

*CONTEXT- AND PHYSIOLOGY-AWARE
MACHINE LEARNING FOR
UPPER-LIMB MYOCONTROL*



**Deutsches Zentrum
für Luft- und Raumfahrt**
German Aerospace Center

UNIVERSITÄTSMEDIZIN
GÖTTINGEN **UMG**

Gauravkumar K. Patel

Georg-August University Göttingen

**Applied Surgical and Rehabilitation Technology Lab (ART-Lab),
Department of Trauma Surgery, Orthopedics and Plastic Surgery,
University Medical Center Göttingen, Germany**

**This dissertation is submitted for the degree of Doctor rerum naturalium (Dr.rer.nat.)
within the doctoral Program in Computer Science (PCS)
of the Georg-August University School of Science (GAUSS)**

May 2018

THESIS SUPERVISORS

Supervisor: Prof. Dr. Dr. Dario Farina

Department of Bioengineering, Imperial College London, London, UK

Previous: Institute for Neurorehabilitation Systems, University Medical Centre
Göttingen, Germany

Second Supervisor: Dr. Strahinja Dosen

The Faculty of Medicine, Department of Health Science and Technology, Centre for
Sensory-Motor Interaction, Aalborg, Denmark

Applied Surgical and Rehabilitation Technology Lab (ART-Lab), Department of
Trauma Surgery, Orthopedics and Plastic Surgery, University Medical Centre
Göttingen, Germany

Third Supervisor: Dr. Claudio Castellini

Institute of Robotics and Mechatronics, DLR - German Aerospace Centre,
Wessling, Germany

Members of Examination Committee,

Prof. Dario Farina (Referee 1)

Jun.-Prof. Marcus Baum (Referee 2)

Prof. Florentin Wörgötter

Prof. Xiaoming Fu

Prof. Arndt Schilling

Prof. David Liebetanz

Date of Oral Examination: **3rd May 2018**

Dedication

This work is dedicated to Bhagwan Shree Swaminarayan and my Gurus, his holiness Shree Gunatitanad Swami, his holiness Shree Gopalanand Swami, Pujya Shastraji Maharaj Shree Dharmajivandasji Swami, Pujya Jogi Swami Shree Hariprakashdasji, Pujya Purani Swami Shree Premprakashdasji and Pujya Hariswarupdasji Swami.



Additionally, I would like to dedicate this work to my family, my father, my mother, my younger brother and my wife.

DECLARATION

This dissertation is the result of my own work and does not include the outcome of work done in collaboration, except where specifically indicated in the text. It has not been previously submitted, in part or whole, to any university or institution for any degree, diploma, or other qualification.

Sign:

Date:

Gauravkumar K. Patel

Göttingen

ABSTRACT

The world around us is shaped in such a way that our hands are necessary to accomplish most activities of daily living. It is therefore undeniable that the loss of the upper limb, partial or total, represents a severe impairment. With current advancements in robotic technology, it is now possible to replace a missing limb with a dexterous upper-limb prosthesis. However, the development of a reliable human machine interface (HMI), connecting the user and the prosthesis, is still an open challenge. Essentially, the HMI defines an invariant mapping scheme to transform electromyogram (EMG) signals generated by the user into movements on the prosthetic device, thereby allowing the user to control available functions by generating appropriate (predefined) EMG signals. An HMI control driven by EMG signals is known as myoelectric control or myocontrol. EMG signals associated with a particular motor task are distinct and repeatable and therefore, it is possible to use one of the many well-known machine learning (ML) algorithms as HMI for estimating different user motor intentions. With ML-based HMIs, users can directly activate a desired prosthesis function by producing EMG signals associated to that function during supervised learning. Although conceptually promising, ML-based control has shown a limited clinical viability, mainly due to the lack of reliability and robustness during real-time use. The aim of this thesis was to improve the reliability and robustness of ML-based control by developing context- and physiology- aware ML methods for upper-limb myocontrol. Today, most ML methods used for myoelectric control follow the conventional pattern recognition paradigm, where training data is collected using a supervised procedure and a mathematical function is fitted over the collected data to define an invariant mapping scheme between the user's EMG and available prosthesis movements. This conventional approach has two limitations. First, the mapping scheme (between the EMG and available movements) remains static (invariant) during use and does not consider the dynamics associated with real-life use of prosthesis. Second, the mathematical function fitted over the training data is assumed to implicitly capture the physiological principles behind generation of EMG; this assumption might not be true, as many commonly applied ML methods do not model the underlying physiology. The first limitation can be solved by developing ML methods which can consider context information describing the state of the system and/or environment during prosthesis use. This context information can be acquired either directly from the user or by placing additional sensors (e.g. inertial units) on the prosthesis. The former idea of deriving context information from the user is quite interesting, as it gives to the ML an opportunity to improve control by considering user's

requirement(s) during use. This thesis proposes one ML method (called Modular Regression, see Chapter 2) which exploits user-generated context information to improve control for different activities of daily living (ADL). Specifically, the proposed ML method organizes each prosthesis function as a module, which the users can insert/remove as required to best accomplish a given ADL. Next, if additional sensors were placed on the prosthesis to automatically derive context information, the ML controller would get an opportunity to (automatically) monitor the state of the prosthesis and react accordingly to maximize reliability and robustness. This thesis proposes one ML approach (called context-driven control, see Chapter 3) which utilizes context information from additional sensors to model different prosthesis states and then, the parameters of ML control were adapted to mitigate expected disturbances in each prosthesis state. Thus, with both new ML methods, the mapping scheme (between the user's EMG and available movements) does not remain static, but becomes reactive to the context information coming from the user or additional sensors. Experiments involving functional tasks were conducted to compare the newly developed context-aware ML methods with the conventional ML-based control. The experimental results indicate that the context-aware methods significantly outperform conventional ML control. The second limitation of conventional ML approaches, i.e. the fitted mathematical function may or may not capture the latent physiology information, can be solved by designing ML methods that are aware of the underlying muscle physiology. This thesis presents one ML algorithm (based on the cosine similarity metric, see Chapter 4) which exploits the principle of muscle coordination to classify EMG for online myoelectric control. Specifically, the principle of muscle coordination states that force production for a given movement relies on the coordination of different muscles and the EMG amplitude of involved muscles scales uniformly with the amount of force exerted. And therefore, the presented physiology-aware ML method was designed based on the assumption that amplitude-related EMG features for each movement are distributed along the line joining the origin of the feature space and the average maximum voluntary contraction of the movement. This assumption led to a simple training procedure and a computationally efficient solution. The presented physiology-aware ML method was extensively compared with the state-of-the-art ML method using four functional tasks. The results indicated that the new method performs significantly better than the standard ML method, while utilizing less training data and smaller computational effort. Overall, this thesis points to the potential advantage(s) of ML methods that exploit context and physiology information for online myocontrol over standard ML methods (with a static mapping scheme and no modelling of physiology), which largely prevail in the literature. Moreover, all ML methods

presented in this thesis are simple, robust and computationally efficient, and therefore, they can be directly used for interfacing most prosthetic devices available in the market, with a minor hardware upgrade.

ACKNOWLEDGEMENTS

Hereby I would like to acknowledge the financial support by the German Ministry for Education and Research (BMBF) under the project INOPRO (16SV7657). Additionally, I would like to acknowledge the timely guidance from all my supervisors, namely, Prof. Dario Farina, Dr. Strahinja Dosen and Dr. Claudio Castellini. Moreover, I would also like to thank Dr. Janne Hahne for his support and advices during this work and also, other colleagues of mine, especially, Dr Marko Markovic, Dr. Ivan Vujaklija, Mr. Markus Nowak, Mr. Hemant Karna, Mr. Shahed Taheri for their support. And lastly, students who did their course work with me, Mr. Mahmoud Al Zaitoun, Mr. Pascal Hecker, Mr. Martin Heinemann and Ms. Souzana Georgiadou.

The graphs and figures presented in the work were created using Inkscape (www.inkscape.org), Matplotlib with Python 3.3 on the Anaconda framework (matplotlib.org, python.org and continuum.io), Blender (blender.org) and Makehuman (www.makehuman.org). And, also thank you to Kayla Friedman and Malcolm Morgan of Cambridge, UK for producing the Word template used to produce this document.

CONTENTS

1 INTRODUCTION.....	1
1.1 PROSTHETIC HANDS: AN OVERVIEW ON DESIGN AND USAGE	2
1.2 STATE-OF-THE-ART INTERFACES FOR SELF-POWERED PROSTHESIS.....	5
1.2.1 <i>Methods to improve the robustness of ML-based myoelectric control...</i>	10
1.2.2 <i>Methods used to validate myoelectric control</i>	13
1.3 THE MAIN AIMS OF THIS THESIS	15
2 USING TASK-SPECIFIC CONTEXT TO CUSTOMIZE MYOCONTROL	19
2.1 MOTIVATION.....	19
2.2 MODULAR REGRESSION TO ADAPT TASK-SPECIFIC CONTEXT	21
2.3 MATERIALS AND METHODS	24
2.4 EXPERIMENTAL RESULTS	29
2.5 DISCUSSION	30
3 CONTEXT-DRIVEN MACHINE LEARNING FOR MYOCONTROL	33
3.1 MOTIVATION AND THE NOVEL CONCEPT.....	33
3.2 CONTEXT-DRIVEN CONTROL.....	35
3.3 MATERIALS AND METHODS	39
3.4 EXPERIMENTAL RESULTS	46
3.5 DISCUSSION	50
4 A PHYSIOLOGY-INSPIRED METRIC FOR CLASSIFYING EMG.....	55
4.1 MOTIVATION.....	55
4.2 CLASSIFICATION METHOD INSPIRED BY MUSCLE COORDINATION	56
4.2.1 <i>Comparing CoS with a data-driven approach.....</i>	57
4.2.2 <i>Cosine similarity: a metric for assessing the quality of EMG patterns..</i>	59
4.3 MATERIALS AND METHODS	61
4.4 EXPERIMENTAL RESULTS	65
4.5 DISCUSSION	68
4.5.1 <i>Integrating Context and Physiology Information.....</i>	70
5 CONCLUSION	73
6 REFERENCES.....	77
7 APPENDICES.....	85

LIST OF TABLES

Table 2.1: Different actions which can be performed using the i-Limb hand and the corresponding motor-commands (y_{action}). For the vector y_{action} , assume that the first component of the vector corresponds to thumb flexion, second component corresponds to thumb adduction, third component corresponds to the motor controlling the index finger flexion and so on for the remaining three fingers. 22

LIST OF FIGURES

Figure 1.1: Examples of active self-powered prostheses. (A) The i-Limb hand from Touch Bionics U.K. (www.touchbionics.com). (B) The Michelangelo Hand from Ottobock, Germany (www.ottobock.com). The images also show the portable batteries which power the prostheses during daily use. 3

Figure 1.2: Typical scheme used to implement machine learning based myoelectric control for dexterous upper-limb prostheses. 6

Figure 1.3: Three approaches for multimodal sensor fusion in upper-limb prostheses: (A) single-stage multimodal control system, (B) multi-stage multimodal control system and (C) semi-autonomous control system. 12

Figure 1.4: Examples of functional tasks used to validate myoelectric control. (A) The Box-and-Blocks test. (B) The clothespin test. 15

Figure 2.1: Visualizing the pattern overlap between six different sEMG patterns in a 2D space of projected RMS features. The presented data was collected from one of the participants during the experiment (see Section 2.3 for more details). Principle Component Analysis (PCA) was used to reduce the dimensions of RMS features extracted from eight sEMG channels and the explained variance was 72%. 20

Figure 2.2: The experimental setup comprising of the i-Limb Revolution prosthetic hand mounted on a splint and the Myo Armband with eight sEMG sensors. 24

Figure 2.3: The experimental protocol used to compare the two control methods, namely, Full Regression (FR) and Modular Regression (MR). The sequence in which both methods were administered across participants was randomized, and the order in which the five tasks (Ironing, Library, Arranging, Shopping and Playing Jenga) were performed was also randomized (indicated by an asterisk). 27

Figure 2.4: An experimental timeline. The participant performed two rounds of five tasks with FR, followed by two rounds with MR. Symbols: '+' - Pattern Update for the given pattern, T1 - Shopping, T2 - Ironing, T3 - Library, T4 - Jenga and T5 - Arranging. 28

Figure 2.5: Summary of the comparison between FR and MR, showing the median Task Completion Time (TCT) in minutes during Round 1 for five different ADLs. ('***' indicates $p < 0.001$, the horizontal line within a box indicates median, the box indicates IQR, the triangle indicates mean and the circles indicate outliers). 29

Figure 3.1: Conceptual scheme of the proposed context-driven ML-based myoelectric control (cxMLC). The scheme integrates a conventional ML-based myoelectric controller (MLC) and an automatically driven context-aware component (CAC). The context inference is realized by the CAC using proprioceptive (e.g. orientation, gyroscope) and exteroceptive (e.g. force, aperture) sensors embedded within the prosthesis. And, the parameters (e.g. activation thresholds) of MLC are adjusted in real-time based on the inferred context information. 34

Figure 3.2: State machine for context aware component (CAC). The state machine detected five prosthesis states: Free, Closing, Grasping, Holding and Moving. The conditions for transition between the states, indicated next to the arrows, were based on comparing the real-time sensor data (from embedded gyroscope, force and aperture sensors) to predefined thresholds. The annotations are: A – normalized grip aperture, F – normalized grasping force, (α, β, γ) – elbow orientation and $|\omega|$ – elbow angular velocity. The threshold values were: $A_T = 0.2$, $F_T = 0.02$, $\theta_T = 10^\circ$ and $\omega_T = 1\text{rad/s}$36

Figure 3.3: State based rules for context aware component (CAC). In each of the five prosthesis state, a predefined set of rules was used to post-process the outputs of the MLC. See the text for a detailed description of these rules. The annotations are: y_i – the predicted movement, T_i – activation threshold, R – Rotation, L – Lateral, P – Palmar and O – Hand Opening.37

Figure 3.4: An example of how different CAC state transitions were triggered during online control with cxMLC. The annotations are: A – normalized grip aperture, F – normalized grasping force, (α, β, γ) – elbow orientation and $|\omega|$ – net elbow angular velocity. And, the thresholds for state-transitions were: $A_T = 0.2$, $F_T = 0.02$, $\theta_T = 10^\circ$ and $\omega_T = 1\text{rad/s}$38

Figure 3.5: Experimental setup. It comprised a Michelangelo prosthesis with an embedded force (I) and aperture sensor (II), a Myo armband (III) with eight sEMG channels and an embedded IMU sensor, and an additional IMU (IV) to measure the upper-arm orientation. Importantly, the IMU sensor placed on the upper-arm (IV) was not required for the online control of prosthesis, but it was only employed for monitoring user movement.40

Figure 3.6: The experimental protocol used to compare conventional (MLC) and context driven (cxMLC) ML-based myoelectric control. Three tasks were administered to the participants sequentially (AWP, WT and HS), and each task was performed using both control methods, but the order in which the two control methods were administered was randomized across participants (indicated by an asterisk).....41

Figure 3.7: Arm Positions (AP). These arm positions were used during the experimental tasks. (A) AP1: forearm down, (B) AP2: elbow flexed, (C) AP3: arm frontally extended, (D) AP4: arm laterally extended.....42

Figure 3.8: Visual Task Instructor (VTI). (A) VTI for AWP task: the participants were required to match the metronomes indicating the current and the target wrist orientations. (B) VTI for WT task: the participants had to track the moving target by using the metronome indicating the current wrist orientation. (C) VTI for HS task: the participants had to shake their forearm in the indicated direction (here, left-and-right) until the pie-chart traversed the full circle.44

Figure 3.9: Summary of the results for the two primary outcome measures obtained with MLC and cxMLC: (A) Number of successful objects transferred per round in three tasks (AWP, WT, HS). (B) Number of objects dropped (while making a transfer) per Round in three tasks. (*, $p < 0.05$; ***, $p < 0.001$; the horizontal line within a box indicates median, the box indicates IQR and the circles indicate outliers).....47

Figure 3.10: Representative signals recorded during the online experiment, demonstrating the possible advantage of cxMLC in relation to MLC. (A) Spurious Opening Commands observed during control with MLC and (B) Spurious Closing Commands observed during control with MLC.....47

Figure 3.11: Representative signals recorded during the online experiment, demonstrating the advantage of cxMLC in terms of how (A) Variations in Force were restored during control with cxMLC, and how (B) an unwanted wrist rotation could have been prevented in the HS task by increasing the corresponding activation

threshold as in the Moving state. (the asterisks indicate two spurious commands which would have crossed the state dependent threshold for the Moving state in cxMLC control.)	48
Figure 3.12: Summary of the results for the secondary outcome measures obtained from the offline analysis. (A) Number of spurious opening commands observed during successful trials in each round with MLC and cxMLC. (B) Number of spurious closing commands observed during successful trials in each round with MLC and cxMLC. (*, $p < 0.05$; **, $p < 0.01$; ***, $p < 0.001$; the horizontal line within a box indicates median, the box indicates IQR and the circles indicate outliers).....	49
Figure 3.13: Summary of the results for the secondary outcome measures obtained from the offline analysis. (A) The mean change in normalized grasping force from beginning till the end of trial for each successful trial in a given round with MLC and cxMLC. (B) The mean change in wrist orientation from beginning till the end of trail for each successful trial in a given round with MLC and cxMLC (HS task only). (*, $p < 0.05$; **, $p < 0.01$; the horizontal line within a box indicates median, the box indicates IQR and the circles indicate outliers).....	49
Figure 3.14: Summary of the results for the questionnaire. (A) Task Physical Demand reported for the three tasks AWP, WT and HS. (B) Control Cognitive Demand reported in terms of Mental Demand and Frustration for control with MLC and cxMLC. (C) Control Comparison between MLC and cxMLC. (*, $p < 0.05$; the horizontal line within a box indicates median, the box indicates IQR and the circles indicate outliers)	50
Figure 4.1: (A) 2D Projection of amplitude-related EMG features generated by an able-bodied participant performing four different movements, namely, wrist- flexion and extension, ulnar- and radial- deviation plus rest. (B) The data for the rest class modelled as a Gaussian distribution. (C) and (D) Visualization of classification boundaries obtained after applying the LDA and CoS model on the transformed data, respectively. Principal Component Analysis (PCA) was used for dimensionality reduction and the explained variance was 81%. The plotted data has been collected according to the procedure explained in Section 4.3.	58
Figure 4.2: Cosine similarity versus Mahanobis distance based modelling of EMG data. A Mahanobis distance models data using a Gaussian distribution (represented by an ellipse), whereas the cosine similarity models each class using a line connecting the origin of the feature space to the MVC of the class. The models have been generated using the data shown in Figure 4.1A.	60
Figure 4.3: Confidence evaluated by applying Bayesian statistics on the transformed training data shown in Figure 4.1A. The value of estimated confidence depends on the distribution density of the training points, i.e. the estimated confidence is high in regions with high density of training points, and it decreases as the density of training point decreases. The presented Bayesian confidence was calculated after applying the Random Fourier Feature kernel [67] on the transformed training data.....	61
Figure 4.4: Experimental setup for (A) able-bodied participants and (B) the amputee participant.....	62
Figure 4.5: The experimental protocol used to compare LDA and CoS classifier. The experiment was divided into two blocks, where either LDA or CoS was administered randomly across participants. In each block, four rounds (4R) of four functional tasks were performed. Abbreviations are, T1: Box and Blocks, T2: Clothespin, T3: Bottle Transfer, T4: Bottle Turn and 4R: four rounds.	65
Figure 4.6: Summary of the results for the able-bodied participants. (A) The Box-and-Blocks test, (B) Clothespin test, (C) the Bottle Transfer task and (D) the Bottle Turn tasks. CoS performed better than LDA in three out of four tasks. (* indicates $p < 0.05$, the horizontal line within a box indicates median, the box indicates IQR, and the circles indicate outliers).	66

Figure 4.7: Summary of the results for the amputee participant. (A) The Box-and-Blocks test, (B) Clothespin test, (C) the Bottle Transfer task and (D) the Bottle Turn tasks. CoS performed consistently better than LDA in all four tasks. (the horizontal line within a box indicates median, the box indicates IQR, and the circles indicate outliers).....66

Figure 4.8: Number of bottles dropped per round by able-bodied participants during (A) the Bottle Transfer and (B) the Bottle Turn tasks. (* indicates $p < 0.05$, the horizontal line within a box indicates median, the box indicates IQR, and the circles indicate outliers).....67

Figure 4.9: Prosthesis command (for hand opening and pronation) generated by the amputee during the Box-and-Blocks test. The hand opening commands were occasionally misclassified as pronation by both LDA and CoS. However, these misclassifications were more frequent in the control with LDA than with CoS. (The Box-and-Blocks test required only hand opening and closing for task completion and so, it is assume that the amputee did not intentionally activate pronation during the test.).....67

Figure 4.10: Post-processing rules based on cosine similarity (CS) for the context aware component (CAC) presented in Chapter 3 (Figure 3.2 and Figure 3.3). In each of the five states, the post-processing rules based-on activation thresholds (T) have been changed to IF-statements based on cosine similarity for evaluating the quality of EMG patterns responsible for the detected movement. The annotations are: y_i – the estimated strength for the classified movement (and $y_i = 0$ for all other movement), CS_i – cosine similarity of the input with respect to the prototype vector of the detected movement, Δ_1, Δ_2 – rejection thresholds ($\Delta_2 > \Delta_1$), R – Rotation, L – Lateral, P – Palmar, O – Hand Opening, F – grip force. Lastly, it is to be noted that, a fixed activation of $T = 0.2$ is always applied to remove uncertainties at low contraction intensities.....71

LIST OF ABBREVIATIONS AND ACRONYMS

ACU	Automatic Control Unit
ADL	Activities of Daily Living
AWP	Arm and Wrist Positioning Task
CAC	Context Aware Component
CoS	Cosine Similarity (based classifier)
cxMLC	context-driven Machine Learning Control/Controller
DoF	Degree-of-Freedom
EMG	Electromyography/Electromyograms
FMG	Force Myography
FR	Full Regression
FSM	Finite State Machine
GPS	Global Positioning System
HMI	Human Machine Interface
HS	Hand Shaking Task
IMU	Inertial Measurement Units
kNN	k-Nearest Neighbors
LDA	Linear Discriminant Analysis
MAP	Muscle Activation Pattern
MAV	Mean Absolute Value(s)
ML	Machine Learning
MLC	(conventional) Machine Learning Control/Controller
MR	Modular Regression
ms	milliseconds
MVC	Maximum Voluntary Contraction
PCA	Principle Component Analysis

PU	Pattern Update
RFF	Random Fourier Features
RMS	Root Mean Square
s	seconds
sEMG	surface Electromyography/Electromyograms
SHAP	Southampton Hand Assessment Protocol
SSC	Slope Sign Change(s)
SVM	Support Vector Machines
TCT	Task Completion Time
UBET	Unilateral Below Elbow Test
VTI	Visual Task Instructor
WFL	Waveform Length
WT	Wrist Tracking Task
ZC	Zero Crossing(s)

LIST OF APPENDICES

<i>Appendix 1: Questionnaire given after the Experiment in Chapter 3.....</i>	<i>87</i>
<i>Appendix 2: Dissemination.....</i>	<i>89</i>

1 INTRODUCTION

The human-hand is a truly outstanding dexterous tool, which is essential for interaction and perception of the world around us. We use our hands to accomplish most activities of daily living (ADL) and therefore, it is not surprising that upper limb injuries are among the most common ones [1], [2]. Most upper limb injuries occur at home, during work, or while performing sports [1], [3], [4]. The loss of upper limb, total or partial, can lead to severe impairments. Worldwide there are millions of people whose lives are affected by the loss of upper limb. In the United States alone, there are approximately more than one million people living with an amputation [5] and, approximately more than ten thousand upper limb amputations are observed annually. A vast majority of them occur either due to trauma or tumor [6].

The loss of upper limb can affect the physical, mental and social state of the patient. To cope up with ADL, patients tend to develop compensatory strategies that might have a negative impact on their musculoskeletal system. A prolonged disability can also result in difficulties reintegrating into the society, thereby affecting the patients' psychological health. For this reason, it is considered best to replace the missing limb with an artificial assistive device. Today, it is possible to replace a missing limb with a dexterous prosthesis, but an efficient and user-friendly human-machine interface (HMI) is still missing. This chapter introduces the current state-of-the-art technologies in upper-limb hand prosthesis, by providing an overview on the design, usage and limitations of available devices. The chapter concludes with an outline of the main aims of this thesis.

1.1 Prosthetic hands: an overview on design and usage

The need for prosthetic devices has been recognized centuries ago, and their earliest mention dates back to a number of ancient civilizations [7]. These prosthetic devices were passive and had a very limited functionality (due to the lack of an active component). The first passive prostheses with some active components appeared in the middle ages, but they were mostly used as a part of the bodily-armor and had springs which could be adjusted by the contralateral hand. A development towards a more active/functional solution was necessary. The first active “*body-powered prostheses*” appeared in the early 20th century, followed by the development of active “*self-powered prostheses*” in the last half of the 20th century. Based on the historical evolution of prosthetic devices, they can be divided into three categories:

- i. **Passive Prosthesis:** Due to the absence of an active component, passive devices are mostly used in bimanual tasks that require fixation or support of an object, or for gesticulation in social interaction. In general, they have a limited potential in terms of supporting complex ADL. Nonetheless, they are a preferred choice of many amputees due to their overall simplicity and exceptional appearance [8].
- ii. **Body-Powered Prosthesis:** A body-powered prosthesis operates by attaching a cable and a harness around the shoulder opposite (contralateral) to the damaged arm. In a body-powered prosthesis allowing the opening/closing of hand (or gripper), the user opens the prosthesis by extending the contralateral shoulder and closes the prosthesis by relaxing the shoulder. Extending the contralateral shoulder pulls the cable and as the cable tightens, it opens the hand and vice-versa closes the hand. The downsides of this system are that it requires the usage of compensatory movements on the unaffected side and the use of a bulky cable/harness configuration makes the setup visually less appealing. Nevertheless, they are still preferred by many amputees, on account of the robust and easy-to-use control strategy and due to the availability of feedback via extended physiological proprioception [9].
- iii. **Self-Powered Prosthesis:** As the name suggests, a self-powered prosthesis is an electrically powered device and it uses motor(s) to make the required movement(s). Each motor represents a single degree-of-freedom (DoF). The polarity and amount of current given to a motor determines the direction and strength of the produced movement. For example, in a self-powered prosthesis allowing opening/closing of hand, the polarity of current given to the motor will determine if the hand (grip) is

opening or closing and, the amount of current will determine the speed at which the grip aperture changes; and ultimately, the exerted grasping force. Similarly, for wrist rotation, changing the polarity of current will switch the direction of rotation (i.e. it will switch between pronation and supination) and controlling the amount of current will determine the speed of rotation in either direction. The Michelangelo Hand from Ottobock (Germany) and the i-Limb from Touch Bionics (U.K.) are typical examples of self-powered prostheses (see Figure 1.1). The Michelangelo Hand has two DoFs (i.e. two motors), one for opening/closing the hand-grip and one for wrist rotation. The i-Limb has five individually controllable fingers and six DoFs – specifically, it has five motors for the flexion of each finger and one additional motor for the rotation of thumb. In real-life application, a prosthesis user should be able to control both the polarity and amplitude of current flowing through each motor. To achieve this, a human-machine interface (HMI) is used to connect the user and the prosthesis. The HMI, essentially, defines a scheme to transform bodily signals generated by the user (such as electrical signals from the muscles [10]) into movements on the prosthetic device, thereby allowing the user to control available functions by generating appropriate (predefined) bodily signals. Section 1.2 discusses various state-of-the-art HMIs used in research and commercial applications.

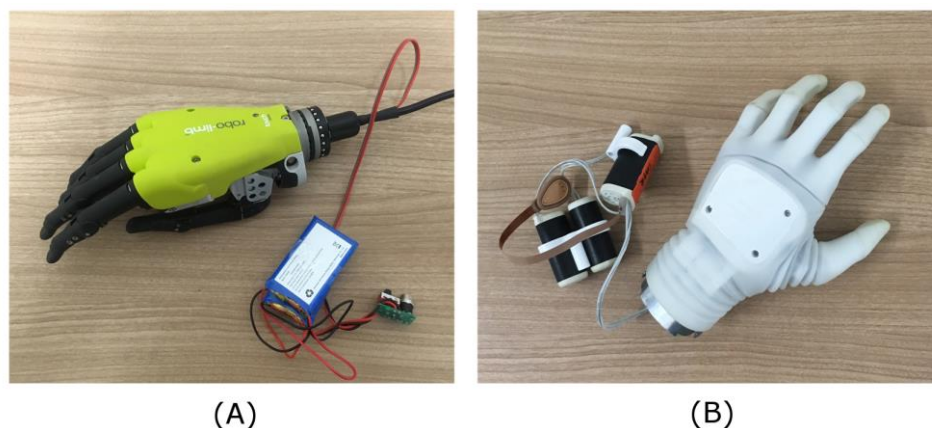


Figure 1.1: Examples of active self-powered prostheses. (A) The i-Limb hand from Touch Bionics U.K. (www.touchbionics.com). (B) The Michelangelo Hand from Ottobock, Germany (www.ottobock.com). The images also show the portable batteries which power the prostheses during daily use.

The three prosthesis types (mentioned above) are the choices currently available to patients. The final choice made by a particular patient depends on subjective preferences, level of amputation, medical recommendation, cost, etc.

During the last two decades, prosthetic hands have improved considerably in terms of overall appearance and offered functionality. But, the rejection rates of upper-limb prosthesis have remained exceptionally high, because the end-users find it very challenging to perform very simple ADLs such as eating, grooming, hygiene and manual work [11], [12]. According to a comprehensive literature review from 2007 [13], the rejection rates for self-powered prostheses were still about one third for paediatric and one fourth for adult patients. An important shortcoming of available devices is the *lack of embodiment*, i.e. these devices are perceived more as a tool and less like a part of one's own body. A large number of amputees fitted with an active body-powered or a self-powered prosthesis use their prosthesis in a passive way [12], [14]. The overall lack of embodiment arises due to the following reasons,

- i. **Lack of Feedback:** The human hand has a sensory-motor coupling, i.e. it is an integration of a sensory system and a motor system. The motor system allows us to perform different actions such as grasping and manipulation. The sensory system provides us a proprioceptive feedback (regarding the relative position of different body parts) and a haptic feedback (regarding the properties of material, touch, etc.). Most commercially available prosthetic devices provide, to some extent, a substitution for the lost motor function(s), but they do not provide any sensory feedback. Restoring the natural sensory feedback in upper-limb prosthetic devices is an important research topic [15], [16], but it will not be covered in the scope of this work.
- ii. **Poor Controllability:** Something that has not changed during the past 50 years is how prostheses are controlled. The mechanisms used to control body-powered prosthesis are inherently unintuitive and tiresome. The advent of self-powered prosthesis opened up a possibility to provide an artificial hand with a natural and intuitive control over multiple DoFs, but the commercially available HMIs for self-powered prosthesis still remain mundane and unintuitive for the user [17]. A number of novel HMI schemes have been proposed to overcome this limitation, but most proposed methods are not robust enough for practical applications (see Section 1.2). In essence, a HMI with a natural control over multiple degrees of freedom is yet to be realized [18].

In the past few decades, the aforementioned rejection-factors have inspired the scientific community to create a number of promising HMI schemes which could promote the day-to-day use of multifunctional prostheses. This thesis, especially, focuses on improving the

reliability and robustness of available HMI schemes. The next section will provide an overview on the HMI schemes currently used in research and commercial applications.

1.2 State-of-the-art interfaces for self-powered prosthesis

The available HMI interfaces for self-powered prostheses operate using a simple physiological phenomenon behind muscle contraction. During contraction, muscles generate electrical potentials called electromyograms (EMG), which can be measured by placing electrodes on the surface of the skin [10]. In amputated users, surface EMG (sEMG) signals are acquired from the muscles remaining in the residual limb. For real-life use, the user's EMG signals are provided as an input to the HMI and in response, the HMI activates the required prosthesis movement(s). Although an HMI can be operated using signals other than EMG (e.g. signals from the brain [19]), sEMG has remained the most important control signal for prosthetic applications since the 1950 [18]. This is due to its easy access (via surface electrodes) and close representation of the underlying muscle physiology. An HMI interfaced using EMG signals is called a myoelectric controller, and a self-powered prosthesis interfaced using EMG signals is called a myoelectric prosthesis.

Essentially, a myoelectric HMI defines an invariant mapping function between the user's EMG signals and the available prosthetic function(s). A number of HMIs have been proposed over the years, but the most commonly used interface is the classic two-channel control. In this system, a pair of electrodes is used to gather EMG signals from a pair of antagonist muscles (e.g. the wrist flexor and extensor muscle pair remaining in the amputated limb of a user). To control prostheses with a single DoF, such as a simple gripper, the user contracts the flexor muscle to close the grip and the extensor muscle to open the hand. The amplitude of the generated EMG signal (which depends on the contraction strength [10]) is used to determine the velocity of opening/closing movement; and ultimately, the grasping force when contact occurs during closing. In a multi-DoF prosthesis, a co-activation of both muscles is used to switch between DoFs [20]. Alternatively, the switching can also be done via prolonged activation of a single muscle [21] or via a simple button on the prosthesis [22]. The two-channel control works very well in case of single-DoF prosthesis, but it becomes very slow and cumbersome when used with a multi-DoF prosthesis, mainly, because switching is required each time the user wants to use a different function [17]. For instance, if the user wants to control the wrist joint after grasping a particular object (e.g. when grasping a water bottle for pouring), the system forces the user to cycle through all available prosthesis movements only to gain the control

over the wrist joint. Moreover, the two-channel control is non-intuitive because the same group of muscles is used to control different functions.

To overcome the limitations of two-channel control, researchers have proposed myoelectric control schemes based on machine learning (ML). It is well-known that, the EMG signal pattern associated with a particular motor task is distinct and repeatable [23]. This provides an opportunity to train one of the many well established ML algorithms to detect and identify different user-movements. With ML-based control, the user can activate a desired prosthesis function directly by producing a muscle activation pattern that was associated to that function during supervised training. A typical ML-based myoelectric control scheme is depicted in Figure 1.2, and its important components are, namely, the feature extractor, the ML controller, the supervised training-unit (required to calibrate the ML controller) and the post-processing unit.

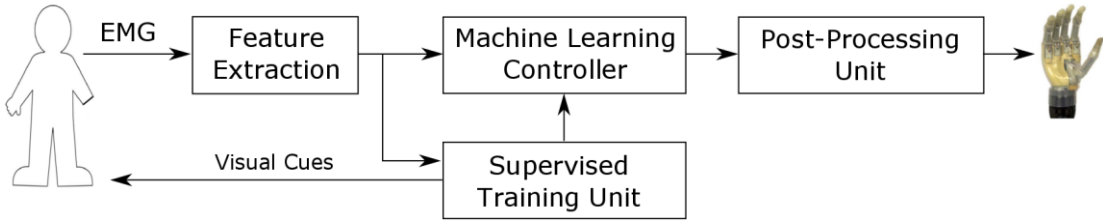


Figure 1.2: Typical scheme used to implement machine learning based myoelectric control for dexterous upper-limb prostheses.

For feature extraction, sEMG signals are acquired from several electrodes placed around the residual limb, and usually about 6 to 12 electrodes are used depending upon the size of the residual limb [24]. The acquired signals are band-pass filtered (usually in the range of 5-500 Hz) and then sampled at 1kHz [25]. From the band-pass filtered sEMG signals, time and/or frequency domain features are extracted over a sliding time window of certain length (e.g. 128 ms) and overlap (e.g. 32 ms) between neighbouring windows [26], [27]. Some of the most popular sEMG features are:

$$\text{Root Mean Square values, } RMS := \sqrt{\frac{1}{N_w} \sum_{k=1}^{N_w} (z_k)^2}$$

$$\text{Mean Absolute Values, } MAV := \frac{1}{N_w} \sum_{k=1}^{N_w} |z_k|$$

$$\text{Zero Crossings, } ZC := \frac{1}{N_w} \sum_{k=1}^{N_w} c; c = \begin{cases} 1; & \text{if } z_k \cdot z_{k+1} < 0 \\ 0; & \text{otherwise or if } |z_k - z_{k+1}| < 0.01V \end{cases}$$

$$\text{Slope Sign Change, } SSC := \frac{1}{N_w} \sum_{k=1}^{N_w} c; c = \begin{cases} 1; & \text{if } z_k \leq z_{k-1} \text{ and } z_k \leq z_{k+1} \\ 0; & \text{o/w or if } |z_k - z_{k\pm 1}| < 0.01V \end{cases}$$

$$\text{Waveform Length, } WFL := \frac{1}{N_w} \sum_{k=1}^{N_w} |z_k - z_{k-1}|$$

where z_k is the k^{th} sample of the observed N_w samples in a given time window. In practise, a *feature set* is constructed by extracting several features from each EMG channel. For example, the most commonly used feature set is the Hudgins time domain feature set [28], where MAV, ZC, SSC and WFL features are evaluated from each channel. A *feature vector* is formed by concatenating features evaluated from all channels, and it has size $d = N_{channels} \times N_{features}$ equal to the product of the number of channels (6~12) and the number of features extracted per channel ($N_{features} = 4$ for Hudgins feature set). In literature, a variety of features sets have been proposed with promising results [29]–[31].

The ML methods used for myoelectric control can be roughly divided into regression [32]–[34] and classification [35]–[37] approaches. The first ones yield a simultaneous and proportional estimate for all available prosthesis movements (e.g. simultaneously turning the wrist and closing the hand to grasp an object lying on a table), while the later provides a sequential and proportional control over each prosthesis function (e.g. first, turning the wrist to orient the hand and then, closing the hand to grasp object on a table). Both methods (regression and classification) represent the paradigm of a supervised machine learning problem. As it is customary in supervised ML, the data-set \mathcal{D} required to train the system is collected during a supervised training procedure. The supervised learning is accomplished by providing visual movement cues on a computer screen and asking the user to reproduce the indicated movement, using either the maximum voluntary contraction (MVC) for the movement [38] or graded contraction strengths (e.g. 30%, 60%, 90%) normalized to MVC [39]. The recorded data-set comprises of N (*feature vector, target value*) pairs $\mathcal{D} = \{(x_1, y_1), \dots, (x_N, y_N)\} \cong (X, Y)$, easily represented by a matrix $X \in R^{N \times d}$ juxtaposing all recorded feature vectors $x_i \in R^d$ and a matrix Y juxtaposing all target values y_i presented as visual cues. In case of regression, $y_i \in R^M$ with M representing the available DoFs on the prosthetic device, and for classification, $y_i \in S_{movements}$ where $S_{movements}$ is the set of all available prosthetic movements represented as discrete labels.

Let us sketch a running example for a regression-based ML controller. Consider the case of the Michelangelo prosthetic hand manufactured by Ottobock, endowed with one motor for the grip open/close function and one additional motor for the rotation of the wrist. In this case $M = 2$ and we assume that the first component of the matrix Y corresponds to grip

open/close and the second component corresponds to wrist rotation. Also, assume that, we are interested in training $K = 5$ actions: the rest action (associated to no movement), the hand-close action (associated to closing movement of the grip), the hand-open action (associated to opening movement of the grip) and pronation/supination actions for the wrist. In this case, each row in matrix $Y \in R^{N \times M}$ will represent one of the five actions, being $y_{rest} = [0, 0]^T$, $y_{close} = [1, 0]^T$, $y_{open} = [-1, 0]^T$, $y_{pronation} = [0, 1]^T$ and $y_{supination} = [0, -1]^T$. The simplest type of regression is the so-called *linear regression*, which can be used to determine a mapping matrix W that predicts prosthesis movement \hat{y} for any arbitrary input vector $\hat{x} \in R^d$ generated by the user, as follows:

$$\hat{y}_{1 \times M} = \hat{x}_{1 \times d} \cdot W_{d \times M} \quad (1.1)$$

The predicted movement $\hat{y}_{1 \times M}$ subsequently determines the velocity with which the M available prosthetic motors are moved. And, by assuming a linear model with Gaussian noise, the mapping matrix W can be calculated from the data-set $\mathcal{D} \cong (X, Y)$, as follows:

$$W = (X^T X + \lambda I_d)^{-1} X^T Y \quad (1.2)$$

where I_d is an identity matrix of order d and $\lambda > 0$ is a regularization term used to prevent overfitting [40]. The regression equation can be extended by applying the so called *kernel-trick*, to incorporate nonlinearities present in the training data. The most straightforward way is to pre-process each feature vector with a non-linear basis function $\varphi : R^d \rightarrow R^D$. The choice of φ is obviously crucial, and it can be chosen from a variety of well-known basis functions based on Fourier transform [41].

Furthermore, let us also consider how a classification based ML-controller in Figure 1.2 would work. In regression, the matrix Y comprised of real valued numbers (representing normalized velocity of motors), but for classification, Y contains discrete labels from the set $S_{movements}$ representing all available movements. In the previous example with $K = 5$ actions, this set would be $S_{movements} = \{Rest, Close, Open, Pronation, Supination\}$. The goal of classification is to determine a mapping function f that estimates the movement class $\hat{y} = f(\hat{x})$ for any input vector $\hat{x} \in R^d$ generated by the user. And, the velocity for the motor corresponding to the estimated movement class is determined by normalizing the sum of input amplitude values (e.g. sum of MAV features from all channels) by the sum of MVC amplitudes of the detected class. The simplest and the most popular classification function used for myoelectric control is the Linear Discriminant Analysis (LDA) [28]. And,

other well-known methods such as k-nearest neighbours (kNN) [42] and support vector machines (SVM) [43] have also been used for myoelectric control.

In myoelectric community, regression is considered very important for providing a natural control over all available prosthetic DoFs, as it can provide a simultaneous control over all DoFs, thereby mimicking natural movements of the human hand (e.g. simultaneously turning the wrist and closing the hand to grasp an object lying on a table). But, regression methods seem to work reasonably well only up to two DoF simultaneous control [44], and they are very susceptible to cross-talk (i.e. the activation of unwanted DoFs when trying to activate desired DoFs). Contrary to this, classification methods are suitable for a stable control (i.e. with less cross-talk) over more than 2 DoFs in a sequential manner. Hence, there is a trade-off between robustness and naturalness of control when choosing between classification and regression as the method of choice for myocontrol [45], [46].

One downside of using off-the-shelf ML methods, such as regression and classification, is that the underlying mathematical function(s) may or may not capture the latent physiological principles responsible for generating the observable EMG signals [47]. For example, it is hypothesized that the central nervous system uses muscle synergies as building blocks for generation of meaningful movement [48], [49]. And, mathematically speaking, muscle synergies can be seen as basis functions applied by the nervous system to move a large number of muscles using only a few independent control signals [50]. In myocontrol literature, this knowledge of underlying physiological principles has been exploited to directly estimate the latent synergy coefficients by using techniques like non-negative matrix factorization (NMF) [51], independent component analysis [47] or deconvolution [52]. For example, Jiang et al. used NMF to determine the latent synergy coefficients, which were then used for simultaneous and proportional control of 2 DoFs [51]. Next, it is also known that force production in a given task (e.g. grasping) relies on the coordination of a set of muscles and the EMG amplitude of activated muscles scales uniformly as a function of applied force [53], [54]. This physiological phenomenon is called the *principle of muscle coordination*. In myocontrol literature, the knowledge of muscle coordination has been applied to extract EMG features invariant against muscle contraction strength and then, standard ML methods (e.g. classification) were applied on the extracted features to achieve myoelectric control. For example, He et al. proposed a novel feature extraction scheme based on discrete Fourier transform to produce EMG features invariant against muscle contraction levels [55]. Recently, Al-Timemy et al. also proposed a feature extraction scheme to minimize the effect of muscle contraction strength, where the required

feature-set was derived by estimating the orientation between the power spectrum of the original EMG signal and its nonlinear version [56]. The extracted features were then given as inputs (for learning and prediction) to a traditional ML method (e.g. LDA). Lastly, it is worth mentioning that, although ML methods inspired from the underlying physiological principles are expected to gradually gained attention in the myocontrol community, standard methods like regression and classification still remain dominant in research application; partially due to the fact that, the latter has been tested more than the former under clinical and/or laboratory settings.

Despite decades of research and promising results under laboratory conditions, ML-based myocontrol methods have shown a limited clinical viability [57], [58]. There is only one commercially available solution (the COAPT complete control [59]) implementing a control based on pattern recognition, but it has still to prevail in clinical practices. Till today, the classic two-channel control remains popular in commercial use on account of its simplicity and robustness during real-life use. An important factor limiting the practical usage of ML-based myocontrol is the inherent non-stationarity of sEMG signals. sEMG is known to change under the influence of electrode displacement [60], [61], change in arm position [62], muscle fatigue, changing conductivity (e.g. due to perspiration, humidity, or temperature), etc. Important strategies proposed to overcome these issues have been summarized in the following subsection.

1.2.1 Methods to improve the robustness of ML-based myoelectric control

A traditional ML-based myoelectric controller defines a *single time-invariant* (stationary) mapping function between the user's EMG signals and prosthesis commands. Naturally, this stationary mapping approach encounters robustness problems on account of the inherent non-stationary behaviour of EMG signals. For example, unexpected changes in sEMG patterns during simple moving of the prosthesis can lead to misclassification of hand movements [63]. One simple approach to overcome such problems is to apply a set of error correction rules in the post-processing unit (shown in Figure 1.2). One common post-processing approach is the majority vote filtering [26], where the outputted decision at a given time represents the most frequently classified movement over the last several time steps. Another post-processing approach is the decision-based velocity ramp [64]. Here, a counter is associated to each class and upon detection of a particular class, the associated counter value for that class is increased while all other counter values are decreased. The velocity of the corresponding motor is then proportional to the counter's value. In practice,

the robustness comes for the fact that a small misclassification has a little effect on the overall value of the counter associated to the class that was observed consistently. Both post-processing approaches, thus, try to filter out spurious misclassifications occurring due to unexpected transient changes in sEMG signals. Furthermore, a slightly different post-processing approach based on parallel classifiers was proposed by [65], [66], where decisions made by multiple classifiers were compared and the prosthesis was forced to take *no action* when there was a disagreement between decisions made by individual classifiers.

Similar to the underlying ML controller, the post-processing rules defined in the beginning remain *static* (time-invariant) and do not account for variations in EMG over time. This shortcoming can be alleviated by performing the so-called *adaptation*, which is implemented by regularly collecting new training samples to adapt the ML. This adaptation can be either supervised, where the user updates the ML model occasionally when the performance degrades [67], or unsupervised, where the ML controller automatically updates itself based on some confidence measure that an incorrect decision was made [68]. Both methods for adaptation work well in case EMG signals slowly varying over time, but supervised adaptation has shown better performance [68]. In myocontrol literature, adaptation is sometimes also called incremental learning [69] or co-adaptive learning [70].

In general, post-processing rules and adaptation are good strategies to improve robustness in *unimodal* systems i.e. systems using *only* EMG as input for prosthesis control. A different strategy for achieving robust control is to use *multimodal sensor fusion*, i.e. utilizing other input modalities besides EMG to overcome its shortcomings [71]. This can be implemented in three different ways, namely, as a single-stage multimodal system, a multi-stage multimodal system or a semi-autonomous system (as illustrated in Figure 1.3).

In a single-stage multimodal system, features are evaluated separately for each modality and then given as inputs to a *single* ML controller. Gijsberts *et al.* proposed a single-stage multimodal classifier utilizing accelerometer data besides EMG and demonstrated that, the multimodal approach outperforms unimodal classification by ~5% when classifying 40 different hand and wrist movements [72]. Similarly, Fougner *et al.* used accelerometer data besides EMG to resolve the so-called *limb position effect* [73], i.e. an effect where a ML trained using single arm position data fails to generalize over different arm positions [74]. They demonstrated that the classification accuracy can be improved by simply adding *just one* 3-axis accelerometer on the forearm alongside the EMG sensors. Recently, Krasoulis *et al.* showed that including all modalities available in a typical inertial measurement unit (IMU), i.e. acceleration, gyroscope and magnetometer data, is better than using just one

modality (namely, acceleration) [75]. Apart from IMU derived modalities, other sensor modalities have also been used to achieve multimodal control [76]. For example, Jaquier *et al.* combined force myography (FMG) [77] with traditional sEMG to perform regression over wrist, hand and single-finger movements [78].

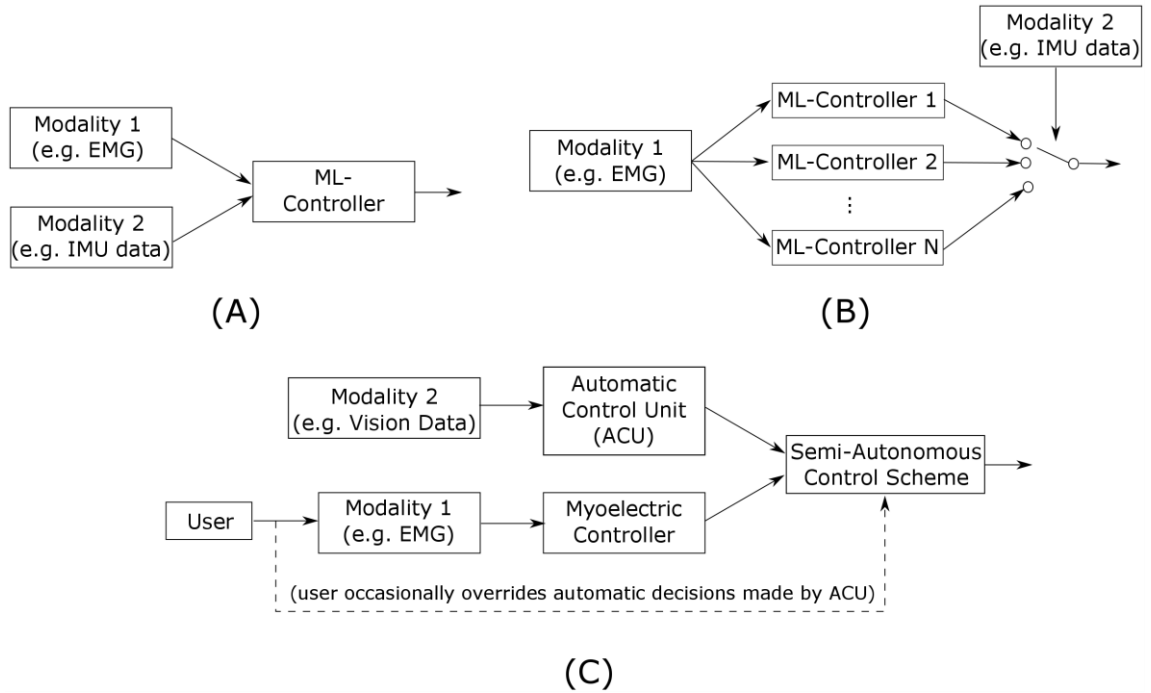


Figure 1.3: Three approaches for multimodal sensor fusion in upper-limb prostheses: (A) single-stage multimodal control system, (B) multi-stage multimodal control system and (C) semi-autonomous control system.

A multi-stage multimodal system uses ML to automatically detect different prosthesis states and switch between different ML-controllers depending on the detected state (see Figure 1.3B). Each ML-controller is optimized to operate in a specific state, and it utilizes a different input modality as compared to the ML responsible for switching. Geng *et al.* used a multi-stage system to compensate for the limb position effect. They measured the limb position using accelerometer data and switched between different unimodal EMG classifiers to mitigate the effect of limb position change [79], [80]; here, each unimodal classifier was training using EMG data collected in the corresponding arm position. This multi-stage switching approach provided more robust classification than simply inputting both EMG and accelerometer data into the same ML controller [79].

A semi-autonomous system integrates a conventional myoelectric controller with an automatic control unit (ACU) (see Figure 1.3C). Here, the end-user has complete control over the behaviour of the myoelectric controller, but the ACU is allowed to make

independent decisions based on information gathered via sensors. Essentially, a semi-autonomous system represents the paradigm of *shared-control*, where the responsibility of task completion is shared between the user and the system (ACU). For example, in [81], a simple contact sensor (a pressure sensitive transducer) was used to automatically select one of the two available prosthesis grasp types (pinch or lateral), and user-generated signals were used to control the movement of the prosthesis. In [82], a slip sensor was used to automatically detect slipping-events, and the applied grip force was spontaneously adjusted to prevent slipping of the object. Moreover, instead of having just one automatic function (as in the case of [81], [82]), a state machine can be used to activate between different automatic functions as required [83]. Next, instead of monitoring sensors placed inside the prosthesis, the ACU can also monitor sensors placed on the user to determine the onset of compensatory movements (e.g. slight raise of shoulder before grasping) and/or complete action(s) automatically. For example, [84] implemented automatic coordination between the (prosthetic) elbow joint and the movement of the (intact) shoulder via IMU. Recently, Markovic *et al.* used artificial vision and inertial sensors to determine the combined state of the user, the prosthesis and the environment [85], [86]. In this scheme, the ACU automatically adjusted the prosthesis parameter (e.g. wrist rotation and hand pre-shaping) to mimic natural interaction between the user and environment. And, by means of the simple two-channel myoelectric interface, the user was allowed to manually correct and fine-tune the automatic decisions made by the system.

Finally, it is also worth mentioning that, this work focuses on improving non-invasive methods used for myoelectric control. Today, however, researchers are also using surgical procedures (e.g. targeted muscle reinnervation [87]) and invasive electrode placement (e.g. osseointegration [63]) to improve the quality of EMG signal recording and thereby, the quality of myoelectric control. These topics, however, are beyond the scope of this thesis.

1.2.2 Methods used to validate myoelectric control

Evaluating the performance of a new myoelectric control is a difficult task [88]. Over the years, researchers have proposed numerous methods for evaluating (quantifying) the performance of myoelectric control. Principally, these evaluation methods can be divided into three categories: offline evaluation, online evaluation using virtual tasks and online evaluation using functional tasks.

For offline evaluation, a participant/user is asked to perform different movements (repetitively) as indicated via visual cues, and the corresponding sensor data is stored for

offline analysis. The most commonly used offline evaluation scores are either R^2 error [89] (for Regression) or classification accuracy [90]. The R^2 error measures the amount of variability a regressor can estimate with respect to the reference cues, and a higher R^2 value indicates a better regressor. For a classification-based control, the classification accuracy measures the percentage of correctly classified movement-labels with respect to the indicated movement cues, and thus, a higher classification accuracy indicates a better classifier. These offline measures were regularly used in the past, but recent investigations show a poor association between offline evaluation and clinical relevance [88]. For this reason, nowadays, new control methods are evaluated online using virtual and/or functional tasks.

An online evaluation is better than an offline evaluation because it keeps the user in loop with the control and as a result accounts for human adaptation to the system. The most popular online virtual task is the Fitts' law test [91], wherein subjects are required to control a cursor on the screen in order to reach random targets on the screen [92], [93]. Lately, virtual reality based online evaluation frameworks are being used to simulate real-life prosthesis use (with a certain level of abstraction), where users are required to steer a prosthetic-avatar in 3D space [94]. The advantage of an online virtual task is that it offers a realistic test scenario while alleviating the need for a full system implementation, e.g. one does not need to invest time in designing sockets for the prosthesis. Nevertheless, the influence of important realistic factors such as weight of the prosthesis, weight of the grasped object, stump dynamics, quality of socket fitting, etc. is not accounted for in virtual tests [95]. These factors can only be incorporated by using an online functional task.

The simplest and the most commonly used functional task for online evaluation is the Box-and-Blocks test [96] shown in Figure 1.4A, wherein the participants are required to transfer as many blocks as possible, from one compartment to the other, within 60 seconds. The outcome measure of performance is the number of blocks transferred per minute. The disadvantage is that the test requires only opening and closing of the hand and thus, it provides an evaluation only for a single DoF. A test commonly used to evaluate two DoFs, specifically the opening/closing of the hand and pronation/supination of the wrist, is the clothespin test [97] shown in Figure 1.4B. It requires a participant to pick up three pins from a horizontal bar, rotate them and place them on a vertical bar. The outcome measure is the time required to successfully transfer three pins. Other functional tests widely used for evaluating myoelectric control with respect to ALD are, namely, the Southampton Hand

Assessment Protocol (SHAP) [98], Jebsen-Taylor Hand Function Test [99], unilateral below elbow test (UBET) [100], etc.

In practice, a functional evaluation is considered very important to assess the operation of a new myocontrol method [88]. And therefore, all myocontrol methods developed as a part of this work were evaluated using functional tasks.

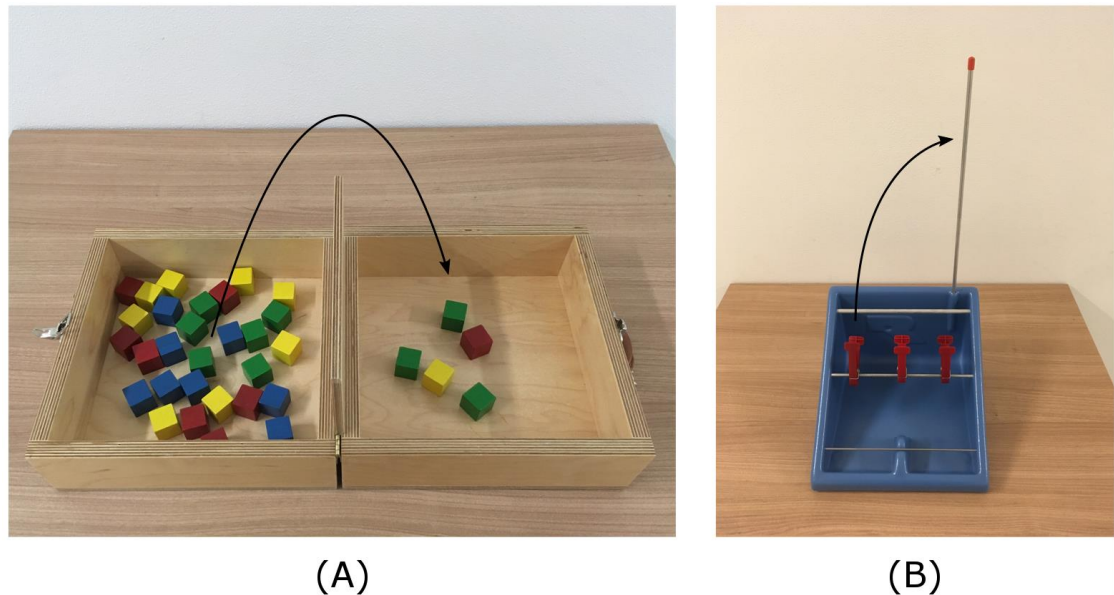


Figure 1.4: Examples of functional tasks used to validate myoelectric control. (A) The Box-and-Blocks test. (B) The clothespin test.

1.3 The main aims of this thesis

Most ML-based methods used for myoelectric control follow the conventional pattern recognition paradigm, where training data is collected using a supervised procedure and a mathematical function is fitted over the collected data to determine an invariant mapping scheme between the user's EMG patterns and available prosthesis functions. This approach may not be optimal, as it might not consider two important aspects associated with online use of prosthesis. First, the context information describing the state of the system and/or environment, which might have the potential to facilitate prosthesis control, is not considered by most ML controllers. Second, the information regarding the underlying physiological phenomenon responsible for the generation of sEMG patterns may not get implicitly modelled by the applied mathematical function [47]. Therefore, *the main objective of this thesis is to develop methods which can exploit context and physiology information to improve the performance/robustness of myocontrol schemes based on machine learning.*

The context information required for describing the state of the system and/or environment can be either derived from additional sensors (placed in a prosthesis) or directly from the user. The latter gives to the user(s) a possibility to interact with the ML controller by generating appropriate context information (which both the user and ML can understand). Most myoelectric systems available today do not provide any possibility of an active interaction between the user and the prosthesis, and as a result, there remains a gap between the user and the underlying mechanisms used to control the prosthesis. Recently, this gap has been recognized by prosthetic manufacturers like Touch Bionics, who nowadays provide a smartphone application that can be used to customize the classic two-channel controller. But now, with the expected increase of ML-based myoelectric systems in the commercial market, it becomes necessary to develop new methods which can be used to bridge the gap between the user and an advanced ML-based myoelectric controller. Chapter 2 presents one such method, which exploits context information generated by the user to improve ML-based myocontrol for different ADL.

A downside of relying solely on user-generated context information to improve control is that the burden of generating information lies on the user. Therefore, multimodal sensor fusion can/should be used to derive context information from additional sensors placed either on the user or inside the prosthesis. Previously, context information available from additional sensors was used to either improve the performance of a typical ML controller (single- and multi- stage control in Figure 1.3A-B) or implement semi-automatic control (Figure 1.3C). The sensor fusion methods used to improve ML control, available today, follow the conventional pattern recognition procedure, where all available information (EMG + other modalities) is presented as input (for training and prediction) to the ML controller, and the underlying mathematical function is expected to implicitly capture the regularity in the presented data. But, the available context information has not been used to determine the state of the prosthesis and/or the user, which can be very helpful in deciding how the ML controller should/must be adapted in order to counter possible problems arising during different states of online use. For example, misclassifications (due to unexpected changes in EMG patterns) may occur when the prosthesis is moving [63] and in such cases, additional sensors (e.g. accelerometers) can be used to detect if the prosthesis is moving and adapt the ML controller to minimize the odds of misclassification when moving; e.g. minimizing the odds of dropping a glass of water while moving. Chapter 3 presents one such multimodal sensing scheme, which models different states associated with online prosthesis use and adapts the ML controller to improve its robustness. Lastly, it is worth

mentioning that, applying multimodal sensor fusion to achieve semi-automatic prosthesis control is an important research topic, but this work will focus only on designing control strategies that are manual.

Next, with most off-the-shelf ML methods, the mathematical function fitted over the training data is assumed to implicitly model the latent physiological principles behind generation of EMG patterns. This assumption may not be true for many commonly applied ML methods. As outlined in Section 1.2.1, there are a number of new ML methods which consider latent physiological principles to achieve myoelectric control. These methods use the knowledge of physiology to either create a model for the underlying physiological process (e.g. by identifying synergies [47], [51], spike trains [52]) or extract EMG features which are classified using off-the-shelf ML methods [55], [56]. But, the knowledge of physiological principles has been rarely focused on improving practical aspects of myoelectric control, such as reduction of training time, minimization of computational cost, simplification of hardware, etc. Chapter 4 presents one such machine learning method, which utilizes the knowledge of physiological principles to minimize training time and computational cost required for online myoelectric control.

In total, this thesis comprises of three main studies focusing on improving the robustness of ML-based myocontrol by: a) integrating context information from the user (Chapter 2), b) integrating context information from on-board sensors (Chapter 3), and c) integrating physiology information to improve the modelling of EMG (Chapter 4). The concepts presented in these individual studies are highly compatible with each other and therefore, it is possible to merge them into a single machine learning scheme – which can exploit both context and physiology information to improve the performance/robustness of ML-based myocontrol. In doing so, the larger goal is to improve the overall end-user experience and thereby, the clinical acceptance-rate of ML-based myoelectric prostheses.

Context- and Physiology-aware Machine Learning for
Upper-Limb Myocontrol

2 USING TASK-SPECIFIC CONTEXT TO CUSTOMIZE MYOCONTROL

This chapter has been based on the recently published paper:

G. K. Patel, M. Nowak and C. Castellini, "Exploiting Knowledge Composition to Improve Real-Life Hand Prosthetic Control" in IEEE Transactions on Neural Systems and Rehabilitation Engineering, vol. 25, no. 7, pp. 967-975, July 2017.

2.1 Motivation

In the commercial market, prosthetic manufacturers (like Touch Bionics, U.K.) have introduced an ‘interactive’ smart-phone App that allows users to customize their classic two-channel controller. In this App, users can setup different daily living situations (e.g. at home, at work, recreation, sports, etc.), and upon selection of a particular situation, the App configures the prosthesis to provide custom grip patterns required for that situation. This new possibility to customize the myoelectric control has been welcomed by the users [101], but it still suffers from the inherent limitation of the underlying two-channel controller, i.e. a co-contraction is required to switch between the selected grip patterns. As mentioned in Section 1.2, ML-based schemes can be used to overcome this limitation. With the advent of ML-based myoelectric control in the commercial market (e.g. COAPT system [59]), it is now important that the users of these systems also enjoy the possibility of customizing their prosthesis as required. With this goal in mind, this chapter discusses a novel approach that brings to the users a possibility to actively ‘customize’ their ML-based myoelectric control. Specifically, the new method makes use of *task-specific context information* (e.g. the user is

ironing clothes, the user is shopping, etc.) to select custom grip patterns required for the task. Thus, the new method provides a similar level of interactivity as the currently available smart-phone App for the two-channel control, but with the added value that the underlying control scheme is ML based.

This custom selection of grip patterns has a clear application in case of a multi-articulated prosthesis offering a large variety of grip types. As the number of offered grip types increases, it become very challenging for the end-users to produce distinct sEMG patterns required to operate the corresponding ML-based control. For didactic purposes, Figure 2.1A depicts pattern overlap between six different sEMG patterns in a 2D space of projected RMS features. If all patterns were to be used during online control (Figure 2.1A), an attempt to perform the power grip would probably result in problems due to an unwanted interaction with the flat grip. But, reducing the set of allowed patterns may lead to a more stable control, since the interaction with undesired pattern(s) can be eliminated (Figure 2.1B). This possibility of having a reduced set of patterns can be helpful in certain real-life situations, for example while Ironing a piece of cloth only two grips types are needed, namely, power grasp (to hold the iron) and pinch grip (to manipulate the cloth).

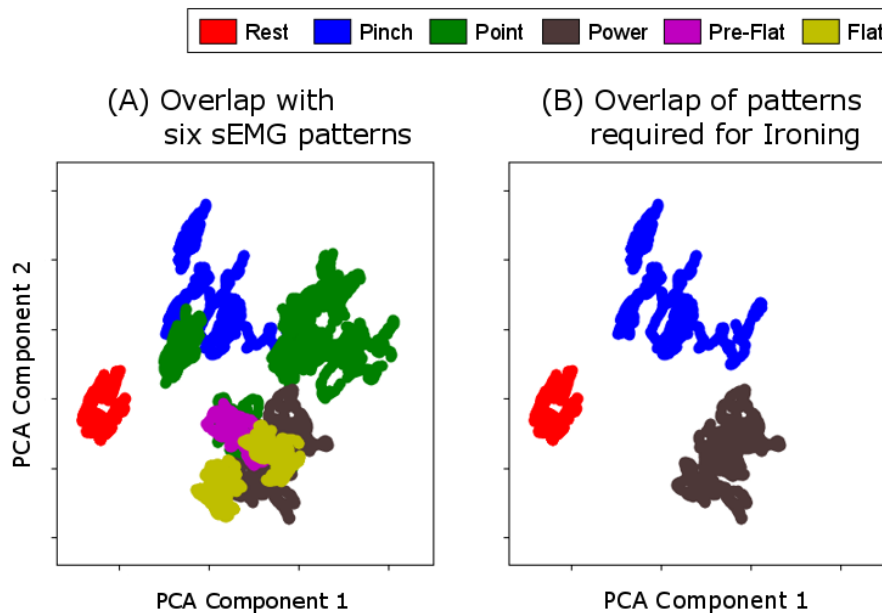


Figure 2.1: Visualizing the pattern overlap between six different sEMG patterns in a 2D space of projected RMS features. The presented data was collected from one of the participants during the experiment (see Section 2.3 for more details). Principle Component Analysis (PCA) was used to reduce the dimensions of RMS features extracted from eight sEMG channels and the explained variance was 72%.

2.2 Modular regression to adapt task-specific context

Consider a user who wants to operate the i-Limb prosthetic hand (from Touch Bionics, UK) using a regression-based ML control. And for the sake of simplicity assume that, the user wants to use $K = 6$ different grip types, namely, the power grasp, pointing index, two-digit pinch, pre-flat, flat-grip and rest (see description in Table 2.1). In order to train these patterns, the user follows a supervised training procedure and generates a data-set $\mathcal{D} = \{(x_1, y_1), \dots, (x_N, y_N)\} \cong (X, Y)$ containing sEMG features $x_i \in R^d$ corresponding to target prosthesis actions $y_i \in R^M$. For the i-Limb hand, $M = 6$ represents the 6 motors available in the robotic hand – one motor for the flexion of each finger and two motors for the rotation and flexion of the thumb each. As discussed in Section 1.2, regression can be used to determine a matrix W that predicts action $\hat{y} = \hat{x}W$ for any user-generated input vector $\hat{x} \in R^d$.

Traditionally, the matrix W is evaluated once in the beginning and remains fixed throughout usage. As described in Section 1.2 and Equation 1.2, the matrix W can be calculated by using the *full* data-set $\mathcal{D} \cong (X, Y)$ as follows,

$$W = (X^T X + \lambda I_d)^{-1} X^T Y \quad (2.1)$$

This traditional approach of using the full data-set \mathcal{D} to evaluate W will be called *full regression* (FR). Understandably, FR does not allow the user to customize ML control as and when required. This, however, can be achieved by employing the newly designed *modular regression* (MR) approach. In MR, the $K = 6$ available patterns are organized as separate modules, and the user is allowed to include/exclude a particular module (or pattern) as required. To implement MR, the data-set \mathcal{D} is first organized into $K = 6$ modules as follows,

$$\mathcal{D} = \left(\begin{array}{c} \left[\begin{array}{c} X_{pw} \\ X_{po} \\ X_{pi} \\ X_{fl} \\ X_{pf} \\ X_{re} \end{array} \right] , \left[\begin{array}{c} Y_{pw} \\ Y_{po} \\ Y_{pi} \\ Y_{fl} \\ Y_{pf} \\ Y_{re} \end{array} \right] \end{array} \right) \quad (2.2)$$

where, $X_j \in R^{N_j \times d}$, $Y_j \in R^{N_j \times M}$ and $N = \sum_{j=1}^K N_j$ for $j = 1, \dots, K$ patterns. And, each matrix Y_j consists of N_j juxtaposed copies of target values representing a particular prosthesis action (see Table 2.1). For example consider Y_{pw} for the power grasp,

$$Y_{pw} = \begin{bmatrix} [1 & 1 & 0 & 1 & 1 & 1 & 1] \\ \vdots \\ [1 & 1 & 0 & 1 & 1 & 1 & 1] \end{bmatrix} = \begin{bmatrix} y_{pw} \\ \vdots \\ y_{pw} \end{bmatrix} \quad (2.3)$$

Any matrix Y_j represents only the minimum and maximum activation of each target motor, and the regression interpolates the intermediate values required during online control. This greatly simplifies the training procedure as the user only needs to enact maximum voluntary contractions for each movement being training [38], [67].

Table 2.1: Different actions which can be performed using the i-Limb hand and the corresponding motor-commands (y_{action}). For the vector y_{action} , assume that the first component of the vector corresponds to thumb flexion, second component corresponds to thumb adduction, third component corresponds to the motor controlling the index finger flexion and so on for the remaining three fingers.

Actions	y_{action}	Description
Power	$y_{pw} = [1 \ 0 \ 1 \ 1 \ 1 \ 1 \ 1]$	All fingers flexed, thumb abducted and flexed.
Point	$y_{po} = [1 \ 0 \ 0 \ 1 \ 1 \ 1 \ 1]$	All fingers flexed but Index, thumb abducted & flexed.
Pinch	$y_{pi} = [1 \ 0 \ 1 \ 1 \ 0 \ 0 \ 0]$	Index and Middle finger flexed, thumb abducted and flexed.
Pre-Flat	$y_{pf} = [0 \ 1 \ 1 \ 1 \ 1 \ 1 \ 1]$	All fingers flexed, thumb adducted but not flexed.
Flat	$y_{fl} = [1 \ 1 \ 1 \ 1 \ 1 \ 1 \ 1]$	All fingers flexed, thumb adducted and flexed.
Rest	$y_{re} = [0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0]$	All fingers at rest, thumb abducted but not flexed.

Next, the terms $X^T X$ and $X^T Y$ in Equation 2.1, which contain information regarding all K actions, can be rewritten to represent each action separately,

$$X^T X = \sum_{j=1}^K X_j^T X_j; \quad X^T Y = \sum_{j=1}^K X_j^T Y_j \quad (2.4)$$

And, since Y_j is a juxtaposition of N_j identical vectors y_j , the equivalence $X_j^T Y_j = \bar{X}_j y_j^T$ can be established, where, $\bar{X}_j = \sum_{k=0}^{N_j} x_{kj}$.

Therefore, Equation 2.1 can be re-written as follows,

$$W = \left(\sum_{j=1}^K X_j^T X_j + \lambda I_d \right)^{-1} \sum_{j=1}^K \bar{X}_j y_j^T \quad (2.5)$$

In this way, W can be (re-)constructed by storing K doublets $(X_j^T X_j, \bar{X}_j)$ for each target activation y_j . The proposed MR approach, thus, gives a possibility of building a reduced model by plugging only some of the doublets into Equation 2.5. For instance, if a particular task only requires a subset $K' \subseteq K$ of all actions, a reduced model $W_{K'}$ can be evaluated on-the-fly as follows,

$$W_{K'} = \left(\sum_{j \in K'} X_j^T X_j + \lambda I_d \right)^{-1} \sum_{j \in K'} \bar{X}_j y_j^T \quad (2.6)$$

In theory, users can employ MR to build one specific model $W_{K'}$ for each ADL, allowing them to use the best possible model at all times. For example, while ironing a piece of cloth, the model $W_{K'}$ may contain only the two required grip types, namely, power grasp and pinch grip. It was hypothesised that the systematic usage of MR would improve the accuracy of prediction, and therefore, the reliability of the myoelectric control. The experimental procedure used to verify this hypothesis has been outlined in the upcoming section.

Lastly, notice that, the FR and MR equations can support incremental learning and therefore, it was always possible to update the prediction model (W or $W_{K'}$) by adding new training samples for individual grip patterns. This is particularly useful for countering the effect of changes in sEMG signals over time or due to fatigue [69]. From here on, the term *Pattern Update* will be used to describe the process of re-computing a new prediction model by adding new training samples for a given pattern. Mathematically speaking, a new set of training samples $\mathcal{D}^+ = \{(x_{j,1}^+, y_j), \dots, (x_{j,N}^+, y_j)\} \cong (X_j^+, Y_j)$ for the j^{th} pattern can be used to update the current doublet for the j^{th} pattern as follows,

$$(X_j^T X_j, \bar{X}_j)_{updated} = (X_j^T X_j, \bar{X}_j)_{current} + (X_j^{+T} X_j^+, \bar{X}_j^+) \quad (2.7)$$

Moreover, for the online experiment, the FR and MR equations were extended by applying the Random Fourier Feature (RFF) kernel, as suggested in [67]. This involves pre-processing each sample $x \in R^d$ with a non-linear sinusoidal transform $\varphi : R^d \rightarrow R^D$, which projects each EMG feature vector x_d to a higher-dimensional feature space with D dimensions. The transformed data $\varphi(x)$ was, then, used for both learning and prediction.

2.3 Materials and Methods

In order to check whether the usage of MR would yield a better myocontrol with respect to FR, an experiment was conducted with fourteen able-bodied participants performing functional ADL tasks. The experiment was conducted according to the Declaration of Helsinki [102]. During the experiment, the participants wore a splint, on which a left-sided i-Limb Revolution prosthesis (from Touch Bionics, U.K.) was mounted as shown in Figure 2.2. The robotic hand was controlled wirelessly using a custom-made serial-over-Bluetooth circuit. The sEMG signals were also acquired wirelessly using a *Myo* bracelet manufactured by Thalmic Labs (www.myo.com) with $d = 8$ sEMG channels. And, for each sEMG channel, the RMS envelope was evaluated every 5 ms over a window size of 150 ms and then, the RMS values were filtered using a low-pass Butterworth filter of order 1 and cut-off frequency 2Hz. The FR and MR based control schemes were implemented as a stand-alone C# program on a standard PC equipped with two Bluetooth dongles, one to acquire sEMG data from the Myo bracelet and the second to control the prosthesis. Both FR and MR were operated using filtered RMS features extracted from $d = 8$ EMG channels.

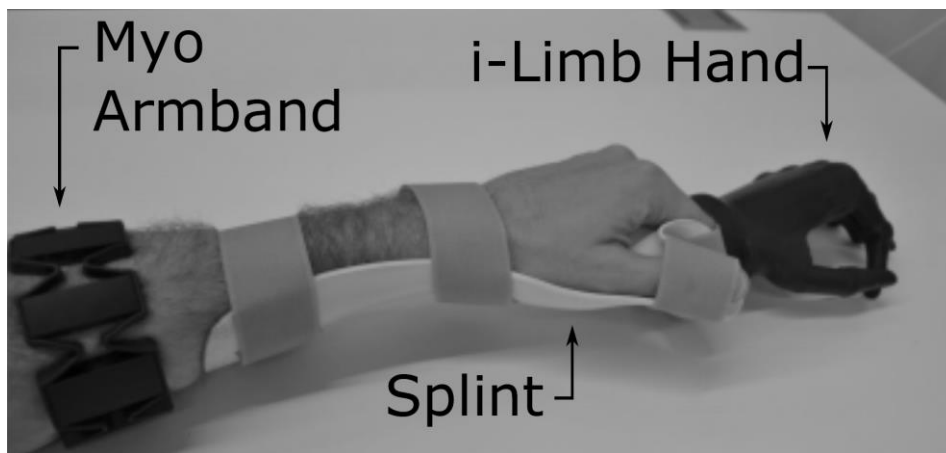


Figure 2.2: The experimental setup comprising of the i-Limb Revolution prosthetic hand mounted on a splint and the Myo Armband with eight sEMG sensors.

The functional tasks performed during the experiment were designed to reflect frequently required ADLs. The participants were able to complete each functional task using a specific subset of $K = 6$ actions mentioned in Table 2.1. A description of each task is as follows,

- i. Ironing: A piece of square-shaped cloth and a standard iron were placed on a table. The participants were instructed to use a power grasp to hold the iron and a pinch grip to handle the cloth. First, the participants had to iron the cloth four times right to left with the prosthetic hand; and for safety reasons, the iron was never turned-on.

The participants were allowed to use their right hand to steadily hold the cloth while ironing. Next, the cloth was flipped using both hands and the back-side was again ironed four times. Finally, the cloth was folded twice (corner to corner) using both hands, reducing its size to one quarter of the original size. In this task, two grip patterns were required with MR, namely, power grasp – to grab the iron, and pinch grip – to grab the cloth for folding.

- ii. Library: A starting point was marked approximately 4 meters away from a book-shelf and a book was placed on the top compartment of the shelf. Additionally, a desk with a keyboard was placed near the shelf, alongside the path between the starting point and the shelf. After walking towards the shelf from the starting point, the participants grabbed the book using a power grasp and placed it on the desk near the keyboard. Next, the participants used a point index grip to enter the title of the book via the keyboard. Lastly, before returning back to the starting point, the participants placed the book back to its original location using a power grasp. Here, two grip patterns were required with MR, namely, power grasp – to grab the book, and point index grip – to type the name.
- iii. Arranging: Two pairs of crosses (×) and vertical lines (|) were marked on an alleviated platform. The task consisted of placing two flyers and two bottles each on the | and × markers respectively. To start the task, participants took six flyers in their right hand and thereafter, they used a combination of pre-flat and flat grasp to pick-and-place two flyers from the right hand onto the vertical line (|) markers. The flyer pick-and-place task was accomplished by first executing the pre-flat grip and then placing the flyer between the thumb and the radial side of the index finger, thereafter the thumb was closed by executing the flat grasp. Furthermore, six bottles were placed between the alleviated platform and the participants, who then used a power grasp to pick-and-place any two bottles on the cross (×) markers. In case, if the participants would lose the grasped bottle during pick-and-place (e.g. due to accidental opening), they would grasp a new bottle from the block of six bottles placed in front of them. Here, three grip patterns were required with MR, namely, power grasp – to grab the bottles, pre-flat and flat grip – to handle the flyers.
- iv. Shopping: A starting point was marked approximately 4 meters away from a shelf, on which three items, namely, water-bottles, fruits and chocolates were placed at different heights. A standard shopping basket was given to the participants in their right hand. A keyboard was placed on a desk alongside the path between the shelf

and the starting point. At first, the participants walked towards the shelf with the basket in their right hand and then, used their prosthetic hand to pick-and-place items from the shelf into the basket. Specifically, the participants used a power grasp to pick-up a bottle, a pinch grip to pick-up a fruit and a chocolate. Next, they used a point grip to enter a PIN (1973) on the keyboard, before walking back towards the starting point. Thus, three grip patterns (power, pinch, point) were required for control with MR.

- v. Playing Jenga: A small Jenga tower with 30 blocks was pre-constructed for the participants, with two blocks coming out of the tower on the left-side. Firstly, using a point grip, the participants were asked to push the two blocks inwards, until a part of the block comes out on the right-side. Next, with the sound hand, the participants removed the two blocks from the right-side of the tower. Finally, they used a pinch grip to pick-and-place two Jenga blocks on top of the tower, from a pile of blocks placed on the side. Thus, two grip patterns were required for MR, namely, point index grip – to push the blocks out and pinch grip – to place new blocks on top of the tower.

The outline of the experimental procedure is shown in Figure 2.3. Before performing the aforementioned functional tasks, the participants were introduced to the concept of myoelectric control and were informed that the system could be trained by producing the MVC of required patterns for 5 seconds. The training data for a given pattern was collected by verbally synchronizing the start of the training phase, which was immediately followed by the robotic hand mimicking the pattern being trained. Meanwhile, the participant would contract the forearm muscles to produce a comfortably high fingertip force on the splint and the corresponding sEMG signals were recorded as training data. The end of the training phase was subsequently indicated by the prosthesis, as it would release the configured action (being trained) and quickly go back to rest. The required prediction model (W or $W_{K'}$) was speedily evaluated by the system, making it possible to immediately transfer the control (of prosthesis) back to the participant at the end of the training phase. Furthermore, all participants were informed that the experiment was tailored to compare two different control approaches, but specific details regarding both approaches (e.g. exact name of each approach, or how one approach was better than the other, etc.) were not disclosed, in order prevent any expectation bias. Soon after, the participants were challenged to perform two round (Round 0 and Round 1) of five tasks using FR and MR as the control method. The sequence in which the two control methods were administered was

randomized order across participants, i.e. half of them used FR followed by MR and the other half had the order reversed; plus, the order in which the tasks were performed was also randomized across participants.

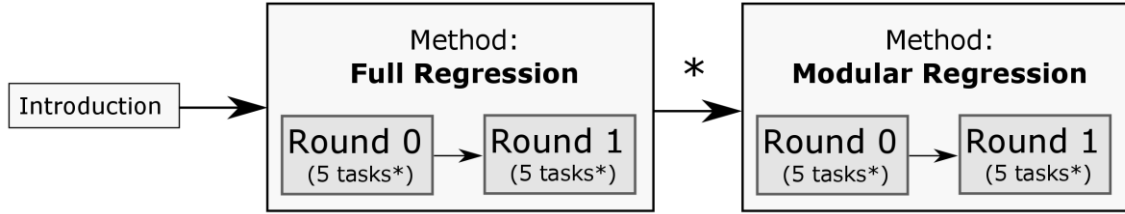


Figure 2.3: The experimental protocol used to compare the two control methods, namely, Full Regression (FR) and Modular Regression (MR). The sequence in which both methods were administered across participants was randomized, and the order in which the five tasks (Ironing, Library, Arranging, Shopping and Playing Jenga) were performed was also randomized (indicated by an asterisk).

The Round 0 was used to familiarize the participants with the tasks and the control method. Here, the participants used the respective control method (FR or MR) for the first time, and the training data required to evaluate the prediction model (W or $W_{K'}$) was gathered in a cumulative fashion by *training-upon-requirement*, i.e. training a given pattern as-and-when it was required for a task, until all five tasks were successfully completed. The prediction model evaluated in Round 0 was carried over to Round 1, and the participants were always allowed to add more training data to the model. The decision to add more training data in Round 0 was collectively taken by the participant and the experimenter via verbal interaction, whereas, in Round 1, the request to perform a Pattern Update was independently expressed by the participant. Next, for Round 1, the same five tasks were administered again (in the same order as in Round 0) and three outcome measures of performance were recorded, namely, the *Task Completion Time* (TCT), number of *Pattern Updates* (PUs) requested and the number of *Errors* incurred. The measure PUs indicate the number of times a Pattern Update was requested (done by adding new training data for 5 seconds) in Round 1. And, the measure *Errors* was calculated by counting the number of objects accidentally dropped during each task and the number of accidental hand openings while typing in the Library and Supermarket task. Lastly, notice that, no outcome measure was evaluated for Round 0, as the aim of this round was to familiarize the participants with the task and the control method.

The experimental protocol can be better understood by considering the experimental timeline presented in Figure 2.4. This particular timeline is taken from a participant

belonging to the group which was administered FR before MR. In the beginning, no training data was available to perform task T1 (in Round 0 with FR), and therefore, the participant trained all the required patterns (power, pinch, point, rest) for task T1. Later on, due to poor performance, the patterns pinch and rest were incrementally updated by collecting new training data (for 5 seconds each). And, after T1 was successfully completed, tasks T2, T3 and T4 were performed, during which the patterns power (in T2) and point (in T3) were incrementally updated. So far, the patterns pre-flat and flat were not required during tasks T1 to T4, and therefore, their training data was not available in the system. Following the paradigm of training-upon-requirement, the patterns pre-flat and flat were trained for the first time before performing task T5, during which the pattern pre-flat was updated twice incrementally. The prediction model evaluated at the end of Round 0 was used to perform tasks in Round 1, where the outcome measures TCT, PUs and Errors were registered for evaluation. In this particular case, PUs = 3 as three updates were requested, namely, one for power (in T2) and two for point (in T3 and T4). Next, with MR, the required training data was collected in Round 0, again by following the paradigm of training-upon-requirement and allowing incremental updates. Thereafter, Round 1 was performed and the three measures were registered for evaluation.

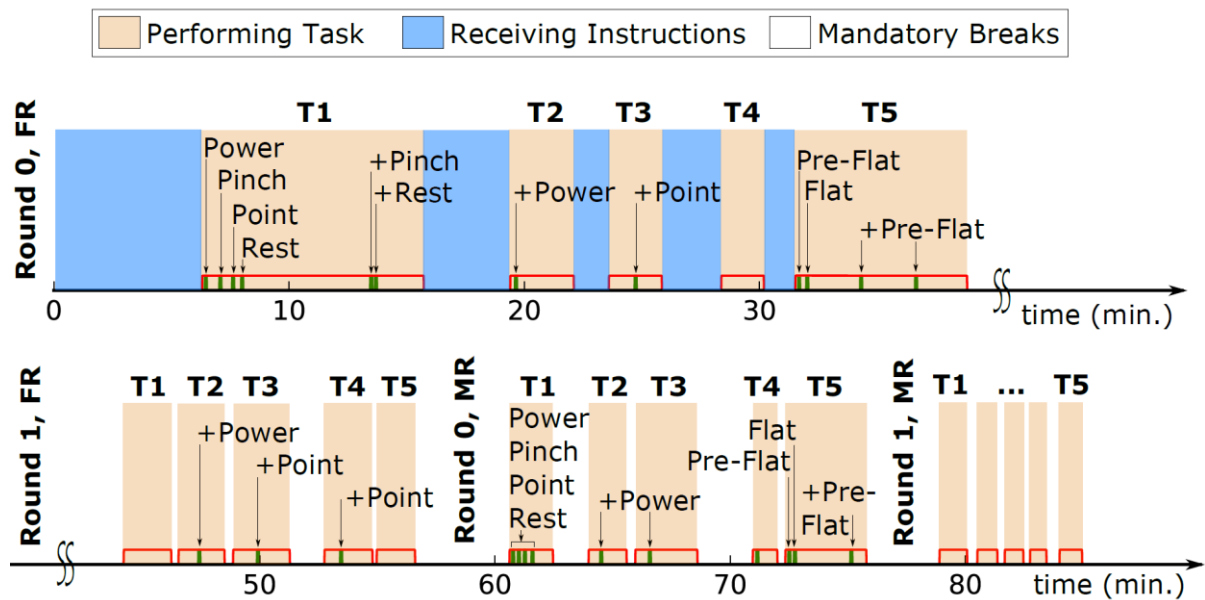


Figure 2.4: An experimental timeline. The participant performed two rounds of five tasks with FR, followed by two rounds with MR. Symbols: '+' – Pattern Update for the given pattern, T1 - Shopping, T2 - Ironing, T3 - Library, T4 – Jenga and T5 - Arranging.

The evaluated outcome measures for FR and MR control were compared using the Wilcoxon signed rank test for dependent samples, and a Bonferroni correction was applied wherever necessary. All results are reported in terms of median and interquartile range (IQR) – denoted as median {IQR}. The required statistical analysis was performed using the software STATISTICA (from Dell, US) and the threshold for significance was set to $p = 0.05$.

2.4 Experimental Results

Figure 2.5 summarizes the results for TCT obtained using FR and MR. The median overall time required to complete the tasks using MR (1.1{0.8–1.7} minutes) was significantly lower than using FR (1.9{1.3–2.4} minutes) with $p < 0.001$. Moreover, the average overall TCTs with FR and MR were 2.0 ± 0.9 and 1.4 ± 0.8 minutes respectively, and this represents an average overall decrease of 30% in TCT with MR.

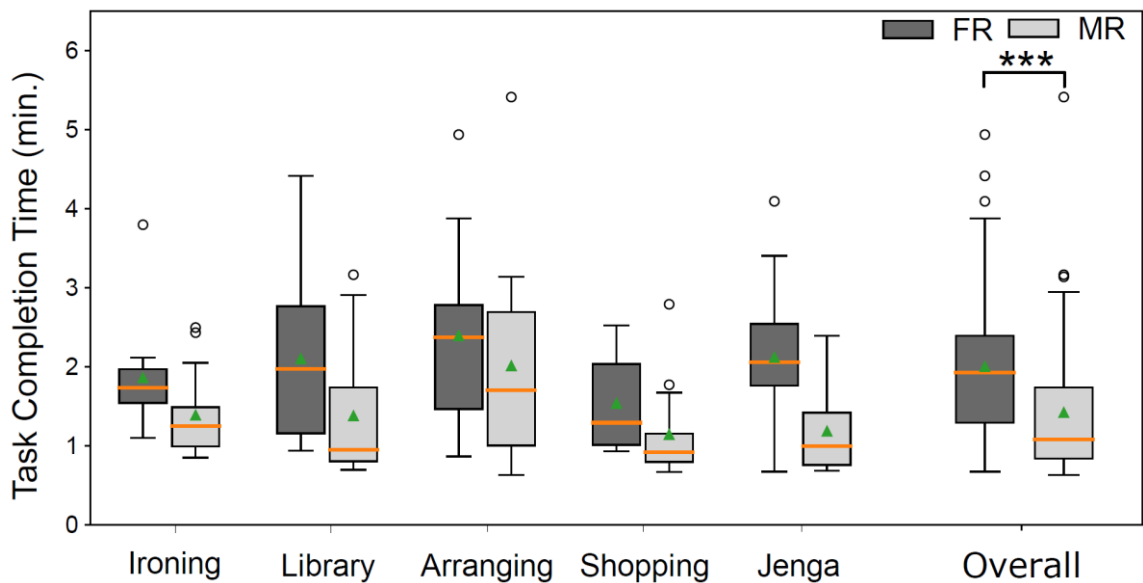


Figure 2.5: Summary of the comparison between FR and MR, showing the median Task Completion Time (TCT) in minutes during Round 1 for five different ADLs. ('***' indicates $p < 0.001$, the horizontal line within a box indicates median, the box indicates IQR, the triangle indicates mean and the circles indicate outliers).

The number of PUs required in Round 1 with MR (1{0-1}) were significantly lower than FR (2.5{2-3}) with $p < 0.05$. But, the total number of Errors incurred in Round 1 with MR (4{1.3-6.5}) and FR (5{4.3-6.8}) were not statistically different ($p > 0.05$). If the total Errors observed in Round 1 were normalized by the number of task (i.e. 5), it approximates to about one error per task with either method. Therefore, it can be suggested that, the

observed Error count per task was too small to make a meaningful comparison between the two approaches.

Lastly, at the end of the experiment, all participants were asked to report their subjective impression/feeling on which approach according to them provided better control. Twelve out of fourteen participants reported that the myoelectric control in the rounds with MR was better, and two participants reported perceiving no difference between the two approaches.

2.5 Discussion

This chapter presented a new method called Modular Regression (MR) which provided a simple way to interact with a regression-based myoelectric controller. With MR, each prosthesis action (grip) was formulated as a module, which the user could insert/remove as required to best accomplish a given task. In an online experiment, the new MR approach was compared with the classic FR approach of using all available actions at all times. The results indicated that, overall, the five ADL tasks in the experiment could be performed significantly faster with MR in comparison to FR. Thus, MR offers not only a simple way for the users to interact with the underlying control, but it also provides functional improvement during online control.

The experimental protocol used to compare the performance of FR and MR was designed to mimic the everyday use of a multi-articulated prosthesis. Ideally, a new training dataset must be provided to the ML controller each time the donning and doffing of sEMG electrodes occurs. And, because the number of offered grip types is high, a user cannot be expected to train all available patterns every morning (when donning and doffing of electrodes may occur before and after shower). At first, the user would train a set of few most frequently required patterns and, then, gradually add new patterns by following the paradigm of training-upon-requirement. As new patterns are added, the overlap in feature space would increase (Figure 2.1A), resulting in a possible degradation of performance over time. In such cases, the presented MR method can be used to minimize the interaction with undesired patterns. Although, depending upon usage and experience, prosthesis users are expected to improve their performance over time [103]. In case of an experienced user, the performance with FR might already so good that the contribution of MR becomes redundant. Therefore, MR could be implemented as an optimal component, i.e. initially activated to support a naive user and then deactivated once the user becomes experienced.

The available user-interactivity and functional improvement with MR can be extended, in the future, by adding more interactive features. One possibility is to update the underlying ML control each time the user indicates that a failure (Error) has occurred. This could, for example, possibly decrease the 4{1.3-6.5} Errors observed with MR during the experiment. Another possibility would be that the ML controller could regularly monitor the quality of generated sEMG patterns and provide suggestions (or tips) on which combination of patterns may or may not work properly. This could, for example, motivate the user to improve the discriminability of patterns belonging to a possibly problematic combination, by using previously proposed methods to improve signal generation [104], [105].

An inherent drawback of relying on user-generated information is that the burden of generating context information lies on the user. Thus, before introducing additional interactive features, care must be taken to maintain a balance between the amount of context information required from the user and the corresponding performance gained. Alternatively, the burden on the user could be minimized by automatically generating context information from additional sensors. For example, with MR, not only can the subset of patterns be chosen willingly by the user, but the prosthetic device can potentially recognise situation(s) and automatically apply a reduced set of patterns tailored for the detected situation. One possibility is to mark various daily living objects (e.g. Iron) using a so-called *grip-chip* [101], and the automatic selection of the required patterns can be applied when the prosthesis finds itself in close proximity to one of these *grip-chips* (e.g. automatic selection of power and pinch grip when close to Iron). Another possibility is to detect various daily living situations via vision and/or GPS (Global Positioning System) sensors. For example, the act of approaching an ATM to withdraw cash is a situation that could be automatically detected via vision and/or GPS sensors. By doing so, an intelligent prosthesis could assume that pointing action (to enter one's own PIN on a keypad) and flat grip (to insert/extract a credit card and to take the cash) are likely to occur, and could automatically generate a model to predict on the reduced set of most likely patterns.

The idea of using additional sensors to automatically adapt the ML controller is especially interesting, since it allows automatic generation of context information and minimizes the burden on the user. But unfortunately, prosthetic hands available today are not equipped with vision or GPS sensors and therefore, the selection of ADLs has to remain manual for now. Nevertheless, many commercially available devices are equipped with embedded sensors that can provide proprioceptive (e.g. acceleration, orientation) and/or exteroceptive (e.g. force, aperture) information. Indeed, this information can be exploited to improve

prosthesis control based on ML. One example is to use an embedded contact (force) sensor to detect if an object is being held by the prosthesis, and once the contact is established, the goal should be to minimize the odds of dropping the object. This could, also, possibly decrease the 4{1.3-6.5} Errors observed with MR during the experiment. The upcoming chapter discusses a multimodal sensors fusion technique which was developed to improve the robustness of ML-based control by exploiting context information available from sensors embedded within prosthesis.

3 CONTEXT-DRIVEN MACHINE LEARNING FOR MYOCONTROL

This chapter has been based on the recently published paper:

G. K. Patel, J. M. Hahne, C. Castellini, D. Farina, S. Dosen, "Context-dependent adaptation improves robustness of myoelectric control for upper-limb prostheses", in Journal of Neural Engineering, vol. 14, no. 5, September 2017.

3.1 Motivation and the novel concept

Most commercial and research prostheses are equipped with sensors that provide proprioceptive (e.g. joint angles) and/or exteroceptive (e.g. grasping force) information. However, this information has not been used to improve prosthesis control based on machine learning. The main idea of this chapter is to show how context information from various on-board prosthesis sensors can be exploited to improve ML-based myoelectric control.

As discussed in section 1.2, a conventional myoelectric controller defines a time-invariant mapping function between the user's muscle activity and the prosthesis commands. This mapping function, usually, remains fixed (static) during use. Contrary to the conventional approach, this chapter proposes a novel method (shown in Figure 3.1) to adapt this mapping online based-on the state of the prosthesis (context information) detected via embedded proprioceptive and exteroceptive sensors. The new control framework is called the context-driven machine learning control (cxMLC), and it comprises of two interacting units: a context-aware component (CAC) and a conventional machine learning controller (MLC).

The CAC infers context information (e.g. prosthesis states) from the real-time sensor data, and this information is used to modulate the parameters of the MLC. Thus, the mapping between the user's EMG signals and the generated prosthesis command (defined by MLC) becomes reactive to the inferred context information (and does not remain static).

In particular, this chapter presents one specific implementation of the general conceptual scheme in Figure 3.1. The presented cxMLC implementation contains a CAC unit which modulates only the output parameters (activation thresholds) of the MLC, leaving the internal parameters of MLC unchanged. Another implementation of the general scheme in Figure 3.1 could include a CAC unit that adapts the internal parameters (e.g. class prior probabilities) of the MLC based on available context information; but this implementation is beyond the scope of this chapter.

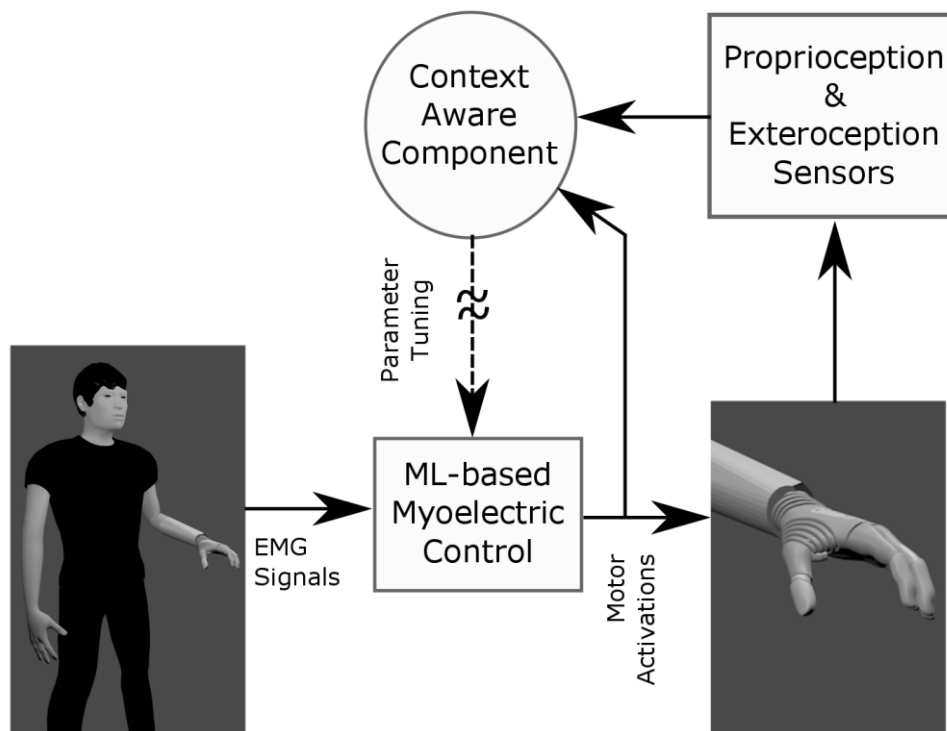


Figure 3.1: Conceptual scheme of the proposed context-driven ML-based myoelectric control (cxMLC). The scheme integrates a conventional ML-based myoelectric controller (MLC) and an automatically driven context-aware component (CAC). The context inference is realized by the CAC using proprioceptive (e.g. orientation, gyroscope) and exteroceptive (e.g. force, aperture) sensors embedded within the prosthesis. And, the parameters (e.g. activation thresholds) of MLC are adjusted in real-time based on the inferred context information.

3.2 Context-Driven Control

The incremental Ridge Regression with Random Fourier Features, discussed in [67], was adapted to implement the conventional myoelectric control (MLC). The implemented MLC was used for the sequential and proportional control of five prosthesis functions available in the Michelangelo Hand, namely, wrist pronation, wrist supination, palmar grasp closing, lateral grasp closing and hand opening. These prosthesis functions were controlled by EMG signals obtained from five muscle contraction patterns, namely, wrist radial deviation, ulnar deviation, wrist flexion, closed fist and wrist extension, respectively. The sEMG signals required to operate the MLC were acquired using the wireless Myo armband (from Thalmic Labs, CA) with $d = 8$ sEMG channels. For each sEMG channel, RMS features were evaluated every 5 ms over a window size of 150 ms and then, filtered using a low-pass Butterworth filter of order 1 and cut-off frequency 2Hz. The most recent sample of the filtered RMS envelope was given as input to the MLC and the output was the predicted movement (\hat{y}_i) for the five prosthesis functions. Next, a threshold T (default value 0.2) was applied to the predicted movement \hat{y}_i to remove the uncertainty at low contraction levels and then, the obtained value was multiplied by a fixed gain G (of 1.25 for grip open/close and 1.00 for wrist rotation). Thus, the relationship between the command generated by the user (\hat{y}_i) and the control command sent to the prosthesis (\hat{y}'_i) can be formulated as,

$$\hat{y}'_i = G_i(\hat{y}_i - T_i) \quad (3.1)$$

where, \hat{y}_i and \hat{y}'_i denote the predicted activations before and after thresholding and gain multiplication were applied; and their values remained in the range [0,1], where 1 indicated maximum activation. The subscript i denotes different prosthesis functions, namely, palmar (P), lateral (L), hand open (O) and wrist rotation (R). The sequential and proportional control was implemented by retaining the maximum value from the set $\{\hat{y}_P, \hat{y}_L, \hat{y}_O, \hat{y}_R\}$ of predicted movements and setting all other values to zero. Thereafter, the term \hat{y}'_i was used to determine the velocity with which the corresponding prosthesis motor was moved.

The CAC was implemented as a finite state machine (FSM) representing typical phases (states) arising during prosthesis use (see Figure 3.2). Specifically, five different prosthesis states, namely, Free, Closing, Grasping, Holding and Moving were considered in this implementation. And, as shown in Figure 3.2, the state-transitions were triggered by monitoring online sensor data coming from three different sensors embedded within the prosthesis socket, namely, the force and grip aperture sensor in the Michelangelo prosthesis (providing normalized force (F) and aperture (A)) and the IMU in the Myo armband

(providing orientation (α, β, γ) and net angular velocity $|\omega|$). In each state, a set of rules was activated for post-processing of the commands \hat{y}_i generated by the MLC (see Figure 3.3). The aim of this post-processing was to minimize the unwanted/spurious activation of different prosthesis functions and therefore, the thresholds T_i were adapted to suppress/eliminate the unwanted activation of specific functions.

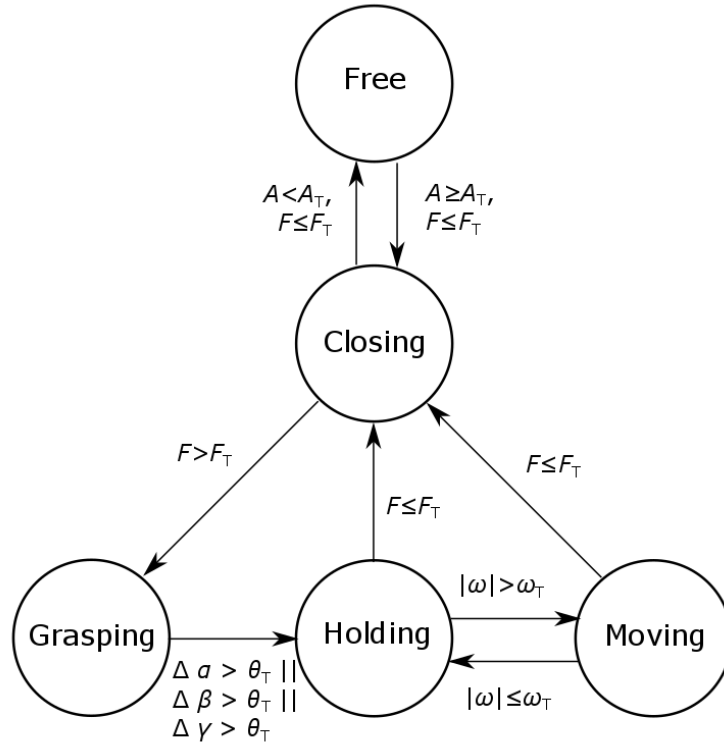


Figure 3.2: State machine for context aware component (CAC). The state machine detected five prosthesis states: Free, Closing, Grasping, Holding and Moving. The conditions for transition between the states, indicated next to the arrows, were based on comparing the real-time sensor data (from embedded gyroscope, force and aperture sensors) to predefined thresholds. The annotations are: A – normalized grip aperture, F – normalized grasping force, (α, β, γ) – elbow orientation and $|\omega|$ – elbow angular velocity. The threshold values were: $A_T = 0.2$, $F_T = 0.02$, $\theta_T = 10^\circ$ and $\omega_T = 1 \text{ rad/s}$.

Figure 3.4 shows an example of how different CAC state-transitions were triggered during online control. Initially, the hand/prosthesis was in the Free state and the threshold T_i was set to its default value of 0.2. Next, the user started closing the hand (either using palmar or lateral grip) to grasp an object and soon, the aperture threshold $A \geq A_T$ was reached. The CAC entered the Closing state to indicate that the hand was configured into a specific grasp, ready to enclose the object. To assist the user in forming a stable grip, the threshold for wrist rotation was increased to reduce the odds of accidental wrist rotation(s) during grip

closure. Next, the force sensor detected contact with the object ($F > F_T$) and so, the Grasping state was activated. In this state, the user adjusted the grasping force as required, before preparing to lift the object. Here, the forearm orientation (α, β, γ) and the final grasping force were registered upon entry and exit from this state, respectively (as indicated in Figure 3.3).

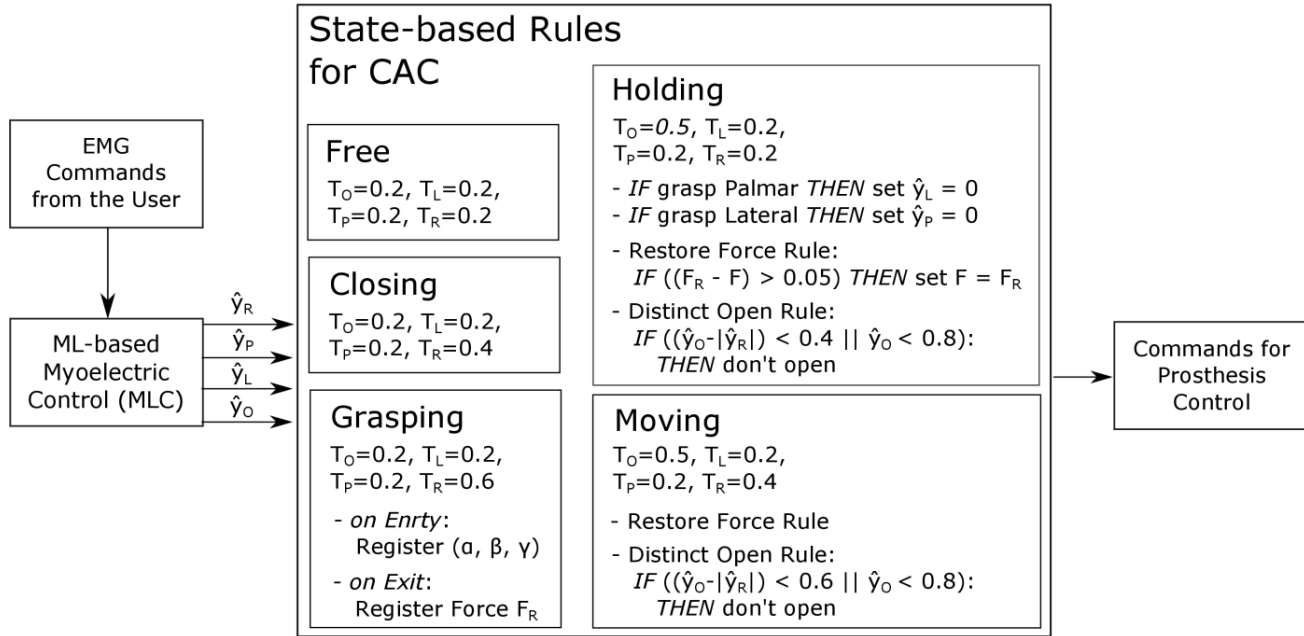


Figure 3.3: State based rules for context aware component (CAC). In each of the five prosthesis state, a pre-defined set of rules was used to post-process the outputs of the MLC. See the text for a detailed description of these rules. The annotations are: \hat{y}_i – the predicted movement, T_i – activation threshold, R – Rotation, L – Lateral, P – Palmar and O – Hand Opening.

In the next step, the user lifted the object. This action was easily detected by monitoring the difference between the current forearm orientation and the orientation that was registered in the Grasping state. Once the object was lifted ($\Delta\beta > \theta_T$), the CAC entered the Holding state. The aim of this state was to allow the user to manipulate the object while simultaneously preventing an accidental opening of the hand. Therefore, the wrist rotation threshold was set back to the default value (to allow easy manipulation) and the hand opening threshold was increased (to prevent an accidental drop). Additionally, to open the hand in this state, the user was required to produce an unambiguous opening command (by generating a strong signal) with no concomitant rotation command. This was enforced by the “distinct open” rule (see Figure 3.3), which assured that the hand opened only when the user clearly indicated an explicit intention to release the object. This rule would not block all spurious opening commands, but it would weaken their effect. A weak spurious opening command could still decrease the applied grasping force, and multiple such commands

generated over time could eventually reduce the contact force down to zero, leading to an unintentional dropping of the object. To prevent this, the “Restore Force” rule was implemented to readjust the grasping force to the level registered when exiting from the Grasping state. This was the force level that the user adjusted as appropriate for the object, just before the object was lifted. Additionally, in the Holding state, the grip type orthogonal to the grip in-use was disabled i.e. if the palmar grip was used to grasp the object, then the lateral grip was disabled and vice versa.

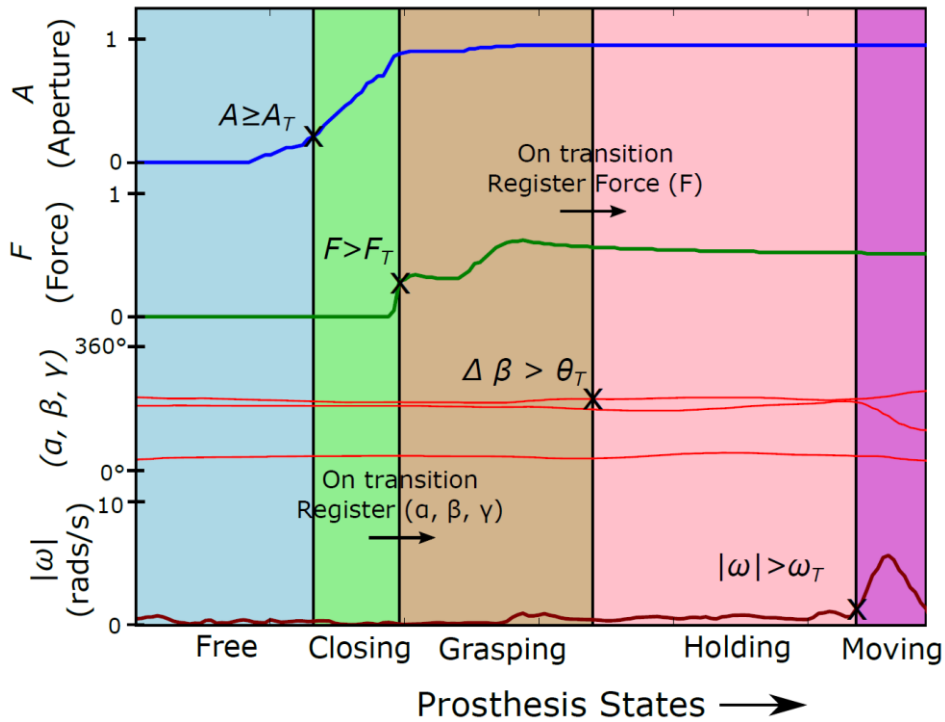


Figure 3.4: An example of how different CAC state transitions were triggered during online control with cxMLC. The annotations are: A – normalized grip aperture, F – normalized grasping force, (α, β, γ) – elbow orientation and $|\omega|$ – net elbow angular velocity. And, the thresholds for state-transitions were: $A_T = 0.2$, $F_T = 0.02$, $\theta_T = 10^\circ$ and $\omega_T = 1\text{rad/s}$.

Lastly, the Moving state was activated when the user started moving with the prosthesis ($|\omega| > \omega_T$). In this state, the wrist rotation threshold as well as the threshold for the distinct open rule was increased in compared to the Holding state (see Figure 3.3). The aim was to stabilize the prosthesis configuration, i.e., to maintain a fixed grasping force and wrist orientation while moving. Once the moving would be finished (detected by $|\omega| < \omega_T$), the CAC would enter back into the Holding state. Lastly, it is important to note that, no prosthesis function was actually disabled in any of the states – it was just made more or less difficult to activate. Put differently, in some states, the user was required to exert more

effort to indicate a clear determination for activating a certain function, whose threshold was increased to eliminate spurious activations.

An experiment involving functional tasks was conducted to examine the benefits of utilizing CAC besides MLC for prosthesis control. During this experiment, participants were administered two control methods, namely, MLC and cxMLC (i.e. CAC + MLC), in a randomized order. For MLC, the prosthesis was controlled by applying a fixed threshold on the commands generated by the user, whereas for cxMLC, the CAC was used to post-process the generated commands as explained above. The experimental procedure used to compare MLC and cxMLC has been outlined in the next section.

3.3 Materials and Methods

An experiment was conducted with ten able-bodied participants performing three different functional tasks designed to test the robustness of myoelectric control. This experiment was conducted according to the Declaration of Helsinki [102]. The experimental setup worn by the participants is shown in Figure 3.5. It comprised a wireless *Myo* armband (from Thalmic Labs, CA) and a left-sided Michelangelo prosthesis (from Ottobock, DE) mounted on a bypass socket. The armband incorporated eight channels for sEMG acquisition and an IMU for measuring the forearm orientation (α, β, γ) and net angular velocity ($|\omega|$). The armband was placed approximately 5 cm below the elbow joint of the left arm and it was oriented such that the positive X-axis of the IMU was pointing towards the volar side of the wrist. The Michelangelo prosthesis had an inbuilt force and aperture sensor, which were used to acquire normalized force and aperture values corresponding to the maximum grasping force (~ 100 N) and hand aperture (~ 11 cm for palmar and ~ 7 cm for lateral grasp), respectively. All prosthesis functions (grasping and rotation) were controlled wirelessly using velocity commands \hat{y}'_i , obtained after post-processing the user-generated commands \hat{y}_i (see Equation 3.1). The MLC and cxMLC based control schemes were implemented as a software program on a standard PC equipped with two Bluetooth dongles, one to acquire sEMG data from the *Myo* bracelet and the second to control the prosthesis. Furthermore, as seen in Figure 3.5, an additional wireless IMU (MTx, Xsens, NE) was also placed on the upper-arm to measure its orientation. The positive X-axis of this IMU was aligned with the X-axis of the *Myo* armband, pointing towards the elbow crease. The sensor data coming from this IMU was not used for online control, but it was only used for the assessment of user movement during the experiment. Thus, in the future, all components (eight sEMG

electrodes, forearm IMU, force and aperture sensor) required for online control can be integrated into a single socket, leading to a self-contained solution.

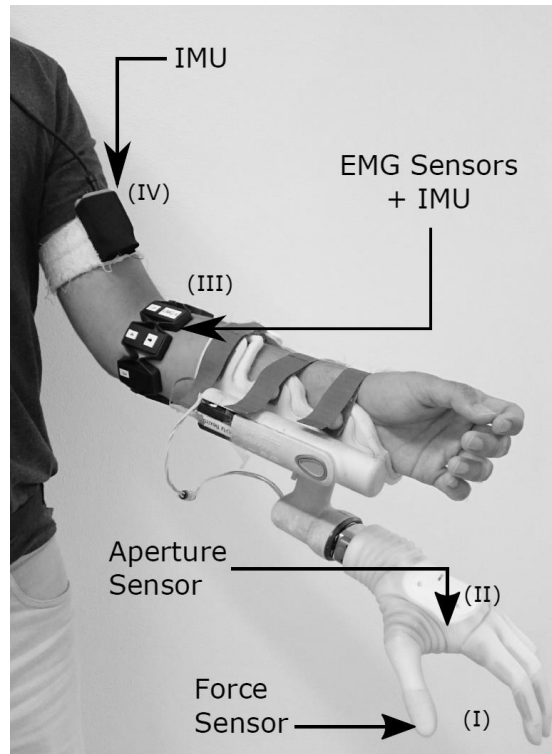


Figure 3.5: Experimental setup. It comprised a Michelangelo prosthesis with an embedded force (I) and aperture sensor (II), a Myo armband (III) with eight sEMG channels and an embedded IMU sensor, and an additional IMU (IV) to measure the upper-arm orientation. Importantly, the IMU sensor placed on the upper-arm (IV) was not required for the online control of prosthesis, but it was only employed for monitoring user movement.

An outline of the experimental protocol is shown in Figure 3.6. First, the participants were introduced to the concept of myoelectric control and then, the system components were placed on their left arm (as shown in Figure 3.5). Next, the participants were made to stand in an upright position with their elbows flexed at 90° and the training data required to calibrate the MLC was collected by producing the MVC of each pattern for 5 seconds. The training data for the pattern rest (corresponding to no movement) was collected using both static and dynamic conditions. For the static-rest, the participants were asked to relax their forearm muscles for 5 seconds while maintaining the elbow at 90° flexion. For the dynamic-rest, the participants were asked to move their elbow and shoulder while keeping their forearm muscles relaxed. Specifically, the following four dynamic-rest movements were trained: (1) repetitive elbow flexion/extension between 0° and 90° for 5 s. (2) repetitive medial and lateral rotation of the shoulder (i.e. forearm moving left and right) for

5 s, (3) repetitive shoulder flexion between 0° and 90° for 5 s and (4) repetitive shoulder abduction and adduction for 5 s. These dynamic-rest movements were included in the training to minimize the effect of spurious EMG activations observed during dynamic arm movements. The collected training data was used to train the MLC prediction model and then, the quality of MLC control was examined by asking the participants to produce the command \hat{y}_i for each prosthesis function in four different arm positions, namely, forearm down, elbow flexed, arm frontally extended and arm laterally extended (as shown in Figure 3.7). Specifically, an on-screen digital oscilloscope (embedded within our software) was used to visualize the activations generated by the MLC and the quality of control in each arm position was examined by asking the participants to produce/modulate \hat{y}_i in the range 0-80%. If the participants were not able to modulate the activation in the range 0-80%, the MLC control was deemed not good enough and the respective pattern was incrementally trained in that position by producing the MVC for 5 s. This was done to improve the quality of MLC control and also to ensure the same baseline quality of control across participants.

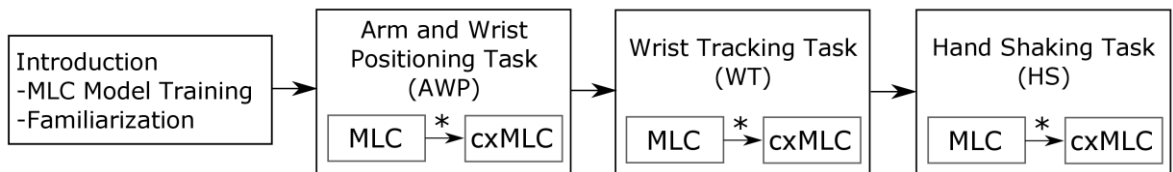


Figure 3.6: The experimental protocol used to compare conventional (MLC) and context driven (cxMLC) ML-based myoelectric control. Three tasks were administered to the participants sequentially (AWP, WT and HS), and each task was performed using both control methods, but the order in which the two control methods were administered was randomized across participants (indicated by an asterisk).

After the MLC training, participants were made to stand in front of a table with adjustable height, where the Box-and-Blocks setup (Figure 1.4A) was placed approximately 20 cm away from them. Two types of wooden blocks were used during this experiment (each compliant with a particular prosthesis grip): (i) the standard cubical wooden block with the side length of 2.5 cm to be grasped with a palmar grip and (ii) a modified cuboidal wooden block with the dimensions of $2.5 \times 3.0 \times 7.5$ cm to be grasped with a lateral grip. And, before starting a countdown timer, either 30 cubical blocks or 6 cuboidal blocks were placed in the left compartment of the setup. A standard PC monitor was placed approximately 125 cm away from the participants to provide visual task instructions. Furthermore, an arm position detector was calibrated for identifying the participants' arm position during functional tasks.

This calibration was done by asking the participants to bring their arm in each of the four arm positions AP1-4 (shown in Figure 3.7), where the pitch angle for the forearm and the pitch and yaw angles for the upper-arm were registered. The online identification of arm position was done by comparing the real-time angles with the registered values, wherein a tolerance/mismatch of $\pm 30^\circ$ was permitted. This online identification of arm position was not used for prosthesis control, but only to ensure that the participants were correctly following the experimental protocol. Lastly, all participants were made to familiarize with each functional task by performing it for approximately 5 minutes using MLC. The description of each functional task follows.

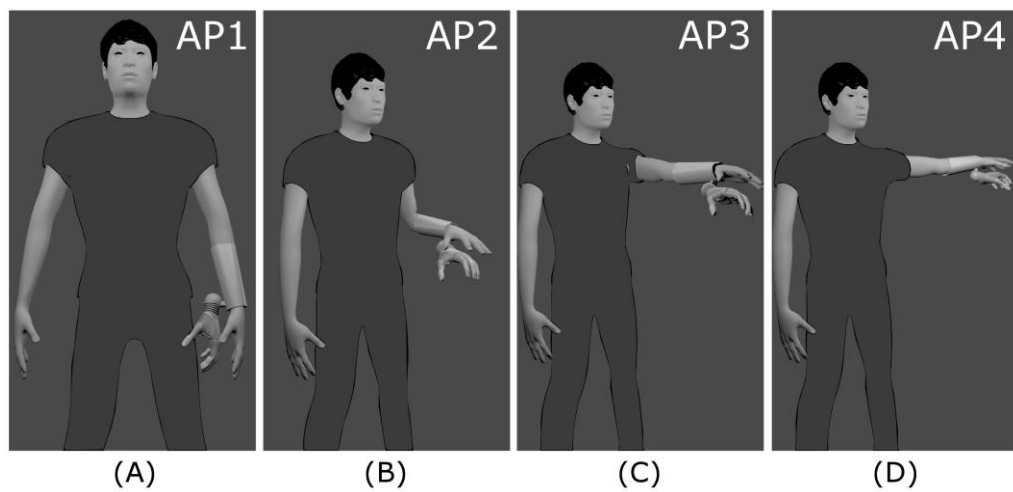


Figure 3.7: Arm Positions (AP). These arm positions were used during the experimental tasks. (A) AP1: forearm down, (B) AP2: elbow flexed, (C) AP3: arm frontally extended, (D) AP4: arm laterally extended.

After calibration and familiarization, all participants performed three functional tasks, namely, Arm and Wrist Positioning (AWP) task, Wrist Tracking (WT) task and Hand Shaking (HS) task (as outlined in Figure 3.6). In all three tasks, the participants were required to perform a predefined number of *rounds* using both control methods (MLC and cxMLC). A *round*, in each task, was defined as a time interval during which the participants performed as many *trials* as possible. To start a *trial*, the participants would pick up a wooden block from the left compartment of the box using the indicated grasp (palmar or lateral). The participants also adjusted the grasping force to be between 30–70% of the maximum force, using a visual force feedback provided on the computer screen. Once the block was appropriately grasped, the so-called Visual Task Instructor (VTI) was triggered by the experimenter (and seen on the computer screen). The VTI was used to indicate a sequence of actions, which had to be performed before returning the block back to the right

compartment of the box (end of trial). A trial was deemed successful if the block was transferred from the left to the right compartment, whereas if the block was dropped during VTI, the trial was considered failed. Lastly, note that the notion of round and trial was the same in all three functional tasks, i.e. each task comprised several rounds and in each round, the participants did multiple trials. However, the VTI and therefore the requirements in each functional task were different, as outlined below,

- i. **Arm and Wrist Positioning Task (AWP):** A single round of this task lasted 4 minutes and in each round, the participants did multiple trials. In a single trial, the VTI presented four different target arm postures (AP1-4), each followed by a target wrist orientation. First, a random arm position from the set AP1-4 was presented as an image on the computer screen. The arm position detector, then, waited for the participants to bring their arm in the required position. Next, the current and target wrist orientations were displayed as metronomes by the VTI as shown in Figure 3.8A. The participants were required to move the metronome indicating the current wrist orientation to match the metronome indicating the target orientation (and remain inside it for 250 ms). The target metronomes were randomly selected from a set of predefined target angles $\{-120^\circ, -60^\circ, 0^\circ, 60^\circ, 120^\circ\}$, and a tolerance of $\pm 15^\circ$ was allowed. Once the target wrist orientation was achieved, a new target arm position was presented to the participants. This process was repeated four times in a single trial i.e. four target arm positions were presented randomly, followed by a random (non-repeating) target wrist orientation. Lastly, all participants did four rounds of this task with each control method (MLC and cxMLC), wherein two rounds were performed by using the palmar grip to grasp the object and the other two were performed with the lateral grip; both grip types were administered alternately.
- ii. **Wrist Tracking Task (WT):** A single round of this task lasted 4 minutes and in total, two rounds (one with palmar and one with lateral) were performed for each control method. In a single trial, the VTI presented one target arm position (from the set AP1-4) which remained fixed during the trial. The task for the participants was to keep their arm fixed in the indicated position and control the wrist orientation, such that the metronome representing the current orientation would track the *moving target* metronome, as indicated in Figure 3.8B. The trajectory for the moving target metronome was generated by randomly selecting eight centre-angles from the set $\{-120^\circ, -90^\circ, -60^\circ, -30^\circ, 30^\circ, 60^\circ, 90^\circ, 120^\circ\}$ and then, the target metronome was

rotated between those centre-angles with a velocity randomly selected from the set $\{30^\circ/s, 45^\circ/s, 60^\circ/s\}$. The centre angles were chosen such that the sign between the consecutive angles would change. In this way, the participants were challenged to repeatedly pronate and supinate. Additionally, each time the target metronome reached one of the eight centre-angles, it stopped until the participants successfully positioned their prosthesis wrist within the target metronome ($\text{centre-angle} \pm 10^\circ$). Hence, a single trial of the WT task comprised of tracking a moving target on the screen while keeping the arm fixed in a given position.

- iii. Hand Shaking Task (HS): A single round of this task lasted 3 minutes and in total, two rounds (one with palmar and one with lateral) were performed for each control method. In a single trial, the VTI indicated a direction in which the participants had to shake their forearm until the *pie-chart* traversed a full circle, as shown in Figure 3.8C. The indicated direction of shaking was either left-and-right or up-and-down. The angular velocity ($|\omega|$) of the elbow joint was monitored and the participants were required to generate angular velocity $|\omega|$ in the range $[\omega_{min}, \omega_{max}]$ for 10 seconds; and, the pie-chart indicated the fraction of time for which $|\omega|$ was in the required range. The thresholds $[\omega_{min}, \omega_{max}]$ for up-and-down shake were $[2, 4.5] \text{ rad/s}$, whereas for left-&-right shake they were $[2.5, 5] \text{ rad/s}$; this difference was due to the greater difficulty when shaking against gravity.

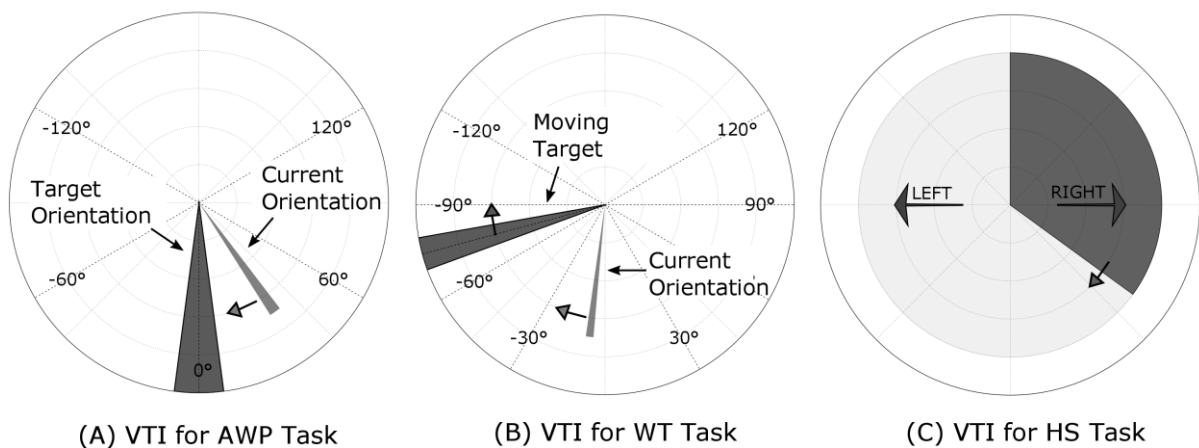


Figure 3.8: Visual Task Instructor (VTI). (A) VTI for AWP task: the participants were required to match the metronomes indicating the current and the target wrist orientations. (B) VTI for WT task: the participants had to track the moving target by using the metronome indicating the current wrist orientation. (C) VTI for HS task: the participants had to shake their forearm in the indicated direction (here, left-and-right) until the pie-chart traversed the full circle.

After the experiment, all participants were given a questionnaire in which they had to report their subjective experience(s) regarding the two control methods. In this questionnaire, the terms *Approach 1* and *Approach 2* were used to indicate either MLC or cxMLC - depending upon the sequence in which they were administered. The complete questionnaire has been provided as a supplementary material in Appendix I. This questionnaire was divided into three parts: (a) *Task Physical Demand*: where, the participants were asked to assess the physical demand required for completing each of the three tasks, independent of the control method, (b) *Control Cognitive Demand*: where, the participants were asked to assess the mental demand and frustration level associated with each control method, independently of the three tasks. (c) *Control Comparison*: where, the participants were asked to make a one-on-one comparison between the two control methods by reporting their subjective impression regarding: *performance of wrist rotation, performance of grasping, confidence of not losing a grasped object, difficulty in releasing a grasped object and delay experience before releasing a grasped object*.

The primary outcome measures used to compare MLC and cxMLC were the number of successful and failed *trials per round* observed during all three tasks. A further comparison between MLC and cxMLC was made by evaluating four offline secondary measures from the recorded data, which were as follows: a) *Number of Spurious Opening Commands*: calculated by counting the number of spurious opening pulses observed between the start and end of VTI in all successful trials of a given round for each task, b) *Number of Spurious Grasping Commands*: calculated by counting the number of spurious grasping pulses generated for the orthogonal grip type between the start and end of VTI in all successful trials of a given round for each task. In cxMLC, the commands for the orthogonal grip were suppressed in the Holding and Moving state and so, this value was always zero for cxMLC, c) *Absolute Change in Force*: the mean absolute difference between the normalized grasping force at the beginning and end of VTI for each successful trial in a given round for each task, d) *Unwanted Wrist Rotation*: the mean absolute difference between the wrist orientation at the beginning and end of VTI for each successful trial in a given round with the HS task. Optimally, all secondary measures should be equal to zero, as participants were not supposed to generate opening/closing commands or change the grip force/grip type during the tasks. Also, in the HS task, the participants were not supposed to rotate the hand.

For a given task and outcome measure, the Wilcoxon signed rank test was used to compare the performance of MLC versus cxMLC. In the questionnaire, the reported *Task Physical Demand* between the three tasks was compared using the Friedman test followed by a

post-hoc average rank test for pairwise comparison. All results are reported in terms of median and interquartile range (IQR) – denoted as median {IQR}. The required statistical analysis was performed using the software STATISTICA (from Dell, US) and the threshold for significance was set to $p = 0.05$.

3.4 Experimental Results

Figure 3.9 summarizes the results for the two primary outcome measures used to compare MLC and cxMLC. In the AWP task, the median number of successful transfers with cxMLC were significantly higher than MLC (5{4-6} vs. 4{2-5} with $p < 0.001$) and the median number of dropped objects with cxMLC were significantly lower than MLC (0{0-0} vs. 2{0-3} with $p < 0.001$). In the WT task, no significant difference was observed between the median number of successful transfers with MLC and cxMLC (3{2-5} vs. 4.5(4-5) with $p > 0.05$), but the median number of failed transfers with cxMLC were significantly lower than MLC (0{0-1} vs. 1{0-2} with $p < 0.05$). And, for the HS task, the median successful transfers with cxMLC were significantly higher than MLC (5{5-6} vs. 5{3-6} with $p < 0.05$) and the number of failed trails with cxMLC were significantly lower than MLC (0{0-0} vs. 0{0-2} with $p < 0.01$). In the HS task, the participants did not drop a single object with cxMLC.

Figure 3.10 and Figure 3.11 illustrate the benefit of using cxMLC in regards to the four secondary outcome measures. Figure 3.10A demonstrates the impact of *spurious opening commands*. Here, the participant issued four accidental opening commands while performing the AWP task with MLC, which led to an accidental decrease in the exerted grasping force. This could have been prevented if cxMLC was used for control, as all spurious opening commands were below the state-dependent threshold. Figure 3.10B demonstrates the impact of *spurious closing commands*. Here, the grasping force exerted on the object increased due to spurious closing commands issued for the orthogonal grip type (here, palmar). The object was grasped using lateral grip, but palmar commands were accidentally issued, causing the tightening of the grip. This could have been prevented if cxMLC was used instead of MLC, as it disables closing commands pertaining to the orthogonal grip type. Figure 3.11A shows the application of the “Restore Force” rule in the Holding state. It can be observed that, the grasping force was restored to its registered value whenever a drop was detected during the control with cxMLC. Figure 3.11B shows an *unwanted wrist rotation* observed during a trial of the HS task, wherein spurious wrist

rotation commands were issued towards the end of the trial. It can be speculated that all but two of these spurious rotation commands would have been blocked if cxMLC were used for control, as the spurious signals would have been under the state-dependent threshold for the Moving state.

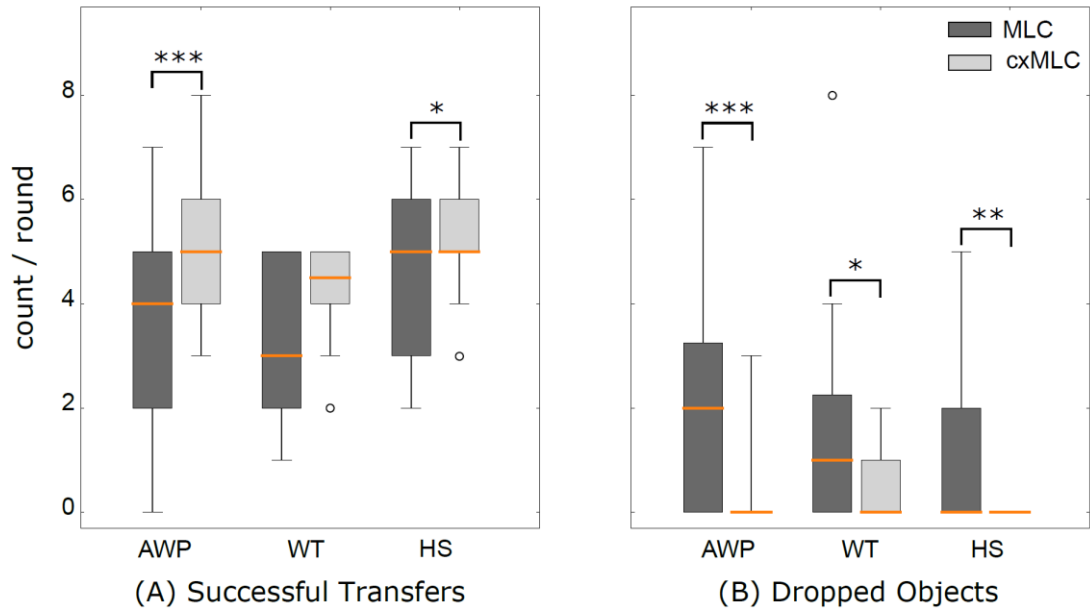


Figure 3.9: Summary of the results for the two primary outcome measures obtained with MLC and cxMLC: (A) Number of successful objects transferred per round in three tasks (AWP, WT, HS). (B) Number of objects dropped (while making a transfer) per Round in three tasks. (*, $p < 0.05$; ***, $p < 0.001$; the horizontal line within a box indicates median, the box indicates IQR and the circles indicate outliers).

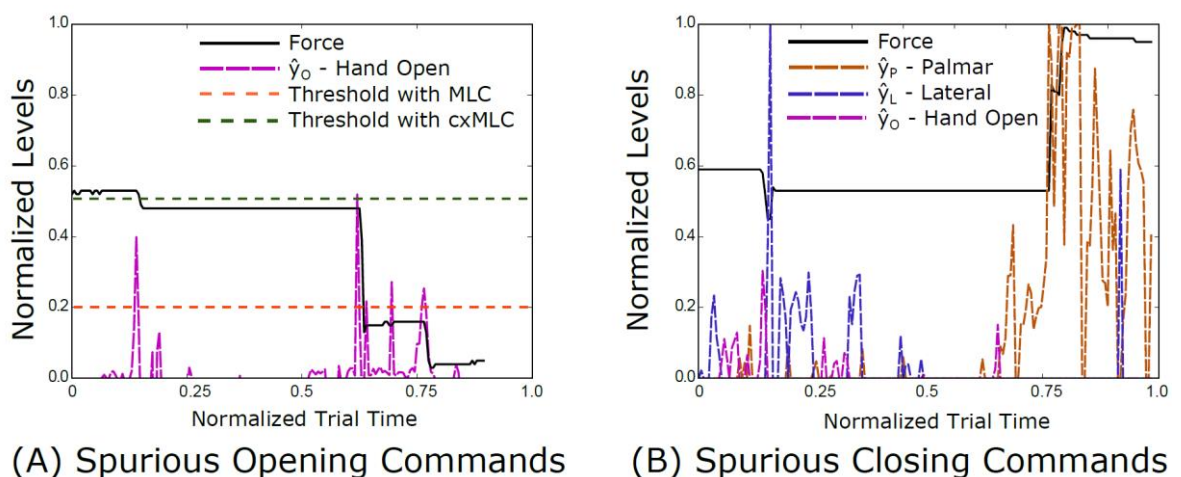


Figure 3.10: Representative signals recorded during the online experiment, demonstrating the possible advantage of cxMLC in relation to MLC. (A) Spurious Opening Commands observed during control with MLC and (B) Spurious Closing Commands observed during control with MLC.

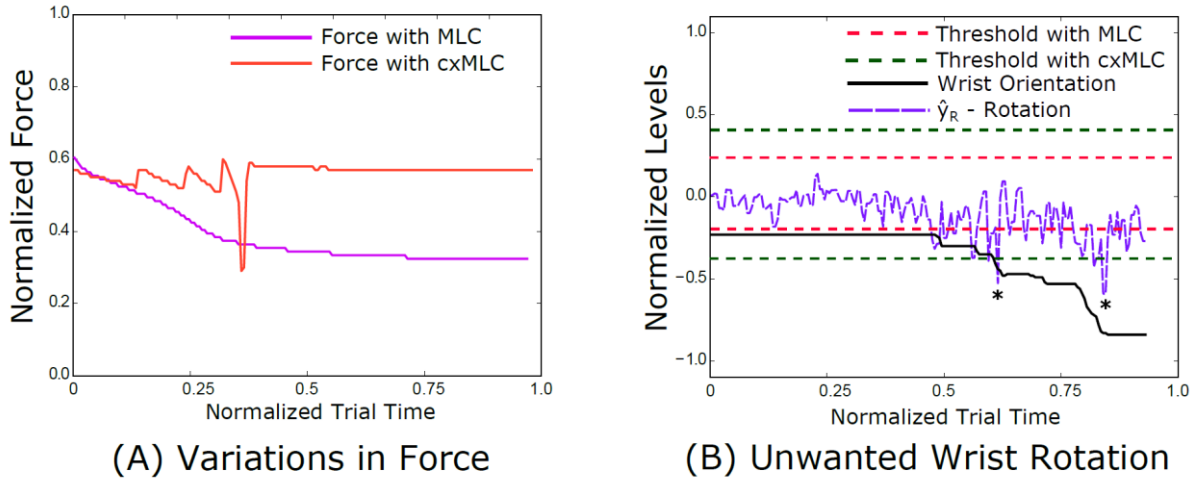


Figure 3.11: Representative signals recorded during the online experiment, demonstrating the advantage of cxMLC in terms of how (A) Variations in Force were restored during control with cxMLC, and how (B) an unwanted wrist rotation could have been prevented in the HS task by increasing the corresponding activation threshold as in the Moving state. (the asterisks indicate two spurious commands which would have crossed the state dependent threshold for the Moving state in cxMLC control.)

Figure 3.12 and Figure 3.13 summarizes the results for the four secondary outcome measures obtained using MLC and cxMLC. The number of spurious opening commands with cxMLC were significantly lower than MLC for the tasks AWP (0{0–0.25} vs. 2{0.75–4} with $p < 0.001$) and HS (0{0–0} vs. 1{0–2} with $p < 0.01$). The number of spurious closing commands with cxMLC were zero (as the orthogonal grip type was disabled), but the number of such commands with MLC were significantly higher than zero in all three tasks AWP (1{0–10}, $p < 0.001$), WT (0.5{0–4.75}, $p < 0.01$) and HS (1.5{0–3}, $p < 0.05$). Next, the absolute change in normalized grasping force with cxMLC was significantly lower than MLC for the tasks AWP (0.05{0.03–0.075} vs. 0.1{0.0375–0.175} with $p < 0.01$) and HS (0.04{0.02–0.05} vs. 0.085{0.03–0.13} with $p < 0.05$). And, the amount of unwanted wrist rotation in the HS task with cxMLC was significantly lower than MLC (15°{4°–32°} vs. 40°{7°–85°} with $p < 0.05$).

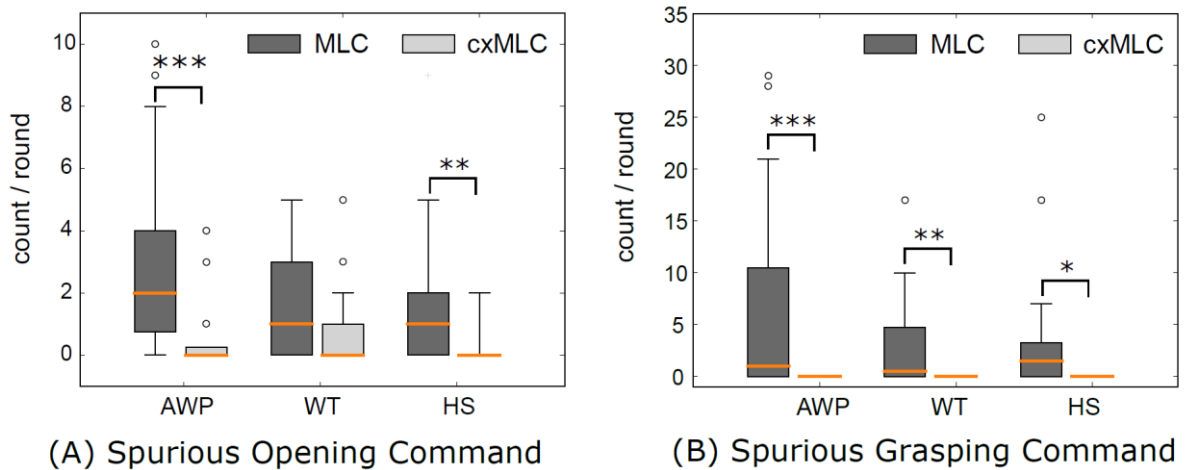


Figure 3.12: Summary of the results for the secondary outcome measures obtained from the offline analysis. (A) Number of spurious opening commands observed during successful trials in each round with MLC and cxMLC. (B) Number of spurious closing commands observed during successful trials in each round with MLC and cxMLC. (*, $p < 0.05$; **, $p < 0.01$; ***, $p < 0.001$; the horizontal line within a box indicates median, the box indicates IQR and the circles indicate outliers).

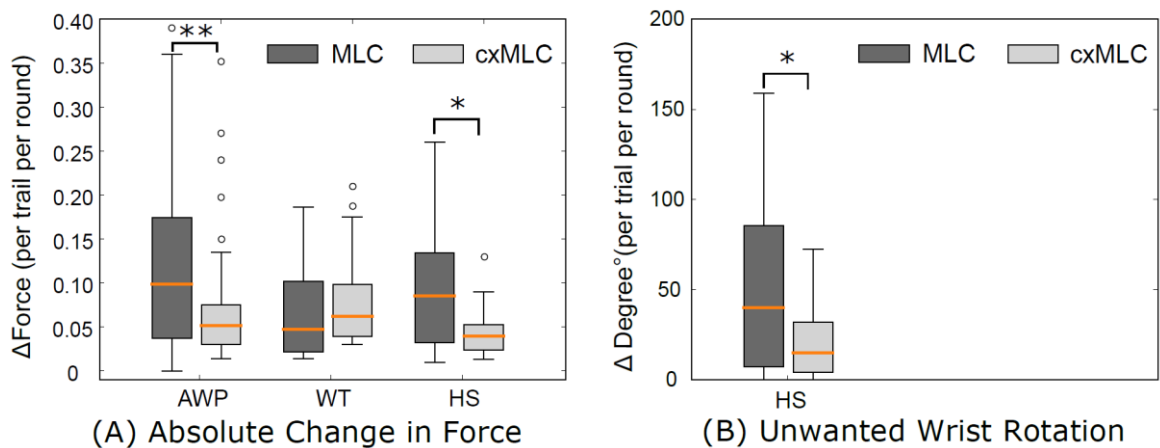


Figure 3.13: Summary of the results for the secondary outcome measures obtained from the offline analysis. (A) The mean change in normalized grasping force from beginning till the end of trial for each successful trial in a given round with MLC and cxMLC. (B) The mean change in wrist orientation from beginning till the end of trail for each successful trial in a given round with MLC and cxMLC (HS task only). (*, $p < 0.05$; **, $p < 0.01$; the horizontal line within a box indicates median, the box indicates IQR and the circles indicate outliers).

Figure 3.14 summarizes the scores given by the participants in the questionnaire. The physical demand for the task WT was significantly higher than tasks AWP and HS ($70\{60-78.75\}$ vs. $40\{35-65\}$ and $70\{60-78.75\}$ vs. $55\{35-55\}$, with $p < 0.05$). It could be speculated that, the constraint of fixing the arm in one position (especially AP3 and AP4) during some trials led the participants to experience a higher physical demand for the WT

task. The reported frustration with MLC was significantly higher than with cxMLC (57.5{41–65} vs. 37.5{31–52.5} with $p < 0.05$), whereas the mental demand required with both control methods was similar (55{31.5–70} vs. 37.5{26.5–52.5} with $p > 0.05$). Next, Figure 3.14C makes a direct point-by-point comparison between the two control methods MLC and cxMLC. The participants reported being more confident of not dropping the grasped object with cxMLC, as indicated by the reported median score of 40{0–50} which was significantly higher than zero (with $p < 0.05$). The reported scores for other comparison questions were not significantly different from zero.

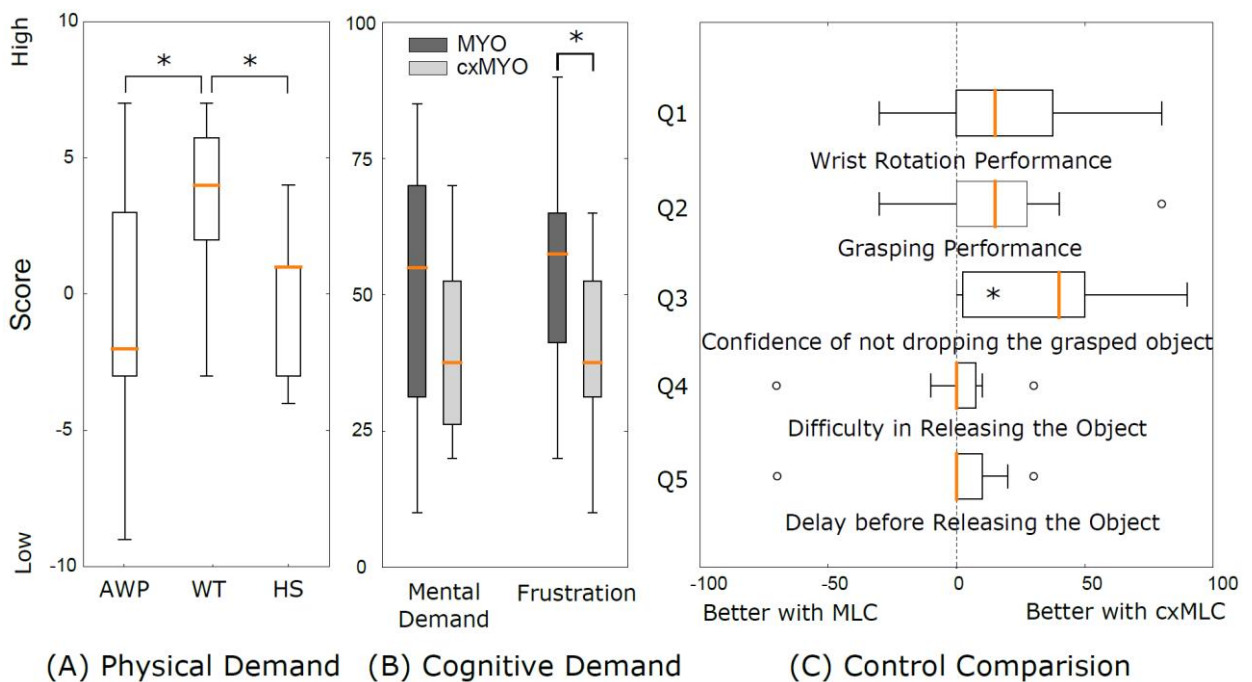


Figure 3.14: Summary of the results for the questionnaire. (A) Task Physical Demand reported for the three tasks AWP, WT and HS. (B) Control Cognitive Demand reported in terms of Mental Demand and Frustration for control with MLC and cxMLC. (C) Control Comparison between MLC and cxMLC. (*, $p < 0.05$; the horizontal line within a box indicates median, the box indicates IQR and the circles indicate outliers)

3.5 Discussion

This chapter presented a new method called context-driven ML control (cxMLC) that provided a simple way to improve the robustness of ML-based control (MLC) by integrating context information available from additional prosthesis sensors. Specifically, the information available for additional sensors was used to determine the prosthesis state (context), and then the robustness of MLC was improved by applying a set of simple post-processing rules to mitigate expected disturbances. In an online experiment, the new

cxMLC method was extensively compared with the conventional method (MLC) using three functional tasks. The experimental results indicated that, cxMLC was able to decrease the number of unwanted opening, closing and rotation commands, and thereby improve performance (success rate and/or dropped objects) in all three functional tasks. Thus, cxMLC offers not only a simple way to integrate context information coming from additional prosthesis sensors, but it also provides functional improvement during online use.

The experimental protocol used to compare the performance of MLC and cxMLC was carefully designed to challenge the robustness of both methods. The functional evaluation integrated several factors that are known to cause problems during the online use of a myoelectric prosthesis, for example: (1) changing the position of arm during use (examined in AWP task), (2) generation of myoelectric commands in rapid succession (examined in WT task), and (3) performing rapid arm movements (examined in HS task). The first problem (of changing arm position during use) has been investigated in several recent studies [73], [79], [80], where it has been shown that repositioning the arm could (slightly) change the EMG patterns and thereby influence the overall performance. Similarly, rapid generation of myoelectric commands is characterized by many transient muscle activations, which could lead to a decrease in the overall performance [106]. Lastly, fast arm movements, such as shaking, could also lead to unintended muscle activations and thereby spurious prosthesis responses. Previously, a test similar to the shaking test was used to demonstrate the robustness of osseointegration [63] against motion artefacts (as shown in the video [107]).

In the past, multimodal sensor fusion has been used to either improve the performance of a ML-based controller [72], [73], [75], [79], [80] or achieve semi-automatic control [82]–[86]. The presented method (cxMLC) advances the state-of-the-art in the former direction, as it utilizes multimodal sensor data to improve the robustness of a ML-based controller. In [72], [73], [75], [79], [80], the context information was derived only from proprioceptive sensors (e.g. IMU), but the new method (cxMLC) utilized information from both proprioceptive and exteroceptive sensors (i.e. force and aperture data along with IMU). Furthermore, the context information in [73], [79], [80] was used to mitigate the effect of a single factor (i.e. arm position), whereas cxMLC compensated for different factors (disturbances) pertaining to different prosthesis states. For example, states Closing and Grasping stabilized the wrist during grip-closure, Holding prevented accidental opening of the hand, and Moving filtered out unwanted activations arising during motion. Next, cxMLC is also different from semi-automatic methods for sensor fusion, where the context

information is processed in parallel with sEMG data to achieve partial automatic control (see Figure 1.3C). Contrary to this, in cxMLC, the context information was processed in the background and the online control always remained manual, i.e. each prosthetic function was activated only in response to EMG command(s) generated by the user.

In the current implementation of cxMLC, the context information was used to influence only the extrinsic parameters (mostly, activation thresholds) of the ML controller. However, the presented concept is general and can be used to influence other extrinsic as well as intrinsic parameters of the ML controller. One possibility is to modulate the gain (an extrinsic parameter) of each DoF depending on the state of the prosthesis. This could be applied to provide a better force control in the Holding state, thereby allowing the user(s) to accurately change the applied force on the object. For example, in the Michelangelo hand prosthesis (used for the experiment), an attempt to decrease the applied force on a rigid object could sometimes result into an accidental dropping of the object, as the hand loses contact with the object when opening commands are given to reduce force. This issue could be addressed by lowering the gain of hand opening commands in the Holding state. Likewise, the control could also be improved by modulating internal parameters such as class priors associated with different movements depending on the prosthesis state. Next, the state-transitions in the current implementation of CAC were triggered using simple IF-THEN rules applied on the observed sensor data. In the future, state-transitions could be triggered by using more advanced methods such as hidden Markov models [108]. Lastly, the presented CAC could also be applied to enhance the performance of the classic two-channel controller. For example, the threshold for hand opening could be increased in the Holding state to prevent unintended opening of the prosthesis during co-contractions (which are needed to switch between available prosthetic DoFs, see Section 1.2).

The current chapter demonstrated that modulating activation thresholds to suppress/eliminate spurious commands can increase the robustness of myoelectric control based on ML. This concept (of suppressing spurious commands) could be improved further by simply utilizing a better criterion (or metric) to suppress/eliminate spurious commands. Currently, the CAC employs activation thresholds as the criterion for eliminating spurious commands, but this has one downside. Specifically, it simply eliminates all estimated activations (coming from MLC) below the state-dependent threshold, but it does not assess the quality of EMG patterns responsible for the observed activation. For example, the activation threshold for hand opening in the Holding state was 0.5 and therefore, all hand

opening commands with an estimated strength lower than 0.5 were always suppressed (without assessing the quality of input EMG patterns). But, before suppressing an estimated command, it could be beneficial to examine whether the command was generated intentionally by the user or caused by unwanted muscle activation. One way to examine this is to compare the input EMG signals with the EMG data recorded during training. The required comparison could be made by leveraging one of the many well-known distance metrics such as Euclidean distance, Mahalanobis distance, cosine similarity, etc., but it would be beneficial to employ a metric which reflects upon the latent physiological principles behind generation of EMG patterns. The next chapter proposes to use cosine similarity as a metric for assessing the quality of EMG patterns. The chapter first describes how the cosine similarity metric reflects the underlying physiological principle of muscle coordination [53], [54] and then investigates how well this metric can be applied for online myoelectric control.

Context- and Physiology-aware Machine Learning for
Upper-Limb Myocontrol

4 A PHYSIOLOGY-INSPIRED METRIC FOR CLASSIFYING EMG

This chapter has been based on the recently submitted paper:

G. K. Patel, C. Castellini, J. M. Hahne, D. Farina, S. Dosen, "A Novel Classification Method for Myoelectric Control of Hand Prostheses Inspired by Muscle Coordination", submitted to IEEE Transactions on Neural Systems and Rehabilitation Engineering.

4.1 Motivation

With most ML methods available today, the modelling of EMG data is done by following the conventional pattern recognition paradigm, where a training dataset is collected using a supervised procedure, and then a mathematical function is fitted over the collected data. One shortcoming of this conventional paradigm is that, the modelling of EMG data is done without considering the underlying physiological phenomenon responsible for the generation of EMG patterns. Instead, the conventional procedure relies solely on the mathematical function to implicitly capture the underlying regularities in the data. This generalization may not be optimal, as many commonly applied ML methods (e.g. LDA [28]) may not model the underlying physiologically relevant principles [47]. This chapter discusses a method that models EMG data based on the knowledge of physiology that forearm muscles act consistently in a coordinated manner.

The neural mechanisms underlying the coordination of forearm muscles for single- and multi-digit force production tasks were studied by Valero-Cuevas [53] and Poston et al. [54]. They demonstrated that force production relies on the coordination of different

forearm muscles and that the EMG amplitude of active muscles scales uniformly as a function of applied force during muscle contraction. Thus, for a given movement, the same set of muscles is consistently recruited and their activation level is proportional to the amount of force exerted. As mentioned earlier, this physiological property is called the *principle of muscle coordination*. This principle was previously exploited by He et al. [55] and Al-Timemy et al. [56] to compute EMG features invariant against muscle contraction strength. The computed invariant feature set was, then, presented as an input (for learning and prediction) to traditional classifiers (e.g. LDA). But instead of defining a new feature set, as in [55] and [56], the aim of this chapter is to investigate how well the principle of muscle coordination can be applied to model the standard amplitude-related EMG features (i.e. RMS or MAV features) for online myoelectric control. Specifically, the chapter proposes to use *cosine similarity* as a metric to model the amplitude-related EMG features. Previously, this metric was used only for analysing the property of muscle coordination [53]–[55], but it was never applied for online myoelectric control. This chapter investigates if and how well the cosine similarity metric can be applied for online myoelectric control, and also discusses why this metric is a better measure for assessing the quality of EMG signals as compared to other metrics such as Mahalanobis distance or Bayesian confidence.

4.2 Classification Method inspired by Muscle Coordination

The amplitude-related EMG activity recorded from d sensors placed on the forearm muscles can be represented by a d -dimensional feature vector $\vec{x}_d = (a_1, a_2, \dots, a_d)$, where a_i is either the root mean square (RMS) or mean absolute value (MAV) of the i^{th} -channel computed over a time window. From here on, any feature vector \vec{x}_d containing only amplitude-related EMG features will be referred to as an amplitude feature vector (AFV) for simplicity. Next, the property of muscle coordination suggests that, for a given movement of forearm muscles, the amplitude of EMG signals acquired from active muscles scales uniformly as a function of contraction strength ([53], [54]). Thus, all AFVs associated with a give movement can be interpreted as vectors pointing in the same direction irrespective of the contraction intensity. Therefore, an entire population of AFVs associated with each movement can be modelled using a single prototype AFV, and the classification of movement can be achieved by measuring the orientation with respect to the registered prototype vector.

In the current implementation, the prototype vector for a given movement was determined during supervised training by averaging the AFVs recorded while performing the maximum voluntary contractions (MVC) of the movement for 3 seconds. In the classification step, the cosine similarity metric was used to compare an input vector \vec{x}_d with the average MVC recorded for each movement class, and the movement with the maximum cosine similarity was given as the classification output:

$$\arg \max_m \left(\frac{\vec{x}_d \cdot \vec{P}_{d,m}}{|\vec{x}_d| \cdot |\vec{P}_{d,m}|}; m \in S_{movements} \right) \quad (4.1)$$

where, \vec{x}_d is an input AFV generated by the user and $\vec{P}_{d,m}$ is the prototype vector for the m^{th} movement representing the average of AFVs recorded while performing the MVC of the m^{th} movement for 3 seconds.

Next, our knowledge of muscle coordination does not provide an understanding for the minuscule EMG activity observed during *rest* (i.e. no muscle contraction). This EMG activity (associated with the hand at rest) is characterized by minuscule noisy activations around some mean value [109]. Thus, the rest class can be modelled as a Gaussian distribution with a fixed mean and covariance. In the classification step, the distance between an input vector \vec{x}_d and the Gaussian distribution was measured by calculating the Mahanobis distance $d_{MH}(\vec{x}_d)$ as follows:

$$d_{MH}(\vec{x}_d) = \sqrt{(\vec{x} - \vec{U})^T C^{-1} (\vec{x} - \vec{U})} \quad (4.2)$$

where, U is the mean and C is the covariance of the rest class. And, the input vector \vec{x}_d was classified as rest, if the measured distance $d_{MH}(\vec{x}_d)$ was less than or equal to a pre-defined threshold T , else the vector \vec{x}_d was classified according to Equation 4.1, i.e. when $d_{MH}(\vec{x}_d) > T$. The threshold T was determined by measuring the Mahanobis distance d_{MH} to the MVC of the nearest prototype, and then setting its value to 30% of $d_{MH}(\vec{P}_{nearest})$. Hereafter, this classification scheme based on cosine similarity will be abbreviated as CoS.

4.2.1 Comparing CoS with a data-driven approach

Figure 4.1 illustrates the difference between a conventional data-driven approach (such as LDA) and the proposed muscle coordination inspired approach (CoS). For didactic purposes, the figure shows modelling of EMG data in a 2D space of projected features, whereas the full dimensionality was used for online control (see experimental procedure in Section 4.3). A data-driven approach (here, LDA) will minimize the model fitting error

without considering the physiological relevance of the training points. For example, with LDA in Figure 4.1C, the MVC of the ulnar-deviation movement was actually assigned to the extension class. Moreover, the area of the feature space assigned to the ulnar-deviation class is closed (a triangle) and rather small compared to other classes. With increasing contraction intensity, an ulnar-deviation vector moves away from the origin and crosses into the extension class. Contrary to this, the CoS classifier generates a physiologically meaningful model for the EMG data, where each movement class is modelled using an open cone radiating away from the origin and the rest class is modelled using an ellipse centred at the mean of the AFVs representing hand at rest.

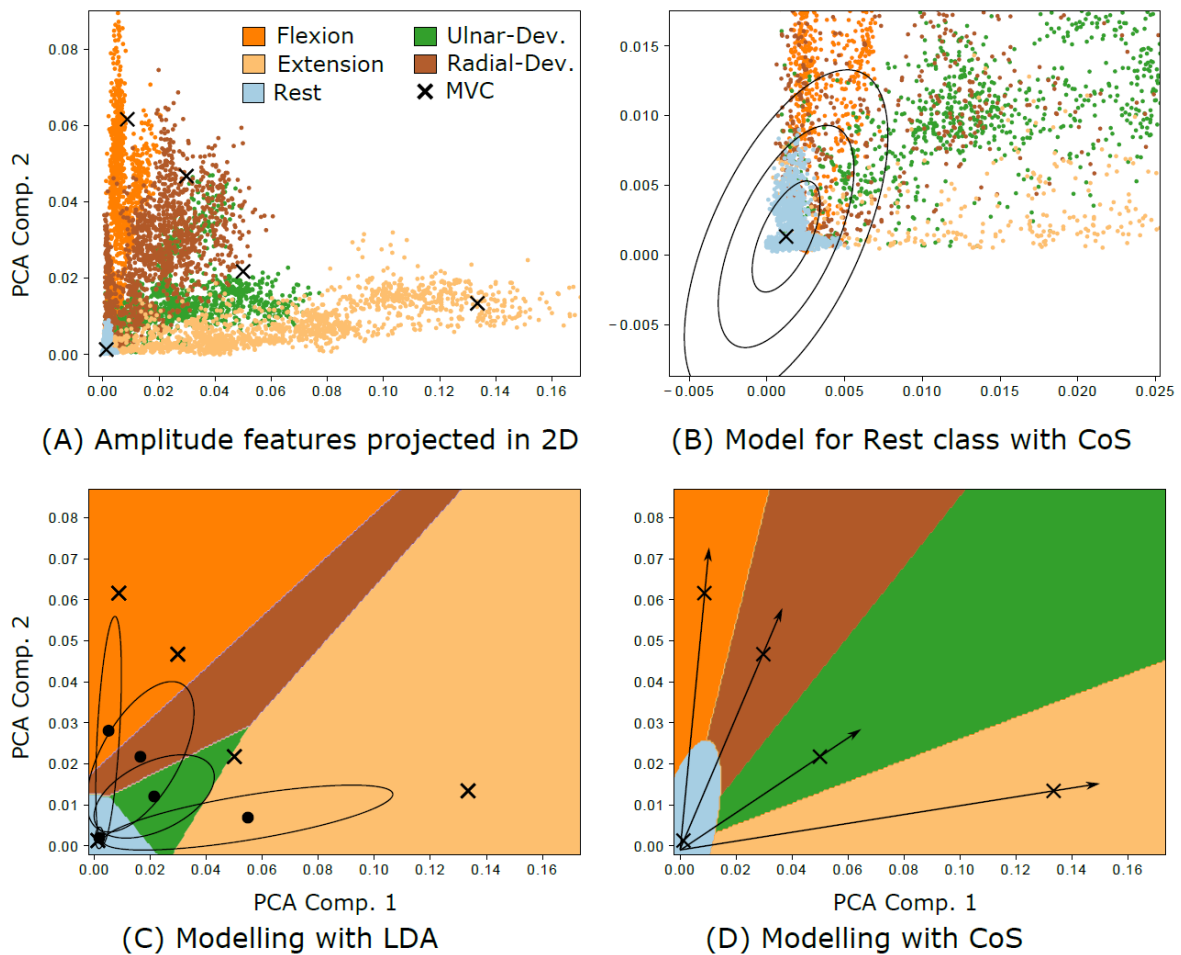


Figure 4.1: (A) 2D Projection of amplitude-related EMG features generated by an able-bodied participant performing four different movements, namely, wrist- flexion and extension, ulnar- and radial- deviation plus rest. (B) The data for the rest class modelled as a Gaussian distribution. (C) and (D) Visualization of classification boundaries obtained after applying the LDA and CoS model on the transformed data, respectively. Principal Component Analysis (PCA) was used for dimensionality reduction and the explained variance was 81%. The plotted data has been collected according to the procedure explained in Section 4.3.

4.2.2 Cosine similarity: a metric for assessing the quality of EMG patterns

A good modelling metric not only provides a way to classify EMG patterns for online control, but it can also be used as a confidence measure to assess the quality of EMG patterns presented as input to the classifier. A good confidence measure can be used to implement a post-processing scheme to reject possibly erroneous decisions made by the classifier, and the prosthesis could be forced to take no action when the measured confidence is low. Figure 4.2 compares the use of cosine similarity versus Mahanobis distance for modelling the distribution of EMG data. The Mahanobis distance models data using a Gaussian distribution and therefore, it represents the case of many widely used classification methods (such as LDA) which assume an underlying Gaussian distribution before fitting a model on the data. In Figure 4.2, the points R_M and R_I represent the radial-deviation class with different contraction intensity and points U_M and U_I represent the ulnar-deviation class. According to the property of muscle coordination, points R_M and R_I are expected to have the same likelihood (or confidence) of belonging to the radial-deviation class, and point U_I is expected to have a lower likelihood of association to the radial-deviation class. But, when a Mahanobis distance metric is used to model the data, point R_M is assigned a lower likelihood of belonging to the radial-deviation class as compared to point R_I , and points R_M and U_I have the same likelihood of belonging to the radial-deviation class. On the contrary, the cosine similarity metric assigns the same level of confidence to points R_M and R_I , and a lower confidence to point U_I for the radial-deviation class. Thus, the cosine similarity metric can be seen as a physiologically meaningful measure of confidence.

For regression-based control, Bayesian statistics can be applied to determine a predictive distribution \mathcal{N} around an estimate $\hat{y} \sim \mathcal{N}(\hat{x} \cdot W, c[1 + \hat{x}^T(X^T X + \lambda I)^{-1}\hat{x}])$, where $\hat{x} \in R^d$ is an input feature vector generated by the user during online control, the term $\hat{x} \cdot W$ represents the mean of the predictive distribution (same as Equation 1.1), the term $c[1 + \hat{x}^T(X^T X + \lambda I)^{-1}\hat{x}]$ represents the confidence interval (or spread of the distribution) and c, λ are constants that depend on the variance in the dataset $\mathcal{D} \cong (X, Y)$. In theory, the confidence interval (determined by $[1 + \hat{x}^T(X^T X + \lambda I)^{-1}\hat{x}]$) can be used to determine the credibility of the predicted simultaneous and proportional movement $\hat{y} \in R^M$. But, as illustrated in Figure 4.3, this confidence measure is highly dependent on the distribution density of the training data points, i.e. confidence is high in regions where the density of training points is high and decreases as the density of training points decreases. Thus, the

Bayesian approach is data-driven and therefore, does not provide a physiologically meaningful measure of confidence.

Currently, there is a lack of confidence measures which can be used to assess the quality of EMG during simultaneous and proportional control, but the same physiological principle behind cosine similarity could be extended to design a confidence measure for such a control. In literature, it has been shown that EMG patterns pertaining to simultaneous activation of multiple DoFs are simple linear combinations of single DoF movements [110] (for example, EMG signals pertaining to the simultaneous extension and ulnar-deviation of the wrist is a linear combination of EMG signals pertaining to just wrist extension and just ulnar-deviation). From this knowledge, it can be speculated that each simultaneous activation could be represented using a new prototype vector obtained via the linear combination of prototype vectors representing single movements, and the cosine similarity with respect to the new prototype vector can be used as a confidence measure for simultaneous activation. The validation of this hypothesis is beyond the scope of this thesis, but it gives us an insight into how the principle behind cosine similarity could be applied in different application scenarios.

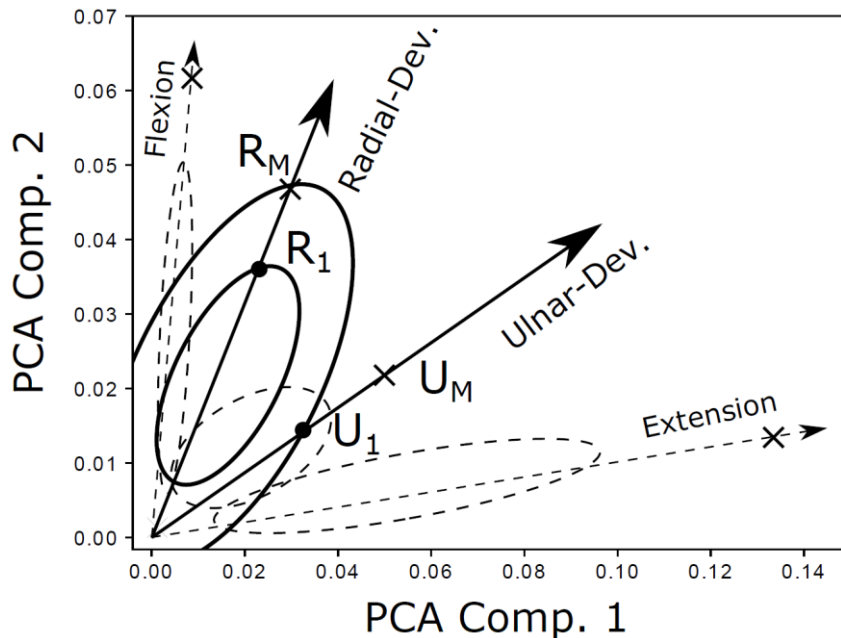


Figure 4.2: Cosine similarity versus Mahalanobis distance based modelling of EMG data. A Mahalanobis distance models data using a Gaussian distribution (represented by an ellipse), whereas the cosine similarity models each class using a line connecting the origin of the feature space to the MVC of the class. The models have been generated using the data shown in Figure 4.1A.

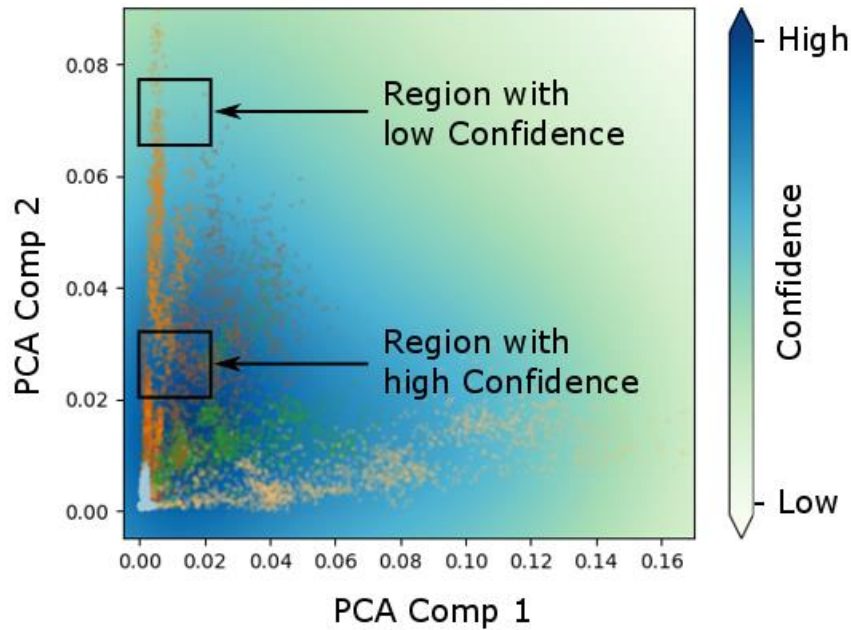


Figure 4.3: Confidence evaluated by applying Bayesian statistics on the transformed training data shown in Figure 4.1A. The value of estimated confidence depends on the distribution density of the training points, i.e. the estimated confidence is high in regions with high density of training points, and it decreases as the density of training point decreases. The presented Bayesian confidence was calculated after applying the Random Fourier Feature kernel [67] on the transformed training data.

This section emphasized the theoretical benefits of using cosine similarity as a metric for modelling EMG data. In practice, an experimental evaluation is required to measure the practical benefits of our theoretical understanding. Therefore, an online experiment was conducted to examine the practical benefits of using the CoS classifier for myoelectric control. During this experiment, the CoS classifier was extensively compared with the academic state-of-the-art LDA classifier [26], [27]. The experimental procedure used to compare CoS and LDA has been described in the next section.

4.3 Materials and Methods

An experiment involving one amputee and eleven able-bodied participants was conducted to compare the performance of the CoS classifier with the LDA classifier. This experiment was conducted according to the declaration of Helsinki [102]. The experimental setup worn by the participants is shown in Figure 4.4. All able-bodied participants were fitted with a bypass socket attached to a right-sided Michelangelo prosthesis (from Ottobock, DE) and the amputee was fitted with a custom-made socket attached to a left-sided Michelangelo prosthesis. The sEMG signals were acquired using eight commercially available double differential EMG electrodes (13E200 AC from Ottobock, DE) placed circumferentially and

equidistantly around the forearm. For able-bodied participants, the electrodes were strapped using an adjustable Velcro armband (Figure 4.4A), whereas the electrodes were integrated within the custom-made socket for the amputee (Figure 4.4B). The captured EMG signals were pre-amplified and band-pass filtered by the electrodes and then sampled at 1 kHz using a wireless data acquisition card (AXON Master 10-bit A/D converter from Ottobock, DE). The CoS and LDA based classification schemes were implemented as a software program on a standard PC equipped with two Bluetooth dongles, one to acquire sEMG data from the electrodes and the second to control the prosthesis. Both classifiers were used for the sequential and proportional control of four prosthesis functions, namely, hand closing (palmar grip), hand opening, wrist pronation and supination. These prosthesis functions were controlled by sEMG signals obtained from four contraction patterns, namely, wrist flexion, wrist extension, radial- and ulnar-deviation, respectively.

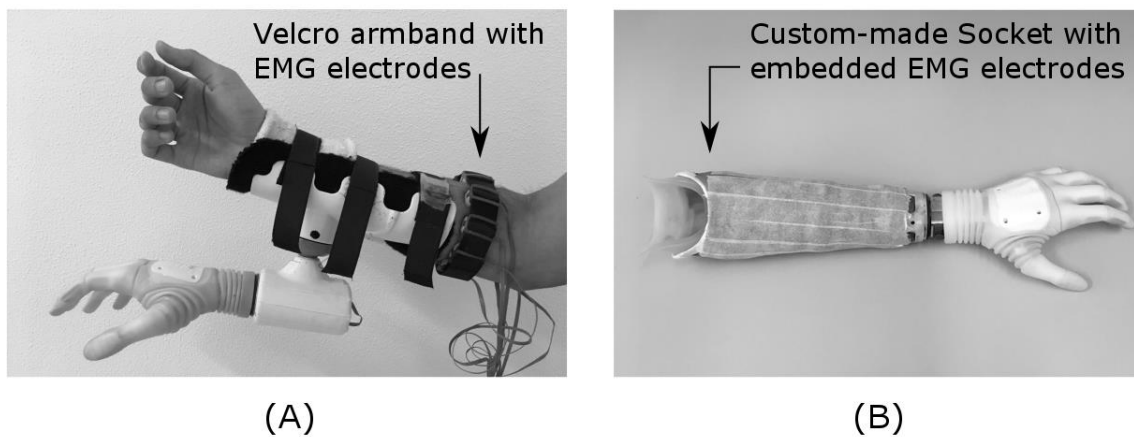


Figure 4.4: Experimental setup for (A) able-bodied participants and (B) the amputee participant.

The LDA classifier was implemented according to the standards recommended by [26], [27], whereas the CoS classifier was implemented as discussed in Section 4.2. The Hudgins time domain features (i.e. MAV plus ZC, SCC, WFL) was extracted by segmenting the raw sEMG signal using time intervals of 128 ms with an overlap of 32 ms per frame [27]. The training data for the CoS classifier was collected by asking the participants to produce the MVC of five classes (four movements plus rest) for 3 seconds. Next, the training data for the LDA classifier was collected by asking the participants to track trapezoidal trajectories, with plateaus normalized to 30%, 60% and 90% of average MVC, using a cursor indicating the normalized sum of amplitude values across all electrodes. Recording data at different contraction levels and in different arm postures is recommended for LDA [26], [27].

Therefore, the training data for LDA was recorded at three different contraction levels (30%, 60%, 90%) and in three different arm positions, namely, elbow bent in front of the torso, hanging arm and arm stretched forward in the sagittal plane at shoulder level. In total, 45 movement trajectories (5 patterns \times 3 levels \times 3 postures) were presented during the training for LDA, where the duration of each trajectory was 5 s (1 s rise, 3 s hold, 1 s fall time) followed by a 2 s rest interval between trajectories. Thus, the recording of the training data took approximately 7 minutes for the LDA and a mere 30 seconds for CoS (as only MVC data was required to train CoS). Moreover, the LDA classifier was operated using the full Hudgins time domain feature set (as recommended in [26], [27]), whereas the CoS was operated using only the MAV features. In the post-processing step, a majority vote filtering of length seven was applied to the classification stream of both classifiers. And, the strength of the classified movement was determined by removing the mean baseline EMG activity and then normalizing the sum of input amplitude values by the sum of MVC amplitudes of the detected movement class. Thereafter, the velocity of the corresponding motor was determined by applying a fixed threshold of 0.2 and gain of 1.2 to the estimated strength of the detected movement.

The comparison between CoS and LDA was made using four functional tasks with varying level of difficulty. A description of each task is as follows,

- i. *Box-and-Blocks Test*: The test setup comprises a box divided in two compartments, one empty and the other filled with blocks (see Figure 1.4A). In a single round, the participants transferred as many blocks as possible, from the filled to the empty compartment, within 60 seconds. The outcome measure was the number of blocks transferred in one minute [96].
- ii. *Clothespin Test*: In this test, a Rolyan Graded Pinch Exerciser with three red pins was used as the test setup (see Figure 1.4B). In a single round, the participants were asked to pick up three pins from the horizontal bar, rotate them and place them on the vertical bar. The outcome measure was the time required to successfully transfer three pins.
- iii. *Bottle Transfer Task*: The purpose of this (and the following) task was to evaluate the robustness of control when dealing with (i.e. transferring/manipulating) heavy objects. Two parallel lines were marked 120 cm away on a 2 meter wide table, and three water-bottles (diameter 6 cm and height 25 cm) filled with one litre water (approx. 1 kg at room temperature) were placed on the side where the prosthesis was connected (i.e. on the right-side for the able-bodied participants and left-side for the

amputee). The floor in front of the table was marked with a square 60 cm in side length, and the participants were instructed not to step out of the square marker during the task. In a single round, the participants had to pick up three bottles, one at a time, and transfer them to the other side of the table. The order in which the bottles were picked up did not matter. Next, the participants had to transfer the same three bottles back to their initial position, thereby completing a single round. If a bottle was dropped during lift / transfer / placement, it had to be placed back to its initial position and then retransferred. The two outcome measures recorded in each round of this task were the *task completion time* and the *number of bottles dropped*.

- iv. *Bottle Turn Task*: The setup from the previous task was adapted in this task. In a single round, the participants had to pick up a bottle, turn it by about 90° to horizontal orientation and lay it down on the other side. This was done one-by-one for all three bottles, with no constraint of a fixed order. Next, the bottles were picked up, turned to vertical orientation and placed back to their initial position, thereby completing a single round. Similar to the previous task, the *task completion time* and the *number of bottles dropped* were recorded as outcome measures.

An outline of the experimental procedure is shown in Figure 4.5. Before performing the aforementioned functional tasks, the participants were introduced to the concept of myoelectric control and the training data required for classification was collected as described previously. All participants were informed that the experiment was tailored to compare two different control approaches, but in order to prevent expectation bias, specific details regarding both approaches (e.g. exact name of each approach, or how one approach was different from other, etc.) were not disclosed. Thereafter, the participants were explained how each of the four tasks had to be completed, and in order to familiarize them with the tasks, they were asked to practice each task at least once. The experimental evaluation was divided into two blocks (as seen in Figure 4.5) and in each block, a different control method (either LDA or CoS) was administered randomly across participants; i.e. half used LDA followed by CoS and vice versa. In a given block, the participants sequentially performed four rounds of the Box-and-Blocks, Clothespin, Bottle Transfer and Bottle Turn task, i.e. they performed four rounds of Box-and-Blocks task followed by four rounds of Clothespin test and so on as indicated in Figure 4.5. At the end of the experiment, the participants had to report their subjective experience regarding the two control methods, wherein each method was subjectively scored with a number between 0 and 10; with 10 representing the best control over all prosthesis movements.

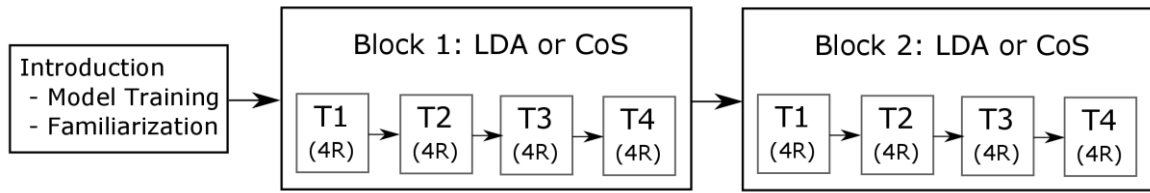


Figure 4.5: The experimental protocol used to compare LDA and CoS classifier. The experiment was divided into two blocks, where either LDA or CoS was administered randomly across participants. In each block, four rounds (4R) of four functional tasks were performed. Abbreviations are, T1: Box and Blocks, T2: Clothespin, T3: Bottle Transfer, T4: Bottle Turn and 4R: four rounds.

For a given outcome measure, the Wilcoxon signed rank test was used to compare the performance of CoS versus LDA. All results are reported in terms of median and interquartile range (IQR) – denoted as median {IQR}. The required statistical analysis was performed using the software STATISTICA (from Dell, US) and the threshold for significance was set to $p = 0.05$.

4.4 Experimental Results

Figure 4.6 summarizes the results for the online assessment performed with the able-bodied participants. For the Box-and-Blocks test, the participants transferred 12.0{9.75-14.25} blocks in one minute using CoS, which was slightly but significantly higher than 12.0{8.75-14.0} blocks with LDA ($p < 0.05$). The time taken to transfer three pins was similar for both LDA and CoS (26.4{20.4-40.0} s vs. 26.6{18.6-35.8} s with $p > 0.05$). And lastly, the task completion time with CoS was significantly lower than LDA for both Bottle Transfer (35.6{31.2-45.0} s vs. 42.2{32.3-62.3} s with $p < 0.05$) and Bottle Turn task (53.3{42.3-66.1} s vs. 70.4{54.8-85.5} s with $p < 0.05$).

Figure 4.7 summarizes the results for the online assessment performed with the amputee participant. Here, the performance of CoS was consistently better than LDA in all tasks. The amputee transferred a median of 24.5{24.0-25.3} blocks per minute with CoS versus a median of 17.5{16.8-18.3} blocks per minute with LDA. And, the median task completion time for the Clothespin, Bottle Transfer and Bottle Turn tasks were 11.6{11.1-12.0} s, 20.4{19.2-22.0} s, 30.0{29.7-33.2} s with CoS versus 15.5{15.2-16.3} s, 36.8{34.7-39.8}s, 44.8{44.4-45.5} s with LDA, respectively.

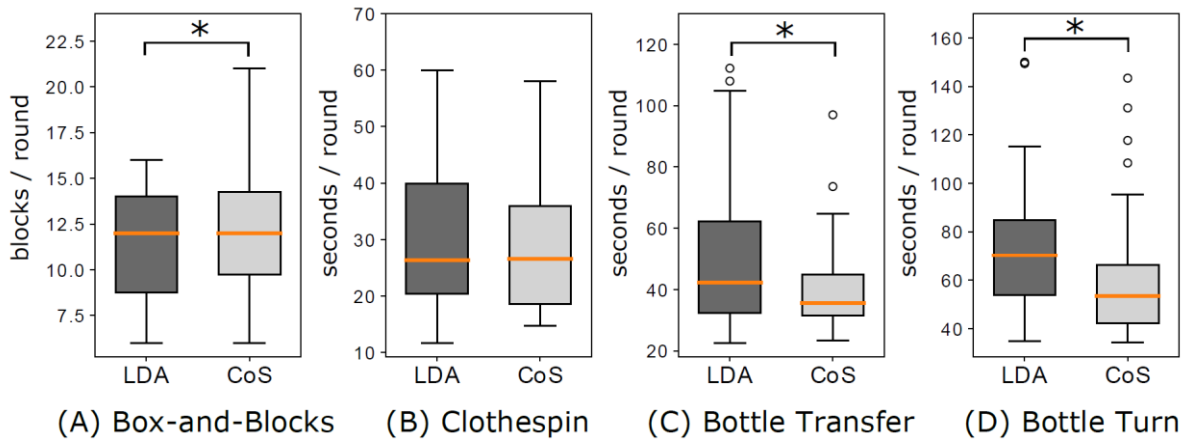


Figure 4.6: Summary of the results for the able-bodied participants. (A) The Box-and-Blocks test, (B) Clothespin test, (C) the Bottle Transfer task and (D) the Bottle Turn tasks. CoS performed better than LDA in three out of four tasks. (* indicates $p < 0.05$, the horizontal line within a box indicates median, the box indicates IQR, and the circles indicate outliers).

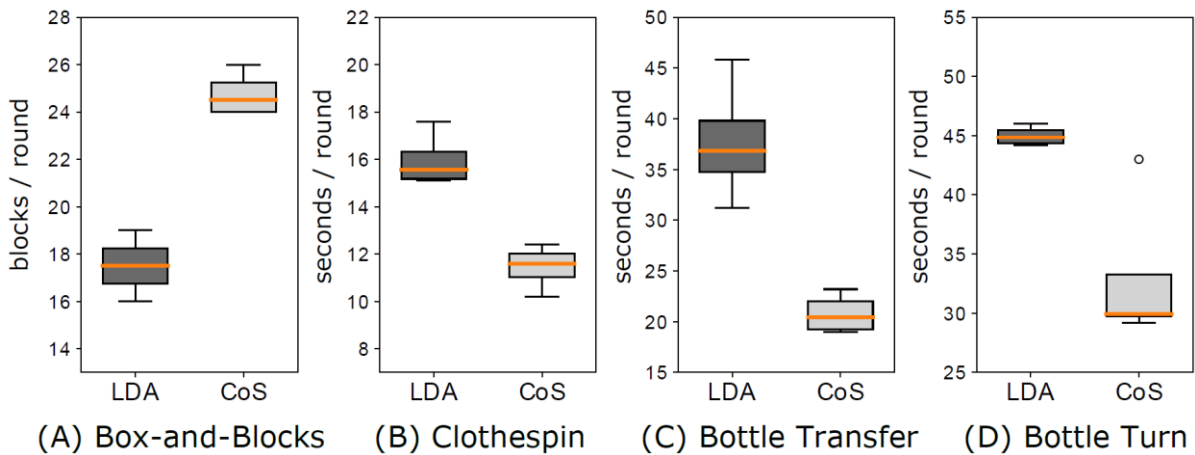


Figure 4.7: Summary of the results for the amputee participant. (A) The Box-and-Blocks test, (B) Clothespin test, (C) the Bottle Transfer task and (D) the Bottle Turn tasks. CoS performed consistently better than LDA in all four tasks. (the horizontal line within a box indicates median, the box indicates IQR, and the circles indicate outliers).

Figure 4.8 summarizes the results for the number of bottles dropped by the able-bodied participants during the Bottle Turn and Transfer tasks. The number of drops observed in the Bottle Transfer task with CoS and LDA were similar, i.e. $0\{0-0\}$ vs $0\{0-0.25\}$ drops per round, respectively. For the Bottle Turn task, the number of drops with CoS ($0\{0-0\}$) was significantly lower than LDA ($0\{0-1\}$). The amputee did not drop any bottles during the Bottle Transfer or Turn task with either of the two methods.

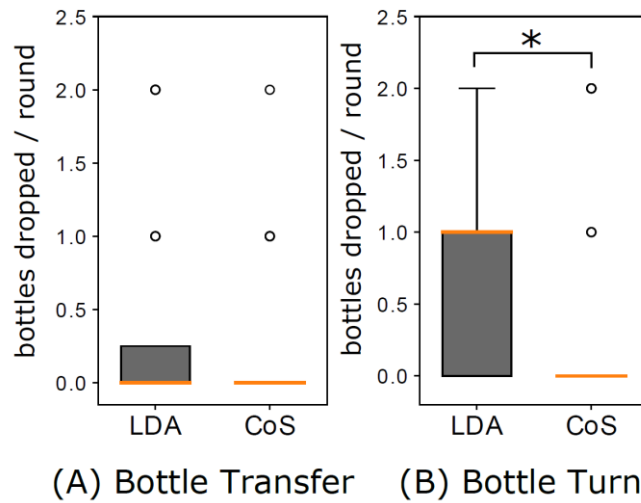


Figure 4.8: Number of bottles dropped per round by able-bodied participants during (A) the Bottle Transfer and (B) the Bottle Turn tasks. (* indicates $p < 0.05$, the horizontal line within a box indicates median, the box indicates IQR, and the circles indicate outliers).

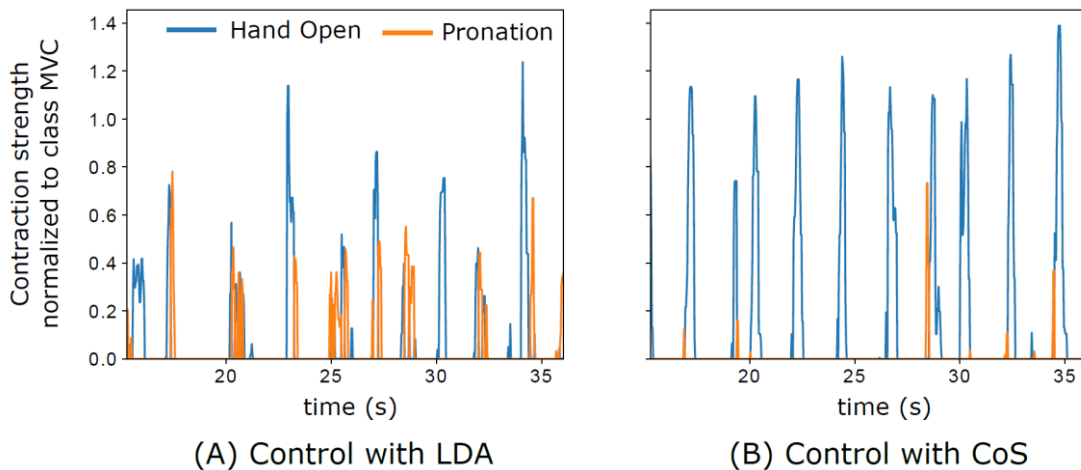


Figure 4.9: Prosthesis command (for hand opening and pronation) generated by the amputee during the Box-and-Blocks test. The hand opening commands were occasionally misclassified as pronation by both LDA and CoS. However, these misclassifications were more frequent in the control with LDA than with CoS. (The Box-and-Blocks test required only hand opening and closing for task completion and so, it is assume that the amputee did not intentionally activate pronation during the test.)

Lastly, the subjective assessment given by the participants was generally in favour of CoS with respect to LDA. Specifically, 8 out of 11 able-bodied participants reported better experience with CoS, 2 participants reported better experience with LDA and 1 participant reported perceiving no difference between LDA and CoS. The amputee gave a subjective score of 8 to CoS and 4 to LDA, i.e. CoS was preferred over LDA. Importantly, the

amputee reported having problems when trying to open the hand, as the hand opening commands were often misclassified as pronation. As shown in Figure 4.9, this problem was more dominant while controlling the prosthesis using LDA. Lastly, the average subjective score given by all participants for CoS (7.1 ± 0.9) was significantly higher than for LDA (5.9 ± 1.6), with $p < 0.05$.

4.5 Discussion

This chapter presented a myoelectric classification scheme based on cosine similarity, which was inspired by the underlying physiological principle of muscle coordination [53], [54]. Specifically, the presented CoS classifier was designed based on the assumption that amplitude-related EMG features for each movement are distributed along the line joining the origin of the feature space and the average maximum voluntary contraction (MVC) of the movement. In an online experiment, the CoS classifier was extensively compared with the academic state-of-the-art LDA classifier using four relevant and challenging functional tasks. The experimental results showed that CoS significantly outperformed LDA in three out of four tasks for the able-bodied participants, and it performed consistently better than LDA in all tasks for the amputee participant. Thus, the CoS classifier not only provides a simple way to model the physiology behind muscle coordination, but it also provides an efficient online myoelectric control. In general, this chapter highlights the potential advantage of approaches that are inspired by the underlying physiological principles over purely data-driven methods that largely prevail in the literature.

Overall, the CoS classifier offered a number of practical advantages over the state-of-the-art LDA classifier. First, the time required to train CoS (30 s) was much lower than LDA (7 min), which is very desirable for the day-to-day use of the system [68]. A brief training time (as with CoS) makes it very easy for the users to (re-)calibrate the system each time the donning and doffing of electrodes occurs (e.g. every morning before and after shower). Second, a 8-dimensional MAV feature space was used to operate the CoS classifier, whereas a 32-dimensional Hudgins time-domain feature space was used by the LDA classifier. Therefore, it is possible to simplify the EMG acquisition setup for operating CoS, as amplitude-related EMG features require a much lower sampling rate ($\sim 200\text{Hz}$) as compared to the high sampling rate ($\sim 1\text{ kHz}$) required for extracting the full time-domain feature set. This, and the fact that only prototype vectors need to be stored in memory, makes CoS suitable for implementation in a simple microcontroller (with a low sampling frequency and limited memory). Lastly, the CoS classifier is also suitable for adaptation (or

incremental learning), which is an important mechanism when considering a daily use [69]. For adaptation, the prototype for a single class or selected subset of classes, whose classification performance needs to be improved, could be easily updated by recording a few additional contractions.

This chapter investigated if and how well the cosine similarity based modelling of EMG can be applied for online myoelectric control. The experimental results demonstrated that this modelling provides a superior performance in comparison to the standard Gaussian distribution based modelling (as in LDA). The current investigation can be seen as a first step towards applying cosine similarity as a metric for assessing the quality of sEMG patterns during online control. Previously, the quality of sEMG patterns was assessed by calculating the log-likelihood probabilities (i.e. via Gaussian modelling), which were then used as a confidence measure to reject potentially erroneous decisions made by the classifier [111], [112]. Similarly, the cosine similarity between the input (amplitude-related) EMG features and the prototype vector of the classified movement could also be used as a confidence measure to reject potentially erroneous decisions. Theoretically speaking, there are two possible ways of implementing a rejection-scheme based on cosine similarity. The first possibility would be the so-called *direct rejection*, in which all classified decisions with similarity less than a predefined threshold would be directly rejected. This scheme is similar to the log-likelihood based rejection scheme proposed by [112], where all classified decisions with likelihood less than 0.97 were rejected. The second possibility is the so-called *cross-talk rejection*, which could be used for suppressing cross-talk between two very close EMG patterns in the feature space. For example, the patterns extension and ulnar-deviation could sometimes be adjacent to each other in the feature space (as in Figure 4.1A) and therefore, the region near the boundary between the two classes is a place where cross-talk could occur. If EMG features were produced in this region during online control, the classified decisions would continuously toggle between the two classes. This problem could be avoided by checking if an input EMG feature is between the prototype vectors of the two adjacent classes and then rejecting the classified decision, if the similarity is less than a predefined threshold. But if the input EMG feature is not between the prototype vectors of the two adjacent classes (with possible cross-talk), the classified decision shall not be rejected (because such an EMG feature is far away from the region where cross-talk could occur).

4.5.1 Integrating Context and Physiology Information

The CoS classifier is compatible with the concepts presented in the previous two chapters. In Chapter 2, Modular Regression (MR) was used to formulate each prosthesis function as a module, which the user could insert/remove as required to best accomplish a given task. Similarly, the CoS classifier associates each prosthesis function to a prototype vector, that can be considered equivalent to a module in MR. Therefore, each prototype vector can be treated as a module, which the user can insert/remove as and when required. The only difference between MR and CoS is that, MR is a regression-based method that can provide a simultaneous and proportional control, whereas CoS is a classification method that provides a sequential and proportional control. But, the concept of modularity and therefore the possibility to integrate user-generated context information is also available with CoS.

In Chapter 3, the context information available from embedded sensors was used to detect prosthesis states, and the robustness of a machine learning controller was improved by adapting the activation thresholds to mitigate expected disturbances in each state. The presented CoS classifier is also an example of a machine learning controller and therefore, the same set of prosthesis states and post-processing rules could be adapted to further improve the robustness of control with CoS. For didactic purposes, Figure 4.10 shows how the context aware component (CAC, from Chapter 3) could be combined with the CoS classifier. Here, a new post-processing scheme based on the cosine similarity metric has been suggested to improve/replace the previous post-processing rules based on simple activation thresholds. Thus, instead of directly suppress the influence of a possibly spurious EMG activation, the cosine similarity metric (and therefore, the physiological principle behind muscle coordination) is being proposed to examine whether the input EMG was generated intentionally by the user or caused due to unwanted muscle activation. This could be achieved by first measuring the cosine similarity of the input EMG with respect to the prototype vector of the detected movement, and then the CAC would reject the decision made by CoS, if the measured similarity is below a certain predefined similarity-threshold.

In theory, combining CAC with CoS to reject erroneous decisions is (perhaps) better than using *direct rejection*, where decisions are rejected based on a fixed (or static) threshold. Specifically, for direct rejection, there is always a trade-off when deciding on the absolute (or exact) value of the rejection threshold. If the rejection threshold is high, the robustness increases at the cost of user experience, because a high threshold would very often reject correct (or good enough) EMG patterns generated by the user. And, a lower rejection

threshold compromises robustness by allowing incorrect decision to pass. Contrary to this, using CAC allows us to adapt the rejection thresholds individually for different movements depending on the prosthesis state. For example, as seen in Figure 4.10, the rejection threshold for hand opening in the Moving state is higher, because it is less likely for the user to release the object while moving, whereas this threshold is lower in the Holding state where the user is more likely to release the object. Furthermore, it is also possible to entirely disable rejection of certain movements in specific states, for example, not rejecting hand open/close commands in the Closing/Grasping state (to allow smooth closing of the grip around an object) or not rejecting rotation commands in the Holding state (to allow smooth rotation/manipulation of object).

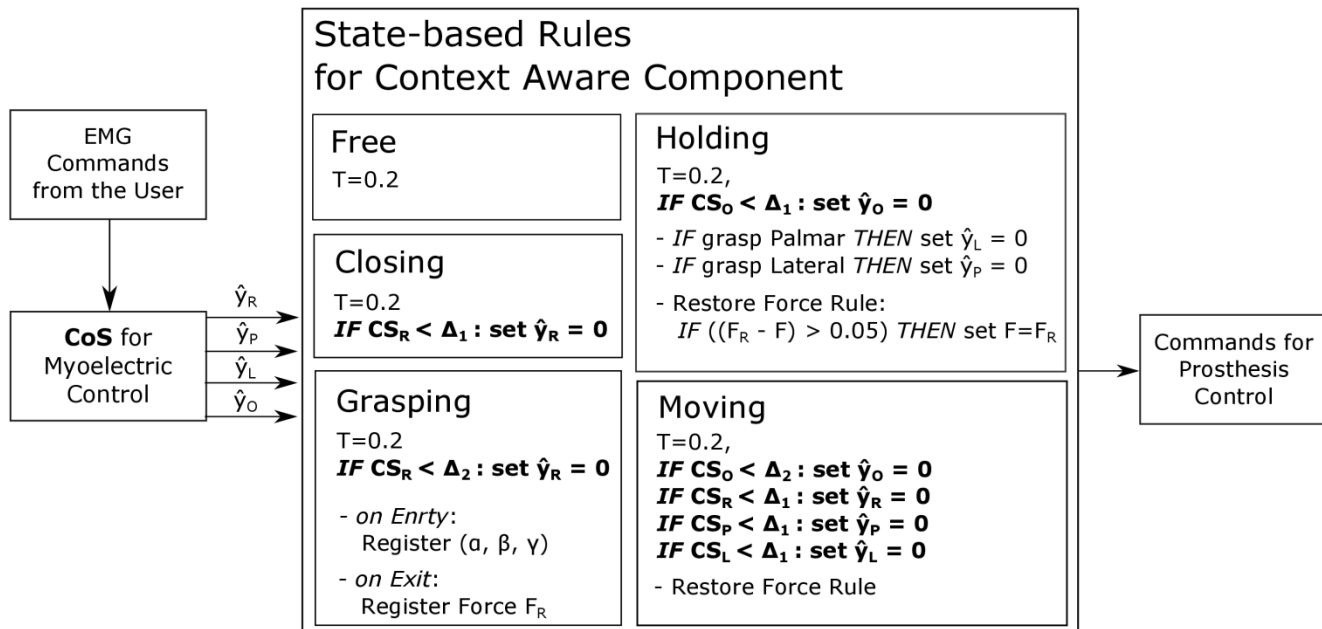


Figure 4.10: Post-processing rules based on cosine similarity (CS) for the context aware component (CAC) presented in Chapter 3 (Figure 3.2 and Figure 3.3). In each of the five states, the post-processing rules based-on activation thresholds (T) have been changed to IF-statements based on cosine similarity for evaluating the quality of EMG patterns responsible for the detected movement. The annotations are: \hat{y}_i – the estimated strength for the classified movement (and $\hat{y}_i = 0$ for all other movement), CS_i – cosine similarity of the input with respect to the prototype vector of the detected movement, Δ_1, Δ_2 – rejection thresholds ($\Delta_2 > \Delta_1$), R – Rotation, L – Lateral, P – Palmar, O – Hand Opening, F – grip force. Lastly, it is to be noted that, a fixed activation of $T = 0.2$ is always applied to remove uncertainties at low contraction intensities.

Context- and Physiology-aware Machine Learning for
Upper-Limb Myocontrol

5 CONCLUSION

The work presented in this thesis addressed the problems associated with most ML-based approaches used for upper-limb myocontrol. Usually, ML-based myocontrol is done by following the conventional pattern recognition paradigm, where training data is collected using a supervised procedure and then, a mathematical function is fitted over the data to define an invariant mapping scheme between the user's EMG patterns and available prosthesis functions. Two problems associated with this conventional paradigm were identified, and possible solutions were presented by developing new ML-based approaches for myocontrol. The first problem with the conventional paradigm was that, the mapping scheme (between user's EMG and prosthesis functions) remained static/invariant during use and did not consider the dynamics associated with the real-life use of a prosthesis. This problem was resolved by developing ML methods that can adapt the mapping scheme based on context information acquired from either the user or additional sensors placed on the prosthesis. In this way, the mapping scheme did not remain static, but it became reactive to the inferred context information. The second problem with the conventional paradigm was that, the mathematical function fitted over the training data was usually assumed to implicitly model the physiological principles behind generation of EMG patterns, but this assumption may not be true for many commonly used ML methods for myocontrol (such as the LDA). This problem was resolved by using a mathematical metric (cosine similarity) that was inspired by the physiological principle of muscle coordination and then, the same metric was applied for online myoelectric control.

The concept of exploiting user-generated context information to improve the performance of ML-based control was investigated in Chapter 2. It was hypothesized that, this concept

(of exploiting user-generated context information) has a twofold advantage for online myoelectric control. First, it gives to the users a possibility to interact with the underlying ML controller and second, it gives to the ML controller a possibility to understand users' requirement(s) and adapt accordingly to optimize performance. This concept was realized by developing a new ML method called Modular Regression (MR), which was able to integrate task-specific context information generated by the user. Specifically, with MR, the EMG patterns associated with different prosthesis functions were organized into modules (or doublets, see Equation 2.6), which the users could interactively insert/remove as required to best accomplish a given ADL. The experimental evaluation indicated that, ADLs (in the experiment) could be performed significantly faster with MR in comparison to the classic approach of using all available functions at all time. In general, the presented method illustrates the benefits of (the concept of) exploiting user-generated context information to improve ML-based myocontrol and motivates the need to further develop new ML methods, which can integrate a wide-variety of context information generated by the user. Although, as mentioned earlier, demanding too much context information from the user could easily turn the interactive aspect into a burden, and therefore care must be taken to identify a balance between the amount of context information required from the user and the corresponding performance gained.

The concept of acquiring context information from proprioceptive and exteroceptive sensors to improve the performance of ML-based myocontrol was investigated in Chapter 3. Although a wide-variety of additional sensors can be placed either on the user or inside the prosthesis, the presented work focused on sensors which could be easily integrated inside the prosthesis socket – leading to a self-contained system. It was hypothesized that, additional sensors could be used to determine the state of the prosthesis and the mapping scheme of the ML-controller could be adapted to improve the performance depending on the detected prosthesis state. This concept was realized by developing a new approach called context-driven ML control (cxMLC), wherein a context aware component (CAC) was used to determine the state of the prosthesis and the mapping scheme was adapted by applying a set of post-processing rules to mitigate expected disturbances in each state. The experimental evaluation indicated that, cxMLC was able to minimize the number of spurious/unwanted prosthesis activations and thereby improve performance (success rate and/or dropped objects) in comparison to the conventional ML approach (where the mapping scheme was not adapted with changing prosthesis states). In general, the presented cxMLC control shows the benefits of (the concept of) acquiring context information from

additional sensors to improve control and motivates the need to further develop new approaches that can integrate more context information coming from new/same additional sensors. Although, care must be taken to keep the system self-contained within the prosthesis socket or to minimize the number of sensors being placed on the user, as having a large number of sensors on the user could directly increase the overhead of wearing and removing additional sensors each time the donning and doffing of system occurs.

The benefit of employing the knowledge of latent physiological principles to achieve online myoelectric control was investigated in Chapter 4. It was hypothesized that, instead of using off-the-shelf mathematical functions to model EMG data for online control, the modelling of EMG could be improved by using the knowledge of latent physiological principles that are responsible for the generation EMG patterns. This concept was realized by developing a new classification scheme based on cosine similarity (CoS), which is a mathematical metric that reflects upon the physiological principle of muscle coordination [53], [54]. An online experiment was conducted to compare the new CoS classifier with the academic state-of-the-art LDA classifier. The experimental evaluation indicated that, the CoS classifier performed better than the LDA, while utilizing less training data and a lower dimensional feature set (amplitude vs. time domain features); thereby, offering a possibility to simplify the training procedure and acquisition hardware. In general, the presented method points to the potential advantage of approaches that are inspired by the underlying physiological principles over purely data-driven methods that largely prevail in the literature.

Different concepts presented in this thesis were strongly linked to each other. The main concept in Chapter 2 was to allow customization of ML control by organizing different prosthesis functions into modules, which the users could interactively insert/remove to improve the robustness of myocontrol. This concept is also applicable to the CoS classifier presented in Chapter 4, where each movement was modelled using a prototype vector, which the users can interactively insert/remove as required. Next, the main concept in Chapter 3 was to combine a ML-based myoelectric controller with a context aware component (CAC), which infers context information coming from additional prosthesis sensors to mitigate expected disturbances. The CoS classifier presented in Chapter 4 is an example of a ML-based myoelectric controller and therefore, the performance of CoS could indeed be improved by combining it with the CAC. Hence, the CoS classifier not only models the physiological principle behind muscle coordination, but it also offers the possibility to utilize context information coming from the user and/or additional prosthesis

sensors – thus, leading to a context- and physiology- aware machine learning scheme for upper-limb myocontrol.

Lastly, the clinical relevance of this work can be emphasized from the fact that, all presented methods were designed to improve the robustness of ML-based myoelectric control. This would potentially give the users more confidence in using their ML-based myoelectric controller, and could eventually increase the acceptance-rate of ML-based myoelectric prostheses in the market. Plus, all presented methods were evaluated using a set of relevant and challenging functional tasks, which can be seen as a first step toward the clinical transferability of the developed methods. Moreover, all presented ML methods are simple and computationally efficient, and therefore can be directly used for interfacing most prosthetic devices available in the market, with a minor hardware upgrade. For example, the use of MR requires a simple smartphone App, which is very easy to provide given that most patients might own a smartphone. Similarly, to use cxMLC, manufacturers would need to place few additional sensors in the prosthesis socket, some of which are already available in existing devices. For example, the Michelangelo hand (from Ottobock, DE) has a force and aperture sensor but does not have an IMU, whereas the i-Limb hand (from Touch Bionics, UK) has an IMU but does not have a force and aperture sensor. Additionally, this work was focused on improving the performance of ML-based myocontrol driven by non-invasive (surface) EMG recording and therefore, it is very suitable for patients who want to avoid surgical procedures targeted towards improving control (via invasive EMG recording [63], [87]), but still want to use a multi-articulated prosthesis with a dexterous control.

6 REFERENCES

- [1] D. Ootes, K. T. Lambers, and D. C. Ring, “The Epidemiology of Upper Extremity Injuries Presenting to the Emergency Department in the United States,” *HAND*, vol. 7, no. 1, pp. 18–22, Mar. 2012.
- [2] C. F. Larsen, S. Mulder, A. M. T. Johansen, and C. Stam, “The Epidemiology of Hand Injuries in the Netherlands and Denmark,” *Eur. J. Epidemiol.*, vol. 19, no. 4, pp. 323–327.
- [3] Kahn A. P., *The Encyclopedia of Work-related Illnesses, Injuries, and Health Issues*. vol. 1. New York, New York, USA, 2004.
- [4] A. C. Rettig, “Athletic Injuries of the Wrist and Hand,” *Am. J. Sports Med.*, vol. 31, no. 6, pp. 1038–1048, Nov. 2003.
- [5] K. Ziegler-Graham, E. J. MacKenzie, P. L. Ephraim, T. G. Trivison, R. Brookmeyer, and A. K. Ziegler-Graham, “Estimating the Prevalence of Limb Loss in the United States: 2005 to 2050.”
- [6] T. R. Dillingham, L. E. Pezzin, and E. J. Mackenzie, “Limb Amputation and Limb Deficiency: Epidemiology and Recent Trends in the United States,” *South. Med. J.*, vol. 95, no. 8, pp. 875–883, Aug. 2002.
- [7] V. PUTTI, “Historical Prostheses,” *J. Hand Surg. J. Br. Soc. Surg. Hand*, vol. 30, Jun. 2005.
- [8] B. Maat, G. Smit, D. Plettenburg, and P. Breedveld, “Passive prosthetic hands and tools: A literature review,” *Prosthet. Orthot. Int.*, p. 30936461769162, Mar. 2017.
- [9] J. A. Doubler and D. S. Childress, “An analysis of extended physiological proprioception as a prosthesis-control technique,” *J. Rehabil. Res. Dev.*, vol. 21, no. 1, pp. 5–18, May 1984.
- [10] R. M. Enoka and J. Duchateau, “Physiology of Muscle Activation and Force Generation,” in *Surface Electromyography: Physiology, Engineering, and Applications*, Hoboken, New Jersey: John Wiley & Sons, Inc., 2016, pp. 1–29.
- [11] S. Lewis, M. F. Russold, H. Dietl, and E. Kaniusas, “Satisfaction of Prosthesis Users with Electrical Hand Prostheses and their Suggested Improvements,” *Biomed. Eng. / Biomed. Tech.*, Jan. 2013.
- [12] E. A. Biddiss and T. T. Chau, “Upper limb prosthesis use and abandonment: A survey of the last 25 years,” *Prosthet. Orthot. Int.*, vol. 31, no. 3, pp. 236–257, Sep. 2007.
- [13] E. Biddiss and T. Chau, “Upper-limb prosthetics: critical factors in device abandonment,” *Am J Phys Med Rehabil*, vol. 86, no. 12, pp. 977–987, 2007.
- [14] W. Schweitzer, M. J. Thali, and D. Egger, “Case-study of a user-driven prosthetic arm design:

bionic hand versus customized body-powered technology in a highly demanding work environment,” *J. Neuroeng. Rehabil.*, vol. 15, no. 1, p. 1, Dec. 2018.

- [15] S. Raspopovic *et al.*, “Restoring natural sensory feedback in real-time bidirectional hand prostheses,” *Sci. Transl. Med.*, vol. 6, no. 222, p. 222ra19, Feb. 2014.
- [16] C. Antfolk *et al.*, “Sensory feedback in upper limb prosthetics,” *Expert Rev. Med. Devices*, vol. 10, no. 1, pp. 45–54, Jan. 2013.
- [17] S. Amsuess, P. Goebel, B. Graimann, and D. Farina, “Extending mode switching to multiple degrees of freedom in hand prosthesis control is not efficient,” in *2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, 2014, pp. 658–661.
- [18] N. Jiang, S. Dosen, K.-R. Müller, and D. Farina, “Myoelectric control of artificial limbs: is there the need for a change of focus?,” *IEEE Signal Process. Mag.*, vol. 29, no. 5, pp. 149–152, 2012.
- [19] Y. Gu, K. Dremstrup, and D. Farina, “Single-trial discrimination of type and speed of wrist movements from EEG recordings,” *Clin. Neurophysiol.*, vol. 120, no. 8, pp. 1596–1600, Aug. 2009.
- [20] P. A. Parker, K. Englehart, and B. Hudgins, “Myoelectric signal processing for control of powered limb prostheses,” *J. Electromyogr. Kinesiol.*, vol. 16, no. 6, 2006.
- [21] “How the i-limb works | Touch Bionics.” [Online]. Available: <http://www.touchbionics.com/products/how-i-limb-works>. [Accessed: 13-Oct-2017].
- [22] “Bionic hand with 14 grip patterns that makes common tasks easy. - bebionic.” [Online]. Available: http://bebionic.com/the_hand/grip_patterns. [Accessed: 13-Oct-2017].
- [23] D. Graupe, J. Salahi, and K. H. Kohn, “Multifunctional prosthesis and orthosis control via microcomputer identification of temporal pattern differences in single-site myoelectric signals,” *J. Biomed. Eng.*, vol. 4, no. 1, pp. 17–22, Jan. 1982.
- [24] Guanglin Li, A. E. Schultz, and T. A. Kuiken, “Quantifying Pattern Recognition—Based Myoelectric Control of Multifunctional Transradial Prostheses,” *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 18, no. 2, pp. 185–192, Apr. 2010.
- [25] J. Wang, L. Tang, and J. E. Bronlund, “Surface EMG Signal Amplification and Filtering,” *Int. J. Comput. Appl.*, vol. 82, pp. 975–8887, 2013.
- [26] K. Englehart and B. Hudgins, “A robust, real-time control scheme for multifunction myoelectric control,” *IEEE Trans. Biomed. Eng.*, vol. 50, no. 7, pp. 848–854, Jul. 2003.
- [27] S. Amsuess, P. Goebel, B. Graimann, and D. Farina, “A Multi-Class Proportional Myocontrol Algorithm for Upper Limb Prosthesis Control: Validation in Real-Life Scenarios on Amputees,” *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 23, no. 5, pp. 827–836, Sep. 2015.
- [28] 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.
- [29] Z. Ju, G. Ouyang, M. Wilamowska-Korsak, and H. Liu, “Surface EMG Based Hand Manipulation Identification Via Nonlinear Feature Extraction and Classification,” *IEEE Sens. J.*, vol. 13, no. 9, pp. 3302–3311, Sep. 2013.
- [30] 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 Trans. Neural Syst. Rehabil. Eng.*, vol. 4320, no. c, pp. 1–1, 2017.
- [31] X. Chen, X. Zhu, and D. Zhang, “A discriminant bispectrum feature for surface electromyogram signal classification,” *Med. Eng. Phys.*, vol. 32, no. 2, pp. 126–135, Mar. 2010.
- [32] L. H. Smith, T. A. Kuiken, and L. J. Hargrove, “Evaluation of Linear Regression Simultaneous Myoelectric Control Using Intramuscular EMG,” *IEEE Trans. Biomed. Eng.*, vol. 63, no. 4, pp. 737–746, Apr. 2016.

- [33] J. M. Hahne *et al.*, “Linear and Nonlinear Regression Techniques for Simultaneous and Proportional Myoelectric Control,” *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 22, no. 2, pp. 269–279, 2014.
- [34] L. H. Smith, T. A. Kuiken, and L. J. Hargrove, “Use of probabilistic weights to enhance linear regression myoelectric control,” *J. Neural Eng.*, vol. 12, no. 6, p. 66030, Dec. 2015.
- [35] K. Englehart, B. Hudgins, P. . A. Parker, M. Stevenson, and T. Hortobágyi, “Classification of the myoelectric signal using time-frequency based representations,” *Med. Eng. Phys.*, vol. 21, no. 6–7, pp. 431–438, Jul. 1999.
- [36] A. H. Al-Timemy, G. Bugmann, J. Escudero, and N. Outram, “Classification of Finger Movements for the Dexterous Hand Prosthesis Control With Surface Electromyography,” *IEEE J. Biomed. Heal. Informatics*, vol. 17, no. 3, pp. 608–618, May 2013.
- [37] L. J. Hargrove, Guanglin Li, K. B. Englehart, and B. S. Hudgins, “Principal Components Analysis Preprocessing for Improved Classification Accuracies in Pattern-Recognition-Based Myoelectric Control,” *IEEE Trans. Biomed. Eng.*, vol. 56, no. 5, pp. 1407–1414, May 2009.
- [38] D. Sierra González and C. Castellini, “A realistic implementation of ultrasound imaging as a human-machine interface for upper-limb amputees,” *Front. Neurobot.*, vol. 7, no. 17, 2013.
- [39] S. Amsuess *et al.*, “Context-Dependent Upper Limb Prosthesis Control for Natural and Robust Use,” *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 24, no. 7, pp. 744–753, Jul. 2016.
- [40] C. Saunders, A. Gammernan, V. Vovk, and R. Holloway, “Ridge Regression Learning Algorithm in Dual Variables,” *Proc. Fifteenth Int. Conf. Mach. Learn. (San Fr. CA)*, 1998.
- [41] A. Rahimi and B. Recht, “Uniform approximation of functions with random bases,” in *2008 46th Annual Allerton Conference on Communication, Control, and Computing*, 2008, pp. 555–561.
- [42] K. S. Kim, H. H. Choi, C. S. Moon, and C. W. Mun, “Comparison of k-nearest neighbor, quadratic discriminant and linear discriminant analysis in classification of electromyogram signals based on the wrist-motion directions,” *Curr. Appl. Phys.*, vol. 11, no. 3, pp. 740–745, May 2011.
- [43] M. A. Oskoei and Huosheng Hu, “Support Vector Machine-Based Classification Scheme for Myoelectric Control Applied to Upper Limb,” *IEEE Trans. Biomed. Eng.*, vol. 55, no. 8, pp. 1956–1965, Aug. 2008.
- [44] N. Jiang, I. Vujaklija, H. Rehbaum, B. Graimann, and D. Farina, “Is Accurate Mapping of EMG Signals on Kinematics Needed for Precise Online Myoelectric Control?,” *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 22, no. 3, pp. 549–558, May 2014.
- [45] 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, no. 1, 2014.
- [46] M. Ison and P. Artemiadis, “The role of muscle synergies in myoelectric control: trends and challenges for simultaneous multifunction control,” *J. Neural Eng.*, vol. 11, no. 5, p. 51001, Oct. 2014.
- [47] G. Rasool, K. Iqbal, N. Bouaynaya, and G. White, “Real-Time Task Discrimination for Myoelectric Control Employing Task-Specific Muscle Synergies,” *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 24, no. 1, pp. 98–108, 2016.
- [48] E. Bizzi, V. C. K. Cheung, A. d’Avella, P. Saltiel, and M. Tresch, “Combining modules for movement,” *Brain Res. Rev.*, vol. 57, no. 1, pp. 125–133, Jan. 2008.
- [49] A. d’Avella and E. Bizzi, “Shared and specific muscle synergies in natural motor behaviors.,” *Proc. Natl. Acad. Sci. U. S. A.*, vol. 102, no. 8, pp. 3076–81, Feb. 2005.
- [50] M. C. Tresch, V. C. K. Cheung, and A. d’Avella, “Matrix Factorization Algorithms for the Identification of Muscle Synergies: Evaluation on Simulated and Experimental Data Sets,” *J. Neurophysiol.*, vol. 95, no. 4, 2006.

- [51] N. Jiang, H. Rehbaum, I. Vujaklija, B. Graimann, and D. Farina, "Intuitive, Online, Simultaneous, and Proportional Myoelectric Control Over Two Degrees-of-Freedom in Upper Limb Amputees," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 22, no. 3, pp. 501–510, May 2014.
- [52] D. Farina *et al.*, "Man/machine interface based on the discharge timings of spinal motor neurons after targeted muscle reinnervation," *Nat. Biomed. Eng.*, vol. 1, no. 2, p. 25, Feb. 2017.
- [53] F. J. Valero-Cuevas, "Predictive Modulation of Muscle Coordination Pattern Magnitude Scales Fingertip Force Magnitude Over the Voluntary Range," *J Neurophysiol*, vol. 83, no. 3, pp. 1469–1479, 2000.
- [54] B. Poston, A. Danna-Dos Santos, M. Jesunathadas, T. M. Hamm, and M. Santello, "Force-Independent Distribution of Correlated Neural Inputs to Hand Muscles During Three-Digit Grasping," *J. Neurophysiol.*, vol. 104, no. 2, pp. 1141–1154, 2010.
- [55] J. He, D. Zhang, X. Sheng, S. Li, and X. Zhu, "Invariant surface EMG feature against varying contraction level for myoelectric control based on muscle coordination," *IEEE J. Biomed. Heal. Informatics*, vol. 19, no. 3, pp. 874–882, 2015.
- [56] A. H. Al-Timemy, R. N. Khushaba, G. Bugmann, and J. Escudero, "Improving the Performance Against Force Variation of EMG Controlled Multifunctional Upper-Limb Prostheses for Transradial Amputees," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 24, no. 6, pp. 650–661, Jun. 2016.
- [57] D. Farina and O. Aszmann, "Bionic limbs: clinical reality and academic promises.," *Sci. Transl. Med.*, vol. 6, no. 257, p. 257ps12, Oct. 2014.
- [58] B. Lock, K. Englehart, and B. Hudgins, "Real-time myoelectric control in a virtual environment to relate usability vs. accuracy," *Proc. MyoElectric Control. Prosthetics Symp., Fredericton, NB, Canada*, 2005.
- [59] "COAPT Complete Control." <https://www.coaptengineering.com/>.
- [60] L. Hargrove, K. Englehart, and B. Hudgins, "The effect of electrode displacements on pattern recognition based myoelectric control," in *2006 International Conference of the IEEE Engineering in Medicine and Biology Society*, 2006, pp. 2203–2206.
- [61] L. Hargrove, K. Englehart, and B. Hudgins, "A training strategy to reduce classification degradation due to electrode displacements in pattern recognition based myoelectric control ," *Biomed. Signal Process. Control* , vol. 3, no. 2, pp. 175–180, 2008.
- [62] N. Jiang, S. Muceli, B. Graimann, and D. Farina, "Effect of arm position on the prediction of kinematics from EMG in amputees," *Med. Biol. Eng. Comput.*, vol. 51, no. 1–2, pp. 143–151, Feb. 2013.
- [63] M. Ortiz-Catalan, B. Håkansson, R. Brånemark, B. Håkansson, and R. Brånemark, "An osseointegrated human-machine gateway for long-term sensory feedback and motor control of artificial limbs," *Sci. Transl. Med.*, vol. 6, no. 257, p. 257re6–257re6, 2014.
- [64] A. M. Simon, L. J. Hargrove, B. A. Lock, and T. A. Kuiken, "A Decision-Based Velocity Ramp for Minimizing the Effect of Misclassifications During Real-Time Pattern Recognition Control," *IEEE Trans. Biomed. Eng.*, vol. 58, no. 8, pp. 2360–2368, Aug. 2011.
- [65] L. J. Hargrove, E. J. Scheme, K. B. Englehart, and B. S. Hudgins, "Multiple Binary Classifications via Linear Discriminant Analysis for Improved Controllability of a Powered Prosthesis," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 18, no. 1, pp. 49–57, Feb. 2010.
- [66] E. J. Scheme, K. B. Englehart, and B. S. Hudgins, "Selective Classification for Improved Robustness of Myoelectric Control Under Nonideal Conditions," *IEEE Trans. Biomed. Eng.*, vol. 58, no. 6, pp. 1698–1705, Jun. 2011.
- [67] A. Gijsberts *et al.*, "Stable myoelectric control of a hand prosthesis using non-linear incremental learning," *Front. Neurobot.*, vol. 8, no. 8, 2014.

- [68] J. W. Sensinger, B. A. Lock, and T. A. Kuiken, "Adaptive Pattern Recognition of Myoelectric Signals: Exploration of Conceptual Framework and Practical Algorithms," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 17, no. 3, pp. 270–278, Jun. 2009.
- [69] C. Castellini, "Incremental learning of muscle synergies: from calibrating a prosthesis to interacting with it," in *Human and robot hands - Sensorimotor Synergies to Bridge the Gap between Neuroscience and Robotics*, A. Moscatelli and M. Bianchi, Eds. Springer Netherlands, 2015.
- [70] J. M. Hahne *et al.*, "Concurrent Adaptation of Human and Machine Improves Simultaneous and Proportional Myoelectric Control," vol. 23, no. 4, pp. 618–627, Jul. 2015.
- [71] D. Novak and R. Riener, "A survey of sensor fusion methods in wearable robotics," in *Robotics and Autonomous Systems*, 2015, vol. 73, pp. 155–170.
- [72] A. Gijsberts and B. Caputo, "Exploiting Accelerometers to Improve Movement Classification for Prosthetics."
- [73] A. Fougner, E. Scheme, A. D. C. Chan, K. Englehart, and Ø. Stavadahl, "Resolving the Limb Position Effect in Myoelectric Pattern Recognition," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 19, no. 6, pp. 644–651, Dec. 2011.
- [74] E. Scheme, A. Fougner, Stavadahl, A. D. C. Chan, and K. Englehart, "Examining the adverse effects of limb position on pattern recognition based myoelectric control," *2010 Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. EMBC'10*, pp. 6337–6340, 2010.
- [75] A. Krasoulis, I. Kyranou, M. S. Erden, K. Nazarpour, and S. Vijayakumar, "Improved prosthetic hand control with concurrent use of myoelectric and inertial measurements," *J. Neuroeng. Rehabil.*, vol. 14, no. 1, p. 71, 2017.
- [76] Y. Fang, N. Hettiarachchi, D. Zhou, and H. Liu, "Multi-modal sensing techniques for interfacing hand prostheses: A review," *IEEE Sens. J.*, 2015.
- [77] M. Wininger, N. Kim, and W. Craelius, "Pressure signature of forearm as predictor of grip force," *J. Rehabil. Res. Dev.*, vol. 45, no. 6, pp. 883–892, 2008.
- [78] N. Jaquier, M. Connan, C. Castellini, and S. Calinon, "Combining Electromyography and Tactile Myography to Improve Hand and Wrist Activity Detection in Prostheses," *Technologies*, vol. 5, no. 4, p. 64, Oct. 2017.
- [79] Y. Geng *et al.*, "Toward attenuating the impact of arm positions on electromyography pattern-recognition based motion classification in transradial amputees," *J. Neuroeng. Rehabil.*, vol. 9, no. 1, p. 74, 2012.
- [80] Y. Geng, O. W. Samuel, Y. Wei, and G. Li, "Improving the Robustness of Real-Time Myoelectric Pattern Recognition against Arm Position Changes in Transradial Amputees," *Biomed Res. Int.*, vol. 2017, pp. 1–10, Apr. 2017.
- [81] R. Tomovic and G. Boni, "An adaptive artificial hand," *IRE Trans. Autom. Control*, vol. 7, no. 3, pp. 3–10, Apr. 1962.
- [82] D. P. J. Cotton, P. H. Chappell, A. Cranny, N. M. White, and S. P. Beeby, "A Novel Thick-Film Piezoelectric Slip Sensor for a Prosthetic Hand," *IEEE Sens. J.*, vol. 7, no. 5, pp. 752–761, May 2007.
- [83] P. J. Kyberd and P. H. Chappell, "The Southampton Hand: an intelligent myoelectric prosthesis," *J. Rehabil. Res. Dev.*, vol. 31, no. 4, pp. 326–34, Nov. 1994.
- [84] N. A. Alshammary, D. A. Bennett, and M. Goldfarb, "Efficacy of coordinating shoulder and elbow motion in a myoelectric transhumeral prosthesis in reaching tasks," *Proc. - IEEE Int. Conf. Robot. Autom.*, vol. 2016–June, pp. 3723–3728, 2016.
- [85] M. Markovic *et al.*, "Stereovision and augmented reality for closed-loop control of grasping in hand prostheses," *J. Neural Eng.*, vol. 11, no. 4, p. 46001, Aug. 2014.
- [86] M. Markovic *et al.*, "Sensor fusion and computer vision for context-aware control of a multi

- degree-of-freedom prosthesis,” *J. Neural Eng.*, vol. 12, no. 6, p. 66022, Dec. 2015.
- [87] T. A. Kuiken *et al.*, “Targeted muscle reinnervation for real-time myoelectric control of multifunction artificial arms.,” *JAMA*, vol. 301, no. 6, pp. 619–28, Feb. 2009.
- [88] I. Vujaklija *et al.*, “Translating Research on Myoelectric Control into Clinics-Are the Performance Assessment Methods Adequate?,” *Front. Neurorobot.*, vol. 11, p. 7, 2017.
- [89] A. Ameri, E. J. Scheme, E. N. Kamavuako, K. B. Englehart, and P. A. Parker, “Real-Time, Simultaneous Myoelectric Control Using Force and Position-Based Training Paradigms,” *IEEE Trans. Biomed. Eng.*, vol. 61, no. 2, pp. 279–287, Feb. 2014.
- [90] M. Ortiz-Catalan, R. Brånemark, and B. Håkansson, “BioPatRec: A modular research platform for the control of artificial limbs based on pattern recognition algorithms.,” *Source Code Biol. Med.*, vol. 8, no. 1, p. 11, Apr. 2013.
- [91] P. M. Fitts, “The Information Capacity of the Human Motor System in Controlling the Amplitude of Movement,” *J. Exp. Biol.*, vol. 47, no. 6, pp. 381–391, 1954.
- [92] J. Park, W. Bae, H. Kim, and S. Park, “EMG — force correlation considering Fitts’ law,” in *2008 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems*, 2008, pp. 644–649.
- [93] E. J. Scheme and K. B. Englehart, “Validation of a Selective Ensemble-Based Classification Scheme for Myoelectric Control Using a Three-Dimensional Fitts’ Law Test,” *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 21, no. 4, pp. 616–623, Jul. 2013.
- [94] A. M. Simon, L. J. Hargrove, B. A. Lock, and T. A. Kuiken, “Target Achievement Control Test: Evaluating real-time myoelectric pattern-recognition control of multifunctional upper-limb prostheses,” *J. Rehabil. Res. Dev.*, vol. 48, no. 6, p. 619, 2011.
- [95] W. Daly, L. Voo, T. Rosenbaum-Chou, A. Arabian, and D. Boone, “Socket Pressure and Discomfort in Upper-Limb Prostheses,” *JPO J. Prosthetics Orthot.*, vol. 26, no. 2, pp. 99–106, Apr. 2014.
- [96] V. Mathiowetz, G. Volland, N. Kashman, K. Weber, and K. Mathiowetz, Virgil and Volland, Gloria and Kashman, Nancy and Weber, “Adult norms for the Box and Block Test of manual dexterity,” *Am. J. Occup. Ther.*, vol. 39, no. 6, pp. 386–391, 1985.
- [97] L. J. Hargrove, B. A. Lock, and A. M. Simon, “Pattern recognition control outperforms conventional myoelectric control in upper limb patients with targeted muscle reinnervation,” in *2013 35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, 2013, pp. 1599–1602.
- [98] C. M. Light, P. H. Chappell, and P. J. Kyberd, “Establishing a standardized clinical assessment tool of pathologic and prosthetic hand function: Normative data, reliability, and validity,” *Arch. Phys. Med. Rehabil.*, vol. 83, no. 6, pp. 776–83, Jun. 2002.
- [99] E. D. Sears and K. C. Chung, “Validity and responsiveness of the Jebsen-Taylor Hand Function Test.,” *J. Hand Surg. Am.*, vol. 35, no. 1, pp. 30–7, Jan. 2010.
- [100] A. M. Bagley, F. Molitor, L. V Wagner, W. Tomhave, and M. A. James, “The Unilateral Below Elbow Test: a function test for children with unilateral congenital below elbow deficiency,” vol. 48, no. 7, pp. 569–575, 2006.
- [101] I. Vujaklija, D. Farina, and O. Aszmann, “New developments in prosthetic arm systems,” vol. 8, pp. 31–39, Jul. 2016.
- [102] World Medical Association, “World Medical Association Declaration of Helsinki: ethical principles for medical research involving human subjects.,” *JAMA*, vol. 310, no. 20, pp. 2191–4, Nov. 2013.
- [103] H. Bouwsema, C. K. van der Sluis, and R. M. Bongers, “Changes in performance over time while learning to use a myoelectric prosthesis,” *J. Neuroeng. Rehabil.*, vol. 11, no. 1, p. 16, Feb.

- 2014.
- [104] Y. Fang, D. Zhou, K. Li, and H. Liu, "Interface Prostheses with Classifier-Feedback based User Training," *IEEE Trans. Biomed. Eng.*, pp. 1–1, 2016.
 - [105] C. Prahm, F. Kayali, I. Vujaklija, A. Sturma, and O. Aszmann, "Increasing motivation, effort and performance through game-based rehabilitation for upper limb myoelectric prosthesis control," in *2017 International Conference on Virtual Rehabilitation (ICVR)*, 2017, pp. 1–6.
 - [106] T. Lorrain, N. Jiang, and D. Farina, "Influence of the training set on the accuracy of surface EMG classification in dynamic contractions for the control of multifunction prostheses," *J. Neuroeng. Rehabil.*, vol. 8, no. 25, pp. 1–9, 2011.
 - [107] "Mind-controlled neuroprosthetics - YouTube." [Online]. Available: https://www.youtube.com/watch?v=ZuJu_uIpvq4. [Accessed: 18-Jun-2017].
 - [108] A. D. C. Chan and K. B. Englehart, "Continuous Myoelectric Control for Powered Prostheses Using Hidden Markov Models," *IEEE Trans. Biomed. Eng.*, vol. 52, no. 1, pp. 121–124, Jan. 2005.
 - [109] C. J. De Luca and Z. Erim, "Common drive of motor units in regulation of muscle force," *Trends Neurosci.*, vol. 17, no. 7, pp. 299–305, 1994.
 - [110] M. Nowak and C. Castellini, "The LET Procedure for Prosthetic Myocontrol: Towards Multi-DOF Control Using Single-DOF Activations," *PLoS One*, vol. 11, no. 9, p. e0161678, Sep. 2016.
 - [111] S. Amsuss, P. M. Goebel, Ning Jiang, B. Graimann, L. Paredes, and D. Farina, "Self-Correcting Pattern Recognition System of Surface EMG Signals for Upper Limb Prosthesis Control," *IEEE Trans. Biomed. Eng.*, vol. 61, no. 4, pp. 1167–1176, Apr. 2014.
 - [112] E. J. Scheme, B. S. Hudgins, and K. B. Englehart, "Confidence-based rejection for improved pattern recognition myoelectric control," *IEEE Trans. Biomed. Eng.*, vol. 60, no. 6, pp. 1563–1570, 2013.

Context- and Physiology-aware Machine Learning for
Upper-Limb Myocontrol

7 APPENDICES

<i>Appendix 1: Questionnaire given after the Experiment in Chapter 3</i>	87
<i>Appendix 2: Dissemination</i>	89

Context- and Physiology-aware Machine Learning for
Upper-Limb Myocontrol

APPENDIX 1: QUESTIONNAIRE GIVEN AFTER THE EXPERIMENT IN CHAPTER 3

Subject Code:

Experiment Questionnaire

Task Physical Demand

Physical Demand: How 'physically demanding' was the task? [Report in terms of arm fatigue or body tiredness experienced in doing each task]

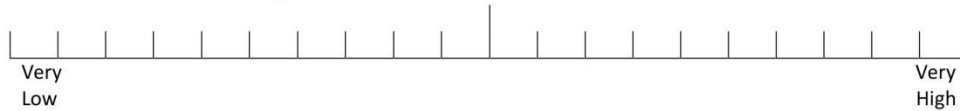
a. Task 1: Arm and Wrist Positioning Task



b. Task 2: Wrist Tracking Task

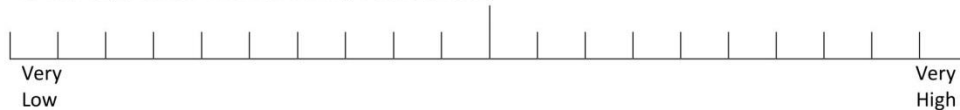


c. Task 3: Hand Shaking Task



Control Cognitive Demand (Approach 1)

1. Mental Demand: How 'mentally demanding' was the prosthesis control? [Report in terms of the cognitive burden experienced when controlling the prosthesis.]

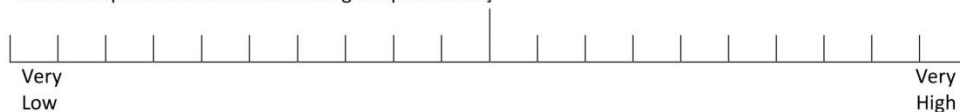


2. Frustration: How 'frustrating' was the prosthesis control? [Report in terms of how well the prosthesis responded to your intentions/commands]



Control Cognitive Demand (Approach 2)

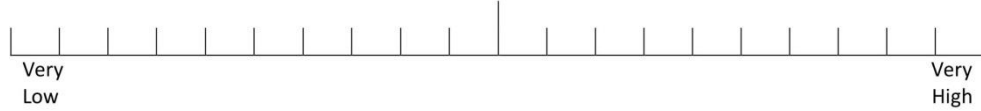
1. Mental Demand: How 'mentally demanding' was the prosthesis control? [Report in terms of the cognitive burden experienced when controlling the prosthesis.]



Context- and Physiology-aware Machine Learning for Upper-Limb Myocontrol

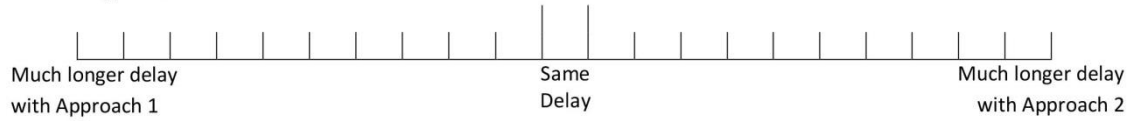
Subject Code:

2. Frustration: How 'frustrating' was the prosthesis control? [Report in terms of how well the prosthesis responded to your intentions/commands]

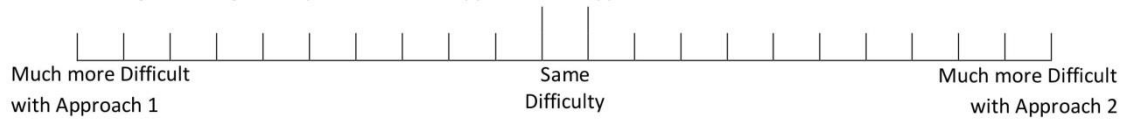


Control Comparison

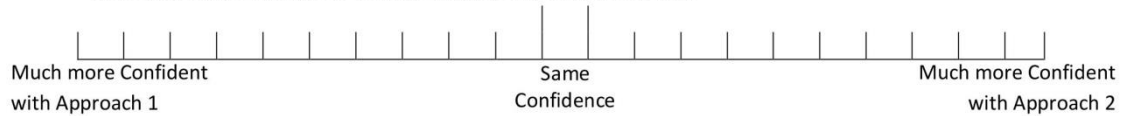
1. Compare the delay perceived when opening the hand. Was the delay before releasing the object longer in Approach 1 or 2?



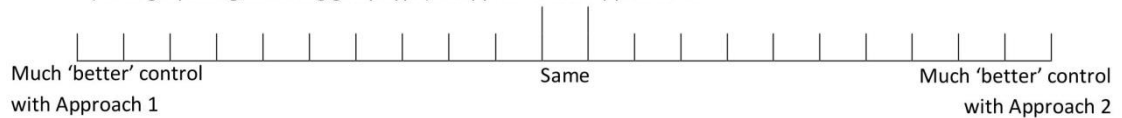
2. Compare the 'difficulty' in opening the hand when trying to release the object. Did you have to contract stronger (or longer) to open the hand in Approach 1 or Approach 2?



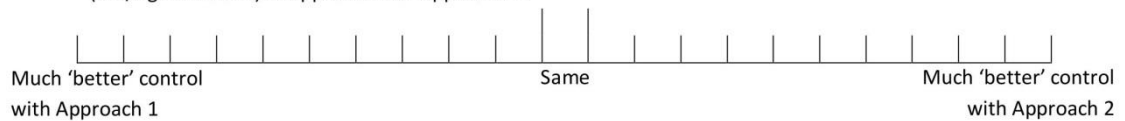
3. Compare the confidence that you will not lose the object during the Test(s) with different Algorithms. Were there more unwanted openings in Approach 1 or Approach 2?



4. Compare the overall Grasping performance. Were you more successful in controlling grasping (closing/opening/selecting grasp type) in Approach 1 or Approach 2?



5. Compare the overall Wrist Rotation performance. Were you more successful in controlling the wrist (left/right rotation) in Approach 1 or Approach 2?



APPENDIX 2: DISSEMINATION

As a part of this PhD project, the author gained 24 credit points through course work, teaching activities and journal/poster contributions. A summary of this work is as follows:

Journal/Poster Contributions	Credits
Accepted Journal Paper: "Exploiting Knowledge Composition to Improve Real-Life Hand Prosthetic Control" in IEEE Transactions on Neural Systems and Rehabilitation Engineering,	3
Accepted Journal Paper: "Context-dependent adaptation improves robustness of myoelectric control for upper-limb prostheses", in Journal of Neural Engineering.	3
Poster Presentation: Context Aware Regression for Myoelectric Control	3
Key Qualifications/Teaching Activities	
Seminar on Key Competencies	3
Supervision of Practical Course on Data Fusion	4
Research Project Supervision for Mr. Mahmoud Al Zaitoun	3
Courses	
Simulation-based Data Fusion and Analysis	5
Total	24

The Curriculum Vitae of the author is attached below,

Gauravkumar PATEL

CURRICULUM VITAE



+ PERSONAL DETAILS

Address Arndtstraße 14 (Zr. 19), Göttingen 37075, DE
Telephone ☎ +49 176 7566 6380
Email ✉ patelgauravk@gmail.com, gauravkumar.patel@bccn.uni-goettingen.de
Nationality Indian
Date of birth May 08, 1991
Family Status Married (to Mrs. Pooja Patel)

+ EDUCATION

March 2016-
Now **Ph.D. in Computer Science** at Georg August University, Göttingen, DE
Title: *Context and Physiology Aware Machine Learning for Upper-Limb Myocontrol*
October 2012-
October 2015 **MSc. in Communication Engineering** at RWTH Aachen University, Aachen, DE
Grade: 1.7 (in German Grading Scale)
This program covered courses related to Image and Signal Processing, Machine Learning and Information Theory.
July 2008-
May 2012 **BTech. in Electronics & Communication** at Nirma University, Ahmedabad, IN
Grade: 8.26/10
This program covered courses related to Embedded System Design, Electronic Circuits Design and Fundamentals of Telecommunication.

+ WORK EXPERIENCE

December 2015-
Now **Research Assistant** at the Department of Trauma Surgery, Orthopedics and Plastic Surgery, University Medical Center Göttingen, Göttingen, DE
As a part of my Ph.D. program, I work at the Medical Center to conduct experiments necessary to validate my hypothesis.
October 2014-
November 2015 **Internship** at the Institute of Robotics and Mechatronics, German Aerospace Center (DLR), Oberpfaffenhofen, DE
Internship on developing a software for closed-loop prosthesis control, mainly focusing on regression based movement prediction for prosthesis control.
March 2014-
September 2014 **Master Thesis** at the Institute für Nachrichtentechnik, RWTH Aachen, DE
Title: *Joint feature coding and pooling for content-based image retrieval*
The thesis focused on development and evaluation of algorithms for an image retrieval system that finds similar images in a known database.
April 2013-
October 2013 **Student job (HiWi)** at the Institute für Nachrichtentechnik, RWTH Aachen, DE
Part-time job on software development for Real Time Audio Processing and Effect Visualization using fast convolution and Fourier transform.
January 2012-
April 2012 **Internship** at Motorola Solutions, Bangalore, IN
Internship on developing multimedia communication systems for real-time video streaming, mainly focusing on server-client session initiation using IP protocols.

+ LIST OF PUBLICATIONS

- G. K. Patel**, S. Dosen, C. Castellini, D. Farina, "Multichannel electrotactile feedback for simultaneous and proportional myoelectric control", *Journal of Neural Engineering* 2016.
- G. K. Patel**, J. M. Hahne, C. Castellini, D. Farina, S. Dosen, "Context-dependent adaptation improves robustness of myoelectric control for upper-limb prostheses", *Journal of Neural Engineering* 2017.
- G. K. Patel**, M. Nowak, C. Castellini, "Exploiting knowledge composition to improve real-life hand prosthetic control", *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 2017.
- G. K. Patel**, C. Castellini, J. M. Hahne, D. Farina, S. Dosen, "A Novel Classification Method for Myoelectric Control of Hand Prostheses Inspired by Muscle Coordination", (submitted).
- I. Heisterklaus, **G. K. Patel**, C. Bulla, "Orthogonal transform of SIFT descriptors and the effect on size and performance of Fisher Vectors in visual search", *Image and Signal Processing and Analysis (ISPA), 9th International Symposium 2015, IEEE*.

+ TEACHING ACTIVITIES

- September 2017 **Practical Course on Data Fusion** was given at the University of Göttingen under the supervision of Prof. Marcus Baum. This course was taught using Python, and it was four weeks long.
- August 2017-
January 2018 **Research Project Supervision** for Mr. Mahmoud Al Zaitoun
Topic: "Machine Learning Algorithms for Resolving the Limb Position Effect".
- November 2017-
May 2018 **Master Thesis Supervision** for Mr. Pascal Hecker
Topic: "Muscle Coordination Inspired Biofeedback for Myoelectric Control".
- December 2017-
March 2018 **Research Project Supervision** for Mr. Martin Heinemann
Topic: "Android Framework for real-life Myocontrol".
- October 2017-
January 2018 **Internship Supervision** for Ms. Souzana Georgiadou
Topic: "Web-portal to train Amputees for Myoelectric Control".

+ SKILLS

- Languages **English** (C1 level), **German** (B1 level), **Hindi** (native), **Gujarati** (native)
- Hardware development Arduino, Atmega 8/16-bit, FPGA (basic)
- Programming **C#, C/C++, MATLAB, Python**
basic: Java for Android, HTML/CSS/JavaScript, VHDL, Cadsoft Eagle

+ OTHER ACTIVITIES

- Volunteering Activities Active volunteer at Shree Swaminarayan Gurukul, Germany Chapter. Every year, we organize social events highlighting India Culture.