

**MULTI-OBJECTIVE USER TUNABLE INTERFACE FOR  
ASSISTANCE CONTROL OF A LOWER LIMB  
EXOSKELETON - A STEP IN THE RIGHT DIRECTION**

by

Kurt Stewart

Bachelor of Science in Electrical Engineering, UNB, 2017

A Thesis Submitted in Partial Fulfillment  
of the Requirements for the Degree of

**Master of Science in Electrical Engineering**

in the Graduate Academic Unit of Electrical and Computer Engineering

Supervisors: Jonathon Sensinger, PhD, Electrical and Computer Engineering

Chris Diduch, PhD, Electrical and Computer Engineering

Examining Board: Julian Meng, PhD, ECE, Chair

Kevin Englehart, PhD, ECE

Howard Li, PhD, ECE

Chris McGibbon, PhD, Kinesiology

This thesis is accepted by the  
Dean of Graduate Studies

THE UNIVERSITY OF NEW BRUNSWICK

March, 2020

©Kurt Stewart, 2020

## **Abstract**

The field of assistive lower limb exoskeletons lacks controllers that allow user adjustment according to their needs and desires. This thesis develops and shows simulation evidence for allowing user-intuitive control by providing adjustment of gait based on gait performance measures the user cares about while walking. Using the NSGA-II multi-objective optimization algorithm to generate trajectories for a virtual constraint-based exoskeleton system a lookup table was generated which provides user adjustment of speed, comfort, effort proxy, and natural walking measures. The findings in this thesis demonstrate a variety of gait across these performance measures which can be used to formulate a user-adjustable controller.

## **Dedication**

I would like to dedicate my thesis to my parents Neil and Margaret Stewart. Thank you for all your support over the years.

## **Acknowledgements**

I would like to acknowledge God first of all, without his direction, and help I would not be where I am today. I would like to acknowledge my parents for letting me pursue a Master's Degree and letting me stay at their house for a few more years. I would like to acknowledge Jonathon Sensinger for being a great supervisor and friend throughout my degree, I would like to acknowledge Chris Diduch for trying to keep me on track in my research timeline. I would like to especially acknowledge Samuel Campbell whose friendship, engineering experience and help during my thesis have been invaluable to my success. I would like to acknowledge Zahed Dastan, who revitalized me in the last months of my time at UNB. I would like to acknowledge the IBME family overall and their support. I would like to acknowledge the members of the Fredericton Gospel Hall who have been a family and support system to me for many years. I would like to acknowledge my close friends Nathaniel Despres, Amos Chiasson, Mitchel Frith, Joshua Smith, and Morgan Macfarlane who have been brothers to me supporting me as I grow as a man and pursue the pathway before my feet.

# Table of Contents

Abstract.....	ii
Dedication.....	iii
Acknowledgements.....	iv
Table of Contents.....	v
List of Tables.....	ix
List of Figures.....	x
List of Symbols.....	xii
1 Introduction.....	1
1.1 Specific Aims.....	2
2 Justification and Feasibility.....	4
2.1 Assistive Exoskeletons.....	4
2.2 Clinical Perspective – Real Challenges.....	5
2.3 Providing Gait.....	5
2.4 Current Control Methodologies.....	6
2.4.1 Zero Moment Point Control Drawbacks.....	10
2.4.2 Virtual Constraint Control.....	10
2.5 Simulation Modelling and Control.....	11
2.5.1 Dynamic Stability Analysis for Autonomous Bipedal Walking.....	12
2.6 Optimization Formulation.....	17
2.6.1 Optimization Techniques.....	18
2.7 Forays in Global Optimization.....	19
2.8 Multi-objective Optimization.....	20

3	Methods.....	22
3.1	Exoskeleton Simulation Model .....	25
3.1.1	Virtual Constraints .....	27
3.1.2	Gait Phase Division of Dynamics.....	28
3.1.3	Gain Selection.....	34
3.2	Performance Measures .....	34
3.2.1	Biomimicry Cost.....	35
3.2.2	Effort Proxy Cost .....	36
3.2.3	Speed Cost .....	36
3.2.4	Comfort.....	37
3.2.5	Multi-objective Formulation.....	37
3.3	Optimization Constraints.....	38
3.4	Non-dominated Sorting Genetic Algorithm II .....	39
3.4.1	Classical Genetic Algorithm Operations .....	40
3.4.2	Non-dominated Sorting.....	41
3.4.3	Crowding Distance Sorting.....	42
3.5	Optimization Problem Formulation .....	43
3.6	Algorithmic Parameters.....	45
4	Results.....	49
4.1	User Interface .....	49

4.2	Range of Outputs.....	52
4.2.1	Speed Range.....	52
4.2.2	Comfort Range.....	53
4.2.3	Effort Proxy Cost Range.....	53
4.2.4	Natural Walking Range.....	53
4.3	Optimization Sensitivity.....	53
4.4	Prioritizing Parameters.....	54
4.4.1	Prioritizing Comfort.....	54
4.4.2	Prioritizing Speed.....	55
4.4.3	Prioritizing Effort Proxy.....	56
4.4.4	Prioritizing Natural Walking.....	57
5	Discussion.....	59
5.1	Range of Outputs.....	59
5.2	Sensitivity.....	60
5.3	Prioritizing Performance Measures.....	60
5.4	Tracking Error.....	61
6	Conclusion.....	63
6.1	Limitations.....	63
6.2	Future Work.....	64
6.2.1	Algorithms and Optimization.....	64

6.2.2	Real-world Implementation .....	65
6.2.3	Control Exploration .....	66
6.3	Final Remarks .....	66
	Bibliography .....	68
	Appendix A: Source Code .....	77
	Curriculum Vitae	

## List of Tables

Table 1: Clinical Exoskeletons .....	7
Table 2: Exoskeleton Model Mechanical Component Values.....	26
Table 3: PD Gains.....	34
Table 4: NSGA-II Algorithmic Parameters .....	48
Table 5: Spread and Sensitivity of Look-up Table .....	52

# List of Figures

Figure 1: Exoskeleton Classifications.....	4
Figure 2: Phases of Gait [22] .....	6
Figure 3: Admittance Control Loop.....	9
Figure 4: Impedance Control Loop.....	9
Figure 5: Van der pol Phase Portrait.....	14
Figure 6: Van der pol Poincaré Map.....	14
Figure 7: Chaotic Van der pol Oscillator Phase Portrait.....	16
Figure 8: Chaotic Van der pol Poincaré Map .....	16
Figure 9: Domination and Pareto Front .....	21
Figure 10: Research Framework.....	22
Figure 11: Phase 1 Controller Overview .....	24
Figure 12: Phase 2 User Interface Controller .....	25
Figure 13: Simulation Model.....	26
Figure 14: Phase Variable Measure .....	28
Figure 15: Swing Phase .....	30
Figure 16: Double Support Phase .....	31
Figure 17: Underactuated Phase .....	33
Figure 18: NSGA-II Algorithm .....	40
Figure 19: Crowding Distance Measure [63].....	42
Figure 20: Pareto Sorting NSGA-II [63] .....	43
Figure 21: Optimization Inputs.....	44

Figure 22: Optimization Flow.....	45
Figure 23: User Interface .....	50
Figure 24: Reference Lookup Table Method.....	51
Figure 25: Prioritizing Comfort Output Gait Trajectories .....	55
Figure 26: Prioritizing Speed Output Trajectories.....	56
Figure 27: Prioritizing Effort Proxy Output Trajectories .....	57
Figure 28: Prioritizing Natural Walking Output Trajectories.....	58

## List of Symbols

$A$  – Forcing Function Amplitude

ADL – Activities of Daily Living

$\alpha$  – Angular Acceleration of Stance Toe

$\mathbf{B}$  – Inertial Matrix

$\mathbf{C}$  – Coriolis-gravitational Matrix

$\eta$  – Number of Joint Links in Bipedal Model

$f$  - Constraint Equation

$i$  – Joint Index

$J$  – Cost Function

$K$  – Kinetic Energy

$k$  – Spring Constant of Ground

$L$  – Lagrangian

$l$  – Link Length

$\lambda_f$  – Constraint Force

$\mathbf{M}$  – Moment Matrix

MOEA/D – Multi-objective Evolutionary Algorithm based on Decomposition

MS – Multiple Sclerosis

$N$  – Number of Individuals in Population for NSGA-II

$n$  – Number of Cost Functions

$n_p$  – Number of Dominated Individuals

NSGA-II – Non-dominated Sorting Genetic Algorithm

$\omega$  – Forcing Function Frequency

$P$  – Potential Energy

$\mathbf{P}_t$  – Total NSGA-II Population

$\mathbf{P}_n$  – Individual ins NSGA-II Population

PD – Proportional-Derivative

$\phi$  – Stance Joint Angle Throughout One Gait Cycle

$\mathbf{Q}_t$  – NSGA-II Parent Population

R – Radius of Stable System

r – Radius of Initial Conditions for Stability

SCI – Spinal Cord Injury

S – Step Length

*S.F.* – Scaling Factor of the Deadzone Size

SMS-EMOA - S-Metric Selection Evolutionary Multi-Objective Algorithm

$\mathbf{S}_p$  – Set of Dominated Individuals

$Speed_{Actual}$  – Averaged Speed of Exoskeleton over one Simulation

$Speed_{Desired}$  – Speed Desired at Input of Optimization

$\mathbf{T}$  – Joint Torque Matrix

$t$  – Time

$\theta$  – Joint Angles

$\Theta_I$  – Input Joint Angle Reference Trajectories

$\Theta_W$  – Winter's Gait Joint Angles

$\mu$  – Van der pol oscillator damping term

$v_{CoMx}$  – Horizontal Center of Mass Velocity

$v_{min}$  – Minimum Velocity Allowed for Stable Walking

$W$  – User Weight

$x$  – State Value

$\mathbf{x}_c$  – State Values at Cross-section

ZMP – Zero Moment Point

# 1 Introduction

Exoskeleton technology has made great advances in helping individuals with disabilities in regaining mobility. In Canada, 1 in 442 people live with spinal cord injuries (SCI), while 1 in 385 people are living with Multiple Sclerosis (MS), and disability rates are constantly increasing [1], [2]. These disabilities often restrict people in their activities of daily living (ADL) and decrease their quality of life. Various diseases and disabilities inhibit people from being able to walk, which limits many tasks that most take for granted. Although mobility is not the primary ADL this population desires to be restored [3], it has been shown that mobility improves bodily functions and allows for increased overall health [4]. There is an opportunity for lower limb exoskeletons to be improved and have an increasing impact on the lives of people with disabilities through providing improved ambulation.

In the last decade, the development of actuators, battery technology, and high-speed microcomputers has led to rapid growth in mobile robotics and exoskeletons [5]–[7]. Although conventional physiotherapy can help in restoring ambulation to some individuals, exoskeletons have a greater ability to provide ambulation to a wider population [8]. Current exoskeleton hardware and control technology have been shown to provide successful gait for individuals with SCI, stroke, MS as well as some other conditions. The current methods, however, are limited, allowing little adjustment, in how they fit individual users' needs and desires. Clinicians and users have reported this drawback as a key problem to remedy. Although some optimize for one or two parameters, they do not allow for user

intuitive customization [9], [10]. There are no assistive controllers that allow a user to select a walking trajectory based on how they value different gait performance objectives.

Current exoskeletons are rigid, not allowing adjustment of the multiple factors users care about while using these devices to best suit an individual. Individuals' needs vary across pathologies and injuries; requiring differing gait strategies. The current rigidity needs to be addressed to allow assistive exoskeletons to better suit the needs and desires of individuals. The specific aim of this thesis is to make progress in solving this gap in exoskeleton technology by providing user input on multiple gait performance measures.

This thesis has the aim of providing a user interface lower limb exoskeleton controller based on gait performance measures users care about to solve the current limitations of exoskeletons. Providing an intuitive user-tunable controller allows exoskeletons to be better suited to a specific user's desires. Such a user-tunable interface is a potential solution to this concern. The verification of this control framework will be evaluated through the following specific aims.

## **1.1 Specific Aims**

**Specific Aim 1:** To generate a variety of stable control reference trajectories that produce human-like gait for an exoskeleton. This objective was attained through the generation of trajectories using a multi-objective global optimization across various performance measures using an exoskeleton simulation model.

**Specific Aim 2:** To develop a trajectory lookup table that provides meaningful user tuning based on pertinent user performance measures. This specific aim was achieved by precomputing the trajectories using open-loop optimization while accounting for the

factors of speed, comfort, effort proxy, and biomimicry and reformulating into a user interface.

**Innovation.** Using multi-objective optimization to generate gait joint angle trajectories for an exoskeletal-bipedal system. Observing that a feedback linearized bipedal system allows for flexible control of inputs while achieving stable walking.

**Significance.** The field of assistive exoskeletons is currently deficient in providing user adjustment according to different preferences. Improving the user control of an exoskeleton in an intuitive manner will allow for more user input with the devices and hopefully, increase the acceptance, user satisfaction and long-term use of exoskeleton devices. The opportunity for users and clinicians to use the same exoskeleton and tune it to personal needs may enhance the number of individuals who use these assistive devices.

## 2 Justification and Feasibility

### 2.1 Assistive Exoskeletons

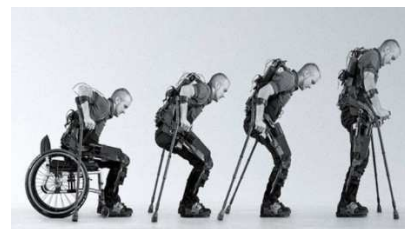
An exoskeleton is a robotic device that aids, improves, or augments human motor tasks. There are three major classifications of exoskeletons: human-augmenting (Figure 1.a), rehabilitative (Figure 1.c), and assistive (Figure 1.b). A human-augmenting exoskeleton improves the power output of a human-controlled action or allows the individual to perform a task repeatedly without strain. Human-augmenting exoskeletons are of interest in the military and industrial applications. The other two classifications, rehabilitative and assistive, are of interest in the medical field. A rehabilitative exoskeleton is such that when an individual uses the device, they become more proficient at the task after using the device. The assistive classification is given to exoskeletons which have the goal of performing a task in an optimal manner for its user. This thesis is focused on the development of an assistive exoskeleton controller and will thus focus on improving current assistance strategies.



1.a) Lockheed Martin [11]



1.b) Cyberdine HAL



1.c) Ekso Bionics [13]

[12]

**Figure 1: Exoskeleton Classifications**

## **2.2 Clinical Perspective – Real Challenges**

Research in the field of medical exoskeletons is new and many issues still exist in making these devices effective for users [14]. Clinicians and exoskeleton users have difficulty adjusting the parameters to suit users' needs and desires. Users have reported being frustrated with the walking speed limitations of current exoskeletons and the inability to adjust the speed at which they wish to walk. There is a reported need for more biomimetic walking in exoskeletons [15]. Interaction forces between the user and the exoskeleton can also be problematic [16]. Energy consumed while walking is as well a key component to improve for user long term acceptance [17]. There is some flexibility in adjusting parameters of current controllers, such as the low-level controller gains to influence the gait outcomes, however, these do not necessarily give intuitive control to the user [18]. Adjustment of an effort proxy, comfort, natural walking and speed parameters to a specific user is a problem that requires an intuitive and effective solution [19]–[21].

## **2.3 Providing Gait**

To provide human-like gait using an exoskeleton, healthy gait should be observed. For a healthy individual, level ground walking is primarily a cyclical motion. A single gait cycle is defined from initial contact to the next initial contact of the same foot. The biomechanical system of the leg, composed of the foot, shank, thigh, and hip attached to the torso for each leg, is a nonlinear coupling of mechanical elements. In healthy individuals, this system is controlled by the motor cortex of the brain that excites neural signals transmitted to the responsible muscle groups. When the corresponding movements are coordinated in a particular manner gait is produced. Healthy gait can be divided up into

different phases, the common phasic division of gait can be seen in Figure 2. It is the desired outcome of a lower limb exoskeleton to aid or produce bipedal gait similar to healthy gait.

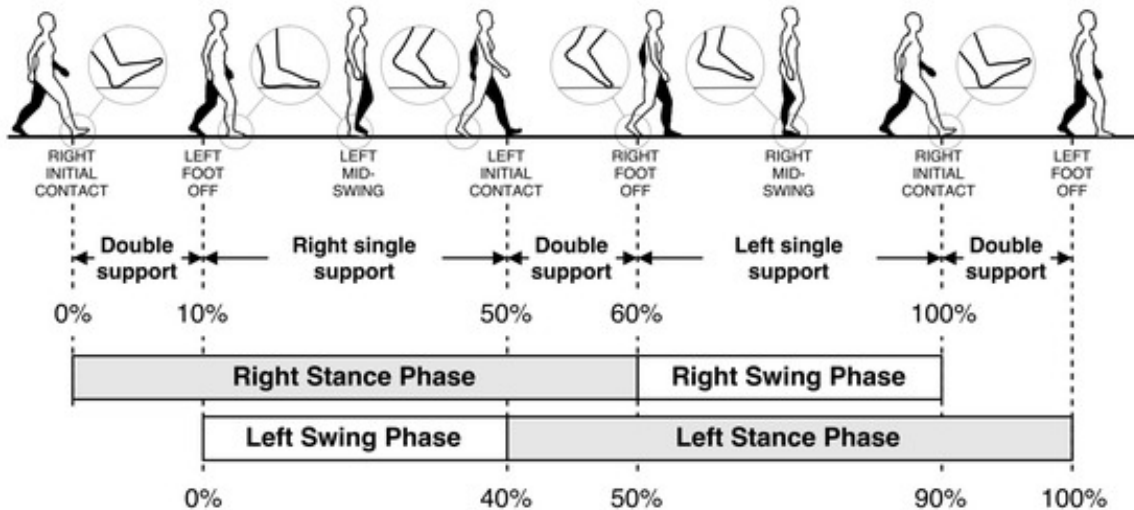


Figure 2: Phases of Gait [22]

## 2.4 Current Control Methodologies

To produce gait for a human the exoskeleton needs a system controller. Choosing a controller that produces stable gait is an ongoing research topic. Various control strategies have been implemented on exoskeletons including reference trajectory enforcing, adaptive oscillators, model-based control, fuzzy controllers, sliding mode control, adaptive control, neural networks, and feedback linearization [7], [23]–[26]. These exoskeleton control strategies have shown to produce stable gait, especially for individuals with SCI; however, they do not inherently solve the issues which have been found in clinical practice.

Clinical testing and federal approval for exoskeletons is a long process. The Ekso, Indego, and ReWalk are three clinical exoskeletons that are on the market internationally.

Besides these devices, there are other lower limb exoskeletons that are under development by companies and research groups around the world. A drawback of the current clinical exoskeletons which provide ambulation for individuals with SCI is not providing actuation at the ankle. Table 1 summarizes the control strategies, whether the ankle is actuated, and the clinical classification of the most prominent exoskeletons in the clinical field, some of which are in development or undergoing approval.

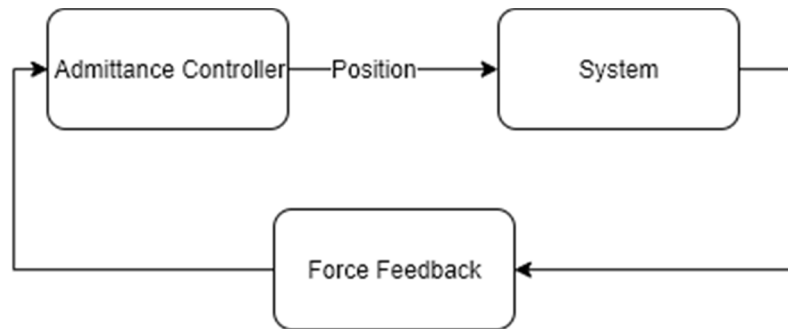
**Table 1: Clinical Exoskeletons**

	Low-level Control	High-Level Control	Actuated Ankle (Yes/No)	Primary Clinical Classification
ReWalk [27]	Impedance	Trajectory following	No - rigid	Assistive
Mina [28]	Admittance	Trajectory following	No - rigid	Assistive
Indego [29]	Impedance	Human initiated feedforward assistance	No – free to move	Rehabilitative
Ekso	Impedance	Assist-as-needed	Rigid ankle	Rehabilitative

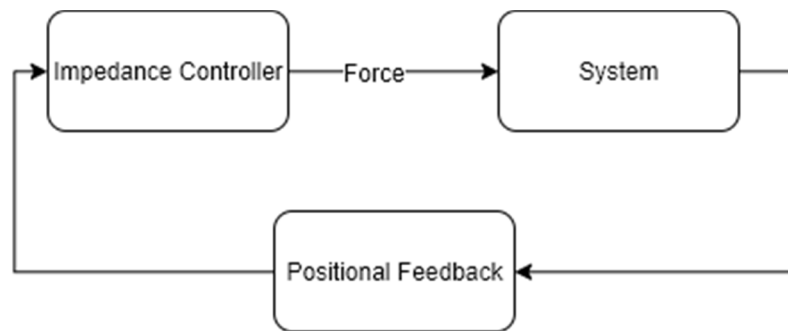
Rex	Unknown	User joystick	Yes	Assistive (most likely)
Lokomat [30], [31]	Admittance and Impedance	Gait pattern adaptation	Passive actuation (spring)	Rehabilitative
Lopes [32]	Impedance	Trajectory following	No – free to move	Rehabilitative
HAL [30], [33]	Admittance	Electromyography on legs	No – passive support	Assistive

The distinct classification between rehabilitative and assistive exoskeletons is a recent and important construct. Some control strategies have been misclassified and there is a good deal of overlap in the field. Devices can be used for both purposes, and some have software that provides both rehabilitative and assistive modes. Although assistive devices can be used in a rehabilitation sense, they do not promote rehabilitative best practices [34]. Strict trajectory following exoskeletons have shown to reduce patient physical effort [35] which is not ideal for rehabilitation as it promotes user slacking. This characteristic, however, is a potentially good feature for an assistive exoskeleton where reducing the user's energetic input would be preferable for some users. The proposed control strategy developed is focused on assistance goals rather than rehabilitation goals.

Following a reference joint-angle trajectory has been established as a reliable method of providing assistive gait control. This method has been employed with both impedance and admittance low-level control. Admittance control, Figure 3, uses a force as an input and outputs a position. Impedance control, Figure 4, takes a positional input and outputs a force. Most assistive exoskeletons employ impedance control by applying torques at the joints to follow preset joint angle reference trajectories. Following these trajectories using joint level controllers has been shown to provide stable walking [36]–[38].



**Figure 3: Admittance Control Loop**



**Figure 4: Impedance Control Loop**

Impedance is the preferred method used for most exoskeletons because of its compliance properties. Compliance control has been shown to be safer for human-machine interaction as humans are compliant and not rigid mechanisms [39]. Providing lower limb

impedance control also allows for varied impedance throughout the gait cycle which is a phenomenon that humans also perform while walking.

#### **2.4.1 Zero Moment Point Control Drawbacks**

Zero moment point (ZMP) control is the control framework that has been used to ensure the stability of many assistive exoskeletons but is quite limiting in what it can achieve [7]. This method is appropriate when having rigid ankles as it implies flat-footed walking. Flat-footed walking with ZMP control requires the system dynamics to be zeroed every step, thus eliminating the underactuated phase of gait. Each step a flat-footed walker takes requires acceleration to and from zero velocity. This method increases the energy required to produce gait as it does not allow for conserving momentum from step to step. Restricting the ankle modifies the topology of natural gait and requires excessive recruitment of the hip and knee to provide ambulation. Considering that both effort and natural walking are performance measures that users wish to optimize, actuated ankles are desirable, making the ZMP control system inappropriate in this case. A dynamic stability analysis method, such as virtual constraint control, is a better alternative compared to ZMP.

#### **2.4.2 Virtual Constraint Control**

A bipedal exoskeleton is a nonlinear dynamic system and requires nonlinear control methods to produce gait. Although the field of linear control is a well-established field, nonlinear control is an area of ongoing research especially in the field of legged robotics. One nonlinear method which has been shown to be effective in producing stable control is a feedback linearization technique known as virtual holonomic constraint control. A holonomic, or position-based, virtual constraint has been used to linearize an otherwise

nonlinear system such that linear control techniques can be used to achieve system stability [40]–[42].

Following a time-dependent trajectory is a non-autonomous strategy for producing gait and has inherent intolerance to perturbations. Proving the stability of non-autonomous systems is also difficult [43]. Virtual constraint control allows for a time-invariant mapping of gait trajectory control. This was used by Grizzle et al. [42] for the bipedal robot Mabel and has been used in a lower-limb prosthesis [44] and recently in an exoskeleton [23]. The method takes a monotonic increasing variable across a limit cycle, called a phase variable, and remaps all system outputs based on this single input. The one-to-one mapping of the phase variable to the system states provides a time-invariant feedback linearization of the nonlinear bipedal system [45]. Existing virtual constraint controllers have used the center of pressure [45] as a phase variable to provide stable control of bipedal robots. A feedback linearized system can be asymptotically controlled using linear joint-level controllers [43]. The autonomous feedback linearized control provides tolerance to perturbations in gait.

## **2.5 Simulation Modelling and Control**

Formulation and verification for this thesis were done using a dynamic simulation. To control a system using reference trajectory control the inverse dynamics needed to be derived. To model, the inverse dynamics of a bipedal system Newtonian mechanics and Lagrangian mechanics are both appropriate choices. Lagrangian mechanics are preferred in the field of robotics due to coordinate-free formulation [46]. The inverse dynamics of a system give the torques,  $\mathbf{T}$ , required to move the system in space and can be derived using the Lagrange dynamics formula,

$$\mathbf{T} = \frac{d}{dt} \frac{\partial \mathbf{L}}{\partial \dot{\boldsymbol{\theta}}} - \frac{\partial \mathbf{L}}{\partial \boldsymbol{\theta}} \quad (1)$$

Where  $\mathbf{L}$  is the Lagrangian,

$$\mathbf{L}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) = \mathbf{K}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) - \mathbf{P}(\boldsymbol{\theta}) \quad (2)$$

Where  $\mathbf{K}$  is the kinetic energy,  $\mathbf{P}$  is the potential energy of the whole body and  $\boldsymbol{\theta}$  are the positional variables of the system. The dynamic model for simulation was derived using Lagrangian methods.

In addition to being nonlinear, gait is a hybrid cycle with changing dynamics. There is a continuous phase and a discrete phase. The swing and stance phases are continuous, whereas each time the swing foot contacts the ground, initial contact, is best described as a discrete event. A hybrid system is best controlled with a hybrid controller that manages the differing dynamics in the continuous and discrete phases [47]. The hybridity and changing dynamics of gait were handled using different dynamic formulations over the gait cycle.

### **2.5.1 Dynamic Stability Analysis for Autonomous Bipedal Walking**

Ensuring an exoskeleton performs stable walking with a user is a primary safety and performance factor. Phase portraits, although a powerful nonlinear tool, do not provide stability conclusions with respect to bipedal walking as this method does not work on high order systems. Various other methods have been used to analyze dynamic bipedal walking including, maximum Lyapunov exponents, Poincaré return maps, long-range correlations, extrapolated center of mass, and variability measures [48], [49]. For feedback linearized systems, the stability of the zero dynamics establishes the stability of the system. For a

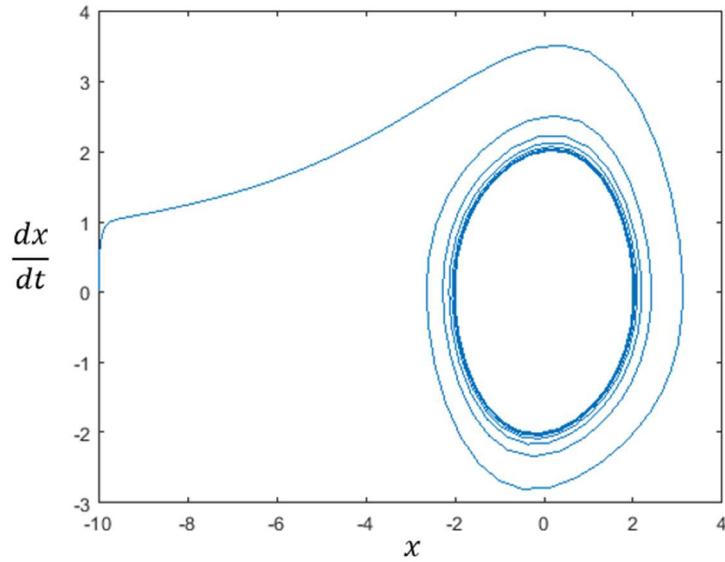
hybrid feedback-linearized bipedal system [47], the stability can be established through analyzing the hybrid zero dynamics. There are different stability analysis methods although no best-practice stability method has been determined to best ensure bipedal gait.

Poincaré mapping is a method that is commonly used in gait analysis as it is able to be extrapolated to determine the stability of a high order limit cycle. A Poincaré return map is formulated by evaluating one state at a cross-section perpendicular to the state-flow over multiple cycle iterations; in the case of walking across multiple steps. The value of the state evaluated at the cross-sections is recorded in a vector of numbers,  $\mathbf{x}_C$ . A return map is shown by plotting this vector against itself shifted by one,  $\mathbf{x}_{C-1}$ , such that each value is plotted against the previous state value. The return map thus shows the convergence of the system. If the Poincaré return map reaches the  $\mathbf{x}_C = \mathbf{x}_{C-1}$  line, there is a limit cycle. If it approaches the line with a derivative of magnitude less than 1, this limit cycle is stable [49], [50].

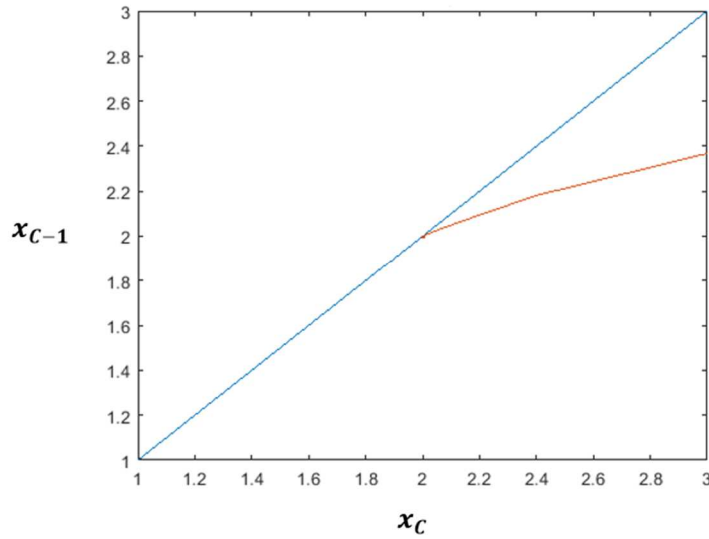
The example of a stable Van der pol oscillator shows the stability of the limit cycle. The Van der pol of (3) of state  $x$ , with damping term of  $\mu=1$ , has a stable limit cycle as seen in the phase portrait of Figure 5.

$$\frac{d^2x}{dt^2} - \mu(1 - x^2) \frac{dx}{dt} + x = 0 \quad (3)$$

The Poincaré return map for this stable system is shown in Figure 6 where it approaches the  $\mathbf{x}_C = \mathbf{x}_{C-1}$  with a slope of magnitude less than 1.



**Figure 5: Van der pol Phase Portrait**



**Figure 6: Van der pol Poincaré Map**

Poincaré stability conclusions require full state information of a system. Each joint of a biped needs to be controlled in a stable manner to produce stable gait. A Floquet multiplier matrix can be used to evaluate the Poincaré method on a system with high order.

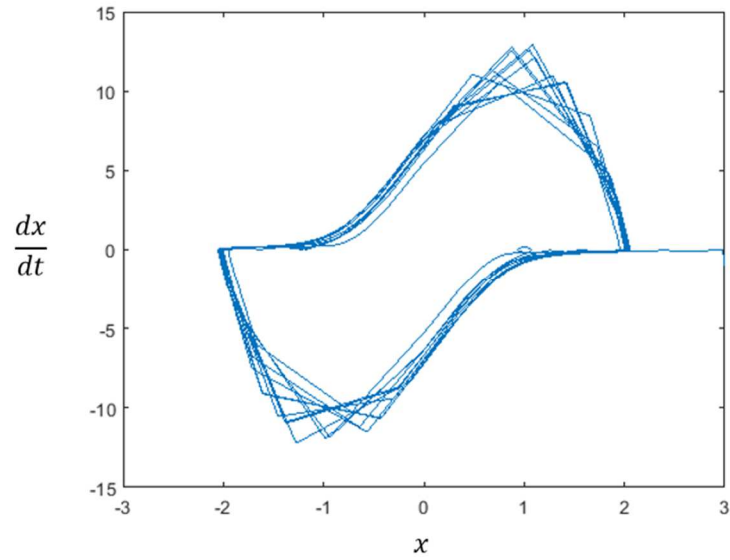
This method determines the derivative of the states with respect to the limit cycle. If all the eigenvalues of the Floquet matrix have a magnitude less than 1, all states have stable limit cycles and full stability conclusions can be made.

Bipedal walking can be approximated as a periodic system, such as a limit cycle. Although, when a bipedal walker approaches a limit cycle it can have some chaotic behaviour [51]. A chaotic system can still be stable if it is bounded. Thus, if a bipedal system is bounded around its limit cycle it can be designated stable as per Lyapunov's definition.

“A system is stable within small radius  $r$  if when it starts in  $r$  it remains within bigger radius  $R$ ” [43]

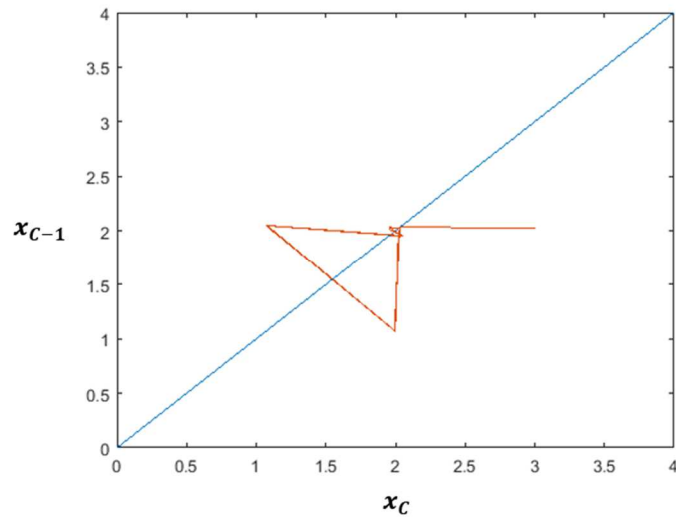
Using a bounded approach, if the system does not go outside of the radius of  $R$  when it has approached its limit cycle, the gait is stable. This radius can be termed as the chaotic margin of stability in the system. A stable chaotic Van der pol can be used to show the boundedness of chaotic behaviour. The state behaviour of a chaotic Van der pol described by (4), with damping term,  $\mu=8.53$ , sinusoidal forcing function with amplitude,  $A=1.2$ , and frequency,  $\omega=\pi/5$ , can be seen in the phase portrait of Figure 7.

$$\frac{d^2x}{dt^2} - \mu(1 - x^2)\frac{dx}{dt} + x - A\sin(\omega t) = 0 \quad (4)$$



**Figure 7: Chaotic Van der pol Oscillator Phase Portrait**

This chaotic Van der pol oscillator approaches a limit cycle, however, when it is near the limit cycle it stays bounded within a radius from the limit cycle. This bounded behaviour can be seen in Figure 8.



**Figure 8: Chaotic Van der pol Poincaré Map**

## 2.6 Optimization Formulation

To provide a user-intuitive controller it is best to give control based on what they want. Humans perform motor tasks, such as gait, in an optimal manner. Optimal control has been shown to be the method of choice for providing human-machine synthesis [52]. To provide intuitive control the user should adjust the values of different gait performance measures they care about. Users value the performance measures of stability, energy consumption, comfort, natural walking, and speed when achieving gait [16], [17], [21]. Control strategies of bipedal control have focused on improving one, or two of these performance criteria. For example, Grizzle et al. [53] have optimized their control of a biped walker using the parameters of stability and energy while using a virtual constraints controller. Collins [10] has adapted human-exoskeleton gait to optimize the energy efficiency of gait over time. An optimal control strategy that uses relative weightings of each of these factors allows for a user to adjust the device according to their desire. Such a controller has not yet been developed. Formulating an offline optimal control problem with cost functions for effort proxy, comfort, natural walking, and speed performance measures provide users with adjustment of the things they care about while walking.

Multiple input formulation of the optimization problem requires the definition of each individual cost and a method of combination. Convex problems are the preferred type for an optimization problem. Convexity allows for the local minima to be the global minima. The field of computational motor control has determined that humans care about most things in a quadratic sense, which provides convexity. Combining multiple objectives, even when each is convex, is not necessarily a convex problem, but some convexity properties

can help [54]. Providing multiple inputs to an optimization problem can be done by combining multiple costs into one single objective or a multi-objective formulation.

To optimize an impedance controlled bipedal system the low-level control parameters, or the reference trajectories could be optimized. The stability of a feedback-linearized system is determined by the gains of the controllers. Optimizing the reference trajectories allows for more change in topology and requires less stability tuning.

An optimal control problem can be formulated in an open-loop manner or a closed-loop manner. In the open-loop case, the optimization would be performed over each step of the exoskeleton, whereas the closed-loop optimization calculates the optimal control input at each time-step throughout the motion. It is desirable that a user is able to walk immediately upon donning an exoskeleton thus the controller must be optimized to produce gait offline. Using a Poincaré stability method, that requires a full evaluation of the limit-cycle, stipulates an offline open-loop optimal control of the gait reference trajectories.

### **2.6.1 Optimization Techniques**

The generation of curves for the bipedal system has been accomplished with global optimization techniques. Mapping of the reference trajectory optimization inputs to the cost functions is a nonlinear mapping due to the non-linear dynamics, thus a global optimization technique which has been shown to solve nonlinear problems is required. Global optimization techniques that have been shown to be capable of handling gait trajectory generation include Genetic Algorithms [55], Sequential Quadratic Programming [56], Genetic Programming [57], Particle Swarm Optimization [58], Differential Evolution and Univariate Dynamic Encoding Algorithms for Searches [57].

## 2.7 Forays in Global Optimization

During the course of the thesis, steps were added to the optimization problem formulation as well as different optimization strategies attempted to generate the look-up table of gait trajectories. The cost function that was used was originally formulated as a single objective,  $J_T$ , that is a sum-of-weighted-squares of  $n$  performance measure cost functions in the form of,

$$J_T = W_1 J_1 + W_2 J_2 \dots + W_n J_n \quad (5)$$

Where  $W_n$  is the weighting factor for the respective individual cost function  $J_n$ . This total cost function,  $J_T$ , was implemented to evaluate the feasibility of the method that included a quadratic biomimicry cost function and a quadratic effort proxy cost function. The methods of Differential Evolution, Genetic Algorithm, and a global Sequential Quadratic Programming (SQP) algorithm were employed on the problem. They did not solve the problem adequately. The SQP algorithm did optimize the problem and produced stable gait trajectories with different weight values. These results were published in *Assistive Exoskeleton Control with User-Tuned Multi-Objective Optimization* [59] at the 2019 IEEE International Conference on Rehabilitation Robotics. The algorithm, however, favoured one cost function over the rest. To address this issue a scaling method was used. Each value was normalized by the greatest value yet found for each cost function. This normalization technique was shown to be unsuccessful. Other normalization techniques did not seem to be appropriate for the problem. Utopian normalization, for instance, was not feasible as the search space is not known and thus the utopian point is unknown. These

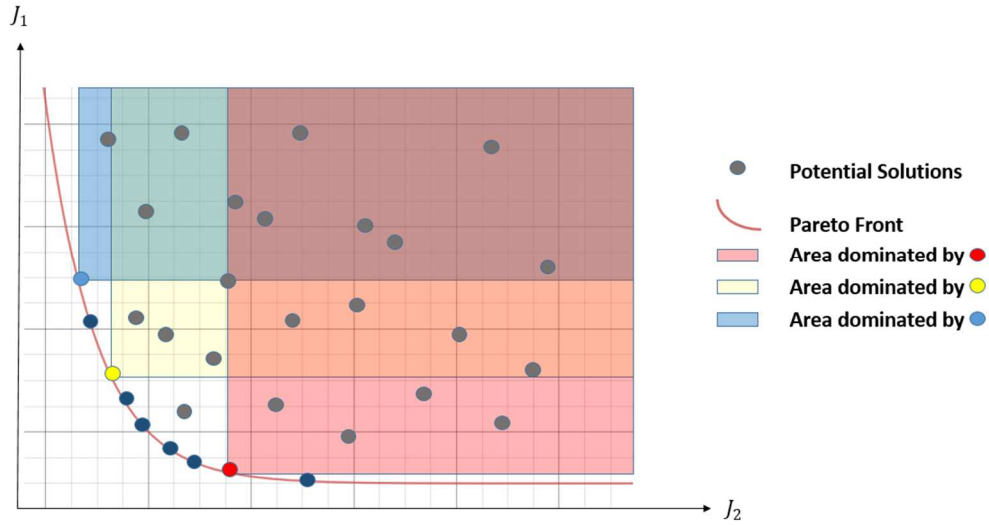
normalization techniques and other similar methods have the same inherent issue of attempting to linearize a nonlinear space [60]. These methods were thus rejected for solving the optimization problem and further literature search was done into multi-objective optimization and problem formulation.

## 2.8 Multi-objective Optimization

Multi-objective optimization has the goal of determining the Pareto front for multiple cost functions [61]. A multi-objective optimizer optimizes  $n$  objective functions together (6). It takes in each separate cost function, not a single cost function with different weights like the sum-of-squares method previously attempted, and optimizes for the Pareto front of all cost functions.

$$J_T = J_1, J_2, \dots, J_n \quad (6)$$

The Pareto front is the set of points in which all cost functions are non-dominated. Domination means that moving an input in any direction there is a decrease in the value of the other cost functions. The development of a Pareto front gives optimal solutions for a range of different individual cost function values. A Pareto front showing non-dominated points of two cost functions,  $J_1$  and  $J_2$ , can be seen in Figure 9. A Pareto front contains all points which cannot be improved upon. Three regions are illustrated in Figure 9 to show the areas which different points along the Pareto front dominate.



**Figure 9: Domination and Pareto Front**

NSGA-II, SMS-EMOA, and MOEA/D [62] are multi-objective algorithms that have been shown to solve nonlinear optimization problems. The Non-dominated Sorting Genetic Algorithm II [63] (NSGA-II) is a multi-objective method that has been shown to be adequate in optimizing multi-objective problems without normalization issues such as were seen using the global optimization algorithms.

### 3 Methods

The research contribution of this thesis is the development of a controller that allows user adjustment of performance measures to select the desired gait topology. The first phase of the research (Figure 10) had the goal of generating a look-up table of virtual constraint trajectories off-line to be accessed by the supervisory controller. The second phase (Figure 10) was providing a user interface framework where the controller accesses the pre-computed look-up table based on the specific user selection. The performance measures chosen were an effort proxy, a measure of natural walking, comfort level, and speed.

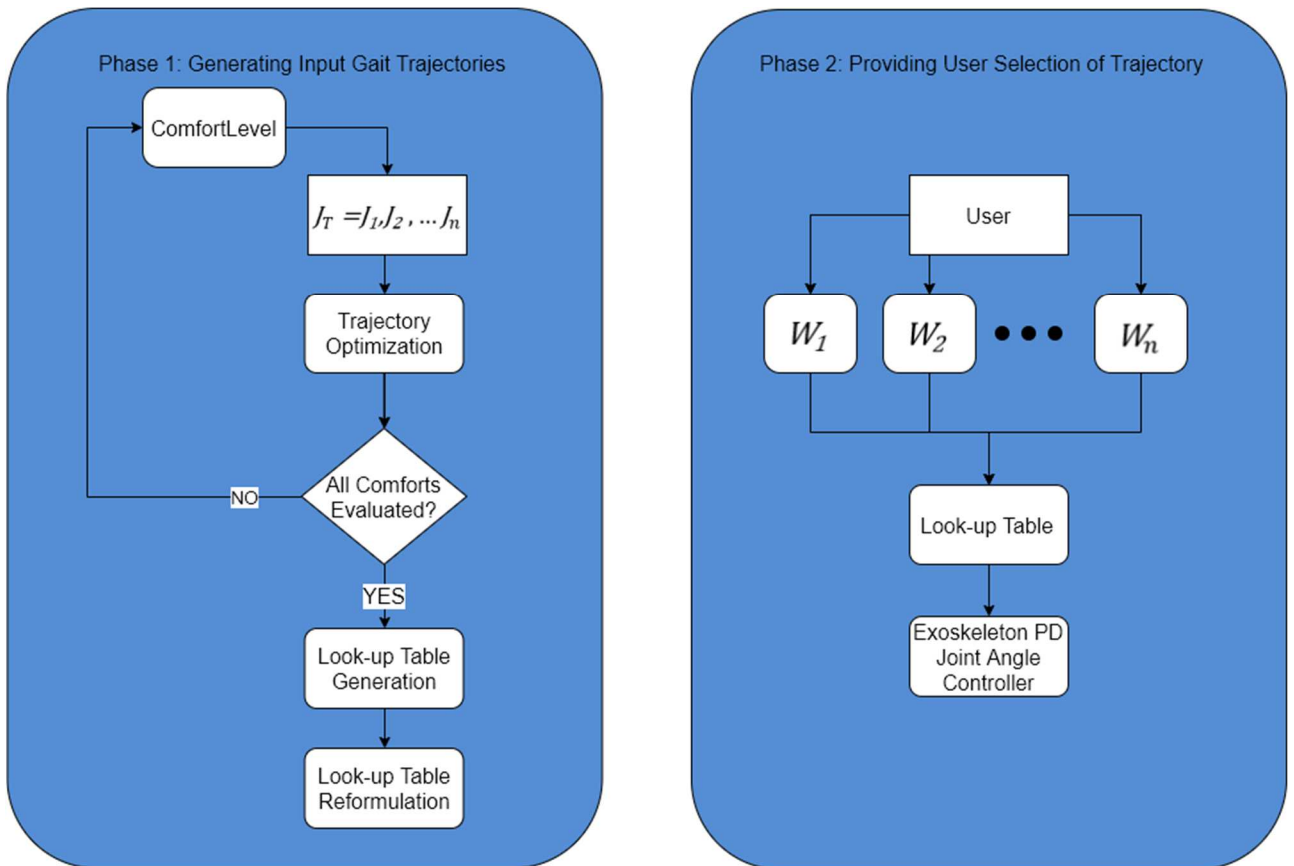


Figure 10: Research Framework

The development of a lookup table that allows for user tuning of exoskeleton gait by the pertinent performance measures was done as an optimization problem. The optimization modified joint angle trajectories,  $\Theta_I$ , as inputs to a simulation model of an exoskeleton. This optimization framework can be seen in Figure 11.

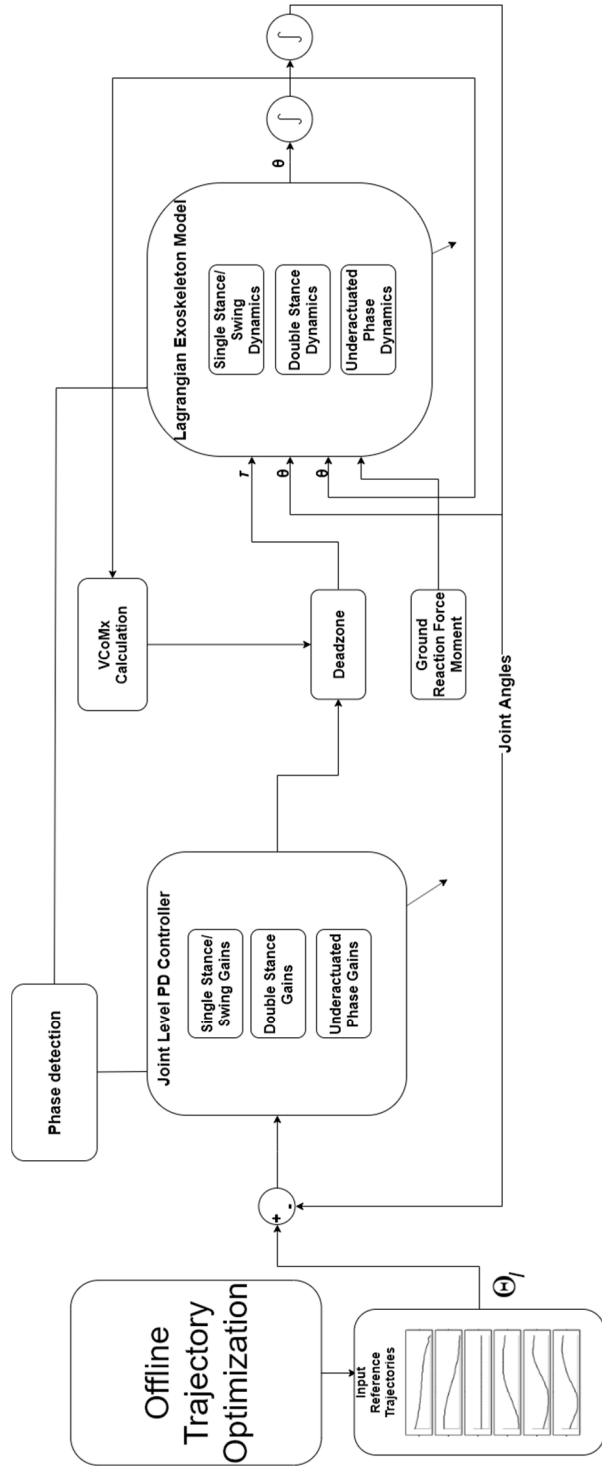
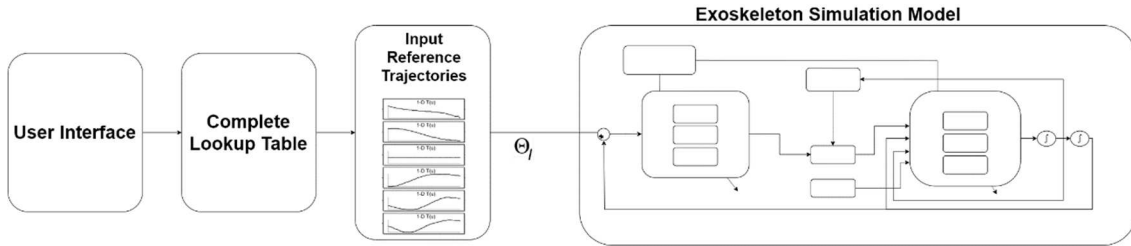


Figure 11: Phase 1 Controller Overview

Once the lookup table was completed a user interface allowing the user tunable control takes the form seen in Figure 12.



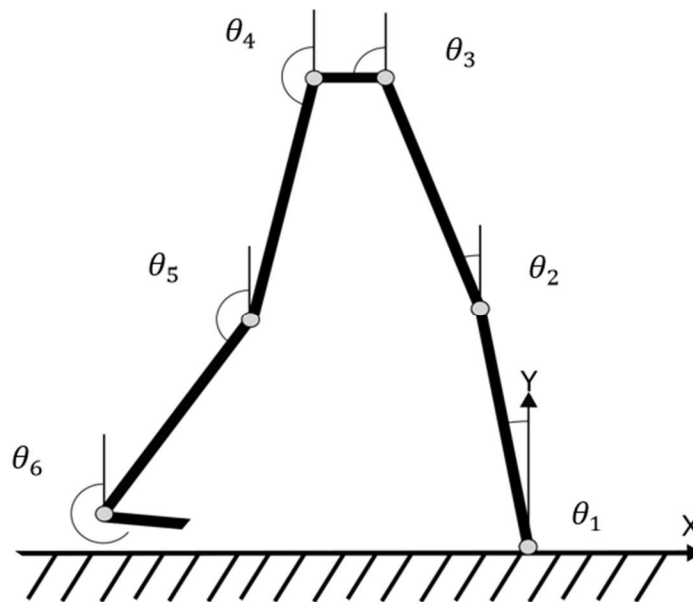
**Figure 12: Phase 2 User Interface Controller**

### 3.1 Exoskeleton Simulation Model

A computational model of an exoskeleton was used for verification of the controller and formulating the look-up table of gait trajectories. The model was based on the simulation of exoskeleton gait by Samuel Campbell [64], which implements Lagrangian mechanics to model gait trajectories based on torque inputs using Simulink. It was developed based on the methods of McGrath and Mu [65], [66]. The bipedal system is modelled using a single kinematic chain, composed of 6 links, a swing and stance thigh and shank, a single swing foot and a hip link; seen in Figure 13. The stance foot was modelled as a point-foot. The link between the two hips was modelled with a length of zero which allows a single branch kinematic chain. To represent the torso dynamics the center of mass of the link between the two hips was defined with an eccentricity of 0.337 m upwards and mass 43.198 kg. The eccentricity places the location of the center of mass away from the central axis of the link to model the center of mass of the torso. The model parameters of mass, eccentricity, and length were chosen to be those used by Mu [65], these parameter values are listed in Table 2.

**Table 2: Exoskeleton Model Mechanical Component Values**

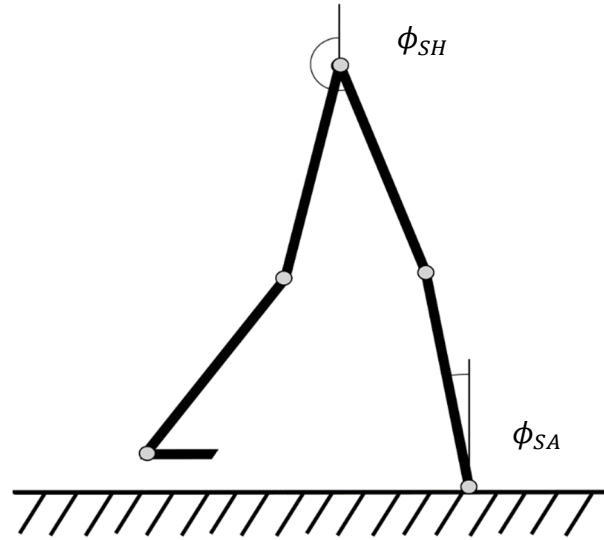
	Stance	Stance	Stance	Swing	Swing	Swing
	Shank	Thigh	Hip	Thigh	Shank	Foot
Mass (kg)	3.724	8.232	43.198	8.232	3.724	1.121
Eccentricity (m)	0	0	0.337	0	0	0
Center of Mass (m)	0.239	0.2405	0	0.184	0.183	0.075
Length (m)	0.42164	0.42418	0	0.42418	0.42164	0.13262
Inertia (kg m <sup>2</sup> )	0.065	0.14	10.809	0.14	0.064	0.006



**Figure 13: Simulation Model**

### 3.1.1 Virtual Constraints

Holonomic virtual constraints were used to provide time-invariant control. The gait reference trajectories were divided into swing and stance for the ankle, knee, and hip joints. The stance phase of gait, including underactuated and double-support, is 60% of the gait cycle and the swing phase is 40% [67]. Winter's data of one gait cycle was divided up into these percentages and each joint angle trajectory was reformulated to the coordinate reference frame with angles from verticle. These angles were then mapped based on the phase variable instead of time. This mapping is done by associating each joint angle at an instance in time to the instantaneous value of the phase variable, the result is a set of monotonic reference trajectories for joint reference control. The phase variable allows for autonomous feedback-linearized control. The specific phase variable employed was the difference between the stance hip angle,  $\phi_{SH}$ , and the stance ankle angle,  $\phi_{SA}$ , seen in Figure 14. At each time-step, the computational model calculates the phase variable value to determine all desired output reference trajectory values using a lookup table. All trajectories were optimized using the same virtual constraint phase variable vector of length 37.



**Figure 14: Phase Variable Measure**

To drive the bipedal system to its desired state proportional-derivative (PD) feedback controllers on the joint angles were used to drive the system to the desired state. The input reference trajectories,  $\Theta_I$ , were manipulated in the optimization to minimize the multi-objective cost function. The overall control diagram used to generate the lookup table of trajectories can be seen in Figure 11.

### 3.1.2 Gait Phase Division of Dynamics

The dynamics of the bipedal model were divided into three different phases: stance-swing, double support, and underactuation. The gait was divided into these phases due to the unique dynamics of each phase. The controller switches between phases to produce the whole gait cycle based on a phase trigger. The transition from stance-swing to underactuation is done by measuring the center of mass location. When the center of mass goes over the stance foot the underactuation phase starts. The transition from underactuation to double support is when the swing foot hits the ground. The transition from double support to

stance-swing is when the swing phase begins. Considering that each phase has its own dynamics each phase also had its own set of controller gains which changes the impedance of the model and ensured stable convergence onto the reference trajectories.

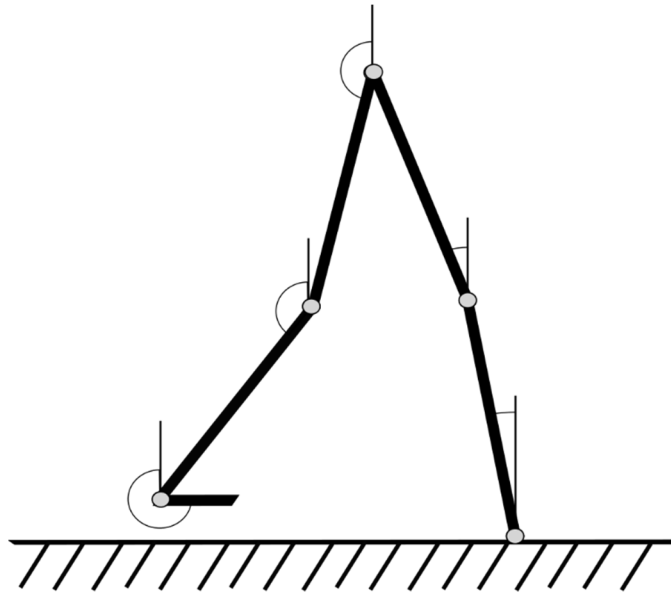
The main phase of walking is the stance-swing phase, as seen in Figure 15, which has continuous dynamics. This phase commences when the trailing leg lifts off the ground. One leg is in contact with the ground making it an open-loop kinematic chain. The mechanics were formulated using Lagrangian methods. The ankle torque is such that all torques allow the forces at the point of contact to sum to zero. The continuous dynamics of the stance-swing phase are described in (7).

$$\mathbf{B}_{(6 \times 6)} \ddot{\boldsymbol{\theta}} = \mathbf{C}_{(6 \times 1)} + \mathbf{M}_{ss} \quad (7)$$

Where  $\ddot{\boldsymbol{\theta}}$  are the joint angular accelerations for each joint,  $\mathbf{B}_{(6 \times 6)}$  is the inertial matrix,  $\mathbf{C}_{(6 \times 1)}$  is the Coriolis-gravitational matrix, and  $\mathbf{M}_{ss}$  are the total moments applied across each link,

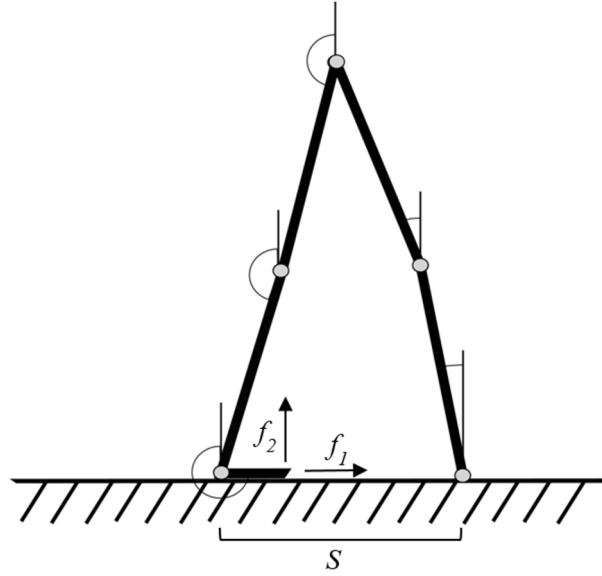
$$\mathbf{M}_{ss} = \begin{bmatrix} T_1 - T_2 \\ T_2 - T_3 \\ T_3 - T_4 \\ T_4 - T_5 \\ T_5 - T_6 \\ T_6 \end{bmatrix} \quad (8)$$

Where  $T_i$  are the corresponding torques of the control input torque matrix,  $\mathbf{T}$ .



**Figure 15: Swing Phase**

The double-support phase is when both feet are on the ground, as seen in Figure 16, making a closed-loop kinematic chain. This phase is an important phase of gait as the interactions of the two legs influence the posture of the exoskeleton. The transition from swing to stance was modelled as an instantaneous impulse that requires remapping of the model kinematics. The closed-form kinematic chain requires constraint equations,  $f_1$  and  $f_2$ , to ensure the rear foot remains on the ground while the step length stays the same (9),(10).



**Figure 16: Double Support Phase**

$$f_1 = \left( \sum_{i=1}^{\eta} (-l_i \sin(\theta_i)) \right) + S = 0 \quad (9)$$

$$f_2 = \left( \sum_{i=1}^{\eta} (l_i \cos(\theta_i)) \right) = 0 \quad (10)$$

where,  $l_i$  are the respective the link lengths,  $\theta_i$  are respective joint angles,  $\eta$  is the number of joint links, and  $S$  is the step length of the bipedal system. These two constraints allow rotation about the trailing toe until the beginning of the swing phase. To develop the fully formed dynamics of the double-support phase the positional constraints are differentiated twice becoming  $\lambda_{f1}$  and  $\lambda_{f2}$ . The fully formed dynamics then become,

$$\begin{bmatrix} \mathbf{B}_{(6 \times 6)} & l_i \cos(\theta_i) & l_i \sin(\theta_i) \\ l_i \cos(\theta_i) & 0 & 0 \\ l_i \sin(\theta_i) & 0 & 0 \end{bmatrix} \begin{bmatrix} \ddot{\boldsymbol{\theta}} \\ \lambda_{f1} \\ \lambda_{f2} \end{bmatrix} = \begin{bmatrix} \mathbf{C}_{(6 \times 1)} \\ \sum_{i=1}^6 l_i \dot{\theta}_i \sin(\theta_i) \\ \sum_{i=1}^6 -l_i \dot{\theta}_i \cos(\theta_i) \end{bmatrix} + \begin{bmatrix} \mathbf{M}_{ds} \\ 0 \\ 0 \end{bmatrix} \quad (11)$$

where,

$$\mathbf{M}_{ds} = \begin{bmatrix} T_1 - T_2 \\ T_2 - T_3 \\ T_3 - T_4 \\ T_4 - T_5 \\ T_5 - T_6 \\ T_6 \end{bmatrix} \quad (12)$$

The underactuated phase is the phase that allows for more natural walking while not constraining the ankle. During this phase, the momentum of the bipedal system is conserved and carries the system through to complete a step. Once the center of mass,  $CoM_x$ , falls over the stance foot it pivots about the toe, adding an additional moment to the system. The dynamics are thus formulated with the additional toe link in the kinematic chain as a seventh order formulation; see Figure 17. This additional moment at the toe is what is produced by pushing off the ground,  $M_{GRF}$ . The ground reaction force producing this torque was modelled as a spring with spring constant  $k=1667$  N/m.

$$\mathbf{B}_{(7 \times 7)} \ddot{\boldsymbol{\theta}}_{ua} = \mathbf{C}_{(7 \times 1)} + \mathbf{M}_{ua} \quad (13)$$

where,

$$\mathbf{M}_{ua} = \begin{bmatrix} M_{CoM} - T_1 \\ T_1 - T_2 \\ T_2 - T_3 \\ T_3 - T_4 \\ T_4 - T_5 \\ T_5 - T_6 \\ T_6 - M_{GRF} \end{bmatrix} \quad (14)$$

where  $M_{GRF}$  is the moment created by the ground reaction force and  $M_{CoM}$  is the moment created by the rotation of the center of mass of the biped.  $\ddot{\theta}_{ua}$  is the augmented matrix with the angular acceleration of uncontrolled toe of the rear foot,  $\alpha$ ,

$$\ddot{\theta}_{ua} = \begin{bmatrix} \alpha \\ \ddot{\theta} \end{bmatrix} \quad (15)$$

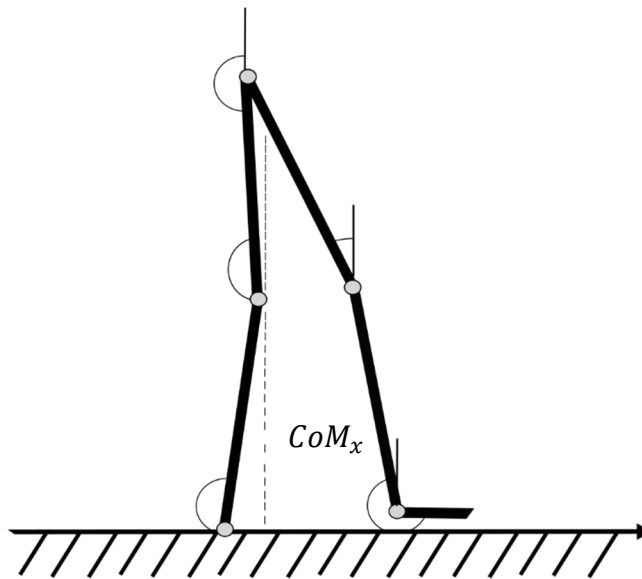


Figure 17: Underactuated Phase

### 3.1.3 Gain Selection

The gains for the joint level PD controllers were chosen heuristically and were proven to provide stability for a wide range of gait through testing. These robust gains, which can be observed in Table 3, were determined to be stable for a range wider than the maximal deadzone size used. The output torques of these controllers were saturated at 160 N, which is the maximal output of the PAULLO exoskeleton motors in development here at UNB.

**Table 3: PD Gains**

Phase	Index	Joint	Proportional	Derivative
Stance-Swing	1	Stance Ankle	400	250
	2	Stance Knee	500	200
	3	Stance Hip	390	190
	4	Swing Hip	325	50
	5	Swing Knee	290	50
	6	Swing Ankle	105	5
Double Support	1	Stance Ankle	50	175
	2	Stance Knee	100	50
	3	Stance Hip	150	75
	4	Swing Hip	100	50
	5	Swing Knee	50	25
	6	Swing Ankle	50	15
Underactuation	1	Stance Ankle	25	380
	2	Stance Knee	330	70
	3	Stance Hip	10	10
	4	Swing Hip	175	125
	5	Swing Knee	50	145
	6	Swing Ankle	2.5	0.5

## 3.2 Performance Measures

To allow for user adjustment of the pertinent performance measures a multi-objective optimization between the performance measures of speed, effort proxy, and

biomimicry was computed. This was done using each as a quadratic cost while ascribing the comfort at a specific torque deadzone size. The selection of a quadratic cost modality is important as it ensures all cost values are positive. In the case of effort proxy, both positive and negative torques incur effort and should both contribute to the cost equally. In order for the total effort proxy to be evaluated squaring the total cost allows both to contribute to a measure of the total power used. The optimization of each convex function was done using the NSGA-II multi-objective optimizer.

### 3.2.1 Biomimicry Cost

To allow the user to tune for more natural walking a cost of biomimicry was implemented. The cost associated with natural walking, biomimicry cost,  $J_{Bio}$ , was defined as a measure of the difference between the input optimized trajectory set,  $\Theta_I$  and D.A. Winter's averaged able-bodied gait trajectory data [68],  $\Theta_W$ . The cost was measured between the two sets of joint angle trajectories across a single gait cycle as,

$$J_{Bio_i} = \int (\Theta_I(t) - \Theta_W(t))^2 dt \quad (16)$$

where  $i$  is the individual cost of each joint. The difference between these trajectories is squared and integrated across one gait cycle. The total biomimicry cost is summed as the total cost from all joint curves,

$$J_{Bio} = \sum J_{Bio_i} \quad (17)$$

### 3.2.2 Effort Proxy Cost

The amount of effort the user requires to walk is another user performance measure. The effort proxy cost used to complete gait through one step was measured as the sum of squares of all joint torques through one step, as there is no damping in the system this gives the effort input into the system. To evaluate the cost of the effort proxy at each joint, the torque output of the controllers,  $\mathbf{T}$ , was squared and summed. To evaluate the fitness of each of the cost functions the following outputs were generated once the model simulation was complete.

$$J_{EffortProxy} = \int \mathbf{T}^T \mathbf{T} dt \quad (18)$$

### 3.2.3 Speed Cost

The speed at which the user wishes to walk is an integral part of tuning to the user's desires. Natural walking speeds are between 0.94 m/s and 1.4 m/s [69]. The cost of speed,  $J_{Speed}$ , was measured as the square of the difference between the actual speed of the model and the desired speed at which the user wishes to walk:

$$J_{Speed} = (Speed_{Desired} - Speed_{Actual})^2 \quad (19)$$

To produce walking speeds across the span of natural walking the lookup table of each comfort weight was taken at  $Speed_{Desired}$  of 0.95 m/s to 1.4 m/s in 0.05 m/s intervals. The  $Speed_{Actual}$  was measured as

$$Speed_{Actual} = mean(v_{CoMx}) \quad (20)$$

Where  $v_{CoMx}$  is the horizontal velocity of the center of mass.

### 3.2.4 Comfort

The comfort user adjustment was implemented as a torque deadzone. The deadzone provides a region for no torque to be applied within a specific distance from the reference trajectory. The deadzone size spans between 0 and 0.9 standard deviations of gait from the natural walking average, taken from Winter's Data [42]. If the user is experiencing more discomfort the user can adjust the size of the deadzone.

#### 3.2.4.1 Momentum Constraint

The implementation of a deadzone in the model causes issues in the feedback linearization allowing the model to lose momentum. A velocity constraint developed by Sam Campbell [64] was used to ensure that the bipedal model achieves stable walking. This velocity constraint narrows the deadzone size when the velocity is lower than  $v_{min}=0.95$  m/s. Otherwise, there would be a momentum loss and gait would not be completed. This momentum throttle was constructed using a sigmoid equation that scales the deadzone size,  $S.F.$ , down to zero at the low speeds.

$$S.F. = -50000 \frac{s^5}{m^5} \left( |v_{CoMx}| - v_{min} \frac{m}{s} \right)^5 - 860 \frac{s^4}{m^4} \left( |v_{CoMx}| - v_{min} \frac{m}{s} \right)^4 + 1586 \frac{s^3}{m^3} \left( |v_{CoMx}| - v_{min} \frac{m}{s} \right)^3 \quad (21)$$

### 3.2.5 Multi-objective Formulation

The final cost function for each trajectory was evaluated as,

$$J_{total} = J_{Bio}, J_{EffortProxy}, J_{Speed} \quad (22)$$

A multi-objective global optimization algorithm was used to determine a Pareto front of these cost functions for ten individual comfort weights. The comfort weights spanned between 0 and 0.9 standard deviations from Winter's Data of natural walking gait at intervals of 0.1 standard deviations. The individual Pareto fronts for each comfort weight is the full-formed reference lookup table.

### **3.3 Optimization Constraints**

To ensure that the result of the optimization produced stable gait, the optimization problem required constraints. Multiple constraints were used to ensure gait stability. The constraint handling technique used for all of the constraints was a hard penalty technique. The penalty value that was used was  $1e14$ . This value was used as it was close to twice the magnitude of any of the maximum values of the cost functions. Using this penalizing penalty value always resulted in unstable solutions being rejected by the optimization algorithm while testing, thus ensuring that the Pareto front only contained stable solutions.

The first constraint that this penalizing method handled were all cases where the computational model was unable to run a complete simulation. Incomplete simulations would occur if the dynamics of the system were highly unstable and the instability would crash the model during simulation.

To ensure the stability of the gait limit-cycle a Poincaré map constraint was used. To address the stability of the full system Poincaré return maps of the stride length and velocity of the center of mass were chosen for full state information. The Poincaré maps of the model were determined, and the gradients of the Poincaré map were calculated at the

crossover point of the  $\mathbf{x}_c = \mathbf{x}_{c-1}$  line. If any of the gradients before the crossover had a magnitude value higher than 1 then the curve penalized.

The final constraint was establishing that the limit cycle was not chaotic. This constraint was done using a margin of stability around the limit cycle. A chaotic margin 0.7 in the Poincaré return space was used as it was shown to be within the range of stable solutions for the biomimetic bounds. In cases that the Poincaré map had a gradient of magnitude greater than 1, but stayed within the chaotic bounds these results were deemed as stable. These constraints were shown adequate to keep the bipedal system within stable walking bounds.

### **3.4 Non-dominated Sorting Genetic Algorithm II**

The Non-dominated Sorting Genetic Algorithm II (NSGA-II) was selected to perform multi-objective optimization and formulate the trajectory lookup table. The algorithm uses a genetic algorithm and a sorting operation to create Pareto-optimal fronts. Once a Pareto optimal front of different performance measure weights has been established this allows for user adjustment to the manner users wish to walk. This algorithm alleviated the normalization issues found in previous attempts of global optimizers to achieve a Pareto front of trajectories.

The conventional genetic algorithm uses selection, mating, and mutation to optimize a system. The NSGA-II uses these same operations to optimize, with the addition of non-dominated sorting and crowding distance sorting to perform multi-objective optimization and the Pareto front formulation. The optimization flow of the NSGA-II

algorithm can be seen in Figure 18. It is important to note that the Pareto front selection always includes both parents and children, making it an elitist algorithm.

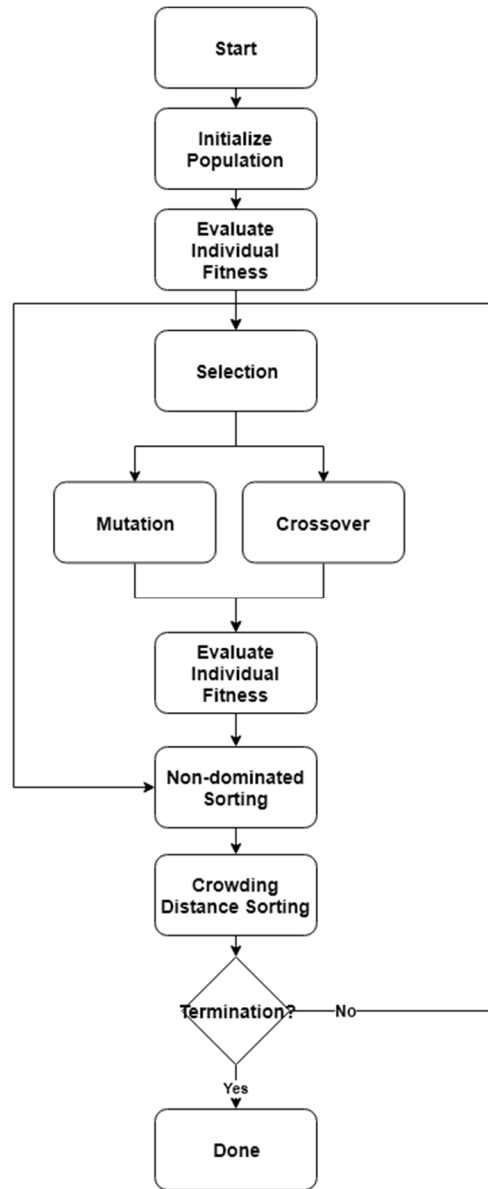


Figure 18: NSGA-II Algorithm

### 3.4.1 Classical Genetic Algorithm Operations

The NSGA-II first completes the classical genetic algorithm operations of selection, mating, and mutation but also contains additional components that provide

Pareto front formulation. The selection operation is the process of determining which individuals should be considered for mating then passes these characteristics on to the future generation. The mating operation is when two individuals in the current, also known as the parent, population exchange data with each other. The mating technique that is commonly used is a crossover operation, where two parents in the population will swap different sections of their genetic information with each other. The mutation operation is when some of the genetic material is randomized to allow exploration of the search space. The specific algorithmic operations were determined iteratively through both intuitive selection and ad hoc methods. The added components of non-dominated sorting and crowding distance sorting formulate the Pareto front of the multi-objective problem.

### **3.4.2 Non-dominated Sorting**

To establish a Pareto front, the NSGA-II first implements non-dominated sorting. This sorting is done by assigning each individual two parameters:  $n_p$  and  $\mathcal{S}_p$ .  $n_p$  is the number of points that dominate that individual.  $\mathcal{S}_p$  is a set of individuals which the current individual dominates. Dominated sorting is the process of checking  $n_p$  of each individual and assigning them to the corresponding Pareto fronts; where  $n_p = 0$  is the best Pareto front found thus far. Once the set of the first Pareto front has been established the algorithm decreases the  $n_p$  of all of the individuals in  $\mathcal{S}_p$ , if any of these individuals then have a  $n_p$  of zero they are added to the second Pareto front and so on until all individuals in the population have been assigned to a Pareto front.

### 3.4.3 Crowding Distance Sorting

Once all the Pareto fronts have been established, the individuals that propagate to the next generation need to be established. Propagation is done using crowding distance sorting. A parallelepiped, termed a cuboid in [63], in all dimensions of the cost functions is calculated for all individuals in the population. The crowding distance value is the perimeter of the parallelepiped drawn with the nearest two individuals as vertices (Figure 19). The algorithm prioritizes genetic diversity and selects those individuals who have a larger crowding distance value. The other individuals are discarded.

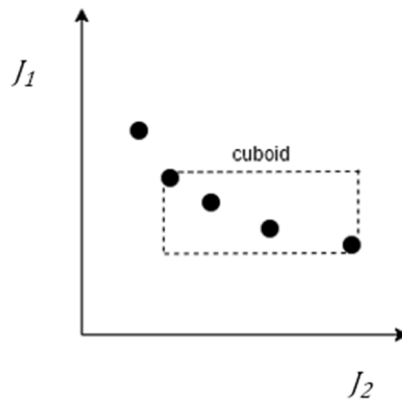


Figure 19: Crowding Distance Measure [63]

The overall Pareto sorting method can be seen in Figure 20, where the population,  $\mathbf{P}_t$ , and its parents,  $\mathbf{Q}_t$ , are first subjected to the non-dominated sorting and then crowding distance sorting to establish the Pareto fronts to be the next generation of individuals,  $\mathbf{P}_{t+1}$ .

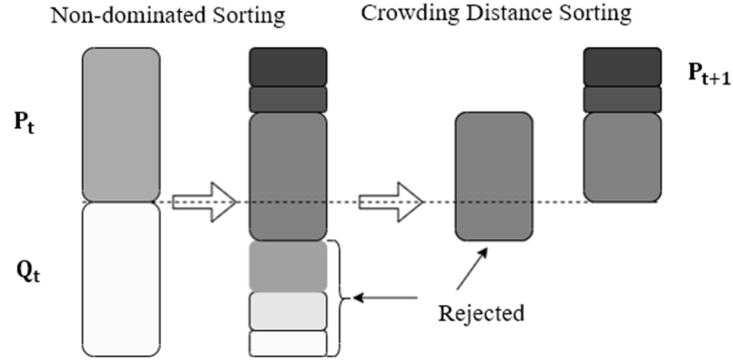
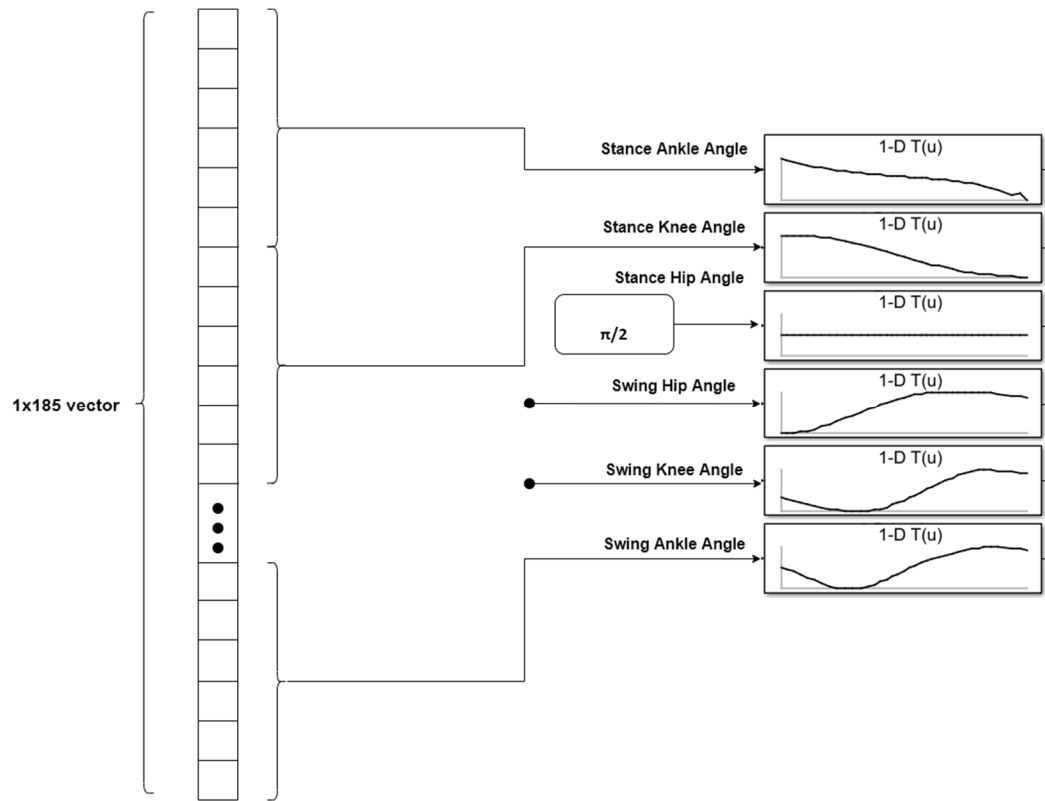


Figure 20: Pareto Sorting NSGA-II [63]

### 3.5 Optimization Problem Formulation

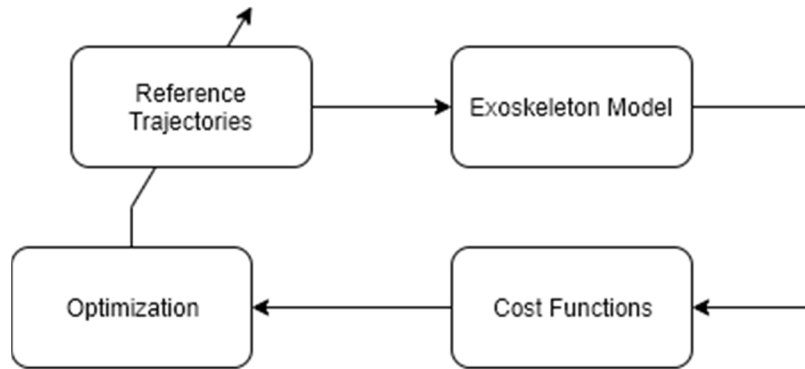
The optimization was done using the NSGA-II algorithm based on MATLAB's implementation in the *gamutobj* application protocol interface. The algorithm was used to modify the input gait trajectories,  $\Theta_I$ , run through the bipedal model to perform the minimization of the multi-objective problem. The phase variable was kept constant and the inputs of the stance ankle, stance knee, stance hip, swing knee, and swing ankle were manipulated using the genetic-based algorithm. The swing hip trajectory was kept constant at  $\pi/2$  radians to keep upright walking for the bipedal model which is desirable for most users. Each individual comprised of a vector of the combined trajectories, which was passed to the model. Each trajectory was a curve of length 37, which is the length of the phase variable vector to ensure a one-to-one lookup table. An individual in the population,  $P_n$ , is a vector of length 185 that define the five curves to be optimized by the algorithm as seen in Figure 21. The optimization ran the model and each cost function and constraint would be evaluated based on the corresponding input trajectory.



**Figure 21: Optimization Inputs**

Each simulation of the model involved inputting the trajectories,  $\Theta_I$ , and the size of the deadzone into the model. The model was simulated in a rapid simulation executable to save computation time. The rapid simulation executable was made by Matlab's C language wrapper and then compiled into an executable. The model was run for a simulation time of seven seconds allowing for approximately ten steps at the highest speed of 1.4 m/s. This number of steps was deemed adequate in evaluating the stability of walking. Once the simulation terminated the required output parameters of the system were then loaded into the fitness function and each cost function evaluated. The optimization was done until reaching one of the stopping criteriums of time, fitness cost, or a maximum

number of generations (Table 4). The optimization was done by evaluating the costs over one step for each trajectory set.



**Figure 22: Optimization Flow**

### **3.6 Algorithmic Parameters**

Selecting the algorithmic parameters was an iterative process. The final parameters used were selected as they were found to solve for a Pareto set. As is the case with any heuristic optimization algorithm, the optimization parameters and operations can be potentially improved to better solve the given problem. The tuning of the algorithmic parameters started using general best practices and intuition from the problem formulation and evolved from there to best fit the design outcomes of the problem.

The initial conditions of any Genetic Algorithm are important values for the success of the algorithm. In most instances using a Genetic Algorithm to solve a problem, the initial population is randomized within the search space. This method was attempted and ultimately was not providing any stable gait solutions due to the long computation time. The initial conditions that were used in the final optimization comprised of a population of  $2N$  individuals that contained input gait trajectories shown to be stable through testing. These individuals were a set of trajectories that were tested and observed to be stable across

all sizes of the comfort deadzone for the set of gains used. The initial population also included individuals which were plus and minus one standard deviation from Winter's average curves,  $\Theta_W$ , to promote diversity in the search. This initial population allowed for exploration including individuals near the bounds of the search space. These initial conditions allowed for a significant increase in the speed of computation.

A population number of  $N=10$  was used, which allowed for fast computation and was sufficiently large to find different points on the Pareto front. The size of the population determines the maximum number of individuals that can be on the final Pareto front. To provide adequate tuneability of the performance measures the population number is required to be of significant value.

The selection operation that was used was binary tournament selection. This selection was the selection method that was used by Deb et al. [63]. The binary tournament selection selects two individuals at random from the population and the one that has a better crowding distance gets selected for the mating and mutation. This selection is done until a new population of size  $N$  has been formed.

The selected mutation method was shown to be a key factor in determining the performance of the algorithm. The mutation method of stepping in a feasible direction was used at first. However, the algorithm took a long time needing over 2000 iterations, taking about a day to generate one trajectory besides the stable initial condition set of curves that were passed to the algorithm at the beginning of the optimization. Various tuning of parameters resulted in little success. Upon changing the mutation method a substantial decrease in time resulted. In two hours of computation, the algorithm yielded a second curve on the Pareto front. This curve did not abide by the constraints of human walking as

the mutation algorithm was allowing the constraints to be violated. The mutation method that was used was a saturated uniform mutation. The method mutates a number of points in each individual based on a mutation probability value. Each mutation produced is a value within the range of the individual's range. The mutation was then further constrained by saturating each mutation within the biomimetic feasibility bounds to ensure that all walking trajectories were biomimetic and did not violate the human range of motion.

Unlike many genetic algorithms, the implemented NSGA-II performs mutation and crossover separately. The rate at which the population is mutated and the rate at which it is subject to crossover adds up to 100%. The crossover rate and mutation rate for the algorithm were chosen to be 20% and 80% respectively. Considering that mutation has been historically the operation that allows for the most learning [70] and the change in mutation allowed for more solutions on the Pareto front it was selected to be a higher value.

The other operational parameters were not as influential in the results, although still were important. The crossover operation was chosen to be an arithmetic crossover, which takes the genotypical mean of the two parents to form the offspring. The algorithm termination was determined based on the termination criterium. The algorithm would terminate if the fitness reached a level of  $-\text{INF}$  or a function tolerance of  $1\text{e-}4$  or a time limit of choosing of the programmer. The final selection of operational parameters for the algorithm is found in Table 4.

**Table 4: NSGA-II Algorithmic Parameters**

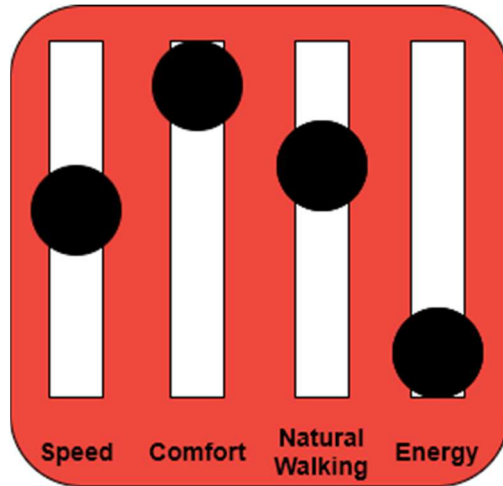
Population	$N=10$
Selection Method	Tournament Selection
Crossover Method	Arithmetic Crossover
Mutation Method	Mutation Saturated
Crossover Rate	20%
Mutation Rate	80%
Stopping Criteria	
Maximum Number of Generations	1000
Function Tolerance	1e-4
Maximum Optimization Time	36 hours

## 4 Results

The goal of this work was to develop a trajectory lookup table that allows user adjustment based on pertinent gait performance measures. Running the NSGA-II optimizer to develop Pareto fronts at different comfort weights was used in achieving this goal. By running the trajectories through a computational model of an exoskeleton and evaluating stability as well as performance measure values the algorithm was then able to formulate appropriate input trajectories based on the output cost optimization.

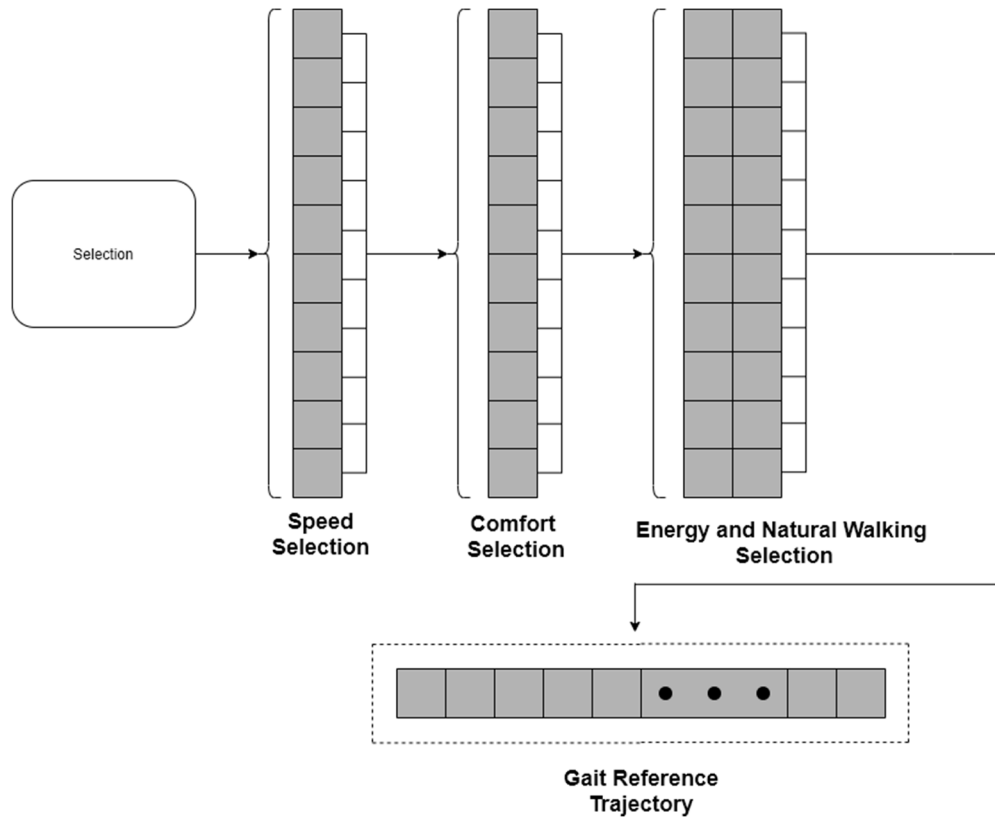
### 4.1 User Interface

The formulation of a user interface that allows intuitive control was accomplished through reformulation of the result of the optimization. A lookup table comprising 811 trajectories was generated that offers different walking trajectories based on different combinations of user performance metric values. This lookup table allows a user to input how much they care about each of these performance measures to select the manner in which they walk while using an exoskeleton. To adjust the controller to their desires, a user selects the values on the user interface, seen in Figure 23, for each of the user performance measures: speed, comfort, effort proxy, and natural walking.



**Figure 23: User Interface**

The input values of the performance metrics select the corresponding trajectory that has been optimized. The lookup table, which can be seen in Figure 24, was built using structures to access each corresponding trajectory. The layers of the lookup table are desired speed, then comfort and then a Pareto front for the costs of speed, effort proxy, and biomimicry.



**Figure 24: Reference Lookup Table Method**

To allow user-tunable control, the lookup table needed to be reformulated. Reformulation of the lookup table was done by mapping the outer layer according to the actual speed achieved. The range of the speeds found was divided into ten segments. Each trajectory in the fully-formed structure was reassigned to the new lookup table based on the segment of the speed range it was in. Mapping of the comfort level was done by ranking each secondary layer of the reformulated lookup table by the deadzone size. The final selection was done by adjusting the effort proxy value or the biomimicry value. The range of each one of the cost functions is mapped onto the corresponding slider to select trajectories in the lookup table.

## 4.2 Range of Outputs

The total range of outputs gives the current limitation of the user-adjustment for the controller. The NSGA-II optimizer was bounded to explore within a stable space although there were potential limitations within these bounds. The limits of these bounds can be seen by observing where unstable results were found, as well as the spread at certain performance values. In observing the full lookup table there was a limited spread in the ranges of speed and comfort. Each final lookup table should have resulted in a Pareto front of  $N$  points. The cases where the Pareto front was smaller are areas where feasible solutions could be limited or further exploration be required. The range of all parameters can be seen in Table 5.

**Table 5: Spread and Sensitivity of Look-up Table**

	Effort Proxy	Speed (m/s)	Biomimicry	Comfort (std)
Range	3.1393e+08	0.9449	1.5816	0.9
Average Value	6.1099e+07	1.2355	11.1387	0.4453
Standard Deviation	6.4433e+07	0.1497	0.0831	0.2896

### 4.2.1 Speed Range

The speed range was 0.9449 m/s for stable walking of the model. The maximum speed that was achieved was 1.73 m/s and the minimum speed was 0.787 m/s.

#### **4.2.2 Comfort Range**

The spread of comfort weights was shown to be stable across the whole set of deadzone sizes from 0-0.9 standard deviations. These stable results were expected as the stable initial conditions and gains were shown to be stable for all sizes of the deadzone for the optimization.

#### **4.2.3 Effort Proxy Cost Range**

The effort proxy cost range was shown to be between  $2.4255e+07 \text{ (N}\cdot\text{m)}^2$  and  $3.3818e+08 \text{ (N}\cdot\text{m)}^2$ .

#### **4.2.4 Natural Walking Range**

The natural walking range found was found to be between  $10.530 \text{ rad}^2$  and  $12.1126 \text{ rad}^2$ .

### **4.3 Optimization Sensitivity**

It is desirable to know the sensitivity of the optimization output to establish the amount of tuning a user can perform. The sensitivity of each of the performance measures varied. The number of different points on the Pareto front for unique effort proxy was 200. The outliers were  $2.4255e+07$  and  $3.3818e+08$ . The average number of unique biomimicry values was 68. The outliers were 10.5309 for a minimum and 12.1126 for maximum. The number of unique speeds was 210. The minimum was 0.7870 m/s and the maximum was 1.7319 m/s. The comfort sensitivity was preset to be 0.1 standard deviations of deadzone. The spread of comfort weight was limited for certain deadzone sizes at certain speeds

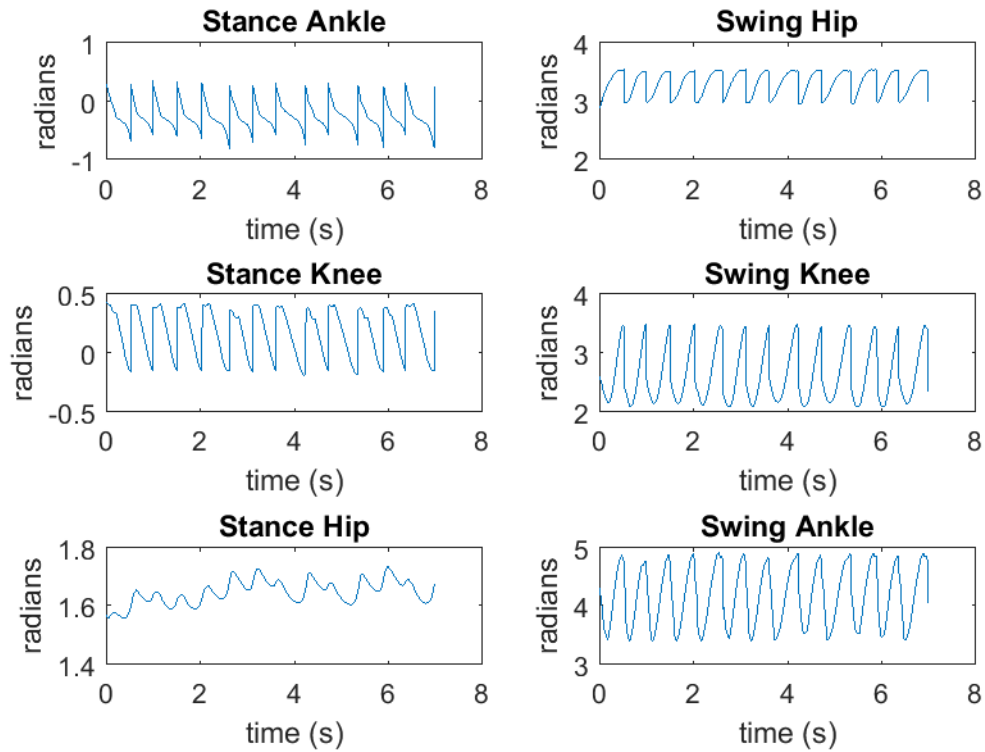
which had limited or no Pareto fronts. These sensitivity values give the amount of difference the user will be able to adjust based on their preferences.

#### **4.4 Prioritizing Parameters**

It is of importance to see how the gait trajectories differ from each other when prioritizing the different performance measures. Observing the difference in gait gives insight into the impact that each of the cost functions influences the gait that is produced. To observe these differences the maximal priority was given to each of the performance measures while minimizing the values of the other parameters.

##### **4.4.1 Prioritizing Comfort**

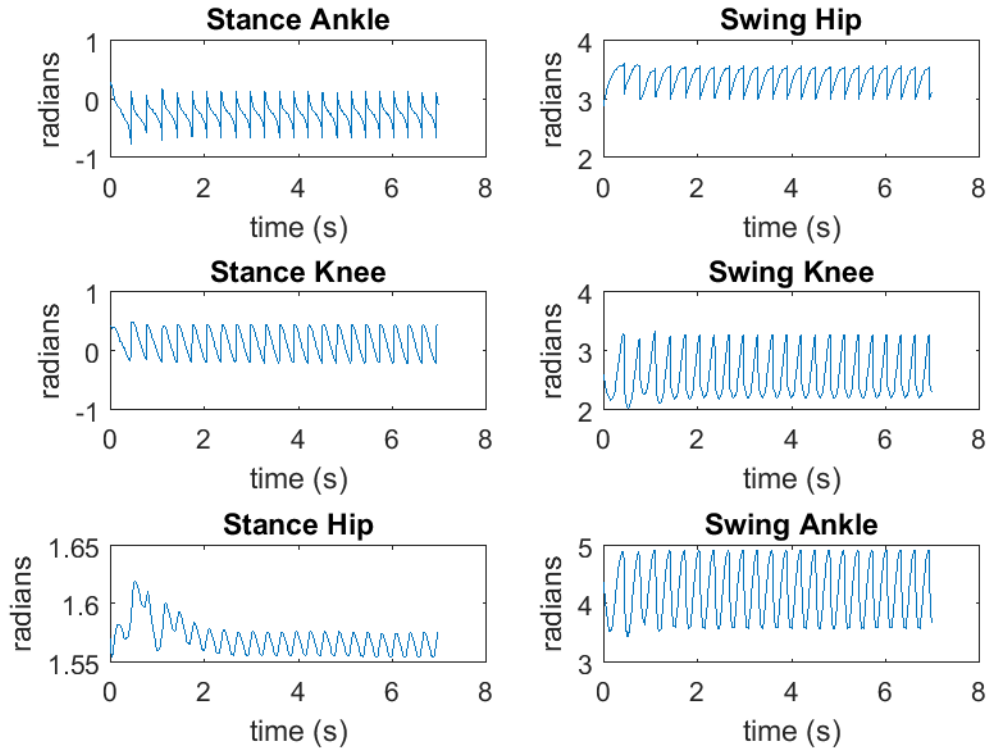
With the deadzone size set to a maximum, the other parameters took on values of speed of 1.1348 m/s, a  $J_{EffortProxy}$  of 4.8389e+07, and a  $J_{Bio}$  of 11.1399. The output gait trajectories with these user inputs can be seen in Figure 25. The system took 13 steps over the course of the simulation while prioritizing comfort.



**Figure 25: Prioritizing Comfort Output Gait Trajectories**

#### 4.4.2 Prioritizing Speed

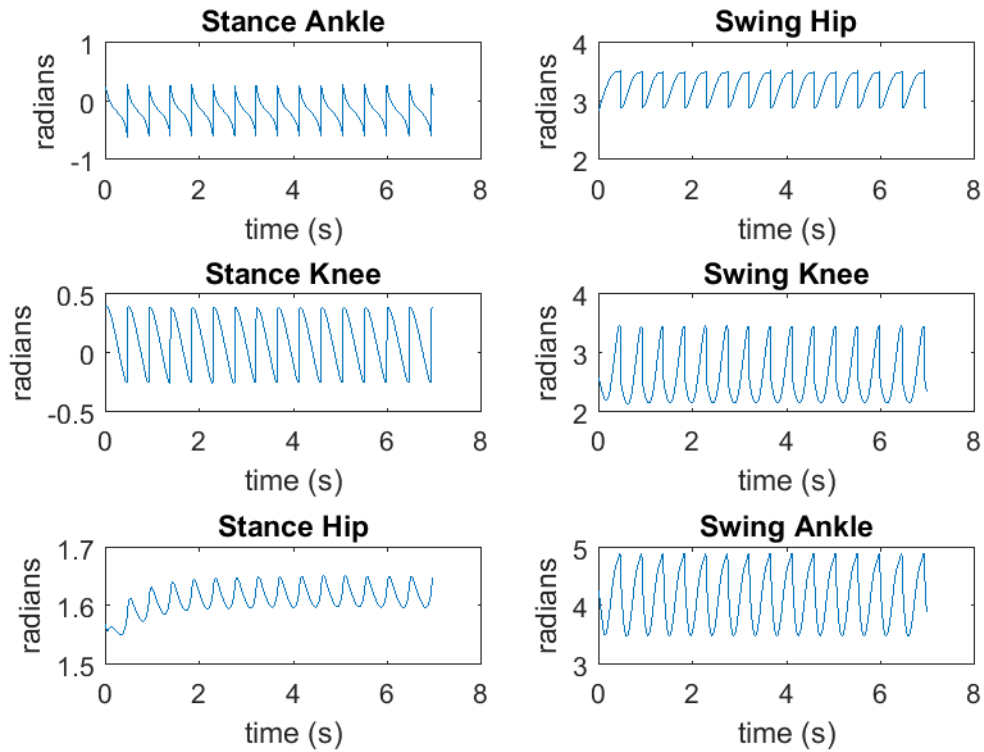
When the user prioritizes speed,  $Speed_{Actual}$  is at a maximum, the parameters of the other parameters took on a comfort level of 0.3 standard deviations,  $J_{EffortProxy}$  of  $2.0143e+08$ , and  $J_{Bio}$  of 11.0416. Figure 26 shows the trajectories resulting in setting the speed value to the maximum. The system took 22 steps over the course of the simulation while prioritizing speed.



**Figure 26: Prioritizing Speed Output Trajectories**

#### 4.4.3 Prioritizing Effort Proxy

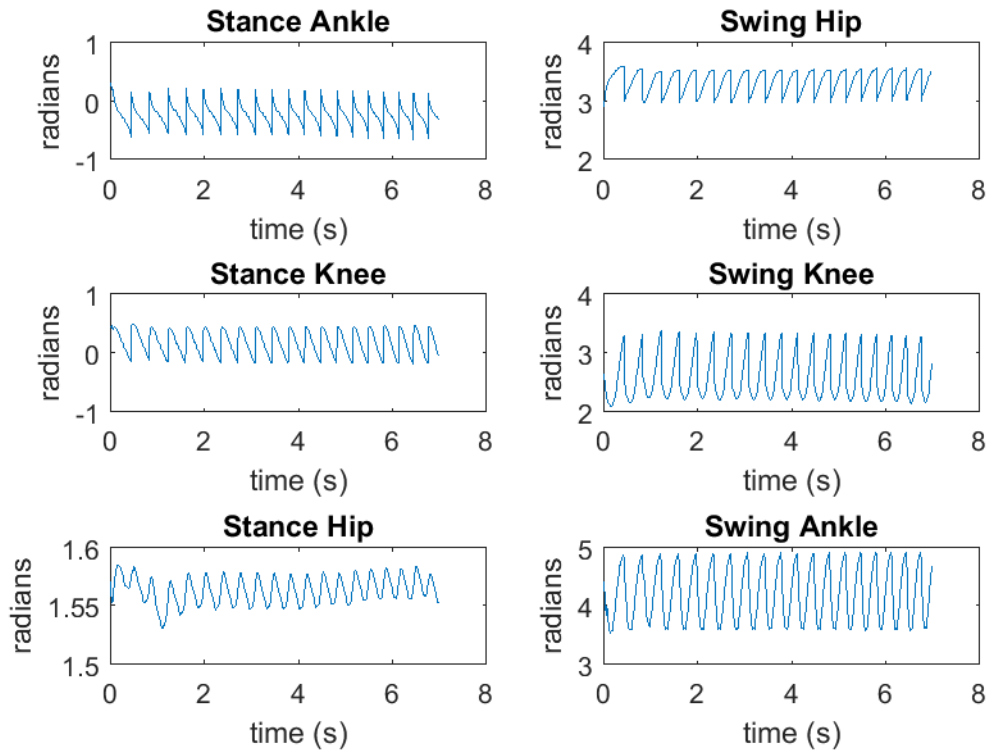
When a user prioritizes the effort proxy required to walk,  $J_{EffortProxy}$  at a minimum, the gait trajectories take the form seen in Figure 27. The parameters took values of speed of 1.3336 m/s, a comfort level of 0.3 standard deviations, and a  $J_{Bio}$  of 12.1126. The system took 15 steps over the course of the simulation while prioritizing the effort proxy.



**Figure 27: Prioritizing Effort Proxy Output Trajectories**

#### 4.4.4 Prioritizing Natural Walking

When a user prioritizes natural walking during gait,  $J_{Bio}$  is at a minimum, at the potential cost of the other performance measures the results are as seen in Figure 28. The parameters took values of speed of 1.5472 m/s,  $J_{EffortProxy} = 2.1684e+08$ , and a comfort level of 0.3 stand deviations. The system took 19 steps over the course of the simulation while prioritizing natural walking.



**Figure 28: Prioritizing Natural Walking Output Trajectories**

## 5 Discussion

The results show that the optimization formulation was appropriate and an acceptable variety of gait trajectories were formulated when prioritizing between different performance measure values. This variation in gait trajectories gives an amount of tuning that should allow meaningful adjustment for an exoskeleton user. These results inform of the eligible search space, control error, and show what future work can be done to further develop this concept.

### 5.1 Range of Outputs

The range of adjustment is an important factor when observing how much each parameter can be adjusted. The range of speeds of 0.7449 m/s offers a good range of walking speeds for user adjustment as it spans the average walking speeds of 0.9 m/s to 1.4 m/s. The maximum speed, of 1.73 m/s is a reasonable maximal value as humans start to jog upwards of 1.8 m/s [71]. Jogging occurring would require a different dynamic formulation in phase divisions. The minimum speed could be a construct of the dynamic formulation and lower speeds could be required for real-world implementation. The range of comfort was over the full range of the deadzone as expected, however, more exploration could be allowed in the future. The range of effort proxy cost was sufficient in providing significant change. The range of natural walking cost does not include an error of zero which means that the system still could use natural walking improvement, perhaps through frontal plane actuation of the hips. The range of each parameter suggests that this amount of adjustment would allow meaningful tuning of the exoskeleton controller.

## 5.2 Sensitivity

The quantity and spread of points within the range are important for allowing user adjustment. Each parameter allowed for a number of different unique values for which the controller can be adjusted. The areas which showed limited feasible solutions are an interesting point to discuss as these regions show an area of potential limitation in the stable bounds, perhaps more exploration could be done in these regions to establish further the limitations of the model. This could be solved by allowing the algorithm to run longer, or by adding more exploration to the algorithm. The spread found suggests a user would be able to adjust the control parameters in a meaningful way across the range of parameters, however, more points would allow for a finer selection to meet a user's need.

## 5.3 Prioritizing Performance Measures

Allowing the user to prioritize input metrics over other metrics gives the user intuitive tuning. The success of achieving a meaningful difference in joint trajectories based on prioritizing different gait profiles suggests the validity of the desired optimization outcome. Comparing the four different priorities it can be seen that while prioritizing for comfort the system walked at the lowest speed of 1.1348 m/s and took the fewest number of steps over the same interval at 13. Observing the instance when speed is prioritized the system took the most steps at 22. This shows a distinctive trade-off between comfort and speed. When the natural walking was prioritized the model walked at a rate of 1.5472 m/s which is higher than a fast average walker. This extra speed could be an artifact of a lack of real-world damping in the model. When prioritizing for effort proxy the model walked at a comfortable 1.336 m/s which is within the range of natural average walking speeds.

The walking speed is slower than when the model prioritizes following natural walking curves which could provide some insight on the validity of human walking being primarily an energy optimizing task. The prioritizing of parameters gives significant differences in the output of the walking trajectories using the virtual constraint framework.

#### **5.4 Tracking Error**

Tracking error is an important concept to observe in most control systems. The tracking error is not so much a concern in this case as the cost functions are mostly output parameters rather than input parameters. The controller output torques were saturated, which restricts the user from being injured which means that the trajectories themselves may not be strictly followed, leading to tracking error. Further restrictions on the trajectory derivatives could be implemented to ensure that the trajectories do not have a change in angle that is too large for the human to withstand. The controller maps the input parameters, the optimized trajectories, to the output parameters, the cost functions. The tracking error is of importance when considering the inputs of the problem as the bio-feasibility constraints and biomimetic cost were placed on the inputs. For instance, the output of the bipedal system could be much the same for two different sets of input trajectories if there is significant tracking error for one or both of the otherwise distinct trajectories. This error potentially reduces the range of results that were produced which satisfy the Poincaré stability output constraint used. In the future, all cost functions being output metrics should be implemented. Although the range of the biomimicry term was found to be significant, there could be less real variability in the output trajectories achieved. Tracking error is also

a consideration when implementing on a physical device as the mechanical system will differ from the simulation and adjustment of controller gains may be necessary.

## 6 Conclusion

The course of this thesis determined that a flexible control providing a user-adjustable control for a lower limb exoskeleton based on multiple gait performances is a feasible goal. A variety of stable gait input trajectories were formulated across different gait performance measure values through multi-objective optimization. A lookup table of these gait trajectories was reformulated into a user interface to allow for user intuitive control of an exoskeleton. It has been observed that a virtual constraint exoskeleton framework allows for flexibility thus allowing for the novel multi-objective optimization across different gait performance measures. The user-adjustable control provides users and clinicians with the adjustment of an exoskeleton in an intuitive manner based on what each user cares about. Allowing more user interaction through this framework hopefully will increase user satisfaction, user acceptance, and long-term use of exoskeleton devices. This thesis presents a concept that can be implemented on a physical device to provide meaningful real-world adjustment of an exoskeleton controller.

### 6.1 Limitations

There are several limitations that are worth noting with respect to the findings in this thesis. As is the case with any simulation, the model is limited by the assumptions and limitations of the computational model. The gait-phase division, the feedback linearized controller and the selection of the phase variable are all limitations that influence the results obtained. The assumptions of the spring constant accurately modelling the floor, the rigid coupling of the joints in the Lagrangian dynamic formulation, and ideal actuators for providing the control torques are some of them. Other methods could be used for modelling

including a spring-damper floor, Newtonian formulation, and a model with real-world motor characteristics that could be used to corroborate the current findings. These assumptions have been shown to be accurate enough assumptions in modelling the real-world and were used with the knowledge of these potential limitations. The fixed model parameters limit the conclusions as to how robust this controller will be when used for different individuals. The low-level gains of the feedback controllers limit the range of stability that could be achieved using the feedback linearized method. Due to actuation only being done in the sagittal plane, there is limited frontal plane stability which would require a walking aid for most SCI users. Winter's data used for biomimicry is not representative of the whole population as it is only representative of average healthy males walking at a moderate speed. These limitations are worth noting when implementing this controller on a physical device.

## **6.2 Future Work**

There is always the option for future work with novel work in a research field. The methodology of providing user tunable control based on performance metrics users care about is a concept shown to be worthy of a patent, thus making further development of the concept of interest to the patent holders. There are different areas that future work can be done to build on the concepts developed in this thesis including the optimization algorithm used, real-world implementation, and different controller features.

### **6.2.1 Algorithms and Optimization**

The area of optimization holds opportunities for future work. Pursuing other algorithms to see how they perform in solving this multi-objective optimization problem

will be worthwhile in providing more insight into the solution space. Further results will be able to better map the search space and have more sensitivity in the parameter tuning. Further exploration with the NSGA-II can also be done to potentially improve the results by adjusting algorithmic parameters. Exploration using other algorithms such as SMS-EMOA and MOEA/D would be interesting for comparison to the NSGA-II to decide which algorithms provide the best results.

A multi-objective method that would be useful for refining the results is a Pareto filter. A Pareto filter filters a Pareto front to meet specified conditions. To further improve the results obtained in providing a user-intuitive interface with a uniform distributed Pareto filter, such as developed by Messac et al. [72], [73], could be used. Additional Pareto filters could be used to study the best manner to provide intuitive control.

Future work can improve the optimization problem formulation. Psychophysical studies could be done to determine the best cost functions for each parameter, as well as which parameters are most important to the user population. Adding online optimal control onto the controller could be done to allow for improved online trajectory adaptation and dynamic stability control.

### **6.2.2 Real-world Implementation**

The most important next step in validating this research would be to implement this controller on an exoskeleton. Once implemented, a case-study could be conducted to validate the real-world implementation. Upon system validation, user testing could be performed on multiple users to see how user acceptance compares to rigid trajectory enforcing. Once physical implementation has been established then trials can be conducted

on target populations to see how they enjoy it. Comparison studies could be conducted in relation to other assistive exoskeletons to observe how this user interface compares to other exoskeletons once implemented on a physical device.

### **6.2.3 Control Exploration**

There are many controller add-ons that would improve the system to be more comprehensive. Providing meaningful extrapolation of this concept from flat ground walking to other ADLs such as walking, jogging, and running states is a worthwhile goal. Adding some online perturbation handling would be important for individuals with muscle syndromes, such as spasticity. A speed adaption mechanism such as the one developed by Lenzi and Sensinger [44] could be added to change between speeds online as well as a finite state machine to transition between states. Frontal plane actuation could be done to alleviate the need for mobility aids, such as a walker. Further study can also be done in relaxing the Lyapunov chaotic margin of stability to establish the amount of chaotic behaviour the current feedback linearized controller gains allows for stable walking to allow for more potential gait trajectories over a wider range of performance measure values. These and other control add-ons could be implemented to build on the user-interface control.

### **6.3 Final Remarks**

The development of a look-up table that contains a spread of different walking trajectories based on different user performance measures has been observed. The framework allows user-tunable control of an exoskeleton. Future work can demonstrate the effectiveness of providing this control to users. This user-in-control operation allows for

more user engagement and leads to the possibility of increasing the continual use of these assistive devices. This novel contribution to the assistive exoskeleton control field will hopefully impact those who live with disabilities and give them more opportunities and a better quality of life.

## Bibliography

- [1] V. K. Noonan *et al.*, “Incidence and prevalence of spinal cord injury in Canada: A national perspective,” *Neuroepidemiology*, vol. 38, no. 4, pp. 219–226, 2012.
- [2] H. Gilmour, P. L. Ramage-Morin, and S. L. Wong, “Multiple sclerosis: Prevalence and impact,” *Heal. Reports*, vol. 29, no. 1, pp. 3–8, 2018.
- [3] L. A. Simpson, J. J. Eng, J. T. C. Hsieh, and D. L. Wolfe and the Spinal Cord Injury Re, “The Health and Life Priorities of Individuals with Spinal Cord Injury: A Systematic Review,” *J. Neurotrauma*, vol. 29, no. 8, pp. 1548–1555, 2012.
- [4] A. S. Gorgey, “Robotic exoskeletons: The current pros and cons,” *World J. Orthop.*, vol. 9, no. 9, pp. 112–119, 2018.
- [5] D. P. Ferris, “The exoskeletons are here,” *J. Neuroeng. Rehabil.*, vol. 6, no. 1, p. 17, 2009.
- [6] A. M. Dollar and H. Herr, “Lower Extremity Exoskeletons and Active Orthoses: Challenges and State-of-the-Art,” *IEEE Trans. Robot.*, vol. 24, no. 1, pp. 144–158, 2008.
- [7] T. Yan, M. Cempini, C. M. Oddo, and N. Vitiello, “Review of assistive strategies in powered lower-limb orthoses and exoskeletons,” *Rob. Auton. Syst.*, vol. 64, pp. 120–136, 2015.
- [8] D. P. FERRIS, G. S. SAWICKI, and M. A. DALEY, “a Physiologist’S Perspective on Robotic Exoskeletons for Human Locomotion,” *Int. J. Humanoid Robot.*, vol. 04, no. 03, pp. 507–528, 2007.

- [9] R. Ronsse, N. Vitiello, T. Lenzi, J. Van Den Kieboom, M. C. Carrozza, and A. J. Ijspeert, "Human-Robot synchrony: Flexible assistance using adaptive oscillators," *IEEE Trans. Biomed. Eng.*, vol. 58, no. 4, pp. 1001–1012, 2011.
- [10] J. Zhang *et al.*, "Human-in-the-loop optimization of exoskeleton assistance during walking," *Sci. Mag.*, vol. 1284, no. June, pp. 1280–1284, 2017.
- [11] E. Adams, "Power-multiplying exoskeletons are slimming down for use on the battlefield," *Popular Science*, 2018. [Online]. Available: <https://www.popsci.com/army-exoskeletons-lockheed-martin/>. [Accessed: 06-Dec-2019].
- [12] C. INC, "Cyberdyne Homepage," *Cyberdyne.jp*, 2019. [Online]. Available: <https://www.cyberdyne.jp/english/>. [Accessed: 06-Dec-2019].
- [13] Andra Keay, "Ekso Bionics goes public for \$ 20 . 6 million," *Robohub*, 2019. [Online]. Available: <https://robohub.org/eksobionics-goes-public-for-20-6-million/>. [Accessed: 06-Dec-2019].
- [14] J. Wolff, C. Parker, J. Borisoff, W. Mortenson, and J. Mattie, "A survey of stakeholder perspectives on exoskeleton technology," *J. Neuroeng. Rehabil.*, vol. 11, no. 1, p. 169, 2014.
- [15] A. Esquenazi, M. Talaty, and A. Jayaraman, "Powered Exoskeletons for Walking Assistance in Persons with Central Nervous System Injuries: A Narrative Review," *Phys. Med. Rehabil.*, vol. 9, no. 1, pp. 46–62, 2017.
- [16] M. Folgheraiter, M. Jordan, S. Straube, A. Seeland, S. K. Kim, and E. A. Kirchner, "Measuring the Improvement of the Interaction Comfort of a Wearable Exoskeleton: A Multi-Modal Control Mechanism Based on Force Measurement

- and Movement Prediction,” *Int. J. Soc. Robot.*, vol. 4, no. 3, pp. 285–302, 2012.
- [17] W. Felt, J. C. Selinger, J. M. Donelan, and C. D. Remy, “‘Body-in-the-loop’: Optimizing device parameters using measures of instantaneous energetic cost,” *PLoS One*, vol. 10, no. 8, pp. 1–21, 2015.
- [18] H. a Quintero, R. J. Farris, and M. G. Members, “Control and Implementation of a Powered Lower Limb Orthosis to Aid Walking in Paraplegic Individuals,” vol. 19, no. 6, pp. 652–659, 2012.
- [19] M. Sczesny-Kaiser *et al.*, “HAL® exoskeleton training improves walking parameters and normalizes cortical excitability in primary somatosensory cortex in spinal cord injury patients,” *J. Neuroeng. Rehabil.*, vol. 12, no. 1, p. 68, 2015.
- [20] H. Herr, “Exoskeletons and orthoses: classification, design challenges and future directions,” *J. Neuroeng. Rehabil.*, vol. 6, no. 1, p. 21, 2009.
- [21] A. D. Kuo and J. M. Donelan, “Dynamic Principles of Gait and Their Clinical Implications,” *Phys. Ther.*, vol. 90, no. 2, pp. 157–174, 2010.
- [22] C. Tunca, N. Pehlivan, N. Ak, B. Arnrich, G. Salur, and C. Ersoy, “Inertial sensor-based robust gait analysis in non-hospital settings for neurological disorders,” *Sensors (Switzerland)*, vol. 17, no. 4, pp. 1–29, 2017.
- [23] O. Harib *et al.*, “Feedback Control of an Exoskeleton for Paraplegics,” pp. 1–23.
- [24] V. Lajeunesse, C. Vincent, F. Routhier, E. Careau, and F. Michaud, “Exoskeletons’ design and usefulness evidence according to a systematic review of lower limb exoskeletons used for functional mobility by people with spinal cord injury,” *Disabil. Rehabil. Assist. Technol.*, vol. 11, no. 7, pp. 535–547, 2016.
- [25] K. A. Strausser and H. Kazerooni, “The development and testing of a human

- machine interface for a mobile medical exoskeleton,” *2011 IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, pp. 4911–4916, 2011.
- [26] M. R. Tucker *et al.*, “Control strategies for active lower extremity prosthetics and orthotics: A review,” *J. Neuroeng. Rehabil.*, vol. 12, no. 1, 2015.
- [27] A. Esquenazi, M. Talaty, A. Packel, and M. Saulino, “The ReWalk Powered Exoskeleton to Restore Ambulatory Function to Individuals with Thoracic-Level Motor-Complete Spinal Cord Injury,” *Am. J. Phys. Med. Rehabil.*, vol. 91, no. 11, pp. 911–921, 2012.
- [28] P. D. Neuhaus, J. H. Noorden, T. J. Craig, T. Torres, J. Kirschbaum, and J. E. Pratt, “Design and evaluation of Mina: A robotic orthosis for paraplegics,” *IEEE Int. Conf. Rehabil. Robot.*, 2011.
- [29] S. Murray and M. Goldfarb, “Towards the use of a lower limb exoskeleton for locomotion assistance in individuals with neuromuscular locomotor deficits,” *Proc. Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. EMBS*, pp. 1912–1915, 2012.
- [30] N. Aliman, R. Ramli, and S. M. Haris, “Design and development of lower limb exoskeletons: A survey,” *Rob. Auton. Syst.*, vol. 95, pp. 102–116, 2017.
- [31] S. Jezernik, G. Colombo, T. Keller, H. Frueh, and M. Morari, “Robotic Orthosis Lokomat: A Rehabilitation and Research Tool,” *Neuromodulation*, vol. 6, no. 2, pp. 108–115, 2003.
- [32] J. F. Veneman, R. Kruidhof, E. E. G. Hekman, R. Ekkelenkamp, E. H. F. Van Asseldonk, and H. Van Der Kooij, “Design and evaluation of the LOPES exoskeleton robot for interactive gait rehabilitation,” *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 15, no. 1, pp. 379–386, 2007.

- [33] H. Kawamoto and Y. Sankai, "Power assist method based on Phase Sequence and muscle force condition for HAL," *Adv. Robot.*, vol. 19, no. 7, pp. 717–734, 2005.
- [34] D. Hebert *et al.*, "Canadian stroke best practice recommendations: Stroke rehabilitation practice guidelines, update 2015," *Int. J. Stroke*, vol. 11, no. 4, pp. 459–484, 2016.
- [35] L. Marchal-Crespo and D. J. Reinkensmeyer, "Review of control strategies for robotic movement training after neurologic injury," *J. Neuroeng. Rehabil.*, vol. 6, no. 1, p. 20, 2009.
- [36] J. Cao, S. Q. Xie, R. Das, and G. L. Zhu, "Control strategies for effective robot assisted gait rehabilitation: The state of art and future prospects," *Med. Eng. Phys.*, vol. 36, no. 12, pp. 1555–1566, 2014.
- [37] M. Fahmi, B. Miskon, M. Bin, and A. Jalil, "Review of Trajectory Generation of Exoskeleton Robots," pp. 12–17.
- [38] K. Kamali, A. A. Akbari, and A. Akbarzadeh, "Trajectory generation and control of a knee exoskeleton based on dynamic movement primitives for sit-to-stand assistance," *Adv. Robot.*, vol. 30, no. 13, pp. 846–860, 2016.
- [39] M. D. C. Sanchez-Villamañan, J. Gonzalez-Vargas, D. Torricelli, J. C. Moreno, and J. L. Pons, "Compliant lower limb exoskeletons: A comprehensive review on mechanical design principles," *J. Neuroeng. Rehabil.*, vol. 16, no. 1, pp. 1–16, 2019.
- [40] J. W. Grizzle, "Virtual Constraints and Hybrid Zero Dynamics for Realizing Underactuated Bipedal Locomotion."
- [41] R. D. Gregg, T. Lenzi, L. J. Hargrove, and J. W. Sensinger, "Virtual constraint

- control of a powered prosthetic leg: From simulation to experiments with transfemoral amputees,” *IEEE Trans. Robot.*, vol. 30, no. 6, pp. 1455–1471, 2014.
- [42] B. C. Chevallereau *et al.*, “RABBIT : A Testbed for Advanced Control Theory,” *IEEE Control Syst. Mag.*, vol. 23, no. 5, pp. 57–79, 2003.
- [43] J.-J. Slotine and W. Li, *Applied Nonlinear Control*. Upper Saddle River, New Jersey: Prentice - Hall, 1991.
- [44] T. Lenzi, L. J. Hargrove, and J. W. Sensinger, “Preliminary evaluation of a new control approach to achieve speed adaptation in robotic transfemoral prostheses,” *IEEE Int. Conf. Intell. Robot. Syst.*, no. Iros, pp. 2049–2054, 2014.
- [45] R. D. Gregg and J. W. Sensinger, “Towards biomimetic virtual constraint control of a powered prosthetic leg,” *IEEE Trans. Control Syst. Technol.*, vol. 22, no. 1, pp. 246–254, 2014.
- [46] H. F. N. Al-Shuka, B. J. Corves, and W.-H. Zhu, “Dynamic Modeling of Biped Robot using Lagrangian and Recursive Newton-Euler Formulations,” *Int. J. Comput. Appl.*, vol. 101, no. 3, pp. 1–8, 2014.
- [47] E. R. Westervelt, S. Member, J. W. Grizzle, D. E. Koditschek, and S. Member, “Hybrid Zero Dynamics of Planar Biped Walkers,” vol. 48, no. 1, pp. 42–56, 2003.
- [48] S. M. Bruijn, O. G. Meijer, P. J. Beek, and J. H. Van Dieen, “Assessing the stability of human locomotion: A review of current measures,” *J. R. Soc. Interface*, vol. 10, no. 83, 2013.
- [49] Y. Hurmuzlu and C. Basdogan, “On the measurement of dynamic stability of human locomotion,” *J. Biomech. Eng.*, vol. 116, no. 1, pp. 30–36, 1994.
- [50] B. Morris and J. W. Grizzle, “A restricted poincaré map for determining

- exponentially stable periodic orbits in systems with impulse effects: Application to bipedal robots,” *Proc. 44th IEEE Conf. Decis. Control. Eur. Control Conf. CDC-ECC '05*, vol. 2005, pp. 4199–4206, 2005.
- [51] M. Garcia, a Chatterjee, a Ruina, and M. Coleman, “The simplest walking model: stability, complexity, and scaling.,” *J. Biomech. Eng.*, vol. 120, no. 2, pp. 281–288, 1998.
- [52] R. Shadmehr and S. Mussa-Ivaldi, *Biological Learning and Control*, no. 1. The MIT Press, 2012.
- [53] A. Eric R. Westervelt, Jessy W. Grizzle, Christine Chevallereau, JunHoChoi, “Feedback Control of Dynamic Bipedal Robot Locomotion,” 2007.
- [54] K. Deb and K. Deb, “Multi-objective Optimization,” in *Search Methodologies: Introductory Tutorials in Optimization and Decision Support Techniques*, E. K. Burke and G. Kendall, Eds. Boston, MA: Springer US, 2014, pp. 403–449.
- [55] G. Capi, S. Kaneko, K. Mitobe, L. Barolli, and Y. Nasu, “Optimal trajectory generation for a prismatic joint biped robot using genetic algorithms,” *Rob. Auton. Syst.*, vol. 38, no. 2, pp. 119–128, 2002.
- [56] D. P. Garg and M. Kumar, “Optimization techniques applied to multiple manipulators for path planning and torque minimization,” *Eng. Appl. Artif. Intell.*, vol. 15, no. 3–4, pp. 241–252, 2002.
- [57] D. Gong, J. Yan, and G. Zuo, “A Review of Gait Optimization Based on Evolutionary Computation,” *Appl. Comput. Intell. Soft Comput.*, vol. 2010, pp. 1–12, 2010.
- [58] N. Shafii, S. Aslani, O. M. Nezami, and S. Shiry, “Evolution of Biped Walking

- Using Truncated Fourier Series and Particle Swarm Optimization,” *Computer (Long. Beach. Calif.)*, vol. 5949, pp. 344–354, 2010.
- [59] K. Stewart, C. Diduch, and J. Sensinger, “Assistive Exoskeleton Control with User-Tuned Multi-Objective Optimization,” *2019 IEEE 16th Int. Conf. Rehabil. Robot.*, pp. 554–558, 2019.
- [60] H. Mausser, “Normalization and therTopics in Multi- Objective Optimization.pdf,” *Proc. fields-MITACS Ind. Probl. Work.*, vol. 2, pp. 89–101, 2006.
- [61] R. T. Marler and J. S. Arora, “Survey of multi-objective optimization methods for engineering,” *Struct. Multidiscip. Optim.*, vol. 26, no. 6, pp. 369–395, 2004.
- [62] M. T. M. Emmerich and A. H. Deutz, “A tutorial on multiobjective optimization: fundamentals and evolutionary methods,” *Nat. Comput.*, vol. 17, no. 3, pp. 585–609, 2018.
- [63] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, “A Fast and Elitist Multiobjective Genetic Algorithm :,” vol. 6, no. 2, pp. 182–197, 2002.
- [64] S. M. Campbell, C. Diduch, and J. W. Sensinger, “Autonomous Assistance-as-Needed Control of a Lower Limb Exoskeleton with Guaranteed Stability,” *IEEE Access, Submitt.*, vol. XX, pp. 1–12, 2019.
- [65] X. Mu, “Dynamics and motion regulation of a five-linked biped robot walking in the sagittal plane,” *Tesis Dr. Univ. Manitoba*, vol. NQ97295, pp. 153-153 p., 2005.
- [66] M. P. Mcgrath, “Appropriately Complex Modelling of Healthy Human Walking Submitted in Partial Fulfilment of the Requirements of the Degree of Doctor of Philosophy CONTENTS,” 2014.
- [67] J. K. Moore, S. K. Hnat, and A. J. Van Den Bogert, “An elaborate data set on

- human gait and the effect of mechanical perturbations,” 2015.
- [68] D. Winter, *Biomechanics and Motor Control of Human Movement*, 4th ed. Waterloo, ON, Canada: Wiley, 2009.
- [69] R. W. Bohannon and A. Williams Andrews, “Normal walking speed: A descriptive meta-analysis,” *Physiotherapy*, vol. 97, no. 3, pp. 182–189, 2011.
- [70] T. Manning, R. D. Sleator, and P. Walsh, “Naturally selecting solutions: the use of genetic algorithms in bioinformatics,” *Bioengineered*, vol. 4, no. 5, pp. 266–278, 2013.
- [71] K. Jordan, J. H. Challis, and K. M. Newell, “Walking speed influences on gait cycle variability,” *Gait Posture*, vol. 26, no. 1, pp. 128–134, 2007.
- [72] C. A. Mattson, A. A. Mullur, and A. Messac, “Smart pareto filter: Obtaining a minimal representation of multiobjective design space,” *Eng. Optim.*, vol. 36, no. 6, pp. 721–740, 2004.
- [73] A. Messac, A. Ismail-Yahaya, and C. A. Mattson, “The normalized normal constraint method for generating the Pareto frontier,” *Struct. Multidiscip. Optim.*, vol. 25, no. 2, pp. 86–98, 2003.

## Appendix A: Source Code

[https://gitlab.com/kurt\\_stewart/thesis-code-remote](https://gitlab.com/kurt_stewart/thesis-code-remote)

For access please request [kstewar4@unb.ca](mailto:kstewar4@unb.ca) – Subject title “Request Thesis Code Access”

- Run\_Thesis\_Runtime – Runs Code
- Simulink\_New – Exoskeleton Model
- /Final Results/ - The Final Results and reformulation code

# Curriculum Vitae

Candidate's full name: Kurt Saul Thomas Stewart

Universities attended (with dates and degrees obtained): UNB BSc.EE 2017

Publications: *Assistive Exoskeleton Control with User-Tuned Multi-Objective Optimization* at 2019 IEEE International Conference on Rehabilitation Robotics

Conference Presentations: *Assistive Exoskeleton Control with User-Tuned Multi-Objective Optimization* Poster Presentation at 2019 IEEE International Conference on Rehabilitation Robotics