figure 0-1: Infinitesimal cylindrical section with hoop stresses **[21]**

Performing a sum of forces and isolating for the hoop stress yields the following:

$$\sigma_{h1} = \frac{P\left(D - 2t_f - 2t_b\right)}{2(t_f + t_b)} \tag{0-1}$$

where t_f is the fibre thickness and t_b is the bladder thickness. Then for the longitudinal stress, Figure 0-2 shows a free body diagram for a cylindrical section in the longitudinal direction where σ_l is the hoop stress.

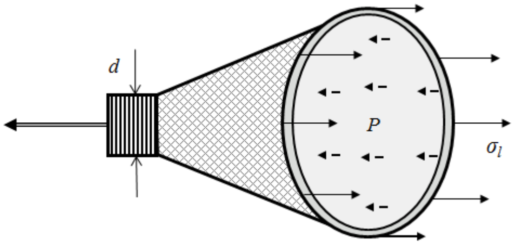


Figure 0-2: Frontal cylindrical section with longitudinal stresses **[21]**

Performing a sum of forces and isolating for the longitudinal stress yields the following:

$$\sigma_l = \frac{P\left(\frac{D}{2} - t_f - t_b\right)^2}{D(t_f + t_b)} \quad \textbf{(0-2)}$$

Once the internal bladder is pressurized, it tends to inflate radially until it is entirely in contact with the braided mesh. Assuming that the overall muscle force is caused by the interaction between the bladder and the fibers in the braided mesh, the tension in the fibre *T* is linked to the hoop force F_h and the longitudinal force F_l as shown in Figure 0-3.

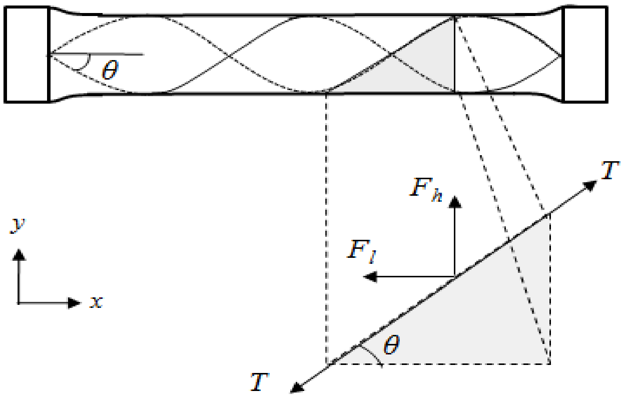


Figure 0-3: Fibre tension analysis [21]

$$F_h = nNT_h sin\theta \quad \textbf{(0-3)}$$

$$F_l = NT_l cos\theta \quad \textbf{(0-4)}$$

where T_h is the fibre tension caused by the hoop forces, T_l is the fibre tension caused by the longitudinal forces, *n* is the number of fiber revolution, and *N* is the number of fibres in the braided mesh. Using the stress definition, the hoop and longitudinal stresses can be linked to their force counterparts:

$$\sigma_{h1} = \frac{F_h}{A_h}, \sigma_l = \frac{F_l}{A_l} \quad \textbf{(0-5)}$$

where A_h is the hoop area and A_l is the longitudinal area. Combining (0-1) to (0-5), it is possible to obtain expressions for the fiber tensions caused by the internal pressure:

$$T_h = \frac{P\left(D - 2t_b - 2t_f\right)}{2nN\, sin\,\theta} L \quad \textbf{(0-6)}$$

$$T_l = \frac{P\left(\frac{D}{2} - t_f - t_b\right)^2}{ND\, cos\,\theta} \frac{L}{n} tan\,\theta \quad \textbf{(0-7)}$$

The overall fibre tension, T_f, is the directional sum of the fibre tensions:

$$\overrightarrow{T_f} = \overrightarrow{T_h} - \overrightarrow{T_l} \quad \textbf{(0-8)}$$

Then, an analytical relationship for the muscle force can be obtained:

$$F_m = NT_f \cos\theta \tag{0-9}$$

$$F_m = \frac{L\, tan\,\theta}{n}\left(\frac{P\left(\frac{D}{2} - t_b - t_f\right)}{(tan\,\theta)^2} - \frac{P\left(\frac{D}{2} - t_f - t_b\right)^2}{D}\right) \tag{0-10}$$

Up to date, the force model takes into account the bladder wall and braided mesh thicknesses, but neglects end effects. To obtain the traditional force model as referenced in Chapter 2, both thicknesses must be neglected. The force model simplifies to:

$$F_m = \frac{PDL}{2n\tan\theta} - \frac{PDL}{4n}tan\,\theta \tag{0-11}$$

Manipulating the above relationship to eliminate the braid angle and the muscle diameter, an expression that relies solely on the muscle length is obtained. This relationship is equivalent to the traditional force model (2-12) but uses the braid length and the muscle length as main variables.

$$F_m = -P\left[\frac{L_y^2 - 3L^2}{4\pi n^2}\right] \tag{0-12}$$

Appendix B. DECISION ANALYSIS MATRICES

Table 0-1: Ankle Dorsiflexion and Plantar flexion Method

Criteria	Weight	Hinge		Gear System		Saddle Joint		Ratchet		Ball and Socket		Total
		Score	W.S.	Score	W.S.	Score	W.S.	Score	W.S.	Score	W.S.	
Precision	11	18	198	23	253	18	198	23	253	18	198	100
Reaction Time	7	21	147	18	126	21	147	19	133	21	147	100
Size	11	23	253	17	187	20	220	18	198	22	242	100
Weight	10	22	220	18	180	20	200	19	190	21	210	100
Effectiveness/Efficiency	11	15	165	23	253	20	220	24	264	18	198	100
Complexity	11	22	242	17	187	21	231	19	209	21	231	100
Energy Consumption	9	22	198	17	153	21	189	19	171	21	189	100
Energy Recycling	8	14	112	17	136	22	176	25	200	22	176	100
Ease-of-use	11	18	198	22	242	19	209	21	231	20	220	100
Safety	11	20	220	19	209	21	231	19	209	21	231	100
Total	100		1953		1926		2021		2058		2042	

Table 0-2: Foot Flexion and Extension Method

Criteria	Weight	Hinge		Free Motion		Total
		Score	W.S.	Score	W.S.	
Accuracy	8	50	400	50	400	100
Reaction Time	8	40	320	60	480	100
Size	11	30	330	70	770	100
Weight	12	30	360	70	840	100
Effectiveness/Efficiency	13	45	585	55	715	100
Complexity	12	45	540	55	660	100
Ease-of-use	13	50	650	50	650	100
Safety	12	65	780	35	420	100
Total	100		4570		5430	

Table 0-3: Foot Permitted Movement

Criteria	Weight	Flex/Ext		No Movement		Total
		Score	W.S.	Score	W.S.	
Size	11	42	462	58	638	100
Weight	11	45	495	55	605	100
Effectiveness/Efficiency	12	57	684	43	516	100
Complexity	11	46	506	54	594	100
Metabolic rate	9	65	585	35	315	100
Ease-of-use	12	60	720	40	480	100
Safety	9	50	450	50	450	100
Comfort	10	55	550	45	450	100
Movement Aesthetics	7	52	364	48	336	100
Total	100		5296		4704	

Table 0-4: Attachment Decision Matrix

Criteria	Weight	None		Belt and Straps		Molded Orthoses		Rigid Support		Total
		Score	W.S.	Score	W.S.	Score	W.S.	Score	W.S.	
Shape	10	28	280	24	240	26	260	22	220	100
Weight	10	28	280	25	250	24	240	23	230	100
Effectiveness/Efficiency	12	13	156	30	360	32	384	25	300	100
Complexity	10	23	230	28	280	23	230	26	260	100
Strength	10	16	160	30	300	31	310	23	230	100
Friction	9	28	252	20	180	27	243	25	225	100
Ease-of-use	9	27	243	24	216	26	234	23	207	100
Adjustability	7	27	189	26	182	21	147	26	182	100
Safety	7	28	196	25	175	25	175	22	154	100
Comfort	8	29	232	24	192	26	208	21	168	100
Total	100		2434		2567		2615		2384	

Appendix C. EXPERIMENTAL PROPERTIES

This section contains data used to perform the simulations, corresponding to the specifications of the various components used for the experiments. For products bought off the shelf, the specifications come from the manufacturer's data sheet.

Table 0-5: Properties and Constants

Variable	Symbol	Units	Value
Gas Constant for air	R_{air}	J/kg*K	287.058
Gravity	g	m/s2	9.81
Polytropic Exponent	ng	-	1.00
Friction coefficient (Fibre-Fibre)	μ_{ff}	-	0.30
Friction coefficient (Fibre-Bladder)	μ_{fb}	-	0.30

Table 0-6 : Pneumatic Muscle Tested and Experimental Conditions

Source	Visual Inspection	Approx.	Set	Measured	Calculated	Specifications					
Variable	Braid	Bladder	Initial Pressure	Velocity	Rest Length	Pressurized Length	Braid Length	Fibre Cycles	Fibre Diameter	Rest Diameter	Fibre Number
Symbol	-	-	Pss	v	Lmax	Lmin or L0	Ly (θ≈20)	n	df	Drest	N
Units	-	-	(psi)	(mm/sec)	(cm)	(cm)	(cm)	(cycle)	(cm)	(cm)	-
n2.05_D19_L337_P276_MYB_BBR_V1.00	Black and Yellow	Rubber	40	1.00	33.7	23.9	35.9	2.05	0.0254	1.9	216
n2.05_D19_L337_P276_MYB_BBR_V6.72	Black and Yellow	Rubber	40	6.72	33.7	23.9	35.9	2.05	0.0254	1.9	216
n2.05_D19_L337_P276_MYB_BBR_V10.00	Black and Yellow	Rubber	40	10.00	33.7	23.9	35.9	2.05	0.0254	1.9	216
n2.00_D19_L328_P414_MYB_BBR_V1.00	Black and Yellow	Rubber	60	1.00	32.8	22.6	34.9	2.00	0.0254	1.9	216
n2.05_D19_L336_P207_MYB_BSR_V0.10	Black and Yellow	Silicone	30	0.10	33.6	25.5	35.8	2.05	0.0254	1.9	216
n2.05_D19_L336_P207_MYB_BSR_V1.00	Black and Yellow	Silicone	30	1.00	33.6	25.5	35.8	2.05	0.0254	1.9	216
n2.05_D19_L336_P207_MYB_BSR_V6.72	Black and Yellow	Silicone	30	6.72	33.6	25.5	35.8	2.05	0.0254	1.9	216
n2.05_D19_L336_P207_MYB_BSR_V10.00	Black and Yellow	Silicone	30	10.00	33.6	25.5	35.8	2.05	0.0254	1.9	216
n2.05_D19_L336_P414_MYB_BSR_V0.10	Black and Yellow	Silicone	60	0.10	33.6	25.5	35.8	2.05	0.0254	1.9	216
n2.05_D19_L336_P414_MYB_BSR_V1.00	Black and Yellow	Silicone	60	1.00	33.6	25.5	35.8	2.05	0.0254	1.9	216
n2.05_D19_L336_P414_MYB_BSR_V6.72	Black and Yellow	Silicone	60	6.72	33.6	25.5	35.8	2.05	0.0254	1.9	216
n2.05_D19_L336_P414_MYB_BSR_V10.00	Black and Yellow	Silicone	60	10.00	33.6	25.5	35.8	2.05	0.0254	1.9	216
n2.82_D13_L316_P276_MNB_BBR_V1.00	Neon Blue	Rubber	40	1.00	31.6	23.8	33.6	2.82	0.0254	1.3	120
n2.82_D13_L316_P276_MNB_BBR_V6.72	Neon Blue	Rubber	40	6.72	31.6	23.8	33.6	2.82	0.0254	1.3	120
n2.82_D13_L316_P276_MNB_BBR_V10.00	Neon Blue	Rubber	40	10.00	31.6	23.8	33.6	2.82	0.0254	1.3	120
n3.00_D13_L337_P414_MNB_BBR_V1.00	Neon Blue	Rubber	60	1.00	33.7	23.5	35.9	3.00	0.0254	1.3	120
n2.82_D13_L316_P414_MNB_BBR_V6.72	Neon Blue	Rubber	60	6.72	31.6	23	33.6	2.82	0.0254	1.3	120
n2.82_D13_L316_P414_MNB_BBR_V10.00	Neon Blue	Rubber	60	10.00	31.6	23	33.6	2.82	0.0254	1.3	120
n2.83_D13_L317_P207_MNB_BSR_V1.00	Neon Blue	Silicone	30	1.00	31.7	24.7	33.7	2.83	0.0254	1.3	120
n2.83_D13_L317_P207_MNB_BSR_V6.72	Neon Blue	Silicone	30	6.72	31.7	23.2	33.7	2.83	0.0254	1.3	120
n2.83_D13_L317_P207_MNB_BSR_V6.72	Neon Blue	Silicone	30	10.00	31.7	23.2	33.7	2.83	0.0254	1.3	120
n2.83_D13_L317_P207_MNB_BSR_V10.00	Neon Blue	Silicone	60	1.00	31.7	23	33.7	2.83	0.0254	1.3	120
n2.83_D13_L317_P414_MNB_BSR_V1.00	Neon Blue	Silicone	60	6.72	31.7	23	33.7	2.83	0.0254	1.3	120
n2.83_D13_L317_P414_MNB_BSR_V6.72	Neon Blue	Silicone	60	10.00	31.7	23	33.7	2.83	0.0254	1.3	120

Appendix D. GAIT PROPERTIES

This section presents the normalized gait dataset for the joint position, extracted from the Gait Profile Score.

Table 0-7: Gait Profile Score Normalized Dataset

Gait Progression (%)	Ankle Dorsiflexion/ Plantarflexion	Knee Flexion/ Extension	Hip Flexion/ Extension
0	-2.202	6.009	36.305
2	-3.989	9.119	36.134
4	-4.959	12.455	35.703
6	-4.564	15.508	34.892
8	-3.194	17.703	33.674
10	-1.379	18.908	32.039
12	0.485	19.152	30.029
14	2.086	18.558	27.739
16	3.413	17.363	25.250
18	4.475	15.894	22.686
20	5.327	14.305	20.122
22	6.007	12.660	17.594
24	6.641	11.085	15.126
26	7.239	9.654	12.743
28	7.847	8.328	10.417
30	8.436	7.143	8.175
32	9.039	6.145	6.009
34	9.639	5.350	3.935
36	10.244	4.795	1.942
38	10.868	4.542	0.070
40	11.479	4.691	-1.676
42	12.025	5.276	-3.285
44	12.308	6.370	-4.738
46	12.109	8.048	-5.946
48	11.051	10.393	-6.819
50	8.713	13.475	-7.208
52	4.813	17.389	-6.969
54	-0.637	22.121	-6.027
56	-6.872	27.684	-4.297
58	-12.656	33.916	-1.745
60	-16.439	40.571	1.566
62	-17.470	47.213	5.481
64	-15.922	53.226	9.747
66	-12.903	57.989	14.054
68	-9.356	61.171	18.237
70	-6.023	62.611	22.108
72	-3.289	62.361	25.612
74	-1.238	60.571	28.692
76	0.189	57.403	31.368
78	1.215	53.026	33.534
80	1.846	47.642	35.250
82	2.089	41.424	36.524

84	2.104	34.621	37.347
86	2.010	27.474	37.787
88	1.866	20.374	37.821
90	1.708	13.843	37.542
92	1.350	8.207	37.082
94	0.935	4.615	36.592
96	0.152	3.071	36.195
98	-1.100	3.617	36.005
100	-2.781	5.891	35.974

Appendix E. MATLAB CODE

This appendix provides the code used to identify the material and size the components of the exoskeleton.

```
function varargout = MAIN(varargin)
%This code was adapted from code developed for the 4th year CAD/CAM course
%in the Fall 2012 by Justin Leclair, Nicole Capela and Christopher Warren

%MAIN MATLAB code for MAIN.fig
%      MAIN, by itself, creates a new MAIN or raises the existing
%      singleton*.
%
%      H = MAIN returns the handle to a new MAIN or the handle to
%      the existing singleton*.
%
%      MAIN('CALLBACK',hObject,eventData,handles,...) calls the local
%      function named CALLBACK in MAIN.M with the given input arguments.
%
%      MAIN('Property','Value',...) creates a new MAIN or raises the
%      existing singleton*.  Starting from the left, property value pairs
are
%      applied to the GUI before MAIN_OpeningFcn gets called.  An
%      unrecognized property name or invalid value makes property
application
%      stop.  All inputs are passed to MAIN_OpeningFcn via varargin.
%
%      *See GUI Options on GUIDE's Tools menu.  Choose "GUI allows only
one
%      instance to run (singleton)".
%
% See also: GUIDE, GUIDATA, GUIHANDLES
% Edit the above text to modify the response to help MAIN
% Last Modified by GUIDE v2.5 21-Jan-2015 12:39:04
% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                   'gui_Singleton',  gui_Singleton, ...
                   'gui_OpeningFcn', @MAIN_OpeningFcn, ...
                   'gui_OutputFcn',  @MAIN_OutputFcn, ...
                   'gui_LayoutFcn',  [] , ...
                   'gui_Callback',   []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end

% End initialization code - DO NOT EDIT
% --- Executes just before MAIN is made visible.
function MAIN_OpeningFcn(hObject, eventdata, handles, varargin)
% This function has no output args, see OutputFcn.
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
% varargin   command line arguments to MAIN (see VARARGIN)

% Choose default command line output for MAIN
handles.output = hObject;

% Update handles structure
guidata(hObject, handles);

% UIWAIT makes MAIN wait for user response (see UIRESUME)
% uiwait(handles.figure1);
%Set the default values on the GUI. It is recommended to choose a valid
set
%of default values as a starting point when the program launches.
clc
Default_Height=
(round((((get(handles.SliderHeight,'Min')+get(handles.SliderHeight,'Max'))
)/2)*1000))/1000;
set(handles.SliderHeight,'Value',Default_Height);
set(handles.TXTHeight,'String',num2str(Default_Height));
Default_Weight=
(round((((get(handles.SliderWeight,'Min')+get(handles.SliderWeight,'Max'))
)/2)*1000))/1000;
set(handles.SliderWeight,'Value',Default_Weight);
set(handles.TXTWeight,'String',num2str(Default_Weight));
Default_Waist=
(round((((get(handles.SliderWaist,'Min')+get(handles.SliderWaist,'Max')))/
2)*1000))/1000;
set(handles.SliderWaist,'Value',Default_Waist);
set(handles.TXTWaist,'String',num2str(Default_Waist));
Default_HipWid=
(round((((get(handles.SliderHipWid,'Min')+get(handles.SliderHipWid,'Max'))
)/2)*1000))/1000;
set(handles.SliderHipWid,'Value',Default_HipWid);
set(handles.TXTHipWid,'String',num2str(Default_HipWid));
Default_HipKnee=
(round((((get(handles.SliderHipKnee,'Min')+get(handles.SliderHipKnee,'Max'
)))/2)*1000))/1000;
set(handles.SliderHipKnee,'Value',Default_HipKnee);
set(handles.TXTHipKnee,'String',num2str(Default_HipKnee));
Default_AnkleKnee=
(round((((get(handles.SliderAnkleKnee,'Min')+get(handles.SliderAnkleKnee,'
Max')))/2)*1000))/1000;
set(handles.SliderAnkleKnee,'Value',Default_AnkleKnee);
set(handles.TXTAnkleKnee,'String',num2str(Default_AnkleKnee));
Default_AnkleHeight=
(round((((get(handles.SliderAnkleHeight,'Min')+get(handles.SliderAnkleHeig
ht,'Max')))/2)*1000))/1000;
set(handles.SliderAnkleHeight,'Value',Default_AnkleHeight);
set(handles.TXTAnkleHeight,'String',num2str(Default_HipKnee));
```

```
Default_FootLength=
(round((((get(handles.SliderFootLength,'Min')+get(handles.SliderFootLength
,'Max')))/2)*1000))/1000;
set(handles.SliderFootLength,'Value',Default_FootLength);
set(handles.TXTFootLength,'String',num2str(Default_FootLength));

%Set the window title with the group identification:
set(handles.figure1,'Name','Justin Leclair Masters Thesis');

%Add the 'subfunctions' folder to the path so that subfunctions can be
%accessed
addpath('Subfunctions');

% --- Outputs from this function are returned to the command line.
function varargout = MAIN_OutputFcn(hObject, eventdata, handles)
% varargout  cell array for returning output args (see VARARGOUT);
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Get default command line output from handles structure
varargout{1} = handles.output;

% --- Executes on button press in BTN_Generate.
function BTN_Generate_Callback(hObject, eventdata, handles)
% hObject    handle to BTN_Generate (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

if(isempty(handles))
    Wrong_File();
else
    %Get the design parameters from the interface (DO NOT PERFORM ANY
DESIGN CALCULATIONS HERE)
    Height = get(handles.SliderHeight,'Value');
    Weight = get(handles.SliderWeight,'Value');
    Waist = get(handles.SliderWaist,'Value');
    HipWid = get(handles.SliderHipWid,'Value');
    HipKnee = get(handles.SliderHipKnee,'Value');
    AnkleKnee = get(handles.SliderAnkleKnee, 'Value');
    AnkleHeight = get(handles.SliderAnkleHeight, 'Value');
    FootLength = get(handles.SliderFootLength, 'Value');

    %The design calculations are done within this function.

FILEWRITE3(Height,Weight,Waist,HipWid,HipKnee,AnkleKnee,AnkleHeight,FootLe
ngth);

    %Show the results on the GUI.
    log_file = 'E:\Maitrise\Thesis\Exoskeleton\Conceptual
Design\Matlab\Log\Thesis_Log.TXT';
    fid = fopen(log_file,'r'); %Open the log file for reading
    S=char(fread(fid)'); %Read the file into a string
    fclose(fid);
    set(handles.TXT_log,'String',S); %write the string into the textbox
    set(handles.TXT_path,'String',log_file); %show the path of the log
file
    set(handles.TXT_path,'Visible','on');
end


% --- Executes on button press in BTN_Finish.
function BTN_Finish_Callback(hObject, eventdata, handles)
% hObject    handle to BTN_Finish (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
close gcf

function TXT_log_Callback(hObject, eventdata, handles)
% hObject    handle to TXT_log (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of TXT_log as text
%        str2double(get(hObject,'String')) returns contents of TXT_log as
a double

% --- Executes during object creation, after setting all properties.
function TXT_log_CreateFcn(hObject, eventdata, handles)
% hObject    handle to TXT_log (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% --- Executes on slider movement.
function SliderHeight_Callback(hObject, eventdata, handles)
% hObject    handle to SliderHeight (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'Value') returns position of slider
%        get(hObject,'Min') and get(hObject,'Max') to determine range of
slider
if(isempty(handles))
    Wrong_File();
else
    value = (get(hObject,'Value'))*1000; %Round the value to the nearest
integer
    set(handles.TXTHeight,'String',num2str((round(value))/1000));
end

% --- Executes during object creation, after setting all properties.
function SliderHeight_CreateFcn(hObject, eventdata, handles)
```

```
% hObject    handle to SliderHeight (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: slider controls usually have a light gray background.
if isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor',[.9 .9 .9]);
end

% --- Gives out a message that the GUI should not be executed directly
from
% the .fig file. The user should run the .m file instead.
function Wrong_File()
clc
h = msgbox('You cannot run the MAIN.fig file directly. Please run the
program from the Main.m file directly.','Cannot run the
figure...','error','modal');
uiwait(h);
disp('You must run the MAIN.m file. Not the MAIN.fig file.');
disp('To run the MAIN.m file, open it in the editor and press ');
disp('the green "PLAY" button, or press "F5" on the keyboard.');
close gcf

function TXTHeight_Callback(hObject, eventdata, handles)
% hObject    handle to TXTHeight (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of TXTHeight as text
%        str2double(get(hObject,'String')) returns contents of TXTHeight
as a double

if(isempty(handles))
    Wrong_File();
else
    value = round(str2double(get(hObject,'String')));

    %Apply basic testing to see if the value does not exceed the range of
the
    %slider (defined in the gui)
    if(value<get(handles.SliderHeight,'Min'))
        value = get(handles.SliderHeight,'Min');
    end
    if(value>get(handles.SliderHeight,'Max'))
        value = get(handles.SliderHeight,'Max');
    end
    set(hObject,'String',value);
    set(handles.SliderHeight,'Value',value);
end

% --- Executes during object creation, after setting all properties.
function TXTHeight_CreateFcn(hObject, eventdata, handles)
% hObject    handle to TXTHeight (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles   empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% --- Executes on slider movement.
function SliderWeight_Callback(hObject, eventdata, handles)
% hObject    handle to SliderWeight (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'Value') returns position of slider
%        get(hObject,'Min') and get(hObject,'Max') to determine range of
slider
if(isempty(handles))
    Wrong_File();
else
    value = (get(hObject,'Value'))*10; %Round the value to the nearest
integer
    set(handles.TXTWeight,'String',num2str((round(value))/10));
end

% --- Executes during object creation, after setting all properties.
function SliderWeight_CreateFcn(hObject, eventdata, handles)
% hObject    handle to SliderWeight (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: slider controls usually have a light gray background.
if isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor',[.9 .9 .9]);
end

function TXTWeight_Callback(hObject, eventdata, handles)
% hObject    handle to TXTWeight (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of TXTWeight as text
%        str2double(get(hObject,'String')) returns contents of TXTWeight
as a double
if(isempty(handles))
    Wrong_File();
else
    value = round(str2double(get(hObject,'String')));

    %Apply basic testing to see if the value does not exceed the range of
the
    %slider (defined in the gui)
    if(value<get(handles.SliderWeight,'Min'))
```

```
        value = get(handles.SliderWeight,'Min');
    end
    if(value>get(handles.SliderWeight,'Max'))
        value = get(handles.SliderWeight,'Max');
    end
    set(hObject,'String',value);
    set(handles.SliderWeight,'Value',value);
end

% --- Executes during object creation, after setting all properties.
function TXTWeight_CreateFcn(hObject, eventdata, handles)
% hObject    handle to TXTWeight (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% --- Executes on slider movement.
function SliderWaist_Callback(hObject, eventdata, handles)
% hObject    handle to SliderWaist (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'Value') returns position of slider
%        get(hObject,'Min') and get(hObject,'Max') to determine range of
slider
if(isempty(handles))
    Wrong_File();
else
    value = (get(hObject,'Value'))*1000; %Round the value to the nearest
integer
    set(handles.TXTWaist,'String',num2str((round(value))/1000));
end

% --- Executes during object creation, after setting all properties.
function SliderWaist_CreateFcn(hObject, eventdata, handles)
% hObject    handle to SliderWaist (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: slider controls usually have a light gray background.
if isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor',[.9 .9 .9]);
end

function TXTWaist_Callback(hObject, eventdata, handles)
% hObject    handle to TXTWaist (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
% Hints: get(hObject,'String') returns contents of TXTWaist as text
%        str2double(get(hObject,'String')) returns contents of TXTWaist as
a double
if(isempty(handles))
    Wrong_File();
else
    value = round(str2double(get(hObject,'String')));

    %Apply basic testing to see if the value does not exceed the range of
the
    %slider (defined in the gui)
    if(value<get(handles.SliderWaist,'Min'))
        value = get(handles.SliderWaist,'Min');
    end
    if(value>get(handles.SliderWaist,'Max'))
        value = get(handles.SliderWaist,'Max');
    end
    set(hObject,'String',value);
    set(handles.SliderWaist,'Value',value);
end

% --- Executes during object creation, after setting all properties.
function TXTWaist_CreateFcn(hObject, eventdata, handles)
% hObject    handle to TXTWaist (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% --- Executes on slider movement.
function SliderHipWid_Callback(hObject, eventdata, handles)
% hObject    handle to SliderHipWid (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'Value') returns position of slider
%        get(hObject,'Min') and get(hObject,'Max') to determine range of
slider
if(isempty(handles))
    Wrong_File();
else
    value = (get(hObject,'Value'))*1000; %Round the value to the nearest
integer
    set(handles.TXTHipWid,'String',num2str((round(value))/1000));
end

% --- Executes during object creation, after setting all properties.
function SliderHipWid_CreateFcn(hObject, eventdata, handles)
% hObject    handle to SliderHipWid (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: slider controls usually have a light gray background.
if isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor',[.9 .9 .9]);
end

function TXTHipWid_Callback(hObject, eventdata, handles)
% hObject    handle to TXTHipWid (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of TXTHipWid as text
%        str2double(get(hObject,'String')) returns contents of TXTHipWid
as a double
if(isempty(handles))
    Wrong_File();
else
    value = round(str2double(get(hObject,'String')));

    %Apply basic testing to see if the value does not exceed the range of
the
    %slider (defined in the gui)
    if(value<get(handles.SliderHipWid,'Min'))
        value = get(handles.SliderHipWid,'Min');
    end
    if(value>get(handles.SliderHipWid,'Max'))
        value = get(handles.SliderHipWid,'Max');
    end
    set(hObject,'String',value);
    set(handles.SliderHipWid,'Value',value);
end

% --- Executes during object creation, after setting all properties.
function TXTHipWid_CreateFcn(hObject, eventdata, handles)
% hObject    handle to TXTHipWid (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function TXTHipKnee_Callback(hObject, eventdata, handles)
% hObject    handle to TXTHipKnee (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of TXTHipKnee as text
%        str2double(get(hObject,'String')) returns contents of TXTHipKnee
as a double
if(isempty(handles))
    Wrong_File();
else
    value = round(str2double(get(hObject,'String')));

    %Apply basic testing to see if the value does not exceed the range of
the
    %slider (defined in the gui)
    if(value<get(handles.SliderHipKnee,'Min'))
        value = get(handles.SliderHipKnee,'Min');
    end
    if(value>get(handles.SliderHipKnee,'Max'))
        value = get(handles.SliderHipKnee,'Max');
    end
    set(hObject,'String',value);
    set(handles.SliderHipKnee,'Value',value);
end

% --- Executes during object creation, after setting all properties.
function TXTHipKnee_CreateFcn(hObject, eventdata, handles)
% hObject    handle to TXTHipKnee (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% --- Executes on slider movement.
function SliderHipKnee_Callback(hObject, eventdata, handles)
% hObject    handle to SliderHipKnee (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'Value') returns position of slider
%        get(hObject,'Min') and get(hObject,'Max') to determine range of
slider
if(isempty(handles))
    Wrong_File();
else
    value = (get(hObject,'Value'))*1000; %Round the value to the nearest
integer
    set(handles.TXTHipKnee,'String',num2str((round(value))/1000));
end

% --- Executes during object creation, after setting all properties.
function SliderHipKnee_CreateFcn(hObject, eventdata, handles)
% hObject    handle to SliderHipKnee (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called
% Hint: slider controls usually have a light gray background.
if isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor',[.9 .9 .9]);
end

function TXTAnkleKnee_Callback(hObject, eventdata, handles)
% hObject    handle to TXTAnkleKnee (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of TXTAnkleKnee as text
%        str2double(get(hObject,'String')) returns contents of
TXTAnkleKnee as a double
if(isempty(handles))
    Wrong_File();
else
    value = round(str2double(get(hObject,'String')));

    %Apply basic testing to see if the value does not exceed the range of
the
    %slider (defined in the gui)
    if(value<get(handles.SliderAnkleKnee,'Min'))
        value = get(handles.SliderAnkleKnee,'Min');
    end
    if(value>get(handles.SliderAnkleKnee,'Max'))
        value = get(handles.SliderAnkleKnee,'Max');
    end
    set(hObject,'String',value);
    set(handles.SliderAnkleKnee,'Value',value);
end

% --- Executes during object creation, after setting all properties.
function TXTAnkleKnee_CreateFcn(hObject, eventdata, handles)
% hObject    handle to TXTAnkleKnee (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% --- Executes on slider movement.
function SliderAnkleKnee_Callback(hObject, eventdata, handles)
% hObject    handle to SliderAnkleKnee (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'Value') returns position of slider
%        get(hObject,'Min') and get(hObject,'Max') to determine range of
slider
```

```
if(isempty(handles))
    Wrong_File();
else
    value = (get(hObject,'Value'))*1000; %Round the value to the nearest
integer
    set(handles.TXTAnkleKnee,'String',num2str((round(value))/1000));
end

% --- Executes during object creation, after setting all properties.
function SliderAnkleKnee_CreateFcn(hObject, eventdata, handles)
% hObject    handle to SliderAnkleKnee (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: slider controls usually have a light gray background.
if isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor',[.9 .9 .9]);
end

function TXTAnkleHeight_Callback(hObject, eventdata, handles)
% hObject    handle to TXTAnkleHeight (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of TXTAnkleHeight as text
%        str2double(get(hObject,'String')) returns contents of
TXTAnkleHeight as a double
if(isempty(handles))
    Wrong_File();
else
    value = round(str2double(get(hObject,'String')));

    %Apply basic testing to see if the value does not exceed the range of
the
    %slider (defined in the gui)
    if(value<get(handles.SliderAnkleHeight,'Min'))
        value = get(handles.SliderAnkleHeight,'Min');
    end
    if(value>get(handles.SliderAnkleHeight,'Max'))
        value = get(handles.SliderAnkleHeight,'Max');
    end
    set(hObject,'String',value);
    set(handles.SliderAnkleHeight,'Value',value);
end

% --- Executes during object creation, after setting all properties.
function TXTAnkleHeight_CreateFcn(hObject, eventdata, handles)
% hObject    handle to TXTAnkleHeight (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
```

```
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% --- Executes on slider movement.
function SliderAnkleHeight_Callback(hObject, eventdata, handles)
% hObject    handle to SliderAnkleHeight (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'Value') returns position of slider
%        get(hObject,'Min') and get(hObject,'Max') to determine range of
slider
if(isempty(handles))
    Wrong_File();
else
    value = (get(hObject,'Value'))*1000; %Round the value to the nearest
integer
    set(handles.TXTAnkleHeight,'String',num2str((round(value))/1000));
end

% --- Executes during object creation, after setting all properties.
function SliderAnkleHeight_CreateFcn(hObject, eventdata, handles)
% hObject    handle to SliderAnkleHeight (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: slider controls usually have a light gray background.
if isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor',[.9 .9 .9]);
end

function TXTFootLength_Callback(hObject, eventdata, handles)
% hObject    handle to TXTFootLength (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of TXTFootLength as text
%        str2double(get(hObject,'String')) returns contents of
TXTFootLength as a double
if(isempty(handles))
    Wrong_File();
else
    value = round(str2double(get(hObject,'String')));

    %Apply basic testing to see if the value does not exceed the range of
the
    %slider (defined in the gui)
    if(value<get(handles.SliderFootLength,'Min'))
        value = get(handles.SliderFootLength,'Min');
    end
    if(value>get(handles.SliderFootLength,'Max'))
        value = get(handles.SliderFootLength,'Max');
    end
    set(hObject,'String',value);
    set(handles.SliderFootLength,'Value',value);
end

% --- Executes during object creation, after setting all properties.
function TXTFootLength_CreateFcn(hObject, eventdata, handles)
% hObject    handle to TXTFootLength (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% --- Executes on slider movement.
function SliderFootLength_Callback(hObject, eventdata, handles)
% hObject    handle to SliderFootLength (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'Value') returns position of slider
%        get(hObject,'Min') and get(hObject,'Max') to determine range of
slider
if(isempty(handles))
    Wrong_File();
else
    value = (get(hObject,'Value'))*1000; %Round the value to the nearest
integer
    set(handles.TXTFootLength,'String',num2str((round(value))/1000));
end

% --- Executes during object creation, after setting all properties.
function SliderFootLengtht_CreateFcn(hObject, eventdata, handles)
% hObject    handle to SliderFootLength (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: slider controls usually have a light gray background.
if isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor',[.9 .9 .9]);
end

% --- Executes on button press in pushbutton5.
function pushbutton5_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton5 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
MinHeight = get(handles.SliderHeight,'Min');
set(handles.SliderHeight,'Value',MinHeight);
set(handles.TXTHeight,'String',num2str(MinHeight));
```

```
MinWeight = get(handles.SliderWeight,'Min');
set(handles.SliderWeight,'Value',MinWeight);
set(handles.TXTWeight,'String',num2str(MinWeight));
MinWaist = get(handles.SliderWaist,'Min');
set(handles.SliderWaist,'Value',MinWaist);
set(handles.TXTWaist,'String',num2str(MinWaist));
MinHipWid = get(handles.SliderHipWid,'Min');
set(handles.SliderHipWid,'Value',MinHipWid);
set(handles.TXTHipWid,'String',num2str(MinHipWid));
MinHipKnee = get(handles.SliderHipKnee,'Min');
set(handles.SliderHipKnee,'Value',MinHipKnee);
set(handles.TXTHipKnee,'String',num2str(MinHipKnee));
MinAnkleKnee = get(handles.SliderAnkleKneee,'Min');
set(handles.SliderAnkleKnee,'Value',MinAnkleKnee);
set(handles.TXTAnkleKnee,'String',num2str(MinAnkleKnee));
MinAnkleHeight = get(handles.SliderAnkleHeight,'Min');
set(handles.SliderAnkleHeight,'Value',MinAnkleHeight);
set(handles.TXTAnkleHeight,'String',num2str(MinAnkleHeight));
MinFootLength = get(handles.SliderFootLength,'Min');
set(handles.SliderFootLength,'Value',MinFootLength);
set(handles.TXTFootLength,'String',num2str(MinFootLength));

% --- Executes on button press in pushbutton6.
function pushbutton6_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton6 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
MaxHeight = get(handles.SliderHeight,'Max');
set(handles.SliderHeight,'Value',MaxHeight);
set(handles.TXTHeight,'String',num2str(MaxHeight));
MaxWeight = get(handles.SliderWeight,'Max');
set(handles.SliderWeight,'Value',MaxWeight);
set(handles.TXTWeight,'String',num2str(MaxWeight));
MaxWaist = get(handles.SliderWaist,'Max');
set(handles.SliderWaist,'Value',MaxWaist);
set(handles.TXTWaist,'String',num2str(MaxWaist));
MaxHipWid = get(handles.SliderHipWid,'Max');
set(handles.SliderHipWid,'Value',MaxHipWid);
set(handles.TXTHipWid,'String',num2str(MaxHipWid));
MaxHipKnee = get(handles.SliderHipKnee,'Max');
set(handles.SliderHipKnee,'Value',MaxHipKnee);
set(handles.TXTHipKnee,'String',num2str(MaxHipKnee));
MaxAnkleKnee = get(handles.SliderAnkleKnee,'Max');
set(handles.SliderAnkleKnee,'Value',MaxAnkleKnee);
set(handles.TXTAnkleKnee,'String',num2str(MaxAnkleKnee));
MaxAnkleHeight = get(handles.SliderAnkleHeight,'Max');
set(handles.SliderAnkleHeight,'Value',MaxAnkleHeight);
set(handles.TXTAnkleHeight,'String',num2str(MaxAnkleHeight));
MaxFootLength = get(handles.SliderFootLength,'Max');
set(handles.SliderFootLength,'Value',MaxFootLength);
set(handles.TXTFootLength,'String',num2str(MaxFootLength));

% --- Executes on button press in pushbutton8.
function pushbutton8_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton8 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
randminHeight = get(handles.SliderHeight,'Min')*1000;
randmaxHeight = get(handles.SliderHeight,'Max')*1000;
RandHeight = (round((randminHeight) + (randmaxHeight-
randminHeight).*rand(1,1)))/1000;
set(handles.SliderHeight,'Value',RandHeight);
set(handles.TXTHeight,'String',num2str(RandHeight));
randminWeight = get(handles.SliderWeight,'Min')*10;
randmaxWeight = get(handles.SliderWeight,'Max')*10;
RandWeight = (round((randminWeight) + (randmaxWeight-
randminWeight).*rand(1,1)))/10;
set(handles.SliderWeight,'Value',RandWeight);
set(handles.TXTWeight,'String',num2str(RandWeight));
randminWaist = get(handles.SliderWaist,'Min')*1000;
randmaxWaist = get(handles.SliderWaist,'Max')*1000;
RandWaist = (round((randminWaist) + (randmaxWaist-
randminWaist).*rand(1,1)))/1000;
set(handles.SliderWaist,'Value',RandWaist);
set(handles.TXTWaist,'String',num2str(RandWaist));
randminHipWid = get(handles.SliderHipWid,'Min')*1000;
randmaxHipWid = get(handles.SliderHipWid,'Max')*1000;
RandHipWid = (round((randminHipWid) + (randmaxHipWid-
randminHipWid).*rand(1,1)))/1000;
set(handles.SliderHipWid,'Value',RandHipWid);
set(handles.TXTHipWid,'String',num2str(RandHipWid));
randminHipKnee = get(handles.SliderHipKnee,'Min')*1000;
randmaxHipKnee = get(handles.SliderHipKnee,'Max')*1000;
RandHipKnee = (round((randminHipKnee) + (randmaxHipKnee-
randminHipKnee).*rand(1,1)))/1000;
set(handles.SliderHipKnee,'Value',RandHipKnee);
set(handles.TXTHipKnee,'String',num2str(RandHipKnee));
randminAnkleKnee = get(handles.SliderAnkleKnee,'Min')*1000;
randmaxAnkleKnee = get(handles.SliderAnkleKnee,'Max')*1000;
RandAnkleKnee = (round((randminAnkleKnee) + (randmaxAnkleKnee-
randminAnkleKnee).*rand(1,1)))/1000;
set(handles.SliderAnkleKnee,'Value',RandAnkleKnee);
set(handles.TXTAnkleKnee,'String',num2str(RandAnkleKnee));
randminAnkleHeight = get(handles.SliderAnkleHeight,'Min')*1000;
randmaxAnkleHeight = get(handles.SliderAnkleHeight,'Max')*1000;
RandAnkleHeight = (round((randminAnkleHeight) + (randmaxAnkleHeight-
randminAnkleHeight).*rand(1,1)))/1000;
set(handles.SliderAnkleHeight,'Value',RandAnkleHeight);
set(handles.TXTAnkleHeight,'String',num2str(RandAnkleHeight));
randminFootLength = get(handles.SliderFootLength,'Min')*1000;
randmaxFootLength = get(handles.SliderFootLength,'Max')*1000;
RandFootLength = (round((randminFootLength) + (randmaxFootLength-
randminFootLength).*rand(1,1)))/1000;
set(handles.SliderFootLength,'Value',RandFootLength);
set(handles.TXTFootLength,'String',num2str(RandFootLength));

% --- Executes during object creation, after setting all properties.
function UserInputs_CreateFcn(hObject, eventdata, handles)
% hObject    handle to UserInputs (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called
% Hint: place code in OpeningFcn to populate UserInputs
imshow('E:\Maitrise\Thesis\Exoskeleton\Conceptual
Design\Matlab\Images\UserInputs.png');

% --- Executes on mouse press over axes background.
function UserInputs_ButtonDownFcn(hObject, eventdata, handles)
% hObject    handle to UserInputs (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hint: place code in OpeningFcn to populate UserInputs
imshow('E:\Maitrise\Thesis\Exoskeleton\Conceptual
Design\Matlab\Images\UserInputs.png');
```

```
function FILEWRITE3(H,BW,~,~,Hthigh,Hshank,Hankle,Lfoot)
%This function writes the variables to the text files, split by component
clc

%% Initial Values
H = 1.83;
BW = 80;

%% Anthropometric dimensions
[Mhat What Hhat Lcomhiphat Lcomhat HipCirc Mthigh Wthigh ThighHeight
Lcomkneethigh Lcomthigh ThighCirc Mshank Wshank ShankHeight Lcomankleshank
Lcomshank ShankCirc Mfoot Wfoot FootLength Lcomgroundankle Lcomfoot
FootWidth AnkleCirc AnkleHeight] = Anthrodata(H,BW);
ShankDiameter = ShankCirc/pi;

%% Gait Data
[GaitCycle, AnkleAngle, KneeAngle, HipAngle, AnkleAngularVel,
KneeAngularVel, HipAngularVel, AnkleAngularAcc, KneeAngularAcc,
HipAngularAcc] = GaitData(H, ThighHeight,ShankHeight,AnkleHeight);

%% Initial Values
% Shank Dimensions
ShankSideWidth = (1/8)*25.4/1000;
ShankBPMAnchorDiameter = 5/1000;
ShankBPMAnchorHeight = 1/100;
ShankBPMAnchorWidth = 1/100;
ShankBPMAnchorDepth = 1/100;
ShankLowHeight = ShankHeight/5;
ShankHighHeight = ShankHeight/2;
ShankSupportHeight = ShankHeight/10;
ShankCalfSupportRadius = ShankDiameter/2;

% Base Dimensions
% BaseWidth = FootWidth;
BaseWidth = (4-2*(1/4))*25.4/1000;
BaseBackWidth = FootWidth/8;
BaseAnkleRotationLength = FootLength/5;
BaseBackHeight = AnkleHeight*2/3;
BasePlateHeight = (1/8)*25.4/1000;

% Casing Dimensions
```

```
CasingThickness = (1/16)*25.4/1000;
Support = (1/16)*25.4/1000;

% Ratchet Dimensions
RatchetTeethNumber = 100;
RatchetHoleDiameter = 12/1000;
RatchetDiameter = 66.6/1000;
RatchetWidth = 6/1000;
RatchetTeethDepth = 1/1000;
RatchetTeethPresentAngle = 9*(360/100);
CenterDistance = 43.9/1000;

% Pawl Dimensions
PawlRadius = 8/1000;
PawlHoleDiameter = 5/1000;
PawlLength = 38/1000;
PawlWidth = 6/1000;
PawlFilet = 2.5/1000;

%Drive Dimensions
DriveSphericalDiameter = 5/1000;
DriveCamPushAngle = 30;
Spacing = RatchetTeethDepth*2;

for i=1:2
%% Calculated Values
%Pawl Hinge
PawlHingeDiameter = PawlHoleDiameter/3;

%Cam Dimension
CamWidth = PawlWidth;

%Slider Dimensions
SliderDiameter = CamWidth*2;

%Drive Dimensions
DriveShaftDiameter = DriveSphericalDiameter*2/3;
DriveCamSpacing = Spacing;
DriveCamBaseSpacing = DriveCamSpacing;
DriveCamDiameter = SliderDiameter*3/2;
DriveSlotWidth = (PawlLength-PawlRadius)/2 - (((PawlLength-
PawlRadius)/2)^2 - (7*Spacing)^2)^0.5;
DriveExtDiameter = DriveSphericalDiameter*1.25;

DriveCamAngleHeight =
0.5*((DriveCamDiameter/2)*(pi/2)/(tan(pi/6)+tan(pi/3)) - DriveCamSpacing);
%DriveCamAngleHeight = DriveCamSpacing/3;
DriveCamHeight = (DriveCamSpacing + DriveCamAngleHeight)*2;

%Slider Dimensions
SliderAngleHeight = DriveCamSpacing/2;

% Casing
CasingDriveDepth = DriveCamDiameter*3/4;
```

```
% Cam Dimensions
CamMinRadius = RatchetHoleDiameter*3/4;
CamMaxRadius = CamMinRadius + Spacing;

% Ankle Shaft Dimensions
DaRatchet = RatchetHoleDiameter;
DaSupport = DaRatchet*1.2;
    % Ankle Shaft Axial Bearings/Bushing
    [DaAxial, DaAxialOut, WaAxialBearing] = BushingSizing(DaSupport*1.2);
    WaAxialBearing = (1/8)*25.4/1000 - Support;
DaShank = DaAxial*1.2;
DaLarge = DaShank*1.3;
DaFree = DaRatchet;
DaSmall = DaFree/1.2;

% Base Dimensions
BaseWallThickness = 8/1000;
BaseConnectionWidth = ShankSideWidth + WaAxialBearing + Support;
BasePawlTop = PawlLength*sin(30*pi/180);
%BasePawlBack = PawlLength*sin(20*pi/180);
BasePawlBack = BaseAnkleRotationLength + BaseConnectionWidth -
(PawlLength-PawlRadius);
BasePawlLength = PawlLength - PawlRadius;
BaseRatchetForward = (RatchetDiameter/2)*1.05;
BasePawlHeight = RatchetDiameter/2;
BaseLowHeight = BaseBackHeight/3;
BaseAnkleSupportHeight = AnkleHeight + BasePawlHeight + BasePawlTop;
BaseBPMAnchorWidth = BaseWallThickness*2;
BaseBPMAnchorHeight = BaseBackHeight/2;
BaseOutterWidth = BaseConnectionWidth - ShankSideWidth;

% Shank Dimensions
ShankHalfWidth = BaseWidth/2;
ShankLowDepth = 2*DaShank;
ShankSupportWidth = (0.237*25.4/1000);
ShankSupportDepth = BaseWidth/2 - ShankSupportWidth/2;
ShankHighDepth = ShankLowDepth;

% Casing Dimensions
CasingWidth = (4)*25.4/1000 + BaseConnectionWidth;
CasingHeight = BasePawlTop + BasePawlHeight + RatchetDiameter/4;
CasingRatchetPosition = BasePawlBack + BasePawlLength;
CasingDriveHeight = DriveCamHeight;
CasingDriveWidth = 2*CasingDriveDepth;
CasingQuarterRatchetDiameter = RatchetDiameter/4;
CasingShaftSupport = 3*DriveShaftDiameter;
CasingShaftSupportHeight = DriveShaftDiameter;

% Ratchet Dimensions
RatchetInnerRadius = RatchetDiameter/2 - RatchetTeethDepth;

%Slider Dimension
CasingDrivePosition = RatchetDiameter/4 + RatchetDiameter/16;
SliderLength = DaSmall/2 + (CasingDrivePosition-RatchetDiameter/4) +
CasingDriveHeight*7/4 - (DriveCamAngleHeight+DriveCamSpacing)/2 -
DriveCamHeight/2 - SliderDiameter/2;
```

```
DriveShaftLength = (RatchetDiameter/2+RatchetDiameter/4) -
(CasingDrivePosition+CasingDriveHeight*7/4 -
(DriveCamAngleHeight+DriveCamSpacing) + DriveCamHeight/2);

%Casing Dimension
CasingShaftSupportPosition = DriveShaftLength*3/4 -
DriveSphericalDiameter/2;
SpringLengthMax = CasingShaftSupportPosition + Spacing;
SpringLengthMin = CasingShaftSupportPosition;

%Ankle BPM Holder
AnkleBPMHeight = ShankHeight/5;
AnkleBPMBackwards = FootLength/5;
AnkleBPMWidth = BaseConnectionWidth;
AnkleBPMDepth = BaseBackWidth;
AnkleBPMFlatHeight = BaseBPMAnchorHeight;
AnkleBPMDiameter = ShankBPMAnchorDiameter;

%% Belt Forces
%Belt forces at the mid shank
[Fnshank,Fvshank,Hbeltshank,Tbeltshank] =
Belts(ShankCirc,ShankSupportHeight,ShankHeight,2);
%Forces in the local reference frame
Fbshank = [-ones(length(AnkleAngle),1)*Fnshank Fvshank*sin(KneeAngle-
HipAngle) -Fvshank*cos(KneeAngle-HipAngle)];
% fprintf('Belt force analysis completed \n');

%% Forces from BPM
[Fbpm, ktotal, Do, Lo, Bladder, Strand, Rabpm] = BPMForces4(BW,
AnkleAngle, GaitCycle, ShankLowDepth, ShankSupportDepth,
ShankSupportWidth, ShankBPMAnchorDepth, ShankHighHeight,
ShankSupportHeight, BaseAnkleRotationLength, AnkleBPMWidth,
AnkleBPMBackwards, AnkleHeight, BaseBackHeight, AnkleBPMHeight);

%% Moment caused by the BPM at the ankle shaft
MomentAnkle = zeros(length(GaitCycle),3);
for u = 1:length(GaitCycle)
    if GaitCycle(u) <= 0.48
        MomentAnkle(u,:) = cross(Rabpm(u,:), Fbpm(u,:));
    else
        MomentAnkle(u,:) = [0 0 0];
    end
end

%% Ratchet Sizing
%Maximum moment
RatchetTorque = max(max(abs(MomentAnkle)));

%Ratchet Selection
SUTMPA = 450; %Steel 1018 SUT
[RatchetWidth, RatchetDiameter, RatchetTeethNumber, RatchetTeethAngle,
RatchetHoleDiameter, MountDistance, CenterDistance, PawlRadius,
PawlLength, PawlHoleDiameter, RatchetPitchAngle, RatchetTeethDepth,
PawlWidth] = RatchetDesign(AnkleHeight, RatchetTorque, SUTMPA);
```

```
%Ratchet Forces
[Fpawlankle, ~] = RatchetForces(GaitCycle, RatchetDiameter,
RatchetTeethDepth, MomentAnkle);

end

% Pawl Shaft Dimensions
Dpawl = PawlHoleDiameter;
Wpawl = PawlWidth;
    % Pawl Shaft Axial Bearings/Bushing
    [DpAxial, DpAxialOut, WpAxialBearing] = BushingSizing(Dpawl/1.2);
    WpAxialBearing = ((1/8)*25.4/1000) - Support;
WpAxial = WpAxialBearing;
Dpawl = DpAxial*1.2;
PawlHoleDiameter = Dpawl;
DpFree = Dpawl*1.2;
    % Pawl Shaft Thrust Bearings/Bushing
    [ DpThrust, DpThrustOut, WpThrustBearing ] =
BushingSizing(DpFree/1.2);
    WpThrustBearing = (1/4)*25.8/1000;
WpThrust = WpThrustBearing;
% Pawl Shaft Key
[KeyPawlWidth, KeyPawlHeight,~] = KeyChooser( Dpawl );
[KeyPawlLength,KeyPawlNumber] = KeyLength(0, Dpawl, Wpawl, KeyPawlWidth );

CasingDriveElevation = Support;
WpFree =
(Support+RatchetWidth+Support+CasingDriveDepth*2+CasingDriveElevation) -
(Support+PawlWidth+WpThrust);

% Ankle Shaft Dimensions
WaLarge = Support;
WaShank = ShankSideWidth;
WaAxial = WaAxialBearing;
WaSupport = Support + Support;
WaRatchet = RatchetWidth;
WaSmall = CamWidth;

% Ankle Shaft Shank Key
[KeyAShankWidth, KeyAShankHeight,~, KeyAShankNumber, KeyAShankLength] =
SplineChooser( DaShank, WaShank );

% Ankle Shaft Ratchet and Cam Key
[KeyARatchetWidth, KeyARatchetHeight,~] = KeyChooser(DaRatchet  );
[ KeyARatchetLength, KeyARatchetNumber ] = KeyLength(0, DaRatchet,
RatchetWidth, KeyARatchetWidth );
% Ankle Shaft Retaining Ring
[ RingARacthetIntDia, RingARacthetExtDia, RingAWidth ] =
RetainingRingSizing(DaRatchet);
RingAHeight = RingARacthetExtDia - RingARacthetIntDia;
RingADepth = DaRatchet - RingARacthetIntDia;

% Casing Dimensions
```

```
CasingDepth = BaseOutterWidth + Support + Wpawl + WpFree + WpThrust +
CasingThickness;
CasingDpThrust = 1.5*DpThrustOut;

WaFree = (WpFree+WpThrust) -
(CasingDriveElevation+CasingDriveDepth+CamWidth/2+Support);

%Cam Sizing
FollowerMax = CasingDriveHeight;
[CamProfile] = CamSizer3(RatchetHoleDiameter, PawlLength,PawlRadius,
FollowerMax);

%Pivot Holder Sizing
PivotReachLength = CasingDriveDepth+Support - DriveShaftDiameter -
Support;

%% Files Linked to SOLIDWORKS
FilePath = 'C:\Users\M.Doumit\Desktop\Exoskeleton\Conceptual Design\V9';
Shank_File = strcat(FilePath,'\','Equations_Shank.txt');
Base_File = strcat(FilePath,'\','Equations_Base.txt');
Casing_File = strcat(FilePath,'\','Equations_Casing.txt');
RatchetPawl_File = strcat(FilePath,'\','Equations_Ratchet_and_Pawl.txt');
Shaft_File = strcat(FilePath,'\','Equations_Shaft_And_Accessories.txt');
Drive_File = strcat(FilePath,'\','Equations_Slider_And_Drive.txt');
ToeBPM_File = strcat(FilePath,'\','Equations_Toe_BPM.txt');
Cam_File = strcat(FilePath,'\','Cam_Profile.txt');
Total_File = strcat(FilePath,'\','Equations_Total.txt');

% Shank Dimensions
fid = fopen(Shank_File,'w+t');
fprintf(fid,strcat('ShankLowDepth=',num2str(ShankLowDepth),'\n'));
fprintf(fid,strcat('ShankHighDepth=',num2str(ShankHighDepth),'\n'));
fprintf(fid,strcat('ShankLowHeight=',num2str(ShankLowHeight),'\n'));
fprintf(fid,strcat('ShankHighHeight=',num2str(ShankHighHeight),'\n'));
fprintf(fid,strcat('ShankSupportHeight=',num2str(ShankSupportHeight),'\n')
);
fprintf(fid,strcat('ShankSupportWidth=',num2str(ShankSupportWidth),'\n'));
fprintf(fid,strcat('ShankSupportDepth=',num2str(ShankSupportDepth),'\n'));
fprintf(fid,strcat('ShankHalfWidth=',num2str(ShankHalfWidth),'\n'));
fprintf(fid,strcat('ShankSideWidth=',num2str(ShankSideWidth),'\n'));
fprintf(fid,strcat('ShankBPMAnchorDiameter=',num2str(ShankBPMAnchorDiamete
r),'\n'));
fprintf(fid,strcat('ShankBPMAnchorHeight=',num2str(ShankBPMAnchorHeight),'
\n'));
fprintf(fid,strcat('ShankBPMAnchorWidth=',num2str(ShankBPMAnchorWidth),'\n
'));
fprintf(fid,strcat('ShankBPMAnchorDepth=',num2str(ShankBPMAnchorDepth),'\n
'));
fprintf(fid,strcat('DaShank=',num2str(DaShank),'\n'));
fprintf(fid,strcat('KeyAShankWidth=',num2str(KeyAShankWidth),'\n'));
fprintf(fid,strcat('KeyAShankHeight=',num2str(KeyAShankHeight),'\n'));
fprintf(fid,strcat('KeyAShankNumber=',num2str(KeyAShankNumber ),'\n'));
fprintf(fid,strcat('KeyAShankLength=',num2str(KeyAShankLength),'\n'));
fprintf(fid,strcat('ShankCalfSupportRadius=',num2str(ShankCalfSupportRadiu
s),'\n'));
fclose(fid);
```

```
% File for Base segment
fid = fopen(Base_File,'w+t');
fprintf(fid,strcat('AnkleHeight=',num2str(AnkleHeight),'\n'));
fprintf(fid,strcat('BaseWidth=',num2str(BaseWidth),'\n'));
fprintf(fid,strcat('BaseBackWidth=',num2str(BaseBackWidth),'\n'));
fprintf(fid,strcat('BaseWallThickness=',num2str(BaseWallThickness),'\n'));
fprintf(fid,strcat('BaseAnkleRotationLength=',num2str(BaseAnkleRotationLen
gth),'\n'));
fprintf(fid,strcat('BaseConnectionWidth=',num2str(BaseConnectionWidth),'\n
'));
fprintf(fid,strcat('RatchetDiameter=',num2str(RatchetDiameter),'\n'));
fprintf(fid,strcat('BaseAnkleSupportHeight=',num2str(BaseAnkleSupportHeigh
t),'\n'));
fprintf(fid,strcat('BaseBackHeight=',num2str(BaseBackHeight),'\n'));
fprintf(fid,strcat('BasePlateHeight=',num2str(BasePlateHeight),'\n'));
fprintf(fid,strcat('BaseLowHeight=',num2str(BaseLowHeight),'\n'));
fprintf(fid,strcat('BaseBPMAnchorWidth=',num2str(BaseBPMAnchorWidth),'\n')
);
fprintf(fid,strcat('BaseBPMAnchorHeight=',num2str(BaseBPMAnchorHeight),'\n
'));
fprintf(fid,strcat('BasePawlBack=',num2str(BasePawlBack),'\n'));
fprintf(fid,strcat('BasePawlLength=',num2str(BasePawlLength),'\n'));
fprintf(fid,strcat('BaseRatchetForward=',num2str(BaseRatchetForward),'\n')
);
fprintf(fid,strcat('BasePawlTop=',num2str(BasePawlTop),'\n'));
fprintf(fid,strcat('BasePawlHeight=',num2str(BasePawlHeight),'\n'));
fprintf(fid,strcat('DpAxialOut=',num2str(DpAxialOut),'\n'));
fprintf(fid,strcat('WpAxial=',num2str(WpAxial),'\n'));
fprintf(fid,strcat('DaAxialOut=',num2str(DaAxialOut),'\n'));
fprintf(fid,strcat('DaShank=',num2str(DaShank),'\n'));
fprintf(fid,strcat('DaAxial=',num2str(DaAxial),'\n'));
fprintf(fid,strcat('CenterDistance=',num2str(CenterDistance),'\n'));
fprintf(fid,strcat('RatchetTeethDepth=',num2str(RatchetTeethDepth),'\n'));
fprintf(fid,strcat('RatchetInnerRadius=',num2str(RatchetInnerRadius),'\n')
);
fprintf(fid,strcat('Support=',num2str(Support),'\n'));
fprintf(fid,strcat('ShankLowDepth=',num2str(ShankLowDepth),'\n'));
fprintf(fid,strcat('ShankSideWidth=',num2str(ShankSideWidth),'\n'));
fclose(fid);

% File for Casing
fid = fopen(Casing_File,'w+t');
fprintf(fid,strcat('PawlRadius=',num2str(PawlRadius),'\n'));
fprintf(fid,strcat('PawlLength=',num2str(PawlLength),'\n'));
fprintf(fid,strcat('PawlHoleDiameter=',num2str(PawlHoleDiameter),'\n'));
fprintf(fid,strcat('RatchetDiameter=',num2str(RatchetDiameter),'\n'));
fprintf(fid,strcat('RatchetWidth=',num2str(RatchetWidth),'\n'));
fprintf(fid,strcat('CamMinRadius=',num2str(CamMinRadius),'\n'));
fprintf(fid,strcat('CamMaxRadius=',num2str(CamMaxRadius),'\n'));
fprintf(fid,strcat('DriveSphericalDiameter=',num2str(DriveSphericalDiamete
r),'\n'));
fprintf(fid,strcat('DriveShaftDiameter=',num2str(DriveShaftDiameter),'\n')
);
fprintf(fid,strcat('DriveCamDiameter=',num2str(DriveCamDiameter),'\n'));
fprintf(fid,strcat('DriveCamSpacing=',num2str(DriveCamSpacing),'\n'));
fprintf(fid,strcat('DriveCamBaseSpacing=',num2str(DriveCamBaseSpacing),'\n
'));
fprintf(fid,strcat('DriveCamAngleHeight=',num2str(DriveCamAngleHeight),'\n
'));
fprintf(fid,strcat('DriveCamHeight=',num2str(DriveCamHeight),'\n'));
fprintf(fid,strcat('DriveCamPushAngle=',num2str(DriveCamPushAngle),'\n'));
fprintf(fid,strcat('DriveShaftLength=',num2str(DriveShaftLength),'\n'));
fprintf(fid,strcat('SliderAngleHeight=',num2str(SliderAngleHeight),'\n'));
fprintf(fid,strcat('SliderDiameter=',num2str(SliderDiameter),'\n'));
fprintf(fid,strcat('SliderLength=',num2str(SliderLength),'\n'));
fprintf(fid,strcat('Dpawl=',num2str(Dpawl),'\n'));
fprintf(fid,strcat('DpAxial=',num2str(DpAxial),'\n'));
fprintf(fid,strcat('WpAxial=',num2str(WpAxial),'\n'));
fprintf(fid,strcat('DpAxialOut=',num2str(DpAxialOut),'\n'));
fprintf(fid,strcat('DpThrustOut=',num2str(DpThrustOut),'\n'));
fprintf(fid,strcat('CasingThickness=',num2str(CasingThickness),'\n'));
fprintf(fid,strcat('CasingDpThrust=',num2str(CasingDpThrust),'\n'));
fprintf(fid,strcat('CasingShaftSupport=',num2str(CasingShaftSupport),'\n')
);
fprintf(fid,strcat('CasingShaftSupportHeight=',num2str(CasingShaftSupportH
eight),'\n'));
fprintf(fid,strcat('CasingShaftSupportPosition=',num2str(CasingShaftSuppor
tPosition),'\n'));
fprintf(fid,strcat('CasingRatchetPosition=',num2str(CasingRatchetPosition)
,'\n'));
fprintf(fid,strcat('CasingDriveWidth=',num2str(CasingDriveWidth),'\n'));
fprintf(fid,strcat('CasingDrivePosition=',num2str(CasingDrivePosition),'\n
'));
fprintf(fid,strcat('CasingDriveHeight=',num2str(CasingDriveHeight),'\n'));
fprintf(fid,strcat('CasingDriveDepth=',num2str(CasingDriveDepth),'\n'));
fprintf(fid,strcat('CasingHeight=',num2str(CasingHeight),'\n'));
fprintf(fid,strcat('CasingDepth=',num2str(CasingDepth),'\n'));
fprintf(fid,strcat('CasingWidth=',num2str(CasingWidth),'\n'));
fprintf(fid,strcat('WpThrust=',num2str(WpThrust),'\n'));
fprintf(fid,strcat('RatchetInnerRadius=',num2str(RatchetInnerRadius),'\n')
);
fprintf(fid,strcat('CenterDistance=',num2str(CenterDistance),'\n'));
fprintf(fid,strcat('CasingQuarterRatchetDiameter=',num2str(CasingQuarterRa
tchetDiameter),'\n'));
fprintf(fid,strcat('BaseOutterWidth=',num2str(BaseOutterWidth),'\n'));
fprintf(fid,strcat('CasingDriveElevation=',num2str(CasingDriveElevation),'
\n'));
fclose(fid);

% File for Ratchel and Pawl
fid = fopen(RatchetPawl_File,'w+t');
fprintf(fid,strcat('PawlRadius=',num2str(PawlRadius),'\n'));
fprintf(fid,strcat('PawlHoleDiameter=',num2str(PawlHoleDiameter),'\n'));
fprintf(fid,strcat('PawlLength=',num2str(PawlLength),'\n'));
fprintf(fid,strcat('PawlWidth=',num2str(PawlWidth),'\n'));
fprintf(fid,strcat('PawlFilet=',num2str(PawlFilet),'\n'));
fprintf(fid,strcat('DriveSphericalDiameter=',num2str(DriveSphericalDiamete
r),'\n'));
fprintf(fid,strcat('RatchetTeethNumber=',num2str(RatchetTeethNumber),'\n')
);
fprintf(fid,strcat('RatchetHoleDiameter=',num2str(RatchetHoleDiameter),'\n
'));
```

```
fprintf(fid,strcat('RatchetDiameter=',num2str(RatchetDiameter),'\n'));
fprintf(fid,strcat('RatchetWidth=',num2str(RatchetWidth),'\n'));
fprintf(fid,strcat('RatchetTeethDepth=',num2str(RatchetTeethDepth),'\n'));
fprintf(fid,strcat('RatchetTeethPresentAngle=',num2str(RatchetTeethPresent
Angle),'\n'));
fprintf(fid,strcat('RatchetTeethAngle=',num2str(RatchetTeethAngle),'\n'));
fprintf(fid,strcat('RatchetPitchAngle=',num2str(RatchetPitchAngle),'\n'));
fprintf(fid,strcat('CenterDistance=',num2str(CenterDistance),'\n'));
fprintf(fid,strcat('CamMinRadius=',num2str(CamMinRadius),'\n'));
fprintf(fid,strcat('CamMaxRadius=',num2str(CamMaxRadius),'\n'));
fprintf(fid,strcat('DriveCamDiameter=',num2str(DriveCamDiameter),'\n'));
fprintf(fid,strcat('SliderDiameter=',num2str(SliderDiameter),'\n'));
fprintf(fid,strcat('CamWidth=',num2str(CamWidth),'\n'));
fprintf(fid,strcat('KeyPawlWidth=',num2str(KeyPawlWidth),'\n'));
fprintf(fid,strcat('KeyPawlHeight=',num2str(KeyPawlHeight),'\n'));
fprintf(fid,strcat('KeyPawlNumber=',num2str(KeyPawlNumber),'\n'));
fprintf(fid,strcat('KeyARatchetWidth=',num2str(KeyARatchetWidth),'\n'));
fprintf(fid,strcat('KeyARatchetHeight=',num2str(KeyARatchetHeight),'\n'));
fprintf(fid,strcat('KeyARatchetNumber=',num2str(KeyARatchetNumber),'\n'));
fprintf(fid,strcat('Support=',num2str(Support),'\n'));
fprintf(fid,strcat('PawlHingeDiameter=',num2str(PawlHingeDiameter),'\n'));
fprintf(fid,strcat('PivotReachLength=',num2str(PivotReachLength),'\n'));
fprintf(fid,strcat('DriveExtDiameter=',num2str(DriveExtDiameter),'\n'));
fclose(fid);

% File for Shafts and Accessories
fid = fopen(Shaft_File,'w+t');
fprintf(fid,strcat('PawlWidth=',num2str(PawlWidth),'\n'));
fprintf(fid,strcat('PawlHoleDiameter=',num2str(PawlHoleDiameter),'\n'));
fprintf(fid,strcat('SliderDiameter=',num2str(SliderDiameter),'\n'));
fprintf(fid,strcat('CamWidth=',num2str(CamWidth),'\n'));
fprintf(fid,strcat('RatchetWidth=',num2str(RatchetWidth),'\n'));
fprintf(fid,strcat('RatchetHoleDiameter=',num2str(RatchetHoleDiameter),'\n
'));
fprintf(fid,strcat('DaRatchet=',num2str(DaRatchet),'\n'));
fprintf(fid,strcat('WaRatchet=',num2str(WaRatchet),'\n'));
fprintf(fid,strcat('DaAxial=',num2str(DaAxial),'\n'));
fprintf(fid,strcat('WaAxial=',num2str(WaAxial),'\n'));
fprintf(fid,strcat('DaShank=',num2str(DaShank),'\n'));
fprintf(fid,strcat('WaShank=',num2str(WaShank),'\n'));



fprintf(fid,strcat('DaLarge=',num2str(DaLarge),'\n'));
fprintf(fid,strcat('WaLarge=',num2str(WaLarge),'\n'));
fprintf(fid,strcat('DaSmall=',num2str(DaSmall),'\n'));
fprintf(fid,strcat('WaSmall=',num2str(WaSmall),'\n'));

fprintf(fid,strcat('DaAxialOut=',num2str(DaAxialOut),'\n'));
fprintf(fid,strcat('WaAxialBearing=',num2str(WaAxialBearing),'\n'));
fprintf(fid,strcat('KeyAShankWidth=',num2str(KeyAShankWidth),'\n'));
fprintf(fid,strcat('KeyAShankHeight=',num2str(KeyAShankHeight),'\n'));
fprintf(fid,strcat('KeyAShankNumber=',num2str(KeyAShankNumber ),'\n'));
fprintf(fid,strcat('KeyAShankLength=',num2str(KeyAShankLength),'\n'));
fprintf(fid,strcat('KeyARatchetWidth=',num2str(KeyARatchetWidth),'\n'));
fprintf(fid,strcat('KeyARatchetHeight=',num2str(KeyARatchetHeight),'\n'));
fprintf(fid,strcat('KeyARatchetNumber=',num2str(KeyARatchetNumber),'\n'));
fprintf(fid,strcat('KeyARatchetLength=',num2str(KeyARatchetLength),'\n'));
fprintf(fid,strcat('RingAWidth=',num2str(RingAWidth),'\n'));
fprintf(fid,strcat('RingAHeight=',num2str(RingAHeight),'\n'));
fprintf(fid,strcat('RingADepth=',num2str(RingADepth),'\n'));
fprintf(fid,strcat('Dpawl=',num2str(Dpawl),'\n'));
fprintf(fid,strcat('Wpawl=',num2str(Wpawl),'\n'));
fprintf(fid,strcat('DpThrust=',num2str(DpThrust),'\n'));
fprintf(fid,strcat('WpThrust=',num2str(WpThrust),'\n'));
fprintf(fid,strcat('DpFree=',num2str(DpFree),'\n'));
fprintf(fid,strcat('WpFree=',num2str(WpFree),'\n'));
fprintf(fid,strcat('DpAxial=',num2str(DpAxial),'\n'));
fprintf(fid,strcat('WpAxial=',num2str(WpAxial),'\n'));
fprintf(fid,strcat('DriveSphericalDiameter=',num2str(DriveSphericalDiamete
r),'\n'));
fprintf(fid,strcat('CasingDriveDepth=',num2str(CasingDriveDepth),'\n'));
fprintf(fid,strcat('KeyPawlWidth=',num2str(KeyPawlWidth),'\n'));
fprintf(fid,strcat('KeyPawlHeight=',num2str(KeyPawlHeight),'\n'));
fprintf(fid,strcat('KeyPawlNumber=',num2str(KeyPawlNumber),'\n'));
fprintf(fid,strcat('KeyPawlLength=',num2str(KeyPawlLength),'\n'));
fprintf(fid,strcat('DpThrustOut=',num2str(DpThrustOut),'\n'));
fprintf(fid,strcat('WpThrustBearing=',num2str(WpThrustBearing),'\n'));
fprintf(fid,strcat('DpAxialOut=',num2str(DpAxialOut),'\n'));
fprintf(fid,strcat('WpAxialBearing=',num2str(WpAxialBearing),'\n'));
fprintf(fid,strcat('Support=',num2str(Support),'\n'));
fprintf(fid,strcat('DriveShaftDiameter=',num2str(DriveShaftDiameter),'\n')
);
fprintf(fid,strcat('PawlRadius=',num2str(PawlRadius),'\n'));
fprintf(fid,strcat('PawlLength=',num2str(PawlLength),'\n'));
fprintf(fid,strcat('WaFree=',num2str(WaFree),'\n'));
fprintf(fid,strcat('DaFree=',num2str(DaFree),'\n'));
fclose(fid);

% Files for Slider and Drive mechanism
fid = fopen(Drive_File,'w+t');
fprintf(fid,strcat('DriveSphericalDiameter=',num2str(DriveSphericalDiamete
r),'\n'));
fprintf(fid,strcat('DriveShaftDiameter=',num2str(DriveShaftDiameter),'\n')
);
fprintf(fid,strcat('DriveCamDiameter=',num2str(DriveCamDiameter),'\n'));
fprintf(fid,strcat('RatchetTeethDepth=',num2str(RatchetTeethDepth),'\n'));
fprintf(fid,strcat('DriveCamSpacing=',num2str(DriveCamSpacing),'\n'));
fprintf(fid,strcat('DriveCamBaseSpacing=',num2str(DriveCamBaseSpacing),'\n
'));
fprintf(fid,strcat('DriveCamAngleHeight=',num2str(DriveCamAngleHeight),'\n
'));
fprintf(fid,strcat('DriveCamHeight=',num2str(DriveCamHeight),'\n'));
fprintf(fid,strcat('DriveCamPushAngle=',num2str(DriveCamPushAngle),'\n'));
fprintf(fid,strcat('DriveShaftLength=',num2str(DriveShaftLength),'\n'));
fprintf(fid,strcat('SliderAngleHeight=',num2str(SliderAngleHeight),'\n'));
fprintf(fid,strcat('SliderLength=',num2str(SliderLength),'\n'));
fprintf(fid,strcat('SliderDiameter=',num2str(SliderDiameter),'\n'));
fprintf(fid,strcat('PawlHingeDiameter=',num2str(PawlHingeDiameter),'\n'));
fprintf(fid,strcat('PawlRadius=',num2str(PawlRadius),'\n'));
fprintf(fid,strcat('DriveSlotWidth=',num2str(DriveSlotWidth),'\n'));
fprintf(fid,strcat('DriveExtDiameter=',num2str(DriveExtDiameter),'\n'));
fclose(fid);
```

```
% Files for Toe and BPM parts
fid = fopen(ToeBPM_File,'w+t');
fprintf(fid,strcat('AnkleBPMHeight=',num2str(AnkleBPMHeight),'\n'));
fprintf(fid,strcat('AnkleBPMBackwards=',num2str(AnkleBPMBackwards),'\n'));
fprintf(fid,strcat('AnkleBPMWidth=',num2str(AnkleBPMWidth),'\n'));
fprintf(fid,strcat('AnkleBPMFlatHeight=',num2str(AnkleBPMFlatHeight),'\n')
);
fprintf(fid,strcat('AnkleBPMDiameter=',num2str(AnkleBPMDiameter),'\n'));
fprintf(fid,strcat('AnkleBPMDepth=',num2str(AnkleBPMDepth),'\n'));
fprintf(fid,strcat('BaseBPMAnchorWidth=',num2str(BaseBPMAnchorWidth),'\n')
);
fclose(fid);

% File for cam profile
fid = fopen(Cam_File, 'w+t'); % Open for writing
for i=1:length(CamProfile)
   fprintf(fid, '%8.6f %8.6f %8.6f\n', CamProfile(i,:));
end
fclose(fid);

%File for Construction in Machine Shop
fid = fopen(Total_File, 'w+t');
fprintf(fid,strcat('ShankLowDepth=',num2str(ShankLowDepth),'\n'));
fprintf(fid,strcat('ShankHighDepth=',num2str(ShankHighDepth),'\n'));
fprintf(fid,strcat('ShankLowHeight=',num2str(ShankLowHeight),'\n'));
fprintf(fid,strcat('ShankHighHeight=',num2str(ShankHighHeight),'\n'));
fprintf(fid,strcat('ShankSupportHeight=',num2str(ShankSupportHeight),'\n')
);
fprintf(fid,strcat('ShankSupportWidth=',num2str(ShankSupportWidth),'\n'));
fprintf(fid,strcat('ShankSupportDepth=',num2str(ShankSupportDepth),'\n'));
fprintf(fid,strcat('ShankHalfWidth=',num2str(ShankHalfWidth),'\n'));
fprintf(fid,strcat('ShankSideWidth=',num2str(ShankSideWidth),'\n'));
fprintf(fid,strcat('ShankBPMAnchorDiameter=',num2str(ShankBPMAnchorDiamete
r),'\n'));
fprintf(fid,strcat('ShankBPMAnchorHeight=',num2str(ShankBPMAnchorHeight),'
\n'));
fprintf(fid,strcat('ShankBPMAnchorWidth=',num2str(ShankBPMAnchorWidth),'\n
'));
fprintf(fid,strcat('ShankBPMAnchorDepth=',num2str(ShankBPMAnchorDepth),'\n
'));
fprintf(fid,strcat('ShankCalfSupportRadius=',num2str(ShankCalfSupportRadiu
s),'\n'));

fprintf(fid,strcat('AnkleHeight=',num2str(AnkleHeight),'\n'));
fprintf(fid,strcat('BaseWidth=',num2str(BaseWidth),'\n'));
fprintf(fid,strcat('BaseBackWidth=',num2str(BaseBackWidth),'\n'));
fprintf(fid,strcat('BaseWallThickness=',num2str(BaseWallThickness),'\n'));
fprintf(fid,strcat('BaseAnkleRotationLength=',num2str(BaseAnkleRotationLen
gth),'\n'));
fprintf(fid,strcat('BaseConnectionWidth=',num2str(BaseConnectionWidth),'\n
'));
fprintf(fid,strcat('BaseAnkleSupportHeight=',num2str(BaseAnkleSupportHeigh
t),'\n'));
fprintf(fid,strcat('BaseBackHeight=',num2str(BaseBackHeight),'\n'));
fprintf(fid,strcat('BasePlateHeight=',num2str(BasePlateHeight),'\n'));
fprintf(fid,strcat('BaseLowHeight=',num2str(BaseLowHeight),'\n'));
fprintf(fid,strcat('BaseBPMAnchorWidth=',num2str(BaseBPMAnchorWidth),'\n')
);
fprintf(fid,strcat('BaseBPMAnchorHeight=',num2str(BaseBPMAnchorHeight),'\n
'));
fprintf(fid,strcat('BasePawlBack=',num2str(BasePawlBack),'\n'));
fprintf(fid,strcat('BasePawlLength=',num2str(BasePawlLength),'\n'));
fprintf(fid,strcat('BaseRatchetForward=',num2str(BaseRatchetForward),'\n')
);
fprintf(fid,strcat('BasePawlTop=',num2str(BasePawlTop),'\n'));
fprintf(fid,strcat('BasePawlHeight=',num2str(BasePawlHeight),'\n'));

fprintf(fid,strcat('CasingThickness=',num2str(CasingThickness),'\n'));
fprintf(fid,strcat('CasingDpThrust=',num2str(CasingDpThrust),'\n'));
fprintf(fid,strcat('CasingShaftSupport=',num2str(CasingShaftSupport),'\n')
);
fprintf(fid,strcat('CasingShaftSupportHeight=',num2str(CasingShaftSupportH
eight),'\n'));
fprintf(fid,strcat('CasingShaftSupportPosition=',num2str(CasingShaftSuppor
tPosition),'\n'));
fprintf(fid,strcat('CasingRatchetPosition=',num2str(CasingRatchetPosition)
,'\n'));
fprintf(fid,strcat('CasingDriveWidth=',num2str(CasingDriveWidth),'\n'));
fprintf(fid,strcat('CasingDrivePosition=',num2str(CasingDrivePosition),'\n
'));
fprintf(fid,strcat('CasingDriveHeight=',num2str(CasingDriveHeight),'\n'));
fprintf(fid,strcat('CasingDriveDepth=',num2str(CasingDriveDepth),'\n'));
fprintf(fid,strcat('CasingHeight=',num2str(CasingHeight),'\n'));
fprintf(fid,strcat('CasingDepth=',num2str(CasingDepth),'\n'));
fprintf(fid,strcat('CasingWidth=',num2str(CasingWidth),'\n'));
fprintf(fid,strcat('RatchetInnerRadius=',num2str(RatchetInnerRadius),'\n')
);
fprintf(fid,strcat('CasingQuarterRatchetDiameter=',num2str(CasingQuarterRa
tchetDiameter),'\n'));
fprintf(fid,strcat('BaseOutterWidth=',num2str(BaseOutterWidth),'\n'));
fprintf(fid,strcat('CasingDriveElevation=',num2str(CasingDriveElevation),'
\n'));

fprintf(fid,strcat('PawlRadius=',num2str(PawlRadius),'\n'));
fprintf(fid,strcat('PawlHoleDiameter=',num2str(PawlHoleDiameter),'\n'));
fprintf(fid,strcat('PawlLength=',num2str(PawlLength),'\n'));
fprintf(fid,strcat('PawlWidth=',num2str(PawlWidth),'\n'));
fprintf(fid,strcat('PawlFilet=',num2str(PawlFilet),'\n'));
fprintf(fid,strcat('RatchetTeethNumber=',num2str(RatchetTeethNumber),'\n')
);
fprintf(fid,strcat('RatchetHoleDiameter=',num2str(RatchetHoleDiameter),'\n
'));
fprintf(fid,strcat('RatchetDiameter=',num2str(RatchetDiameter),'\n'));
fprintf(fid,strcat('RatchetWidth=',num2str(RatchetWidth),'\n'));
fprintf(fid,strcat('RatchetTeethDepth=',num2str(RatchetTeethDepth),'\n'));
fprintf(fid,strcat('RatchetTeethPresentAngle=',num2str(RatchetTeethPresent
Angle),'\n'));
fprintf(fid,strcat('RatchetTeethAngle=',num2str(RatchetTeethAngle),'\n'));
fprintf(fid,strcat('RatchetPitchAngle=',num2str(RatchetPitchAngle),'\n'));
fprintf(fid,strcat('CenterDistance=',num2str(CenterDistance),'\n'));
fprintf(fid,strcat('CamMinRadius=',num2str(CamMinRadius),'\n'));
fprintf(fid,strcat('CamMaxRadius=',num2str(CamMaxRadius),'\n'));
fprintf(fid,strcat('CamWidth=',num2str(CamWidth),'\n'));
fprintf(fid,strcat('PawlHingeDiameter=',num2str(PawlHingeDiameter),'\n'));
fprintf(fid,strcat('PivotReachLength=',num2str(PivotReachLength),'\n'));

fprintf(fid,strcat('DaRatchet=',num2str(DaRatchet),'\n'));
fprintf(fid,strcat('WaRatchet=',num2str(WaRatchet),'\n'));
fprintf(fid,strcat('DaAxial=',num2str(DaAxial),'\n'));
fprintf(fid,strcat('WaAxial=',num2str(WaAxial),'\n'));
fprintf(fid,strcat('DaShank=',num2str(DaShank),'\n'));
fprintf(fid,strcat('WaShank=',num2str(WaShank),'\n'));

fprintf(fid,strcat('DaLarge=',num2str(DaLarge),'\n'));
fprintf(fid,strcat('WaLarge=',num2str(WaLarge),'\n'));
fprintf(fid,strcat('DaSmall=',num2str(DaSmall),'\n'));
fprintf(fid,strcat('WaSmall=',num2str(WaSmall),'\n'));

fprintf(fid,strcat('DaAxialOut=',num2str(DaAxialOut),'\n'));
fprintf(fid,strcat('WaAxialBearing=',num2str(WaAxialBearing),'\n'));
fprintf(fid,strcat('KeyAShankWidth=',num2str(KeyAShankWidth),'\n'));
fprintf(fid,strcat('KeyAShankHeight=',num2str(KeyAShankHeight),'\n'));
fprintf(fid,strcat('KeyAShankNumber=',num2str(KeyAShankNumber ),'\n'));
fprintf(fid,strcat('KeyAShankLength=',num2str(KeyAShankLength),'\n'));
fprintf(fid,strcat('KeyARatchetWidth=',num2str(KeyARatchetWidth),'\n'));
fprintf(fid,strcat('KeyARatchetHeight=',num2str(KeyARatchetHeight),'\n'));
fprintf(fid,strcat('KeyARatchetNumber=',num2str(KeyARatchetNumber),'\n'));
fprintf(fid,strcat('KeyARatchetLength=',num2str(KeyARatchetLength),'\n'));
fprintf(fid,strcat('RingAWidth=',num2str(RingAWidth),'\n'));
fprintf(fid,strcat('RingAHeight=',num2str(RingAHeight),'\n'));
fprintf(fid,strcat('RingADepth=',num2str(RingADepth),'\n'));
fprintf(fid,strcat('Dpawl=',num2str(Dpawl),'\n'));
fprintf(fid,strcat('Wpawl=',num2str(Wpawl),'\n'));
fprintf(fid,strcat('DpThrust=',num2str(DpThrust),'\n'));
fprintf(fid,strcat('WpThrust=',num2str(WpThrust),'\n'));
fprintf(fid,strcat('DpFree=',num2str(DpFree),'\n'));
fprintf(fid,strcat('WpFree=',num2str(WpFree),'\n'));
fprintf(fid,strcat('DpAxial=',num2str(DpAxial),'\n'));
fprintf(fid,strcat('WpAxial=',num2str(WpAxial),'\n'));
fprintf(fid,strcat('KeyPawlWidth=',num2str(KeyPawlWidth),'\n'));
fprintf(fid,strcat('KeyPawlHeight=',num2str(KeyPawlHeight),'\n'));
fprintf(fid,strcat('KeyPawlNumber=',num2str(KeyPawlNumber),'\n'));
fprintf(fid,strcat('KeyPawlLength=',num2str(KeyPawlLength),'\n'));
fprintf(fid,strcat('DpThrustOut=',num2str(DpThrustOut),'\n'));
fprintf(fid,strcat('WpThrustBearing=',num2str(WpThrustBearing),'\n'));
fprintf(fid,strcat('DpAxialOut=',num2str(DpAxialOut),'\n'));
fprintf(fid,strcat('WpAxialBearing=',num2str(WpAxialBearing),'\n'));
fprintf(fid,strcat('Support=',num2str(Support),'\n'));
fprintf(fid,strcat('WaFree=',num2str(WaFree),'\n'));
fprintf(fid,strcat('DaFree=',num2str(DaFree),'\n'));
fprintf(fid,strcat('DaSupport=',num2str(DaSupport),'\n'));
fprintf(fid,strcat('WaSupport=',num2str(WaSupport),'\n'));

fprintf(fid,strcat('DriveSphericalDiameter=',num2str(DriveSphericalDiamete
r),'\n'));
fprintf(fid,strcat('DriveShaftDiameter=',num2str(DriveShaftDiameter),'\n')
);
```

```
fprintf(fid,strcat('DriveCamDiameter=',num2str(DriveCamDiameter),'\n'));
fprintf(fid,strcat('DriveCamSpacing=',num2str(DriveCamSpacing),'\n'));
fprintf(fid,strcat('DriveCamBaseSpacing=',num2str(DriveCamBaseSpacing),'\n
'));
fprintf(fid,strcat('DriveCamAngleHeight=',num2str(DriveCamAngleHeight),'\n
'));
fprintf(fid,strcat('DriveCamHeight=',num2str(DriveCamHeight),'\n'));
fprintf(fid,strcat('DriveCamPushAngle=',num2str(DriveCamPushAngle),'\n'));
fprintf(fid,strcat('DriveShaftLength=',num2str(DriveShaftLength),'\n'));
fprintf(fid,strcat('SliderAngleHeight=',num2str(SliderAngleHeight),'\n'));
fprintf(fid,strcat('SliderLength=',num2str(SliderLength),'\n'));
fprintf(fid,strcat('SliderDiameter=',num2str(SliderDiameter),'\n'));
fprintf(fid,strcat('DriveSlotWidth=',num2str(DriveSlotWidth),'\n'));
fprintf(fid,strcat('DriveExtDiameter=',num2str(DriveExtDiameter),'\n'));

fprintf(fid,strcat('AnkleBPMHeight=',num2str(AnkleBPMHeight),'\n'));
fprintf(fid,strcat('AnkleBPMBackwards=',num2str(AnkleBPMBackwards),'\n'));
fprintf(fid,strcat('AnkleBPMWidth=',num2str(AnkleBPMWidth),'\n'));
fprintf(fid,strcat('AnkleBPMFlatHeight=',num2str(AnkleBPMFlatHeight),'\n')
);
fprintf(fid,strcat('AnkleBPMDiameter=',num2str(AnkleBPMDiameter),'\n'));
fprintf(fid,strcat('AnkleBPMDepth=',num2str(AnkleBPMDepth),'\n'));

fclose(fid);
%% Final Message
fprintf('Log files written. \n Program Ending \n');
t = datestr(now());
fprintf(strcat('Analysis performed on :', num2str(t)));
end
```

```
function [Mhat What Hhat Lcomhiphat Lcomhat HipCirc Mthigh Wthigh Hthigh
Lcomkneethigh Lcomthigh ThighCirc Mshank Wshank ShankHeight Lcomankleshank
Lcomshank ShankCirc Mfoot Wfoot FootLength Lcomgroundankle Lcomfoot
FootWidth AnkleCirc AnkleHeight] = Anthrodata(H,BW)
%Computes mass, weight, length and center of mass of the HAT, thigh and
%lower leg based on Height (H) and bodyweight (BW)

%Head, Arms and Trunk
Mhat = 0.678*BW; %Mass
What = Mhat*9.81; % Weight
Hhat = (0.87 - 0.53)*H; %Length of the HAT section
Lcomhiphat = 0.374*Hhat; %Location of COM from the hips
Lcomhat = Lcomhiphat + (0.53*H); %Location of Center of Mass of HAT from
the ground
HipCirc = (1.278 - 0.719)*(H-1.524)/(1.856 - 1.524) + 0.719; %Hip
circumference interpolation

%Thigh. Each value is for one individual thigh
Mthigh = 0.100*BW; %Mass
Wthigh = Mthigh*9.81; %Weight
Hthigh = (0.53 - 0.285)*H; %Length
Lcomkneethigh = (0.567*Hthigh); %Location of COM from the knees
Lcomthigh = Lcomkneethigh + (0.285*H); %Location of Center of Mass of
thigh from ground
ThighCirc = (0.646 - 0.424)*(H-1.524)/(1.856 - 1.524) + 0.424; %Thigh
circumference interpolation
```

```
%Shank
Mshank = 0.0465*BW; %Mass
Wshank = Mshank*9.81; %Weight
ShankHeight = (0.285-0.039)*H; %Length
Lcomankleshank = 0.567*ShankHeight; %Location of COM from the ankles
Lcomshank= Lcomankleshank + 0.039*H; %Location of COM of shank from the
ground
ShankCirc = (0.464 - 0.318)*(H-1.524)/(1.856 - 1.524) + 0.318;  %Thigh
circumference interpolation

%Foot
Mfoot = 0.0145*BW; %Mass
Wfoot = Mfoot*9.81; %Weight
FootLength = 0.152*H; %Length of foot
Lcomgroundankle = 0.039*H/2; %Estimation of the location of COM vertically
from ground
Lcomfoot = 0.5*FootLength; %Location of COM along length of foot.
FootWidth = 0.055*H; %Width of foot
AnkleCirc = 0.133*H; %Ankle Circumference
AnkleHeight = 0.039*H; %Ankle Height
end
```

```
function [GaitPercent, AnkleAngle, KneeAngle, HipAngle, AnkleAngularVel,
KneeAngularVel, HipAngularVel, AnkleAngularAcc, KneeAngularAcc,
HipAngularAcc] = GaitData(H, Hthigh,Hshank,AnkleHeight)
%This function extracts the joint angle data from the a tabulated file
%based on data from Walking with Richard website. It outputs the angles,
%angular, velocities and angular accelerations that are needed for the
%dynamic force analysis, based on the maximum possible walking speed.

%% Determining the Time constant
Vwalking = 0.820*H; %Estimation of the walking velocity relative to height
based on Bohannon 1997 article
CycleLength = 2*0.414*H; %Estimation of the Cycle lenght (2 steps) based
on the average cycle length (see online)
TimeCycle = CycleLength/Vwalking;

%% Extracting Data
PathName = pwd; %Get file path
filepath = strcat(PathName, '\GaitAngles.txt'); %Concatenate filepath and
filename
fileID = fopen(filepath); %Open desired file
C = textscan(fileID,'%s %f %f %f','HeaderLines',1); %Read file without the
headers
fclose(fileID); %Close file
%Convert first column to numbers
for i = 1:length(C{1})
    C{1}{i} = (i-1)*2/100;
end
C{1} = cell2mat(C{1});

%% Reassigning Data
GaitPercent = C{1};
AnkleAngle = C{2}*pi/180;
KneeAngle = C{3}*pi/180;
HipAngle = C{4}*pi/180;
Time = TimeCycle*GaitPercent;
```

```
%% Initializing
AnkleAngularVelx = zeros(length(Time), 1);
KneeAngularVelx = zeros(length(Time),1);
HipAngularVelx = zeros(length(Time),1);
AnkleAngularAccx = zeros(length(Time),1);
KneeAngularAccx = zeros(length(Time),1);
HipAngularAccx = zeros(length(Time),1);

%% Calculating the Angular velocities
AnkleAngularVelxint = diff(AnkleAngle)/TimeCycle;
KneeAngularVelxint = diff(KneeAngle)/TimeCycle;
HipAngularVelxint = diff(HipAngle)/TimeCycle;

for i = 1:length(Time)
    if i == 1
        AnkleAngularVelx(i) = AnkleAngularVelxint(length(Time)-1);
        KneeAngularVelx(i) = KneeAngularVelxint(length(Time)-1);
        HipAngularVelx(i) = HipAngularVelxint(length(Time)-1);
    else
        AnkleAngularVelx(i) = AnkleAngularVelxint(i-1);
        KneeAngularVelx(i) = KneeAngularVelxint(i-1);
        HipAngularVelx(i) = HipAngularVelxint(i-1);
    end
end

%% Calculating the Angular acceleration
AnkleAngularAccxint = diff(AnkleAngularVelx)/TimeCycle;
KneeAngularAccxint = diff(KneeAngularVelx)/TimeCycle;
HipAngularAccxint = diff(HipAngularVelx)/TimeCycle;

for i = 1:length(Time)
    if i == 1
        AnkleAngularAccx(i) = AnkleAngularAccxint(length(Time)-1);
        KneeAngularAccx(i) = KneeAngularAccxint(length(Time)-1);
        HipAngularAccx(i) = HipAngularAccxint(length(Time)-1);
    else
        AnkleAngularAccx(i) = AnkleAngularAccxint(i-1);
        KneeAngularAccx(i) = KneeAngularAccxint(i-1);
        HipAngularAccx(i) = HipAngularAccxint(i-1);
    end
end

%% Bundling up the data
AnkleAngularVel = [AnkleAngularVelx (zeros(length(AnkleAngularVelx),1))
(zeros(length(AnkleAngularVelx),1))];
KneeAngularVel = [KneeAngularVelx (zeros(length(AnkleAngularVelx),1))
(zeros(length(AnkleAngularVelx),1))];
HipAngularVel = [HipAngularVelx (zeros(length(AnkleAngularVelx),1))
(zeros(length(AnkleAngularVelx),1))];
AnkleAngularAcc = [AnkleAngularAccx (zeros(length(AnkleAngularVelx),1))
(zeros(length(AnkleAngularVelx),1))];
KneeAngularAcc = [KneeAngularAccx (zeros(length(AnkleAngularVelx),1))
(zeros(length(AnkleAngularVelx),1))];
HipAngularAcc = [HipAngularAccx (zeros(length(AnkleAngularVelx),1))
(zeros(length(AnkleAngularVelx),1))];
```

```
end
```

```
function [BushingIntDia, BushingExtDia, BusingLength] =
BushingSizing(ShaftDiameter)
%This function determines the bushing dimensions from the base shaft
%diameter. Specifications are taken from THX-Prelubricated Bushings from
%Daemar Inc.

%Table has 3 columns: Interior Diameter, Exterior Diameter, Length
Table = [0 0 0;
         8  10 8;
         10 12 10;
         12 14 10;
         15 17 10;
         16 18 15;
         18 20 15;
         20 23 15;
         22 25 15;
         24 27 15;
         25 28 20;
         28 31 30;
         30 34 20;
         32 36 20;
         35 39 20;
         37 41 20;
         40 44 20;
         45 50 20;
         50 55 40;
         60 65 40;
         65 70 40;
         70 75 40;
         75 80 40;
         80 85 40;
         90 95 40;
         95 100 60;
         100 105 60;
         105 110 60;
         110 115 60;
         115 120 70;
         120 125 60;
         130 135 60;
         135 140 60;
         140 145 60;
         150 155 60;
         160 165 60;
         170 175 60;
         180 185 60;
         190 195 60;
         200 205 60;
         220 225 60;
         240 245 60;
         250 255 60;
         260 265 60;
         280 285 60;
         300 305 60];
Table = Table/1000;
```

```
%Bushing Selection
     Go = 0; i=1;
while Go == 0
    if i < 46
        if ShaftDiameter > Table(i,1) && ShaftDiameter < Table(i+1,1)
         Go = 1;
        else
            i=i+1;
        end
    else
        Go = 1;
        error('No bushings found')
    end
end
BushingIntDia = Table(i+1,1);
BushingExtDia = Table(i+1,2);
BusingLength = Table(i+1,3);
end
```

```
function [Fnormal,Fvertical,Hbelt,Tbelt] = Belts(Circumference,
ContactWidth, Hsegment, Type)
%This function calculates the dimensions and the forces involved in the
%belts used on the thigh. The theory is based on weight carrying capacity
%for backpacks of 40bls or 18kg.
%Circumference: Circumference of the limb interested
%ContactWidth: Width of the contact area between the user and the device
at
%the belt location
%Hsegment: Height of the segment where the belt is attached
%Type: 1 for the waist calculations, 2 for the other locations

%% Initializing
Pressure = 8000; %Maximum pressure on the skin before irritation in
Pascals
Tbelt = 1.5/100; %Thickness of belt in meters
Radius = Circumference/pi/2; %Radius of the segment assuming circular form
WeightSupp = 18.1437; % Supported weight in kg from a realistic bag belt
TensionWaist = 100 ; % Tension in waist belt found from literature review
in Newton

%% Calculating dimensions
if Type == 1
    % Waist Belt Dimension
    Hbelt = TensionWaist/(Pressure*Radius);
elseif Type == 2
    % Other Location Belt dimensions
    Hbelt = ceil(Hsegment*100/20)/100; %Height of belt in meters
end

%% Forces in play
Tension = Pressure*Radius*Hbelt; %Tension in the belt
Fnormal = 2*Pressure*Hbelt*Radius; %Normal force from contact on device
Fvertical = (WeightSupp*Tension/TensionWaist)*9.81; %Vertical force
supported by the belt
end
```

```
function [FbpmSplit, ktotal, Do, Lo, Bladder, Strand, Rabpm] =
BPMForces4(BW, AnkleAngle, GaitCycle, ShankLowDepth, ShankSupportDepth,
ShankSupportWidth, ShankBPMAnchorDepth, ShankHighHeight,
ShankSupportHeight, BaseAnkleRotationLength, AnkleBPMWidth,
AnkleBPMBackWards, AnkleHeight, BaseBackHeight, AnkleBPMHeight)
%This function calculates the maximum force required from the BPM to size
%the BPM with regards to the total weight of the individual.


%% Initialization
%We are comparing the force at the length when the GaitCycle = 0.56.
ind = find(GaitCycle == 0.48);

%% Muscle length
%The maximum muscle length corresponds to the length between the
attachment
%points on the Shank and Ankle Segments. The muscle will have to be
smaller
%than that length to account for attachment mechanics and to reduce the
%muscle force, in order to cater to the user.
LBPM = zeros(length(AnkleAngle), 3);
Rabpm = zeros(length(AnkleAngle), 3);
for i=1:length(AnkleAngle)
    %Vector of muscle
    LBPM(i,:)  = [0 (ShankLowDepth/2-ShankSupportDepth-ShankSupportWidth-
ShankSupportHeight) (ShankHighHeight+ShankSupportHeight*2)] - [0 (-
BaseAnkleRotationLength-AnkleBPMWidth-AnkleBPMBackWards) (-
AnkleHeight+BaseBackHeight+AnkleBPMHeight)]*[1 0 0; 0 cos(AnkleAngle(i))
sin(AnkleAngle(i)); 0 -sin(AnkleAngle(i)) cos(AnkleAngle(i))];

    %Vector of ankle shaft to muscle anchor in the ankle segment
    Rabpm(i,:) = [0 (-BaseAnkleRotationLength-AnkleBPMWidth-
AnkleBPMBackWards) (-AnkleHeight+BaseBackHeight+AnkleBPMHeight)]*[1 0 0; 0
cos(AnkleAngle(i)) sin(AnkleAngle(i)); 0 -sin(AnkleAngle(i))
cos(AnkleAngle(i))];
end

%Muscle direction upward from ankle (unit vector)
RBPMx = LBPM(:,1)./(LBPM(:,1).^2 + LBPM(:,2).^2 + LBPM(:,3).^2).^(1/2);
RBPMy = LBPM(:,2)./(LBPM(:,1).^2 + LBPM(:,2).^2 + LBPM(:,3).^2).^(1/2);
RBPMz = LBPM(:,3)./(LBPM(:,1).^2 + LBPM(:,2).^2 + LBPM(:,3).^2).^(1/2);
RBPM = -[RBPMx RBPMy RBPMz];

%% Maximum Human Ankle Moment from Loading Response to Mid-Stance
%From literature review, the maximum ankle moment during gait between the
%loading response and the terminal stance is about 1.461*BW Nm at the
ankle
%in the sagittal plane at about 56% of the gait cycle. Lets reduce the
%moment by a tenth due to BPM contribution
M = [1.461*BW 0 0];

%If we assume that the maximum ankle moment is to be replaced by
%the BPM then the force of the BPM can be found.
FTentative = abs((M*0.25)/(cross(Rabpm(ind,:), RBPM(ind,:))));
Fmin = 0.7*FTentative(1);
```

```
Fmax = 1.3*FTentative(1);

%% Initial guess of the BPM length
%The BPM has to have minimal force when the ankle angle is near 0.
Lactive = zeros(length(GaitCycle),1);
for i = 1:length(GaitCycle)
    if GaitCycle(i) <= 0.48
        Lactive(i) = (LBPM(i,1).^2 + LBPM(i,2).^2 + LBPM(i,3).^2).^(1/2);
    else
        Lactive(i) = 0;
    end
end
%Let the muscle be a 75% shorter than the available space. It will be
%attached via cables for the rest of the space.
Lmin = 0.75*min(Lactive(Lactive~=0));

%% BPM sizing loop
Go = 0; j=1; %Loop control
while Go == 0
    % Muscle Length at inflated rest
    Lo = Lmin;
    Ly = Lo/cos(48*pi/180); %Braid length
    % Elongation at maximum force
    Elongation = max(Lactive(Lactive~=0)) - min(Lactive(Lactive~=0));

    % Muscle Length at the maximum force
    Lmax = Lmin + Elongation;

    % Maximum force occurs at the toe off when the muscle is being
released
    MuscleMove = 2;

    % BPM Muscle Force
    [Force, ~, ~, ~, ~] = BPMForceLength2(Lo, Lmax, MuscleMove);

    %Test to get out of sizing loop
    if Force>=Fmin && Force<=Fmax && Lmin <= 0.95*min(Lactive(Lactive~=0))
        Go = 1;
    elseif j == 10000
        fprintf('Could not find a solution')
        error('No solutions were found')
    elseif Force == 0 || Force < Fmin
        Lmin = Lmin + 0.01*min(Lactive(Lactive~=0));
    elseif Lmin > 0.95*min(Lactive(Lactive~=0)) || Lmin<=
0.05*min(Lactive(Lactive~=0));
        fprintf('Could not find a solution within the range of length')
         error('No solutions were found')
    elseif Force > Fmax
        Lmin = Lmin + 0.01*min(Lactive(Lactive~=0));
    end
    j=j+1;
end

%% Set the forces throughout the cycle
Fbpm = zeros(length(GaitCycle),1);
```

```
ktotal = zeros(length(GaitCycle),1);
Length = Lmin + ((LBPM(:,1).^2 + LBPM(:,2).^2 + LBPM(:,3).^2).^(1/2)) -
((LBPM(1,1).^2 + LBPM(1,2).^2 + LBPM(1,3).^2).^(1/2));
for i = 1:length(Length)
    if GaitCycle(i)<=0.48
            MuscleMove=1;
        [Fbpm(i), ktotal(i), Do, Bladder, Strand] = BPMForceLength2(Lo,
Length(i), MuscleMove);
    elseif GaitCycle(i) <= 0.62
        MuscleMove=2;
        [Fbpm(i), ktotal(i), Do, Bladder, Strand] = BPMForceLength2(Lo,
Length(i), MuscleMove);
    else
        Fbpm(i) = 0; ktotal(i) = 0;
    end
end

%% Seperating the muscle force into components (local reference frame)
FbpmSplit(:,1) = Fbpm.*RBPM(:,1);
FbpmSplit(:,2) = Fbpm.*RBPM(:,2);
FbpmSplit(:,3) = Fbpm.*RBPM(:,3);
end
```

```
function [Force, ktotal, Do, Bladder, Strand] = BPMForceLength2(Lo, L,
MuscleMove)
%%This function calculates the theoretical force and stiffness models for
%BPM undergoing eccentric contractions. The proposed models takes into
%account the geometry, bladder-braid friction and inter-braid friction

%The output are the BPM force, diameter, internal bladder and strand used

%% Set up values
Bladder = 1; %Bladder Material: 1 Butyle, 2 Silicon, 3 Balloon'
ng = 1.0; %Polytropic Coefficient: 1 to 1.4
Strand = 1; %TechFlex Product: 1 for FlexoOverexpanded, 2 for FlexoPet

%% Friction coefficient
% From absence of data, assume the friction coefficient to be 0.3 for both
% friction mechanisms
uff = 0.3;%friction coefficient for fiber-fiber friction
ufb = 0.3;%friction coefficient for fiber-bladder friction

%% Braid Information
df = 0.01*2.54/100; %Braid diameter is irrelevant of the braid used

if Strand == 1
    Do = 1.3/100; %Base diameter of Overexpanded Pet
    N = 120; %Base diameter of Overexpanded Pet
elseif Strand == 2
    Do = 1.9/100; %Base diameter of Flexo Pet
    N = 3*72; %Base diameter of Flexo Pet
end

%% Geometric Properties
%Static Geometric Properties. Assume rest angle is approx 48 degrees
(measured)
```

```
Ly = Lo/cos(48*pi/180); %Braid length
n = Ly*sin(20*pi/180)/(Do*pi); %Number of braid revolutions

%% Pressure Formulas
% Initial condition occur when the muscle is inflated initialy but not
% external forces are applied.
Lss = Lo; %Length at initial pressure
Pss = 100000; %Initial Pressure

%Pressure is assumed to change according to a polytropic process where
ng=1
%for an isothermal process and ng=1.4 for an adiabatic process
P = Pss*((Lss*Ly^2-Lss^3)/(L*Ly^2-L^3))^ng;

%Pressure derivative assuming the polytropic process
dPdL = -Pss*ng*(Ly^2-3*L^2)/(Lss*Ly^2-Lss^3);

%% Stiffness's
%Geometric stiffness
kg = -dPdL*(Ly^2-3*L^2)/(4*pi*n^2) + 3*P*L/(2*pi*n^2);

%Fibre-Fibre frictional stiffness
kff = uff*((df^2)*(Ly^2)/((Ly^2-L^2)^(1/2)))*(N/4)*(n*N+1)*(dPdL/L - P/L^2
+ P/(Ly^2-L^2));

%Fibre-Bladder friction stiffness
kfb = ufb*df*Ly*dPdL - ufb*((df^2)*(Ly^2)/((Ly^2-
L^2)^(1/2)))*(N/8)*(n*N+1)*(dPdL/L - P/L^2 + P/(Ly^2-L^2));

if MuscleMove == 1
    ktotal = kg - kff - kfb;
elseif MuscleMove == 2
    ktotal = kg + kff + kfb;
end

%% BPM Force
if L > Ly
    Force = 0;
else
    Force = ktotal*(L-Lss);
end
end
```

```
function [RatchetWidth, RatchetDia, TeethNumber, TeethAngle,
RatchetHoleDiameter, MountDistance, CenterDistance, PawlRadius,
PawlLength, PawlBore, PitchAngle, TeethDepth, PawlWidth] =
RatchetDesign(AnkleHeight, MomentNm, SUTMPA)
% This function sizes the ratchet based on the Hankbook of Machining and
% Metalworking Calculations by Ronald A. Walsh

%Teeth and Teeth Factor
TeethNumber = 100; %50 teeth gives a precision of 3.6 degrees which is
acceptable
if TeethNumber <= 12
    Teethfactor = 50;
elseif TeethNumber < 20
```

```
    Teethfactor = 35;
elseif TeethNumber >= 20
    Teethfactor = 20;
end

%Initial guess Circular Pitch
MaxDia = AnkleHeight;
MaxDiaIP = (MaxDia*100)/2.54;
Pitch_Max = pi*MaxDiaIP/TeethNumber;
Pitch_Min = Pitch_Max/2;

%Maximum bending stress (1/3 of tensile stress)
Ss = (SUTMPA*145.0377)/3;

%Convert Moment from Nm to lb*in
MomentIP = MomentNm*8.85074576738;

%% Minimizing the RatchetWidth with the pitch
[Pitch,RatchetWidthIP] =
fminbnd(@(Pitch)myfun(Teethfactor,MomentIP,TeethNumber,Ss,Pitch),Pitch_Min
,Pitch_Max);

%% Convert dimensions to Metric
RatchetWidth = RatchetWidthIP*2.54/100; %Face Length in meters
RatchetDia = (Pitch*TeethNumber/pi)*2.54/100; %Ratchet Diameter in meters

%% Get the other dimensions
%Let the friction coefficient be 0.25
TeethAngle = atan(0.25);
PitchAngle = 2*pi/TeethNumber;

%Ratchet Hole diameter
RatchetHoleDiameter = RatchetDia/6;

%Pawl Dimensions
PawlWidth = RatchetWidth;
PawlRadius = RatchetDia/8;
PawlLength = RatchetDia*1/2 + PawlRadius;
PawlBore = PawlRadius;

%Center Distance and Mounting Distance
CenterDistance = ((RatchetDia/2)^2 + (PawlLength-PawlRadius)^2)^(1/2);
MountDistance = RatchetDia/2;

%Teeth
TeethDepth = (RatchetDia/2)*PitchAngle*(2*sin(pi/2 - pi/4 -
TeethAngle)/(sin(pi/2 - pi/4 - TeethAngle) +
2*sin(pi/4)))/cos(TeethAngle);

%% Convert the angles to degrees
TeethAngle = TeethAngle*180/pi;
PitchAngle = PitchAngle*180/pi;
end
```

```
function F = myfun(Teethfactor,MomentIP,TeethNumber,Ss,Pitch)
F = Teethfactor*MomentIP/(TeethNumber*Ss*Pitch^2); %Face Length in IP
end
```

```
function [Fpawlprime, Fpawl] = RatchetForces(GaitCycle, Dratchet,
TeethDepth, Mankle)
%This function calculates the forces on the ratchet teeth through one gait
%cycle. It is based on the geometry of the ratchet, which is centered with
%the ankle joint: the ankle accelerations are applicable to the ratchet.

%% Initialization
%Friction
us = 0.25;
%Forces
Fpawl = zeros(length(GaitCycle),3);
Fpawlprime = zeros(length(GaitCycle),3);
%Ratchet dimensions
RootRadius = (1/2)*Dratchet - TeethDepth/2;

%% Forces calculations
Fpawl(:,2) = Mankle(:,1)/(RootRadius + TeethDepth/2);
Fpawl(:,3) = Fpawl(:,2)*us;

%% Transfer to the reference frame of the ankle segment
for i=1:length(GaitCycle)
Fpawlprime(i,:) = Fpawl(i,:)*[1 0 0; 0 1 0; 0 0 1];
end
end
```

```
function [KeyWidth, KeyHeight,KeyDepth] = KeyChooser( ShaftDiameter )
%Calculates the Key used for the desired shaft diameter
IPDiameter = ShaftDiameter*1000/25.4; %Converts m in inches

%Table 7-6 in Shigley [Width, Height, KeywayDepth]
KeySize = [1/32 1/32 1/64;
            3/32 3/32 3/64;
            1/8 3/32 3/64;
            3/16 1/8 1/16;
            1/4 3/16 3/32;
            5/16 1/4 1/8;
            3/8 1/4 1/8;
            1/2 3/8 3/16;
            5/8 7/16 7/32;
            3/4 1/2 1/4];

if IPDiameter<=3/16
    Row = 1;
elseif (IPDiameter>3/16)&&(IPDiameter<=5/16)
    Row = 1;
elseif (IPDiameter>5/16)&&(IPDiameter<=7/16)
    Row = 2;
elseif (IPDiameter>7/16)&&(IPDiameter<=9/16)
    Row = 3;
elseif (IPDiameter>9/16)&&(IPDiameter<=7/8)
    Row = 4;
elseif (IPDiameter>7/8)&&(IPDiameter<=1.25)
    Row = 5;
elseif (IPDiameter>1.25)&&(IPDiameter<=1.375)
```

```
    Row = 6;
elseif (IPDiameter>1.375)&&(IPDiameter<=1.75)
    Row = 7;
elseif (IPDiameter>1.75)&&(IPDiameter<=2.25)
    Row = 8;
elseif (IPDiameter>2.25)&&(IPDiameter<=2.75)
    Row = 9;
elseif (IPDiameter>2.75)&&(IPDiameter<=3.25)
    Row = 10;
end

KeyWidth = KeySize(Row, 1)*25.4/1000;
KeyHeight = KeySize(Row, 2)*25.4/1000;
KeyDepth = KeySize(Row, 3)*25.4/1000;
end
```

```
function [SplineWidth, SplineHeight,SplineDepth, SplineNumber,
SplineLength] = SplineChooser( ShaftDiameter, SegmentLength )
%Calculates the spline used for the desired shaft diameter
% Data taken from Mark's Standard Handbook. Only the 4 spline is
considered
% for the case at hand

%% Spline Initialization
%Spline table: MinDiameter MaxDiameter SplineWidth SplineHeight
SplineDepth
Splines = zeros(13,5);

%Max Diameter
Splines(:,2) = [0.75 0.875 1 1.125 1.25 1.375 1.5 1.625 1.75 2 2.25 2.5
3];
%Min Diameter (Permanent fit)
Splines(:,1) = Splines(:,2)*0.850;
%Spline Depth (Permanent fit)
Splines(:,5) = Splines(:,2)*0.075;
%Spline Height ((Permanent fit)
Splines(:,4) = Splines(:,5)*2;
%Spline Width
Splines(:,3) = Splines(:,2)*0.241;
%Convert to meters
Splines = Splines*25.4/1000;

%% Spline Sorting
Go = 0 ; i =1;
while Go == 0 && i <= 13
    if i == 1
        if ShaftDiameter > 0 && ShaftDiameter <= Splines(i,1)
            Go = 1;
        elseif ShaftDiameter < 0
            error('Shaft is negative')
        else
            i=i+1;
        end
    else
        if ShaftDiameter > Splines(i-1,1) && ShaftDiameter <= Splines(i,1)
            Go = 1;
        else
```

```
            if i < 13
                i=i+1;
            elseif i==13
                error('Shaft is larger than largest tabulated shafts')
            end
        end
    end
end

%% Spline Selection
SplineWidth = Splines(i,3);
SplineHeight = Splines(i,4);
SplineDepth = Splines(i,5);
SplineNumber = 4;
SplineLength = SegmentLength;
end
```

```
function [ Length, N ] = KeyLength(Torque, ShaftDiameter, SegmentLength,
KeyWidth )
%Calculates the key length required based on calculation shown in
Shigley's
%mechanical engineering design

Force = 2*Torque/ShaftDiameter;

%Initial Number of Keys
N = 2;

%All Keys are assumed to be made of AISI 1015 Steel CD
Sy = 320000000; %Yield Strength in Pa
Ssy = Sy*0.577; %Shear Strength in Pa

%Safety Factor assumed at 1.5
n=1.5;
Selector = 0;
counter = 0;

while Selector == 0 && counter<500
    LengthDistort = Force*n/(Ssy*KeyWidth*N);%Distortion energy
    LengthCrush = Force*2*n/(Sy*KeyWidth*N);%Crushing

    %Decision Length
    if max(LengthDistort, LengthCrush)< SegmentLength
        Length = SegmentLength;
        Selector = 1;
    else
        N = N+1;
    end
    counter = counter +1;
    if counter == 8
        disp ' key length is not working'
    end
end
end
```

```
function [ InternalDiameter, ExternalDiameter, Width ] =
RetainingRingSizing(ShaftDiameter)
```

```
%Retaining Ring Sizing

InternalDiameter = ShaftDiameter - ShaftDiameter/16;
ExternalDiameter = ShaftDiameter + ShaftDiameter/8;

Thickness = [0.4 0.6 0.7 0.8 1 1.2 1.5 1.75]/1000; %Thickness of retaining
ring based on EMILE MAURIN Parts

if ShaftDiameter<=(4/1000)
    Col = 1;
elseif ShaftDiameter<=(5/1000)
    Col = 2;
elseif ShaftDiameter<=(6/1000)
    Col = 3;
elseif ShaftDiameter<=(8/1000)
    Col = 4;
elseif ShaftDiameter<=(17/1000)
    Col = 5;
elseif ShaftDiameter<=(26/1000)
    Col = 6;
elseif ShaftDiameter<=(35/1000)
    Col = 7;
elseif ShaftDiameter<=(48/1000)
    Col = 8;
end

Width = Thickness(Col);
end
```

```
function [CamProfile] = CamSizer3(RatchetHoleDiameter, PawlLengthLong,
PawlRadius, FollowerMax)
%This function sizes the cam to obtain a simple harmonic motion over the
%desired range of motion for the pen click mechanism

%% Initializing
Points = 360;
%Set the Base Radius
PawlLength = PawlLengthLong - PawlRadius;
BaseRadius = RatchetHoleDiameter*3/4;
Theta = ceil(atan((PawlLength/2)/BaseRadius)*180/pi);

%% Frofile
Profile = zeros(Points+1, 1);
for i=0:Points+1
    if i >= 1 && i < 91
        Profile(i) = BaseRadius;
    elseif i >= 91 && i < (90+Theta)
        Profile(i) = BaseRadius/cos((i-90)*pi/180);
    elseif i >= (90+Theta) && i < (90+Theta+14)
        k = (90+Theta);
        Profile(i) = ((BaseRadius + (FollowerMax/2)*(1-cos(pi*(i-
k)/(14))))^2 + (BaseRadius*tan((i-90)*pi/180))^2)^0.5;
    elseif i >= (90+Theta+14) && i < (270-Theta)
        L1 = (Profile(90+Theta+14-1)^2 - BaseRadius^2)^0.5;
        L3 = (PawlLength/2);
        L2 = L3 - L1;
        MaxR = ((BaseRadius + (FollowerMax/2)*(1-cos(pi)))^2 + L2^2)^0.5;
        MinR = BaseRadius;
        Radius = (MaxR-MinR)*(i-(270-Theta))/((90+Theta+14)-(270-Theta)) +
MinR;
        ang1 = asin((PawlLength/2)*sin((i-180)*pi/180)/Radius)*180/pi;
        ang2 = 180 - (i-180) - ang1;
        Profile(i) = ((PawlLength/2)^2 + Radius^2 -
2*(PawlLength/2)*Radius*cos(ang2*pi/180))^0.5;
    elseif i >= (270-Theta) && i < 270
        Profile(i) = BaseRadius/cos((270-i)*pi/180);
    elseif i >= 270 && i <= 361
        Profile(i) = BaseRadius;
    end
end

%% Cam
%Set the CamProfile
CamProfile = zeros(Points+1, 3);
for i=1:Points+1
CamProfile(i,1) = Profile(i)*cos(i*pi/180);
CamProfile(i,2) = Profile(i)*sin(i*pi/180);
CamProfile(i,3) = 0;
end
%plot(CamProfile(:,1),CamProfile(:,2))
end
```