

Mechanical Design of the Legs for OLL-E, a Fully Self-Balancing, Lower-Body Exoskeleton

Bradford Asin Wilson

Thesis submitted to the faculty of the Virginia Polytechnic Institute and State University in
partial fulfillment of the requirements for the degree of

Master of Science

In

Mechanical Engineering

Alan T. Asbeck, Chair

Alexander Leonessa

Steve Southward

July 26, 2019

Blacksburg, VA

Keywords: Mechanical Design, Linkage Design, Gait Analysis, Humanoid Robot, Exoskeleton,
Finite Element Analysis

Mechanical Design of the Legs for OLL-E, a Fully Self-Balancing, Lower-Body Exoskeleton

Bradford Asin Wilson

Academic Abstract

Exoskeletons show great promise in aiding people in a wide range of applications. One such application is medical rehabilitation and assistance of those with spinal cord injuries. Exoskeletons have the potential to offer several benefits over wheelchairs, including a reduction in the risk of upper-body injuries associated with extended wheelchair use. To fully mitigate this risk of injury, exoskeletons will need to be fully self-balancing, able to move and stand without crutches or other walking aid. To accomplish this, the Orthotic Lower-body Locomotion Exoskeleton (OLL-E) will actuate 12 Degrees of Freedom, six in each leg, using custom design linear series elastic actuators. The placement of these actuators relative to each joint axis, and the geometry of the linkage connecting them, were critical to ensuring each joint was capable of producing the required outputs for self-balancing locomotion. In pursuit of this goal, a general model was developed, relating the actuator's position and linkage geometry to the actual joint output over its range of motion. This model was then adapted for each joint in the legs and compared against the required outputs for humans and robots moving through a variety of gaits. This process allowed for the best placement of the actuator and linkages within the design constraints of the exoskeleton. The structure of the exoskeleton was then designed to maintain the desired geometry while meeting several other design requirements such as weight, adjustability, and range of motion. Adjustability was a key factor for ensuring the comfortable use of the exoskeleton and to minimize risk of injury by aligning the exoskeleton joint axes as close as possible to the wearer's joints. The legs of OLL-E can accommodate users between 1.60 m and 2.03 m in height while locating the exoskeleton joint axes within 2 mm of the user's joints. After detailed design was completed, analysis showed that the legs met all long-term goals of the exoskeleton project.

Mechanical Design of the Legs for OLL-E, a Fully Self-Balancing, Lower-Body Exoskeleton

Bradford Asin Wilson

General Audience Abstract

Exoskeletons show great promise in aiding people in a wide range of applications. One such application is medical rehabilitation and assistance of those with spinal cord injuries. Exoskeletons have the potential to offer several benefits over wheelchairs, including a reduction in the risk of upper-body injuries associated with extended wheelchair use. To best reduce this risk of injury, exoskeletons will need to be fully self-balancing, able to move and stand without crutches or relying on any other outside structure to stay upright. To accomplish this, the Orthotic Lower-body Locomotion Exoskeleton (OLL-E) will use a set of custom designed motors to apply power and control to 12 joints, six in each leg. Where these motors were placed, and how they connect to the joints they control, were critical to ensuring the exoskeleton was able to self-balance, walk, and climb stairs. To find the correct position, a set of equations was developed to determine how different positions changed each joints' speed, strength, and range of motion. These equations were then put into a piece of custom software that could quickly evaluate different joint layouts and compare the capabilities against measurements from people and robots walking, climbing stairs, and standing up out of a chair. This process allowed for the best placement of the motors and joints while still keeping the exoskeleton relatively compact. The rest of the exoskeleton was then designed to connect these joints together, while meeting several other design requirements such as weight, adjustability, and range of motion. Adjustability was very important for ensuring the comfortable use of the exoskeleton and to minimize risk of injury by ensuring that the exoskeleton legs closely matched the movements of the person inside. The legs of OLL-E can accommodate users between 1.60 m and 2.03 m in increments of 7 mm. After detailed design was completed, additional analyses were performed to check the strength of the structure and ensure it met other long-term goals of the project.

Academic Abstract	ii
General Audience Abstract	iii
List of Abbreviations	v
Tables	vi
Figures	vii
1. Introduction	1
1.1. Overview/Motivation	1
1.2. Summary of Previous Work	3
1.3. Thesis Organization	4
2. Requirements	5
3. Joint Output Modeling	9
3.1. Basics	9
3.2. Derivation	10
3.3. Linkage Geometry Refinement	12
3.3.1. Results	20
3.4. Gait Cycle Comparison	22
3.4.1. Gait Cycle	22
3.4.2. Script Behavior	25
3.4.3. Results	26
4. Design	35
4.1. Upper Leg	36
4.2. Lower Leg	40
4.3. Ankle/Foot	45
4.4. Joints	47
5. Finite Element Analysis	49
5.1. Upper Leg	49
5.2. Lower Leg	52
6. Conclusion	58
6.1. Future Work	58
References	59
Appendix A: LeverArmCalc	61
Appendix B: GaitCycleCheck	67

List of Abbreviations

DOF: Degree of Freedom

FEA: Finite Element Analysis

OLL-E: Orthotic Lower-body Locomotion Exoskeleton

STS: Sit-to-Stand

CotS: Commercial-off-the-Shelf

Tables

Table 1: Typical limits for joint range of motion [8].	7
Table 2: Fractional and typical segment lengths [11]. Percentiles are for a 50/50 mixed population of men and women.	7
Table 3: Derived requirements for peak joint power, torque, and velocity [9].	8
Table 4: Joint geometries selected by initial analysis	21
Table 5: Final joint geometry values from the second analysis.	34
Table 6: Parts and materials used in each upper leg subassembly.	37
Table 7: Parts and materials used in the knee joint assembly.	41
Table 8: Parts and materials used in the lower leg shin assembly.	43
Table 9: Foot parts and materials.	46
Table 10: Ankle parts and materials.	47
Table 11: Peak stress in main components across all load cases.	52
Table 12: Summary of FEA Results of the lower leg analysis	55

Figures

Figure 1: SuitX's Phoenix (left) [5] and Cyberdyne's HAL (right) [6]. Both exoskeletons require walking aids to balance. 2

Figure 2: Exoskeleton leg joint layout, showing alignment with human leg joint axes. The dashed line indicates the human joint axis, and a ringed circle indicates a joint in-plane..... 6

Figure 3: Simplified linkage diagram. 10

Figure 4a: Linkage lengths used in derivation..... 10

Figure 5: Reference points overlaid onto knee actuator assembly (some components hidden). . 11

Figure 6: Joint output colormaps as a function of θ (x-axis) and r (y-axis). In the top plots, solid lines define the region inside which the output exceeds the peak requirement established by Kendrick. The dashed line shows the shape of that boundary for the other output, indicating a region in which both requirements are satisfied. The horizontal dotted line shows the selected value of r used in the next set of plots, and the vertical dotted line was used as a reference to where peak requirements were expected..... 13

Figure 7: Color map of outputs for the knee joint. Note how little overlap exists between the regions of sufficient torque and speed, as well as the significant negative bias of the u-joint angle. 14

Figure 8: Joint output color map for ankle pitch, using the combined output of two MkIII Lite actuators. 15

Figure 9: Joint output colormap for ankle roll. 16

Figure 10: Joint output as a function of theta for a specific lever arm length. As before, the vertical dotted line indicates the expected angle of peak need. The horizontal dotted line indicates the required maximum speed or torque as defined by Kendrick. 17

Figure 11: Knee pitch outputs for the selected lever arm length. The 90° angle of peak torque represents standing from a seated position, not from the typical walking cycle. 18

Figure 12: Joint output for the selected lever arm length. 19

Figure 13: Joint output for the selected lever arm length. 20

Figure 14: Space allocation within y dimension..... 21

Figure 15: Sample gait cycle, showing joint angle as a function of gait percentage [9]. 23

Figure 16: Sample gait cycle showing joint velocity as a function of gait percentage [9]. 23

Figure 17: Sample gait cycle showing joint torque as a function of gait percentage. Torque has been scaled to that of a 140kg person, the maximum combined mass of the exoskeleton and user [9].	24
Figure 18: Gait cycle comparison for hip pitch. The vertical dotted lines indicate the joint angle limits.	27
Figure 19: Gait cycle comparison for knee pitch.	28
Figure 20: Gait cycle comparison for the front ankle actuator on the ankle pitch axis.	29
Figure 21: Gait cycle comparison for the rear ankle actuator on the ankle pitch axis.	30
Figure 22: Gait cycle comparison for the front ankle actuator on the ankle roll axis.	31
Figure 23: Gait cycle comparison for the rear ankle actuator on the ankle roll axis.	32
Figure 24: The whole exoskeleton leg with subassembly and joints labeled.	35
Figure 25: Exploded view of the upper leg segment.	37
Figure 26: Adjustability of the upper leg, from 393mm to 498mm.	39
Figure 27a: Exploded view of the outside of the knee joint assembly.	40
Figure 28: Exploded view of the shin assembly.	42
Figure 29: Actuator package length as a function of ankle pitch and roll for the front (top) and rear (bottom) actuators. The solid black lines indicate the maximum and minimum lengths for the actuator, while the dashed lines indicate those same limits for the opposing actuator.	44
Figure 30: Exploded view of the foot assembly (foot plate hidden).	46
Figure 31: Ankle exploded view.	47
Figure 32: Exaggerated displacement of the upper leg assembly in the worst-case loading.	50
Figure 33: Von mises stress distribution in the worst-case loading. (Some components hidden.)	51
Figure 34: Close-up of stress concentration near the knee joint. (Some components hidden.)	51
Figure 35: Exaggerated displacement of the lower leg assembly.	53
Figure 36: Von mises stress distribution in the lower leg assembly. (Some components hidden.)	54
Figure 37: Close up of peak stresses in knee joint housing and shin body.	55
Figure 38: Overall von Mises stress distribution in the modified knee joint housing.	56
Figure 39: Closeup of contact stress from the actuator pin on the modified knee joint housing.	56
Figure 40: Displacement in the modified knee joint housing.	57

1. Introduction

In recent years, exoskeletons have increasingly been the focus of research and development efforts, employing different technologies for a variety of applications in many different industries. The goal is simple: provide a structure around the user to enhance their capabilities and make tasks easier. Depending on the application, this can be done in a variety of ways. Some exoskeletons are passive and aim to ease the strains of repetitive motions, using springs or some other way of storing energy during one motion to release it in another. Others are active, using motors, hydraulics, or some other system to provide power at various joints, making them stronger and more versatile at the cost of complexity.

One promising application for exoskeletons is in the medical industry. For example, exoskeletons can be used for physical therapy, guiding the patient's motion in specific ways while carrying some of the load. Exoskeletons can also go farther, carrying a larger portion of the load and supplying more of the power to assist those who cannot support their own weight as they go about their daily lives. This would be a great benefit for those with limited mobility, who are typically dependent on wheelchairs. While simple and functional, extended wheelchair use has several drawbacks. Users are prone to ailments such as sores [1], obesity, and reduced bone density caused by the extended time sitting in one position [2]- [3]. Also, the repetitive exertions of manual wheelchairs can injure the user's upper body, especially in the shoulders [4]. And while wheelchairs offer some improvements in mobility, their users are still limited in what they can do, and expensive modifications are often needed to make the home or office more accessible. Tasks that are trivial for a healthy individual, such as climbing stairs or reaching a high shelf, can be difficult for someone confined to a wheelchair. Medical exoskeletons mediate or eliminate several of these problems. Returning the user to a standing position extends their reach and enables them to handle a wider range of obstacles, while having them carry some portion of their weight can mitigate obesity or loss of bone density.

1.1. Overview/Motivation

Before they can be widely used, medical exoskeletons have several challenges that need to be overcome. One such challenge is the inability of the wearer to maintain balance without assistance. Many exoskeletons, such as SuitX's PHOENIX [5], Cyberdyne's HAL [6], and Berkley's Austin

[7], are not fully self-balancing, meaning they require outside input to stay balanced. A healthy individual can provide that input themselves without any issues, but when the user does not have full use of their legs something else is needed. Many medical exoskeletons rely on crutches for balancing, but this limits their usefulness since the user needs to have enough strength and dexterity in their upper body to use crutches. It also reintroduces several of the drawbacks of wheelchairs, namely the risk of injury to the upper body, and occupies the user's hands, limiting their capabilities. Alternatively, an exoskeleton fully capable of keeping itself balanced would avoid these issues and provide a more versatile product.



Figure 1: SuitX's Phoenix (left) [5] and Cyberdyne's HAL (right) [6]. Both exoskeletons require walking aids to balance.

Medical exoskeletons also have additional challenges not present in other exoskeleton applications. Depending on the type of injury, the users may find it difficult or even impossible to guide the motion of the exoskeleton using their legs, requiring a less direct control interface. Another concern is the strength of the user. The weakened muscles and bones common in those dependent on wheelchairs (whether from the original injury or extended wheelchair use) means special care needs to be taken on how forces are transferred between the user and the exoskeleton

to prevent the device from injuring the patient. This issue is exacerbated by any loss of sensation related to the injury. Without any feeling in their limbs, the user may be slow to identify any injuries resulting from using the exoskeleton, such as chafing, pinching, or overextension.

The Orthotic Lower-body Locomotion Exoskeleton (OLL-E), being developed by the Virginia Tech Assistive Robotics Lab and sponsored by the National Science Foundation, aims to develop a fully self-balancing lower body exoskeleton for medical applications. This is being accomplished by adding actuation and control to six Degrees of Freedom (DOFs) in each leg (three in the hip, one in the knee, and two in the ankle). Specifically, adding actuation to the hip yaw and ankle roll axes greatly increases the capability of the exoskeleton to walk unaided. OLL-E will also need to address the other challenges mentioned, and will eventually incorporate a more natural, straight-legged gait. This portion of the project is focused on developing the mechanical design for the legs of the exoskeleton, including their actuation.

1.2. Summary of Previous Work

This thesis is building on the work of several others from this lab. Much of the groundwork comes from research by Xiao Li [8] and John Kendrick [9] among others in the Assistive Robotics Lab, led by Dr. Alan Asbeck. Li's work established many of the basic requirements, such as each joint's range of motion, the exoskeleton's basic kinematic structure, and outlined a general design strategy for the exoskeleton. His work also included the detailed design for the hips, providing actuation to the hip pitch and yaw axes as well as mounting for computers, batteries, and a central cooling system in a rigid backpack supported by the exoskeleton.

John Kendrick's work focused on designing the linear actuators that will be used throughout the exoskeleton. These actuators use small Maxon motors to drive high-strength precision ball screws which in turn provide torque at each joint. Most of these actuators are mounted on springs, adding compliance to the system which offers several advantages for a force-controlled system [10]. Where more power is needed, liquid cooling is used to greatly increase the performance of each motor, instead of using larger actuators at each joint. This approach moves the weight to a central cooling system in the backpack, reducing the total power at each joint needed to move the legs. Three actuator packages were designed to cover different bands of torque and speed, and each can be further customized by adjusting the internal gearing of the actuator.

1.3. Thesis Organization

This thesis focuses on the development of the mechanical design of the legs of the exoskeleton. Chapter 2 outlines the requirements developed by this and previous efforts. As described in Chapter 3, an actuator linkage was chosen and then modeled for each joint in question (hip and knee pitch, ankle pitch and roll). This modeling was then refined by analyzing several different gaits that the exoskeleton should be capable of (flat walking, stair climbing, and sit-to-stand). In Chapter 4, a structure was built around these joints, connecting them and aligning them with the user, while also incorporating the other design requirements. Finally, Chapter 5 discusses how the design was further refined and validated using finite element analysis (FEA).

2. Requirements

In the early stages of the project, several goals were established for OLL-E:

- OLL-E should be self-balancing, able to support a 70kg person without any outside assistance.
- OLL-E should be able to help this person walk around and climb stairs.
- OLL-E should walk at the same speed or faster than the humanoid robots at Virginia Tech.
- OLL-E should be adjustable, able to accommodate users with a range of statures
- OLL-E itself should have a mass of no more than 70kg.

These goals were then used to develop more detailed requirements for the exoskeleton. Self-balancing can be achieved by providing actuation and control to enough degrees of freedom in each leg. In OLL-E, each leg will have control over hip yaw, pitch, and roll, knee pitch, and ankle pitch and roll. To be able to walk and climb stairs, the joints of the exoskeleton will need a large enough range of motion and to provide enough power and torque to do so at the expected speeds. Additionally, the torque requirements must be sized on the 140kg combined mass of the system (70kg for the user, 70kg for the exoskeleton).

Since the exoskeleton surrounds a person, and that user moves with it, the exoskeleton must move in a way that closely matches the motion of the human body. This can be a challenge, since the human leg incorporates several joints that have multiple degrees of freedom. Specifically, the hip is a ball joint that can rotate around all three axes, and the ankle is a saddle joint, capable of bending in two axes. These joints can be emulated by using a series of 1-DOF pivot joints where each joint's axis of rotation is aligned with the corresponding human joint. The exoskeleton itself serves as a framework to keep these joints in alignment throughout the range of motion of the wearer. These 1-DOF pivots will be named according to the joint (hip, knee, or ankle) and axis (yaw, pitch, or roll). Figure 2 shows a rough layout of how the joints will be connected in the exoskeleton.

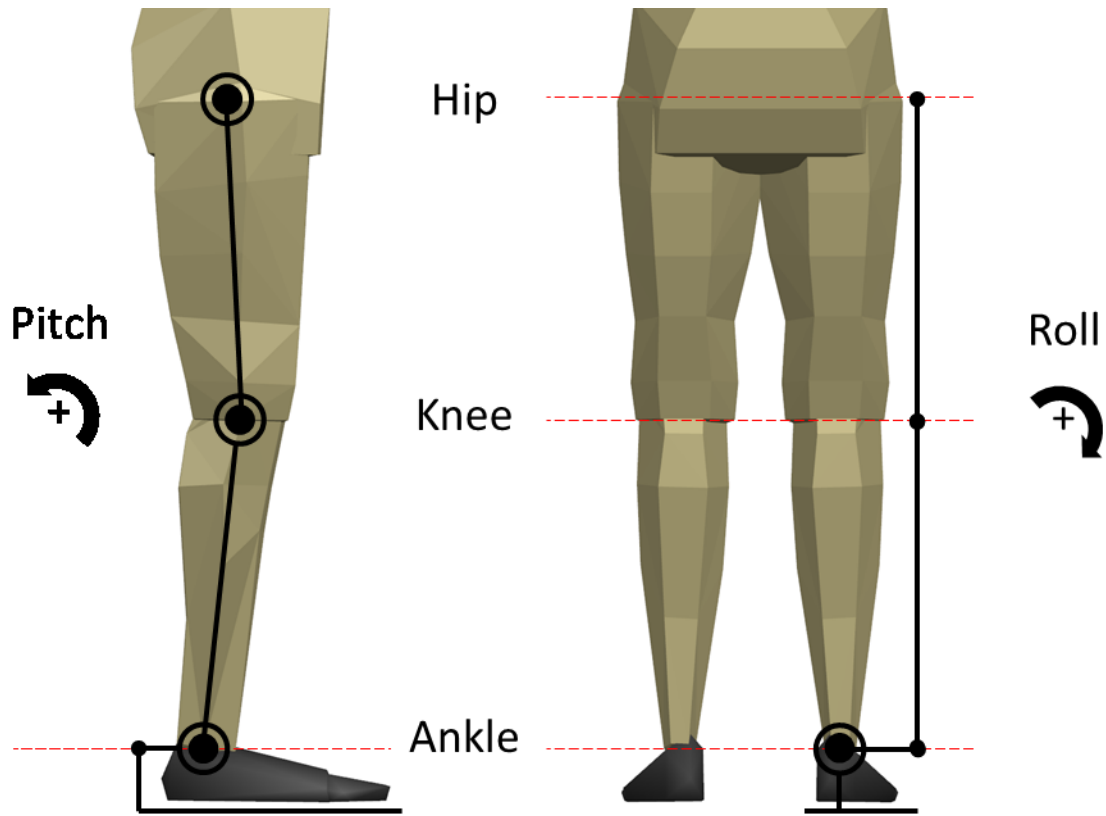


Figure 2: Exoskeleton leg joint layout, showing alignment with human leg joint axes. The dashed line indicates the human joint axis, and a ringed circle indicates a joint in-plane.

Also, splitting the joints' motion in this way means focused requirements can be developed for each degree of freedom. These range of motion requirements were determined by Li. A survey of different studies identified a target window for the range of motion of each degree of freedom. These results, summarized below in Table 1, come from studies examining the natural range of motion of a variety of individuals. A later analysis will augment these requirements with the range of motion needed for different gaits.

Table 1: Typical limits for joint range of motion [8].

Joint	Motion		Typical Limit [deg]
Hip	Flexion	(positive pitch)	0 - 30
	Extension	(negative pitch)	90 - 130
Knee	Extension	(positive pitch)	0 - 10
	Flexion	(negative pitch)	87 - 122
Ankle	Dorsiflexion	(positive pitch)	5 - 20
	Plantarflexion	(negative pitch)	10 - 45
	Inversion	(positive roll)	
	Eversion	(negative roll)	

Li's research also included a breakdown of typical lengths for the segments of the legs. Table 2 shows the fractional lengths for each segment, and what that corresponds to for different total heights. When configuring the exoskeleton for different heights, it is important that each segment has its own independent, fine adjustability. This will allow the exoskeleton to accommodate the user's actual geometry, and keep the exoskeleton's joint axes closely aligned with those of the patient.

Table 2: Fractional and typical segment lengths [11]. Percentiles are for a 50/50 mixed population of men and women. These values can serve as a guide for what total height can be accommodated by a certain segment length.

Segment	Fractional Height	5th [cm]	50th Percentile [cm]	95th [cm]
Total Height	1.00H	155.10	170.40	188.70
Upper Leg (Thigh)	.245H	38.00	41.75	46.23
Lower Leg (shin)	.246H	38.15	41.92	46.42
Foot Height	.039H	6.05	6.65	7.36
Foot Length	.152H	23.58	25.90	28.68
Total Leg	.530H	82.20	90.31	100.01

Speed and torque requirements were found by Kendrick and used to design the actuator packages that will be used throughout the exoskeleton. Kendrick used mass-dependent gait data to determine the peak speed and torque of each joint for a person walking forward on flat ground. This data came from a series of studies intended to analyze the motion of person and make inferences on how much power was required by the actions. These analyses will be discussed further in Section 3.4.1. A similar analysis was done using a simulation of ESCHER, (a bipedal robot from the

Virginia Tech TREC Lab), where ESCHER had been scaled up to have a mass of 140kg. Both a human and a robot were used for this because the two bipeds approach walking differently, requiring different speeds and torques to accomplish the same task. These findings are summarized below in Table 3. For the speed and torque of each degree of freedom, the larger of the two peaks was chosen as the requirement.

Table 3: Derived requirements for peak joint power, torque, and velocity [9].

Joint Axis	Power [W]	Torque [N-m]	Velocity [rad/s]
Hip Pitch	707.5	200.0	5.1
Knee Pitch	780.0	300.0	7.2
Ankle Pitch	515.9	197.2	5.1
Ankle Roll	116.3	89.5	5.9

Other important factors addressed the usability and practicality of the exoskeleton. For example, keeping the total mass of the exoskeleton low helps ease the torque requirements at each joint. That, combined with keeping the width of the system low gives the exoskeleton and its user more freedom in where they can go. Also, the structure of the legs will need to include mounting interfaces for the other subsystems that will be incorporated later, such as power, control, cooling, and a safety cover. Since these designs have not been finalized, the interfaces should be easily modifiable to accommodate these systems as they are refined. And as always, keeping the design simple was beneficial for the project, making analysis, manufacture, and assembly easier as well as reducing possible failure points. As part of this effort, components from the hip design, such as fasteners, bearings, sensors, and mounting structures were reused wherever feasible.

3. Joint Output Modeling

3.1. Basics

One of the first steps for designing the leg was determining the location of each actuator relative to the joint it acted on, since that would have a significant effect on the joint's capabilities. Each joint in the exoskeleton is being actuated using a linear actuator. A two-bar linkage is used to convert the linear motion and force of the actuator to rotational motion and torque at the joint, as shown in Figure 3. A two-bar linkage was chosen because of its relative compactness and simplicity, though it results in variable outputs over the joint's range of motion. This setup results in two important considerations. First, the distance from the joint pivot, O , to where the linkages connect, B , affects the relationship between input force and output torque, as well as input linear velocity to output radial velocity. Second, the angle between the linkages (the actuator, AB , and the lever arm, OB), α , determines how much of the input force and velocity are converted to torque and radial velocity. At 90° , the output peaks as all the inputs are converted. However, as that angle changes, some fraction of the inputs is lost and the outputs drop below that maximum value. This relationship can be summarized in the following equations:

$$T = rF\sin(\alpha) \quad (1)$$

$$\omega = \frac{v}{r}\sin(\alpha) \quad (2)$$

where T and ω are the output torque and velocity, respectively, F and v are the input force and velocity, r is the length of the lever arm OB , and α is the angle between the two linkages. As can be seen from these equations, torque is proportional to r , while radial velocity is inversely proportional to r . This means that, since the maximum power input is already defined by the available actuators, the ideal value of r needs to be found that balances the torque and velocity needed at each joint.

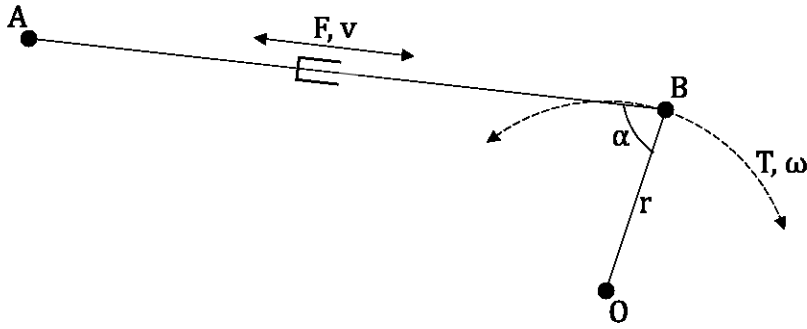


Figure 3: Simplified linkage diagram.

3.2. Derivation

When designing the joints for the exoskeleton, two main factors were available to configure this linkage. Figure 4a and b below show the basic layout and important distances and angles used in deriving these relationships, while Figure 5 shows these points applied to the final knee joint and actuator. First was the x and y position of the actuator pivot, **A**, relative to the joint pivot, **O**. The other factor was the length of the lever arm, r , and its position relative to the actuated segment, OC . The angle between the segment and the lever arm, ϕ , determines what joint angle has the most available force and velocity. Other factors were important to keep track of as well, such as the angle between the actuator and its spring (u) and the total length of the actuator package, l . In order to get the best results within the design constraints, it was necessary to identify and model how each of these factors affected the joint output.

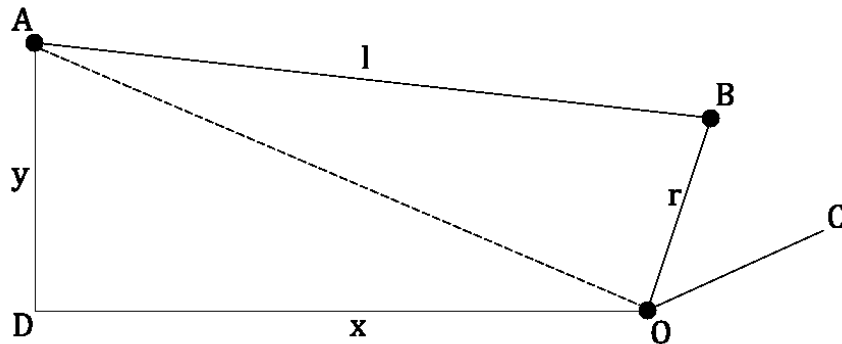


Figure 4a: Linkage lengths used in derivation.

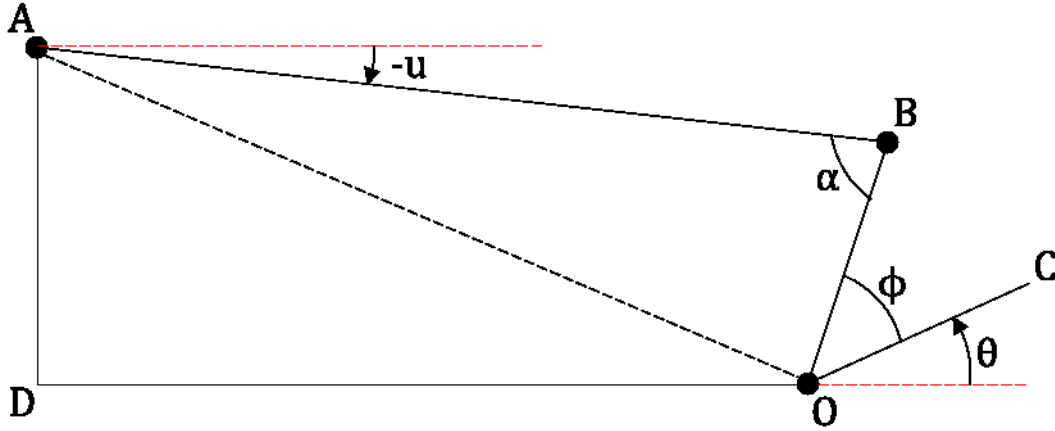


Figure 4b: Linkage angles used in derivation.

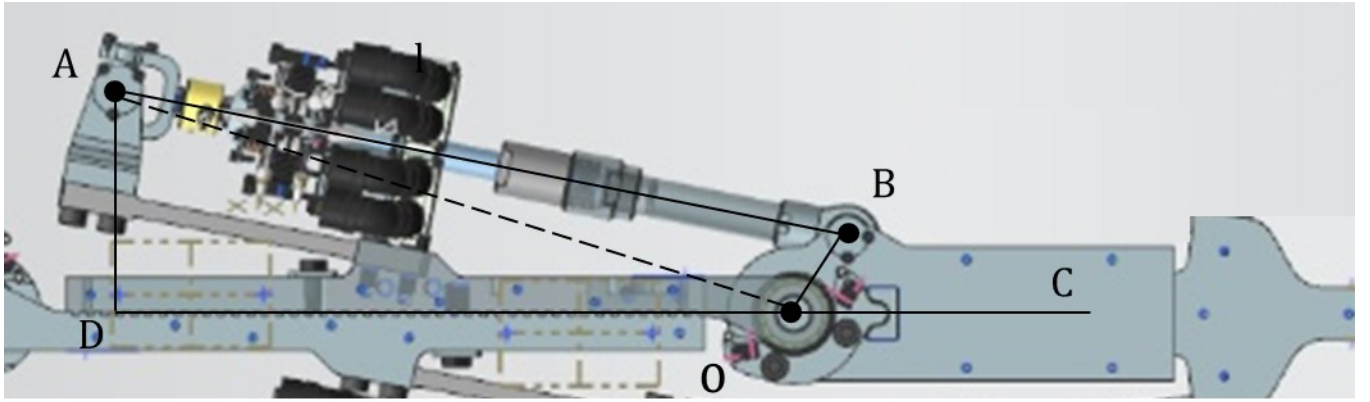


Figure 5: Reference points overlaid onto knee actuator assembly (some components hidden).

First, the ideal value of ϕ needs to be found, such that the linkage's peak output lines up with expected peak need, θ_c . Since peak output occurs when $\alpha = 90^\circ$, $\angle ADO$ and $\angle ABO$ form two right triangles. The length of the shared hypotenuse, AO , can be found using the Pythagorean theorem,

$$|AO| = \sqrt{x^2 + y^2}. \quad (3)$$

The interior angles, $\angle AOD$ and $\angle AOB$ can also be found using the inverse tangent and inverse cosine function, respectively.

$$\angle AOD = \tan^{-1}\left(\frac{y}{x}\right) \quad (4)$$

$$\angle AOB = \cos^{-1}\left(\frac{r}{|AO|}\right) = \cos^{-1}\left(\frac{r}{\sqrt{x^2 + y^2}}\right) \quad (5)$$

Finally, the offset can be found by subtracting these angles from 180° ,

$$\phi = 180^\circ - \angle AOD - \angle AOB - \theta_c = 180^\circ - \tan^{-1}\left(\frac{y}{x}\right) - \cos^{-1}\left(\frac{r}{x^2 + y^2}\right) - \theta_c \quad (6)$$

where θ_c is the desired joint angle for peak output. As can be seen here, the needed offset is a function of both the lever arm length and the location of the actuator pivot, so it needs to be recalculated whenever one of these values changes.

With ϕ known, the equation for α as a function of joint angle can be found for each configuration. Relative to the joint pivot, \mathbf{O} , the position of \mathbf{A} is constant, while \mathbf{B} can be described as a function of the joint angle, θ .

$$A = (-x, y) \quad (7)$$

$$B = [r\cos(\phi + \theta), r\sin(\phi + \theta)] \quad (8)$$

The length of the actuator assembly is the distance between the actuator assembly

$$l = A - B = [-x - r\cos(\phi + \theta), y - r\sin(\phi + \theta)] \quad (9)$$

$$-u = \text{atan}(l_y, l_x) \quad (10)$$

$$\alpha = \theta + \phi - u \quad (11)$$

These equations can be combined to find alpha as a function of θ , ϕ , x , y , and r .

$$\alpha = \theta + \phi + \text{atan}\left(\frac{y - r\sin(\phi + \theta)}{x + r\cos(\phi + \theta)}\right) \quad (12)$$

Finally, with this equation for α , the joint outputs T and ω can be calculated as a function of the joint position, θ , and the design inputs ϕ , x , y , and r using Equations 1 and 2.

3.3. Linkage Geometry Refinement

To narrow down the possible configurations for each joint, a MATLAB script (see Appendix A: LeverArmCalc) was created to rapidly see the possible outputs for different input parameters. For each joint, a starting value for x , y , and θ_c were chosen. Using these values, the script would then plot T , ω , l , and u as a function of r and θ for a given actuator package. The plotted range of θ

represents the desired joint range of motion determined by Li. The peak speed and torque requirements for the joint being examined were overlaid on these plots so that it could easily be determined which values of r satisfied both requirements. These plots for the are shown in Figure 6 through Figure 9.

Hip Pitch, MkIII High Speed

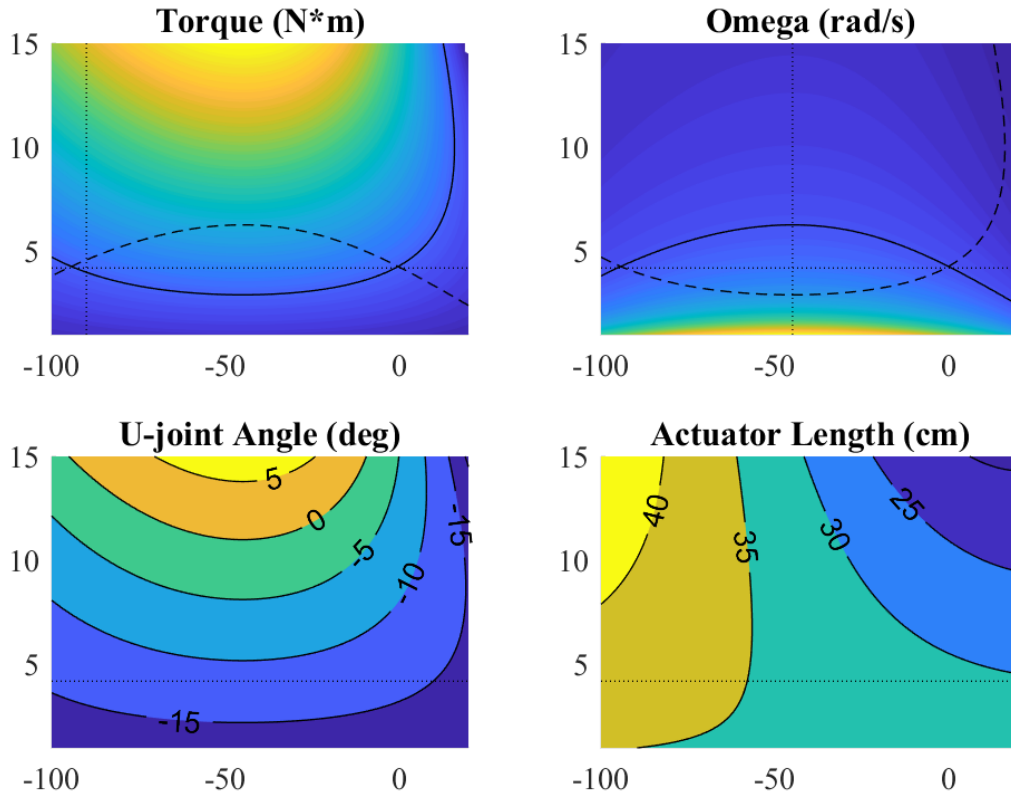


Figure 6: Joint output colormaps as a function of θ (x-axis) and r (y-axis). In the top plots, solid lines define the region inside which the output exceeds the peak requirement established by Kendrick. The dashed line shows the shape of that boundary for the other output, indicating a region in which both requirements are satisfied. The horizontal dotted line shows the selected value of r used in the next set of plots, and the vertical dotted line was used as a reference to where peak requirements were expected.

Knee Pitch, MkIII High Speed

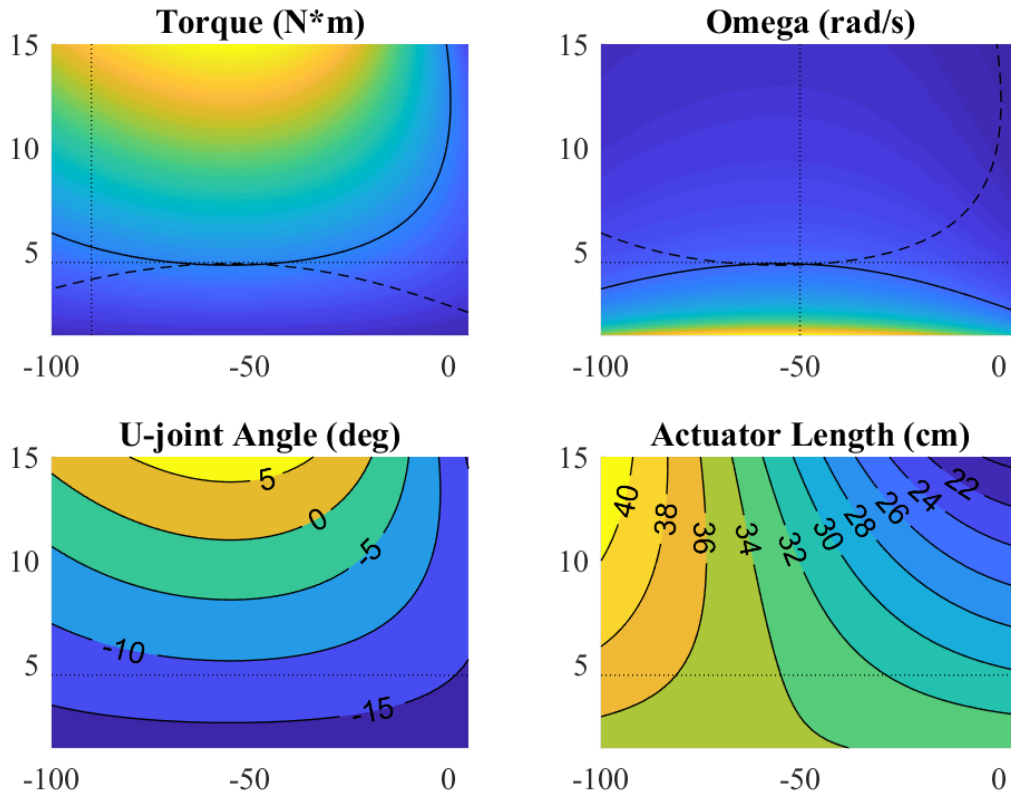


Figure 7: Color map of outputs for the knee joint. Note how little overlap exists between the regions of sufficient torque and speed, as well as the significant negative bias of the u-joint angle.

Ankle Pitch, x2 MkIII Lite High Speed

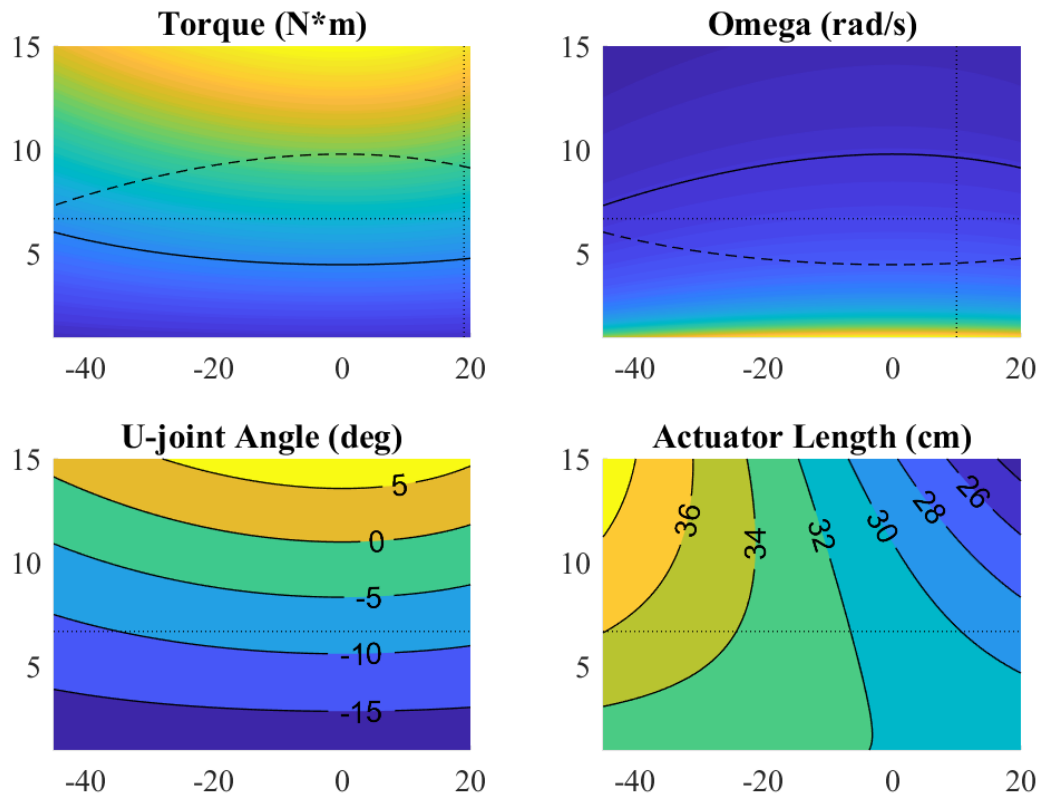


Figure 8: Joint output color map for ankle pitch, using the combined output of two MkIII Lite actuators.

Ankle Roll, x2 MkIII Lite High Speed

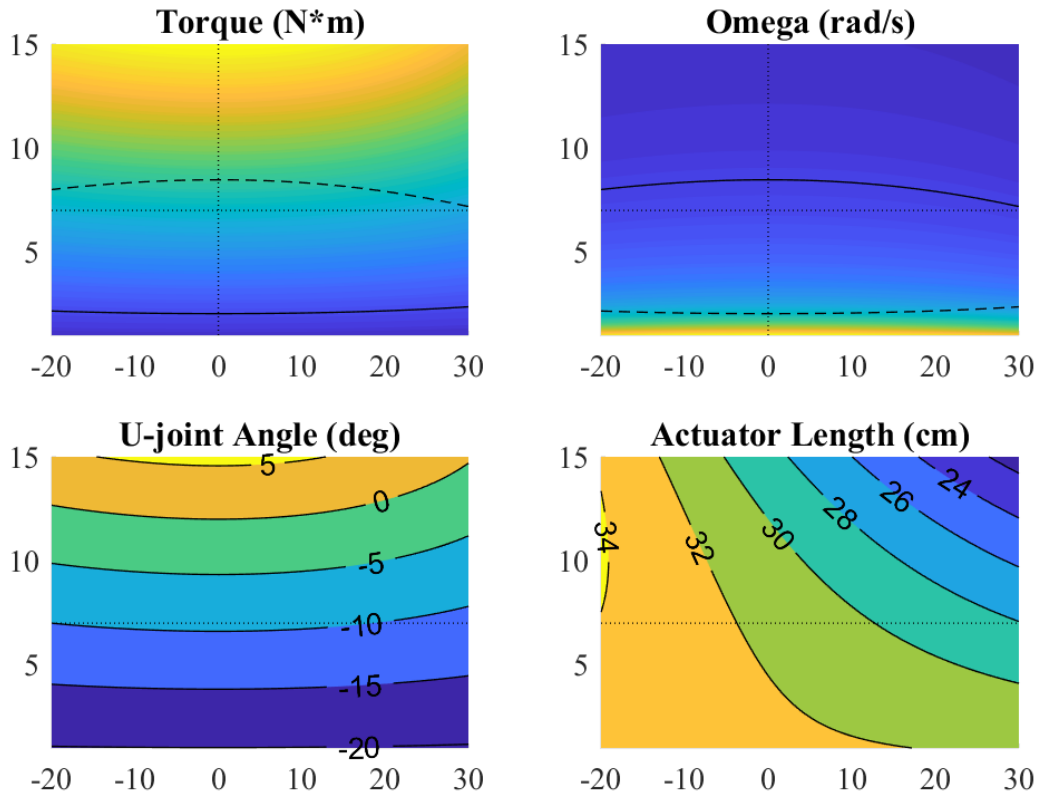


Figure 9: Joint output colormap for ankle roll.

LeverArmCalc also checked if the linkage approached a singularity. Allowing α to reach 180° would cause several problems. Continuing to retract the actuator would have no effect on the joint but would put significant tension on the components in the actuator package. On the other hand, extending the actuator would have no control over which direction the joint moved, making it difficult to consistently switch from region to the other. Also, as the linkage approaches this singularity, the conversion factor drops significantly, resulting in very little of the actuator's output reaching the joint. To indicate this, regions where α reached or exceeded 180° were whited out in the color map. This effect was not needed in any of the plots shown above.

Finally, the script was also capable of examining the output for a specified lever arm length, shown in Figure 10 through Figure 13. The same four variables were plotted as a function of the joint angle, and several numerical values were determined. These included:

- the minimum and maximum angles of the u joint

- the minimum and maximum lengths of the actuator assembly, and the thread length necessary for that amount of motion, and
- the offset needed for the desired angle of maximum output.

Hip Pitch, MkIII High Speed, $r = 4.2$ cm

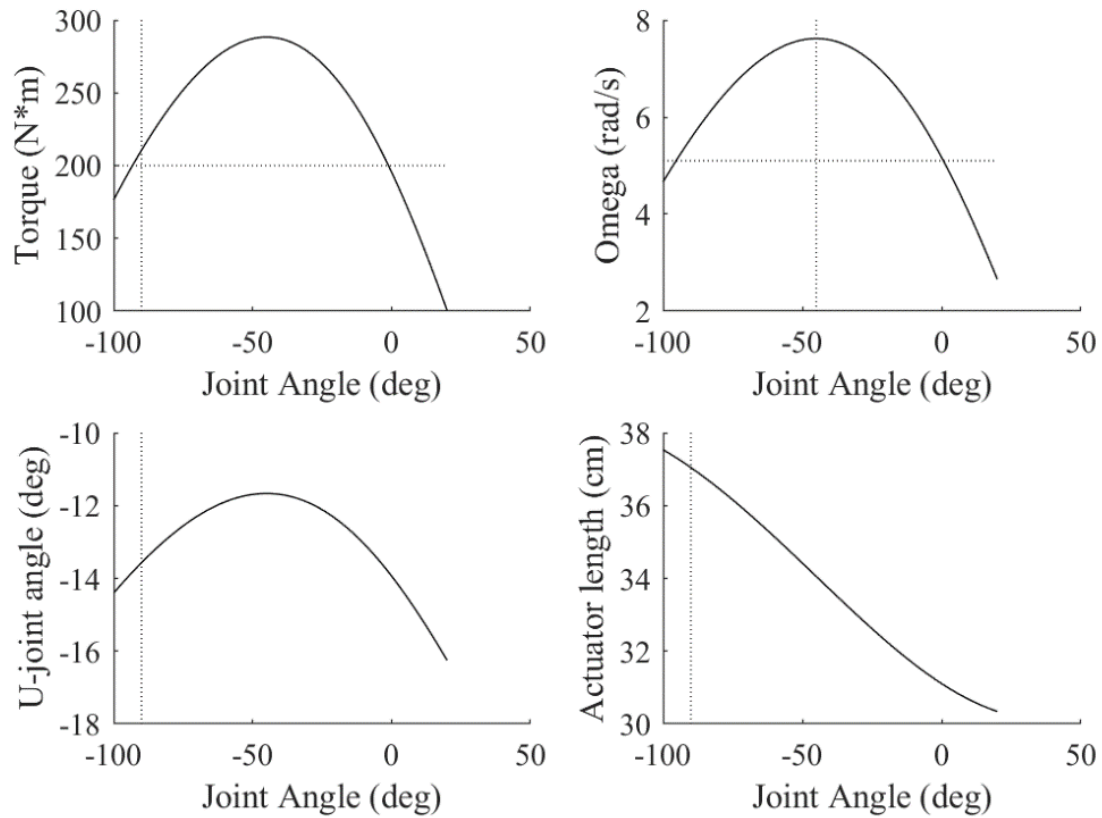


Figure 10: Joint output as a function of theta for a specific lever arm length. As before, the vertical dotted line indicates the expected angle of peak need. The horizontal dotted line indicates the required maximum speed or torque as defined by Kendrick.

Knee Pitch, MkIII High Speed, $r = 4.5$ cm

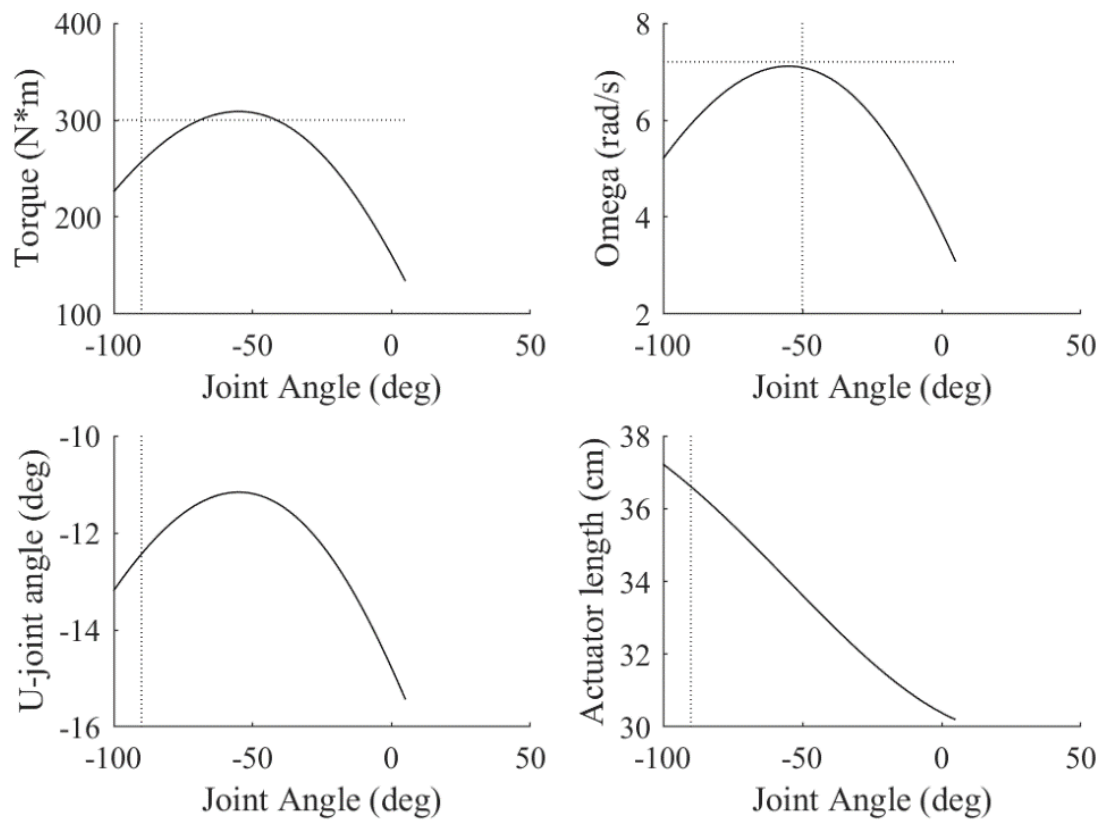


Figure 11: Knee pitch outputs for the selected lever arm length. The 90° angle of peak torque represents standing from a seated position, not from the typical walking cycle.

Ankle Pitch, x2 MkIII Lite High Speed, $r = 6.7$ cm

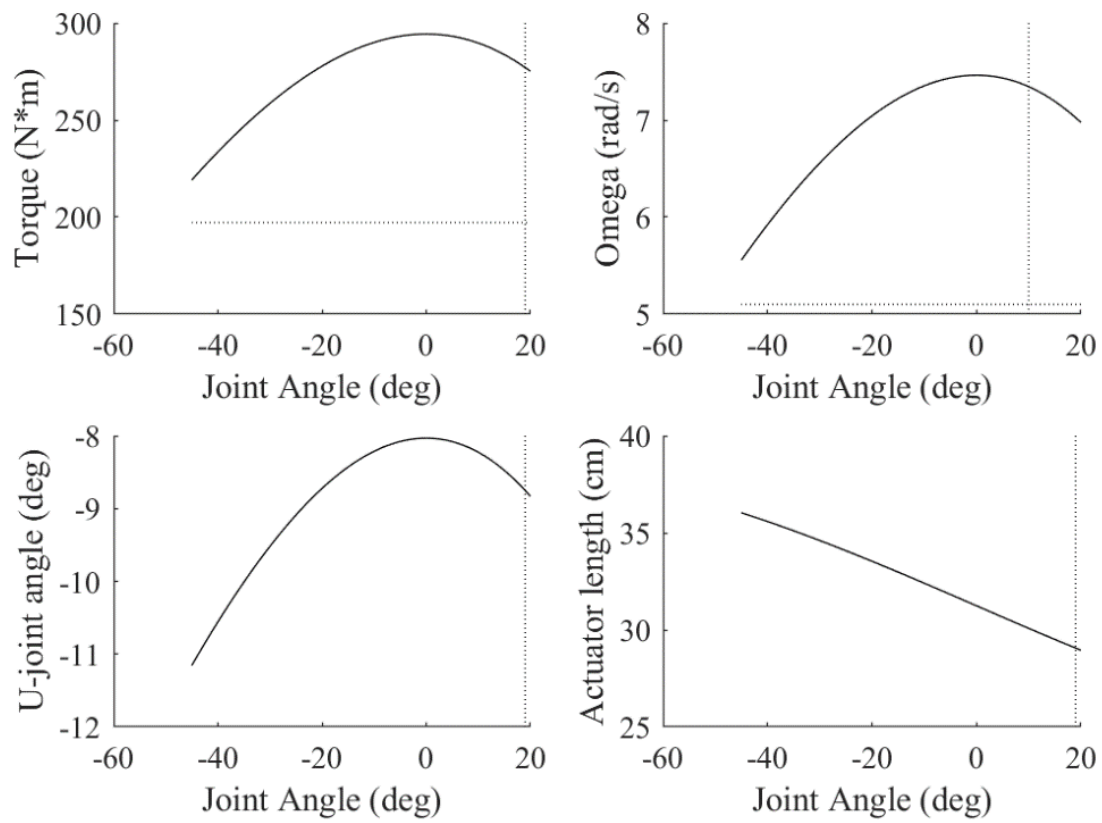


Figure 12: Joint output for the selected lever arm length.

Ankle Roll, x2 MkIII Lite High Speed, $r = 7$ cm

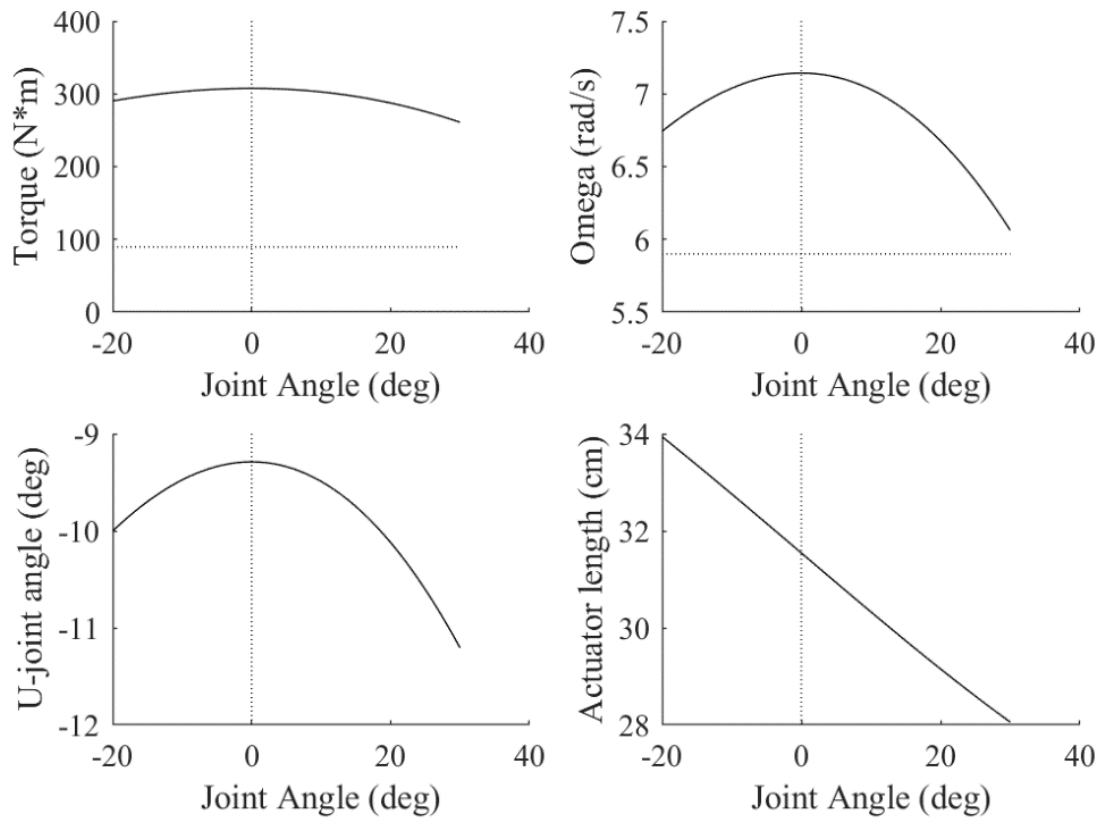


Figure 13: Joint output for the selected lever arm length.

3.3.1. Results

LeverArmCalc was used repeatedly, modifying x , y , and θ_c , to determine the key impacts of each variable. These results can be summarized as:

- Increasing x flattens all curves, but especially the u curve. A larger x means that the same vertical change in the position of B results in a smaller angular change at A and α . Decreasing u results in a more consistent spring response [9].
- Changing y will have a similar or inverse effect on the u curve, depending on whether y is greater than or less than r . Moving y away from r will increase the magnitude of u .
- Increasing r will shift the torque response up and the velocity response down. Also, increasing r will result in a larger range in u and l for the same range of θ .
- Changing θ_c , and thus ϕ will shift both the torque and velocity responses laterally, controlling what joint angle results in maximum output.

The results of this analysis were very useful for informing the initial design of the legs. Without risking buckling of the actuator assembly, x was set to slightly less than the minimum desired joint-to-joint length of each segment. This would maximize x while avoiding interference between segments. To keep the design compact, y was minimized keeping in mind that y would need to include enough space for the body of the segment and the spring mounting hardware (60 mm from the trunnion pivot to the bottom of the spring), as well as room for the spring to deflect downwards. At that time, FEA results of the springs were not yet available, so y was tentatively set to 110 mm, leaving 30 mm for spring deflection and 20 mm for material (giving the segment a total thickness of 40mm, since y extends from the actuator pivot to the segment's center line). Figure 14 outlines this allocation. Later FEA results from Kendrick would validate this assumption as only 14.2 mm was needed by the MkIII spring, and 9.7 mm for the MkIV. Table 4 shows the values chosen at the end of this analysis.

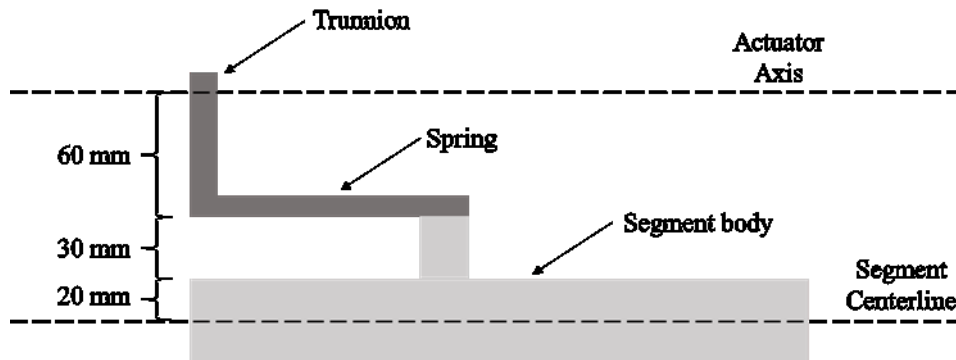


Figure 14: Space allocation within y dimension.

Table 4: Joint geometries selected by initial analysis

	Hip Pitch	Knee Pitch	Ankle Pitch	Ankle Roll
x [mm]	325	325	300	300
y [mm]	110	110	110	120
r [mm]	42	45	67	70
ϕ [deg]	46.9	56.9	88.1	*

* Ankle roll offset will be defined by ankle pitch geometry.

This analysis also showed some possible issues that would need to be addressed. First, even with the high-speed actuator configuration, for all joints the desired lever arm length was significantly

less than y . This meant that u was heavily skewed negative and outside of the allowable range [9]. Adding additional linkages to the system would help alleviate this issue, making it possible to separate the key variables and bring u back within its limits. However, this would bring its own problems, adding size and complexity, creating pinch hazards, and reducing the space available to fit the ball-screw, possibly causing issues with range of motion. Instead, it was determined that simply rotating the spring assembly would also solve this issue, while also having other benefits. Rotating the spring would have no effect on the actuator or its output but would bring the force on the spring's input angle back within acceptable limits. Tilting the actuator would also give more space for deflection and would make it easier to mount the base of the spring closer to the main body of each segment.

A more pressing issue, however, was apparent in the knee and hip joints. No combination of actuator, placement, and lever arm was able to satisfy the torque and speed requirements across the whole range of motion. While the hip actuator was mostly successful in meeting these requirements, for the knee actuator the curves barely intersected. However, recall that these requirements came from the maximum output seen in either Escher or a person across the whole gait cycle. In reality, the needed torque and speed changes constantly throughout the gait cycle, and does so in a repetitive, consistent way. To determine if it was possible to take advantage of this fact, a more detailed analysis would be needed to compare the actuator's output against these changing requirements.

3.4. Gait Cycle Comparison

3.4.1. Gait Cycle

Analysis of walking motion is typically done in terms of the gait cycle. The gait cycle is the sequence of motions repeated with each step, so that everything ends where it started at the beginning of the cycle. Data such as joint velocity or torque are plotted against cycle percentage, the elapsed time normalized by the total cycle time. A typical gait cycle for a 140 kg human walking at approximately 1.3 m/s is shown below in Figure 15 through Figure 17, digitized by Kendrick from Perry's *Gait Analysis: Normal and Pathological Function* [12]. Examining gaits in this way makes it much easier to compare the motions of different gaits. The repetitive nature of the gait cycle also makes it useful for determining requirements. If the exoskeleton can meet the

requirements throughout one gait cycle, then it will be able to maintain that gait for as long as desired.

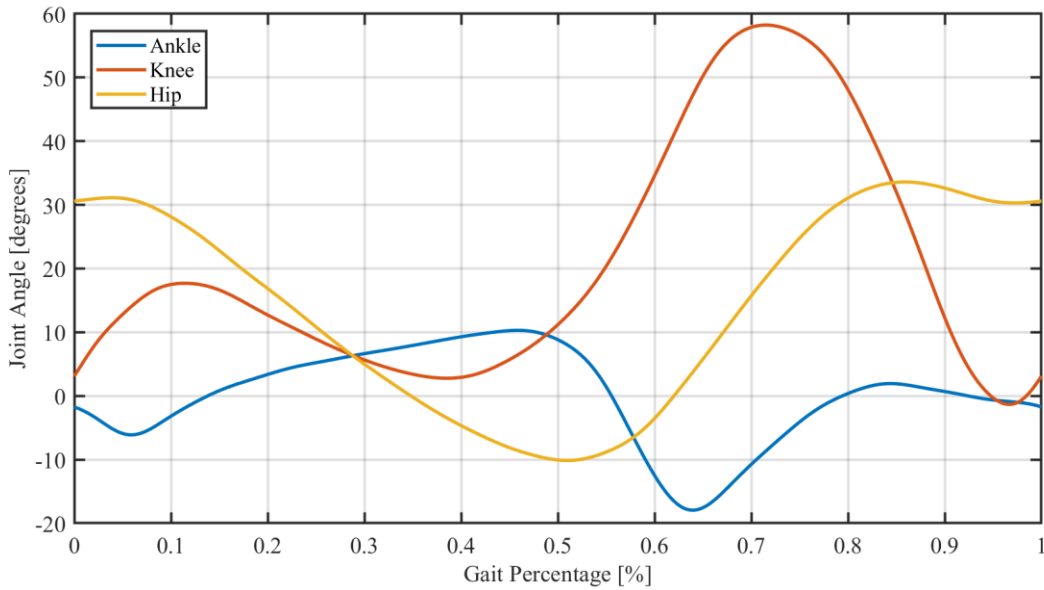


Figure 15: Sample gait cycle, showing joint angle as a function of gait percentage [9].

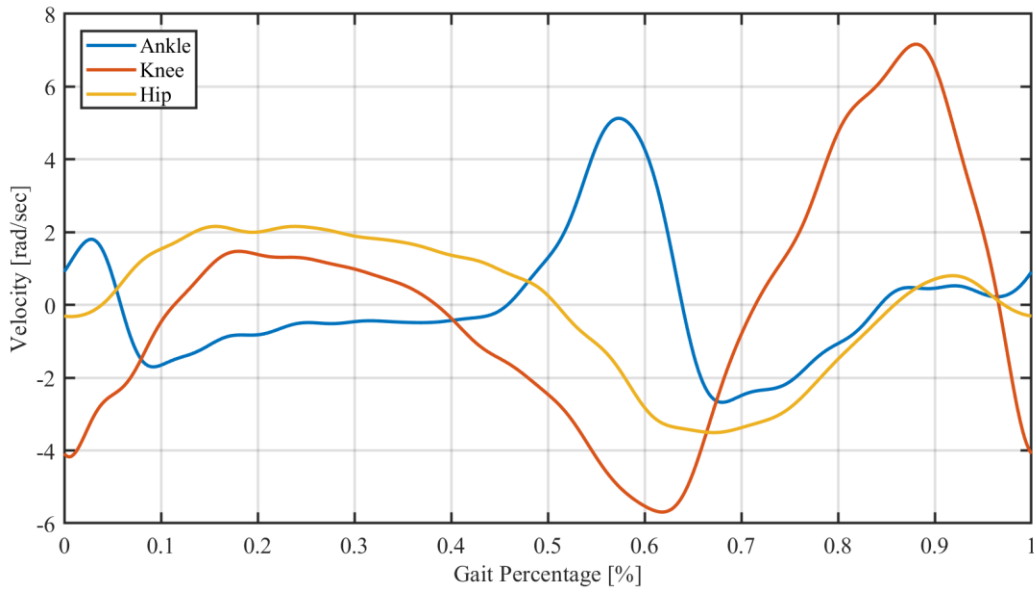


Figure 16: Sample gait cycle showing joint velocity as a function of gait percentage [9].

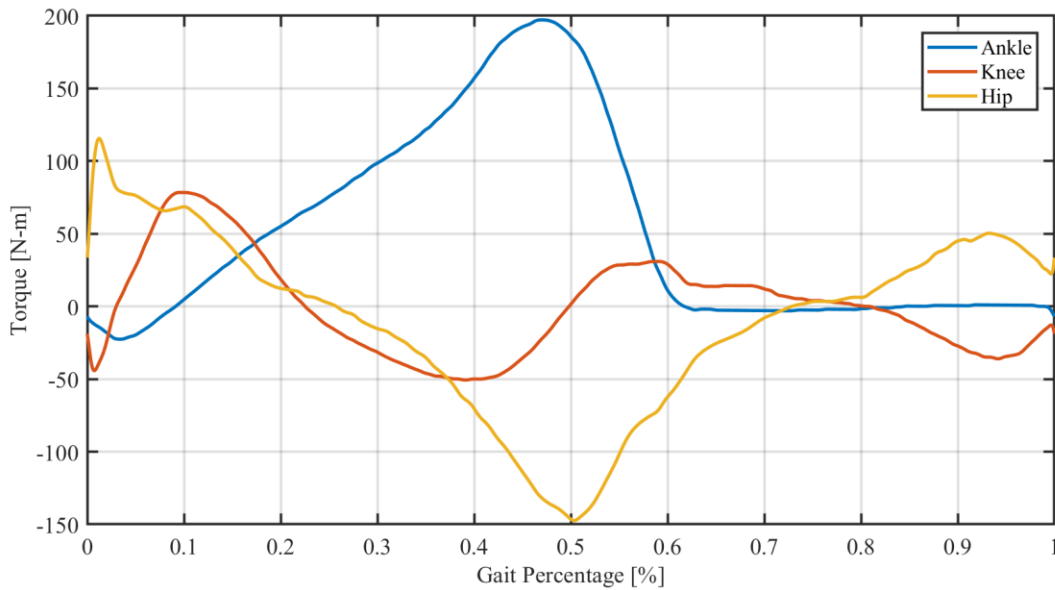


Figure 17: Sample gait cycle showing joint torque as a function of gait percentage. Torque has been scaled to that of a 140kg person, the maximum combined mass of the exoskeleton and user [9].

In this analysis, several different gait cycles were examined. The normal walking motion of a person [12] and two bipedal robots (Virginia Tech’s ESCHER [9] and Boston Dynamics’ ATLAS [13] using walking strategies developed at Virginia Tech’s TREC Lab). These gaits were compared to see how different walking strategies would affect the requirements of each joint. When walking, the robots keep their knees bent much more than a person would and use the knee to provide most of the power for forward walking. A person, on the other hand, almost completely straightens their knee while the leg swings forwards, and a lot of the power for forward motion actually comes from tendons in the ankle acting as springs. Two other gaits were examined, ESCHER’s stair climbing [9] and a human’s Sit-to-Stand (STS) [14], to see how the requirements of these common actions might differ from normal walking. In addition to looking at torque and speed requirements, examining the different gaits in this way made it possible to refine each joint’s range of motion requirements.

The robotic gait requirements came from simulations performed by the TREC Lab. For its simulation, the mass of ESCHER was scaled up to 140kg, the expected combined mass of the exoskeleton and user. This was done so that the simulated torques would closely match those needed by the exoskeleton if it employed the same walking strategy as ESCHER. The ATLAS data, however, was not scaled to 140 kg, so the magnitude of its torque requirements cannot be

used. The ATLAS data was included because it employs an improved, more straight-legged walking strategy, so it was used as a reference to see how the range of motion changed and which angles needed peak input. The human data was provided in a mass-dependent format, so the torque requirements were also scaled to that of a 140 kg person.

The bent-legged walking of the robots has two effects on the curves for each joint. First, the knee bending shifted all of the curves laterally, so that peak input would happen at a larger joint angle. Also, the robots required significantly more torque throughout the gait cycle, since the larger bends in each joint needed more torque to support the weight of the robot. Because of this, this analysis focused on matching each actuator's output to the human requirement curves.

When examining gaits in this way, it is also important to remember what information is hidden. For example, while the three walking cycles (human, ESCHER, and ATLAS) have similar joint speeds, their actual forward speeds are different. The human gait data is for a 1.70 m tall person walking forward at 1.3 m/s [12]. Atlas is moving slower, at only 0.78 m/s. This is partly because ATLAS is only 5' (1.5 m) tall, so each of its steps covers less ground. Though ESCHER is on the tall end at 5'10" (1.78 m) [14], its motion is less fluid and it pauses between each step, reducing the overall forward speed of the robot.

3.4.2. Script Behavior

Due to the other design factors, such as part clearances, the designed actuator and lever arm positions could not perfectly match the predicted ideals from the first script. Because of that, this second script (Appendix B: GaitCycleCheck) would perform its analysis using the actual joint geometry of the design, so it took x , y , r , and ϕ as inputs. These four values were then used with the relations found above to find the resulting outputs (T , ω , l , and u) as a function of joint angle for that specific configuration. Similarly, the torque and speed of the different gait cycles were also plotted as a function of joint angle (since the actuator's changing output will be the same in forward or reverse, the magnitude of torque and velocity were plotted). The joint output curves were then overlaid on these plots to easily determine if the decreasing outputs were enough to meet the requirements of each gait cycle.

GaitCycleCheck also plotted l and u as the joint angle changed, taking into account the tilted spring mount's reduction to the effective u-joint angle. It also determined the maximum allowable

length of the ball screw based on the minimum length of l and comparing the available travel on that ball screw to the amount of travel needed to reach the maximum length of l .

This information was used to evaluate how well the designed configuration met each joint's requirements and, where feasible, those designs were modified in an iterative process to get the best coverage possible. During this process, emphasis was placed on meeting the requirements of the human gait cycles. Since the exoskeleton will be worn and guided by a human operator, its gait cycle will more closely match that of a typical human's. Also, as discussed previously, the changes in the robotic gait cycles make them much demanding than normal straight-legged walking.

3.4.3. Results

When compared against the actual changing requirements, the actuators were largely successful in meeting the needs of each gait cycle throughout the range of motion. Figure 18 through Figure 23 show the output plots for the final configurations of each actuator configuration.

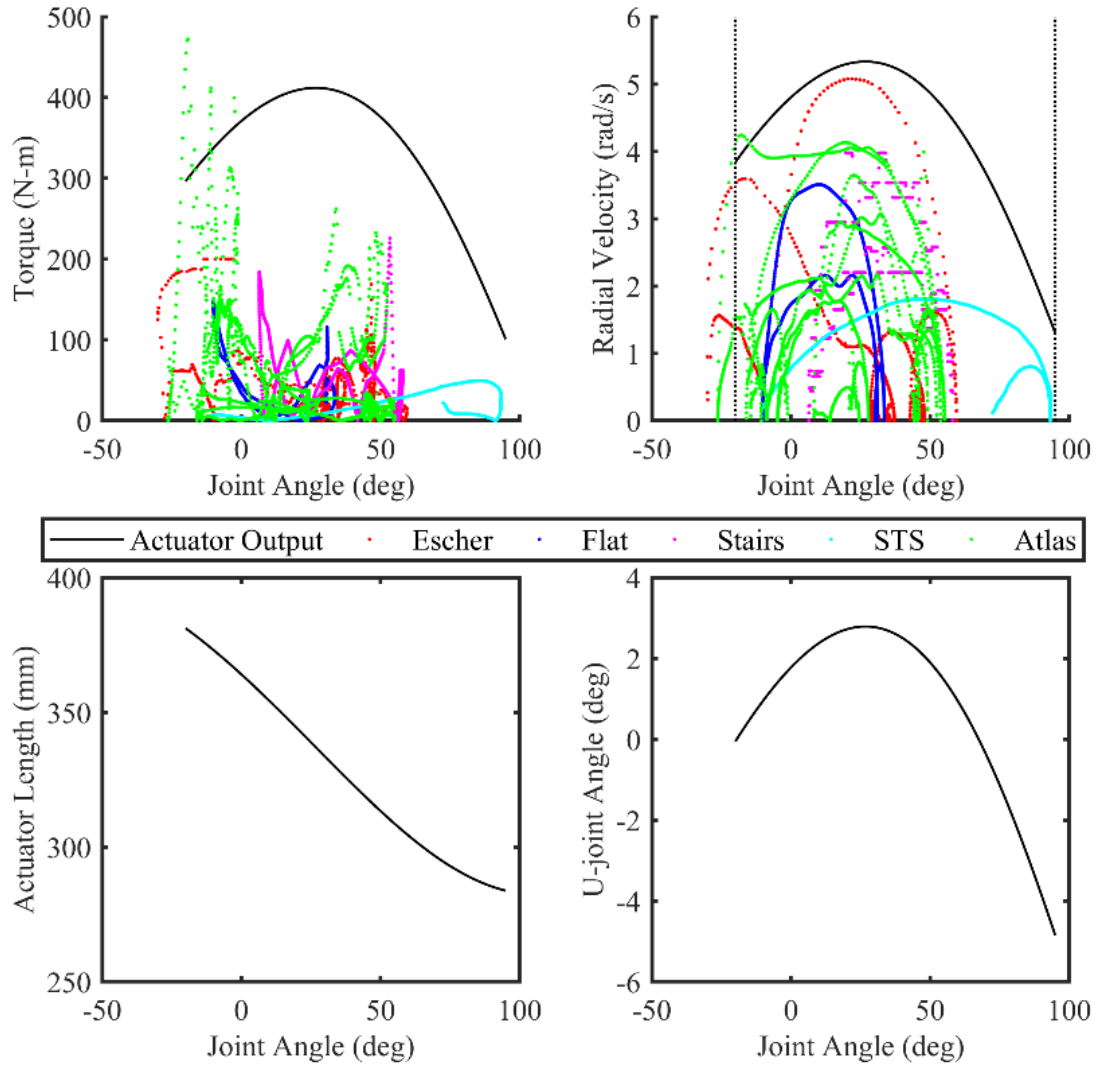


Figure 18: Gait cycle comparison for hip pitch. The vertical dotted lines indicate the joint angle limits.

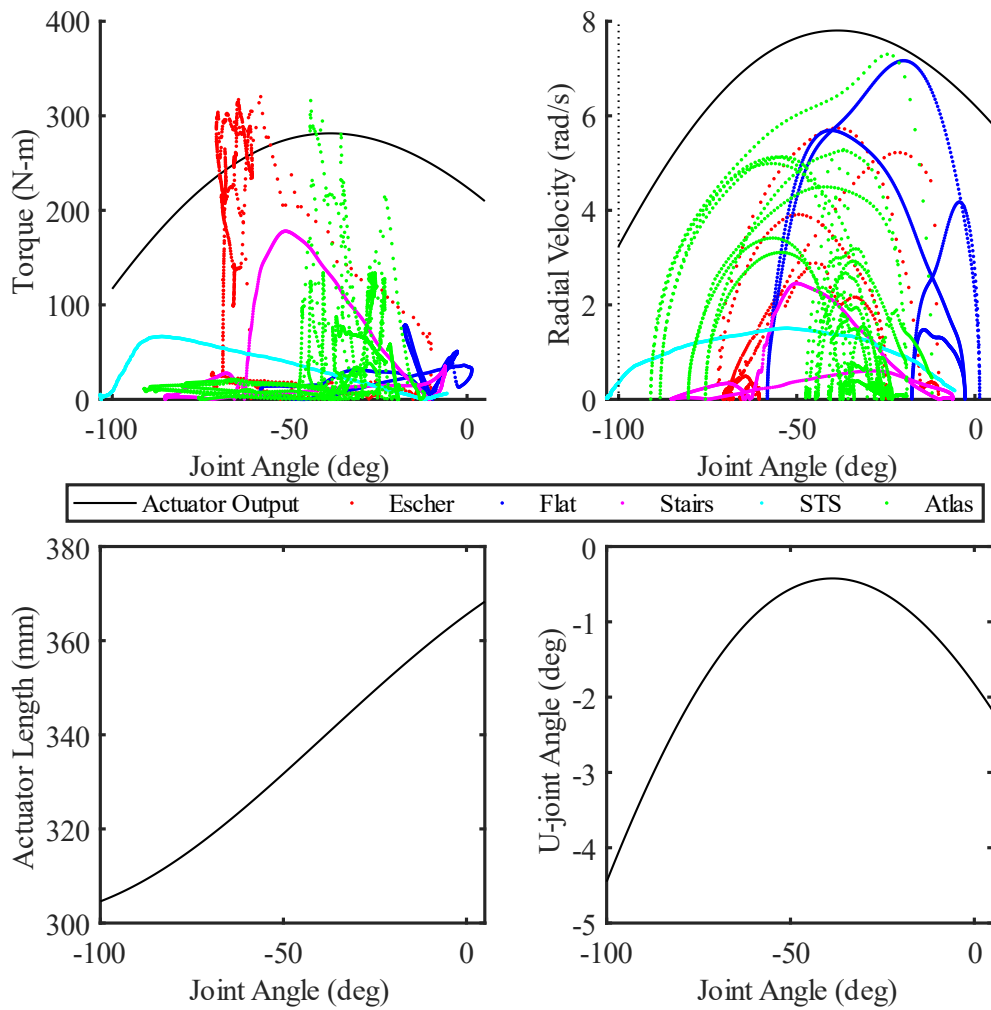


Figure 19: Gait cycle comparison for knee pitch.

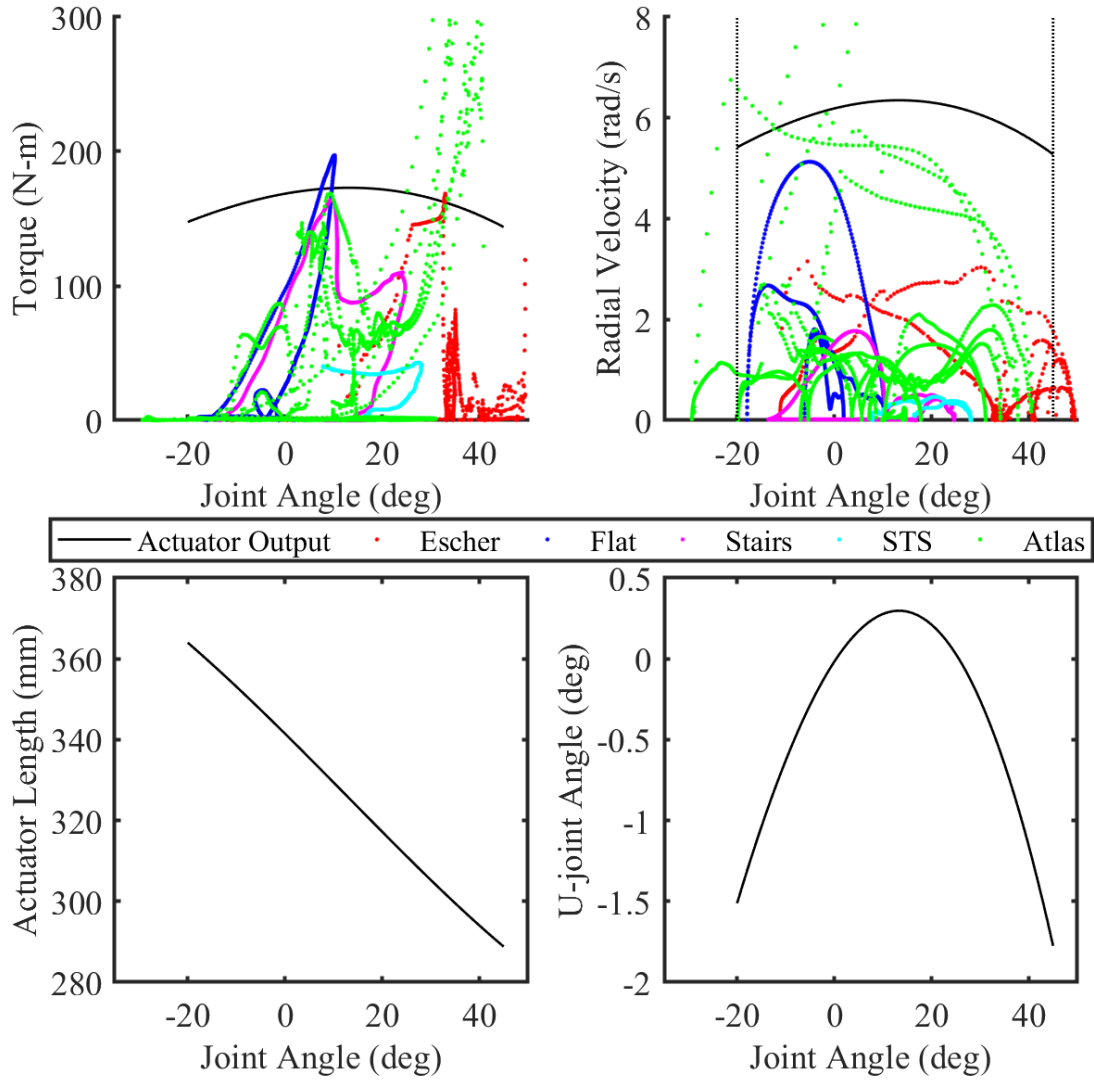


Figure 20: Gait cycle comparison for the front ankle actuator on the ankle pitch axis.

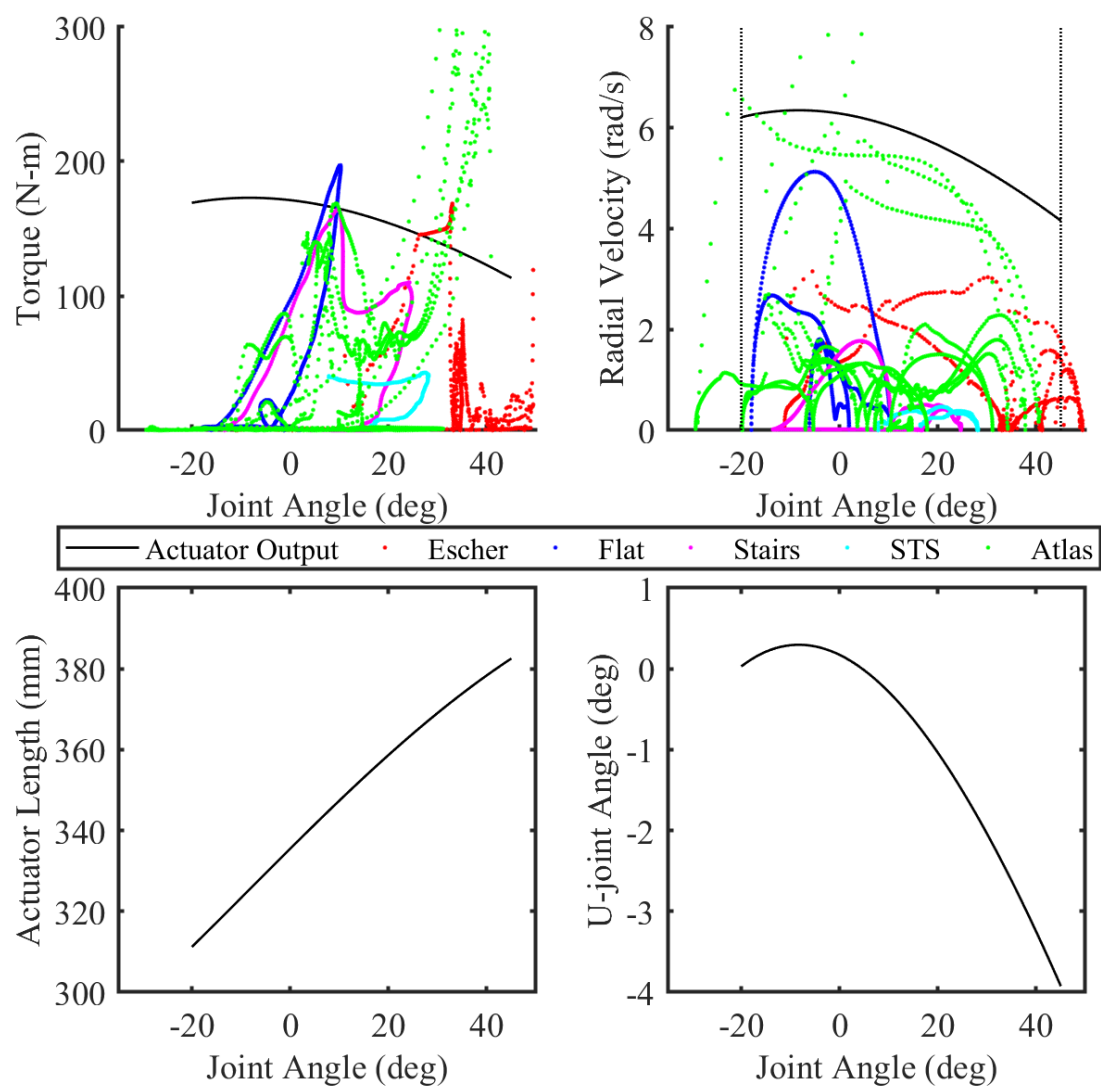


Figure 21: Gait cycle comparison for the rear ankle actuator on the ankle pitch axis.

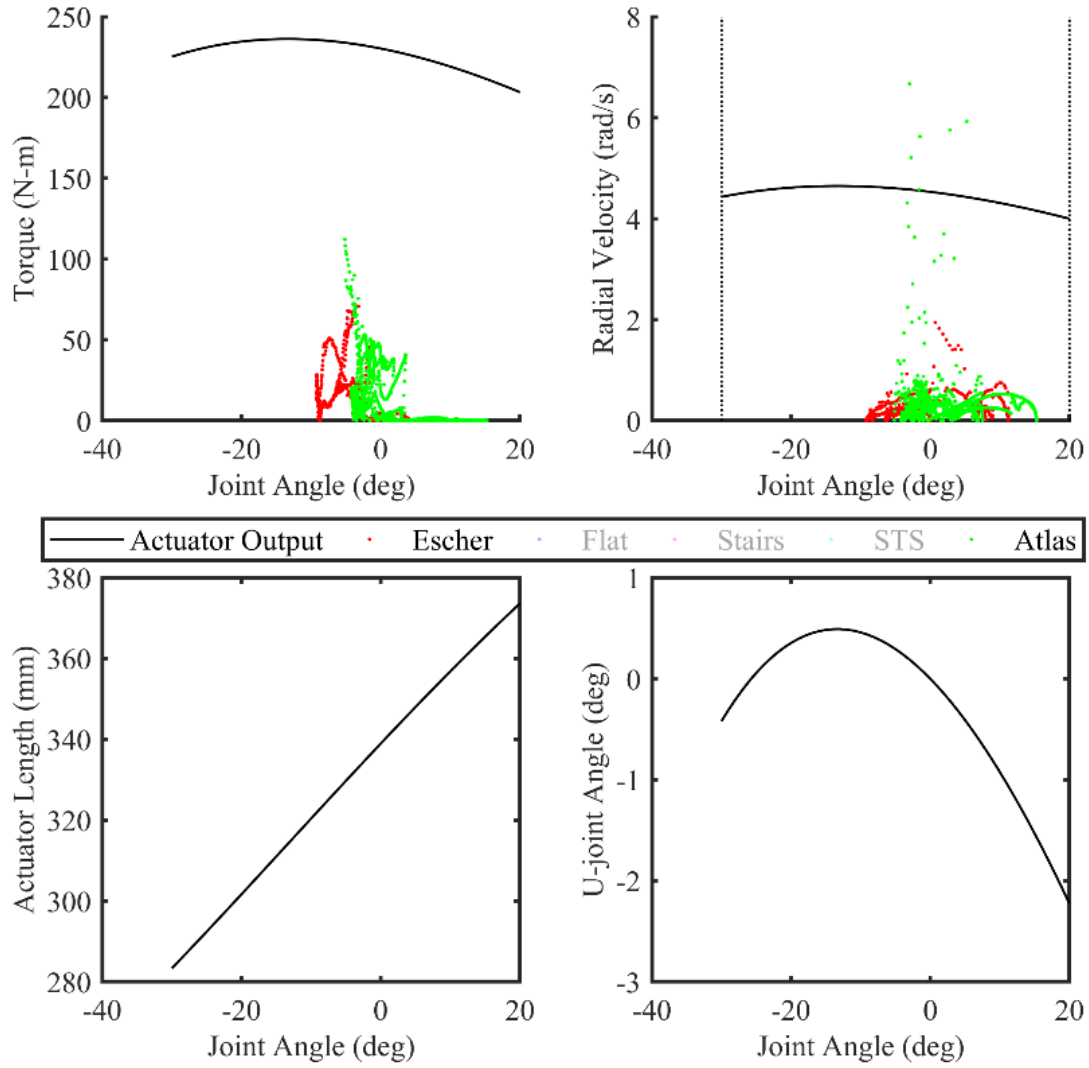


Figure 22: Gait cycle comparison for the front ankle actuator on the ankle roll axis.

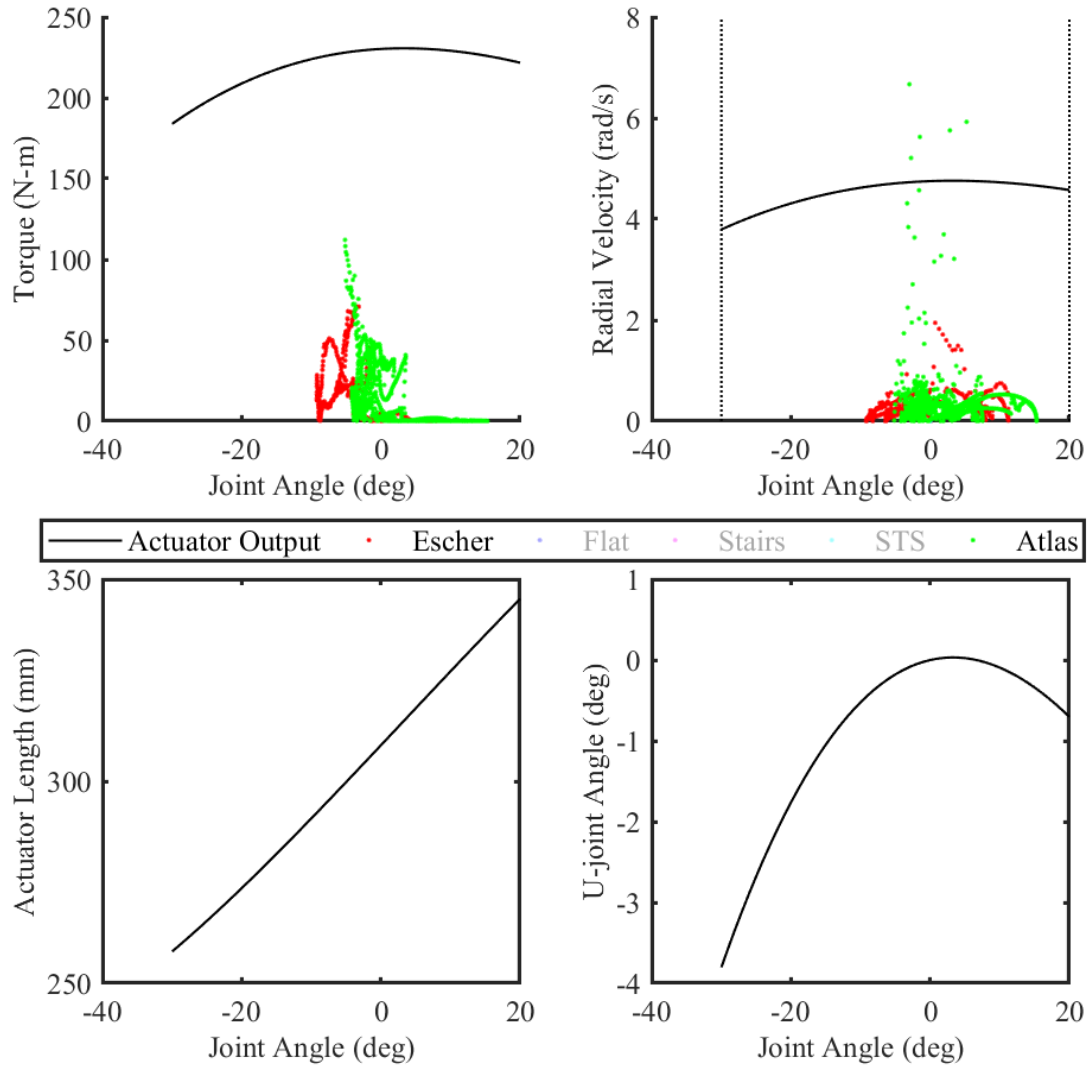


Figure 23: Gait cycle comparison for the rear ankle actuator on the ankle roll axis.

The hip actuator provides more than enough torque throughout its range of motion (recall that the ATLAS torque curve is not scaled correctly). Joint velocity has less margin, but is still sufficient, especially when compared against the needed velocity for the human cycles. This is not surprising, since the first analysis showed that the actuator could meet the original peak requirements throughout most of its range of motion. Having excess available speed and torque is desirable, since it will give the controller greater authority to keep the system balance and recover from a stumble.

The knee, as suggested in the first analysis, struggles with some of the cycles. The robotic gait torque requirements are outside the output curve. While not ideal, this was considered acceptable because most of that gap is caused by peak need being away from peak input, and the exoskeleton will not walk with the extreme knee bend that the robots used. A more pressing issue was the available speed. Here, the output barely meets the requirements of the flat walking cycles. Part of the issue with this joint is a simple lack of space. The limits of the ball screw mean that the heavy actuator cannot produce a faster linear velocity. However, at 6 cm, the actuator pivot is as close to the joint axis as it can be with the current design. Moving it closer while maintaining its current angular offset (ϕ) causes interference issues at extreme joint angles, while changing ϕ moves peak output away from peak need, counteracting any benefit from shortening r at the angle where its most needed. Several modifications were considered, however they all had significant drawbacks, so it was decided that accepting a small decrease in the overall forward velocity was the most practical solution.

In the final design, the ankle actuators ended up being asymmetrical, so they are shown separately. Generally, for each joint angle in ankle pitch and roll, the maximum speed will be the slower of the two actuators, while the maximum torque will be the sum of the two actuators. However, since these two actuators will be working together to control both pitch and roll, at any given moment the torque will likely not be evenly split between the actuators. This is not expected to cause problems, since one actuator on its own is almost enough to meet peak torque requirements for pitch, and more than sufficient for roll. The speed plot also has some margin, but not as much. Ideally, the lever arms would be shortened to balance torque and speed more, but doing so risked part collisions at some combinations of pitch and roll.

GaitCycleCheck was also used to determine the impact of spring deflection changing the location of the actuator pivot. Finite element analysis of the two springs showed a maximum deflection of 9.7 mm for the MkIII Lite springs, and 14.9 mm for the MkIII Heavy springs [9]. Adjusting the y values of each actuator by these amounts showed negligible changes to the output of each configuration and did not risk causing a singularity in the linkages.

Table 5: Final joint geometry values from the second analysis.

	Hip Pitch	Knee Pitch	Ankle Pitch	Ankle Roll
x [mm]	325	325	325	325
y [mm]	107.5	107.5	107.5	70
r [mm]	45	60	65	70
ϕ [deg]	55.8	43.2	72.6	90.0
tilt [deg]	10.5	10.5	7.25	0

4. Design

With the target actuator positions and linkage geometries identified, the next step was to design the structure connecting the actuators, joints, and the wearer. To maximize the user's comfort and best emulate a natural walking gait, the exoskeleton joints would need to be placed as close as possible to the person's joints. This would require the ability to finely adjust each segments length, ideally without changing the linkage geometries and thus the joint outputs. Figure 24 highlights the subsections of the leg.

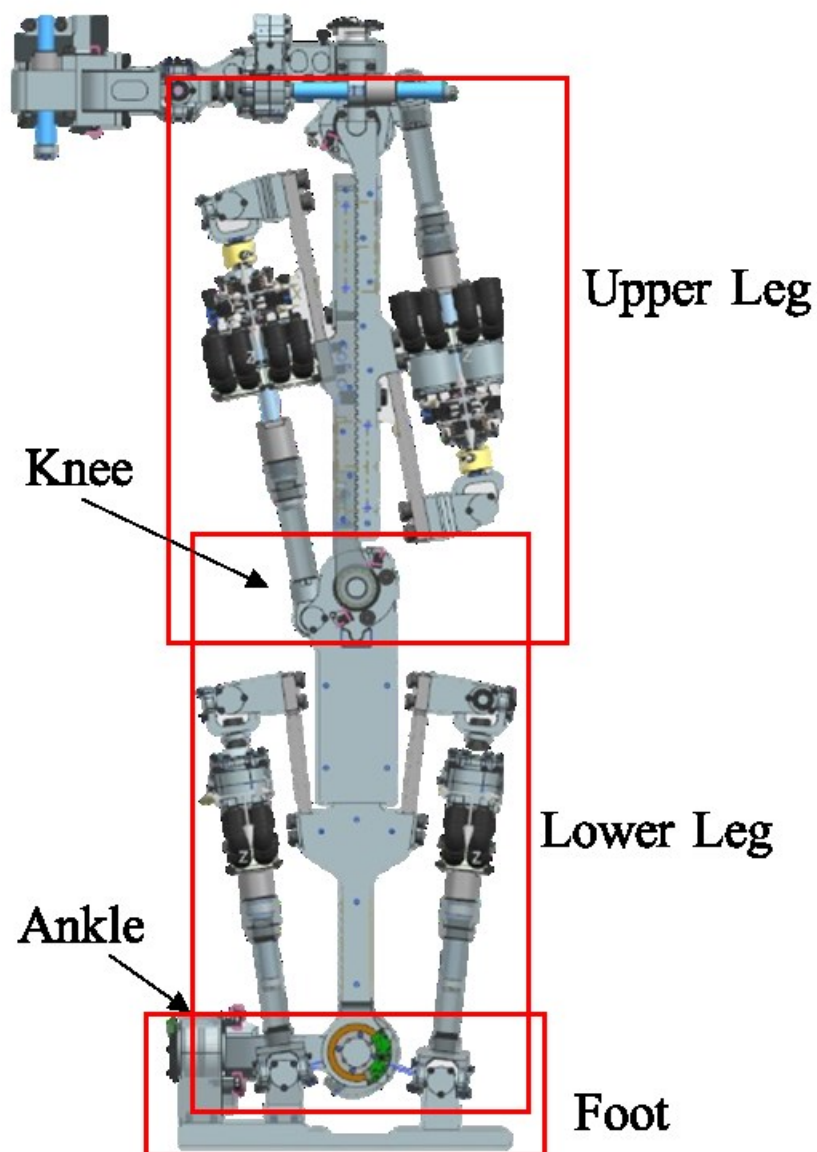


Figure 24: The whole exoskeleton leg with subassembly and joints labeled.

4.1. Upper Leg

The upper leg segment connects the hip to the knee and consists of two identical subassemblies that mesh together along the length of the segment. Each half consists of a main body section and support bracket. The main body provides the structure for the segment and mounts the actuator at a fixed position relative to the joint axis. It also contains large flat surfaces where a variety of hole patterns can be placed for mounting other systems, such as water cooling, power, control, and a safety cover. The main body is sufficiently over designed such that adding these mounting features will not risk structural failure. Since the design of these systems has not been finalized, the exact placement of these holes can be easily modified to match the final designs. The current holes provide mounting for the two motor controllers, the user support structure, and a safety cover. At the extremes of the hip and knee joint's range of motion, hard stops are in place to prevent the exoskeleton from injuring the wearer or damaging itself.

The support bracket provides a second support for the joint pivot and was made a separate piece for ease of manufacturing and assembly. It also over hangs the teeth on the main body to help align the two halves, and mounts the magnetic ring used in the AksIM rotary encoder at each joint. It is mounted to the main body using two M6 bolts. Figure 25 below shows how the pieces fit together, and Table 6 lists the components and materials used in the upper leg.

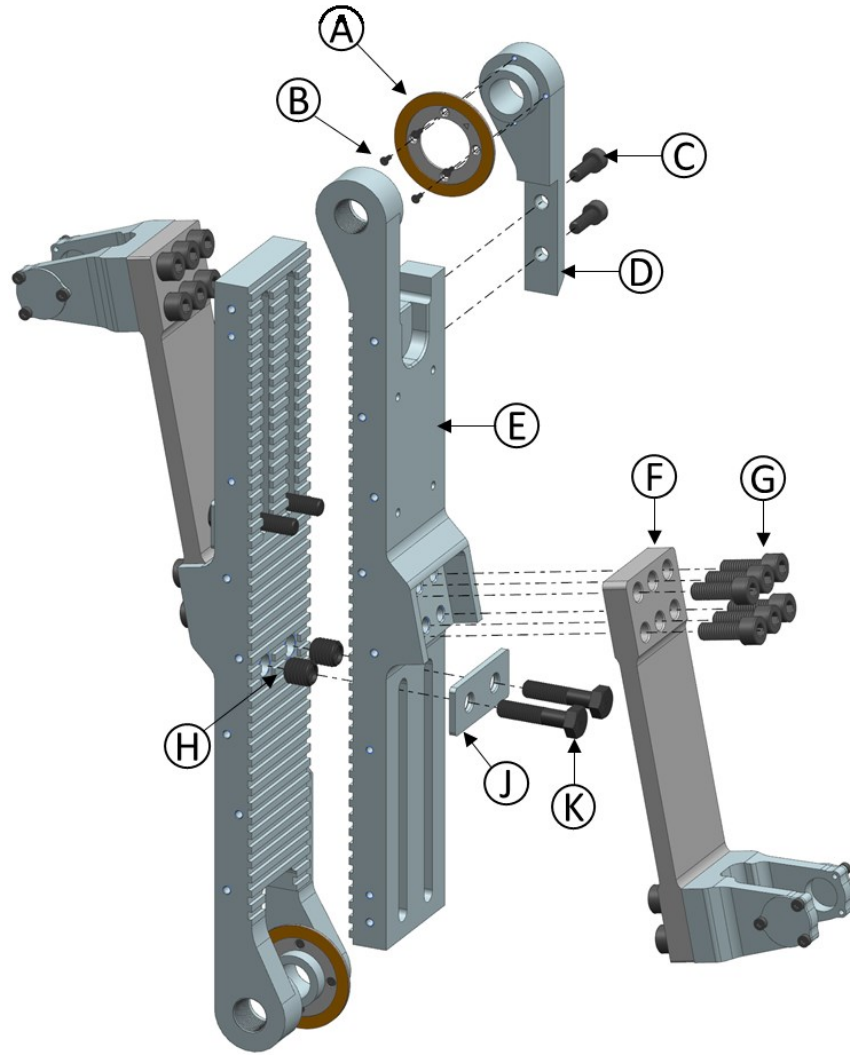


Figure 25: Exploded view of the upper leg segment.

Table 6: Parts and materials used in each upper leg subassembly, including Commercial-off-the-Shelf (CotS) part numbers.

	Part	Qty	Material	CotS Part Number
A	D49 encoder ring	2	Various	MRA7D049AB025E00
B	M2x0.4x6 Flat Screw	8	Class 12.9 Steel	91263A413
C	M6x1x18 Socket Head Cap Screw	4	Class 12.9 Steel	MC91290A323
D	Joint Support Bracket	2	Al 7075	Custom
E	Upper Leg Main Body	2	Al 7075	Custom
F	MkIII Heavy Spring Assembly	2	Various	Custom
G	M8x1.25x22 Socket Head Cap Screw	12	Class 12.9 Steel	91290A428
H	M8x1.25x12.5 Threaded Insert	12	Class 12.9 Steel	97084A240
J	Washer Plate	2	Steel	Custom
K	M8x1.25x35 Hex Head Bolt	4	Class 12.9 Steel	95327A566

The two halves are held together using four 8mm hex head bolts. These bolts fit through slots in the main body of one half and thread into the other half. The bolts tighten against a steel plate and threaded inserts to better distribute the clamping forces and to reduce the wear from repeated loosening and tightening. Originally, these bolts were placed ahead of the spring mount to make adjusting the segment's length easier. However, this was changed for several reasons. The region behind the spring was larger, allowing for a longer slot and a larger range of adjustment. It also meant that, when configured for a larger height, the extended portion of the leg was solid which stiffened the assembly. Using hex head bolts offsets the difficulty some, since a wrench can be used to loosen the bolts enough to change the segment's length without having to remove the spring assembly.

The design was split in half along its axis so that adjustments to the length of the segment would not impact the actuator's performance. Since the joint pivot and actuator are attached by a single piece, their relative position will not change as the segment length is adjusted. Also, because all of the adjustment happens in-line with the segment, changing that length has no input on the angle of the actuator relative to the joint, which would change the angle of peak output. The upper leg can be extended from 393mm to 498mm, accommodating heights from approximately 1.60m to 2.03m (see Figure 26). This adjustability almost completely covers the 5th to 95th percentile range of heights from Table 2 (1.55m – 1.89m) [11]. The height adjustment is in 7mm increments, meaning that the hip or knee pitch axes should never be more than 2mm out of alignment. This range of accommodated heights was skewed towards the tall end for several reasons. Raising the minimum joint-to-joint distance allowed for longer slots and a wider range of accommodated heights and gave more room for the actuator package and other subsystems.

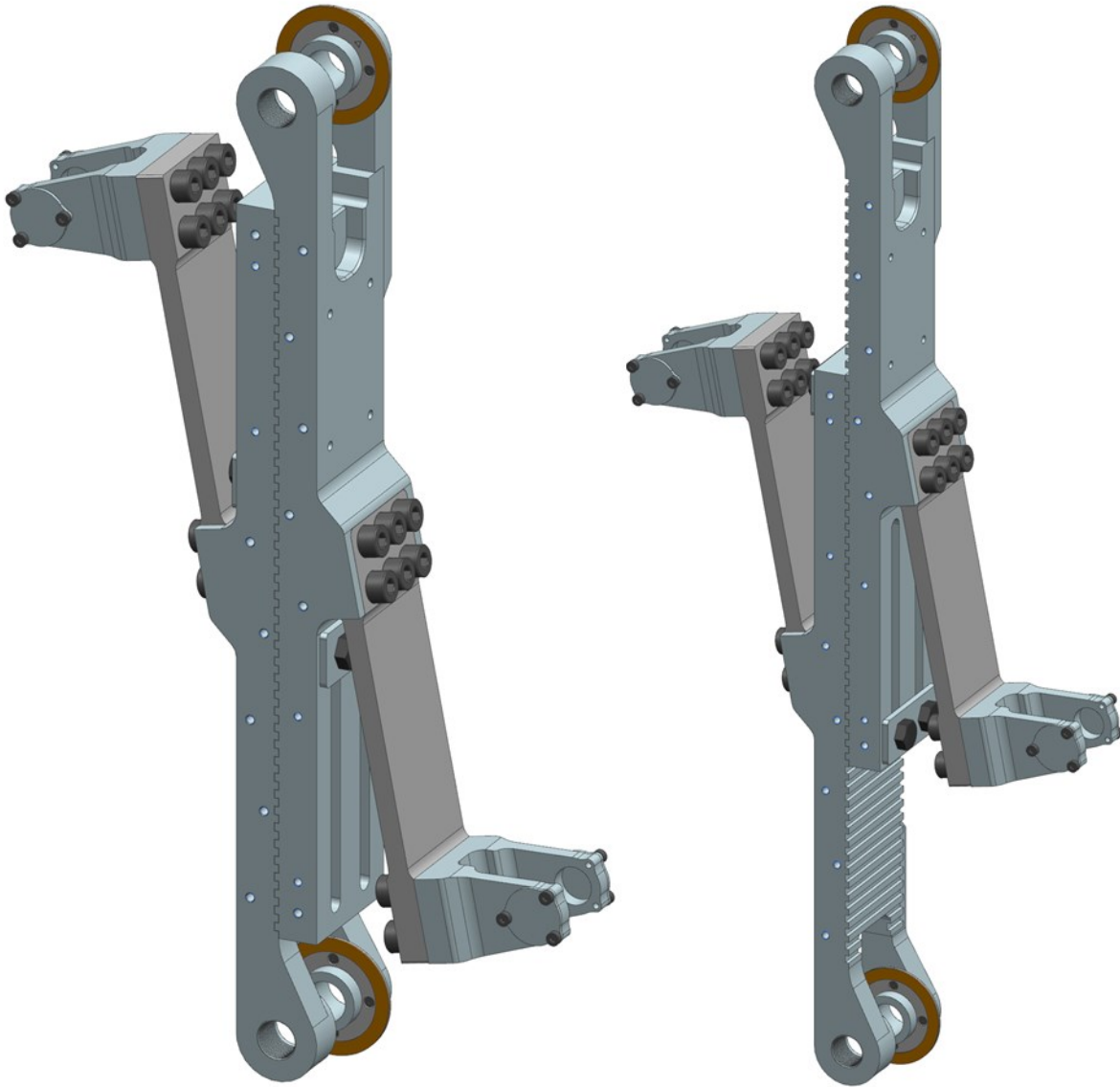


Figure 26: Adjustability of the upper leg, from 393mm to 498mm.

The analysis performed earlier showed that mounting the actuator springs at an angle was necessary to keep the u-joint within acceptable limits. To keep the two halves identical, the same angle of tilt was used for both the hip and knee actuators, 11 degrees. This offset results in the hip actuator trunnion angle ranging from 2.8 to -4.9 degrees, while the knee actuator ranges from 1.9 to -4.4 degrees. This moves the range of angles within the limits established by Kendrick [9]. The spring mounts also included positioning features so that the actuator could be accurately located relative to the joints. Each actuator was held in place with six M8 fasteners.

4.2. Lower Leg

The lower leg consists of two main subassemblies, the knee joint assembly (Figure 27a and b, Table 7) and the shin assembly (Figure 28, Table 8). The knee joint assembly houses the bearings for the knee pitch joint, the pivot for the knee actuator, and clamps to the shin assembly for adjusting the height. The actuator pivot is a 10mm rotary shaft held in place by two tabs on the body of the knee housing. Even with the faster configuration of the MkIII Heavy¹, the actuator pivot needed to be very close to the joint axis to reach the maximum needed speeds during free swing. Placing the pivot so close to the joint required a cutout in the joint housing for the end of the thrust assembly and meant that the actual offset, ϕ , was larger than recommended by the analysis. While this contributed to the lag between the needed and provided joint velocities, its effect was relatively minor. MATLAB (GaitCycleCheck) was used to determine what changes could be made that met the joint needs while still fitting space requirements.

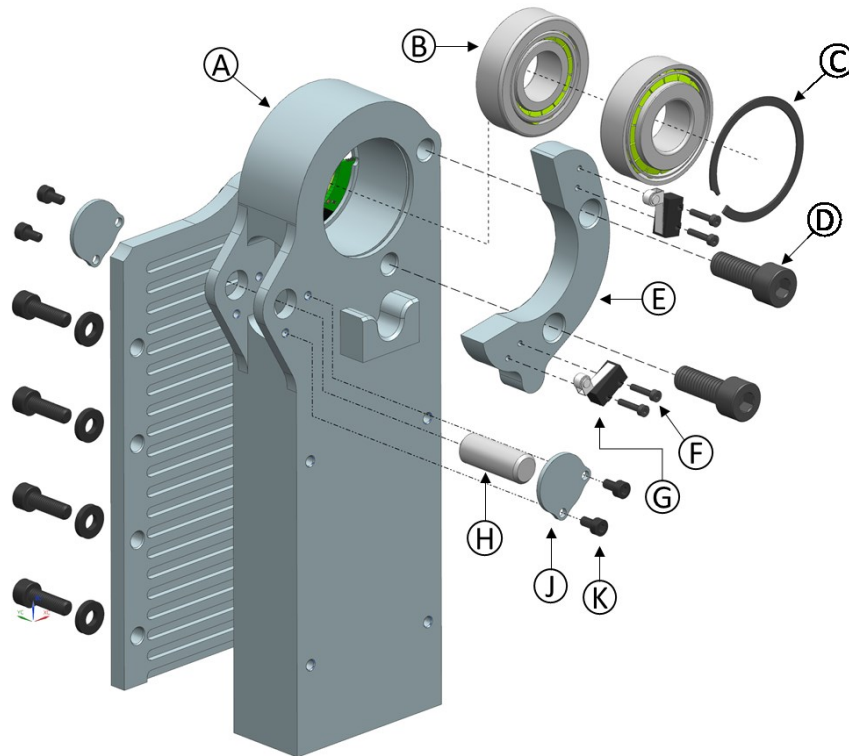


Figure 27a: Exploded view of the outside of the knee joint assembly.

¹ A faster internal gearing for the knee actuator was not possible, as it would exceed the safe rotational speed of the ball screw. Attempts to find a ball screw capable of faster speeds while still having the same strength and precision were unsuccessful.

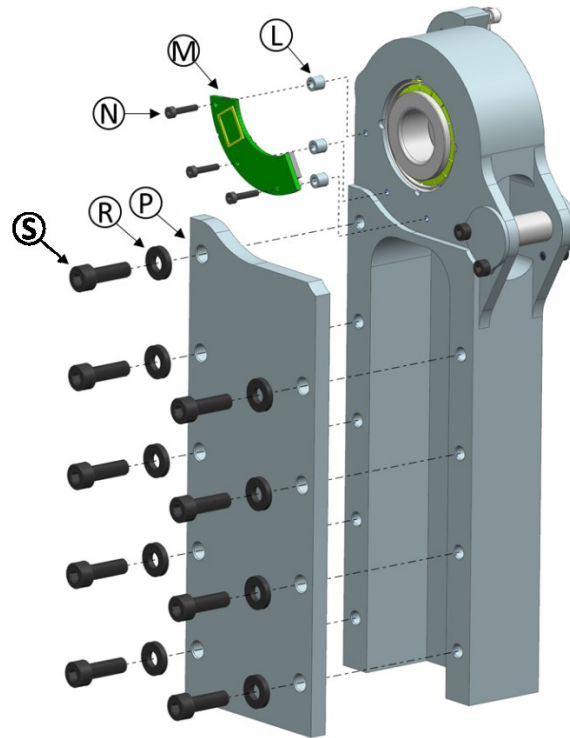


Figure 27b: Exploded view of the inside of the knee joint assembly.

Table 7: Parts and materials used in the knee joint assembly.

	Part	Qty	Material	CotS Part Number
A	Knee Joint Housing	1	Al 7075	Custom
B	SKF 17mm Tapered Roller Bearing	2	Various	SKF-30203 J2
C	40mm Internal Snap Ring	1	Steel	VHM-40-S16
D	M6x1x18 Socket Head Cap Screw	2	Class 12.9 Steel	91290A323
E	Knee Pitch Hardstop	1	Al 6061	Custom
F	M2x0.4x10 Socket Head Cap Screw	4	Class 12.9 Steel	91290A017
G	Digikey Limit Switch	2	Various	D2F 01FL2
H	M10x26 Rotary Shaft	1	Steel	1482K11
J	Pin Cover Plate	2	Al 6061	Custom
K	M3x0.5x6 Socket Head Cap Screw	4	Class 12.9 Steel	91290A111
L	2.7mmID x 4.5OD x 4mm Spacer	3	Aluminum	94669A096
M	AksIM MBA7-C42 Encoder Read Head	1	Various	MBA7-C42
N	M2x0.4x14mm Socket Head Cap Screw	3	Class 12.9 Steel	91290A045
P	Knee Clamp Plate	1	Al 7075	Custom
R	M5 Flat Washer	9	Steel	98035A103
S	M5x0.8x16 Socket Head Cap Screw	9	Class 12.9 Steel	91290A232

The shin assembly mounts both MkIII Lite actuators for the ankle joints and houses the bearings for the ankle pitch joint. Like the upper leg, the shin assembly has large flat areas for mounting control boards, cooling, and a safety cover. Extended up from the actuator mounts is a large tongue with the same 7mm tooth pattern as the upper leg. This tongue fits into the knee joint assembly and is held in place by a clamping plate with matching slots. The lower leg can be extended from 295mm to 500mm, accommodating the same range of heights as the upper leg. The clamping plate is secured to the knee assembly with seven M5 bolts. To simplify manufacturing, both actuators are mounted at the same position relative to the joint, mirrored across the segment axis, using the same angle of tilt. This gives the front actuator a range of -1.78 to 0.3 degrees and the rear -3.92 to 0.3 degrees when moving through the pitch range of motion. When moving through ankle roll, the actuator pivots move through -2.23 to 0.5 degrees and -3.8 to 0.5 degrees, respectively.

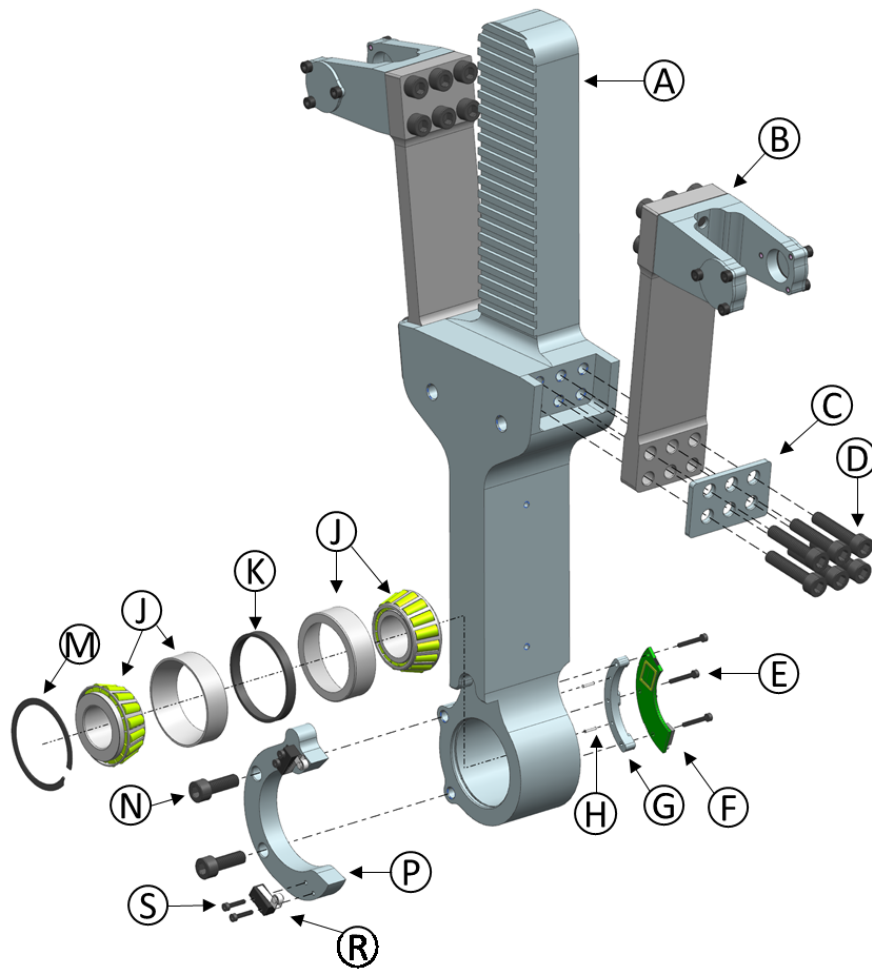


Figure 28: Exploded view of the shin assembly.

Table 8: Parts and materials used in the lower leg shin assembly.

	Part	Qty	Material	CotS Part Number
A	Shin Body	1	Al 7075	Custom
B	MkIII Lite Spring Assembly	2	Various	Custom
C	Washer Plate	2	Steel	Custom
D	M6x1x30 Socket Head Cap Screw	12	Class 12.9 Steel	91290A332
E	M2x0.4x14 Socket Head Cap Screw	3	Class 12.9 Steel	91290A045
F	AksIM Encoder Read Head	1	Various	MBA7-C42
G	Shin Read Head Mount	1	Plastic	Custom
H	M1.5x6 Dowel Pin	3	Stainless Steel	91585A007
J	SKF 20mm Tapered Roller Bearing	2	Various	SKF-32004 X/Q
K	39mm Bearing Spacer	1	Steel	8486A901
M	42mm Internal Snap Ring	1	Steel	VHM-42-S16
N	M6x1x18 Socket Head Cap Screw	2	Class 12.9 Steel	91290A323
P	Ankle Pitch Hard Stop	1	Al 6061	Custom
R	Digikey Limit Switch	2	Various	D2F 01FL2
S	M2x0.4x10 Socket Head Cap Screw	4	Class 12.9 Steel	91290A017

Because both actuators drive motion for both the pitch and roll axis, changes in pitch or roll will impact the performance of the actuators in the other axis. This is not an issue for torque and speed, as the combined actuators easily exceed the requirements for both axes. However, this interaction does limit the range of motion of the ankle joints. Each actuator has a limited amount of travel dependent on the threaded length on the ball screw. The distance between the joint pivots and the actuator allows for a thread length of 130 mm. When only moving in one axis, that thread is sufficient for the joint to move through its full range of motion. But, as one joint moves, it changes the center point for the other axis, restricting its motion in one direction or the other. Figure 29 below shows what combinations of pitch and roll are allowed by the two actuators.

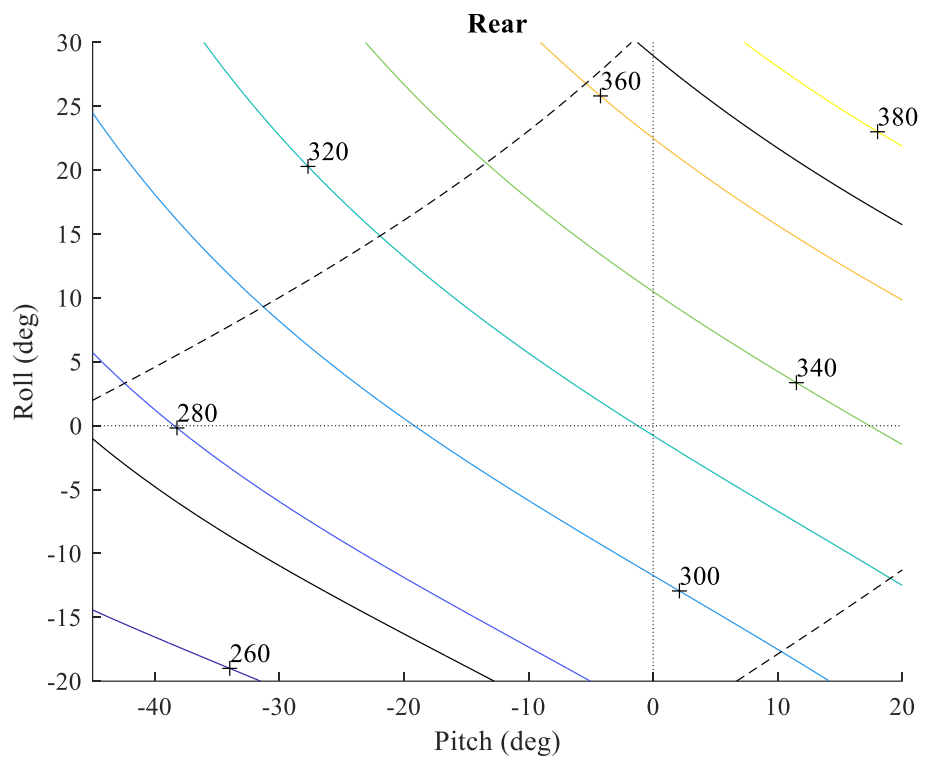
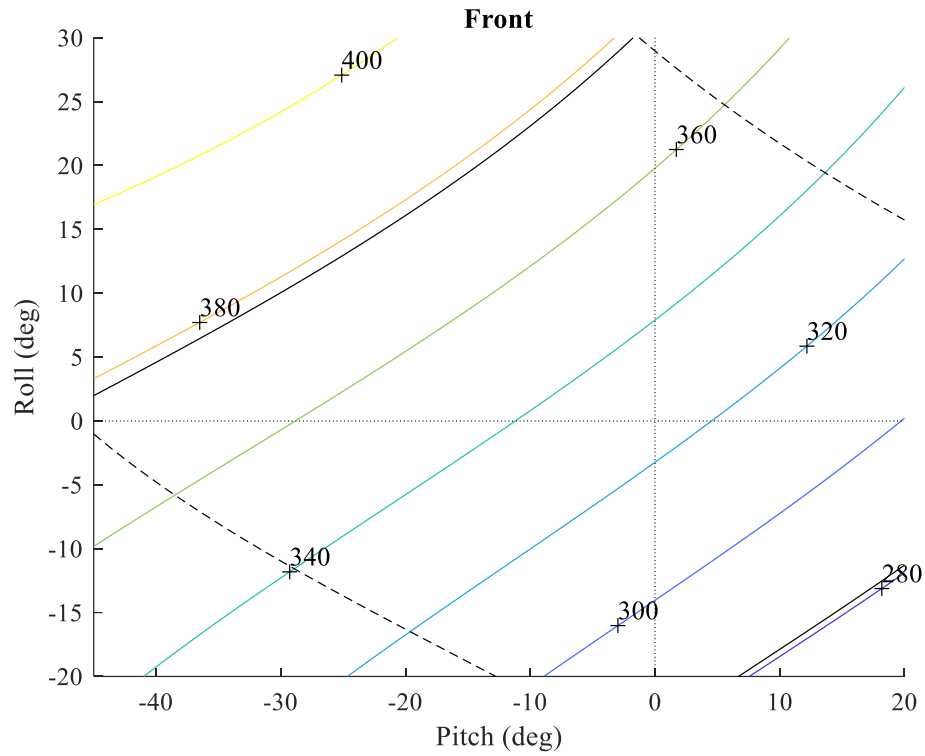


Figure 29: Actuator package length as a function of ankle pitch and roll for the front (top) and rear (bottom) actuators. The solid black lines indicate the maximum and minimum lengths for the actuator, while the dashed lines indicate those same limits for the opposing actuator.

Moving the actuators up the leg would allow for longer ball screws and a wider range of motion at the ankle, but would interfere with the upper leg as the knee approaches 90° while in the minimum height configuration.

4.3. Ankle/Foot

The foot of the exoskeleton consists of a simple plate to support the user's foot and an L-shaped strengthener that mounts the ankle roll housing and the lower u-joints for the two ankle actuators. To reduce part count, these are the same u-joints used to mount the actuators to the spring. The foot is connected to the lower leg via the ankle assembly, a curved piece with threaded holes to accept the pins for both ankle joints. The ankle piece needs to fit in the very tight space between the bearing housings, actuator mounts, and the user's ankle. Examination of the 3D model show that the current design should have enough clearance, but physical prototyping using 3D printing should be used to confirm this once the user support system is finalized. Attaching the rear actuator to the ankle instead of the foot would eliminate the clearance issue but would leave only the front actuator with roll authority. In this configuration, some combinations of needed roll and pitch torque would significantly exceed the capabilities of a single MkIII Lite, and a MkIII Heavy could not move fast enough.

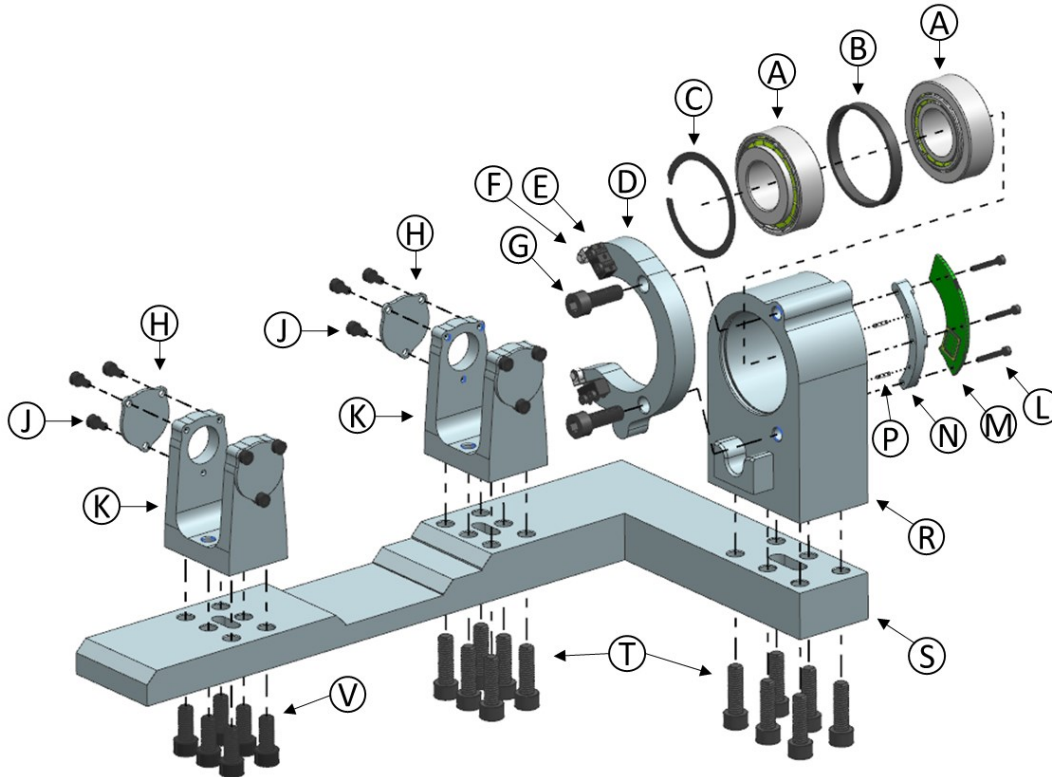


Figure 30: Exploded view of the foot assembly (foot plate hidden).

Table 9: Foot parts and materials.

Part	Qty	Material	CotS Part Number
A	2	Various	SKF 32004 X/Q
B	1	Steel	8486A901
C	1	Steel	VHM-42-S16
D	1	Al 6061	Custom
E	2	Various	D2F 01FL2
F	4	Class 12.9 Steel	91290A017
G	2	Class 12.9 Steel	91290A323
H	4	Al 6061	Custom
J	12	Class 12.9 Steel	91290A111
K	2	Al 7075	Custom
L	3	Class 12.9 Steel	91290A045
M	1	Various	MBA7-C42
N	1	Plastic	Custom
P	3	Stainless Steel	91585A007
R	1	Al 6061	Custom
S	1	Al 6061	Custom
T	12	Class 12.9 Steel	91290A326
V	6	Class 12.9 Steel	91290A045

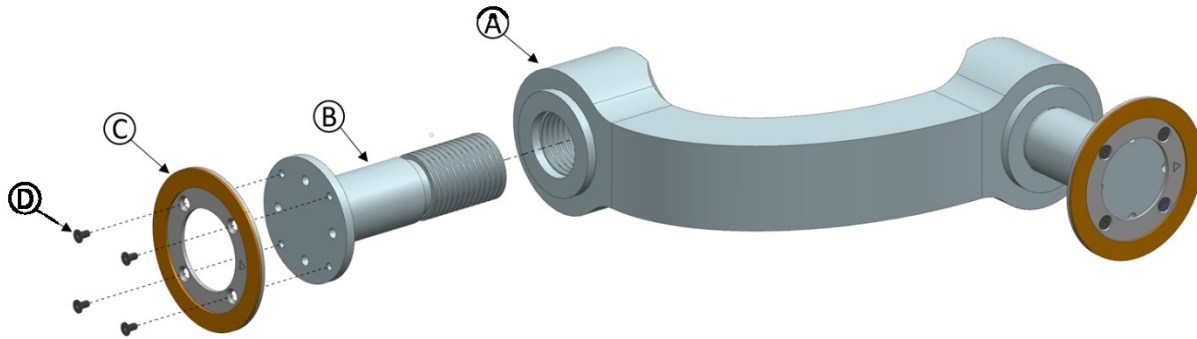


Figure 31: Ankle exploded view.

Table 10: Ankle parts and materials.

	Part	Qty	Material	CotS Part Number
A	Ankle Body	1	Al 7075	Custom
B	Ankle Joint Shaft	2	AISI 1144 Steel	Custom
C	D49 encoder ring	2	Various	MRA7D049AB025E00
D	M2x0.4x6 Flat Screw	8	Class 12.9 Steel	91263A413

The u-joints for the ankle actuators are positioned in the ideal position around the pitch joint when roll is zero. Movement about the roll axis changes the effective lever arm, altering the output profiles for both pitch and roll, but this should not be a problem as both joints have significant overhead.

For simplicity, the foot does not include any adjustability in the height from the floor to the ankle joints. To match the adjustability of the upper and lower legs, the foot height only needs to change from 63mm to 80mm. The height is fixed at 90 mm to also allow for the user's shoe. If this distance needs to be decreased, additional padding can be added between the user's foot and the exoskeletons to bring the ankle joints into alignment.

4.4. Joints

Each joint in the legs rotates around a custom steel pin through two tapered roller bearings. The bearings are mounted in opposition of each other so that the clamping force from the pin centers the bearings, allowing the joint to handle loads from multiple directions. For position information, each joint includes an AksIM MBA7-C42 off axis magnetic encoder. The read head is mounted

on the joint housing, while the magnetic ring attaches to the connected assembly for double supported joints. For the single support ankle joints, the magnetic ring is attached directly to the pin after it has been tightened. To ensure that each sensor and ring pair have the exact required spacing, the read heads are mounted on custom spacers that can be easily modified for the perfect fit.

Each joint also has removable hard-stops that are the first physical point of contact when each joint reaches the extremes of its range of motion. These hard-stops protect the exoskeleton from damaging expensive components, as the ball screws or thrust tube assemblies would generally otherwise be the first point of contact. The hard-stops also protect the user from overextending their joints. With the exception of hip pitch, all of the hard stops are removable and could be customized to further limit a joint's range of motion to protect the user if they have reduced flexibility in that joint. The hard stops also mount two Digikey D2F 01FL2 limit switches, one for each limit that let the control system know that one the limits have been reached in case the position sensor fails. These limit switches can be adjusted to change when exactly they trigger, and right now are set to trigger when the joint comes within two degrees of the limit.

As mentioned earlier, the interaction of the ankle pitch and roll joints means the limits of each individual axis are constantly changing based on the position of the other axis. While the hard stops prevent each axis from hitting its extreme limits, they cannot account for the shifting thread limits of each actuator. These limits will need to be carefully monitored in software to prevent damage to the ball screw.

5. Finite Element Analysis

Before finalizing the design, key components were validated using Finite Element Analysis (FEA). FEA was used in addition to classical analysis in order to better capture the interactions between the components and to check for stress concentrations in individual pieces. Many of the smaller structures, such as the u-joints and actuators, were previously examined by Kendrick [9] and Li [8], so this analysis focused on the upper and lower leg assemblies and their response to bending and axial loads.

Both assemblies were analyzed using similar strategies. The structures were set to the maximum design length, and both pivot points of each joint were fixed in place as if some interference was preventing the leg from moving. Then, the maximum force from each actuator was applied in different combinations of tension and compression, so that the structure would experience both axial and bending loads.

The analysis was performed in Siemens ABAQUS 2016 using C3D8R elements. The C3D8R element is an 8-node hexahedral element that uses reduced integration to shorten the analysis time. This allows for a finer mesh to be used and faster iteration on the design. When modeled, most components were simplified to some extent to allow for a simpler model and mesh. These simplifications were carefully monitored to make sure they had not hidden or added a critical location. If so, the part was remodeled to more closely match the actual design.

5.1. Upper Leg

For the upper leg, the main body and support bracket, as well as the fasteners and clamp plate were modeled as deformable bodies. The pivot pins and actuator assemblies were modeled as rigid bodies. To represent the teeth on the main body preventing the halves from sliding relative to each other, a constraint was used to prevent sliding in the z direction. Similarly, the bolts were modeled at the nominal diameter with constraints preventing them from sliding out of the threaded holes. As mentioned previously, the pivot pins in each joint were fixed in place. This force was angled as if the person was standing from a seated position, the position where peak loads are expected. Additionally, a 50% preload was applied to each fastener.

The mesh was refined iteratively until stress reached 3% convergence using approximately 600,000 elements, typically 1mm in length. The largest stresses were seen when both actuators

were in compression. Peak stresses were seen around the fasteners holding the pieces together. The joint brackets show a peak stress of 209 MPa around the first bolt hole. The main body sees a max stress of 200 MPa around the large bolts holding the two halves together. Since this was an area that had been simplified in the model, and because these fasteners will be repeatedly removed and reinserted to adjust the length of the assembly, threaded inserts were added to strengthen this joint and prevent thread deformation. Figure 32 through Figure 34 show displacement and stress distributions through the assembly for the different load cases, and Table 11 summarizes the results of the analysis.

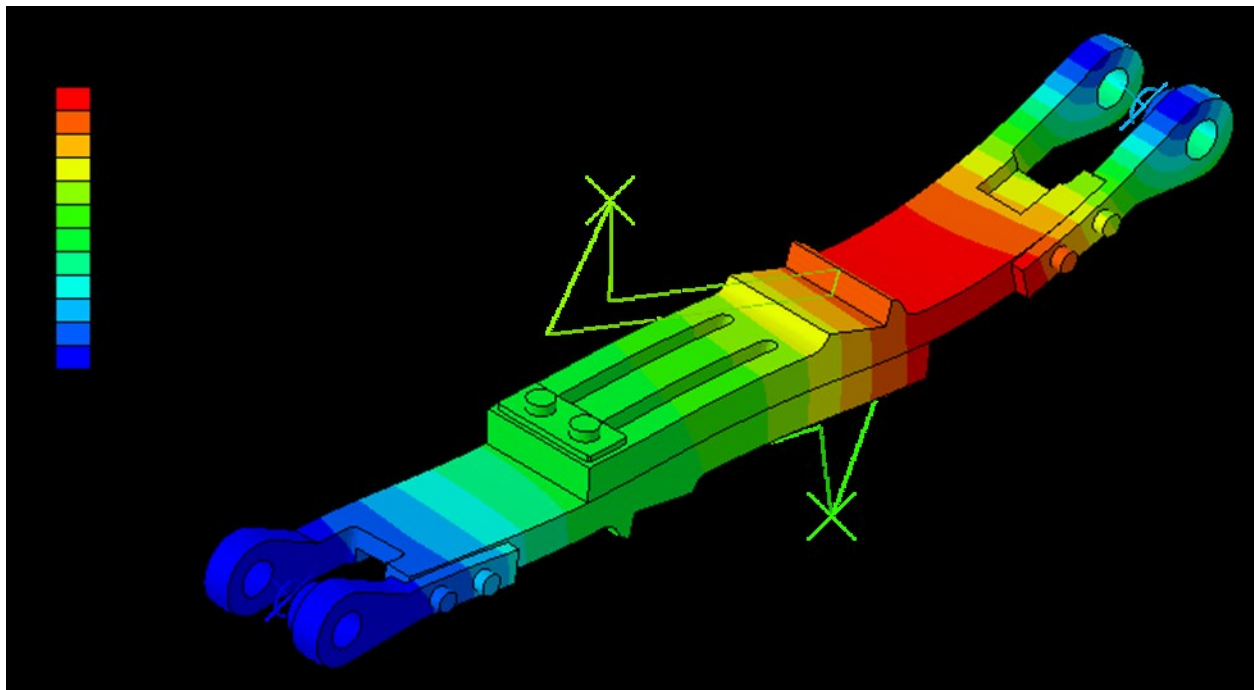


Figure 32: Exaggerated displacement of the upper leg assembly in the worst-case loading.

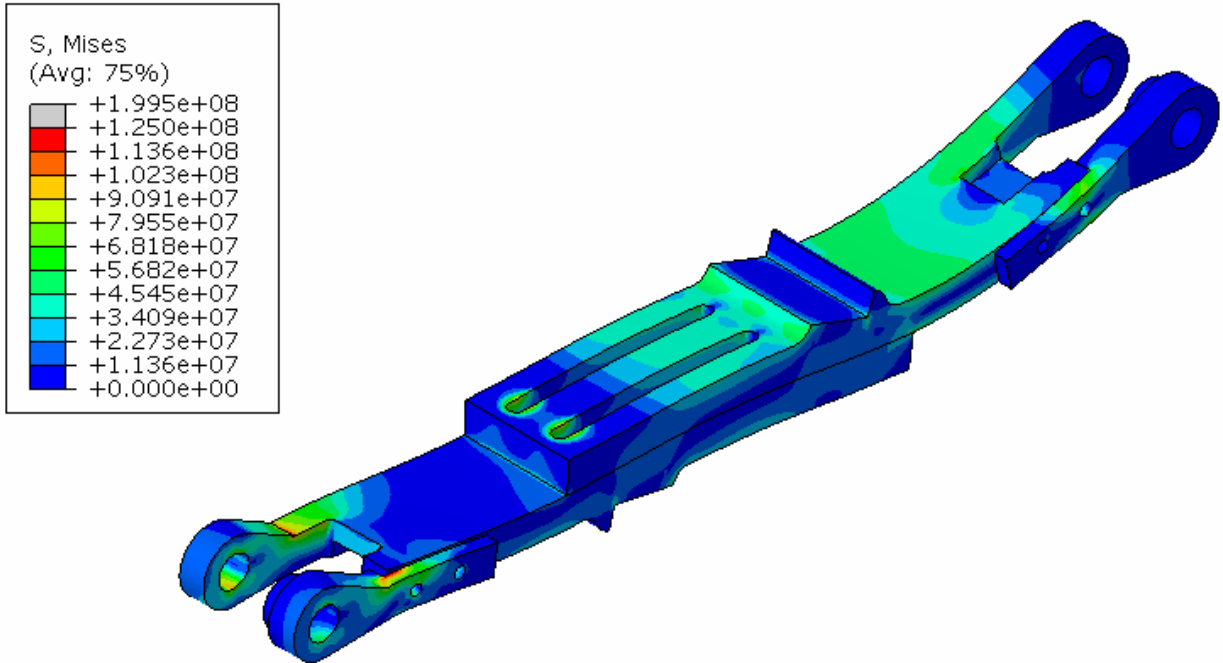


Figure 33: Von mises stress distribution in the worst-case loading. (Some components hidden.)

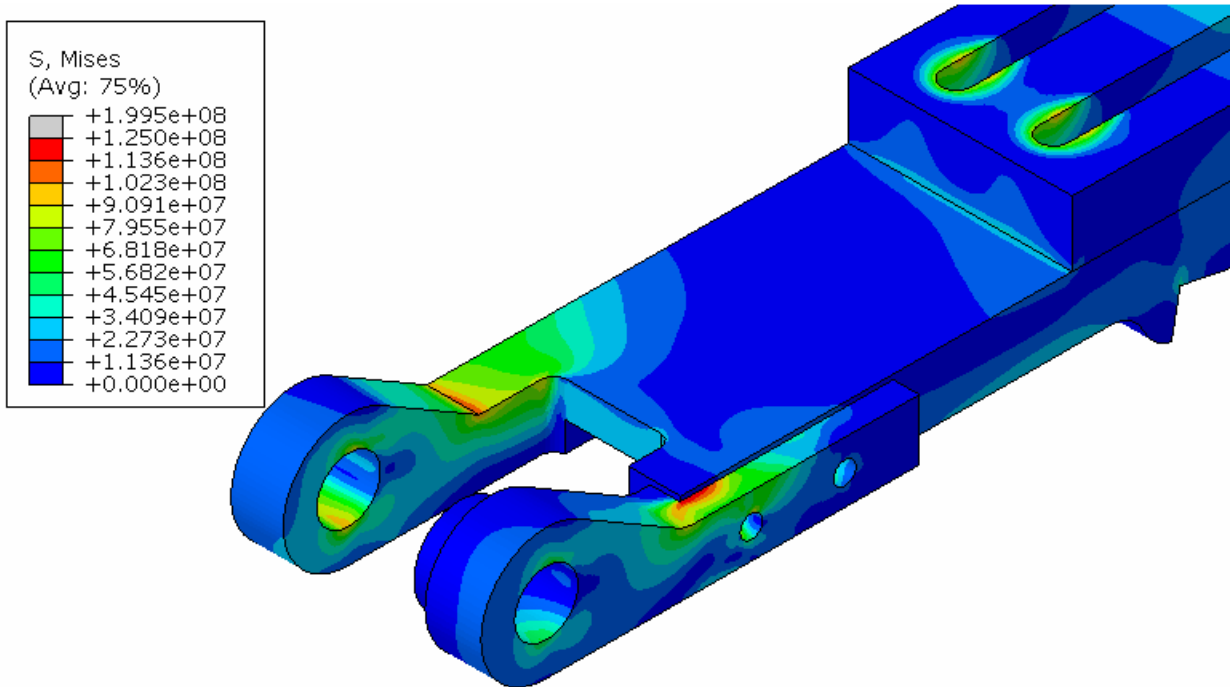


Figure 34: Close-up of stress concentration near the knee joint. (Some components hidden.)

Table 11: Peak stress in main components across all load cases.

Part	Peak Stress		Yield Strength	Safety
	[MPa]	Material		
Upper Leg Body (Hip)	1.85E+08	Al 7075	4.62E+08	2.49
Upper Leg Body (Knee)	2.00E+08	Al 7075	4.62E+08	2.32
Joint Bracket (Hip)	2.09E+08	Al 7075	4.62E+08	2.22
Joint Bracket (Knee)	2.09E+08	Al 7075	4.62E+08	2.21
Washer Plate	1.63E+08	Steel	5.00E+08	3.07
M6 Fastener	1.93E+08	Class 12.9 Steel	8.96E+08	4.64
M8 Fastener	2.93E+08	Class 12.9 Steel	8.96E+08	3.06

5.2. Lower Leg

The lower leg assembly was examined using two analyses. The first modeled the assembly as a whole, using a similar strategy as the upper leg. After this analysis had been performed, the knee joint housing was redesigned to bring the knee actuator mount closer to the pivot axis. This redesign removed a significant amount of material around the joint housing and pivot, so a second analysis was performed to ensure that the new design still held up to the worst-case actuator loading. This analysis focused on just the knee joint housing and how it reacted to the knee actuator load.

Like the upper leg, peak stresses were located around the fasteners holding the assembly together in the case where both actuators were pushing in the same direction. The knee joint housing had a peak stress of 308 MPa at the threads of the bottom right fastener, while the plate had a peak stress of 334 MPa underneath the head of the same bolts. These stresses correspond to safety factors of 1.45 and 1.36 respectively. The ankle body saw a peak stress of 180 MPa where the bending pressed it against the knee joint housing. This was the only peak stress that changed noticeably as the actuator loading changed. Peak deflection was 0.23 mm in the same load case. Figure 35 - Figure 37 and Table 12 summarize the results of this analysis. For the redesigned knee joint housing, peak stress was 133 MPa, located directly under the contact point of the steel pin. Figure 38 - Figure 40 show the results of this analysis.

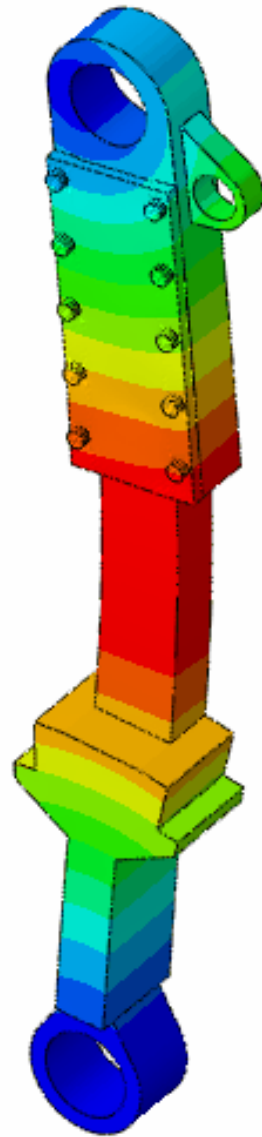
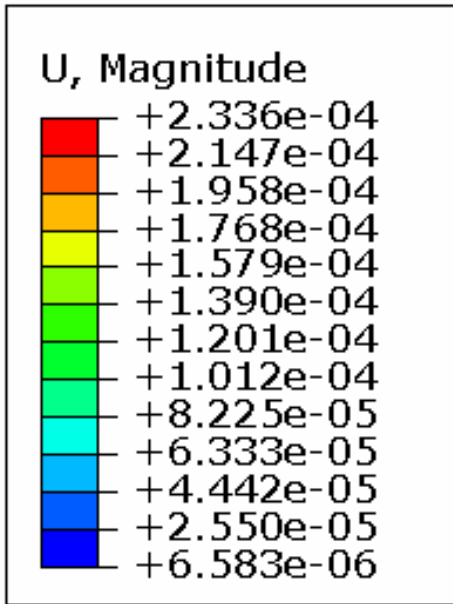


Figure 35: Exaggerated displacement of the lower leg assembly.

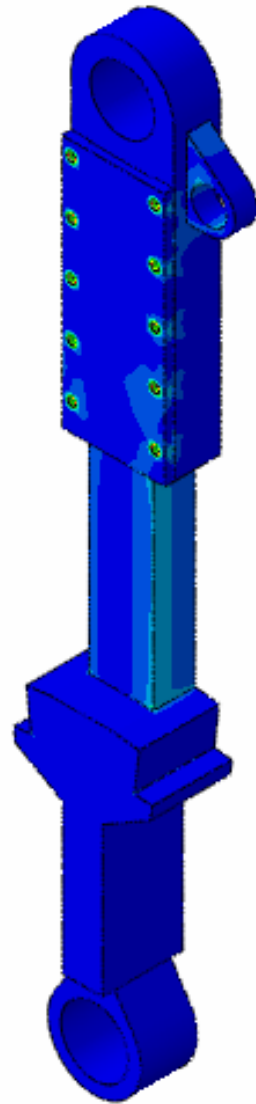
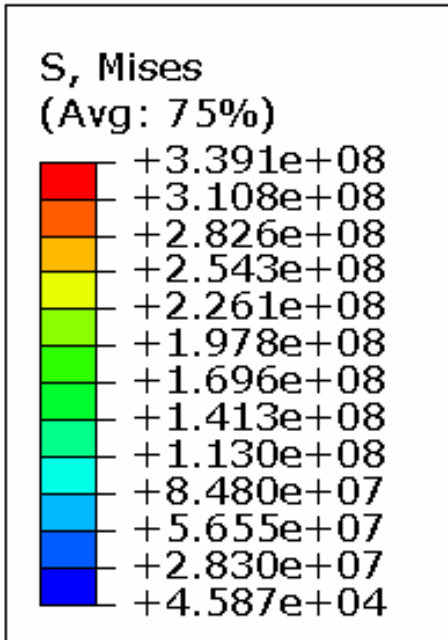


Figure 36: Von mises stress distribution in the lower leg assembly. (Some components hidden.)

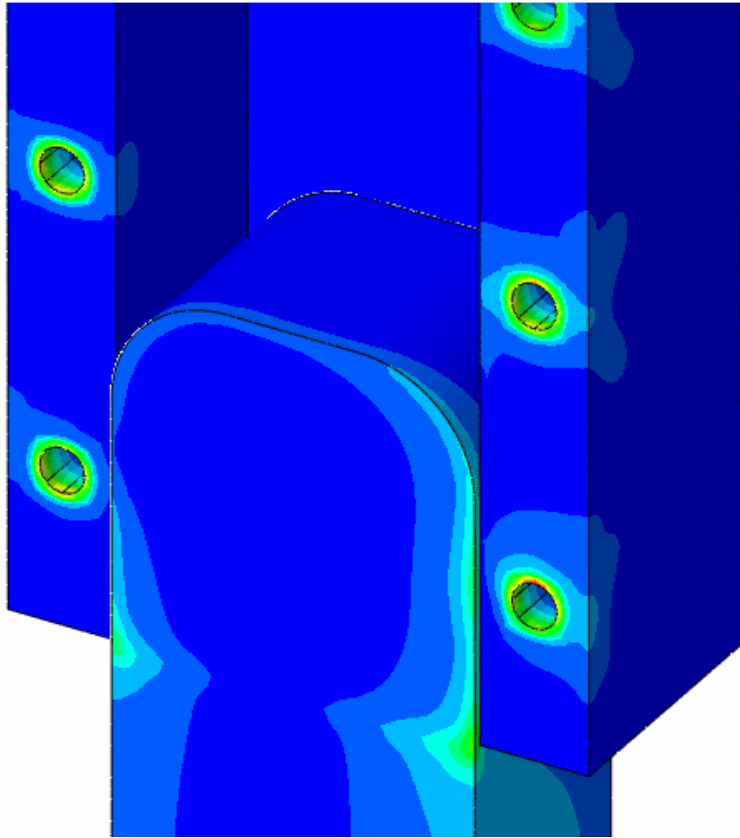
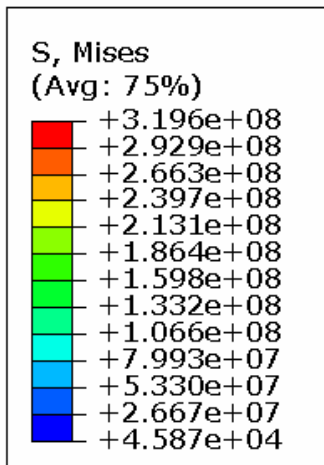


Figure 37: Close up of peak stresses in knee joint housing and shin body.

Table 12: Summary of FEA Results of the lower leg analysis

Part	Peak Stress [MPa]	Material	Yield Strength [MPa]	Safety Factor
Knee Joint Housing (V1)	3.20E+08	Al 7075	4.62E+08	1.45
Knee Joint Housing (V2)*	1.33E+08	Al 7075	4.62E+08	3.49
Knee Clamp Plate	3.39E+08	Al 7075	4.62E+08	1.36
Shin Body	1.80E+08	Al 7075	4.62E+08	2.56
M5 Fasteners	4.52E+08	Class 12.9 Steel	8.96E+08	1.98

* Only includes loading from knee actuator

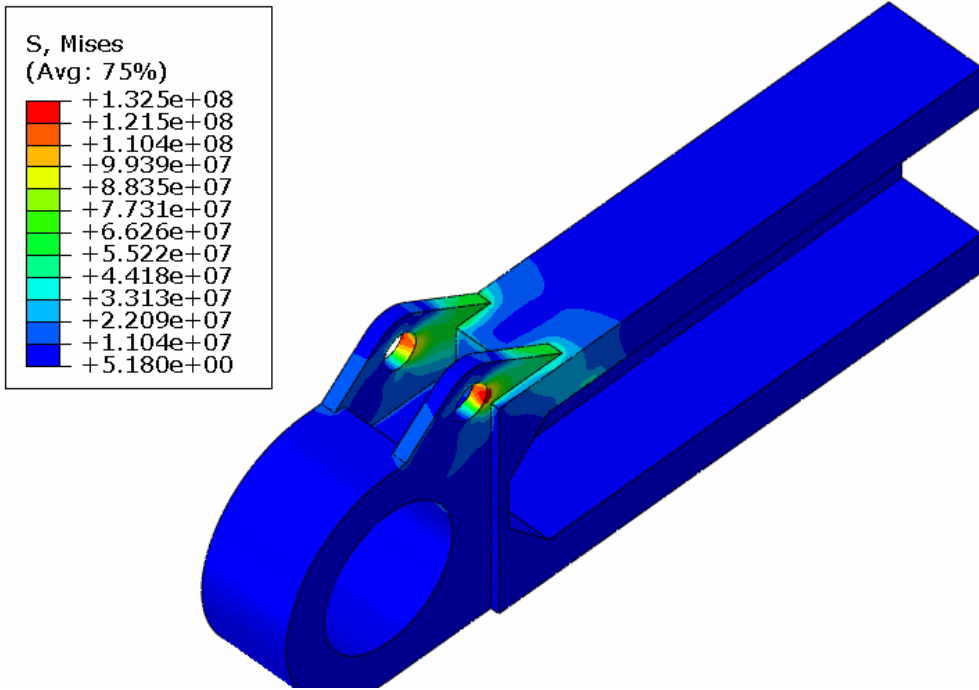


Figure 38: Overall von Mises stress distribution in the modified knee joint housing.

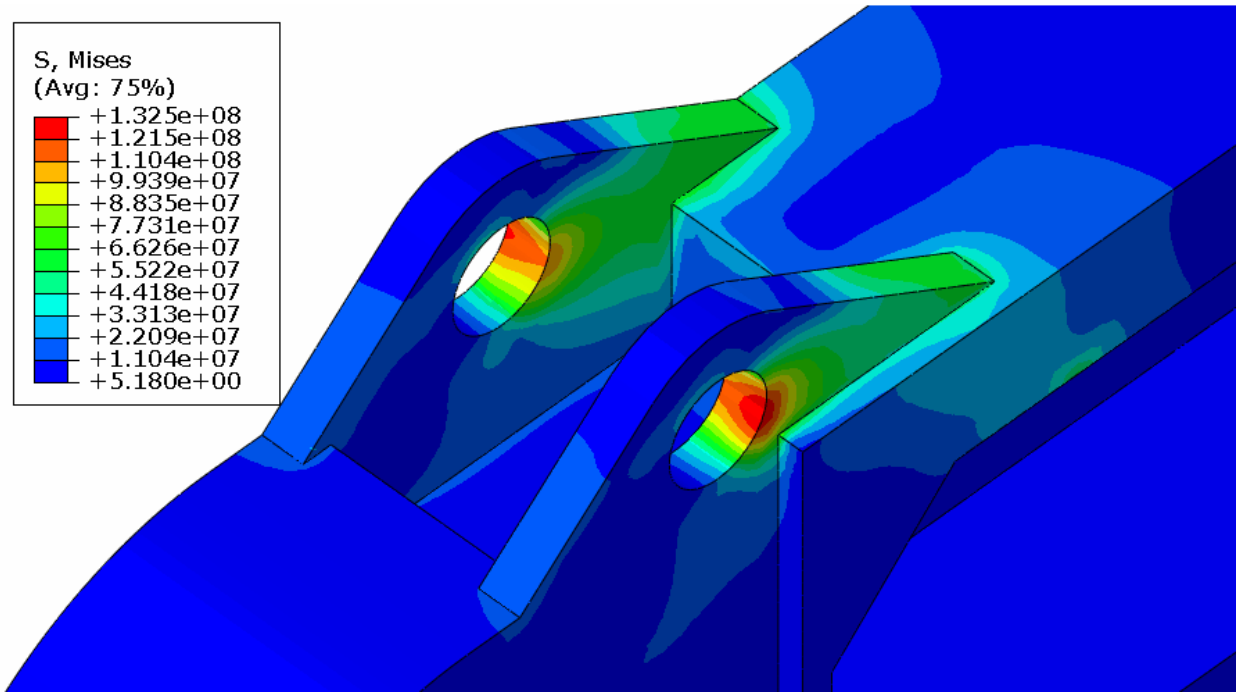


Figure 39: Closeup of contact stress from the actuator pin on the modified knee joint housing.

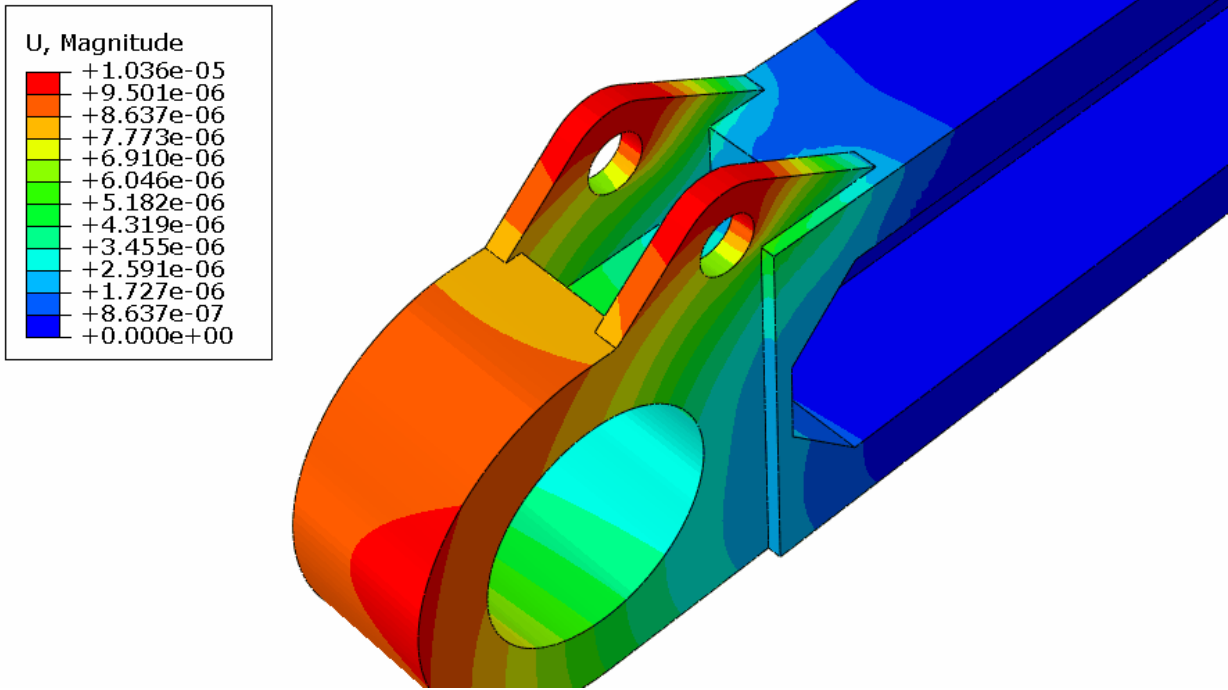


Figure 40: Displacement in the modified knee joint housing.

6. Conclusion

The completed mechanical design was largely successful in meeting its requirements. The exoskeleton can be precisely adjusted to accommodate heights of 1.60 m to 2.03 m. The system came in well under its mass budget, using only 22 kg of the allotted 35 kg. The remaining 13 kg will be more than sufficient for the actuators, controllers, cooling and cover. The exoskeleton's range of motion is more than enough for normal walking and stair climbing. When the exoskeleton is near its minimum height, possible contact between the rear ankle actuator and upper leg restricts how tightly the knee can bend. However, this could be compensated for by increasing hip and ankle bending or using one's arms to help push themselves to the point where the ankles can take over. Joint speed was able to match the requirements for the different gaits examined, though only barely at the knee. In order to have additional margin for stumble recovery, a slight reduction in walking speed may be needed. While this would be slower than a healthy human walking, it would still outpace the robots currently being developed at Virginia Tech. Torque was also sufficient at the hip and ankle, with plenty of margin for stumble recovery. Knee torque may be sufficient, depending on the walking strategy used. The bent-legged gait currently used by most robots require much more torque than the targeted straight-legged human gait. Additionally, the completed system coming in under its weight budget will also decrease the torque requirements for all gaits and joints.

6.1. Future Work

With the mechanical design completed, several tasks still remain to complete the design of the legs. Electronics, cabling, and coolant hoses will need to be mounted on the upper and lower legs, as well as attachment points for whatever system is used to support the wearer. Once these designs are completed, covers will be added to protect any of these systems from snagging or other harm, as well as to protect the user from any pinch hazards around the joints and actuators. Tentative interfaces were put in place to accommodate these systems throughout the legs. As the design evolves, the size and location of the mounting holes could easily change to work with the new system.

References

- [1] J. Wall and T. Colley, "Preventing pressure ulcers among wheelchair users: preliminary comments on the development of a self-administered risk assessment tool," *J. Tissue Viability*, vol. 13, no. 2, pp. 48-50, 52-54, 56, 2003.
- [2] H. K. Uthoff and Z. F. Jaworski, "Bone loss in response to long-term immobilisation," *Bone Jt. J.*, Vols. 60-B, no. 3, pp. 420-429, 1978.
- [3] E. G. Pearson, P. W. Nance, W. D. Leslie and S. Ludwig, "Cyclical etidronate: Its effect on bone density in patients with acute spinal cord injury," *Arch. Phys. Med. Rehabil.*, vol. 78, no. 3, pp. 269-272, 1997.
- [4] K. A. Curtis, G. A. Drysdale, R. D. Lana, M. Kolber, R. S. Vitolo and R. West, "Shoulder pain in wheelchair users with tetraplegia and paraplegia," *Arch. Phys. Med. Rehabil.*, vol. 80, no. 4, pp. 453-457, 1999.
- [5] "PHOENIX Medical Exoskeleton," [Online]. Available: <https://www.suitx.com/phoenix-medical-exoskeleton>.
- [6] "What's HAL? The world's first cyborg-type robot "HAL"," [Online]. Available: <https://www.cyberdyne.jp/english/products/HAL/index.html>.
- [7] "Austin | Berkley Robotics & Human Engineering laboratory," [Online]. Available: <https://bleex.me.berkeley.edu/research/exoskeleton/medical-exoskeleton/>.
- [8] X. Li, "Structural Design of a 6-DoF Hip Exoskeleton using Linear Series Elastic Actuators," Blacksburg, 2017.
- [9] J. T. Kendrick, "Design of High-Performance, Dual-Motor Liquid-Cooled, Linear Series Elastic Actuators for a Self-Balancing Exoskeleton," Blacksburg, 2018.

- [10] N. Pain, S. Oh and L. Sentis, "Design and Control Considerations for High-Performance Series Elastic Actuators," *IEEE/ASME Transactions on Mechatronics*, vol. 19, no. 3, pp. 1080-1091, 2019.
- [11] The Eastman Kodak Company, *Kodak's Ergonomic Design for People at Work*, Hoboken: John Wiley & Sons., 1986, pp. 50-53.
- [12] J. Perry and J. M. Burnfield, *Gait Analysis: Normal and Pathological Function*, Thorofare: SLACK Incorporated, 1992.
- [13] C. L. Vaughan, B. L. Davis and J. C. O'Connor, *Dynamics of Human Gait*, Human Kinetic Publishers, 1992.
- [14] R. Griffin, *Atlas Fast Straight Leg Joint Data*, Virginia Polytechnic Institute and State University: TREC Lab, 2018.
- [15] M. E. Roebroeck, C. A. Doorenbosch, R. Jacobs and G. J. Lankhorst, "Biomechanics and muscular activity during sit-to-stand transfer.," *Clinical Biomechanics*, vol. 9, no. 4, pp. 235-44, 1994.
- [16] C. Knabe, J. Seminatore, J. Webb, M. Hopkins, T. Furukawa and B. L. Alexander Leonessa, "Design of a series elastic humanoid for the DARPA Robotics challenge," in *2015 IEEE-RAS 15th International Conference on Humanoid Robots (Humanoids)*, Seoul, 2015.

Appendix A: LeverArmCalc

```
clear; clc;

%%%Customization%%%
%Joint selection:
%options: 'hip', 'knee', 'a_pitch', 'a_roll'
joint = 'hip';

%Motor Selection:, h = high speed, l = low
%double_h, double_l
%single_h, single_m, single_l

% Plot Setup
set(0,'DefaultTextFontname', 'Times New Roman');
set(0,'DefaultAxesFontname', 'Times New Roman');
set(0, 'DefaultTextFontSize', 11);
set(0, 'DefaultAxesFontSize', 10.5);
set(0,'defaultfigurecolor',[1 1 1]);
format long;

%%%Swatches for Customization%%%
%x,y: Motor Placement (cm)
%r_selected: selected lever arm length (cm)
tilt = 0;

switch joint
case 'hip'
    x = 32.5;
    y = 11;
    r_selected = 4.2;
    motor = 'double_h';
case 'knee'
    x = 32.5;
    y = 11;
    r_selected = 4.5;
    motor = 'double_h';
case 'a_pitch'
    x = 30;
    y = 11;
    r_selected = 6.7;
    motor = 'single_h';
case 'a_roll'
    x = 30;
    y = 12;
    r_selected = 7;
```

```

    motor = 'single_h';
end

%Parameters
%Motor Outputs [N cm/s]
switch motor
    case 'double_h'
        F_motor = 6864.4; %N
        v_motor = 32.0; %cm/s
        motor = 'MkIII High Speed';
    case 'double_l'
        F_motor = 7000; %N
        v_motor = 26.7; %cm/s
        motor = 'MkIII Low Speed';
    case 'single_h'
        F_motor = 2196.6*2; %N (since work together on ankle)
        v_motor = 50; %cm/s
        motor = 'x2 MkIII Lite High Speed';
    case 'single_m'
        F_motor = 2333.9*2; %N (since work together on ankle)
        v_motor = 47.06; %cm/s
        motor = 'x2 MkIII Lite Medium Speed';
    case 'single_l'
        F_motor = 2471.2*2; %Na(since work together on ankle)
        v_motor = 44.44; %cm/s
        motor = 'x2 MkIII Lite Low Speed';
    otherwise
        fprintf('Error: Bad Motor Choice\n')
end

%Joint Requirements (deg)
% + for right leg using RHR
switch joint
    case 'hip'
        theta_min = -100; %-30; %deg
        theta_max = 20; %130; %deg
        theta_peak = -45; %applied angle of peak torque and speed(deg)
        theta_critT = -90; %needed angle of peak torque (deg)
        theta_critW = -45; %needed angle of peak speed (deg)
    case 'knee'
        theta_min = -100; %-122; %deg
        theta_max = 5; %10; %deg
        theta_peak = -55; %deg
        theta_critT = -90; %deg
        theta_critW = -50; %deg
    case 'a_pitch'

```

```

    theta_min = -45; %deg
    theta_max = 20; %deg
    theta_peak = 0; %deg
    theta_critT = 19; %deg
    theta_critW = 10; %deg
case 'a_roll'
    theta_min = -20; %deg
    theta_max = 30; %deg
    theta_peak = 0; %deg
    theta_critT = 0; %deg
    theta_critW = 0; %deg
otherwise
    fprintf('Error: Bad Joint Choice\n')
end

%[rad/s Nm] Joint minimum requirements
switch joint
case 'hip'
    omega_min = 5.1; %rad/s
    torque_min = 200; %N*m
    joint = 'Hip Pitch';
case 'knee'
    omega_min = 7.2; %rad/s
    torque_min = 300; %N*m
    joint = 'Knee Pitch';
case 'a_pitch'
    omega_min = 5.1; %rad/s
    torque_min = 197.2; %N*m
    joint = 'Ankle Pitch';
case 'a_roll'
    omega_min = 5.9; %rad/s
    torque_min = 89.5; %N*m
    joint = 'Ankle Roll';
end

%%Constant Calculations%%
AOD = atand(y/x); %deg
c = sqrt(x^2 + y^2); %cm
u_min = -AOD;

%%Initialize arrays%%
r_min = 1; r_max = 15; %cm
r = r_min:1:r_max;
theta = theta_min:1:theta_max;
size = [length(h) length(theta)];
phi = zeros(1, length(h));

```

```

AOB = zeros(size);
l = zeros(size);
u = zeros(size);
mask = zeros(size);
beta = zeros(size);
torque = zeros(size);
omega = zeros(size);

%%Calculate Values%%
for i = 1:length(h)
    if r(i) == r_selected
        i_selected = i;
    end
    AOB_c = acosd(r(i)/c); %deg
    phi(i) = 180 - AOB_c - AOD - theta_peak; %lever arm offset (deg)
    for j = 1:length(theta)
        AOB(i,j) = 180 - AOD - theta(j) - phi(i); %(deg)
        l(i,j) = sqrt(r(i)^2 + c^2 - 2*r(i)*c*cosd(AOB(i,j))); %screw length, no deflection (cm)
        u(i,j) = asind(r(i)/l(i,j)*sind(AOB(i,j))) - AOD; %u-joint angle (deg)
        if u(i,j) < u_min
            mask(i,j) = NaN;
        else
            mask(i,j) = 1;
        end
        beta(i,j) = AOB(i,j) + (AOD + u(i,j)); %angle of application (deg)
        torque(i,j) = F_motor*sind(beta(i,j))*r(i)/100; %N/m
        omega(i,j) = v_motor*sind(beta(i,j))/r(i); %rad/s
    end
end

%Change reference of trunion angle
u = u - tilt;

%Print data for copy & Paste
fprintf('Joint: %s\n',joint)
fprintf('x: %g cm, y: %g cm\n',x, y)
fprintf('tilt: %g deg\n',tilt)
fprintf('Angle of Max Application: %g deg\n', theta_peak)
fprintf('Offset Angle: %g deg\n', phi(i_selected))
fprintf('U-Joint angle: %g/%g deg\n',min(u(i_selected,:)),max(u(i_selected,:)))
fprintf('Lever Arm Length %g cm\n', r_selected)
fprintf('Screw Length: %g cm\n', max(l(i_selected,:)) - min(l(i_selected,:)))
fprintf('Column Length: %g cm\n', min(l(i_selected,:)))
fprintf('Minimum Angle of Application %g deg\n',min(beta(i_selected,:)))

```

```

%%Plots%%
figure(1) %Heatmap
clf
subplot(2,2,1)
hold on
contourf(theta,h,torque.*mask,100,'LineColor','none')
contourf(theta,h,torque,[torque_min torque_min],'Fill','off')
contourf(theta,h,omega, [omega_min omega_min],'Fill','off','LineStyle','--')
line([theta_critT theta_critT], [r_min r_max],'Color','k','LineStyle',':')
line([theta_min theta_max], [r_selected r_selected],'Color','k','LineStyle',':')
title('Torque (N*m)')
%xlabel('Joint Angle (deg)')
%ylabel('Lever Arm Length (cm)')
%colorbar()
hold off

subplot(2,2,2)
hold on
contourf(theta,h,omega.*mask,100,'LineColor','none')
contourf(theta,h,omega, [omega_min omega_min],'Fill','off')
contourf(theta,h,torque/100,[torque_min/100 torque_min/100],'Fill','off','LineStyle','--')
plot([theta_critW theta_critW], [r_min r_max],'k:')
plot([theta_min theta_max], [r_selected r_selected],'k:')
title('Omega (rad/s)')
%xlabel('Joint Angle (deg)')
%ylabel('Lever Arm Length (cm)')
hold off

subplot(2,2,3)
hold on
[C,r] = contourf(theta,h,u);
contourf(theta,h,u, [u_min u_min],'Fill','off')
clabel(C,r)
plot([theta_min theta_max], [r_selected r_selected],'k:')
title('U-joint Angle (deg)')
%xlabel('Joint Angle (deg)')
%ylabel('Lever Arm Length (cm)')
hold off

subplot(2,2,4)
hold on
[C,r] = contourf(theta,h,l);
clabel(C,r)
plot([theta_min theta_max], [r_selected r_selected],'k:')
title('Actuator Length (cm)')
%xlabel('Joint Angle (deg)')

```

```

ylabel('Lever Arm Length (cm)')

suptitle(cat(2,joint,' ',motor))
hold off

figure(2) %At chosen r
clf
subplot(2,2,1)
hold on
plot(theta,torque(i_selected,:),'k')
plot([theta_min theta_max], [torque_min torque_max],'k:')
yl = ylim();
plot([theta_critT theta_critT], [yl(1) yl(2)],'k:')
xlabel('Joint Angle (deg)')
ylabel('Torque (N*m)')
hold off

subplot(2,2,2)
hold on
plot(theta,omega(i_selected,:),'k')
plot([theta_min theta_max], [omega_min omega_max],'k:')
yl = ylim();
plot([theta_critW theta_critW], [yl(1) yl(2)],'k:')
xlabel('Joint Angle (deg)')
ylabel('Omega (rad/s)')
hold off

subplot(2,2,3)
hold on
plot(theta,u(i_selected,:),'k')
yl = ylim();
plot([theta_critT theta_critT], [yl(1) yl(2)],'k:')
xlabel('Joint Angle (deg)')
ylabel('U-joint angle (deg)')
hold off

subplot(2,2,4)
hold on
plot(theta,l(i_selected,:),'k')
xlabel('Joint Angle (deg)')
ylabel('Actuator length (cm)')
yl = ylim();
plot([theta_critT theta_critT], [yl(1) yl(2)],'k:')

suptitle(cat(2,joint,' ',motor,' r = ',num2str(r_selected),' cm'))
hold off

```

Appendix B: GaitCycleCheck

```
%% Initialization
clear; clc;

% Plot Setup
set(0,'DefaultTextFontname', 'Times New Roman');
set(0,'DefaultAxesFontname', 'Times New Roman');
set(0, 'DefaultTextFontSize', 11);
set(0, 'DefaultAxesFontSize', 10.5);
set(0,'defaultfigurecolor',[1 1 1]);
format long;

%h/k/ap1/ap2/ar1/ar2
joint = 'ar1';
regen = 1;
dtheta = 1;

%% Load Human Reference data
load gaitalldata.mat;
M = 140; %subject mass, kg
% M_ankle      [Nm/kg]
% M_hip        [Nm/kg]
% M_knee       [Nm/kg]
% P_ankle      [W/kg]
% P_hip        [W/kg]
% P_knee       [W/kg]
% theta_ankle_deg [deg]
% theta_hip_deg [deg]
% theta_knee_deg [deg]
% thetadot_ankle_deg [deg/s]
% thetadot_hip_deg [deg/s]
% thetadot_knee_deg [deg/s]

%% Load Escher Reference data
load_log('140_kg_0.48_mps.csv')
beginning_point = 1; % use for 0.48_mps.csv
end_point = 800; % use for 0.48_mps.csv
dat = 'at 0.48 m/s';
t = timestamp(beginning_point:end_point) - timestamp(1);

%% Load Roebeck Sit to Stand data
load Roebeck_STS.mat

%% Load Atlas Reference Data
```

```

load atlast_fast_straight_leg_joint_data.mat

%% Select joint characteristics
switch joint
case 'h'
    %Motor location Geometry
    x = 325; y = 107.5; r = 60;    % mm
    phi = 55; tilt = 11;    % deg
    theta_L = [-20 95];    % deg

    % Human Reference Data
    thetaRefH = theta_hip_deg;    % deg
    tauRefH = M_hip*M;    % N/m
    omegaRefH = thetadot_hip_deg*pi/180; % rad/s after conversion

    % Escher Right leg Reference Data
    thetaRefER = q_estimate_9/pi*180; % deg after conversion
    tauRefER = tau_estimate_9;    % N/m
    omegaRefER = qdot_estimate_9;    % rad/s

    % Escher Left Leg Reference Data
    thetaRefEL = q_estimate_3/pi*180; % deg after conversion
    tauRefEL = tau_estimate_3;    % N/m
    omegaRefEL = qdot_estimate_3;    % rad/s

    %Human Stair Reference Data (requires interpolation)
    GC = 0:0.1:100; %Gait cycle percentage as in Knee, Ankle data
    SCData = xlsread('SC Hip Data');
    thetaRefSC = interp1(SCData(:,1),SCData(:,2),GC,'linear','extrap');
    omegaTemp = diff(thetaRefSC); %differentiate
    omegaRefSC = [thetaRefSC(end) - thetaRefSC(1) omegaTemp]/0.001*pi()/180;
    tauTemp = interp1(SCData(1:end-6,5),SCData(1:end-6,6),GC,'linear','extrap');
    tauRefSC = tauTemp*140*10; %Scale by mass and fudge

    %Human Sit to Stand data (requires interpolation)
    thetaRefSTS = interp1(STS_Hip_Angle(:,1),STS_Hip_Angle(:,2),GC,'linear','extrap');
    omegaTemp = interp1(STS_Hip_Velocity(:,1),STS_Hip_Velocity(:,2),GC,'linear','extrap');
    omegaRefSTS = omegaTemp/180*pi;
    tauTemp = interp1(STS_Hip_Moment(:,1),STS_Hip_Moment(:,2),GC,'linear','extrap');
    tauRefSTS = tauTemp*140;

    %Atlas Reference Data
    thetaRefAtlas = root.atlas.q_r_leg_hpy;
    omegaRefAtlas = root.atlas.qd_r_leg_hpy;
    tauRefAtlas = root.atlas.tau_r_leg_hpy;

```

```

motor = 'double_h';
joint = 'Hip';
fig = 325;
case 'k'
x = 325; y = 97.5; r = 45;
phi = 55; tilt = 11;
theta_L = [-5 100];

thetaRefH = theta_knee_deg;
tauRefH = M_knee*M;
omegaRefH = thetadot_knee_deg*pi/180;

% Notes, escher skews to extremes b/c knees always bent
thetaRefER = q_estimate_10/pi*180; % deg after conversion
tauRefER = tau_estimate_10; % N/m
omegaRefER = qdot_estimate_10; % rad/s

thetaRefEL = q_estimate_4/pi*180; % deg after conversion
tauRefEL = tau_estimate_4; % N/m
omegaRefEL = qdot_estimate_4; % rad/s

% Human Reference SC Data
SCData = xlsread('SC Knee Data');
thetaRefSC = SCData(:,2);
tauRefSC = SCData(:,7);
omegaRefSC = SCData(:,8);

%Human Sit to Stand data (requires interpolation)
GC = 0:0.1:100; %Gait cycle percentage
thetaRefSTS = interp1(STS_Knee_Angle(:,1),STS_Knee_Angle(:,2),GC,'linear','extrap');
omegaTemp =
interp1(STS_Knee_Velocity(:,1),STS_Knee_Velocity(:,2),GC,'linear','extrap');
omegaRefSTS = omegaTemp/180*pi;
tauTemp = interp1(STS_Knee_Moment(:,1),STS_Knee_Moment(:,2),GC,'linear','extrap');
tauRefSTS = tauTemp*140;

%Atlas Reference Data
thetaRefAtlas = root.atlas.q_r_leg_kny;
omegaRefAtlas = root.atlas.qd_r_leg_kny;
tauRefAtlas = root.atlas.tau_r_leg_kny;

motor = 'double_h';
joint = 'Knee';
fig = 326;

case 'ap1'

```

```

x = 315; y = 107.5; r = 70; %mm
phi = 70; tilt = 7; %deg
theta_L = [-20 45];

thetaRefH = theta_ankle_deg;
tauRefH = M_ankle*M;
omegaRefH = thetadot_ankle_deg*pi/180;

thetaRefER = q_estimate_11/pi*180; % deg after conversion
tauRefER = tau_estimate_11; % N/m
omegaRefER = qdot_estimate_11; % rad/s

thetaRefEL = q_estimate_5/pi*180; % deg after conversion
tauRefEL = tau_estimate_5; % N/m
omegaRefEL = qdot_estimate_5; % rad/s

motor = 'single_l';
joint = 'Ankle Pitch 1';
fig = 327;

% Human Reference SC Data
SCData = xlsread('SC Ankle Data');
thetaRefSC = SCData(:,2);
tauRefSC = SCData(:,7);
omegaRefSC = SCData(:,8);

%Human Sit to Stand data (requires interpolation)
GC = 0:0.1:100; %Gait cycle percentage
thetaRefSTS = interp1(STS_Ankle_Angle(:,1),STS_Ankle_Angle(:,2),GC,'linear','extrap');
omegaTemp =
interp1(STS_Ankle_Velocity(:,1),STS_Ankle_Velocity(:,2),GC,'linear','extrap');
omegaRefSTS = omegaTemp/180*pi;
tauTemp = interp1(STS_Ankle_Moment(:,1),STS_Ankle_Moment(:,2),GC,'linear','extrap');
tauRefSTS = tauTemp*140;

%Atlas Reference Data
thetaRefAtlas = root.atlas.q_r_leg_aky;
omegaRefAtlas = root.atlas.qd_r_leg_aky;
tauRefAtlas = root.atlas.tau_r_leg_aky;

case 'ap2'
x = 315; y = 107.5; r = 70;
phi = 75; tilt = 7; %83.4
theta_L = [-45 20];

thetaRefH = theta_ankle_deg;

```

```

tauRefH = M_ankle*M;
omegaRefH = thetadot_ankle_deg*pi/180;

thetaRefER = q_estimate_11/pi*180; % deg after conversion
tauRefER = tau_estimate_11; % N/m
omegaRefER = qdot_estimate_11; % rad/s

thetaRefEL = q_estimate_5/pi*180; % deg after conversion
tauRefEL = tau_estimate_5; % N/m
omegaRefEL = qdot_estimate_5; % rad/s

% Human Reference SC Data
SCData = xlsread('SC Ankle Data');
thetaRefSC = SCData(:,2);
tauRefSC = SCData(:,7);
omegaRefSC = SCData(:,8);

%Human Sit to Stand data (requires interpolation)
GC = 0:0.1:100; %Gait cycle percentage
thetaRefSTS = interp1(STS_Ankle_Angle(:,1),STS_Ankle_Angle(:,2),GC,'linear','extrap');
omegaTemp =
interp1(STS_Ankle_Velocity(:,1),STS_Ankle_Velocity(:,2),GC,'linear','extrap');
omegaRefSTS = omegaTemp/180*pi;
tauTemp = interp1(STS_Ankle_Moment(:,1),STS_Ankle_Moment(:,2),GC,'linear','extrap');
tauRefSTS = tauTemp*140;

%Atlas Reference Data
thetaRefAtlas = root.atlas.q_r_leg_aky;
omegaRefAtlas = root.atlas.qd_r_leg_aky;
tauRefAtlas = root.atlas.tau_r_leg_aky;

motor = 'single_l';
joint = 'Ankle Pitch 2';
fig = 328;

case 'ar1'
x = 315; y = 104.85; r = 107.5487;
phi = 77.138; tilt = 0;
theta_L = [-20 30];

thetaRefH = 0;
tauRefH = 0;
omegaRefH = 0;

thetaRefER = q_estimate_12/pi*180; % deg after conversion
tauRefER = tau_estimate_12; % N/m

```

```

omegaRefER = qdot_estimate_12;    % rad/s

thetaRefEL = q_estimate_6/pi*180; % deg after conversion
tauRefEL = tau_estimate_6;        % N/m
omegaRefEL = qdot_estimate_6;    % rad/s

% No human Reference SC Data
thetaRefSC = 0;
tauRefSC = 0;
omegaRefSC = 0;

%No human Sit to Stand data
thetaRefSTS = 0;
omegaRefSTS = 0;
tauRefSTS = 0;

%Atlas Reference Data
thetaRefAtlas = root.atlas.q_r_leg_akx;
omegaRefAtlas = root.atlas.qd_r_leg_akx;
tauRefAtlas = root.atlas.tau_r_leg_akx;

motor = 'single_h';
joint = 'Ankle Roll';
fig=329;

case 'ar2'
x = 315; y = 104.85; r = 105.0273;
phi = 93.33; tilt = 0;
theta_L = [-20 30];

thetaRefH = 0;
tauRefH = 0;
omegaRefH = 0;

thetaRefSC = 0;
tauRefSC = 0;
omegaRefSC = 0;

thetaRefSTS = 0;
tauRefSTS = 0;
omegaRefSTS = 0;

thetaRefER = q_estimate_12/pi*180; % deg after conversion
tauRefER = tau_estimate_12;        % N/m
omegaRefER = qdot_estimate_12;    % rad/s

```

```

thetaRefEL = q_estimate_6/pi*180; % deg after conversion
tauRefEL = tau_estimate_6; % N/m
omegaRefEL = qdot_estimate_6; % rad/s

%Atlas Reference Data
thetaRefAtlas = root.atlas.q_r_leg_akx;
omegaRefAtlas = root.atlas.qd_r_leg_akx;
tauRefAtlas = root.atlas.tau_r_leg_akx;

motor = 'single_h';
joint = 'Ankle Roll';
fig=330;

otherwise
    fprintf('Error: Bad Joint Choice\n')
    quit
end

% Select motor characteristics
switch motor
case 'double_h'
    F0 = 6864.4; %N
    v0 = 320; %mm/s
    motor = 'Mk III Heavy, High Speed';
case 'double_l'
    F0 = 7000; %N
    v0 = 267; %mm/s
    motor = 'Mk III Heavy, Low Speed';
case 'single_h'
    F0 = 2196.6; %N
    v0 = 500; %mm/s
    motor = 'Mk III Lite, High Speed';
case 'single_m'
    F0 = 2333.9; %N
    v0 = 470.6; %mm/s
    motor = 'Mk III Lite, Medium Speed';
case 'single_l'
    F0 = 2471.2; %N
    v0 = 444.4; %mm/s
    motor = 'Mk III Lite, Low Speed';
otherwise
    fprintf('Error: Bad Motor Choice\n')
end

% Constants
Y = atand(y/x);

```

```

z = sqrt(x^2 + y^2);

% Variables
theta = theta_L(1):dtheta:theta_L(2);
%syms theta
L = 180 - Y - theta - phi;
l = sqrt(r^2 + z^2 - 2*z*r.*cosd(L));

beta = acosd((-z^2 + l.^2 + r^2)./(2*l*r));
alpha = Y + L + beta - 180;
alpha = alpha - tilt; %add tilt

tau = F0.*sind(beta)*r/1000;
omega = v0.*sind(beta)/r;

%% Remove Regeneration from Reference Data
if regen == 0
    regenH = tauRefH.*omegaRefH < 0;
    regenER = tauRefER.*omegaRefER < 0;
    regenEL = tauRefEL.*omegaRefEL < 0;

    thetaRefH = thetaRefH(regenH);
    thetaRefER = thetaRefER(regenER);
    thetaRefEL = thetaRefEL(regenEL);

    tauRefH = tauRefH(regenH);
    tauRefER = tauRefER(regenER);
    tauRefEL = tauRefEL(regenEL);

    omegaRefH = omegaRefH(regenH);
    omegaRefER = omegaRefER(regenER);
    omegaRefEL = omegaRefEL(regenEL);
end

%% Fix Reference frames as necessary
alpha = -alpha;
switch joint
    case 'Hip'
        %theta = -theta;
        thetaRefER = -thetaRefER;
        thetaRefEL = -thetaRefEL;
        thetaRefSTS = -thetaRefSTS;
        thetaRefAtlas = -thetaRefAtlas;
    case 'Knee'
        theta = -theta;
        theta_L = -theta_L;

```

```

thetaRefH = -thetaRefH;
thetaRefER = -thetaRefER;
thetaRefEL = -thetaRefEL;
thetaRefSC = -thetaRefSC;
thetaRefAtlas = -thetaRefAtlas;

case 'Ankle Pitch 1'
    thetaRefER = -thetaRefER;
    thetaRefEL = -thetaRefEL;
    thetaRefSTS = -thetaRefSTS;
    thetaRefAtlas = -thetaRefAtlas;
case 'Ankle Pitch 2'
    theta = -theta;
    theta_L = -theta_L;
    %thetaRefH = -thetaRefH;
    thetaRefER = -thetaRefER;
    thetaRefEL = -thetaRefEL;
    thetaRefSTS = -thetaRefSTS;
    thetaRefAtlas = -thetaRefAtlas;
case 'Ankle Roll'
    theta = -theta;
    theta_L = -theta_L;
    thetaRefH = -thetaRefH;
    thetaRefEL = -thetaRefEL;
end

%% Post Process
alpha_1 = [min(alpha) max(alpha)];
beta_1 = [min(beta) max(beta)];
l_1 = [min(l) max(l)];
L1 = 57.1525;%65.2; %ujoint to start of axle
L2 = 20; %distance from pivot to end of axle (conservative)
L3 = 40 + 5 + 48 + 5 + 10; %unusable length
L_shaft = l_1(1) - L1 - L2;
L_motion = l_1(2) - l_1(1);
L_thread = L_shaft - L3;

%% Report Results
clc
fprintf('Settings: \n')
fprintf([' Joint: ', joint,'\n'])
fprintf([' Motor: ', motor,'\n'])
fprintf(' x = %.1f mm, y = %.1f mm, r = %.1f mm\n',x,y,r);
fprintf(' tilt = %.1f deg, phi = %.1f deg\n',tilt,phi);
fprintf('\nTotal Length: %.1f to %.1f mm\n',l_1(1), l_1(2))
fprintf(' Max Shaft Length: %.2f mm\n',L_shaft)

```

```

fprintf(' Motion Needed: %.2f mm\n',L_motion)
fprintf(' Available Thread: %.2f mm\n',L_thread)
fprintf('\nU-joint angle range: %.2f to %.2f deg\n', alpha_l(1),alpha_l(2))
fprintf('Application angle range: %.2f to %.2f deg\n',beta_l(1),beta_l(2))

%% Plots
figure(fig); clf;

subplot(2,2,1)
hold on
plot(theta,tau,'k')
plot(thetaRefER,abs(tauRefER),'r.')
%plot(thetaRefEL,abs(tauRefEL),'g.')
plot(thetaRefH,abs(tauRefH),'b.')
plot(thetaRefSC,abs(tauRefSC),'m.')
plot(thetaRefSTS,abs(tauRefSTS),'c.')
plot(thetaRefAtlas/pi*180,abs(tauRefAtlas),'g.')
ylabel('Torque (N-m)')
xlabel('Joint Angle (deg)')
legend1 = legend('Actuator
Output','Escher','Flat','Stairs','STS','Atlas','location','best','orientation','horizontal');
set(legend1,...
    'Position',[0.0815408843089566 0.471603317320096 0.835931688038261
0.0360219255264794],...
    'Orientation','horizontal',...
    'FontSize',12);

subplot(2,2,2)
hold on
plot(theta,omega,'k')
plot(thetaRefER,abs(omegaRefER),'r.')
%plot(thetaRefEL,abs(omegaRefEL),'g.')
plot(thetaRefH,abs(omegaRefH),'b.')
plot(thetaRefSC,abs(omegaRefSC),'m.')
plot(thetaRefSTS,abs(omegaRefSTS),'c.')
plot(thetaRefAtlas/pi*180,abs(omegaRefAtlas),'g.')
yl = ylim;
plot([theta_L(1) theta_L(1)], yl,'k:')
plot([theta_L(2) theta_L(2)], yl,'k:')
ylabel('Radial Velocity (rad/s)')
xlabel('Joint Angle (deg)')

subplot(2,2,3)
plot(theta,l,'k')
ylabel('Actuator Length (mm)')
xlabel('Joint Angle (deg)')

```

```
subplot(2,2,4)
plot(theta,alpha,'k')
ylabel('U-joint Angle (deg)')
xlabel('Joint Angle (deg)')

axishandles = findall(0,'type','axes');
for ichild = 1:length(axishandles)
    axishandle = axishandles(ichild);
    set(findobj(axishandle,'Type','line'),'LineWidth',1,'MarkerSize',5)
    set(axishandle,'FontSize',14,'LineWidth',1.5)
    set(get(axishandle,'Xlabel'),'FontSize',14)
    set(get(axishandle,'Ylabel'),'FontSize',14)
    set(get(axishandle,'Title'),'FontSize',14)
end
```