

நல்லதோர் வீணை செய்தே – அதை  
நலங்கெடப் புழுதியில் எறிவதுண்டோ?  
சொல்லடி சிவசக்தி – எனைச்  
சுடர்மிகும் அறிவுடன் படைத்துவிட்டாய்,  
வல்லமை தாராயோ – இந்த  
மாநிலம் பயனுற வாழ்வதற்கே !  
சொல்லடி சிவசக்தி – நிலச்  
சுமையென வாழ்ந்திடப் புரிகுவையோ?

விசையுறு பந்தினைப்போல் — உள்ளம்  
வேண்டிய படிசெலும் உடல்கேட்டேன்,  
நசையறு மனங்கேட்டேன், — நித்தம்  
நவமெனச் சுடர்தரும் உயிர்கேட்டேன்,  
தசையினைத் தீச்சுடினும் — சிவ  
சக்தியைப் பாடும்நல் அகங்கேட்டேன்,  
அசைவறு மதிகேட்டேன்; — இவை  
அருள்வதில் உனக்கெதுந் தடையுளதோ?

– மகாகவி பாரதியார்

Multi-Objective User Priority-Based Optimal Tuning of Myoelectric Prostheses

by

Anjana Gayathri Arunachalam

Bachelor of Technology (Honours) in Instrumentation and Control Engineering,

National Institute of Technology Tiruchirappalli, 2017

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

Master of Science in Engineering

in the Graduate Academic Unit of Electrical and Computer Engineering

**Supervisors:** Jonathon Sensinger, PhD, Electrical and Computer Engineering

Kevin Englehart, PhD, Electrical and Computer Engineering

**Examining Board:** Erik Scheme, PhD, Electrical and Computer Engineering

Chris Diduch, PhD, Electrical and Computer Engineering

Juan Antonio Carretero, PhD, Mechanical Engineering

This thesis is accepted by the  
Dean of Graduate Studies

THE UNIVERSITY OF NEW BRUNSWICK

August, 2024

©Anjana Gayathri Arunachalam, 2024

## **ABSTRACT**

Current myoelectric prosthesis control methods lack personalized tuning based on individual user preferences leading to potentially suboptimal user performance. This thesis introduces a model for personalized myoelectric prosthetic control, integrating user preferences into device dynamics through an optimization approach. Drawing on principles of computational motor control, evolutionary multi-objective optimization, and control theory, the model identifies optimal device dynamic parameters based on user preferences for effort, movement time, and reliability. Results from our simulated prosthesis model suggest that customized prosthetic devices could appreciably improve movement outcomes compared to conventional devices. This study provides a foundation for intuitive and effective prosthetic device control. These improvements may have potential applications beyond prosthesis including various human-machine interfaces.

## **DEDICATION**

For Amma, thank you for being my pillar of support and strength.

To Athul and Aadhi, your smiles can bring warmth and light to even the darkest days.

I sincerely hope the world reciprocates your kindness.

## **ACKNOWLEDGEMENTS**

This document would not have been possible without the support and encouragement of many people. I want to thank my supervisors, Dr. Jon Sensinger and Dr. Kevin Englehart, for their time and profound impact on my life. Thanks to Mitacs for introducing me to UNB and encouraging me to return. Many thanks to Dr. Umapathy from NITT for encouraging me to pursue a graduate degree.

Nathan Quinn and Christian Grandmaison, your positivity and mentorship during my first summer at UNB were invaluable. Dr. Biden, thank you for genuinely caring about students and the memorable Beers with Biden sessions. Kristel, your check-ins, kind words, and cheerful banter always brightened my days. Dr. Erik Scheme, thank you for stopping by and saying hi whenever you could. Thanks to Dr Juan Carretero and Dom Blakely for some of the most fun and thought-provoking graduate courses I have taken at UNB. Adam Wilson, thank you for being an incredible mentor and believing in me even when I struggled to believe in myself. Dr. Petersen, working as your teaching assistant has been a pleasure, and I've learned so much from you. Becky, Leslie, and Matt from the SUB office, thank you for always finding me a workspace during the pandemic and for your constant encouragement and care.

I have been fortunate to make many genuine friends throughout this journey. Neil and Danielle, thank you for always lending a hand and treating me like family. Ish, your support from day one has been incredible; I honestly can't thank you enough for the brainstorming sessions, market adventures, regular check-ins, and always being there when I needed help. Kurt, thank you for introducing me to the mountains. Michelle, you were the best work

buddy during the pandemic; our city walks, optimization talks, and your genuine care meant so much. Special thanks to Chitom for being my friend and mom-away-from-home, always offering a supportive shoulder and an open door. Shri, thank you for the walks, whiteboard sessions, Python lessons, sanity checks, hugs, calls, and more. Ahmed, I am grateful for your patience, positive thoughts, and guidance - they have been invaluable. Ramin, I couldn't have asked for a better co-TA, thanks for being a good friend.

I am also deeply grateful for my friends outside of UNB. Guru, Ankita, Mike, Anu, Trupti(s), Siddarth, Prathamesh, and Akshit, your support helped me feel at home and kept homesickness at bay. Special thanks to Akshit for being an amazing listener and for giving me a safe space whenever I needed it. Cindy Howie, thank you for treating me like your daughter and for our walks with Jack - they were a source of comfort. Terry Brodie, your encouragement inside and outside the band room has been wonderful. Aarthi, Anand, Vishnu, Aditya, and Sriharish, thank you for always having my back through the years and for all the calls and words of comfort. Madhu, your love, care, and patience during my moments of panic have been a lifeline.

Finally, to my family: Amma and Appa, thank you for your unwavering love. Koushik and Sandhya manni, I am grateful for your patience and support, which have helped me stay grounded. Thank you for always being just a call away. Special thanks to Athullya and Aadhya for their stories, songs, dances, and laughter, which kept me going. Paru chitti, Gopu chitappa, and Sri mama, thank you for always keeping me in your prayers. And Lucky, thank you for the beautiful memories.

## Table of Contents

ABSTRACT.....	ii
DEDICATION.....	iii
ACKNOWLEDGEMENTS.....	iv
Table of Contents.....	vi
List of Tables.....	x
List of Figures.....	xi
List of Abbreviations.....	xiv
List of Symbols and Nomenclature.....	xv
1. Chapter 1: Introduction.....	1
1.1 Specific aims.....	2
1.2 Significance.....	3
1.3 Innovation.....	4
1.4 Overview/ Thesis structure.....	4
2. Chapter 2: Background information and literature review.....	5
2.1 Modelling human movement using computational motor control tools.....	7
2.1.1 Internal models to describe motor control.....	8

2.1.2 Cost functions in biological systems.....	9
2.2 Multi-objective optimization .....	11
2.2.1 Terms and definitions .....	11
2.2.2 Classification of MOO.....	17
2.2.3 Linear scalarization.....	19
2.2.4 Goals of MOO.....	20
2.3 Device dynamics and virtual dynamics .....	21
3. Chapter 3: Methods.....	23
3.1 Incorporating multiple objectives for motor control.....	26
3.1.1 Costs considered for the task of interest .....	26
3.1.2 Combined cost function .....	29
3.2 Human part of the HMI model.....	33
3.2.1 Model of the user intent or the neural drive.....	33
3.2.2 Model of surface EMG signal corresponding to the control intent .....	34
3.3 Device part of the HMI model: virtual dynamics .....	36
3.4 Structure of the full HMI model .....	38
3.4.1 Hierarchical optimization.....	38
3.4.2 Modelled task.....	40
3.4.3 Cost calculation for each evaluation .....	41

3.5 Optimization within the HMI interaction model .....	43
3.5.1 Particle swarm optimization algorithm (PSO).....	44
3.5.2 Hierarchical implementation of the PSO algorithm.....	46
3.5.3 Special parameter implementations within the PSO.....	47
3.5.4 Hyperparameters of the algorithm .....	50
4. Chapter 4: Results .....	52
4.1 Influence of the user preference level on the generated motor commands.....	52
4.2 Effect of tuning both the human and the device parts of the HMI interaction model according to the user preference level .....	56
4.3 The importance of tunable device dynamics .....	62
5. Chapter 5: Discussion .....	66
5.1 Verification of specific aim 1 .....	66
5.2 Verification of specific aim 2 .....	68
5.3 Significance.....	69
5.4 Limitations .....	70
5.5 Future work.....	72
6. Chapter 6: Conclusions .....	74
Bibliography .....	75
Appendix A: Source Code .....	90

Appendix B: Additional terms and definitions for MOO .....	91
Appendix C: Hierarchical PSO implementation details .....	96
Appendix D: Result data and additional plots .....	100
Curriculum Vitae	

## List of Tables

Table 1: Solutions points for sample MOO minimization problem from Equation (2.1).	16
Table 2: List of tested user preference conditions and their corresponding user preference values. ....	32
Table 3: Hyperparameter values of the algorithm .....	51
Table 4: Lower and upper limits of the optimization variables .....	99
Table 5: Optimized costs and virtual dynamics for different relative weights .....	100

## List of Figures

Figure 1: Objective space visualization of the Simple MOO example (Eq 2.1). The dashed blue line corresponds to the objective space representation of the points displayed in Figure 2. The solid red line highlights the hyperplane created by connecting all the non-dominated solutions in the objective space (Pareto front). The solutions from Table 1 are labelled accordingly. ....	15
Figure 2: Comparison of the two objective function values, $f_1$ (dashed blue) and $f_2$ (dotted red) for a range of design variable values. The dotted vertical lines indicate the range of $x$ values that belong to the Pareto optimal solution set. ....	17
Figure 3: Simplified block diagram representation of the User-Prosthesis interaction model.....	25
Figure 4: Example of the simulated control signal and corresponding EMG in the time domain.....	34
Figure 5: Comparison of the power spectral densities of Clancy's EMG model and the proposed EMG model.....	36
Figure 6: Schematic model of human interacting with a myoelectric prosthesis to perform a reaching task.....	39
Figure 7: Schematic of the modelled task. Each trial of the task consists of reaching two targets.....	40
Figure 8: Block diagram indicating how the total cost is calculated within the simulation. The green lines indicate the input (user preference vector, simulated neural drive parameters, and simulated VD parameters) and the red lines indicate the output (total cost). ....	43

Figure 9: Understanding the influence of the user preference level on the human part of the HMI interaction model – through the generated control signal and the device trajectory for a device with fixed VD. The top row indicates the produced optimal user control signals (on the left) and simulated device trajectories (on the right) for the first target. The bottom row shows the produced optimal user control signal (on the left) and the corresponding device trajectory for the second target. The dotted lines on the second column of panels indicate the desired target position. .... 53

Figure 10: Understanding the influence of the user preference level on the human part of the HMI interaction model – through the generated control signal and the device trajectory for a device with fixed VD. The time prioritized setting for the 10 unit target reach is omitted as the results of the optimization failed to converge. .... 55

Figure 11: Optimal control signal and device trajectories when the device dynamics is optimized according to the user preference level. The top row indicates the produced optimal user control signals (on the left) and simulated device trajectories (on the right) for the first target. The bottom row shows the produced optimal user control signal (on the left) and the corresponding device trajectory for the second target. .... 58

Figure 12: Normalized spider plots indicating the individual costs of effort, time and reliability for the 4 user priority settings that were simulated. The image on the left corresponds to the costs to reach the first target, and the image on the right corresponds to the costs to reach the second target. .... 59

Figure 13: In the clockwise direction from the top left-hand corner, each subplot shows the optimal VD values of the ( $a1$ ,  $a2$ ,  $b2$ ,  $b1$ ) parameters for each of the simulated user preference levels. .... 61

Figure 14: Comparing the control signals and device trajectories when using a device with non-tunable proportional velocity control mapping (in blue) and a tunable device dynamic mapping (in red) to reach the second target that is 15 units from the origin. .... 63

Figure 15: Comparison of the total movement cost at different user preference levels between the tunable and the non-tunable control dynamics. .... 64

Figure 16: Pareto space visualization of the individual costs comparing the tunable device dynamics with the conventional velocity control device dynamics..... 65

Figure 17: (From left to right) Images of a convex function, convex set, and non-convex set [106], [107]..... 92

Figure 18: Depiction of the ideal point, the nadir point, and the worst point for a general two-objective multi-objective optimization on problem. This image is inspired by and modified from the work of Deb *et al.* [109]..... 94

Figure 19: Algorithmic flowchart of the Particle Swarm Optimization Algorithm..... 97

Figure 20: Comparing the control signals and device trajectories when using a device with non-tunable proportional velocity control mapping (in blue) and a tunable device dynamic mapping (in red) to reach the first target, which is 10 units from the origin. .... 101

## List of Abbreviations

HMI	Human Machine Interface
MOO	Multi-Objective Optimization
EMG	Electromyography
SOO	Single-Objective Optimization
DOF	Degree Of Freedom
CMC	Computational Motor Control
VD	Virtual Dynamics
WSO	Weighted Sum Optimization

## List of Symbols and Nomenclature

$J_t$	:	Cost of time
$\beta$	:	Temporal discounting factor
$p$	:	Total movement time
$J_u$	:	Cost of effort
$u(t)$	:	Control signal magnitude at timestep $t$
$\underline{u}$	:	Control signal trajectory
$J_r$	:	Cost of reliability
$E[ ]$	:	The expected value operator
$g$	:	The target position or the goal of the task
$x(t)$	:	End effector position at timestep $t$
$\underline{x}$	:	End effector trajectory
$\dot{x}(t)$	:	End effector velocity at timestep $t$
$C_1, C_2, C_3$	:	Constraint functions
$J_{Total}$	:	Total cost
$\underline{\alpha}$	:	User preference vector
$\alpha_1$	:	User preference of time
$\alpha_2$	:	User preference of Effort
$\alpha_3$	:	User preference of reliability
$a_1, a_2$	:	End effector position and EMG coefficients within the VD model

- $b_1, b_2$  : End effector position and EMG exponents within the VD model  
 $\chi_i^k$  : Coordinate position of the  $i^{\text{th}}$  swarm particle at the  $k^{\text{th}}$  iteration  
 $v_i^k$  : Velocity of the  $i^{\text{th}}$  swarm particle at the  $k^{\text{th}}$  iteration  
 $r_1, r_2$  : Uniformly distributed random numbers used in the PSO algorithm  
 $w^k$  : Inertial coefficient of the swarm at the  $k^{\text{th}}$  iteration  
 $c_1$  : Cognitive coefficient of swarm  
 $c_2$  : Social coefficient of swarm  
 $P_i^k$  : Personal best coordinate identified by the  $i^{\text{th}}$  swarm particle by the  $k^{\text{th}}$  iteration  
 $G^k$  : Global best coordinate identified by the swarm by the  $k^{\text{th}}$  iteration  
 $\underline{u}^*$  : Optimal control signal trajectory  
 $VD^*$  : Vector of optimal virtual dynamic parameters

## 1. Chapter 1: Introduction

Several day-to-day tasks in our lives require interaction with machines or devices [1]. Many of these devices are ubiquitous, like the computer mouse, keyboard, and car that we drive to work. Additionally, there are more specialized devices like prosthetic limbs or wheelchairs that greatly enhance the mobility and independence of users who may otherwise struggle with movement. In this work, we refer to any human interaction with such devices as occurring through human-machine interfaces (HMIs). These HMIs take input signals from the user and translate these signals to provide the required results for the user—like moving a cursor on a screen based on the position of a physical mouse or enabling a prosthesis user to hold a cup of water using their electrical muscle activity.

Irrespective of the device, whenever a human interacts with a machine, both parties play a role in determining the outcome. If the amount of fuel consumed as a car is driven is being monitored, the skills of the driver and the car's fuel efficiency play a role in determining the outcome. A different driver or vehicle could give altered results even if the exact same route is taken under identical external conditions. The optimal outcome can be achieved only when the goals of both the user and the designer of the machine are aligned.

People and animals instinctively modify their input to devices they interact with to achieve their performance goals [2], [3]. For example, a driver might accelerate when they go up a hill to maintain the speed of the car. However, machines themselves have specific dynamics that are sometimes adjustable/tunable, which dictate how they respond to the user's input. For a car, these adjustable dynamics come in the form of the driver mode

settings, which affect how the machine responds to a user's input, like how quickly the vehicle can accelerate or decelerate based on the user's input. Despite people implicitly adjusting their input to achieve their goals, and cases of general settings such as these driving modes, machines are typically not tuned with the user's performance goal in mind.

In this work, it is hypothesized that the default mapping of most HMIs produces suboptimal movements as they do not account for the performance goals of the user. For the focus of this thesis, myoelectric prostheses are adopted as the HMI device under investigation to understand how the device tuning affects user performance. The central thesis of this work is that optimizing the machine dynamics to align with the user's performance goals will enhance the overall user performance. For a prosthetic device, it is hypothesized that optimally tuning the device dynamics according to the user preference or performance goals will lead to improved performance.

The overarching goal of this thesis is to advance understanding of human-machine interaction, specifically for a user interacting with a myoelectric prosthesis, by exploring how coupling the device tuning with user performance goals can improve the interaction outcome. This objective is implemented through the following specific aims.

## **1.1 Specific aims**

*Specific Aim 1: Create a model of the human-machine interaction scenario in which a user interacts with a prosthetic device using myoelectric signals.*

Develop a mathematical model grounded in engineering principles to quantify the relationship between the prosthetic device tuning parameters and the user's performance

goals and priorities. This objective was attained by leveraging computational motor control techniques to create a model of user-device interaction that incorporates specific movement-relevant performance goals that the user can adjust.

***Specific Aim 2: Investigate the impact of performance-based device tuning***

Examine how optimizing the device tuning parameters according to the user's performance goals influences the overall task performance requiring a human-machine interaction. The model developed from Aim 1 was used to simulate the interaction scenarios and to assess the performance of a simulated user interacting with a tunable HMI.

**1.2 Significance**

The findings of this research indicate that people's performance when interacting with a device can be improved by coupling the device parameters to the user's goals.

An important contribution of the proposed algorithm is to potentially simplify the customization and calibration process of assistive devices like prostheses. Conventionally, the calibration of these devices requires a clinician to manually tune several abstract parameters that are not directly related to the user's movement objectives. This study provides an alternative that could aid the clinician and significantly reduce the time required for device customization and the overall calibration process.

An extension of this work could have far-reaching impacts on device manufacturing, for devices that can benefit from user-based tuning. The HMI devices with parameters that can be adjusted to improve the user performance, could result in increased uptake, acceptance,

and usage if they are produced with preference-based tunable parameters that are user accessible.

### **1.3 Innovation**

This study frames the user-device interaction problem using a mathematical model that is posed from the perspective of cost functions or user goals. The idea of coupling the device's tuning parameters to these goals and optimizing these tuning parameters to minimize the cost to the user completely reframes the traditional device tuning approach.

Another novel contribution of this work is the incorporation of the concepts of multiple objective and virtual device tuning parameters within the context of computational motor control.

### **1.4 Overview/ Thesis structure**

The following chapters of this thesis describe the necessary background information, the mathematical framework of the model, the tests run on the developed model, and the results obtained from the study. Chapter 2 delves into a review of the relevant literature in the fields of movement modelling and multi-objective optimization. A detailed description of the developed model and the optimization algorithm used is included in Chapter 3. Chapter 4 presents the results and findings obtained from the various tests conducted on the model. Chapter 5 includes a high-level discussion of the results, acknowledges the limitations of the study, and lists the potential future research that could be built on this work. Finally, Chapter 6 ties everything together by summarizing the key contributions of the research and highlighting the significance of the work.

## **2. Chapter 2: Background information and literature review**

The primary motivation for this work comes from the field of myoelectric prostheses. Regular upper limb prosthesis use for unilateral amputees can have several benefits, including improved symmetry and better function in a personal, social, and professional context [4]. Yet, a large number of users reject their prosthetic devices and choose not to continue using them regularly [5 - 7]. Some of the reasons generally stated for this rejection include a lack of intuitive control of the device, lack of sensory feedback from the device, and the long training time required to learn and use the device [8].

The goal of this thesis was to understand if changing the effective device dynamics or the input-output mapping behaviour of the prosthetic device based on the user's movement goals could improve the user's experience with the device. It is expected that a better user experience with the device could potentially increase the uptake and usage of prostheses. For a powered prosthesis, the device mapping is generally adjusted during the calibration or tuning phase, in which the clinician manually adjusts the tunable parameters of the device based on their observation of the patient and their interaction with the device. This thesis proposes a prosthesis calibration algorithm that could help the clinician as they adjust the device for different individuals. Because the concept of tunable parameters applies to most HMI, we can build a generalized mathematical model that can inform the users how the device behaviour could be adjusted to improve the user's overall performance. To build a model that informs device tuning, tools and techniques from different fields were used, including computational motor control, control theory, and optimization.

The field of computational neuroscience, or computational motor control (CMC), describes how human movement behaviour changes according to the costs and rewards associated with an action [9 - 11]. Framing the problem of human behaviour and movement using the cost and reward functions help create mathematical models of stereotypical movement and movement behaviours.

Optimization is a fundamental concept in mathematics and engineering that seeks to identify the optimal solution for a given problem by minimizing or maximizing an objective function while satisfying problem-specific constraints. By systematically exploring the solution space, optimization techniques allow the decision-maker to make informed choices, leading to resource savings or improved productivity or performance. There are two broad approaches to finding optimal solutions – the classical mathematical approach and evolutionary or heuristic algorithms.

The field of multiobjective optimization is associated with multiple-criteria decision-making in which several potentially conflicting costs or objective functions are optimized simultaneously [12]. This concept allows the broadening of the conventional computational models of motor control and helps to explain how the generated motor commands differ based on the user's preference for the multiple criteria or the cost functions involved in motor control. In practice, people simultaneously care about several aspects of movement, like the amount of effort required to perform the action, the time required to complete the action, and the likelihood of success [11]. So, creating a multi-objective motor control model in which the user's prioritization of these individual costs can be varied was the essential first step to ensuring that the model has practical relevance.

The following sections describe a detailed discussion of the computational motor control model, the concept of performance goals, the terms and background necessary to understand the basics of multi-objective optimization (MOO), and the concept of virtual dynamics.

## **2.1 Modelling human movement using computational motor control tools**

Humans and animals show high degrees of regularity in their movement. They show consistency and repeatability in specific movements as well as high-level movement behaviours. Movements like eye saccades [13] and gait [14] have been studied empirically and show repeatability in output behaviour within subjects, as well as broadly across the populations. High-level movement behaviours like foraging patterns in birds show that birds change when they walk or fly to get their prey based on the net rate of energy gained from the action [15]. All biological beings show this movement regularity because every movement is a result of an optimal motor planning strategy that tries to maximize the reward or the value that we attain from the movement by minimizing the costs that negatively affect the value obtained from the movement [11], [16], [17].

It is believed that the basal ganglion of the brain is responsible for learning the rewards associated with an action and calculating the motor execution that maximizes these rewards [18], [19]. The basal ganglia can calculate these motor executions based on our brain's representation of our body and the world around us. These abstract representations built by our nervous system are called internal models [20], [21]. The concept of the internal model and the meaning of the costs and rewards that it uses to determine the best movement strategy are described in Section 2.1.1 and Section 2.1.2.

### **2.1.1 Internal models to describe motor control**

Internal models are computational representations of how our nervous system keeps track of our physiological and external world parameters [21]. These internal models within our brain allow us to make predictions based on our past observations and make corrections to movements if a disturbance occurs [22 - 24]. From a control theoretic point of view, the internal model can be thought as the brain's approximate description of the dynamics and parameters of our movement apparatus, like our limbs and eyes, and the external world in which we are present. For example, internal models keep track of parameters like the weight of a limb, the acceleration due to gravity, the viscosity of the environment, etc.

There are two types of internal models – forward and inverse models [20]. Forward models predict the sensory consequences of motor commands. For example, the forward model helps predict how a ball would move once it is struck. They are termed forward models because they help predict the effect based on the cause. Inverse models, on the other hand, estimate the motor commands required to achieve a desired sensory state. For example, if the desired trajectory for a ball to travel is known, the inverse model determines the necessary control signal that should be used to strike it. They are termed inverse models because they determine the cause that produces the desired effect. Our brains utilize both the forward and the inverse internal models in sensorimotor control.

Knowledge of the parameters in the biological internal model to optimize the movement trajectory or calculate the best course of action that maximizes the reward obtained [10], [22 - 24]. The best strategy is the one that gives the person the most value or reward for performing a given action [25], [26].

Computational motor control aims to mathematically describe these brain models and their relevant optimal movement strategies. This does not mean that these math models are the same as those used by the brain, but that they are capable of approximating the overall behaviour of the brain model.

### **2.1.2 Cost functions in biological systems**

The biological internal model computes the best movement strategies for motor control by maximizing the reward that it gets from the action. Using computational motor control, we can describe these optimal motor control strategies as those in which the value of the movement reward is worth the costs of effort and time required to reap the rewarding state of the action. For example, if we were to pick a dessert from a collection of assorted desserts, we would reach for our favourite one first, because it gives us a larger reward in terms of the dopamine release in our brain. But the action of reaching to pick this dessert comes with its own metabolic costs and the brain is evolutionarily trained to minimize this cost.

These ‘costs’ are mathematical representations of quantities that our nervous system tries to minimize when generating motor commands [11]. Several cost functions like effort [25], [26], metabolic energy [14], [27], endpoint variance [28], and trajectory smoothness [29] have been used to describe specific movements. This diversity in the literature suggests that humans optimize a combination of different costs [30 - 32]. In real life, we care about several conflicting costs. For example, the comfort of the passengers, fuel efficiency, and the time required to reach the destination and important on a drive; the driver will modify how they drive according to their priorities.

One of the most popular laboratory tasks that tell us about the tradeoff between conflicting goals is the Fitts' style reaching task [33]. Fitt's law is commonly used in the fields of human-computer interaction [34], user experience [35], [36], and studies involving the usability and performance of electromyography signal classifiers [37], [38]. The original Fitts' task was set up as a reciprocal reaching task in which the subjects were asked to land a stylus within an acceptable tolerance of a target that was placed at different distances. The subjects were asked to make the reaches as quickly as possible such that they maximized the number of targets they reached within a finite block of time.

The results of the Fitts' study showed that the movement time of the subjects increased as the distance between the targets increased, and the movement time increased as the acceptable tolerance of the targets decreased. This latter result is commonly referred to as the speed-accuracy tradeoff because of their inverse relationship. If the targets have a smaller acceptable tolerance or width, the participants must slow down and take more time to stop within the acceptable target range.

This observed speed-accuracy tradeoff in reaching tasks could be explained by the internal model simultaneously optimizing for the two conflicting costs of time and reaching accuracy. Though computational models have considered scalarized costs that are a combination of different conflicting costs like effort and accuracy [24], [39], [40], the literature has not yet explained how the user prioritization of these costs can affect the generated movement, even though it is clearly observed in real-life actions. Hence, this work incorporates the idea of multi-objective optimization with user-adjustable cost

preference weights within the context of computational motor control models, aiming to make them more generalizable and bring the mathematical models closer to reality.

## **2.2 Multi-objective optimization**

Multi-objective optimization (MOO) is a field of multi-criteria decision-making in which optimal decisions should be made when there is a trade-off between different goals [41]. MOO is often used in engineering design, manufacturing, and product monitoring applications [42], [43] to help make a design or policy choice.

The field of MOO has contextualized how an optimization problem can be expressed when several costs are involved. The costs could be optimized independently, or the priorities/trade-off points between the different goals could be expressed to identify the best solution. Whenever a problem has contradictory or conflicting goals/objectives, there is no single best solution; an infinite number of optimal solutions exist depending on how the different goals are prioritized.

In what follows, Section 2.2.1 describes an overview of the terms and definitions of MOO. The high-level classification of MOO techniques and the goals of MOO solvers are described in Sections 2.2.2 and 2.2.3, respectively. Section 2.2.4 expands on the interpretation of the MOO coefficients or the preference vectors and explains how they should be interpreted.

### **2.2.1 Terms and definitions**

(i) **Space:** refers to the mathematical landscape that consists of the objective functions or design variables.

- (ii) **Set:** refers to a mathematical set or group of points within the landscape.
- (iii) **Objective function:** refers to the function or mathematical expression that measures the quality of the solution. The goal of any optimization problem is to either maximize or minimize the objective function. The minimization and maximization actions are interchangeable, and in the rest of the document, the objective function is defined as needing to be minimized, unless specifically mentioned otherwise.
- (iv) **Design variables:** refers to the input parameters or the decision variables of the optimization problem that can be adjusted to achieve the minimal objective function value. Examples of common engineering design variables include the dimensions, weight, or type of material used in the design. For an HMI device, some of the common design variables could include the gain or sensitivity of the device, sensitivity to input change, acceleration, etc.

*As designers, we control the design variables. The objective function value corresponding to the design variables measures the quality of the solution.*

- (v) **Objective space / Criterion space:** refers to the space that consists of all the objective functions involved in the multi-objective optimization problem.
- (vi) **Design space / parameter space:** refers to the space that consists of all the design variables that can be altered to improve the objectives of interest. Each design variable will be an axis in the hyperspace.

Consider a MOO example of manufacturing a beam such that the monetary cost and the deflection of the beam is minimized. This can be implemented by changing the length, breadth, thickness, and width of the beam. For this problem, the design space is four-dimensional and consists of four axes: the length, breadth, width, and thickness of the

beam. The objective space is two-dimensional and consists of two objective functions: the fabrication cost and the deflection of the beam.

*Any point in the parameter space can be converted to the objective space by evaluating the value of each objective function corresponding to the parameter space point.*

(vii) **Optimality:** Mathematical optimization refers to the process of finding the best design variables with regard to the objective function criteria from the set of available alternatives. The condition for optimality for single and multi-objective problems is different, as described below.

For a single objective optimization problem, the optimality of the solutions is simply based on the value of the objective function. The region considered for the optimality separates the solutions into local and global solutions. A locally optimum solution for a single objective optimization refers to the point in the design space with the least objective function value within a locally defined region of the input space. A globally optimal solution for a single objective optimization problem is the point in the design space with the minimum value of the objective function within the entire design space. If the objective function and constraints are convex functions and the search space belongs to a convex set, the locally optimal solution is also guaranteed to be the globally optimal solution.

For multi-objective optimization problems, there is no clear minimum, as what decreases the value of one objective function can increase the value of another objective. The optimality of the solutions is defined in terms of its Pareto optimality [44], [45]. The region considered for the optimality separates the solution into local and global solutions. A solution is locally Pareto optimal if there exist no other feasible solutions in its

neighbourhood that dominate it. A solution is globally Pareto optimal if it is not dominated by any other feasible solution in the entire search space [46]. The definitions for the terms domination and Pareto optimality are discussed in detail below.

(viii) **Dominance:** If we take the example of two potential solutions A1 and A2, in the feasible objective space, then solution A1 dominates solution A2 if and only if:

- Solution A1 is no worse than A2 in all the objectives AND,
- Solution A1 is strictly better than A2 in at least one of the objectives.

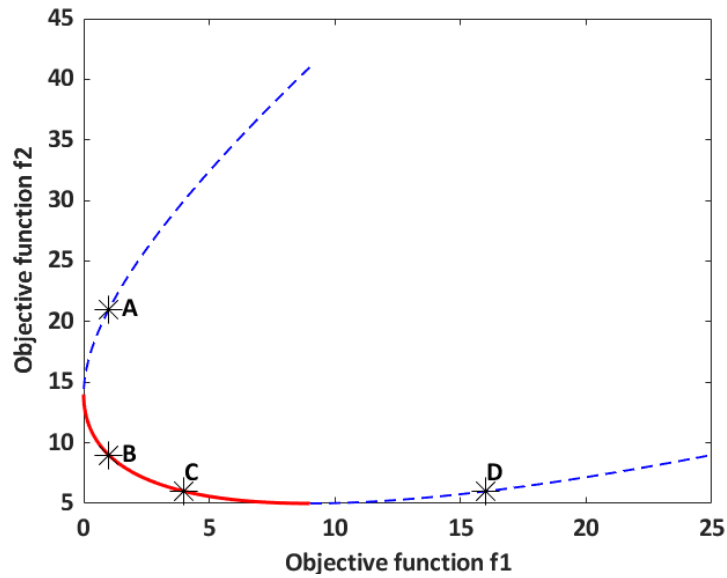
To better understand the concept of dominance, let us consider a simple MOO problem of simultaneously minimizing the two mathematical objective functions  $f_1$  and  $f_2$  according to the design variable  $x$  :

$$\begin{aligned} f_1(x) &= (x + 1)^2, \\ f_2(x) &= 5 + (x - 2)^2. \end{aligned} \tag{2.1}$$

Let us define 4 solution points A, B, C, and D, as the solutions given by setting  $x = -2$ ,  $x = 0$ ,  $x = 1$  and  $x = 3$ , respectively. The objective function values at each of these points is as given below in Table 1. The objective space visualization of the 4 solution points A, B, C, and D are shown in Figure 1.

Solution point B clearly dominates solution A – even though the value of the first objective function remains the same, the value of the second objective function of B is much smaller than that of A. Similarly, Solution C dominates Solution D – as the first objective of solution C is lower than the corresponding objective of solution D even though the second objective function values are the same.

However, if solutions B and C are compared, neither solution dominates the other, as an improvement in one objective function results in a worse value for the second objective function.



**Figure 1: Objective space visualization of the Simple MOO example (Eq 2.1). The dashed blue line corresponds to the objective space representation of the points displayed in Figure 2. The solid red line highlights the hyperplane created by connecting all the non-dominated solutions in the objective space (Pareto front). The solutions from Table 1 are labelled accordingly.**

**Table 1: Solutions points for sample MOO minimization problem from Equation (2.1)**

	<b>Design Variable</b>	<b>Objective function</b> $f_1$	<b>Objective function</b> $f_2$
<b>Solution A</b>	$x = -2$	$f_1(x = -2) = 1$	$f_2(x = -2) = 21$
<b>Solution B</b>	$x = 0$	$f_1(x = 0) = 1$	$f_2(x = 0) = 9$
<b>Solution C</b>	$x = 1$	$f_1(x = 1) = 4$	$f_2(x = 1) = 6$
<b>Solution D</b>	$x = 3$	$f_1(x = 3) = 16$	$f_2(x = 3) = 6$

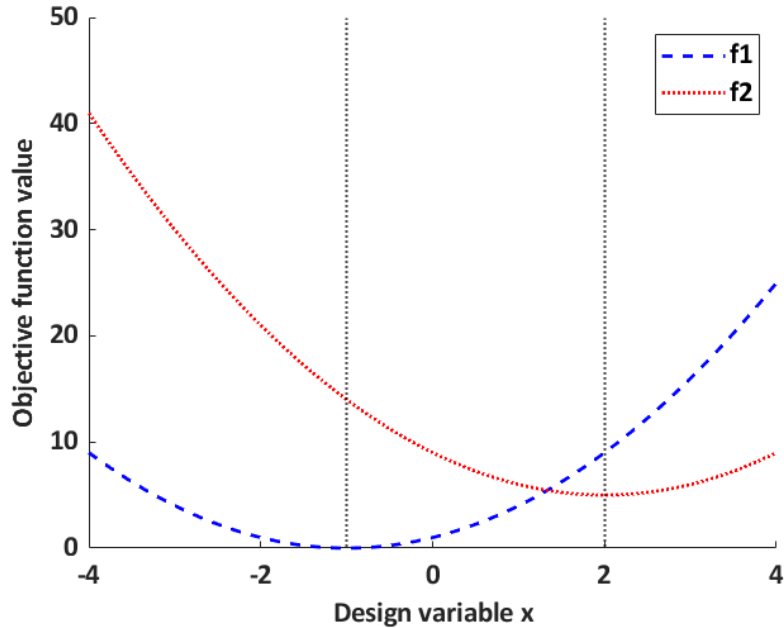
(ix) **Pareto solution**/Pareto optimal point: A point  $x^*$  is Pareto optimal if there is no other point  $x$  in the feasible criterion space that improves at least one objective function without worsening another objective [47]. In other words, a Pareto optimal solution is one that is non-dominated.

If we reconsider the MOO example from Equation (2.1), the individual objective functions  $f_1$  and  $f_2$  have their minimum values at  $x = -1$  and  $x = 2$ , respectively. If we look at this problem in the context of MOO, any value of  $x$  in the range of  $[-1,2]$  will lead to solutions such that we cannot reduce one objective function without increasing the value of the other one, as shown in Figure 2. Any solution with  $x$  in the range of  $[-1,2]$  is, hence, a Pareto optimal solution or a Pareto optimal point.

Note that the Pareto optimal solutions are discussed in terms of the design space.

(x) **Non-dominated set**: Given a set of solutions, the non-dominated set is the group of solutions that are not dominated by any member of the set. From the above example, the

design parameters of points B and C will form the non-dominated set when the solutions A, B, C and D are considered. This is indicated in the objective space visualization of the example problem in Figure 1.



**Figure 2: Comparison of the two objective function values,  $f_1$  (dashed blue) and  $f_2$  (dotted red) for a range of design variable values. The dotted vertical lines indicate the range of  $x$  values that belong to the Pareto optimal solution set.**

### 2.2.2 Classification of MOO

For a multi-objective problem, several “optimal” solutions exist such that no objective can be improved without degrading at least one of the other objective function values. If a single solution is to be identified from the ensemble of possible optimal solutions, a relative preference for different objectives (or the lack thereof) must be defined. MOO techniques are classified into four general categories based on how this user preference is incorporated into the algorithm [48], [49].

These categories are:

- (i) No articulation of preference
- (ii) *A priori* articulation of preference
- (iii) *A posteriori* articulation of preference
- (iv) Interactive articulation of preference.

The no-preference methods do not assume the importance of objectives but use a simple condition to obtain a single solution. The global criterion is commonly used when the decision-maker does not have any expectations of the solution. In that method, the distance between a reference point (for example, the ideal point) and the feasible space is minimized [50]. The *a priori* methods use the information on the objective priorities before the optimization and find one preferred Pareto optimal solution. The *a posteriori* methods make use of the preference information after the optimization to select a preferred solution from the points obtained in the Pareto front. Interactive methods use preference information progressively throughout the optimization process and are the preferred approach when the global preference structure of the decision maker is not known ahead of time. Throughout the iterations of the optimization process, the decision maker adjusts and selects their preferences based on the obtained solutions. For a detailed description of the classification and different MOO algorithms used in the literature, refer to the reviews by Marler *et al.* [48], [51].

There are broadly two types of solvers or algorithms that can be used to find optimization solutions for any of the four categories of preference: classical mathematical approach and evolutionary or heuristic algorithms. Classical techniques use mathematically appropriate

functions to find the saddle points of the Lagrange functions and identify one optimal solution point in a single simulation run. The evolutionary or heuristic methods, on the other hand, can identify several solutions within a simulation run due to the population approach and can be easily parallelized. However, the solutions obtained from the evolutionary algorithms cannot be guaranteed to be locally optimal.

Evolutionary algorithms that incorporated *a priori* articulation of preference were used to implement the full HMI model in this project. Hence, the focus of this document will be on *a priori* articulation of preference using evolutionary approaches.

### **2.2.3 Linear scalarization**

One of the most popular ways to express a MOO problem such that the priority of the objectives is defined ahead of time is the Linear scalarization method. The technique scalarizes a set of objectives into a single objective by pre-multiplying each objective with a user-supplied weight or preference value. The weight should ideally correspond to the importance of the objective for a given problem. If each objective of a multi-objective optimization problem is represented by  $f_i(x)$ , where  $i \in \{1, \dots, k\}$  where  $k$  represents the number of objectives in the problem, the linear scalarization methods would transform the problem to be:

$$\min_x \sum_{i=1}^k \alpha_i f_i(x), \quad (2.2)$$

where the weight terms  $\alpha_i$  are strictly positive.

Weighted sum optimization is a special case of the linear scalarization approach in which the sum of the weights  $\alpha_i$  adds up to 1.

Scaling each objective is essential when the individual objectives take different orders of magnitude for a given problem. Scaling or normalization helps ensure that the objectives are comparable and approximately equally important, thus allowing the user preference or the weights to influence the optimal solution found.

One advantage of the scalarization technique is that the solution to any weighted sum optimization problem is Pareto-optimal if the weights are positive for all objectives. Additionally, for a convex MOO problem, any Pareto optimal solution can be found using the linear scalarization or the weighted sum approach. However, there are also disadvantages to this technique. For nonlinear MOO problems, a uniformly distributed set of weights does not necessarily result in a uniformly distributed set of Pareto optimal solutions, making it difficult to pick weight vectors that yield Pareto optimal solutions in the desired region of the objective space. Moreover, different weight vectors do not necessarily lead to different Pareto optimal solutions, and the linear scalarization approach cannot find certain Pareto optimal solutions if the objective space is non-convex.

#### **2.2.4 Goals of MOO**

The main goals of any method used to solve a MOO problem are to find solutions that are part of the Pareto optimal front and ensure that the set of obtained solutions is diverse in the desired space: design space, or the objective space. Based on the application, it might be more effective to look for diversity in a certain space. The focus throughout this document will be to look for a diverse set of solutions in the objective or the criterion space, as that is the space of interest in most engineering applications.

The Pareto front will only have more than one point within it if the objectives considered are conflicting. Whenever there is a set of non-conflicting objectives, the Pareto optimal front will only have one solution; in other words, the best solution corresponding to any objective function is the same.

### **2.3 Device dynamics and virtual dynamics**

The ease of using any HMI greatly depends on the input-output mapping of the device. For physical devices like the shock absorption system of a bike, device dynamics can be described using the damping coefficient and the spring stiffness of the elements that make up the shock absorber. The device dynamics of the physical system describe how it will react to an input and the corresponding output it will produce.

Herein, the term “virtual dynamics” is used to refer to systems in which the input-output mapping behaviour can be programmed by the designer. For example, a computer mouse typically maps the input signal from the user (relative position of the physical mouse) to the position of a cursor on the screen. On the other hand, most joysticks map the input displacement to the velocity of the cursor on the screen. A designer programs the input-output mappings of both these devices. Such devices are tuned using the device driver software without having to swap out physical elements of the devices. Hence, these input-output mappings can be referred to as virtual dynamics that the designer can program.

This work adopts the use case of myoelectric prostheses. These devices have their own physical dynamics determined by the mechanical design of the device and the actuators used. They typically also include a controller that can be used to determine the overall

input-output mapping between the user's EMG signal input and the corresponding movement of the prosthetic limb. This input-output mapping describes the desired closed-loop dynamics of the prosthesis and the controller. Because the controller can be programmed to determine the closed-loop behaviour of the system, the closed-loop behaviour can be considered to be the virtual dynamics of the device.

Specific device dynamic mappings or virtual dynamic mappings might be favourable for different tasks [52-56] or could simply be preferred by different individuals [57]. The current clinical standard for the control mapping of myoelectric prostheses is proportional velocity control [58], which maps the intensity of muscle contraction to the velocity of the device. Some of the mathematical filters used to achieve a virtual dynamic mapping include the Kalman filter, 1 Euro filter [59], adaptive EMG filter [60], nonlinear recursive Bayesian filters [61], [62], linear filters, and kurtosis filters [63]. The concepts of nonlinear dynamics [10], [23] and blended control [24] have been independently explored in recent years. Inspired by the superior performance of the velocity-squared control dynamic mapping [23], the concept of a nonlinear blended virtual dynamic model is proposed in this work, wherein the input control signal is correlated with the position of the joints of the device and their derivatives, such as velocity and acceleration. This thesis aims to investigate the influence of the user's preference (or the relative importance of costs) on the optimal control mapping for a prosthetic device.

### 3. Chapter 3: Methods

The two specific aims of this work are to create a model of the human-machine interaction scenario of a user and their myoelectric prosthetic device and to investigate the impact of performance-based device tuning.

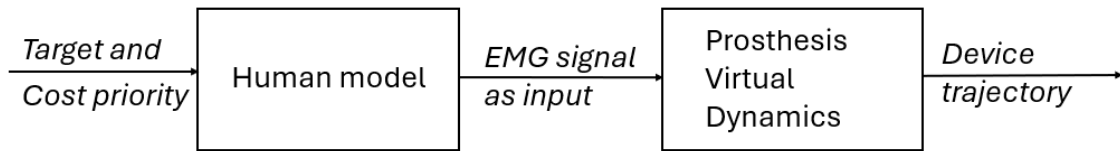
To create a model of the HMI interaction scenario, reaching tasks were adopted for modelling using optimal control theory to calculate the best set of motor commands that minimized the costs associated with that action. Whenever there is a set of non-conflicting objectives, the best solution corresponding to any objective function is the same. Due to this condition, only movement-relevant costs that are conflicting were used to mathematically simulate the biological internal model's process of computing the optimal motor control. Situations in which the objectives are not conflicting were not explored as these non-conflicting objectives could easily be modelled as a single objective optimization (SOO) problem instead. The hand-reaching models were hence reframed to be described as tasks with multiple objectives requiring the participant's internal model to optimize for the tradeoff between several costs to compute the best movement trajectory.

Inverse reinforcement learning algorithms aim to identify the objective function, which, when optimized, results in the observed behaviour [64]. Simulation studies based on inverse reinforcement learning [30], [65] suggest that the user priority for the individual costs could change from one phase to another within an action and vary with time. These cost priorities can also change from person to person [66]. For example, within a Fitts' law test, some participants might try to reach the center of the target, while others might prefer to just make it within the boundary of the acceptable target so they can focus on improving

the speed of the action [67]. Different tasks could also force the user's cost priorities to change [68]. For example, if the target size for a Fitts' law task is reduced, the participants are forced to move slower so that they can stop within the acceptable target width [69].

This work focuses on creating a model of how reaching tasks are optimized within the internal model of humans. The composite costs of effort, accuracy, reliability, and time sufficiently describe how humans consistently coordinate their joints and perform movements [11]; these are the four costs that will be focused on for the remainder of this thesis. The exact mathematical description of each of these costs and a description of how they fit into the overall simulation model are specified in Section 3.1.

The research contribution of this thesis includes creating the HMI interaction model that incorporates multiple objective functions and analyzing how the user preference level impacts the optimal user input signal and the optimal virtual dynamic parameters from the combined model. The HMI model consists of two parts: a model of a human user producing EMG signal according to their relative cost preference and a model of a tunable myoelectric prosthesis whose parameters can be adjusted according to the user preference. A block diagram representing the parts of the model is shown in Figure 3. The model simulates a single-degree of freedom (DOF) target-reaching tasks performed by a user of a myoelectric prosthesis.



**Figure 3: Simplified block diagram representation of the User-Prosthesis interaction model.**

The human user part of the model determines the optimal control signal in the form of the neural drive to perform an action based on computational neuroscience models. The model's optimal neural signals are then used to estimate what the surface myoelectric signals from the user would look like. The tunable virtual dynamic of the prosthesis is used to model the machine component of the HMI interaction. Because virtual dynamics are used to represent the prosthesis, EMG signals from the human model are used as inputs, and the output of the dynamics model gives the states of the end effector of the prosthesis. The output, hence, provides the state trajectory and its derivatives like velocity and acceleration.

The process of formulating the HMI interaction model can be discussed in three subparts or phases. The first phase incorporates multiple costs within the CMC framework and contextualizes user preference. The second phase involves modelling a human user's EMG output in accordance with the biological internal model theory when the user cares about multiple costs. The third phase incorporates the tunable virtual dynamics of the prosthesis to the human model so that the combined effect of the user interacting with a prosthetic device can be studied. The full model allows us to understand the effect of the user

preference level on the neural signals produced by the user, and the effect of the optimal prosthesis parameters on the overall task performance.

A heuristic optimization algorithm (described in Section 3.5) was used to understand how the user's trade-off between the costs, also known as the user preference value, affects the optimal control signals and the optimal virtual dynamic parameters within the model.

It is important to note that the HMI interaction mode (as shown in Figure 3) is set up as an open-loop optimization problem such that the human user does not get feedback about the progression of the device during the task. This choice of implementation means that the user cannot correct their input signal to the device mid-trajectory to adjust or accommodate for any undesired output from the device.

### **3.1 Incorporating multiple objectives for motor control**

The first phase of this work involved selecting the conflicting costs that people implicitly care about when moving their limbs and understanding how the user preference between those costs can be described mathematically to combine these costs as a collective.

#### **3.1.1 Costs considered for the task of interest**

To constrain the scope of the project, only models that describe reaching tasks in which the user intends to move the end effector of their prosthetic device from one point to another using their EMG signals were explored.

EMG is inherently noisy and contains both signal-dependent and signal-independent noise [70], which results in movement variability or errors at the endpoint. Because of the nature of the task being modelled, the value of reward obtained from performing the reach

would depend on the amount of effort required to produce the EMG signals necessary to move the prosthesis, the time required to reach the target, the variance of the prosthesis about the target when it comes to a stop, and the accuracy with which the user is able to reach the target. The task was modelled with the constraints of having to necessarily reach the target on average over several trials and always have zero velocity at the endpoint. The variables that negatively affect the reward obtained from performing the task were modelled as a composite cost that needs to be minimized.

The composite cost function for the model was composed of three costs and three constraints.: The three movement relevant costs include the effort spent in making the movement, the time required to execute the task, and the endpoint variance in reaching the target. The cost of endpoint variance is referred to as the cost of reliability or the risk involved in making the movement, as it represents the amount of uncertainty in reaching the target [71]. Throughout this document, the terms “risk” and “cost of reliability” are used interchangeably to refer to the cost of endpoint variance.

The cost of time for humans is well represented by a hyperbolic function, as described by Haith *et al.* [72] and is given by:

$$J_t = 1 - \frac{1}{1+\beta p}, \quad (3.1)$$

where  $\beta$  is the rate of temporal discounting and  $p$  is the time required to complete the reaching task. The value of  $\beta$  in this model is set to 0.09, according to the temporal discounting rate of humans, as observed in the study conducted by Jimura *et al.* [73]. It is assumed to be always preferable to attain a reward sooner rather than later, so the longer it takes to get the rewarding state, the smaller the value of the reward for the internal model.

The cost of the effort was taken as the summation of the squared control signal  $u$  for the entire movement duration. This is represented by,

$$J_u = \sum_{t=1}^p u(t)^2. \quad (3.2)$$

The term  $u$  corresponds to the neural intent or the motor commands from the brain that produce a movement in the skeletal muscles. The cost of effort prevents people from taking actions that do not result in a valuable reward.

The cost of reliability was a function of the endpoint variance and is defined as:

$$J_r = E[(g - x(p))^2] - E[g - x(p)]^2, \quad (3.3)$$

where  $g$  is the goal or the target position,  $x(p)$  is the actual position of the device at the end of the movement, and  $E[\ ]$  represents the expected value operator. The cost of reliability emphasizes the importance of the likelihood of reaching the target. The noise inherently present in the EMG signals results in a stochastic distribution of the endpoint states; in other words, the outcome cannot be perfectly predicted, nor can it be guaranteed that the reward will be attained. This cost prevents people from taking action if the expected reward is not worthy of the action.

Apart from these costs, three constraints ( $C_1$ ,  $C_2$ , and  $C_3$ ) were added to create the scalarized objective. The first constraint,  $C_1$ , ensured that the target was reached on average when the reaching trials were repeated several times,

$$C_1 = E[g - x(p)]^2. \quad (3.4)$$

The second constraint,  $C_2$ , ensured that the device stopped at the end of the movement,

$$C_2 = E[\dot{x}(p)^2], \quad (3.5)$$

where  $\dot{x}(p)$  refers to the velocity of the device at the end of the movement.

The third constraint,  $C_3$ , was added to include a penalty on large spikes in the initial velocity to deter the control input from changing rapidly at the first timestep,

$$C_3 = E[\dot{x}(1)^2]. \quad (3.6)$$

Here,  $\dot{x}(1)$  represents the velocity of the device at the first time step of the movement.

These constraints were implemented by adding penalties to the cost function as described by Stengel [74].

### 3.1.2 Combined cost function

The HMI interaction model assumes that the user is able to indicate their preference between the costs or indicate how much they care about each of the conflicting costs involved in the task. In the field of multi-objective optimization, knowledge of the user's preference is used in the *a priori* articulation of preference strategies to identify the best solution according to the user's preference level.

The linear scalarization approach was implemented to create a single scalar cost function from the multiple costs by scaling each objective with a coefficient and adding the scaled costs to create a single objective, as shown in Eq (2.2).

The conceptual meaning of these coefficients, also known as the weights, is that they indicate the user's preference for each cost. In other words, the cost that is more important to the user has a weight value greater than a cost that is of lower importance to the user. This direct correlation between user preference and the weight value of the different costs makes intuitive sense if the individual costs themselves are comparable. But more often

than not, the individual costs cannot be directly compared as they correspond to different quantities – and a direct comparison would look like comparing apples to oranges. When the problem is not adjusted to account for the difference in the magnitude and the range of the different objective functions in a multi-objective problem, the interpretation of the weights in the linear scalarization method will be incorrect [75].

The objective functions involved in movement, including the costs of effort, time, endpoint variability, and accuracy, have vastly different relative magnitudes, different ranges in terms of the values that they can take, and different dimensionality or units that they can take. For a reaching task, the cost of accuracy is expressed in units of distance, whereas the cost of effort is in terms of the magnitude of neural drive which doesn't have any physical measurable units. Because the value of the multiplier weight is meaningful relative to the value of the other weights and the absolute magnitude of each weight is irrelevant [75], individual costs themselves must get transformed such that they are comparable when added together to create a single weighted objective. There are several ways to perform this translation: the user preference can be expressed as a scaled function of the average value of the individual costs, the maximum value of the individual costs, the minimum value of the individual costs, or as a function of the range of values for each cost [75-77].

With the HMI interaction model, transforming the individual costs and normalizing them about the average, maximum, minimum, or range of values that each objective can take is not appropriate due to the practical relevance of the model - the minimum possible value for each of each individual cost is zero, there are no mathematical limits for the maximum effort or time required for a movement, and the average and the range of these costs depend

on the parameters of the task itself. Hence, the transformation was implemented by linearizing the individual cost functions about the region in which clinicians and patients might operate. It was assumed that an arbitrary time of one second would be required to make a 10 cm reaching task using a prosthetic device with proportional velocity control dynamics, which is a feature commonly available in most commercial myoelectric prosthesis [78]. Proportional velocity control refers to a method where the speed and the force of the prosthetic hand's movements are directly proportional to the magnitude of the electromyographic signals generated by the user's muscles. It was also assumed that the user would perform a minimum jerk trajectory [79] to reach the target and operate around 35% of their maximum voluntary contraction levels. This submaximal muscle contraction intensity was selected to simulate sustainable contractions in daily activities without fatigue [80]. The trials were repeated to get a reliable average of the costs due to the stochasticity in the problem. The total number of simulated trials was selected based on when the average costs stabilized and plateaued. A total of 100 reaching trials to the target were simulated, and the average individual costs for effort, time, reliability and constraints were used to normalize the respective cost functions. The individual objective functions were normalized such that the magnitude of the average values of individual costs were comparable and within the range of [0.99 to 1.3]. Once the individual weights were comparable about this practically relevant operating point, the magnitude of the weights could be used as a proxy for the user's relative preference between the different costs.

The total cost was defined as the weighted sum of the three cost functions (Eq 3.1 to Eq 3.3), the two stopping constraints (Eq 3.4 to Eq 3.5), and the penalty on the initial spike in velocity (Eq 3.6) and was represented by:

$$J_{Total} = \alpha_1 \cdot J_t + \alpha_2 \cdot J_u + \alpha_3 \cdot J_r + 100C_1 + 10C_2 + C_3 \quad (3.7)$$

The relative weights of the individual cost functions are represented by the coefficients  $\alpha_1$ ,  $\alpha_2$  and  $\alpha_3$ . Four different user preference scenarios were tested. To represent a situation in which the user cares equally about the effort, time, and reliability costs of reaching the target, the coefficients  $\alpha_1$ ,  $\alpha_2$  and  $\alpha_3$  were all set to one. To represent circumstances in which the user cares more about one of the individual costs, the corresponding preference level, or  $\alpha$  value, was set to five, while keeping the other preference levels at one as shown in Table 2. The coefficients of the two constraints  $C_1$  and  $C_2$  were based on recommendations from the supplementary notes of the modelling work done in [24]. The coefficient of  $C_3$  was set to unity to add a small penalty for spikes in the initial velocity.

**Table 2: List of tested user preference conditions and their corresponding user preference values.**

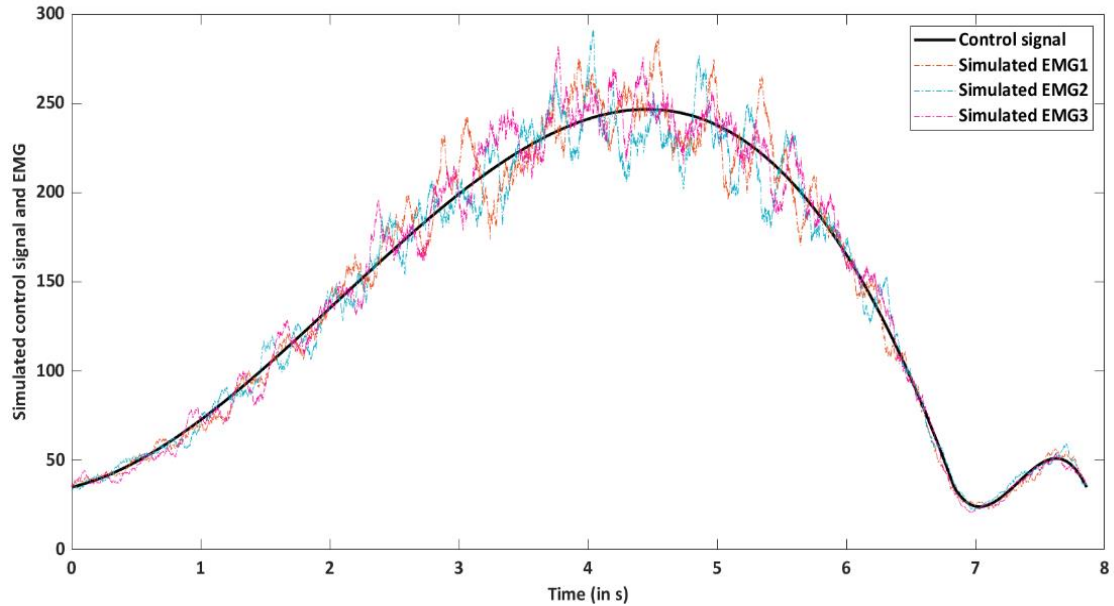
Preference condition	Preference vector $\underline{\alpha} = [\alpha_1, \alpha_2, \alpha_3]$
Equal preference	[ 1, 1, 1]
Time prioritized	[5, 1, 1]
Effort prioritized	[1, 5, 1]
Reliability Prioritized	[1, 1, 5]

## **3.2 Human part of the HMI model**

As shown in Figure 3, the HMI interaction model consisted of two parts: a model of a human user producing electromyographic signals to interact with a myoelectric prosthetic device and a model of the myoelectric prosthesis using the EMG signals from the user model as the input to produce a resulting motion.

### **3.2.1 Model of the user intent or the neural drive**

The optimal neural signals were modelled based on CMC with the weighted scalarized objective, as shown in Eq (3.6). It was expressed as an optimization problem in which the neural signals driving the muscles, referred to in this document as  $u$ , are one of the arguments optimized to minimize the total cost to the user. Because the neural drive is a continuous time-varying signal, it was modelled using cubic splines that were concatenated together. It was found that the shapes obtained from two concatenated cubic splines sufficiently described the control intent, as shown by the example control signal in Figure 4. Two concatenated cubic splines could produce shapes including submovements which have been observed in real world movements. Nine tunable parameters corresponding to the time duration, maximum amplitude and the initial and final slopes of the splines were optimized to simulate the natural optimal control intent produced by the user when reaching for a target.

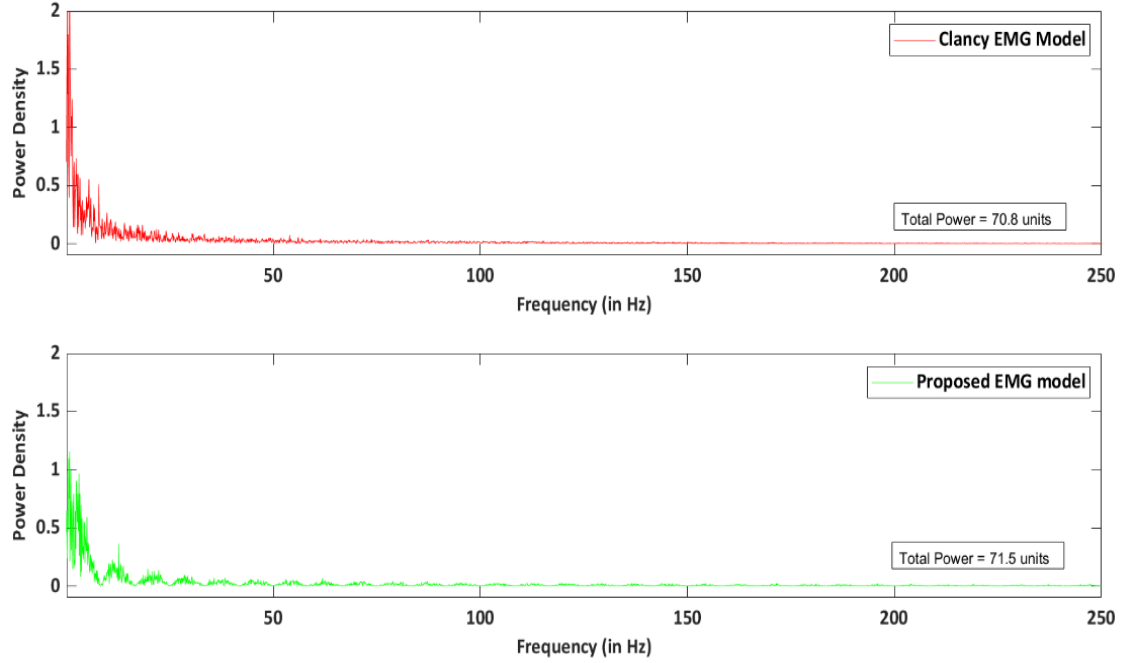


**Figure 4: Example of the simulated control signal and corresponding EMG in the time domain.**

### **3.2.2 Model of surface EMG signal corresponding to the control intent**

A control signal is computed by the brain every time a movement is planned, but the corresponding motor output is always corrupted by noise. Two variants of noise affect biological systems [81], [82]: the first is the additive noise that is independent of the control signal amplitude; the second is the multiplicative noise whose standard deviation increases linearly with the signal amplitude [83], [84]. Both of these sources are present in EMG signals, even after the raw signals have been filtered and smoothed. Hence, the control signal obtained from the cubic splines was corrupted by both noises in order to simulate EMG signals measured from a human. This noise was included in the model using a technique that is loosely based on Clancy's EMG estimation model [85].

Additive and multiplicative noises were added to the control signal model, and a moving average filter was used to create the effect of the smoothing low pass filter used in EMG amplitude estimation [86]. The EMG signal was simulated to have been sampled at 1000Hz with a moving average filter window of 120 milliseconds. The smoothing window size was selected to fall within the commonly used range of 100 to 200 millisecond windows for EMG amplitude estimation [87 - 90]. Smaller windows capture the dynamic changes in the signal and larger windows result in smoother amplitude estimates. The scaling factors for the additive and the multiplicative noises were set up such that the noise added to the system had amplitude levels and frequency range comparable to that of Clancy's model for the entire range of voluntary muscle contraction from 0 to 100%. Figure 4 shows the simulated EMG for an example control signal in the time domain. Figure 5 compares the frequency domain behaviour of rectified and low-pass filtered signals from Clancy's EMG model and the EMG approximation used in this thesis.



**Figure 5: Comparison of the power spectral densities of Clancy's EMG model and the proposed EMG model**

### **3.3 Device part of the HMI model: virtual dynamics**

The virtual dynamics of the model refers to the mapping between the input signal from the user and the corresponding output of the device. For the model proposed here, the input from the user is the processed amplitude of the EMG signal, and the device produces a reaching trajectory in response to it. The mapping between the myoelectric EMG signal and the output velocity of the prosthesis was defined using a first-order dynamic equation as,

$$a_1 x(t)^{b_1} + a_2 (EMG(t))^{b_2} = \dot{x}(t), \quad (3.8)$$

where:

$x(t)$  is the displacement of the end effector at time  $t$ ,

$\dot{x}(t)$  is the velocity of the end effector at time  $t$ ,

$a_1, a_2, b_1, b_2$  are tunable parameters of the device.

The term  $EMG(t)$  here refers to the amplitude of muscle contraction estimated from the smoothed measurement of input EMG signal as a mean rectified value over a measurement window that corresponds to time  $t$ . The parameters  $a_1, a_2, b_1$  and  $b_2$  are varied to produce different mappings between the input signal and the device output and are hence referred to as the virtual dynamic parameters or the VD parameters. Here,  $a_1$  and  $a_2$  are the relative weights of the position and the input signal, while  $b_1$  and  $b_2$  refer to their exponents or powers.

The equation was initially defined as a more conventional differential dynamics equation, with the forcing term on the right-hand side and coefficients and powers on the velocity term rather than the EMG term as shown below,

$$a_1 x(t)^{b_1} + a_2 (\dot{x}(t))^{b_2} = EMG(t). \quad (3.9)$$

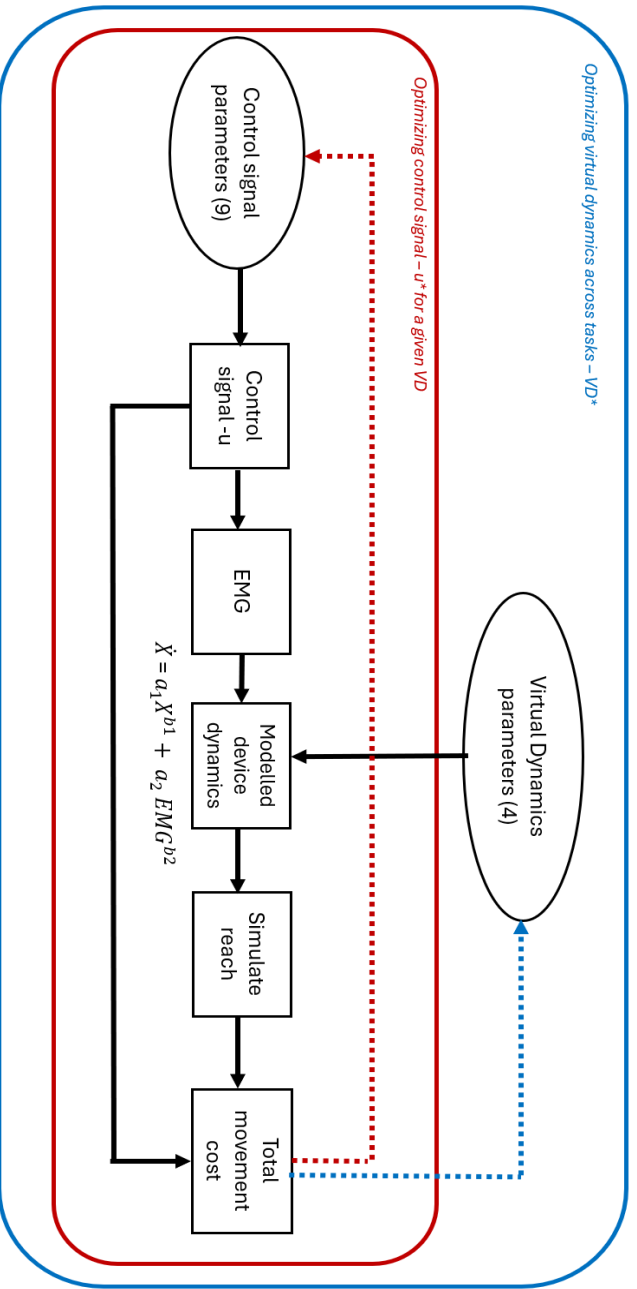
However, that convention required the optimization to solve for the  $n^{\text{th}}$  roots of the equation for a given EMG signal, which was computationally expensive and required considerable CPU time. Accordingly, the equation was rearranged into the more computationally efficient form shown in (3.8). Equation (3.8) also provides the flexibility to describe mappings that have non-integer exponent values for the position and the EMG terms.

### **3.4 Structure of the full HMI model**

The full HMI interaction model consisted of several optimization parameters that affected the total cost or the objective function value. The human model was expected to produce the optimal control intent for any given VD setting to mimic the behaviour of real users with sufficient experience using an HMI device. Because the control intent was modelled as a spline, the parameters of the spline became the optimization variables that were chosen to minimize the objective of the total cost to the user. Nine parameters that were modified to change the shape of the spline. The device part of the model had tunable VD parameters that determined the dynamic behaviour of the human-machine interaction. There were four such parameters used in this model, resulting in a total of 13 variables that could be adjusted to minimize the total cost that the internal model of the user would perceive.

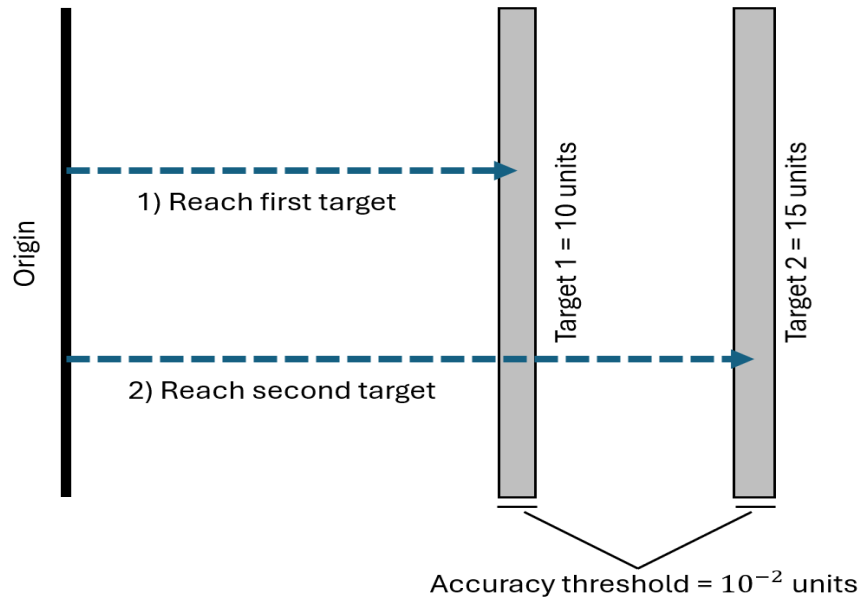
#### **3.4.1 Hierarchical optimization**

It was found that framing this optimization problem as a hierarchical structure in which the optimal control signal parameters and the optimal VD parameters are identified in a two-step process was computationally more efficient than trying to identify the optimal values of all 13 parameters simultaneously. The hierarchical optimization was set up such that the optimal control signal parameters for a given VD parameter and preference vector were identified first. Following this, the optimal VD parameters were determined for the task given the user's preference vector and the calculated optimal control signals for each of the tested VD parameters. The high-level schematic of this hierarchical optimization structure is shown in Figure 6.



**Figure 6: Schematic model of human interacting with a myoelectric prosthesis to perform a reaching task.**

### 3.4.2 Modelled task



**Figure 7: Schematic of the modelled task. Each trial of the task consists of reaching two targets.**

The full HMI interaction model was used to simulate a target-reaching task in which a prosthesis user produced myoelectric signals to move the end effector of a simulated prosthetic device to the desired target location. The human model optimization attempted to mimic a scenario in which the prosthesis user is given sufficient time to learn the behaviour of a given device and produce the best signals they can to control the device and perform the desired action. The VD optimization part of the model attempted to mimic a scenario in which the prosthesis user is given multiple devices (each with a different input-output mapping) to choose from and is asked to select the device that gives them the least cost when performing a specific task.

To ensure that the optimization solver doesn't select VD parameters that are fine-tuned or overfitted to be useful only for specific target positions, it was important to include at least two different target positions that the user model attempted to reach with each given VD parameter as shown in Figure 7. Including an ensemble of targets as a part of each reaching task would make the model more generalizable, but the number of targets within each task was kept to two to strike a balance between the generalizability and the computational time required to simulate each user preference value and calculate the corresponding VD parameters.

The two targets for the reaching task were set at 10 and 15 units from the origin. The same VD parameters were used for both target locations within a reaching task. For each target, the spline parameters were optimized to represent the optimal control signal from the user. Once the optimal control input had been determined for both targets, the total cost for the given virtual dynamics was calculated.

### **3.4.3 Cost calculation for each evaluation**

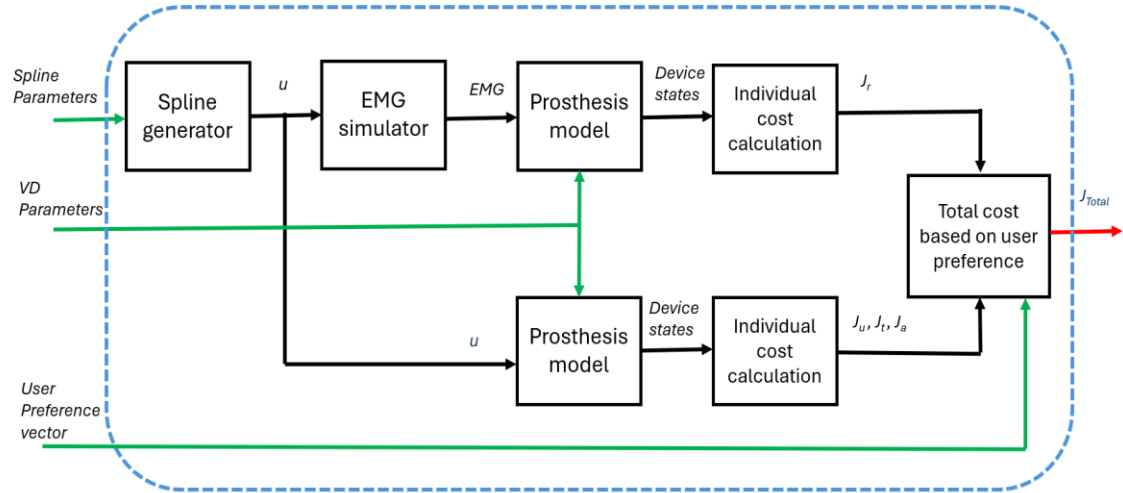
As described earlier, the goal of any optimization algorithm is to minimize (or maximize, depending on the problem) the objective function or the cost by identifying the input parameters that result in the minimum objective function value. This is done by systematically selecting input parameters and evaluating the objective function each time. In the case of the proposed HMI interaction model, the objective function value for any set of input parameters was calculated as shown in Figure 8.

The optimization algorithm could systematically pick the spline and VD parameters that are shown using the green inputs in Figure 8. For any given control signal and VD parameter input, the corresponding control signal and estimated EMG response were used to calculate the individual costs. The costs of effort and time depend only on the selected control signal and were hence calculated without adding noise to mimic the EMG response. The endpoint accuracy, stopping constraint, and initial velocity penalty ( $C_1$ ,  $C_2$ , and  $C_3$  as shown in Equation 3.7) were approximated to be the position accuracy, the stopping velocity error, and initial velocity spikes that would occur in the absence of noise and were calculated based on the control signal. The reliability cost was based on the variance of the endpoint position and hence needed to account for the stochastic and noisy nature of the EMG signals. The endpoint variance was estimated by simulating the reach 20 times using randomly generated EMG responses for the same control signal input. It was found that 20 simulations were sufficient to get a stable estimate of the endpoint variance.

If any of the candidate solutions included parameters that were outside the permitted range (See Section 3.5.5), or had endpoint position errors<sup>1</sup> greater than  $10^{-2}$ , a penalty was added, and the total cost was reset to  $10^{30}$  to ensure that such candidate solutions were always discarded.

---

<sup>1</sup> The accuracy tolerance of  $10^{-2}$  units was selected to provide room for numerical precision errors in the average accuracy calculation.



**Figure 8: Block diagram indicating how the total cost is calculated within the simulation. The green lines indicate the input (user preference vector, simulated neural drive parameters, and simulated VD parameters) and the red lines indicate the output (total cost).**

### 3.5 Optimization within the HMI interaction model

The HMI interaction model and the parameter optimization were implemented in MATLAB (The MathWorks, Inc., Natick MA). The hierarchical HMI interaction model was optimized using two independent levels of the Particle Swarm optimization algorithm. The justification for using an optimization algorithm that can only solve a scalarized single objective problem is that we were interested in understanding how the user's cost preference levels affect their motor commands and understanding if tuning the dynamics of a device according to the user's preference benefits them. The user's cost preference

level was directly incorporated into the objective function and was used to scalarize the individual costs.

### **3.5.1 Particle swarm optimization algorithm (PSO)**

Particle Swarm Optimization (PSO) is a population-based heuristic optimization technique developed by Eberhart and Kennedy [91]. This algorithm is based on the social behaviour of a flock of birds or a school of fish and works on the principle of self-organization. The significance of the swarm is that these systems can achieve a much higher level of intelligence, which cannot be reached independently by any of the individual units.

The PSO algorithm mimics the behaviour of a swarm moving through a search space in search of the point with the minimum objective function value. The optimization parameters are represented as the coordinates of the search space. The population of candidate solutions is referred to as particles. These particles are moved around in the search space according to simple mathematical formulae relating to the particle's position and velocity. Each particle keeps track of two special coordinates in the hyperspace, called the personal and the global best positions, that have been explored by the swarm so far. The best solution achieved by the particle up to the current iteration is called the "personal best." The overall best location found by any particle in the swarm is called the "global best."

PSO is a derivative-free algorithm, which is well suited for optimizing the HMI interaction problem due to its ability to handle bounded constraints and non-smooth or noisy cost functions. The objective function evaluation for this specific problem is time-consuming and noisy (as discussed in Section 3.4.3), making the objective function derivative

calculation required for a classical optimization technique unreliable and impractical to compute.

The particles are first initialized with a collection of random positions and velocity vectors that represent the particle's current solution and its moving direction. With each iteration of the algorithm, the population continues to search for the coordinate point that corresponds to the minimum objective value.

The individual candidate velocities and positions at each iteration are updated based on three components:

- (i) **Inertial component:** This component adds inertia to the update equation and ensures that one component of the velocity update for the next iteration continues in the same direction as the current velocity.
- (ii) **Cognitive component:** This component guides the particles toward the best position they have personally found so far.
- (iii) **Social Component:** This component guides the particles toward the best position found by the entire swarm so far.

These three components are combined linearly with the help of inertial, cognitive and social coefficients that determine the extent to which each component influences the next step. Random variables are used to introduce stochasticity in the algorithm. The particle update equations are represented as follows,

$$v_i^{k+1} = w^k \cdot v_i^k + c_1 \cdot r_1 \cdot (P_i^k - \chi_i^k) + c_2 \cdot r_2 \cdot (G^k - \chi_i^k), \quad (3.10)$$

$$\chi_i^{k+1} = \chi_i^k + v_i^{k+1}, \quad (3.11)$$

where:

$\chi_i^k$  – refers to the coordinate position of the  $i^{\text{th}}$  particle of the swarm at the  $k^{\text{th}}$  iteration of the algorithm

$v_i^k$  – refers to the velocity of the  $i^{\text{th}}$  particle in the swarm at the  $k^{\text{th}}$  iteration of the algorithm

$r_1$  and  $r_2$  – refer to uniformly distributed random numbers chosen between 0 and 1.

$w^k$  – refers to the inertial coefficient at the  $k^{\text{th}}$  iteration of the algorithm.

$c_1$  and  $c_2$ - refer to the cognitive and social coefficients, respectively, of the algorithm.

$P_i^k$ - is the personal best position found by the  $i^{\text{th}}$  particle by the  $k^{\text{th}}$  iteration

$G^k$ - is the global best solution found by the swarm by the  $k^{\text{th}}$  iteration of the algorithm.

### 3.5.2 Hierarchical implementation of the PSO algorithm

Two levels of particle swarm optimization were used to independently optimize for the user control intent and the virtual dynamics of the device. The outer-level PSO optimized the virtual dynamic parameters  $\{a_1, a_2, b_1, b_2\}$  to minimize the total cost of reaching the two targets (at 10 and 15 units from the origin) using the optimal control signal  $u^*$  from the human part of the HMI interaction model for each target reach. The optimization function for the outer-level PSO could be represented as:

$$\min_{\{a_1, a_2, b_1, b_2\}} \sum_i J_{Total} |_{(\alpha, \underline{u}_i^*)}, \quad i \in \{1, 2\}, \quad (3.12)$$

where:

$\alpha$  - refers to the user preference level,

$\underline{u}_i^*$  - refers to the optimal control signal identified to minimize the cost of reaching the  $i^{\text{th}}$  target using the current VD parameter settings, and

$J_{Total}|_{(\alpha, \underline{u}_i^*)}$  - refers to the total cost of reaching the  $i^{\text{th}}$  target using the optimal control signal when the user preference level  $\alpha$  is known.

The arguments that minimized the expression in Equation (3.12) correspond to the optimal virtual dynamic parameters. In this document, the dynamic system that corresponds to this optimal set of VD parameters is referred to as VD\*.

Equation (3.13) represents how VD \* is obtained as,

$$VD^* = \underset{\{a_1, a_2, b_1, b_2\}}{\text{arg min}} \sum_i J_{Total}|_{(\alpha, \underline{u}_i^*)} \cdot \quad i \in \{1,2\} \quad (3.13)$$

The optimal control signal  $\underline{u}^*$  for each target was identified by the inner-level PSO, in which the objective function to be optimized was the total cost of reaching a specific target.

The optimization expression for the inner-level PSO can be represented as shown by:

$$\underset{\{u_i \text{ parameters}\}}{\text{min}} J_{Total}|_{(i, VD, \alpha)}, \quad i \in \{1,2\} \quad (3.14)$$

where the total cost of the  $i^{\text{th}}$  target was calculated based on the user's preference level and the VD dynamics tested by the outer-level PSO.

The optimal control signal of a specific reach can be represented as:

$$\underline{u}_i^* = \Psi \left( \underset{\{u_i \text{ parameters}\}}{\text{arg min}} J_{Total}|_{(i, VD, \alpha)} \right), \quad (3.15)$$

where the operator  $\Psi ( )$  represents the transformation function that maps the optimal spline parameters to the optimal simulated neural control trajectory.

### 3.5.3 Special parameter implementations within the PSO

Some parameters, like the inertial coefficient of the update equation and the maximum velocity of the particles within the swarm, were empirically set to improve the

convergence rate of the algorithm and to reduce any instability in the progression of the algorithm.

### **Dynamics of the outer-level inertial coefficient**

The inertial coefficient  $w^t$  of the outer-level PSO was set to exponentially decrease over time, as follows,

$$w^{k+1} = 0.99w^k. \quad (3.16)$$

Because the inertial coefficient of the algorithm allowed the individual particles within the swarm to explore the search space, setting an exponentially decreasing inertial coefficient ensured that the particles within the swarm performed more exploration of the search space at the beginning of the optimization and focused on exploiting the solutions already known to the swarm over time. The initial value of the inertial coefficient was set to one.

### **Dynamics of the inner-level inertial coefficient**

The inertial coefficient for the inner-level optimizer was set to vary in an adaptive manner loosely based on the adaptive staircase algorithm in psychophysics [92] or the damping factor of the Levenberg Marquardt [93] regularization algorithm. The swarm increased the exploratory behaviour if a better global solution was found. But if the global best solution of the swarm did not improve after several iterations, the exploratory behaviour of the swarm was decreased. The absence of a better global best solution indicates that the swarm had already identified the local region about the true minimum, and it would be best to focus on the local search about the known solution. A counter was used to identify when

the inertial coefficients must be changed, as shown below in **Algorithm 1**. The inertial values were clamped such that they were set within the range of [0.1, 1.5].

---

**Algorithm 1:** Adaptive inertial coefficient,  $w^k$

---

Step 1 : **SET** counter = 0

Step 2 : **For**  $k = 1$ : Maximum number of iterations

Step 2.1 : **If** better *globalBest*  
solution is found:

$counter = \max(0, counter-1)$

**Else:**

$counter = counter + 1$

Step 2.2 :  $w^{k+1} = \begin{cases} 2w^k & \text{If } counter \geq 2 \\ w^k & \text{If } 2 < counter < 5 \\ w^k/2 & \text{If } counter \geq 5 \end{cases}$

---

**Maximum velocity of the particles**

To ensure that the optimization iterations were stable and didn't show oscillations in their objective value, a maximum velocity limit was set for the high-level optimizer. For the outer level PSO, the maximum velocity a particle could take at any iteration was limited to  $1/20^{\text{th}}$  the range of the search space in each dimension. It was found that clamping the positions of the inner-level PSO candidates such that the position coordinates are within the search space was sufficient to prevent oscillation. Hence velocity clamping was not implemented for the inner PSO loop.

#### **3.5.4 Hyperparameters of the algorithm**

The performance of any heuristic optimization algorithm depends heavily on the hyperparameters. These parameters were tuned to improve the computational speed and convergence rate of the defined problem. The tuning was performed iteratively, starting with the algorithm's default parameters, and each hyperparameter was adjusted based on small-scale pilot studies. Table 3 shows the list of the hyperparameter values used for optimizing the HMI interaction model. The selected hyperparameter values for the two levels of the PSO algorithm helped improve the overall convergence rate of the algorithm.

The design variable values of the of the first iteration of particles was selected using a uniformly distributed random number within the permitted range of each design parameter. The entire optimization procedure was repeated 3 times with different initial random seeds to ensure that the results were stable and statistically reliable.

**Table 3: Hyperparameter values of the algorithm**

	<b>Outer Level PSO</b>	<b>Inner Level PSO</b>
<b>Population size</b>	20	20
<b>Maximum iterations</b>	60	500
<b>Maximum runs</b>	3	1
<b>Inertial coefficient type</b>	Exponential decay	Adaptive adjustment
<b>Inertial coefficient specification</b>	$w(0) = 1$	$w \in [0.1, 1.5]$
<b>Cognitive coefficient</b>	1	1.5
<b>Social coefficient</b>	2	2.75

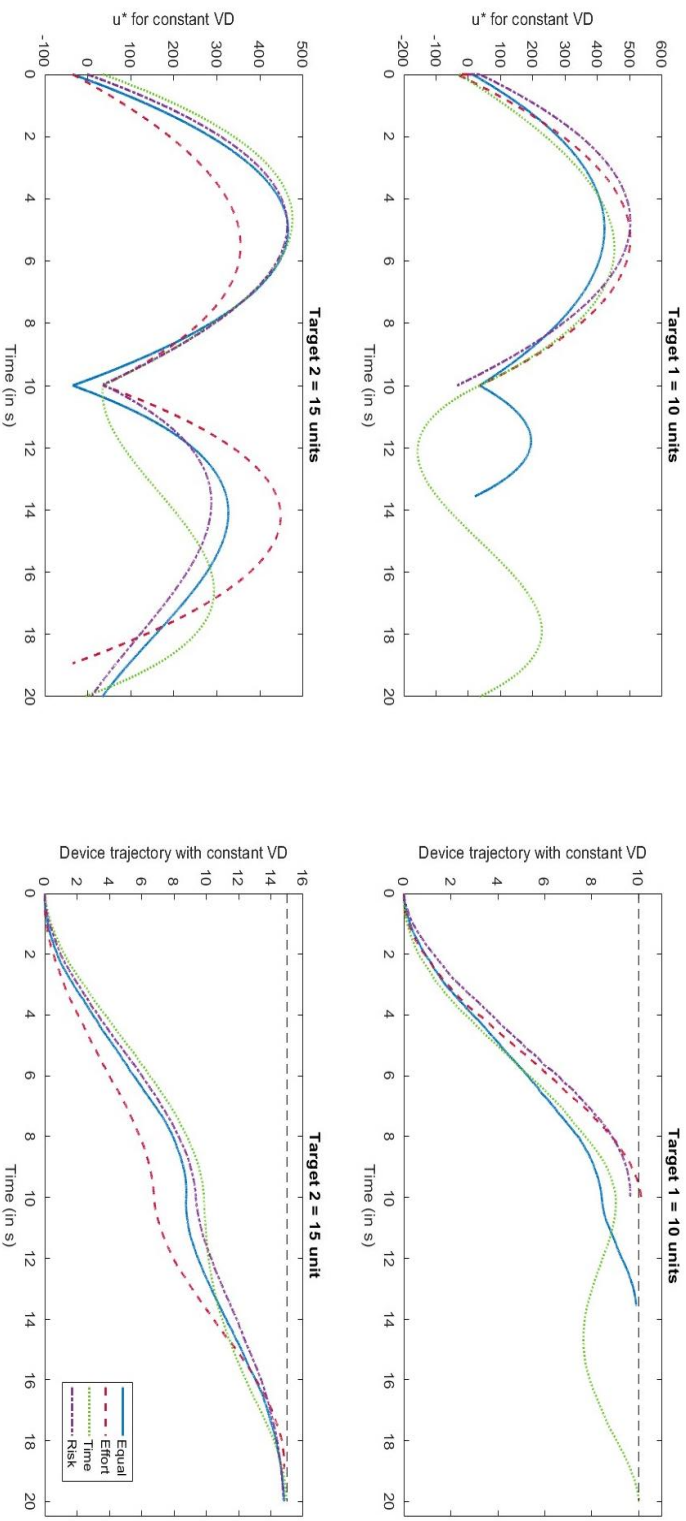
## 4. Chapter 4: Results

To understand how the user preference for individual costs affects the optimal movement produced and the subsequent virtual dynamics, four different relative weight scenarios were simulated. Tests were conducted for situations when the user cares more about each one of the three cost functions- effort, time, and reliability, and when they show an equal preference for the three costs. The HMI interaction model's results were analyzed by studying three major outputs: the optimal control signals of the user model, the optimized virtual dynamics of the machine interface, and the overall trajectory profile of the device.

### 4.1 Influence of the user preference level on the generated motor commands

The first step of validating the created model was to verify that the spline approximation of the motor commands sufficiently described how a user would interact with a device, and that it reflected how a user's cost priorities influence the simulated control signals that correspond to the motor commands and result in the simulated EMG. A proportional velocity control dynamic was used to model the prosthesis dynamics. The  $a_2$  gain parameter was determined by optimizing the  $a_2$  VD parameter for a linear velocity control scheme at the equal cost preference setting which resulted in:

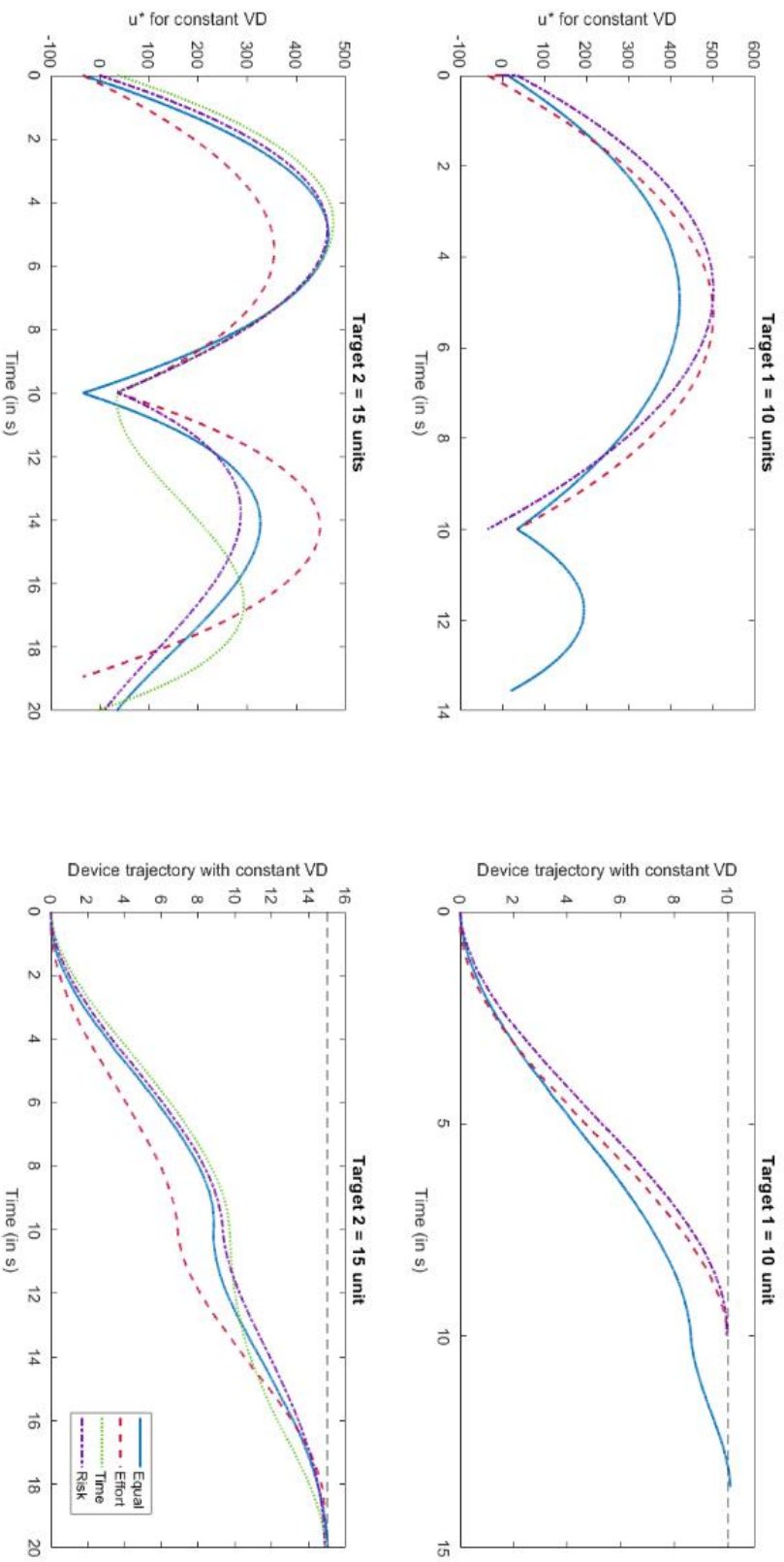
$$a_2 EMG = \dot{x}, \text{ where } a_2 = 0.003. \quad (4.1)$$



**Figure 9: Understanding the influence of the user preference level on the human part of the HMI interaction model – through the generated control signal and the device trajectory for a device with fixed VD. The top row indicates the produced optimal user control signal (on the left) and simulated device trajectories (on the right) for the first target. The bottom row shows the produced optimal user control signal (on the left) and the corresponding device trajectory for the second target. The dotted lines on the second column of panels indicate the desired target position.**

Figure 9 shows the simulated user control signals and the corresponding device trajectories for the four user preference levels when reaching the two targets. The dashed lines in the trajectory plots indicate the true target position or the goal. The four different user preference conditions that were simulated correspond to situations when the user shows equal preference to the three individual costs of effort, time, and reliability; and when they care 5 times as much about each individual costs.

It is important to note that the optimization for the ‘time prioritized’ user preference category did not converge to a feasible solution even after 18 hours of computation and selected a suboptimal candidate that required more movement time than any of the other user priority levels for the first target reaching scenario. The simulations were not repeated with a larger population size, increased iterations for convergence or any other modifications to the hyperparameters to ensure that all model simulations were consistent. Hence, the results from the time-priority user preference condition will not be discussed in any comparisons or validation tests performed in the rest of this document. Figure 10 shows the simulated optimal control signals and device trajectories for the two target reaches, after omitting the results from the time prioritized setting for the 10 unit target reach in which the PSO failed to converge to a solution.



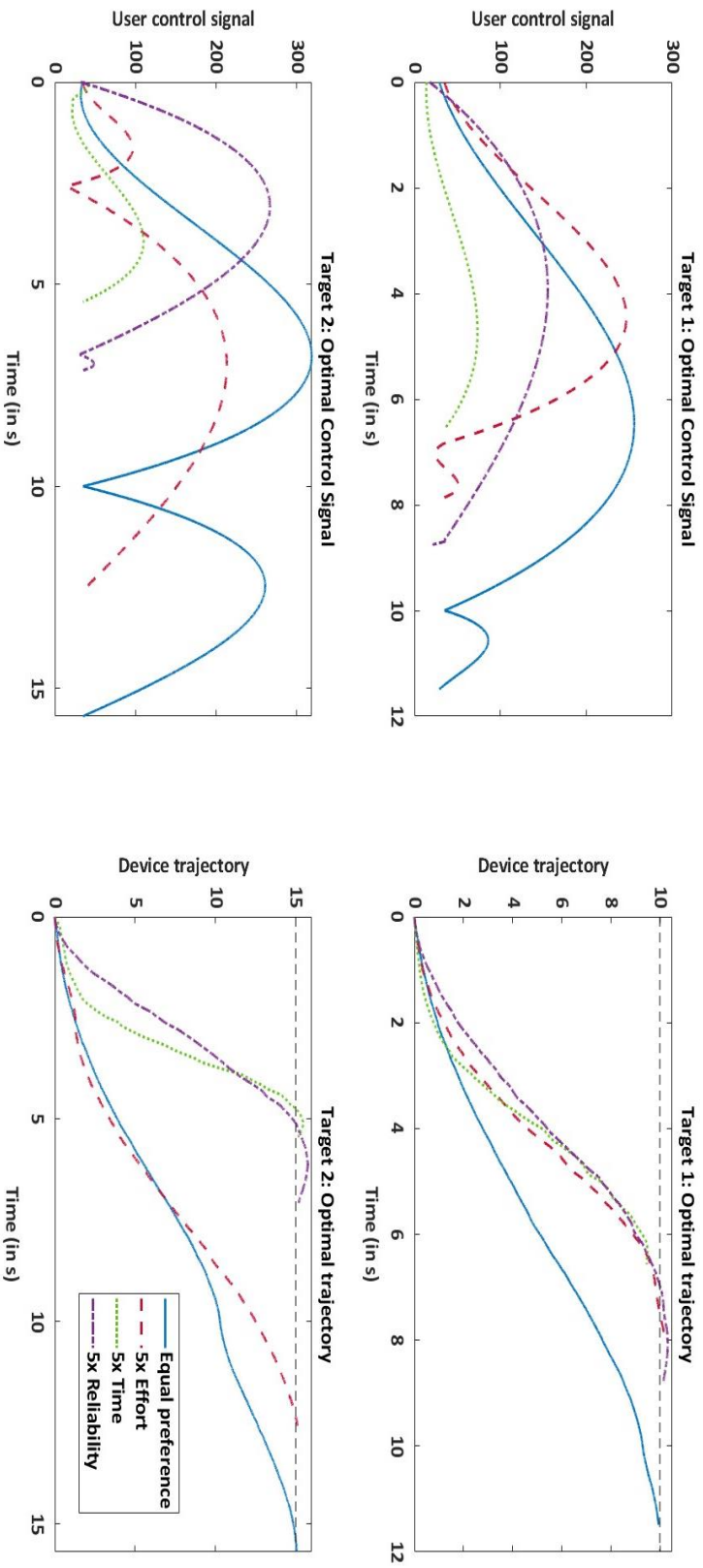
**Figure 10: Understanding the influence of the user preference level on the human part of the HMI interaction model — through the generated control signal and the device trajectory for a device with fixed VD. The time prioritized setting for the 10 unit target reach is omitted as the results of the optimization failed to converge.**

In the three user-preference scenarios of interest (Equal preference, Effort prioritized, and Reliability prioritized), the simulated control signals follow the expectations and observations in the literature. When reliability is prioritized, the model produced control signals whose magnitudes were smaller towards the end of the movement or in the second sub-movement. This behaviour aligns with the results of the study [94] explaining the stereotypy of eye saccades by minimizing the endpoint variance of the movement (equivalent to the cost of reliability or risk) in the presence of multiplicative noise. Due to the multiplicative nature of the noise in the signal, it is better to make larger contractions earlier in the movement when corrections are possible rather than later in the movement when the multiplicative noise could increase the variability of the movement endpoint. Though not easily observable from the image, the integral of the control magnitude is the smallest in the effort prioritized setting compared to the other user preference levels. This observation supports that our implementation of the multi-objective CMC framework follows the expected user behaviour under different priority conditions.

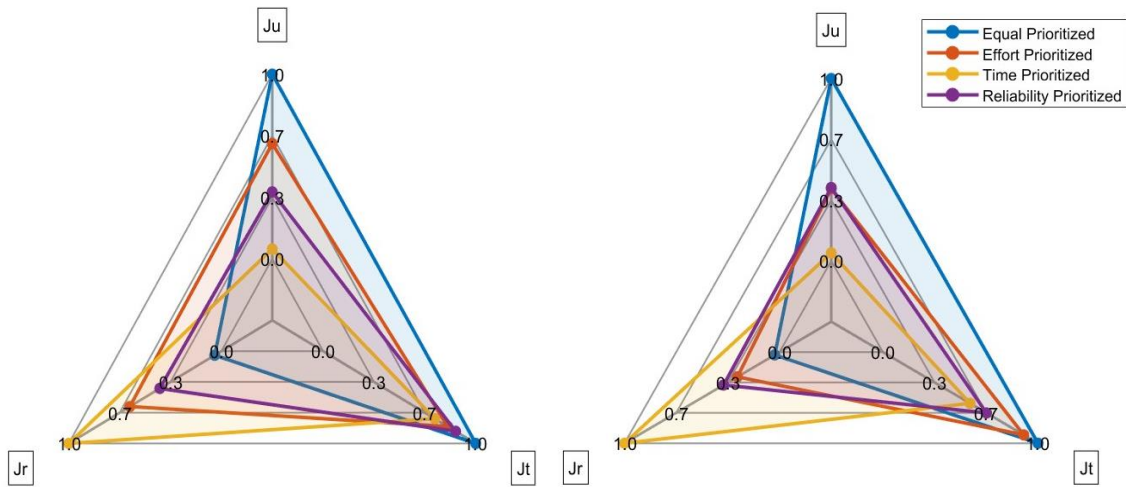
#### **4.2 Effect of tuning both the human and the device parts of the HMI interaction model according to the user preference level**

To investigate the impact of using a tunable device for a human-machine interaction task, another layer was added to the model to include the dynamic parameters of the device as optimization variables that can be tuned to minimize the total cost. Figure 11 shows the results of this hierarchical model when the user control signals, and the VD parameters of the device are both optimized at four different user preference levels. The figure shows the optimized control signal produced by the user model to reach the two targets and the overall

device trajectory generated when these optimal signals were used as input to a device with the optimal virtual dynamics. Figure 12 shows how the individual costs of effort, time and reliability varied for the different user preference settings that were simulated. The fact that the 4 scenarios resulted in different optimum virtual dynamics, different optimum user input signals, and different reach trajectories indicates that the user preference for cost in a task plays an important role in determining how the movement should occur.



**Figure 11: Optimal control signal and device trajectories when the device dynamics is optimized according to the user preference level. The top row indicates the produced optimal user control signals (on the left) and simulated device trajectories (on the right) for the first target. The bottom row shows the produced optimal user control signal (on the left) and the corresponding device trajectory for the second target.**



**Figure 12: Normalized spider plots indicating the individual costs of effort, time and reliability for the 4 user priority settings that were simulated. The image on the left corresponds to the costs to reach the first target, and the image on the right corresponds to the costs to reach the second target.**

Within the simulated HMI interaction model, when one individual cost was prioritized, the optimization process adjusted the values of both the human-model motor command parameters and the virtual dynamic parameters of the device to minimize the total cost. In a practical setting, if HMI devices were allowed to have tunable dynamic parameters, the user's biological internal model would inherently adjust the user's control strategy to reflect their preference in costs, and an independent controller on the device could be used to modify the parameters of the device or its overall closed-loop behaviour such that the total cost to the user was minimized for a given cost preference setting.

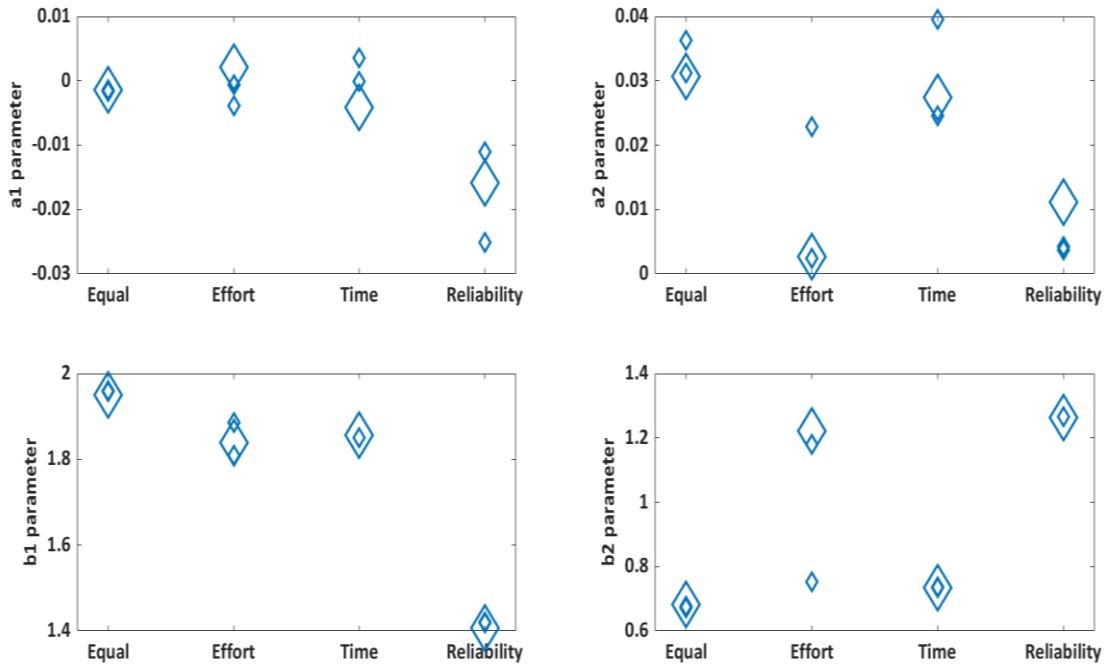
Similar to the results when fixed device dynamics were used to understand the influence of the user preference level on the user's control strategy (Figure 9), it was observed that the

user preference category appreciably impacted the shape and the magnitude of the simulated motor command signals. For example, when the user cared more about the cost of time involved in making a movement, the generated control signal had the least movement time with larger endpoint accuracy and velocity constraint errors, and a set of VD parameters was chosen such that the smaller control signal from the user still gets the device as close as possible to the desired target. But if the user cared more about the risk involved in making the movement, the generated control signal had a much smaller magnitude towards the end of the movement to avoid large multiplicative noise errors that increase the endpoint variance of the reach.

The optimal control signal profile produced by the user was found to be asymmetric and was composed of 2 distinct peaks for the cases in which the user preference was set to be higher for the reliability of movement, and when the preference was set to be equal. These peaks correspond to submovements, which are separate phases of acceleration and deceleration within a single movement trajectory [95]. The emergence of submovements in this simulated open-loop reaching task was a very interesting phenomenon that aligned well with the predictions of [96], [97]. It should also be noted that the amplitude of the first peak was always slightly higher than that of the second one. This difference in magnitude indicates that the cost for large control signals is less at the beginning of the movement than towards the end of the movement.

Figure 13 shows how the selected VD parameters varied for the different user preference levels that were simulated. Because a heuristic algorithm was used for the optimization, a formal proof or verification for the true optimality of the identified solutions is not possible.

Three candidates with the least total cost were chosen from the pool of 180 particles at the last iteration of the optimization process. The VD parameters of these three candidate particles were plotted for each simulated user preference level to ensure that the variability of the identified VD parameters was sufficiently small and that there was a distinct shift in the VD parameters at each user preference level. Each subplot within Figure 13 shows the value of one of the VD parameters ( $a_1, a_2, b_1, b_2$ ) and the candidate with the best cost value is highlighted with a marker of a larger size.



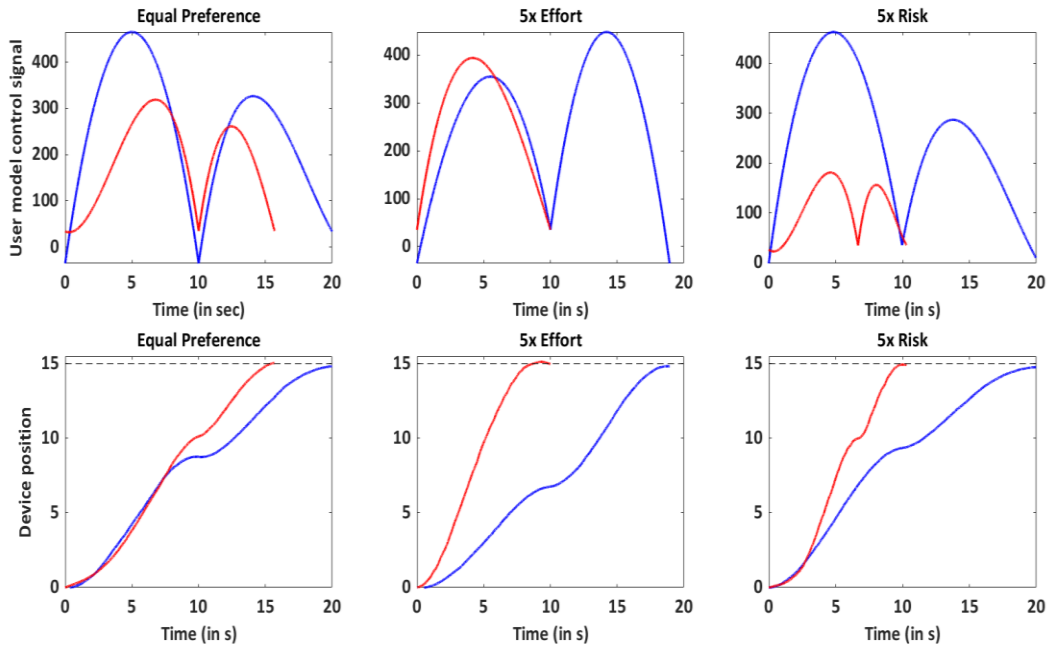
**Figure 13:** In the clockwise direction from the top left-hand corner, each subplot shows the optimal VD values of the ( $a_1, a_2, b_2, b_1$ ) parameters for each of the simulated user preference levels. The parameter values of the three candidates with the lowest total cost values are plotted for each condition, with the best candidate identified as the one with the larger marker size.

### 4.3 The importance of tunable device dynamics

Conventional prosthesis control maps the EMG signal only to the velocity of the device, but most of the optimal virtual dynamic solutions found using our HMI interaction model used a combination of both position and velocity-based control. This outcome suggests that conventional prosthesis controllers with position-only or velocity-only control schemes might produce suboptimal movements. To analyze this further, a one-to-one comparison was made between a device model with a velocity control scheme and a device model with a blended tunable device dynamic scheme. Only the user preference levels of Equal preference, 5x Effort, and 5x Reliability were simulated for this comparison, as the PSO algorithm fails to identify a feasible control signal trajectory for the 5x Time user preference setting with a proportional velocity control VD for the device model.

In a clinical setting, the gain of the velocity control mapping scheme of the prosthesis is hand-tuned by the clinician to best suit each prosthetic device user. To compare the velocity control scheme with the proposed nonlinear blended control scheme in this project, the gain of the proportional control scheme was set by optimizing just the  $a_2$  VD parameter for the Equal preference setting. The blended control scheme continued to optimize all the VD parameters at each user preference level. The simulated optimal motor control strategies, the overall reach trajectories, the total cost, and the individual costs of effort, time, and reliability for the two control schemes were compared at the different simulated user preference levels. Figure 14 shows the simulated optimized control commands from the user and the corresponding device trajectories for the two modelled device mappings at different user preference levels for the 15 unit target reaching task. The plot confirms

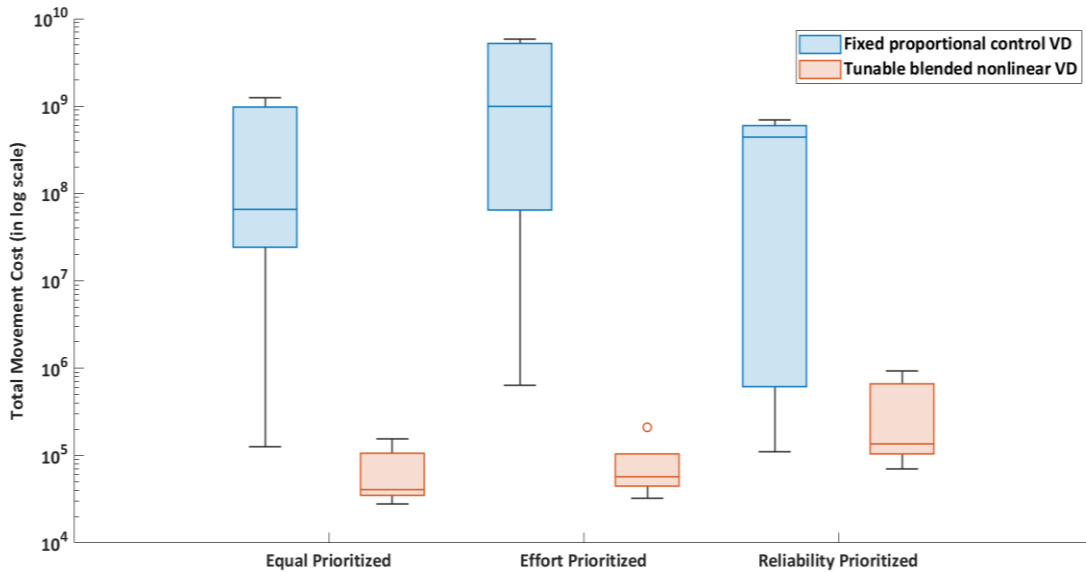
that using a tunable device whose parameters can be adjusted according to the user's preference level can simultaneously reduce the required movement time to make a reach and the magnitude of the control signal that might be required from the user.



**Figure 14: Comparing the control signals and device trajectories when using a device with non-tunable proportional velocity control mapping (in blue) and a tunable device dynamic mapping (in red) to reach the second target that is 15 units from the origin.**

The total movement cost of reaching the two targets using the tunable and the non-tunable device dynamics when the user part of the model generates the optimal motor commands is shown in Figure 15 as a boxplot with a semilog scale. It was observed that even when the user produces the best motor commands to control the device at each user preference level, the tunable device dynamics could significantly reduce the total cost involved in the movement when compared to the conventional non-tunable device dynamics with the

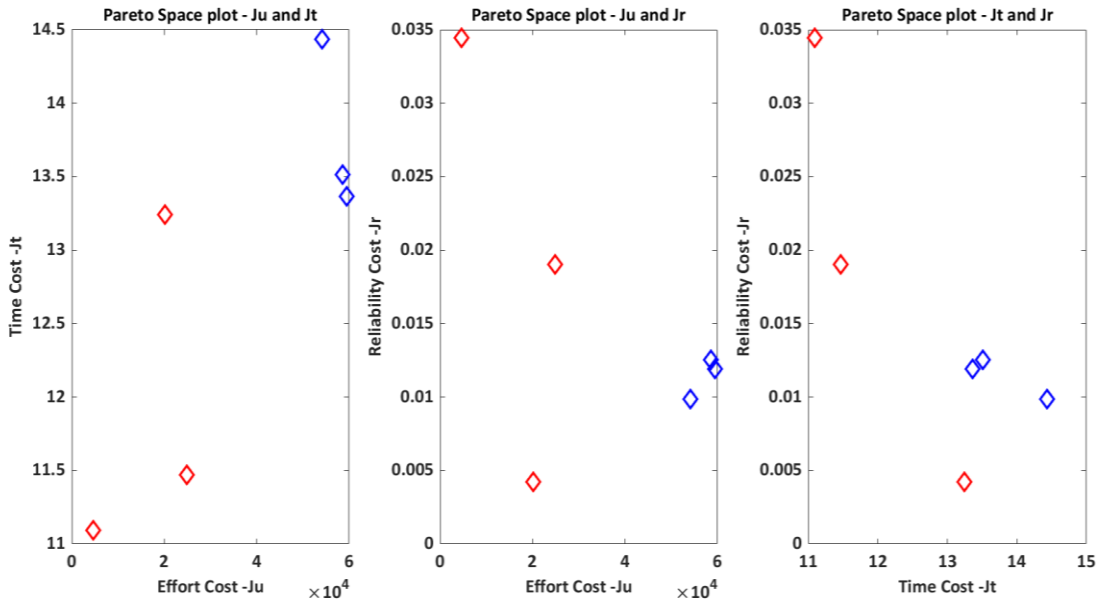
standard proportional velocity control mapping. At the equal preference setting, the cost of moving with a tunable VD was slightly greater than that of using the non-tunable device. That could be attributed to the heuristic optimization algorithm converging to a solution before sufficiently exploring the search space.



**Figure 15: Comparison of the total movement cost at different user preference levels between the tunable and the non-tunable control dynamics**

The individual cost of effort, time, and reliability of the movement made with the two device models is examined in Figure 16. Each marker on the plot corresponds to a solution generated for a specific user preference level – equal user preference, 5x effort, and 5x reliability. The red markers represent the costs of movement with tunable device dynamics; the blue markers represent the costs of movement when a device with non-tunable proportional velocity control device dynamics was used. The objective space visualization helps highlight the tradeoff between individual cost values and identifies the structure of the obtained Pareto solutions.

In the case of Figure 16, the objective space visualization also helps identify the dominance of different solutions. All the solutions from the non-tunable device dynamics are dominated by solutions obtained when using tunable devices, clearly indicating that allowing devices to have tunable dynamic parameters can significantly benefit the users when interacting with HMI devices.



**Figure 16: Pareto space visualization of the individual costs comparing the tunable device dynamics with the conventional velocity control device dynamics.**

## **5. Chapter 5: Discussion**

The first aim of this study was to create a model of a human-machine interaction in which a person uses myoelectric signals from their muscles to control an upper limb myoelectric prosthesis to reach from one point to another. This aim of the study was implemented by creating a mathematical framework that incorporated ideas from the fields of computational neuroscience of motor control, evolutionary multiobjective optimization, and control theory. The proposed HMI interaction model consisted of two parts: a user model that generates EMG-like signals that are optimized according to the cost priorities of the user, and a device model that takes these EMG-like signals as input and produces a reach trajectory based on the defined dynamics of the device model.

The second aim of this study was to understand if tuning the parameters of the device dynamics so that they are optimized for the user's different cost preferences was beneficial. To identify the best device dynamics, the device dynamics are represented as virtual dynamics whose parameters were optimized to minimize the total cost according to the user's cost priorities.

### **5.1 Verification of specific aim 1**

Once the full interaction model was built, the user part of the proposed HMI interaction model was tested with conventional fixed device dynamics to ensure that the generated user control signals were meaningful and appropriate for the selected cost preference level. The model's generated control signal trajectories were tested at 4 distinct cost preference levels. The algorithm always optimized for the total cost, so the individual costs of effort, time, and reliability were weighted to indicate the user preference.

The optimization solutions from the time-prioritized setting failed to converge to feasible solutions within the allotted computational time of 18 hours per run of the algorithm, so the motor command solutions obtained from time-prioritized setting were discounted from the discussions of this study. The optimization algorithm could have failed to converge to a solution due to reasons like the initial values of the particles of the PSO, or the algorithm getting stuck in infeasible regions of the design space. It is possible that repeating the optimization for this user priority with a different set of hyperparameters or random seed values could have improved the convergence rate of this specific simulation. But the adjustments were not made to ensure that a consistent set of hyperparameters was used for all the simulations.

The generated motor commands at the other cost priority settings appeared to be meaningful and followed the expected norms in terms of the relative amplitude, duration, and shape of the generated control signal. The motor commands generated in the effort-prioritized setting always had the least cumulative squared control signal amplitude compared to the other two user settings for both the target reaches. The generated control signals for the reliability prioritized setting had smaller control signal amplitude towards the end of the movement. This observation is in line with the studies [94 - 96] that explain the stereotypical movement due to the presence of signal-dependent multiplicative noise in our muscle signals. No empirical studies have yet been performed with people to establish the influence of the user cost priorities on the generated control signals, so a direct comparison of the model was not made to existing real-life data. Nevertheless, the results

from the developed model align well with the theoretical expectations and hence sufficiently describe the user behaviour at different cost priorities.

Analyzing movement generation from the perspective of a multi-objective optimization problem could yield insights into why everyone produces slightly different movements even when performing the same task. One individual may prioritize spending the least metabolic energy, whereas another may prioritize their time over the energy spent.

## **5.2 Verification of specific aim 2**

To understand the effect of the user cost priorities on the virtual dynamic parameters of the device model and to understand if the cost-priority-based tuning of a device could benefit a user, a hierarchical optimization scheme was performed to independently optimize the parameters of the user and the device models within the full HMI interaction model. The developed model demonstrated the feasibility of using a blended virtual dynamic mapping to describe any device that is a part of an HMI. The cost priorities clearly influenced the selected virtual dynamic parameters as observed in Figure 13.

The benefit of using a device that is tuned according to the user's priority setting was then studied by comparing the generated control signal, end effector trajectories, the individual movement costs, and the total movement cost to the user in two modelled scenarios:

- (i) when using a device that can be tuned at different user priority levels
- (ii) when using a device that cannot be tuned at different user priority levels.

It was clear on all accounts that an optimally tuned device dynamic significantly benefits the user in terms of the effort, time, and reliability costs at different user priority levels.

This result indicates that myoelectric users may be able to benefit from device customization that better reflects their personality and priorities regarding movement. The results also suggest that it would be beneficial to determine optimal device parameters that are best for specific tasks (that inherently prioritize different costs) performed with the help of the prosthetic device. Conventionally, if someone wants to use their prosthetic device when painting and when performing yard work, they need to adjust their input EMG signals to change the behaviour of their device. Instead, the proposed method could be used to predetermine the optimal device parameters for specific tasks when the user wants to switch from one type of task to another.

In addition, the solutions established from a tunable device model are also mathematically superior to those from a device model whose parameters cannot be adjusted for the user priorities as shown in Figure 16. The performance objective values of a tunable device always dominate the performance of a device that cannot be user adjusted, indicating that it is always better to use a tunable device dynamic when compared to a fixed device dynamic for any given user priority. This result shows the significance of the relative weights of individual costs in determining how a given task needs to be performed.

### **5.3 Significance**

This study provides a preliminary step towards understanding how an individual's preference can influence movement and the importance of device parameters in determining the outcome of a user interacting with a device. The results strongly suggest that human-machine interfaces could be customized for users depending on the type of

tasks that are performed more frequently. The proposed concept of optimal device tuning for HMI has been patented [101].

Clinical tuning of a prosthetic device usually requires the clinician to tune for abstract parameters like the device gain or the threshold values that cause a desired effect in the movement. Instead, allowing them to directly tune for the relative preference in terms of speed, energy spent, or the reliability of the device would be much more intuitive. The idea of relative tuning that has been proposed here could allow both patients and clinicians to tune the control strategy for those things that they directly care about.

The concept of optimal device tuning could also be extended beyond myoelectric prostheses to encompass other HMI devices, and the proposed optimal tuning strategy could be used to establish predetermined modes similar to the driver mode settings in a car. In addition to setting specific parameters that are better suited for different tasks and situations, parameters could be established that are customized to each individual's cost priorities.

#### **5.4 Limitations**

A heuristic optimization procedure was used to calculate the best user control signal and the optimal set of virtual dynamic mappings. Hence, it cannot be guaranteed that the solutions obtained are globally optimal. Even though it was shown that the solutions found were similar for various initial random seeds and that the variance of the identified VD parameters at each cost priority level was small, it cannot be concluded that the solutions obtained are the global minimum solutions for the given problem. Hence, the best user

control signal found by the developed model might not accurately represent the optimal signal that a human would produce to reach the given target. Moreover, the optimal virtual dynamics solution found here could similarly be the result of our optimizer getting stuck in local minima or running out of the allotted computational time of 18 hours per run.

Increasing the swarm size and the number of generations of evolution in the optimization or improved tuning of the hyperparameters of the PSO algorithm could potentially give better solutions. The solutions obtained here could also be constrained by the exploration space that was chosen. These strategies to improve the solutions of the PSO were not implemented due to the computational time required by the hierarchical structure of the model, which required the inner loop to be optimized for all possible outer loop VD parameters that were considered. A hard computational time limit of 18 hours was set for each run of the hierarchical PSO algorithm.

We acknowledge that normalizing the individual costs of effort, time, and reliability about a nominal operating point using the minimum jerk trajectory with a velocity control scheme does not generalize the normalization to other operating points. This was clearly seen in the individual costs of the obtained results, in which the range of values taken by the costs of effort and reliability were in the order of  $10^6$  units. This meant that we could not guarantee that the individual user cost priorities changed by a factor of 5, even though it was evident that changing the user priority setting forced the algorithm to improve the specific individual cost. Iteratively normalizing the individual costs such that they have comparable values at equal relative weights could improve the effectiveness of the linear scalarization approach used in the model.

The intent for the limits on the control signal amplitude was to constrain the simulation of the motor commands and EMG signals to be similar to that expected of a prosthesis user. It can be seen in Figure 11 that simulated motor command values well exceed the specified 35% limits. This was a result of an unintentional loophole in the model in which only the initial and final amplitudes of the motor commands were constrained to 35% contraction intensity. The selected slope parameters permitted the generated spline to have values outside the  $\pm 35\%$  constraints that were set. A better way to model the motor command would have been to model the signal as a trajectory with piecewise constant values at each time step. This approach was not pursued in the current work as the alternative method will greatly increase the number of optimization parameters and, hence, the complexity and the computational time required by the algorithm.

## **5.5 Future work**

The results discussed in this thesis are based on a preliminary modelling study used to demonstrate the feasibility of the proposed idea. Future work on this concept should expand the research by performing experiments with human subjects. To further enhance the model and algorithm, it would be important to improve the model by providing the user with feedback from the device output and performing a closed-loop optimization that closely resembles how humans make decisions about their movement [83]. Parts of the optimization problem could also be solved analytically to potentially optimize the virtual dynamic parameters in a real-time application for a given set of cost function weights using direct or indirect numerical optimization methods. Another option would be to find the set of all possible optimal solutions for the multi-objective optimization process and create a

look-up table that could be used to tune according to the user preference. The set of all optimal solutions is termed the Pareto set and can be found using analytical methods [102] or heuristic evolutionary optimization algorithms [103].

Another important direction would be to compare the performance of our customized virtual dynamics with the other existing models like the Kalman filter, 1 Euro filter, adaptive EMG filter, nonlinear recursive Bayesian filters, linear filters, and or kurtosis filters. We would also like to study the overall performance of our proposed optimal virtual dynamics with the clinical standard proportional velocity control and the velocity squared control [104] approaches.

## 6. Chapter 6: Conclusions

Our study aimed to develop a model of human-machine interaction (HMI) for upper limb myoelectric prosthetic control, with a focus on incorporating user cost preferences into device dynamics. Integrating principles from computational motor control, multiobjective optimization, and control theory, our approach addresses the gap in current prosthetic control methods by considering individual user preferences.

We found that user movement preferences significantly influence both control signals and the selected virtual dynamic parameters of the device. Comparing scenarios with and without tunable device parameters, we observed substantial improvements in total movement cost, effort, time, and reliability costs when using optimally tuned devices. This highlights the potential of personalized prosthetic settings to enhance movement outcomes and user experience.

This study represents a preliminary step toward understanding how individual preferences influence movement and the importance of device parameters in HMI. Our findings suggest that customization of HMIs based on user priorities is beneficial, and our proposed optimal tuning strategy could revolutionize prosthetic device control. With further refinement and experimentation, our approach has the potential to enhance the quality of life for individuals using myoelectric prostheses and other HMIs.

## Bibliography

- [1] B. Lipp and S. Dickel, “Interfacing the human/machine,” *Distinktion: Journal of Social Theory*, vol. 24, no. 3, pp. 425–443, Sep. 2023, doi: 10.1080/1600910X.2021.2012709.
- [2] P. Van Donkelaar, K. C. Siu, and J. Walterschied, “Saccadic Output Is Influenced by Limb Kinetics During Eye—Hand Coordination,” *J Mot Behav*, vol. 36, no. 3, pp. 245–252, Sep. 2004, doi: 10.3200/JMBR.36.3.245-252.
- [3] T. Ljungberg, P. Apicella, and W. Schultz, “Responses of monkey dopamine neurons during learning of behavioral reactions,” *J Neurophysiol*, vol. 67, no. 1, pp. 145–163, Jan. 1992, doi: 10.1152/jn.1992.67.1.145.
- [4] L. Resnik, M. Borgia, and M. Clark, “Function and Quality of Life of Unilateral Major Upper Limb Amputees: Effect of Prosthesis Use and Type,” *Arch Phys Med Rehabil*, vol. 101, no. 8, pp. 1396–1406, Aug. 2020, doi: 10.1016/J.APMR.2020.04.003.
- [5] C. H. Jang *et al.*, “A survey on activities of daily living and occupations of upper extremity amputees.,” *Ann Rehabil Med*, vol. 35, no. 6, pp. 907–21, Dec. 2011, doi: 10.5535/arm.2011.35.6.907.
- [6] C. Igual, L. A. Pardo, J. M. Hahne, and J. Igual, “Myoelectric Control for Upper Limb Prostheses,” *Electronics 2019, Vol. 8, Page 1244*, vol. 8, no. 11, p. 1244, Oct. 2019, doi: 10.3390/ELECTRONICS8111244.
- [7] H. Basumatary and S. M. Hazarika, “State of the art in bionic hands,” *IEEE Trans Hum Mach Syst*, vol. 50, no. 2, pp. 116–130, Apr. 2020, doi: 10.1109/THMS.2020.2970740.

- [8] F. Cordella *et al.*, “Literature review on needs of upper limb prosthesis users,” *Front Neurosci*, vol. 10, no. MAY, p. 175438, May 2016, doi: 10.3389/FNINS.2016.00209/BIBTEX.
- [9] S. H. Scott, “Optimal feedback control and the neural basis of volitional motor control,” *Nature Reviews Neuroscience 2004 5:7*, vol. 5, no. 7, pp. 532–545, 2004, doi: 10.1038/nrn1427.
- [10] E. Todorov, “Optimality principles in sensorimotor control,” *Nat Neurosci*, vol. 7, no. 9, pp. 907–915, Sep. 2004, doi: 10.1038/nn1309.
- [11] R. Shadmehr and S. Mussa-Ivaldi, *Biological Learning and Control*. The MIT Press, 2012. doi: 10.7551/mitpress/9780262016964.001.0001.
- [12] Kaisa. Miettinen, *Nonlinear Multiobjective Optimization*, vol. 12. in International Series in Operations Research & Management Science, vol. 12. Boston, MA: Springer US, 1998. doi: 10.1007/978-1-4615-5563-6.
- [13] R. J. van Beers, “Saccadic Eye Movements Minimize the Consequences of Motor Noise,” *PLoS One*, vol. 3, no. 4, p. e2070, Apr. 2008, doi: 10.1371/journal.pone.0002070.
- [14] R. N. Marshall, G. A. Wood, and L. S. Jennings, “Performance objectives in human movement: A review and application to the stance phase of normal walking,” *Hum Mov Sci*, vol. 8, no. 6, pp. 571–594, Dec. 1989, doi: 10.1016/0167-9457(89)90004-3.
- [15] L. M. Bautista, J. Tinbergen, and A. Kacelnik, “To walk or to fly? How birds choose among foraging modes,” *Proc Natl Acad Sci U S A*, vol. 98, no. 3, pp. 1089–1094, Jan. 2001, doi: 10.1073/PNAS.98.3.1089.

- [16] E. Summerside, R. Shadmehr, and A. A. Ahmed, “Vigor of reaching movements: reward discounts the cost of effort.,” *J Neurophysiol*, pp. 1–30, 2018, doi: 10.1152/jn.00872.2017.
- [17] R. Shadmehr, T. R. Reppert, E. M. Summerside, T. Yoon, and A. A. Ahmed, “Movement vigor as a reflection of subjective economic utility,” *Trends Neurosci*, vol. 42, no. 5, p. 323, May 2019, doi: 10.1016/J.TINS.2019.02.003.
- [18] R. Shadmehr and J. W. Krakauer, “A computational neuroanatomy for motor control.,” *Exp Brain Res*, vol. 185, no. 3, pp. 359–81, Mar. 2008, doi: 10.1007/s00221-008-1280-5.
- [19] S. Haar and O. Donchin, “A Revised Computational Neuroanatomy for Motor Control,” *J Cogn Neurosci*, vol. 32, no. 10, pp. 1823–1836, Oct. 2020, doi: 10.1162/jocn\_a\_01602.
- [20] M. Kawato, “Internal models for motor control and trajectory planning,” *Curr Opin Neurobiol*, vol. 9, no. 6, pp. 718–727, Dec. 1999, doi: 10.1016/S0959-4388(99)00028-8.
- [21] P. Cisek, “Internal models,” in *Encyclopedia of Neuroscience*, 2009, ch. Internal M.
- [22] E. Todorov, “Efficient computation of optimal actions,” *Proceedings of the National Academy of Sciences*, vol. 106, no. 28, pp. 11478–11483, Jul. 2009, doi: 10.1073/pnas.0710743106.
- [23] E. Todorov, “Optimal Control Theory,” in *Bayesian Brain*, MIT Press, 2006, ch. Optimal Co.

- [24] E. Todorov and M. I. Jordan, “Optimal feedback control as a theory of motor coordination,” *Nat Neurosci*, vol. 5, no. 11, pp. 1226–1235, 2002, doi: 10.1038/nn963.
- [25] R. Shadmehr, H. J. Huang, and A. A. Ahmed, “A Representation of Effort in Decision-Making and Motor Control,” *Current Biology*, vol. 26, no. 14, pp. 1929–1934, 2016, doi: 10.1016/j.cub.2016.05.065.
- [26] E. M. Summerside, R. Shadmehr, and A. A. Ahmed, “Control of Movement: Vigor of reaching movements: reward discounts the cost of effort,” *J Neurophysiol*, vol. 119, no. 6, p. 2347, Jun. 2018, doi: 10.1152/JN.00872.2017.
- [27] R. M. N. Alexander, “A minimum energy cost hypothesis for human arm trajectories,” *Biol Cybern*, vol. 76, no. 2, pp. 97–105, 1997, doi: 10.1007/S004220050324.
- [28] R. J. van Beers, “Saccadic Eye Movements Minimize the Consequences of Motor Noise,” *PLoS One*, vol. 3, no. 4, p. e2070, Apr. 2008, doi: 10.1371/journal.pone.0002070.
- [29] T. Flash and N. Hogan, “The coordination of arm movements: an experimentally confirmed mathematical model,” *J Neurosci*, vol. 5, no. 7, pp. 1688–703, Jul. 1985, doi: 10.1523/JNEUROSCI.05-07-01688.1985.
- [30] S. Cao, Z. Luo, and C. Quan, “Online Inverse Optimal Control for Time-Varying Cost Weights,” *Biomimetics 2024, Vol. 9, Page 84*, vol. 9, no. 2, p. 84, Jan. 2024, doi: 10.3390/BIOMIMETICS9020084.
- [31] B. Berret, E. Chiovetto, F. Nori, and T. Pozzo, “Evidence for Composite Cost Functions in Arm Movement Planning: An Inverse Optimal Control Approach,”

- PLoS Comput Biol*, vol. 7, no. 10, p. e1002183, 2011, doi: 10.1371/JOURNAL.PCBI.1002183.
- [32] H. Ryu, “Optimality, Objectives, and Trade-Offs in Motor Control under Uncertainty,” 2023.
- [33] P. M. Fitts, “The information capacity of the human motor system in controlling the amplitude of movement,” *J Exp Psychol*, vol. 47, no. 6, pp. 381–391, Jun. 1954, doi: 10.1037/H0055392.
- [34] M. J. McGuffin and R. Balakrishnan, “Fitts’ law and expanding targets,” *ACM Transactions on Computer-Human Interaction (TOCHI)*, vol. 12, no. 4, pp. 388–422, Dec. 2005, doi: 10.1145/1121112.1121115.
- [35] K. Bortko, J. Jankowski, P. Bartków, P. Pazura, and B. Śmiałkowska, “Attracting User Attention to Visual Elements within Website with the use of Fitts’s Law and Flickering Effect,” *Procedia Comput Sci*, vol. 176, pp. 2756–2763, Jan. 2020, doi: 10.1016/j.procs.2020.09.281.
- [36] C. J. Lin and S. H. Ho, “Prediction of the use of mobile device interfaces in the progressive aging process with the model of Fitts’ law,” *J Biomed Inform*, vol. 107, p. 103457, Jul. 2020, doi: 10.1016/J.JBI.2020.103457.
- [37] E. N. Kamavuako, E. J. Scheme, and K. B. Englehart, “On the usability of intramuscular EMG for prosthetic control: a Fitts’ Law approach,” *J Electromyogr Kinesiol*, vol. 24, no. 5, pp. 770–777, 2014, doi: 10.1016/J.JELEKIN.2014.06.009.
- [38] S. M. Wurth and L. J. Hargrove, “A real-time comparison between direct control, sequential pattern recognition control and simultaneous pattern recognition control

- using a Fitts' law style assessment procedure.," *J Neuroeng Rehabil*, vol. 11, p. 91, May 2014, doi: 10.1186/1743-0003-11-91.
- [39] W. Li and E. Todorov, "Iterative Linear Quadratic Regulator Design for Nonlinear Biological Movement Systems," in *Proceedings of the 1st International Conference on Informatics in Control, Automation and Robotics*, 2004, pp. 222–229. doi: 10.5220/0001143902220229.
- [40] E. Todorov, "Stochastic Optimal Control and Estimation Methods Adapted to the Noise Characteristics of the Sensorimotor System," *Neural Comput*, vol. 17, pp. 1084–1108, 2005.
- [41] K. Deb, "Multi-objective optimization," *Search Methodologies: Introductory Tutorials in Optimization and Decision Support Techniques*, pp. 273–316, 2005, doi: 10.1007/0-387-28356-0\_10.
- [42] S. C. Cerda-Flores, A. A. Rojas-Punzo, and F. Nápoles-Rivera, "Applications of Multi-Objective Optimization to Industrial Processes: A Literature Review," *Processes 2022, Vol. 10, Page 133*, vol. 10, no. 1, p. 133, Jan. 2022, doi: 10.3390/PR10010133.
- [43] J. L. J. Pereira, G. A. Oliver, M. B. Francisco, S. S. Cunha, and G. F. Gomes, "A Review of Multi-objective Optimization: Methods and Algorithms in Mechanical Engineering Problems," *Archives of Computational Methods in Engineering*, vol. 29, no. 4, pp. 2285–2308, Jun. 2022, doi: 10.1007/S11831-021-09663-X/FIGURES/6.
- [44] P. Ngatchou, A. Zarei, and M. A. El-Sharkawi, "Pareto multi objective optimization," *Proceedings of the 13th International Conference on Intelligent*

- Systems Application to Power Systems, ISAP'05*, vol. 2005, pp. 84–91, 2005, doi: 10.1109/ISAP.2005.1599245.
- [45] W. Stadler, *Multicriteria Optimization in Engineering and in the Sciences*, vol. 37. Springer Science and Business Media, 1988.
- [46] C. A. C. Coello, G. B. Lamont, and D. A. Van Veldhuizen, *Evolutionary Algorithms for Solving Multi-Objective Problems*, Second. 2007. doi: 10.1007/978-0-387-36797-2.
- [47] J. Arora, *Introduction to Optimum Design*. Elsevier, 2012. doi: 10.1016/C2009-0-61700-1.
- [48] R. T. Marler and J. S. Arora, “Survey of multi-objective optimization methods for engineering,” *Structural and Multidisciplinary Optimization*, vol. 26, no. 6, pp. 369–395, Apr. 2004, doi: 10.1007/s00158-003-0368-6.
- [49] J. S. Arora, “Multi-objective Optimum Design Concepts and Methods,” in *Introduction to Optimum Design*, Elsevier, 2017, pp. 771–794. doi: 10.1016/B978-0-12-800806-5.00018-4.
- [50] K. Miettinen, *Nonlinear Multiobjective Optimization*, vol. 12. in International Series in Operations Research & Management Science, vol. 12. Boston, MA: Springer US, 1998. doi: 10.1007/978-1-4615-5563-6.
- [51] T. Marler, “A study of Multi-objective optimization methods for engineering application,” University of Iowa, 2005. doi: 10.1017/S0165115300023299.
- [52] S. Mick, D. Cattaert, F. Paquet, P. Y. Oudeyer, and A. De Rugy, “Performance and usability of various robotic arm control modes from human force signals,” *Front Neurorobot*, vol. 11, no. OCT, pp. 1–16, 2017, doi: 10.3389/fnbot.2017.00055.

- [53] S. Zhai and P. Milgram, "Human performance evaluation of manipulation schemes in virtual environments," *Proceedings of IEEE Virtual Reality Annual International Symposium*, pp. 155–161, 1993, doi: 10.1109/VRAIS.1993.380784.
- [54] S. Zhai, "User performance in relation to 3D input device design," *ACM SIGGRAPH Computer Graphics*, vol. 32, no. 4, pp. 50–54, Nov. 1998, doi: 10.1145/307710.307728.
- [55] S. Zhai and P. Milgram, "Quantifying coordination in multiple DOF movement and its application to evaluating 6 DOF input devices," *Proceedings of the SIGCHI conference on Human factors in computing systems - CHI '98*, pp. 320–327, 1998, doi: 10.1145/274644.274689.
- [56] Teather Robert J and I. S. Mackenzie, "Position Vs Velocity control for Tilt based interaction," in *Proceedings of Graphics Interface 2014*, 2014, pp. 51–58. [Online]. Available: <https://dl.acm.org/doi/abs/10.5555/2619648.2619658>
- [57] H. H. Koester and J. Mankowski, "Automatic adjustment of mouse settings to improve pointing performance," *Assistive Technology*, vol. 26, no. 3, pp. 119–128, 2014, doi: 10.1080/10400435.2013.862583.
- [58] A. Fougner, O. Stavadahl, P. J. Kyberd, Y. G. Losier, and P. A. Parker, "Control of upper limb prostheses: Terminology and proportional myoelectric control review," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 20, no. 5, pp. 663–677, Sep. 2012, doi: 10.1109/TNSRE.2012.2196711.
- [59] G. Casiez, N. Roussel, and D. Vogel, "1 Euro Filter: A simple Speed-based Low pass-Filter for Noisy input in Interactive systems," in *Conference on human factors*

- in computing systems*, New York, New York, USA: ACM Press, 2012, p. 2527. doi: 10.1524/9783486594973.1.
- [60] S. Meek, J. Wood, and S. Jacobsen, “Model-based, Multi-muscle EMG control of Upper-Extremity Prostheses,” in *Multi Muscle Systems: Biomechanics and Movement Organization*, Springer-Verlag, 1990. doi: 10.1007/978-1-4613-9030-5.
- [61] T. D. Sanger, “Bayesian Filtering of Myoelectric Signals,” *J Neurophysiol*, vol. 97, no. 2, pp. 1839–1845, 2006, doi: 10.1152/jn.00936.2006.
- [62] D. Hofmann, N. Jiang, I. Vujaklija, and D. Farina, “Bayesian Filtering of Surface EMG for Accurate Simultaneous and Proportional Prosthetic Control,” *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 24, no. 12, pp. 1333–1341, 2016, doi: 10.1109/TNSRE.2015.2501979.
- [63] M. Dyson, J. Barnes, and K. Nazarpour, “Abstract myoelectric control with EMG drive estimated using linear, kurtosis and Bayesian filtering,” *International IEEE/EMBS Conference on Neural Engineering, NER*, pp. 54–57, 2017, doi: 10.1109/NER.2017.8008290.
- [64] L. Gan, J. W. Grizzle, R. M. Eustice, and M. Ghaffari, “Energy-Based Legged Robots Terrain Traversability Modeling via Deep Inverse Reinforcement Learning,” *IEEE Robot Autom Lett*, vol. 7, no. 4, pp. 8807–8814, 2022, doi: 10.1109/LRA.2022.3188100.
- [65] W. Liu, S. Member, R. Wu, and J. Si, “Optimization Method Facilitated by Inverse Reinforcement Learning,” vol. 7, no. 4, pp. 10882–10889, 2022.

- [66] A. Alili *et al.*, “A Novel Framework to Facilitate User Preferred Tuning for a Robotic Knee Prosthesis,” *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 31, pp. 895–903, 2023, doi: 10.1109/TNSRE.2023.3236217.
- [67] Y. Guiard, H. B. Olafsdottir, and S. T. Perrault, “Fitts’ law as an explicit time/error trade-off,” *Conference on Human Factors in Computing Systems - Proceedings*, pp. 1619–1628, 2011, doi: 10.1145/1978942.1979179.
- [68] I. S. MacKenzie and P. Isokoski, “Fitts’ throughput and the speed-accuracy tradeoff,” *Conference on Human Factors in Computing Systems - Proceedings*, pp. 1633–1636, 2008, doi: 10.1145/1357054.1357308.
- [69] P. M. Fitts and B. K. Radford, “Information capacity of discrete motor responses under different cognitive sets,” *J Exp Psychol*, vol. 71, no. 4, pp. 475–482, 1966, doi: 10.1037/h0022970.
- [70] C. M. Harris and D. M. Wolpert, “Signal-dependent noise determines motor planning,” *Nature*, vol. 394, no. 6695, pp. 780–784, Aug. 1998, doi: 10.1038/29528.
- [71] A. J. Nagengast, D. A. Braun, and D. M. Wolpert, “Risk-sensitive optimal feedback control accounts for sensorimotor behavior under uncertainty,” *PLoS Comput Biol*, vol. 6, no. 7, p. 15, 2010, doi: 10.1371/journal.pcbi.1000857.
- [72] A. M. Haith, T. R. Reppert, and R. Shadmehr, “Evidence for Hyperbolic Temporal Discounting of Reward in Control of Movements,” *Journal of Neuroscience*, vol. 32, no. 34, pp. 11727–11736, Aug. 2012, doi: 10.1523/JNEUROSCI.0424-12.2012.
- [73] K. Jimura, J. Myerson, J. Hilgard, T. S. Braver, and L. Green, “Are people really more patient than other animals? Evidence from human discounting of real liquid

- rewards,” *Psychon Bull Rev*, vol. 16, no. 6, pp. 1071–1075, Dec. 2009, doi: 10.3758/PBR.16.6.1071.
- [74] R. F. Stengel, *Optimal Control and Estimation*, Second. Dover Publication Inc., New York, 1994, 1994.
- [75] R. T. Marler and J. S. Arora, “The weighted sum method for multi-objective optimization: New insights,” *Structural and Multidisciplinary Optimization*, vol. 41, no. 6, pp. 853–862, 2010, doi: 10.1007/s00158-009-0460-7.
- [76] T. R. Marler, “A Study of Multi-objective Optimization methods for engineering applications,” University of Iowa, 2005. [Online]. Available: [https://www.cambridge.org/core/product/identifier/S0165115300023299/type/journal\\_article](https://www.cambridge.org/core/product/identifier/S0165115300023299/type/journal_article)
- [77] K. Deb, *Multi-objective optimization using evolutionary algorithms (2001, John Wiley & Sons)*. 2008.
- [78] H. H. Sears and J. Shaperman, “Proportional myoelectric hand control: an evaluation,” *Am J Phys Med Rehabil*, vol. 70, no. 1, pp. 20–28, 1991, doi: 10.1097/00002060-199102000-00005.
- [79] T. Flash and N. Hogan, “The coordination of arm movements: an experimentally confirmed mathematical model.” *J Neurosci*, vol. 5, no. 7, pp. 1688–703, Jul. 1985, doi: 10.1523/JNEUROSCI.05-07-01688.1985.
- [80] J. L. Taylor and S. C. Gandevia, “A comparison of central aspects of fatigue in submaximal and maximal voluntary contractions,” *J Appl Physiol*, vol. 104, no. 2, pp. 542–550, Feb. 2008, doi:

10.1152/JAPPLPHYSIOL.01053.2007/ASSET/IMAGES/LARGE/ZDG00208769  
70003.JPEG.

- [81] R. A. Schmidt, H. Zelaznik, B. Hawkins, J. S. Frank, and J. T. Quinn, “Motor-output variability: a theory for the accuracy of rapid motor acts.,” *Psychol Rev*, vol. 47, no. 5, pp. 415–51, Sep. 1979.
- [82] K. E. Jones, A. F. Hamilton, and D. M. Wolpert, “Sources of signal-dependent noise during isometric force production.,” *J Neurophysiol*, vol. 88, no. 3, pp. 1533–44, Sep. 2002, doi: 10.1152/jn.2002.88.3.1533.
- [83] E. Todorov, “Stochastic Optimal Control and Estimation Methods Adapted to the Noise Characteristics of the Sensorimotor System,” *Neural Comput*, vol. 17, pp. 1084–1108, 2005.
- [84] C. M. Harris and D. M. Wolpert, “Signal-dependent noise determines motor planning,” *Nature*, vol. 394, no. 6695, pp. 780–4, Aug. 1998, doi: 10.1038/29528.
- [85] E. A. Clancy, S. Bouchard, and D. Rancourt, “Estimation and application of EMG amplitude during dynamic contractions,” *IEEE Engineering in Medicine and Biology Magazine*, vol. 20, no. 6, pp. 47–54, 2001, doi: 10.1109/51.982275.
- [86] E. A. Clancy, “Electromyogram amplitude estimation with adaptive smoothing window length,” *IEEE Trans Biomed Eng*, vol. 46, no. 6, pp. 717–729, 1999, doi: 10.1109/10.764948.
- [87] B. Hudgins, P. Parker, and R. N. Scott, “A New Strategy for Multifunction Myoelectric Control,” *IEEE Trans Biomed Eng*, vol. 40, no. 1, pp. 82–94, 1993, doi: 10.1109/10.204774.

- [88] S. Tam *et al.*, “Towards Robust and Interpretable EMG-based Hand Gesture Recognition using Deep Metric Meta Learning,” Apr. 2024, Accessed: Aug. 06, 2024. [Online]. Available: <https://arxiv.org/abs/2404.15360v1>
- [89] E. N. Kamavuako, E. J. Scheme, and K. B. Englehart, “On the usability of intramuscular EMG for prosthetic control: A Fitts’ Law approach,” *Journal of Electromyography and Kinesiology*, vol. 24, no. 5, pp. 770–777, 2014, doi: 10.1016/j.jelekin.2014.06.009.
- [90] R. N. Khushaba, A. H. Al-Timemy, A. Al-Ani, and A. Al-Jumaily, “A Framework of Temporal-Spatial Descriptors-Based Feature Extraction for Improved Myoelectric Pattern Recognition,” *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 25, no. 10, pp. 1821–1831, Oct. 2017, doi: 10.1109/TNSRE.2017.2687520.
- [91] J. Kennedy and R. Eberhart, “Particle swarm optimization,” in *Proceedings of ICNN’95 - International Conference on Neural Networks*, IEEE, 1995, pp. 1942–1948. doi: 10.1109/ICNN.1995.488968.
- [92] L. Faes *et al.*, “Small-sample characterization of stochastic approximation staircases in forced-choice adaptive threshold estimation.,” *Percept Psychophys*, vol. 69, no. 2, pp. 254–62, Feb. 2007, doi: 10.3758/bf03193747.
- [93] K. Levenberg, “A method for the solution of certain non-linear problems in least squares,” *Q Appl Math*, vol. 2, no. 2, pp. 164–168, 1944, doi: 10.1090/qam/10666.
- [94] R. J. van Beers, “Saccadic Eye Movements Minimize the Consequences of Motor Noise,” *PLoS One*, vol. 3, no. 4, p. e2070, Apr. 2008, doi: 10.1371/journal.pone.0002070.

- [95] T. D. Green and J. H. Flowers, “Structuring of early reaching movements: A longitudinal study,” *J Mot Behav*, vol. 23, no. 4, pp. 293–300, Dec. 1991, doi: 10.1080/00222895.1991.9942040.
- [96] K. R. Wilson, K. B. Englehart, and J. W. Sensinger, “The effects of control signal noise on simultaneous submovements,” *IEEE International Conference on Rehabilitation Robotics*, pp. 96–100, 2017, doi: 10.1109/ICORR.2017.8009228.
- [97] A. Fishbach, S. A. Roy, C. Bastianen, L. E. Miller, and J. C. Houk, “Deciding when and how to correct a movement: Discrete submovements as a decision making process,” *Exp Brain Res*, 2007, doi: 10.1007/s00221-006-0652-y.
- [98] L. Peternel, O. Sigaud, and J. Babič, “Unifying speed-accuracy trade-off and cost-benefit trade-off in human reaching movements,” *Front Hum Neurosci*, vol. 11, p. 315561, Dec. 2017, doi: 10.3389/FNHUM.2017.00615/BIBTEX.
- [99] E. Guigon, “A computational theory for the production of limb movements.,” *Psychol Rev*, vol. 130, no. 1, pp. 23–51, Jan. 2023, doi: 10.1037/rev0000323.
- [100] R. J. Van Beers, P. Haggard, and D. M. Wolpert, “The Role of Execution Noise in Movement Variability,” *J Neurophysiol*, vol. 91, no. 2, pp. 1050–1063, 2004, doi: 10.1152/jn.00652.2003.
- [101] J. Sensinger, K. Stewart, A. G. Arunachalam, and K. B. Englehart, “Systems and methods for inductive tuning of human-machine interface,” *US Patent US20210307940A1*, 2021, Accessed: Apr. 16, 2024. [Online]. Available: <https://patents.google.com/patent/US20210307940A1/en>

- [102] R. T. Marler and J. S. Arora, “The weighted sum method for multi-objective optimization: New insights,” *Structural and Multidisciplinary Optimization*, vol. 41, no. 6, pp. 853–862, 2010, doi: 10.1007/s00158-009-0460-7.
- [103] Kalyanmoy. Deb, *Multi-objective optimization using evolutionary algorithms*. John Wiley & Sons, Wiley-Interscience series in systems and optimization, 2001.
- [104] E. Scheme, B. Lock, L. Hargrove, W. Hill, U. Kuruganti, and K. Englehart, “Motion normalized proportional control for improved pattern recognition-based myoelectric control,” *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 22, no. 1, pp. 149–157, Jan. 2014, doi: 10.1109/TNSRE.2013.2247421.
- [105] M. S. Elmusrati, “Preliminaries of mathematics in business and information management,” *Handbook on Business Information Systems*, pp. 791–815, Jan. 2010, doi: 10.1142/9789812836069\_0033.
- [106] “Convex set - Wikipedia.” Accessed: Jun. 23, 2021. [Online]. Available: [https://en.wikipedia.org/wiki/Convex\\_set](https://en.wikipedia.org/wiki/Convex_set)
- [107] “Convex function - Wikipedia.” Accessed: Jun. 23, 2021. [Online]. Available: [https://en.wikipedia.org/wiki/Convex\\_function](https://en.wikipedia.org/wiki/Convex_function)
- [108] P. M. Pardalos, A. Žilinskas, and J. Žilinskas, “Definitions and examples,” *Springer Optimization and Its Applications*, vol. 123, pp. 3–12, 2017, doi: 10.1007/978-3-319-61007-8\_1.
- [109] K. Deb, K. Miettinen, and S. Chaudhuri, “Toward an Estimation of Nadir Objective Vector Using a Hybrid of Evolutionary and Local Search Approaches,” *IEEE Transactions on Evolutionary Computation*, vol. 14, no. 6, pp. 821–841, Dec. 2010, doi: 10.1109/TEVC.2010.2041667.

## Appendix A: Source Code

The relevant code and data can be found on the public repository on GitHub:

<https://github.com/anjana-arunachalam/ThesisSubmissionCode>

- main2.m is the main script that performs the optimization calculations.
- /Results/VD\_const/ - contains the optimization results for the proportional control virtual dynamic
- /Results/VD\_optimised/ – contains the optimization results for the nonlinear blended Virtual dynamic system that optimizes all four parameters of the VD for different  $\alpha$  settings

## Appendix B: Additional terms and definitions for MOO

Some helpful but not directly relevant definitions in the field of multi-objective optimization are listed below.

### 1) **Weak Pareto Optimality:**

The concept of weak Pareto optimality is closely related to the concept of Pareto optimality. A point  $x^*$  is weakly Pareto optimal if there is no other point in the feasible space that improves all the objective functions simultaneously; however, there may be points that improve some of the objectives while keeping others unchanged. Mathematically, for a MOO problem with  $k$  objectives  $f_1(x), f_2(x), \dots, f_k(x)$ , a solution  $x^*$  is weakly Pareto optimal if there exists no other point  $x$  such that:

$$f_i(x) < f_i(x^*) \forall i \in \{1, 2, \dots, k\}$$

Every Pareto optimal solution is also weakly Pareto optimal, but the converse is not true. That is a set of weakly Pareto optimal solutions contains the Pareto optimal set as a subset within it [105].

### 2) **Pareto set /Pareto optimal set:**

The non-dominated set from a local region corresponds to the local Pareto optimal set, whereas the nondominated set from the entire feasible search space corresponds to the global Pareto set. The term Pareto set can be used to refer to solutions in both the design and the objective space [49].

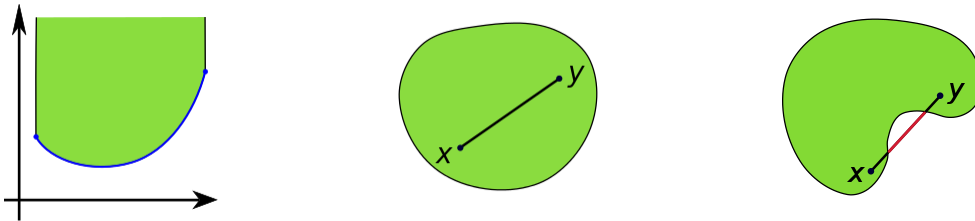
Based on the math example we have discussed, the set of points with  $x = [-1, 2]$  is part of the global Pareto optimal set.

### 3) Pareto optimal front:

refers to the plot, curve or hyperplane created by joining the points within the Pareto optimal set in the *objective* space. The Pareto front can also be defined as the set of non-dominated points in the entire feasible objective space. The Pareto front of the example from Eq 2.1 is shown in red in Figure 1.

### 4) Convexity of a function:

A real-valued function is convex if the line segment between any two points on the graph of the function lies above the graph between the two points. A single variable function is convex, if and only if its second derivative is non-negative in the entire domain. For a function with several variables, the Hessian matrix of the partial second-order derivatives needs to be positive semidefinite on the entire domain in order for the function to be convex.



**Figure 17: (From left to right) Images of a convex function, convex set, and non-convex set [106], [107].**

(i) **Convex set:** A given set is convex if the line segment joining any two points within the set is fully contained within the set.

(ii) **Convex MOO:** If the feasible region and all the objective functions of a multi-objective optimization problem are convex, the problem falls under the special category of convex MOO.

Let us consider the following mathematical multi-objective optimization example from Pardalos *et al.* [108].

CASE 1:

$$f_1(\underline{x}) = 5(x_1 - 0.1)^2 + (x_2 - 0.1)^2$$

$$f_2(\underline{x}) = (x_1 - 0.9)^2 + 5(x_2 - 0.9)^2$$

Subject to the constraints:  $-0.5 \leq x_1 \leq 0.5$  and  $-0.5 \leq x_2 \leq 0.5$

In this case, the feasible space forms a convex set as the constraints bound a unit square in the design space. Each objective is also convex as it is a linear sum of quadratic terms.

By contrast if the problem had been defined as:

CASE 2:

$$f_1(\underline{x}) = 5(x_1 - 0.1)^3 + (x_2 - 0.1)^5$$

$$f_2(\underline{x}) = (x_1 - 0.9)^2 + 5(x_2 - 0.9)^2$$

Subject to the constraints:  $-0.5 \leq x_1 \leq 0.5$  and  $-0.5 \leq x_2 \leq 0.5$

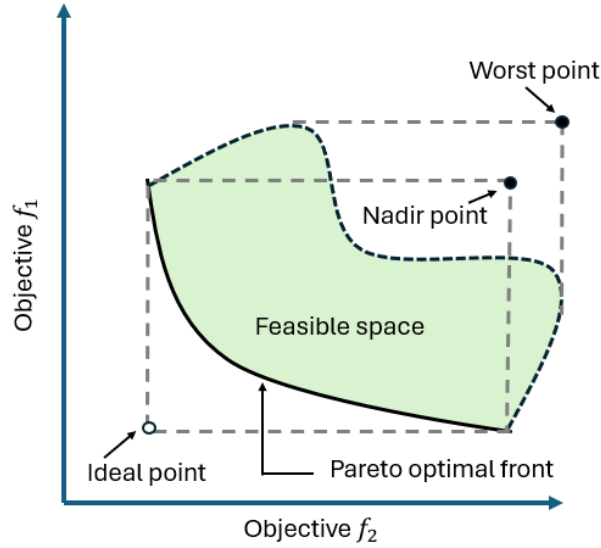
The feasible space continues to be a convex set. But the first objective is not a convex function. As not all objectives are convex, the overall problem is a non-convex MOO problem.

5) **Ideal point**/ Ideal objective vector:

The ideal objective vector (or ideal point) represents the lower bound of each objective in the entire feasible search space. If the objective functions are conflicting, then the ideal point will not be part of the Pareto solution. This is because in a MOO with conflicting objectives, improving the value of one objective will lead to worse performance for at least some of the other objectives.

6) **Nadir point**/ Nadir vector:

The nadir objective, on the other hand, represents the upper bound of each objective in the entire Pareto optimal set. Note that the nadir point is different from the worst objective vector point in the feasible space as shown in Figure 18.



**Figure 18: Depiction of the ideal point, the nadir point, and the worst point for a general two-objective multi-objective optimization on problem. This image is inspired by and modified from the work of Deb *et al.* [109].**

To normalize each objective in the entire range of the Pareto optimal region, the knowledge of the nadir and the ideal objective vectors can be used as:

$$\theta_i^{norm} = \frac{\theta_i - z_i^*}{z_i^{nad} - z_i^*} \quad (2.3)$$

Where  $\theta_i^{norm}$  refers to the normalized version of the  $i^{\text{th}}$  objective function,

$\theta_i$  refers to the actual value of the objective,

$z_i^*$  - refers to the magnitude of the  $i^{\text{th}}$  objective function value of the ideal point,

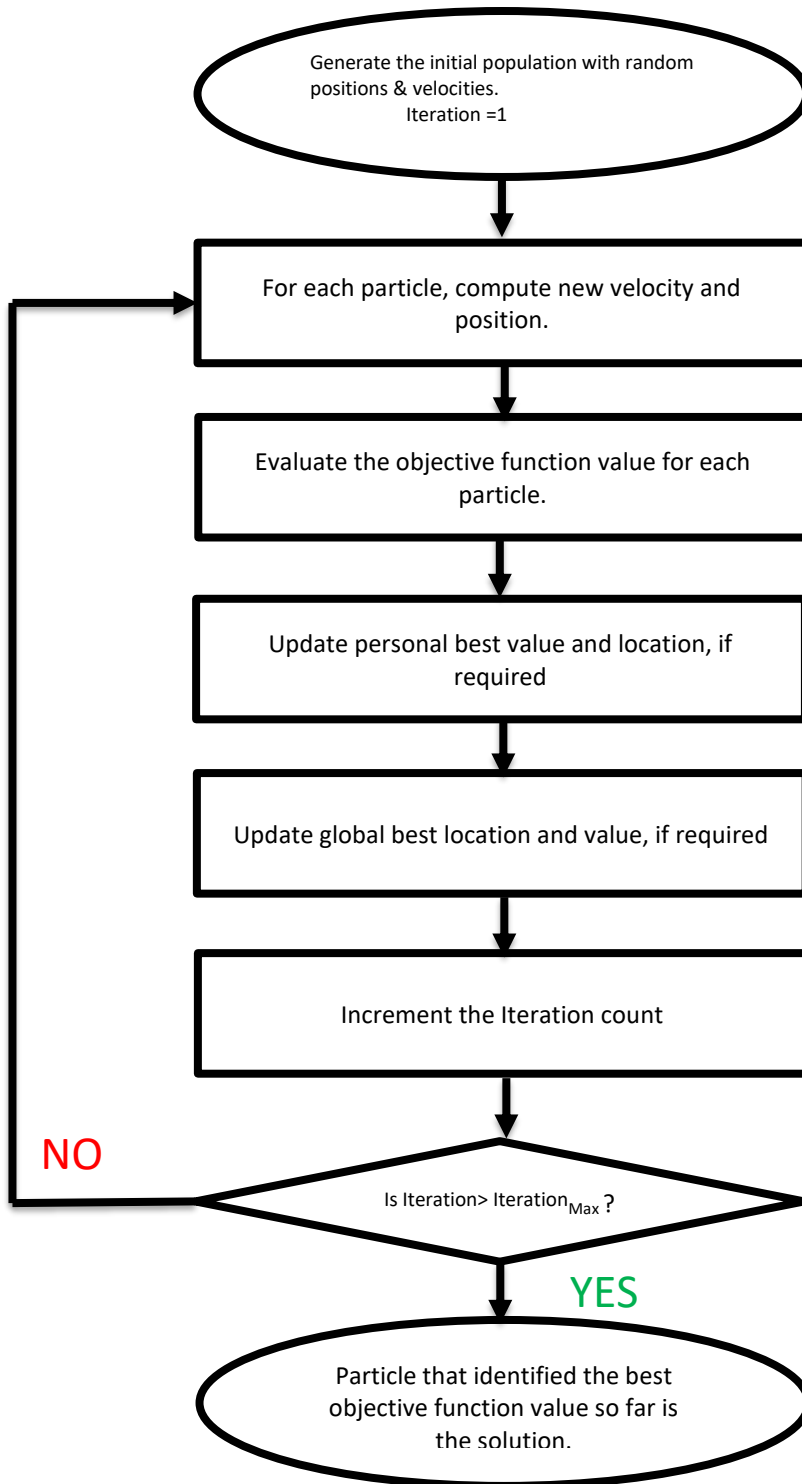
$z_i^{nad}$  – represents the magnitude of the  $i^{\text{th}}$  objective function at the nadir point.

The ideal and the nadir points for different MOO problems can be estimated by normalizing each objective independent of the others for the given set of constraints. The vector of the best objective values obtained from the independent optimization of each objective gives the ideal point. The nadir vector is constructed from the worst value of each objective from the Pareto front when one of the other objectives is optimized independently (Refer Figure 18).

## **Appendix C: Hierarchical PSO implementation details**

### **(1) Algorithm schematic**

Figure 19 shows how the PSO algorithm can be implemented using a block diagram format.



**Figure 19: Algorithmic flowchart of the Particle Swarm Optimization Algorithm.**

## **(2) Limits on the optimization variables**

Table 4 shows the selected range of the optimization variables used. The upper and lower limits of these variables were bounded to keep the HMI interaction problem practical while allowing sufficient flexibility and a large search space for tuning. The coefficients of virtual dynamics were given a symmetric search space about the origin. The exponent parameters of virtual dynamics could only take non-negative real numbers up to 4. The maximum time duration of each cubic spline was set to be 10 seconds. This gives a maximum movement time of 20 seconds, which was found to be more than sufficient to make the 10-unit and 15-unit target reaches that were modelled in the HMI interaction model. The limits on the amplitude of the control signal were set at +/- 35 to mimic the 35% maximum voluntary contraction levels about which myoelectric prosthesis users tend to operate.

**Table 4: Lower and upper limits of the optimization variables**

<b>Optimization variable</b>	<b>Lower Bound</b>	<b>Upper Bound</b>
<b>VD coefficient <math>a_1</math></b>	-5	+5
<b>VD coefficient <math>a_2</math></b>	-5	+5
<b>VD exponent <math>b_1</math></b>	0	4
<b>VD exponent <math>b_2</math></b>	0	4
<b>Individual cubic spline durations: <math>[T_1, T_2]</math></b>	[0.001, 0.001] seconds	[10, 10] seconds
<b>Individual cubic spline initial slopes: <math>[\varphi_1^i, \varphi_2^i]</math></b>	[-200, -200]	[+200, +200]
<b>Individual cubic spline final slopes: <math>[\varphi_1^f, \varphi_2^f]</math></b>	[-200, -200]	[+200, +200]
<b>Individual cubic spline initial amplitude: <math>[M_1^i, M_2^i]</math></b>	[-35, -35]	[+35, +35]
<b>Individual cubic spline final amplitude: <math>[M_1^f, M_2^f]</math></b>	[-35, -35]	[+35, +35]

## Appendix D: Result data and additional plots

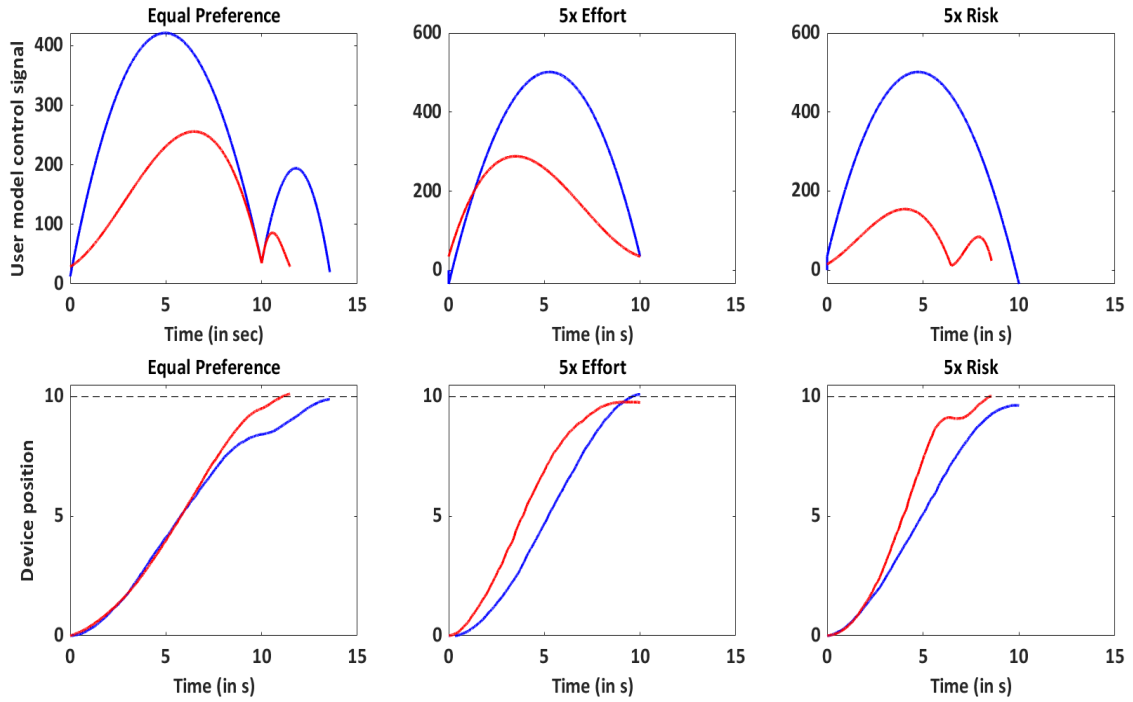
Table 5 lists the individual costs of effort, time, and reliability, the optimized virtual dynamic parameters and the total movement cost corresponding to each optimized target reach for the four user preference scenarios simulated. The total cost listed in this table is the sum of the costs of effort, time, reliability and the constraint values.

**Table 5: Optimized costs and virtual dynamics for different relative weights**

Optimized for	Weights $[\alpha_1, \alpha_2, \alpha_3]$	Optimized Cost*				Best VD parameter s
		$J_u$	$J_t$	$J_r$	$J_{Total}$	
Equal	[1 1 1]	$6.68 \times 10^3$	6.15	$1.01 \times 10^{-3}$	$2.04 \times 10^4$	[-0.001 0.031 1.95 0.68]
		$1.35 \times 10^4$	7.08	$2.52 \times 10^{-3}$	$6.93 \times 10^4$	
Effort	[5 1 1]	$4.17 \times 10^3$	5.02	$1.38 \times 10^{-2}$	$2.86 \times 10^5$	[0.002 0.002 1.84 1.22]
		$5.35 \times 10^3$	6.43	$2.20 \times 10^{-2}$	$2.08 \times 10^5$	
Time	[1 5 1]	$3.57 \times 10^2$	4.50	$2.30 \times 10^{-2}$	$6.68 \times 10^5$	[-0.02 0.001 1.831 1.81]
		$5.96 \times 10^2$	3.98	$8.09 \times 10^{-2}$	$6.68 \times 10^5$	
Risk	[1 1 5]	$2.42 \times 10^3$	5.34	$9.29 \times 10^{-3}$	$1.51 \times 10^6$	[-0.025 0.004 1.42 1.27]
		$5.43 \times 10^3$	4.73	$2.92 \times 10^{-2}$	$2.08 \times 10^6$	

\*The two rows represent the costs involved in moving to the two targets - 10 and 15 units from the origin, respectively.

Figure 20 provides a direct comparison of the control signal and device trajectories when using a default constant virtual dynamic system or a tunable blended nonlinear virtual dynamic system for a target that is 10 units from the origin.



**Figure 20: Comparing the control signals and device trajectories when using a device with non-tunable proportional velocity control mapping (in blue) and a tunable device dynamic mapping (in red) to reach the first target, which is 10 units from the origin.**

## Curriculum Vitae

**Candidate's full name:** Anjana Gayathri Arunachalam

**Universities attended:** National Institute of Technology, Tiruchirappalli (2017)

**Degree Obtained:** Bachelor of Technology (Honours) in  
Instrumentation and Control Engineering

### Publications:

A.G. Arunachalam, K. B. Englehart and J. W. Sensinger, "Optimized control mapping through user-tuned cost of effort, time, and reliability\*," *2019 IEEE 16th International Conference on Rehabilitation Robotics (ICORR)*, 2019, pp. 837-842, doi: 10.1109/ICORR.2019.8779397.

A. G. Arunachalam, K. Englehart, and J. Sensinger, "Intuitive optimal prosthesis tuning through user-tuned costs of effort and accuracy", *MEC20 Symposium*, Jul. 2020

### Conference Presentations:

(Podium Presentation) *Tuning device dynamics of Human-Machine Interaction systems*, 2019 Atlantic Association for Research in the Mathematical Sciences, Industrial Problem Solving Workshop

(Poster Presentation) *Optimized control mapping through user-tuned cost of effort, time, and reliability*, 2019 IEEE International Conference on Rehabilitation Robotics

(Podium Presentation) *Intuitive prosthesis tuning through user-tuned preference of effort, time, reliability, and accuracy*, 2020 Atlantic Mobility Action Project

(Poster Presentation) *Optimal Tuning of Upper Limb Myoelectric Prosthesis through User preference of costs*, 2021 Annual Health Research Conference New Brunswick.