REPUBLIC OF TÜRKİYE
YILDIZ TECHNICAL UNIVERSITY
GRADUATE SCHOOL OF SCIENCE AND ENGINEERING

DESIGN AND IMPLEMENTATION OF AN EMG DRIVEN UPPER LIMB EXOSKELETON

Berkem VURAL

MASTER OF SCIENCE THESIS
Department of Control and Automation Engineering
Control and Automation Engineering Program

Supervisor
Asst. Prof. Dr. Claudia F. YASAR

August, 2024

REPUBLIC OF TÜRKİYE
YILDIZ TECHNICAL UNIVERSITY
GRADUATE SCHOOL OF SCIENCE AND ENGINEERING

DESIGN AND IMPLEMENTATION OF AN EMG DRIVEN UPPER LIMB EXOSKELETON

A thesis submitted by Berkem VURAL in partial fulfillment of the requirements for the degree of **MASTER OF SCIENCE** is approved by the committee on 05.08.2024 in Department of Control and Automation Engineering, Control and Automation Engineering Program.

Asst. Prof. Dr. Claudia F. YASAR
Yildiz Technical University
Supervisor

Approved By the Examining Committee

Asst. Prof. Dr. Claudia F. YASAR, Supervisor
Yildiz Technical University _____

Assoc. Prof. Levent UCUN, Member
Yildiz Technical University _____

Assoc. Prof. Ilker USTOGLU, Member
Istanbul Technical University _____

I hereby declare that I have obtained the required legal permissions during data collection and exploitation procedures, that I have made the in-text citations and cited the references properly, that I haven't falsified and/or fabricated research data and results of the study and that I have abided by the principles of the scientific research and ethics during my Thesis Study under the title of Design and Implementation of an EMG Driven Upper Limb Exoskeleton supervised by my supervisor, Asst. Prof. Dr. Claudia F. YASAR. In the case of a discovery of false statement, I am to acknowledge any legal consequence.

Berkem VURAL

Signature

Dedicated to those who made it possible:
Ataturk
and the pioneers of science

ACKNOWLEDGEMENTS

First and foremost, I am deeply indebted to my supervisor, Asst. Prof. Claudia F. YASAR. She taught me, guided me, and always supported me. She enabled me to take firm steps forward with her valuable guidance at every opportunity. I am grateful to her for what she has given me.

Words cannot express my gratitude to my colleagues, two outstanding PhD candidates, and researchers from the YTU Control and Automation Engineering Department; Ammar Urgan and Kadircan Kurtulus. Without them, I would not have found my way with the scientific blindness that I faced during this thesis. I could not have overcome the deep problems in realizing the project without their help. At this point, I would like to express my gratitude to Berke Apaydinli for his important contribution.

I would like to express my deepest appreciation to my family, Their support and contribution to my academic studies since my earliest childhood are matchless.

I am also thankful to all my friends whose names I cannot list here, who were tired of listening to the updates of this process from me, shared my excitement, and supported me.

Finally, I would like to extend my sincere thanks to the founder of the republic, a great revolutioner Ataturk. I have studied and will carry on studying under his guidance. As he said once: "Our true mentor in life is science."

Berkem VURAL

TABLE OF CONTENTS

LIST OF SYMBOLS vii

LIST OF ABBREVIATIONS ix

LIST OF FIGURES xi

LIST OF TABLES xiv

ABSTRACT xv

ÖZET xvii

1 INTRODUCTION 1

1.1 Motivation 1

1.2 Literature Review 2

1.3 Objective of the Thesis 5

1.4 Hypothesis 6

2 PHYSIOLOGICAL BACKGROUND 7

2.1 Pathophysiology of Upper Limb Muscle Disorders 9

2.2 Anthropometry of Upper Limb 10

3 EMG 12

3.1 Action Potential 12

3.2 Muscle Contraction Mechanism 13

3.3 An EMG Signal: Acquisition and Its Features 15

3.4 EMG Pre-processing 17

3.5 EMG Classification By The Neural Network 26

3.5.1 The Levenberg-Marquardt Algorithm 30

3.6 From EMG to Muscle Force and Torque 37

3.6.1 Hill's Muscle Model 38

3.7 EMG Acquisition Procedure 42

4 CONTROLLER DESIGN 44

4.1 Mathematical Model of the Robot Dynamics 44

4.2 Computed Torque Control 47
4.3 Model Reference Adaptive Control 56

5 IMPLEMENTATION **67**
5.1 Mechanical Design and Electronic Components 67
5.1.1 CAD of the Exosuit and Material Selection 67
5.1.2 Strength Test Simulation and Production 69
5.1.3 Torque Requirements and Actuator-Transmission System . . 71
5.1.4 Microcontroller Board and Its Software 73
5.1.5 EMG Signal Acquisition Components 74

6 CONCLUSION **76**

REFERENCES **78**

A EQUATIONS **83**
A.1 Computed Torque Control 83
A.2 Model Reference Adaptive Control 84

PUBLICATIONS FROM THE THESIS **87**

LIST OF SYMBOLS

B	Bias value vector
A	Target function
W	Weights of the first layer
V	Weights of thelast layer
k	Iteration counter
J	Jacobian matrix
μ	Marquardt constant
a_j	Activation of muscle
u_j	Postprocessed EMG of muscle
A	Nonlinear shape factor
ϕ	Pennation angle
e	Joint position error vector
$\dot{e}$	Joint velocity error vector
$\ddot{e}$	Joint acceleration error vector
I	Identity matrix
M	Inertia matrix
C	Coriolis and Centripetal Forces matrix
q_d	The desired joint rotation vector
$\dot{q_d}$	The joint velocity vector
$\ddot{q_d}$	The joint acceleration vector
q	The joint rotation vector
$\dot{q}$	The joint velocity vector
$\ddot{q}$	The joint acceleration vector

G	Gravity matrix
L	The Lagrangian
T	Kinetic energy
V	Potential energy
K_p	Proportional gain
K_d	Derivative gain
V	Lyapunov function
Y	Regressor matrix
s	Error vector
W_M	Weighting matrix of inertia
W_C	Weighting matrix of Coriolis
W_G	Weighting matrix of gravity
Q_M	Positive definite weighting matrix of inertia
Q_C	Positive definite weighting matrix of Coriolis
Q_G	Positive definite weighting matrix of gravity
Z_M	Basis function matrix of inertia
Z_C	Basis function matrix of Coriolis
Z_G	Basis function matrix of gravity

LIST OF ABBREVIATIONS

ACL	Acrylonitrile Butadiene Styrene
ADC	Analog-to-Digital Converter
ANN	Artificial Neural Network
AUC	Area Under Curve
BP	Backpropagation
CAD	Computer-Aided Design
CE	Contractile Element
CMRR	Common Mode Rejection Ratio
DOF	Degree(s) of freedom
EMG	Electromyography
FAT	Function Approximation Technique
FPR	False Positive Rate
FT	Fourier Transform
ISE	Integral of Squared Error
LM	Levenberg-Marquardt
MLP	Multilayer Perceptron
MRAC	Model Reference Adaptive Control
MSE	Mean Squared Error
MSK	Musculoskeletal
MU	Motor Unit
MUAP	Motor Unit Action Potential
PD	Proportional-Derivative
PID	Proportional-Integral-Derivative

PE	Parallel Elastic Element
PLA	Polylactic Acid
RMS	Root Mean Square
ROC	Receiver Operating Characteristic
SE	Series Elastic Element
sEMG	Surface electrode acquired EMG
SNR	Signal-to-Noise Ratio
TKEO	Teager-Kaiser Energy Operator
TPR	True Positive Rate
ULMD	Upper Limb Musculoskeletal Disorder

LIST OF FIGURES

Figure 1.1 The overview of a intelligent exoskeleton system 3
Figure 1.2 A rehabilitation exosuit and overview of its control system . . . 4
Figure 2.1 Anatomical nomenclature for the upper limb 8
Figure 2.2 Musculoskeletal system elements of the upper 9
Figure 2.3 A stick figure in which the lengths are shown as a percentage of stature 10
Figure 3.1 Generation of an action potential 13
Figure 3.2 Generation of the muscle contraction 14
Figure 3.3 A motor unit and muscle fibers 15
Figure 3.4 Healthy and pathophysiologic EMG signals 16
Figure 3.5 Examples of needle and surface electrode used EMG signals in milliseconds 16
Figure 3.6 Surface electrodes for EMG acquisition 17
Figure 3.7 Preprocessing steps for an EMG signal 18
Figure 3.8 Filtering process of the signal 20
Figure 3.9 The RMS of the healthy EMG signal 21
Figure 3.10 The RMS of all types of the EMG signals 22
Figure 3.11 Local peak values for healthy EMG 23
Figure 3.12 Local peak values for myopathic EMG 23
Figure 3.13 Local peaks values for neuropathic EMG 24
Figure 3.14 Interpolation results of each signal 24
Figure 3.15 The axes of the human arm movements 25
Figure 3.16 The simulation results on tremor detection 26
Figure 3.17 The fundamental parts of the neural networks 27
Figure 3.18 An example of an MLP structure 28
Figure 3.19 The flowchart of BP algorithm 29
Figure 3.20 The flowchart of LM algorithm 31
Figure 3.21 The confusion matrix for the training for 150 epochs, 6 hidden layers 34
Figure 3.22 Performance metrics of the NN validation 35
Figure 3.23 Error histogram of the best training sessions 36
Figure 3.24 The ROC of the selected hidden layer-epochs training process . 37

Figure 3.25 The Hill muscle model 38
Figure 3.26 The Hill muscle model forces 38
Figure 3.27 Torque estimations for healthy and myopathy cases for elbow flexion-extension scenario 41
Figure 4.1 The double pendulum model representation of the robotic arm . 45
Figure 4.2 The block diagram of computed torque control 47
Figure 4.3 Torque values for each joint 50
Figure 4.4 Step response of the shoulder rotation for different cases 51
Figure 4.5 Rotation angles for each joint by the PD-computed torque controller 53
Figure 4.6 Error values in rotation angles for each joint for the PD-computed torque controller 54
Figure 4.7 The increased frequency mode input on the elbow joint and the output of the model 55
Figure 4.8 The increased frequency mode error value of the elbow joint path tracking 55
Figure 4.9 The block diagram of MRAC 57
Figure 4.10 The block diagram of FAT-based adaptive controller 62
Figure 4.11 Torque values for each joint by the adaptive controller 63
Figure 4.12 Rotation angles for each joint by the adaptive controller 64
Figure 4.13 Error values in rotation angles for each joint for the adaptive controller case 65
Figure 4.14 The increased frequency mode on the elbow joint and the output of the model for the adaptive controller case 66
Figure 4.15 The increased frequency mode error on the elbow joint for the adaptive controller case 66
Figure 5.1 The CAD parts of the system 68
Figure 5.2 The material properties window on the CAD software 68
Figure 5.3 Overview of the CAD 69
Figure 5.4 Strength test of the CAD 70
Figure 5.5 The produced shoulder complex 70
Figure 5.6 The produced upper arm 71
Figure 5.7 The produced lower arm 71
Figure 5.8 The stepper motor 72
Figure 5.9 A4988 stepper motor driver 72
Figure 5.10 Gears for shoulder and elbow joint shafts 73
Figure 5.11 The data acquisition board and its software 73
Figure 5.12 EMG signal acquisition components 74
Figure 5.13 Electrode placement on the upper limb 75

Figure 5.14 EMG acquisition experiments 75

LIST OF TABLES

Table 2.1 Range of motion for each joint of the upper limb 8
Table 3.1 Parameters of the Butterworth High Pass Filter 19
Table 3.2 F1 scores for different classes 32
Table 3.3 Accuracy percentages of different trials of the training by LM . . 33
Table 3.4 MSE values for the training process 37
Table 3.5 Parameters used in the Hill-type muscle model for torque estimation from processed EMG signals. 41
Table 4.1 Parameters of the exoskeleton for computed torque control simulations 49
Table 4.2 Data points for step response ISE calculation 52
Table 4.3 ISE values for elbow movement under the adaptive controller . . 65

ABSTRACT

Design and Implementation of an EMG Driven Upper Limb Exoskeleton

Berkem VURAL

Department of Control and Automation Engineering
Master of Science Thesis

Supervisor: Asst. Prof. Dr. Claudia F. YASAR

In this thesis project, a fully integrated design study on designing an upper limb exoskeleton for the rehabilitation of a physiologically problematic arm is presented. It is aimed to diagnose and classify the selected and a priori-determined conditions. The design involves an Electromyography (EMG) start and with operating novel signal processing steps, the classification of such muscle signals is focused on the diagnosis. Through the Levenberg–Marquardt algorithm training method-based Artificial Neural Network (ANN), the design is achieved by determining the operation phase for the rehabilitation process. According to Hill's muscle model based required torque development by the processed surface EMG signal is proposed then the system is operated well by an adjustable unique adaptive controller.

In the signal processing part, contrary to general methods in the literature, the Teager-Kaiser method is placed at the center of these steps. This approach ensures the minimum loss of usable data on the EMG signal and provides much more information for conversion of the signal data into a more deterministic nature. This more deterministic nature is supported by the proposed method of peak selection on the data process. Then, an ANN is well-trained to classify these signals and specify the difference among neuropathic, myopathic, and healthy muscle signals. This determination is realized in the flexion-extension movement of the sagittal plane anatomically. As one of the design limits, coronal plane movements, namely abduction-adduction movement are obtained passively; however, according to the Fourier transform-based frequency analysis, the tremor of Parkinson's disease can

be diagnosed by the proposed signal processing method without EMG.

Another contribution of this research is the proposed Function Approximation Technique (FAT) based adaptive controller. Commercial upper limb exoskeletons and generally industrial robotic arms are designed by the usage of traditional control techniques such as PID controller, PD-based computed torque controller, etc. In this study, the comparative insight is depicted between PD based computed torque controller and a novel designed adaptive controller. Due to the defects of the computed torque controller on the path tracking which is crucial for the rehabilitation process, a proposed Fourier series derived function approximation technique operated adaptive controller, better performance in terms of the trajectory planning is obtained. The controller can predict the essential parameters of the robotic arm without the usage of regressor matrix which also generally occurs among the adaptive controller approaches in the literature. This novelty makes the design unique and adjustable for the different anthropometric data of the different individuals.

Mechanical design is prepared via 3D printed parts which can be produced easily, quickly, cheaply, and user-oriented. Therefore, the system can be used for any person who suffers the muscle problems of any sociocultural origin.

In a nutshell, this thesis project presents a comprehensive approach to the design and implementation of an upper limb exoskeleton aimed at effective rehabilitation through advanced EMG signal processing and innovative control strategies. The signal processing, combined with the robust classification capabilities of the ANN, allows for precise diagnosis of muscle conditions. Furthermore, the FAT-based adaptive controller demonstrates superior performance in trajectory planning and adaptability to individual anatomical variations. These contributions highlight the potential for significant advancements in rehabilitation technology, offering a promising solution for personalized and effective therapy for individuals with upper limb impairments.

Keywords: Upper limb exoskeleton, EMG, adaptive control, computed torque control, neural networks.

ÖZET

EMG Tahrikli Üst Uzuv Dış İskelet Robot Tasarımı ve Uygulaması

Berkem VURAL

Kontrol ve Otomasyon Mühendisliği Anabilim Dalı
Yüksek Lisans Tezi

Danışman: Dr.Öğr.Üyesi Claudia F. YAŞAR

Bu tez projesinde, fizyolojik olarak sorunlu bir kolun rehabilitasyonu için bir üst ekstremite dış iskeleti tasarlamaya yönelik tam entegre bir tasarım çalışması sunulmaktadır. Seçilen ve önceden belirlenmiş koşulları teşhis etmek ve sınıflandırmak amaçlanmaktadır. Tasarım, bir Elektromiyografi (EMG) sinyali başlangıcını içerir ve önerilmiş özgün sinyal işleme adımlarını çalıştırarak, bu tür kas sinyallerinin sınıflandırılması ve tanıya odaklanır. Levenberg-Marquardt algoritması eğitim yöntemi tabanlı Yapay Sinir Ağı (YSA) aracılığıyla, tasarım rehabilitasyon süreci için operasyon fazını belirleyerek elde edilir. Hill kas modeline göre, işlenmiş yüzey EMG sinyali tarafından gerekli tork gelişimi önerilir, ardından sistem ayarlanabilir benzersiz bir adaptif denetleyici tarafından iyi bir şekilde çalıştırılır.

Sinyal işleme bölümünde, literatürdeki genel yöntemlerin aksine, Teager-Kaiser yöntemi bu adımların merkezine yerleştirilir. Bu yaklaşım, EMG sinyalinde kullanılabilir verilerin minimum kaybını sağlar ve sinyal verilerinin daha kesin bir yapıya dönüştürülmesi için çok daha fazla bilgi sağlar. Bu kesin yapı, veri işleminde önerilen tepe seçimi yöntemi tarafından desteklenir. Daha sonra, bir YSA bu sinyalleri sınıflandırmak ve nöropatik, miyopatik ve sağlıklı kas sinyalleri arasındaki farkı belirlemek için efektif biçimde eğitilir. Bu belirleme, sagital düzlemin anatomik olarak fleksiyon-ekstansiyon hareketi üzerinde gerçekleştirilir. Tasarım sınırlarından biri olarak, koronal düzlem hareketleri, yani abdüksiyon-adüksiyon hareketi pasif olarak elde edilir; ancak, Fourier dönüşümü tabanlı frekans analizine göre, Parkinson hastalığının titremesi,

EMG olmadan da önerilen sinyal işleme yöntemi ile teşhis edilebilir.

Bu araştırmanın bir diğer katkısı, önerilen Fonksiyon Yaklaşım Tekniği (FAT) tabanlı adaptif denetleyicidir. Ticari üst ekstremite dış iskeletleri ve genellikle endüstriyel robotik kollar, PID denetleyici, PD tabanlı hesaplanmış tork denetleyici vb. gibi geleneksel kontrol tekniklerinin kullanımıyla tasarlanır. Bu çalışmada, PD tabanlı hesaplanmış tork denetleyici ile yeni tasarlanmış adaptif denetleyici arasındaki karşılaştırmalı içgörü tasvir edilmiştir. Rehabilitasyon süreci için kritik öneme sahip olan yol takibindeki hesaplanan tork kontrolcüsünün kusurları nedeniyle, önerilen bir Fourier serisi türetilmiş fonksiyon yaklaşım tekniği ile çalıştırılan adaptif kontrolcü, yörünge planlaması açısından daha iyi performans elde eder. Kontrolcü, literatürdeki adaptif kontrolcü yaklaşımları arasında genellikle görülen regresör matrisinin kullanımı olmadan robotik kolun temel parametrelerini tahmin edebilir. Bu yenilik, tasarımı farklı bireylerin farklı antropometrik verileri için benzersiz ve ayarlanabilir hale getirir.

Mekanik tasarım, kolay, hızlı, ucuz ve kullanıcı odaklı olarak üretilebilen 3D yazdırılmış parçalar aracılığıyla hazırlanır. Bu nedenle, sistem herhangi bir sosyokültürel kökenli kas problemi çeken herhangi bir kişi için kullanılabilir.

Özetle, bu tez projesi, gelişmiş EMG sinyal işleme ve yenilikçi kontrol stratejileri yoluyla etkili rehabilitasyonu amaçlayan bir üst ekstremite dış iskeletinin tasarımı ve uygulanmasına yönelik kapsamlı bir yaklaşım sunmaktadır. Sinyal işleme, ANN'nin sağlam sınıflandırma yetenekleriyle birleştiğinde, kas rahatsızlıklarının kesin teşhisine olanak tanır. Ayrıca, FAT tabanlı uyarlanabilir kontrolör, yörünge planlama ve bireysel anatomik varyasyonlara uyum sağlamada üstün performans göstermektedir. Bu katkılar, üst ekstremite bozuklukları olan bireyler için kişiselleştirilmiş ve etkili terapi için umut verici bir çözüm sunarak rehabilitasyon teknolojisinde önemli ilerlemeler için potansiyeli vurgulamaktadır.

Anahtar Kelimeler: Üst ekstremite dış iskelet robot, EMG, adaptif kontrol, hesaplamalı tork kontrolü, sinir ağları.

1
INTRODUCTION

In recent years, engineers have been practically proposing many convenience technologies to facilitate daily life. To achieve this, interdisciplinary research fields have been paced such as biomechatronics. Biomechatronics integrates robotics-related utilizations with biology, mechanics, electronics, and control engineering study fields which cover a wide range of applications from neurology to cardiovascular, respiratory, and musculoskeletal systems. To enhance the musculoskeletal system, exoskeletons have been an important research and development area for many years. This technology has emerged as a promising solution for both rehabilitation and assistive purposes in the field of biomechatronics. These wearable robotics outcomes have been designed to be used as an auxiliary device for movement and/or regaining human physical abilities. In this study, a unique design and implementation of an electromyography (EMG)-driven upper limb exoskeleton has been presented to contribute to this main goal.

1.1 Motivation

Upper limb musculoskeletal disorders (ULMSDs) are widespread and expensive to overcome issues during daily life, causing significant health and financial burdens that involve the nerves, tendons, cartilage, ligaments, joints, and muscles on the upper limbs. ULMSDs can mainly occur due to work-related problems such as repetitive motion, static loading, problematic workstation designs, etc. Since one of the most mobilized parts of the body is the upper limbs, their disorder directly affects the efficiency of the employee. According to 2021 statistics, almost 45% of work-related musculoskeletal disorders have been reported as related to the upper limb in Great Britain [1]. To prevent such disorders, assistive robotic technologies have been developed since last century. While they reduce the load, they facilitate difficult positional movements for the employee.

Moreover, stroke is considered a major global health concern due to its prevalence and disabling effects worldwide. Aging and uncertain lifestyle with urbanization are two major causes of stroke. Recent statistics indicate that approximately 650 million individuals are aged 60 or older globally, with projections estimating this number to rise to 2 billion by 2050 [2]. After a stroke, efforts in rehabilitation are moving towards creating more holistic methods for muscle training to prevent lasting muscle problems. Robot-assisted stroke rehabilitation is showing promise in this area, offering hope for better outcomes. Physiotherapy is still the main treatment for such disabilities, this process consists of hands-on activities to relearn independent movement abilities by a physiotherapist. However, it is costly and these services are generally not able to provide adequate and timely treatment. Rehabilitation robots have the potential to overcome the limitations of conventional rehabilitation methods which can be counted as traveling difficulties, subjectivity, and limited complexity of the rehabilitation [3].

A tremor is when your body shakes uncontrollably, usually in a rhythmic way. It's like a shaking that happens without you wanting it to. It is a common symptom seen in conditions like Parkinsonism, where it is often one of the main signs that someone might have the condition. In accordance with the statistics, the overall prevalence rate of Parkinsonian syndrome has been estimated to be about 100 to 150 per 100000 population with an incidence of 20 cases per 100000 per year [4]. Therefore, it can be accepted as an important obstacle for upper limb mobilization.

Finally, it can be said that both assistive and rehabilitation modes including exoskeleton robots can overcome many problems according to ULMSDs, and cancel the tremors according to Parkinson's. A compact design with tremor-canceling can be an efficient solution in many aspects.

1.2 Literature Review

Upper limb exoskeletons are biomechatronics systems that are implemented to assist, gain power amplification, and/or help, rehabilitation of lost motor function individuals. They can be classified in many different manners such as their actuation and control mechanisms, the materials used for their designs, their application areas, their degrees of freedom, and so on.

Assistive exoskeletons are designed to enhance the power and ability of individuals for tasks related to everyday life, employment in industrial processes, or military service. They can be classified as soft or rigid robotics or dependent on the specific location in terms of assistance like elbow, wrist, or hand [5–7]. It can be said that,

mainly, assistive exoskeletons are used for post-stroke therapy assistance. As it is known, on average, one in four adults aged 25 and above will experience a stroke during their lifetime, with approximately 12.2 million people worldwide suffering from a stroke each year. This leads to neuromotor disorders in 20–40% of stroke survivors [8]. For instance, to assist this kind of target patients, Lee et al. [8] introduced an upper limb exoskeleton system which has a deep learning applied system to diagnose the intention of movement. The decision-making mechanism of the system predicted joint motions for several different movements with high accuracy. They also proposed a novel embedded sensory to get feedback and collect real-time muscle activities through EMG signals [8].

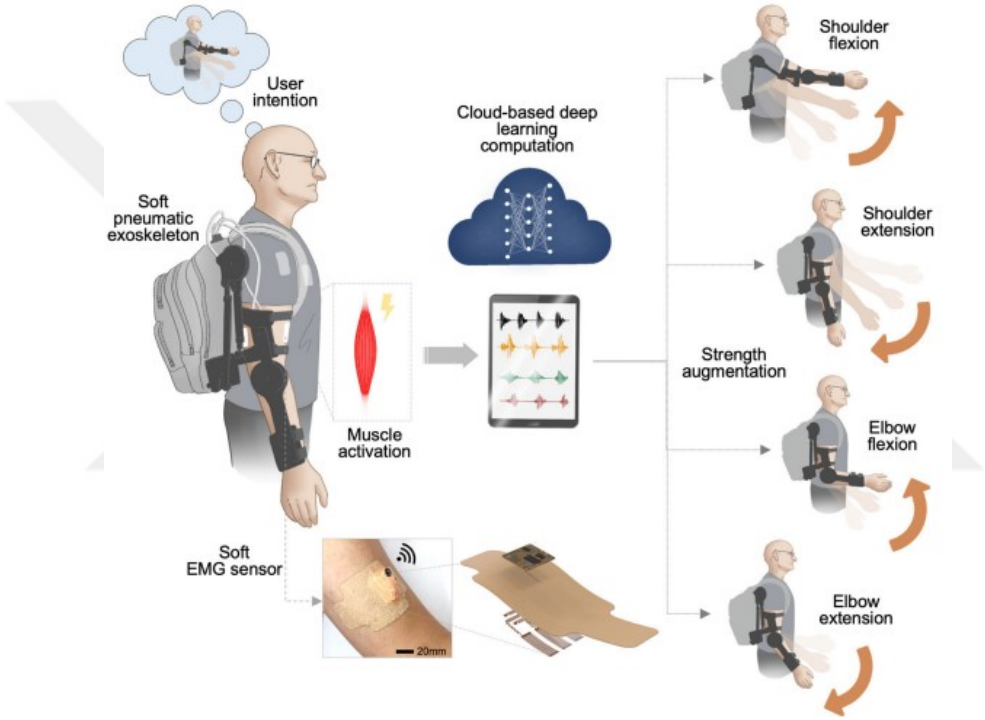


Figure 1.1 The overview of a intelligent exoskeleton system [8]

Besides active actuator-operated assistive exoskeletons, there are also passive ones. These exoskeletons rely on mechanical structures, springs, or other passive elements to provide support and assistance without active power sources. Generally, this kind of passive exoskeletons is used to get gravity compensation, within a reliable design that contains elastic materials [9, 10].

Furthermore, rehabilitation robots are created to aid individuals in their recovery from injuries or surgeries through physical therapy and rehabilitation purposes. Even though the individual has lost her/his upper limb mobility, a zero-to-one approach can be applied to such exoskeletons. This means that muscles can be thought to operate again according to repeated rehabilitation sessions within

the exoskeletons. There are many different approaches to control upper limb rehabilitation targeted exoskeletons such as fuzzy logic, adaptive, model predictive, impedance control and so on[11–13]. As mentioned before, there are many valid approaches to control such exoskeletons, one of them being biofeedback control. It broadly gets an EMG signal from the individual to modulate the assistance by the exoskeleton. Actually, to control upper limb exoskeletons by means of the acquisition of EMG signals, there are some main myoelectric control schemes. Those are: on-off control by setting a threshold value which is determined by the EMG signal mean absolute value or root square mean value, proportional control which is based on the speed of the terminal device becomes proportional to the levels of EMG signals, direct control from the nerves, finite-state machine control and pattern recognition based control [14]. For example, Lotti et al.[15] have created a system that is controlled in real time. The system has a main controller called the "myoprocessor" which figures out how much force is needed at the joints by analyzing EMG signals. Another controller, operating at a lower level, gives the required support to the joints based on this information. The control scheme of the system can be described as it follows: the myoprocessor calculates accurate torque profiles based on the muscle activity around the elbow joint, then an admittance controller, comprising an outer torque loop and an inner velocity loop tracks these torque profiles. The myoprocessor considers both the geometric and dynamic properties of musculotendinous units to anticipate the overall elbow flexion-extension torque. This anticipation is according to the muscle activation data obtained from surface EMG electrodes. The torque estimated by the myoprocessor is compared to the actual torque applied by the robot to the human joint. The system and the general scheme can be seen below.

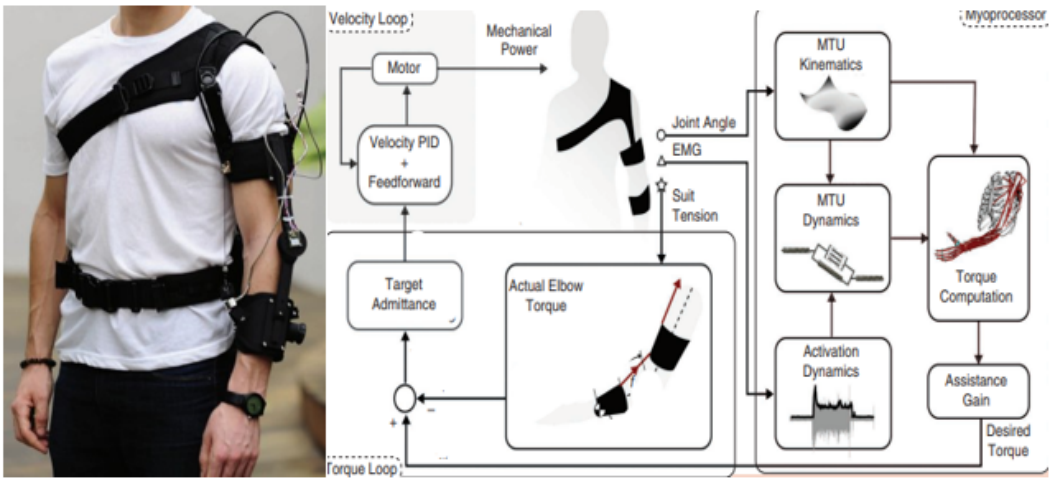


Figure 1.2 A rehabilitation exosuit and overview of its control system [15]

Due to the nonlinear nature of the human arm, the conventional linear control

approaches give some drawbacks in practical usage. Unknown uncertainties and modeling errors must be taken into account in the design of the robotic controller. Regretfully, conventional controllers are not optimal as they do not possess the self-adjustment capability to handle variations in parameters, such as individual variations in the mass of the human upper limb. Nonlinear adaptive control techniques have made significant progress in managing nonlinearities and uncertainties in a variety of nonlinear systems, including robotic systems [16]. In the presence of uncertainties, the adaptive control technique has a fair chance of providing a solution to ensure consistently high dynamic tracking performance[17–20]. Since such applications are in their infancy, adaptive controller usage on the upper limb exoskeleton systems can be counted as a new attractive field for research. One of the brand-new robot designs by Toro-Ossaba et. al [21], their design focuses on developing a myoelectric Model Reference Adaptive Controller (MRAC) integrated with an Adaptive Kalman Filter to control a cable-driven soft elbow exoskeleton. The MRAC controller proved effective for both passive and active control modes. The MRAC controller was robust enough to handle uncertainties in the exoskeleton and external disturbances. Additionally, the adaptive scheme allowed the system to function with only two surface EMG channels and joint angle measurements estimated using two Inertial Measurement Units. Although usage of the Kalman filter gives more accurate predictions of the rotation angles of the system, it can cause more process unit load and for the practical usage of the microcontrollers, required discretization of the MRAC with it might cause some problems.

1.3 Objective of the Thesis

The objective of the thesis is to design and implement an EMG signal-driven upper limb exoskeleton for the rehabilitation for whom the upper limb system is problematic due to the selective two disease groups, namely myopathy, and neuropathy, and also develop assistive support for the individuals who suffer from the upper limb disorders such as Parkinson's and fully non mobilized conditions like stroke. To achieve this main objective, developing a mechanically adjustable system for various anthropometric data and constructing a stable, robust, and adaptive controller system should be achieved. Developing a decision-making system based on an artificial neural network to interpret the condition of the individual through signal processing of the surface EMG data has been also targeted. The project also seeks to build signal-processing steps for more rapid and accurate signal conditioning for the stochastic and low-amplitude signal nature such as EMG has.

1.4 Hypothesis

The hypothesis of the thesis is tripedal. One of the hypotheses is adaptive and more stable controllers can be achieved with the model reference adaptive controller. Through the function approximation basis, the system does not require the regressor matrix as it exists in the common adaptive controller in the literature. This statement can be tested with comparisons with a conventional PD-based computed torque controller. Secondly, the thesis has the objective of using a neural network for making decisions. To do this, a more rapid and accurate training algorithm can be examined. Finally, the system diagnoses the problems and tries to cancel all the problems by all the proposed methods.

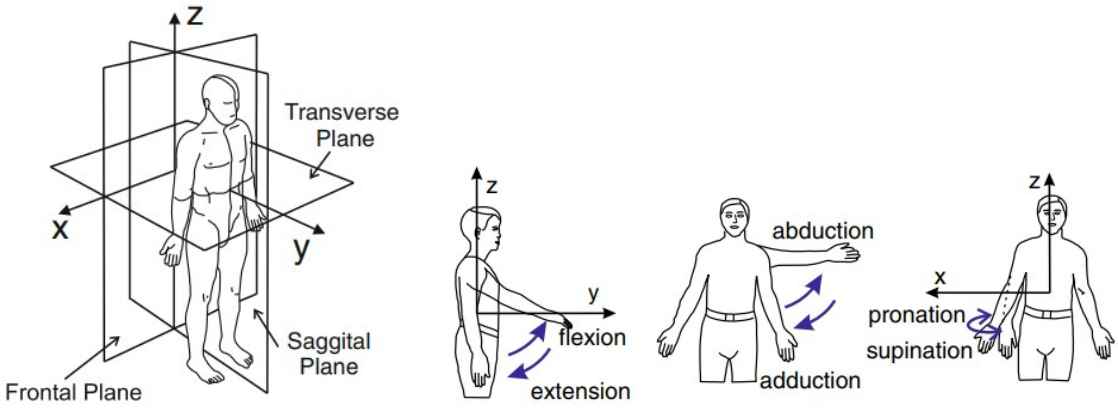
In this thesis second chapter aimed to give a brief introduction to the physiological information on the upper limb extremity. The third chapter depicts EMG signal processing, the main steps and simulation results have been picturized. The fourth chapter included all the information on the controller design of the system. The fifth one showed the Computer Aided Designed (CAD) model and implementation of the system in real-time applications. Finally, the results and conclusion have been prepared to give the results of the entire thesis and conclusion comments, respectively.

2
PHYSIOLOGICAL BACKGROUND

The musculoskeletal (MSK) system consists of muscles, bones, and connective tissues. The upper limb is a functional unit of the upper body, which is an important part of the MSK system. To indicate its movement scientifically, the nomenclature should be specified. The reference planes which are based on the standard anatomical position can be counted as the sagittal plane, frontal plane, and transverse plane. In the sagittal plane, flexion reduces the angle, and extension opposedly increases the angle between parts of the body. In the frontal plane, abduction is an outward movement of the arm or leg away from the median plane of the body, whereas adduction movement makes the parts closer to the mid axes of the body. Supination describes the rotation of the forearm. It occurs when the forearm rotates so that the palm position is anterior. Pronation is a rotation of the forearm that moves the palm from an anterior position to a posterior position (palm facing down) [22]. As it is known, the upper limb has three main joints in terms of rotation. The shoulder, elbow, and wrist joints have limited ranges of motion, these rotation angles can be seen in Table 1. The pronation and supination movements are intentionally not given since they are out of scope in the perspective of the design in this thesis project.

More specifically, the upper limb has a total of 9 degrees of freedom (DOF) excluding the finger joints. The glenohumeral joint at the shoulder complex is a ball and socket joint that allows the humerus to rotate about the glenohumeral head in 3 DOF which can be counted as shoulder flexion/extension, shoulder abduction/adduction and medial/lateral rotation. The shoulder joint has two more DOFs which are elevation/depression and retraction/protraction. More importantly for this thesis design, the elbow complex is capable of 2 DOF which are elbow flexion/extension and forearm pronation/supination. The wrist has also 2 DOFs and these are called wrist flexion/extension and radial/ulnar deviation [23].

In the human body, there are three main muscle types: skeletal, smooth, and cardiac. The skeletal muscle is the most abundant type of muscle in the body and is attached



(a) Anatomical planes **(b)** Movements of the upper limb

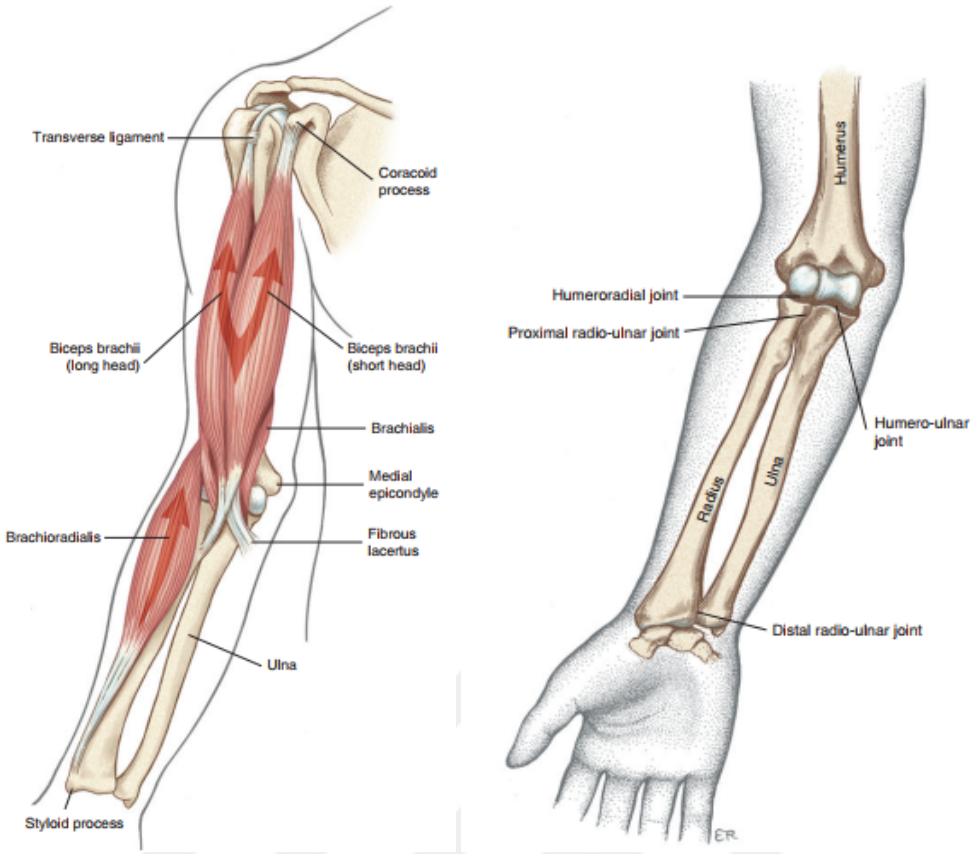
Figure 2.1 Anatomical nomenclature for the upper limb[22]

to bones by tendons, these muscles supply the ability of voluntary movement. The skeletal muscles that can be called primary elbow flexors are the biceps brachii, brachialis, brachioradialis, and pronator teres while triceps brachii and anconeus are elbow extensors[24].

Table 2.1 Range of motion for each joint of the upper limb [22]

Part	**Flexion**	**Extension**	**Abduction**	**Adduction**
Shoulder	-180°	+80°	-180°	+50°
Elbow	-10°	+145°	-	-
Wrist	-90°	+70°	-15°	+40°

Besides, to understand EMG signal behavior and the nature of the pathologic conditions, the muscle contraction mechanism should be mentioned here. The contraction mechanism starts with a signal from the nervous system. When a motor neuron releases neurotransmitters at the neuromuscular junction, it triggers an action potential in the muscle fiber. This electrical phenomenon triggers some cumulative steps in terms of chemical changes. Since the details of the process are out of scope, knowing that the contraction involves electrical and chemical processes is adequate. The behavior of an EMG signal and the acquisition of such signals will be given in more detail in one of the following subsections.



(a) Muscles **(b)** Osteology

Figure 2.2 Musculoskeletal system elements of the upper limb[24]

2.1 Pathophysiology of Upper Limb Muscle Disorders

Pathophysiology refers to the study of abnormal changes in body functions. Muscle disorders affecting the upper extremity bring about pathophysiological conditions. The previously mentioned Parkinson's disease can be classified in this area in that it causes dysfunctions due to tremors. Moreover, neuropathy and myopathy indicate other subdivisions of such diseases. Myopathy involves intrinsic muscle dysfunctions, which can be seen with a variety of symptoms, such as fatigue, cramping, discomfort, stiffness, and weakness in the muscles. The term neuropathy indicates the problems caused by nerve damage. Autoimmune disorders, physical injuries, and post-trauma conditions can be a cause of this kind of muscle disorder group. All over, the literature research claims that sustained muscle activity can be a leading factor in ULMDs.These disorders may involve changes in skeletal muscle structure, blood flow, and muscle function. The accumulation of calcium ions in the sarcoplasm is thought to contribute to muscle cell damage in such pathologies. Additionally, suboptimal blood flow is believed to be a credible factor in the development of these disorders [25]. Since all of them are related to the

electrical behavior of the drivers of the muscles, they can affect the EMG signal recordings differently. The different behaviors of neuropathic and myopathic EMG signals are interpreted in the specific subsection that is dedicated to the EMG signals and their features.

2.2 Anthropometry of Upper Limb

Anthropometry is the scientific study of the measurements and proportions of the human body. The engineering designs in biomechatronics must match the users' anthropometric data. Although the measurements of the body segments have a huge variety all around the world according to thousands of conditions such as geography, habitats, climate, etc., the anthropometric data can give a starting point for the beginning of the design steps. In Dempster and Gaughran's study [26], the bodies of average build cadavers have been examined and researchers revealed the proportions according to each body segment. The data are presented as mean values accompanied by standard deviations, and when applicable, they are compared and analyzed alongside other findings reported in the literature. According to their studies, while the length of the upper arm generally equals 0.1735 times the whole body height, the length of the lower arm is 0.1572 times the total length of the entire body.

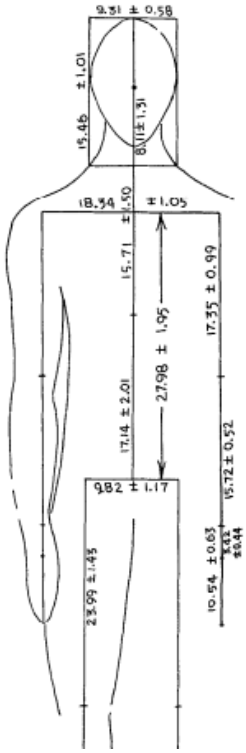


Figure 2.3 A stick figure in which the lengths are shown as a percentage of stature.[26]

Another important anthropometric data in terms of designing is the weights of the upper limb parts since users' weights directly affect the calculations of momentum and joint motor features. According to the same study [26], the proportion of upper arm weight has been found as almost 2.64% of the total body weight. The ratio for the forearm has also found as 1.531%.

Finally. the center of mass location of each part of the upper limb is also an important parameter. As already known, the required torque values for each joint can be analyzed by the mass and center of mass. In another study on the body segment parameters [27], normalized distances from proximal joint segment lengths

have been found as 0.427, 0.432, 0.417, and 0.402 for the right-left upper arm, and right-left forearm, respectively. As can be seen clearly, choosing the center of mass in the middle of the segment is acceptable from the perspective of engineering design.

Body height parameters vary around the world. This thesis project had to aim at a target group in terms of design assumptions and constraints for narrowing the scope. Therefore, the Europe region has been determined as the target place. According to the statistics, the average male height in European countries varies approximately from 185 cm to 172 cm [28]. Taking into account that female height values on average are shorter than male ones, the male maximums were considered for the design. However, the adjustable design of the exoskeleton is convenient for a wide range of anthropometric data.

3
EMG

Electromyography (EMG) is a technique for recording biomedical electrical signals obtained from neuromuscular activities. These electrical signals are used to diagnose physical abnormalities and activation levels and analyze the biomechanics of human movements [29]. To understand how to get EMG and how to analyze it, one should know the process of the electrical stimulation and contraction of the muscle first.

3.1 Action Potential

An action potential is a rapid change in the electrical potential across the cell membrane of a neuron or a myocyte which is the specific name for a muscle cell. The action potential is the mandatory process for the transmission of signals through the nervous system and the contraction of the muscles. It occurs through ion exchange in special voltage channels in the cell membrane. Ion channels are closed when the cell's membrane potential is around its resting potential. The rapid opening of the channels indicates that a potential difference above the threshold value has occurred in the cell. It occurs with the polarization change which is created by sodium and potassium ions in the cell membrane. When a cell receives a trigger signal, it begins a chain reaction. This starts with sodium ions rushing into the cell, making it more positive (depolarization) and creating an action potential. This electrical change then causes special channels in the membrane to open up, letting more sodium ions in, making the cell even more positive. When a certain level of positivity is reached, another channel opens, allowing potassium ions to leave the cell. This repolarization process brings the cell back to its resting state [30].

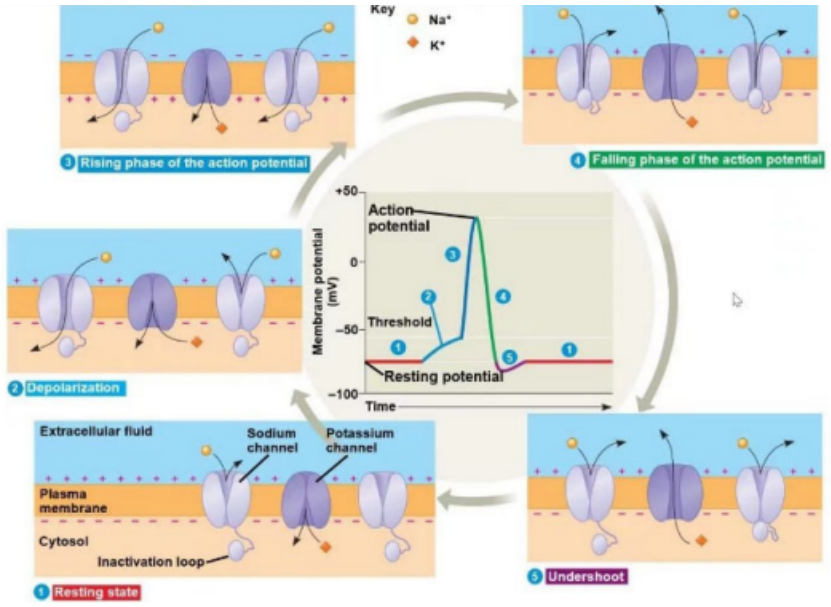


Figure 3.1 Generation of an action potential [31]

3.2 Muscle Contraction Mechanism

Since EMG measures the electrical activity which is essentially related to the action potential changes through the cell membrane during the muscle contraction, the contraction process should be mentioned. It begins with an action potential then the signal reaches the connection between the neuron and muscle fiber which is called neuromuscular junction. The reached action potential triggers to release of some of the chemicals that are neurotransmitter acetylcholine into the synaptic cleft and it causes the transmission of the action potential to the muscle fibers. Then, a chain of processes triggers the release of calcium ions in those cells. Finally, it activates some proteins that enable actin and myosin filaments to slide each other. So, the muscle fiber contracts when stimulated. The necessary stimulus comes via the motor nerve. The muscle also responds to a stimulus such as an electric current. Contraction of the muscle can be static (isometric), in which its length remains constant and only expands, or dynamic (isotonic), in which it both shortens and swells [30]. A general overview of the system can be seen in the following figure, details are out of the scope of the thesis.

The Motor Unit (MU) is the biological unit of muscle function. It is basically a motor neuron and all the muscle fibers it supplies; it has a motor nerve that originates from the central nervous system and is distributed to the motor end plates. Each of the end plates is connected to a muscle fiber, and when they are stimulated, the muscle fibers they are connected to are also stimulated. The number of MUs is different for muscles in various parts of the body. Generally, as the muscle

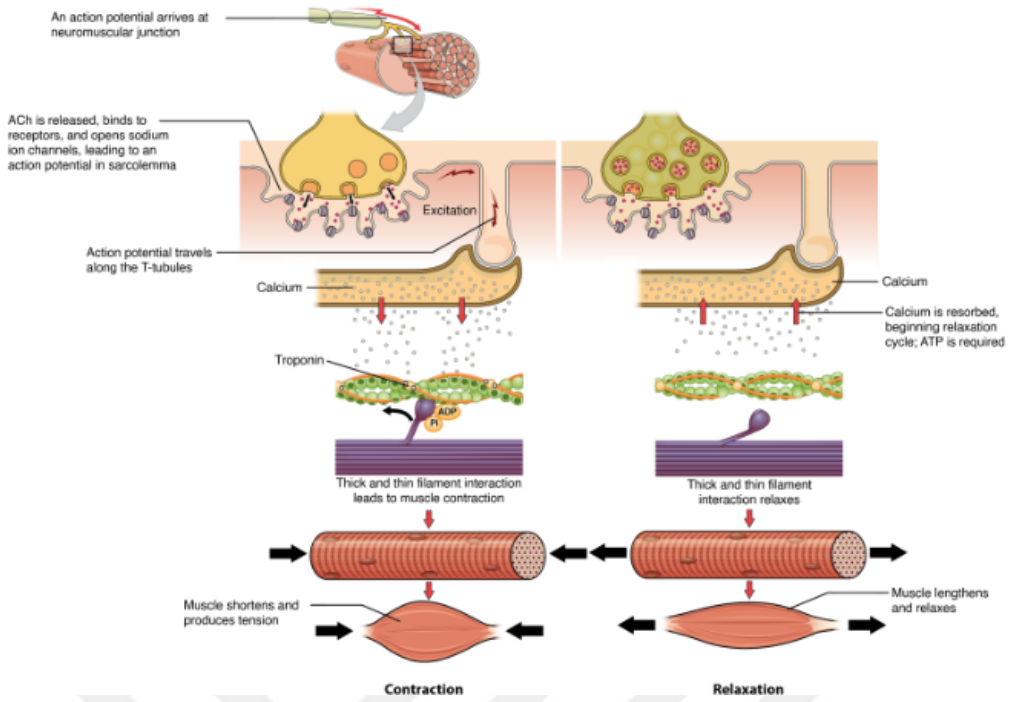


Figure 3.2 Generation of the muscle contraction [32]

grows, the number of MUs increases. It may contain more than 1000 muscle fibers [33]. The fibers forming the same unit are not gathered, but the fibers of various units are intertwined. Each motor nerve can be polarized or depolarized and sends "on-off" information to the motor end plates. Thus, each muscle fiber is either in a resting state or in an excited state. Several MUs become active in proportion to the magnitude of the muscle movement. As the muscle movement grows, many MUs become active and when all of them are active, the muscle movement is at its highest level. This ensures some smoothness in movement. Additional movement smoothness is achieved by modulating the number of muscle fibers stimulated. Although each MU can only give one level of contraction, the number of contractions will increase the strength of the muscle fibers. Thus, the smoothness of a muscle's movement is controlled by both the number of motor units stimulated and the rate of stimulation of these motor units [30].

The combination of the muscle fiber action potentials from all the muscle fibers of a single MU is the MU action potential (MUAP) which can be detected by a skin surface electrode (non-invasive) located near this field, or by a needle electrode (invasive) inserted in the muscle. So, finally, it can be said that an EMG signal is a collection of many MUAPs that have particular shapes that vary with the anatomical and physiological properties of muscles.

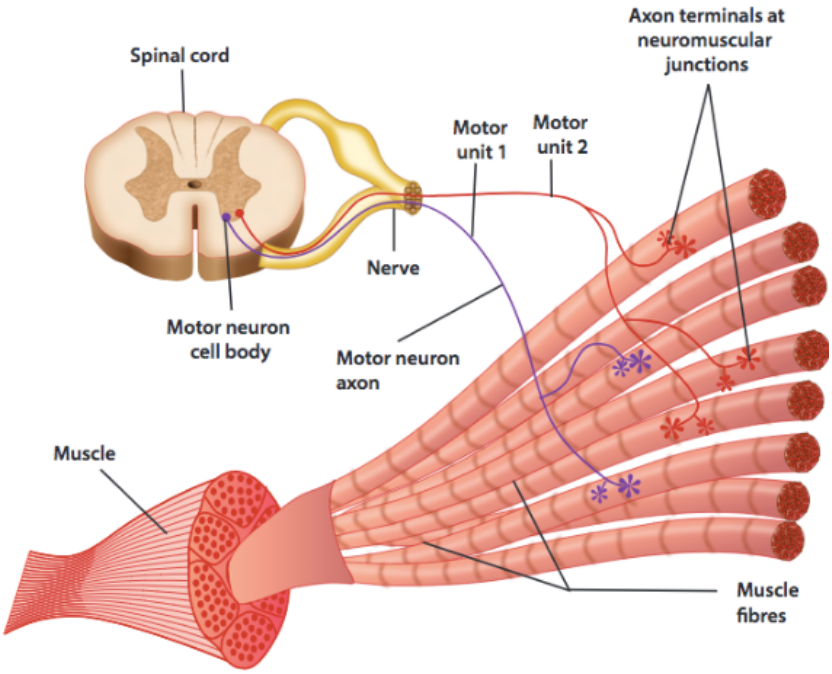


Figure 3.3 A motor unit and muscle fibers [34]

3.3 An EMG Signal: Acquisition and Its Features

As already mentioned, an EMG signal consists of several MUs. Those are acquired by electrodes which might be surface or needle type. Using surface electrodes is a non-invasive method whose ability is adequate for capturing superficial muscle movements that are enough for the proposed method in this thesis. Being careful during electrode placement is crucial to get the signal without or with less noise interference. After electrodes are placed, pre-processing steps must be applied to the signals whose amplitudes are very low, and noisy behavior should be filtered out. After amplification and filtration, rectification, smoothing, feature extraction, and normalization can be performed. For real-time applications within embedded systems, discretization also should be applied.

EMG signals are stochastic which means that they might have random or probabilistic behavior. Such signals do not have repeatable patterns in general and illustrate variety due to the random process or noise. According to the study by Goldberger et al., [35] healthy, myopathic, and neuropathic EMG signals can be picturized as in Figure 3.4.

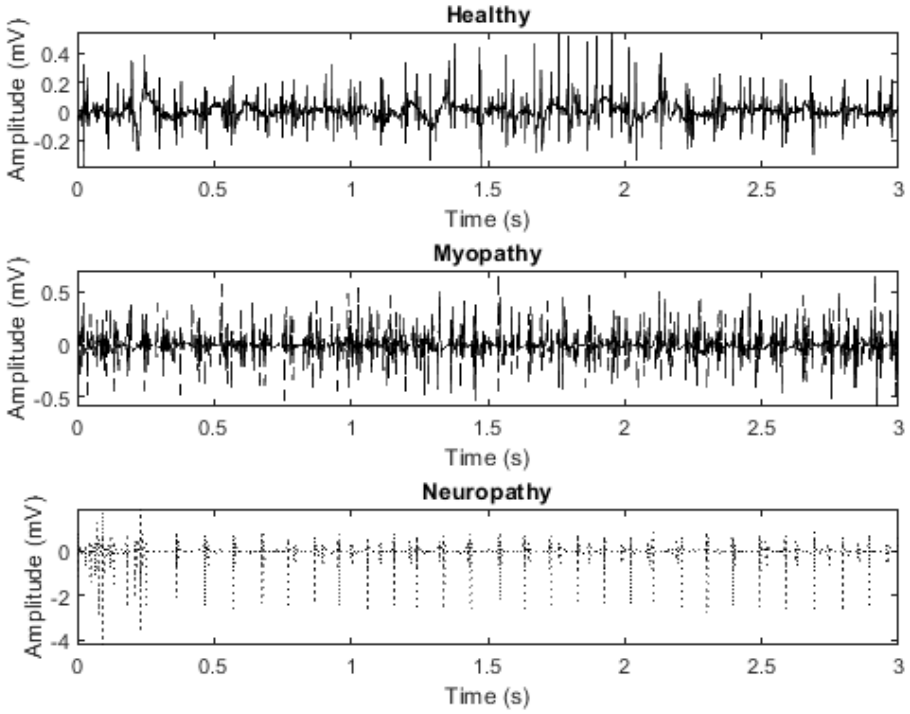


Figure 3.4 Healthy and pathophysiologic EMG signals

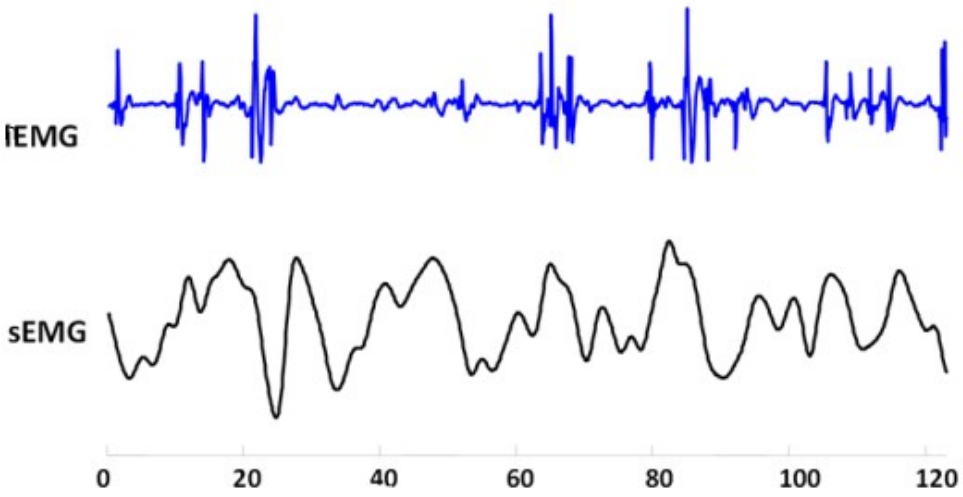


Figure 3.5 Examples of needle and surface electrode used EMG signals in milliseconds [36]

The general overview can differ according to the type of the used electrodes. The following figure depicts possible differences between surface and needle electrodes in time (sEMG stands for surface EMG). As can be seen clearly, the invasive method might be more accurate; however, the surface electrode measurements have a good tendency to get a mask of it for classification. Non-invasive methods are also

more appropriate for academic research without any restrictions in terms of medical ethics norms. There are several different types of surface electrodes according to based material, the used sensor type, etc. In this thesis, a disposable, AgCl (silver chloride) sensor used, covered with hydrogen to conduct electricity better is used. These electrodes are composed of a metal base, usually comprised of silver with the skin-contacting surface coated with silver chloride. The conductive surface of the silver chloride coating facilitates the effective transfer of electrical signals from the skin to the electrode.



Figure 3.6 Surface electrodes for EMG acquisition

3.4 EMG Pre-processing

One of the aims of the thesis suggests that EMG signal classifies and the suggested exoskeleton decides according to the class of the signal. First, as mentioned, it has to be pre-processed. Generally, while the acquisition is processed, signal filtering, rectification, smoothing, feature extraction, and normalization steps are applied. Although, the first step is high pass filtering in most of the studies, in this thesis project a claimed better approach is examined. This method, the Teager-Kaiser energy operator (TKEO), is a signal-processing method that is used for data feature extraction and signal enhancement during pre-processing. It is primarily used to suppress low-frequency noise and baseline drift while enhancing the high-frequency part. Actually, the traditional high pass filtering on the EMG signals can increase the magnitude of the error in detecting the onset of the EMG burst [37]. TKEO output has an advantage over other onset detection techniques in that the computed energy is based on the amplitude and frequency of the signal. As a result, as the amplitude and frequency of signals fluctuate quickly due to the depolarization of the muscle cell membrane during contraction, TKEO may enhance our capacity to study muscle activity. As a result, it highlights the amplitude and frequency of

motor unit action potentials, improving signal-to-noise ratio (SNR) and potentially increasing the accuracy of onset detection [38]. TKEO operator, essentially can be calculated as

$$TKEO(n) = x(n)^2 - x(n-1)x(n+1) \tag{3.1}$$

Here $x(n)$ indicates the signal point at a certain time step while $x(n-1)$ represents the previous step and $x(n+1)$ represents the next step. This process enhances the high-frequency components of the signal while attenuating the low-frequency components. The TKEO algorithm has advantages over a simple high-pass filter in EMG signal processing. Although a high-pass filter is effective at removing low-frequency noise and baseline drift, it can also distort high-frequency components, including EMG spikes or action potentials that are necessary for muscle activity analysis [37].

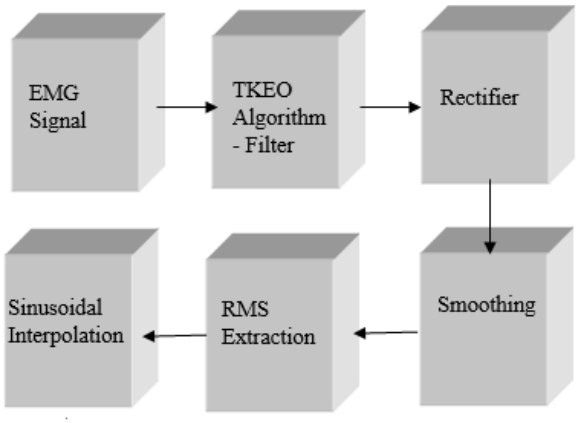


Figure 3.7 Preprocessing steps for an EMG signal

To see the advantages of the TKEO, simply a high pass filter has been applied and compared to it. The general expression of the applied 4th order high pass Butterworth filter can be expressed as:

$$H(s) = \frac{Gs^4}{D(s)} \tag{3.2}$$

Here $H(s)$ is the transfer function of the Butterworth high pass filter. G is the gain of the filter and is typically taken as 1 for normalization. While ζ is the damping ratio, ω_c is the cut-off frequency, and ω_p is the pole frequency and is usually chosen larger than the cut-off frequency to obtain the desired rounding. The cutoff frequency of a common high-pass filter is usually set between 20 Hz and 50 Hz for this kind of application. This range is often preferred because EMG signals typically contain useful information between 20 Hz and 500 Hz, and a higher cutoff frequency helps elimination of the low-frequency noises [39]. It should be noticed that $D(s)$ is equal to:

$$D(s) = s^4 + 2\zeta\omega_c s^3 + (\omega_c{}^2 + 2\zeta\omega_c\omega_p)s^2 + 2\zeta\omega_c{}^2\,\omega_p s + \omega_c{}^2\omega_p{}^2 \tag{3.3}$$

The other parameters have been shown numerically in the following table:

Table 3.1 Parameters of the Butterworth High Pass Filter

Parameter	**Value**
ω_c	188.5 rad/s
ω_p	282.7 rad/s
ζ	0.707

For the healthy EMG signal subject, the result of the applied filter can be picturized as the following figures.

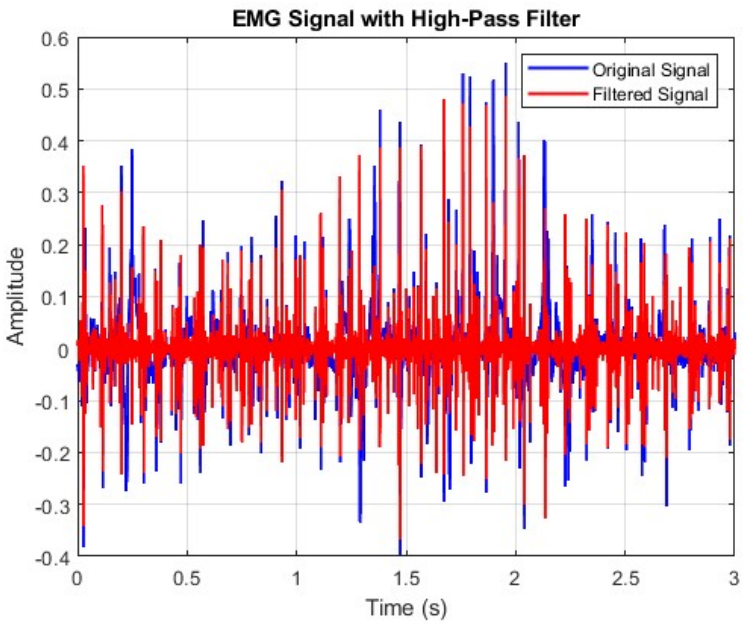
As it is clearly depicted the TKEO method could get rid of the useless signal at the low range which might be a much better outcome for the usage of the feature extraction.

The absolute value of the signal obtained in the rectification process after the filtering process was simply taken. A moving average filter was applied for smoothing. Thus, while eliminating polarization, amplitude information was preserved and the signal-to-noise ratio was improved by reducing random variations. The mathematical formula used for smoothing is expressed below.

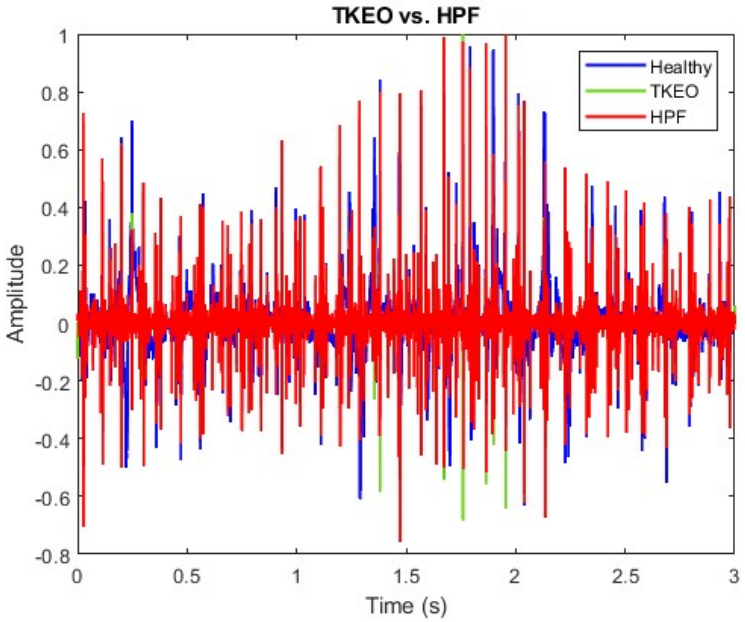
$$y(n) = \frac{1}{N}\sum_{k=0}^{N-1} x(n-k) \tag{3.4}$$

Here, y indicates the output while x stands for input. N symbolizes the filter length. The window size for the moving average filter applied for the smoothing process was taken as 100 and applied to all signals. The window length of the signals, whose square root mean was then taken, had been evaluated as 200 samples. Now, the most important step of the pre-processing is the feature extraction. To do this, various techniques are applied. In this study, the root mean square method (RMS) has been applied which has been widely used in previous studies for EMG signal processing. Basically, the root mean square of a signal is calculated by the equation given below.

$$RMS = \sqrt{\frac{1}{N}\sum_{n-1}^{N} x(n)^2} \tag{3.5}$$



(a) High-pass filtered healthy EMG signal



(b) TKEO vs. HPF

Figure 3.8 Filtering process of the signal

Cumulatively, the RMS values have been calculated as 0.0035 for the healthy sign, 0.0085 for the myopathic, and 0.1111 for the neuropathic one, and it gives the system decision-making data to choose the healthy one for the control sign to be used and to use the lowest in the process of simulating the healthy one. The relatively lower RMS value offered by the preferred TKEO algorithm compared to the high-pass filter shows that the signal contains less noise. Generally, the decision-making systems focus on the threshold values as the RMS operation values, for such systems. However, the proposed method has utilized the neural network approach for classifying these healthy and problematic signals.

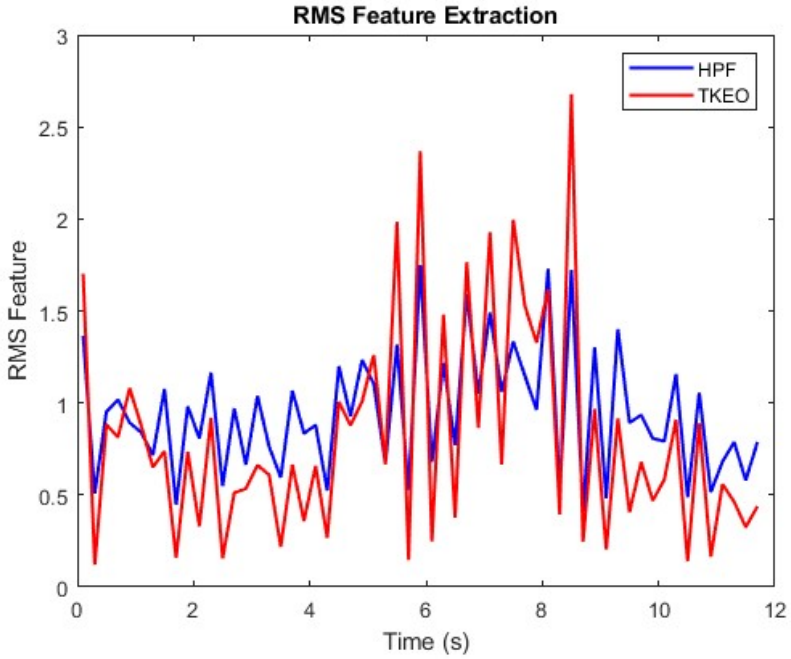


Figure 3.9 The RMS of the healthy EMG signal

These steps have been operated for the myopathic and neuropathic signal sets as well. After this step, normalization and segmentation stages are followed. Optionally, a final process can be performed here for the optimization of the signals to be used, the main purpose of which is to detect and remove any artifacts or unwanted marks that may have appeared during the data collection or pre-processing steps. Since the maximum points of the signals used in this study have an important role in tracking the desired pathways of the joints, local maximum points were first determined in order to apply the sinusoidal interpolation technique. Then, sinusoidal interpolation was applied and the final signals obtained with pre-processing applications were obtained.

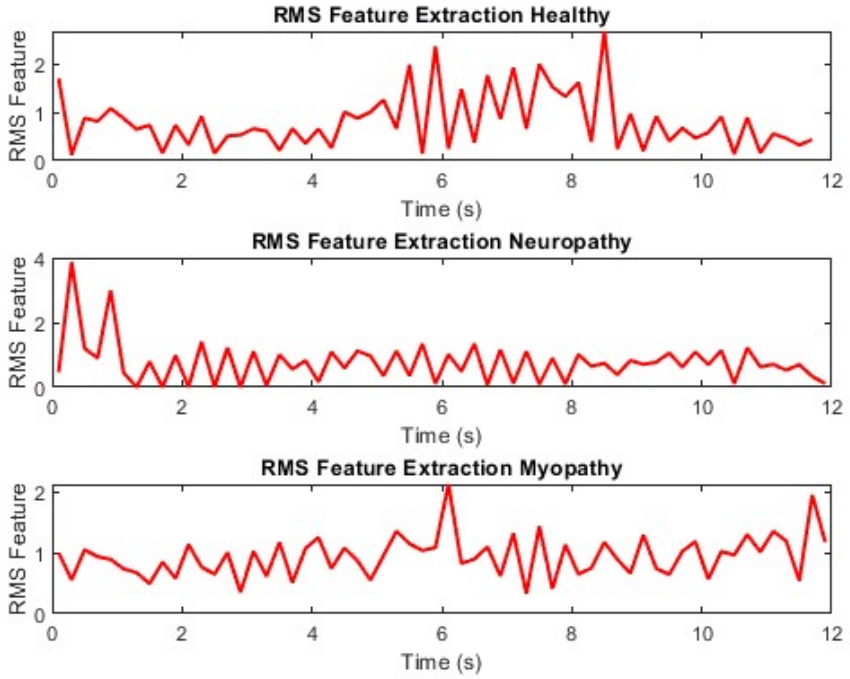


Figure 3.10 The RMS of all types of the EMG signals

In conclusion, the conversion of a stochastic EMG signal into a more deterministic form can be processed through a series of signal-processing steps that have already been explained. By RMS value extractions, the meaningful amplitude values of such signals can be obtained. Subsequent sinusoidal interpolation of the peak values of these amplitude values allows for the generation of the continuous signal which eases the analysis and further usage for the classification. Although there are valid important time domain analyses such as zero crossing, mean absolute value, and slope sign changes, those also give stochastic results which are very hard to implement for artificial intelligence methods to classify the signals whether problematic or not. These proposed steps can be said to be basic and new methodologies for discussed achievements.

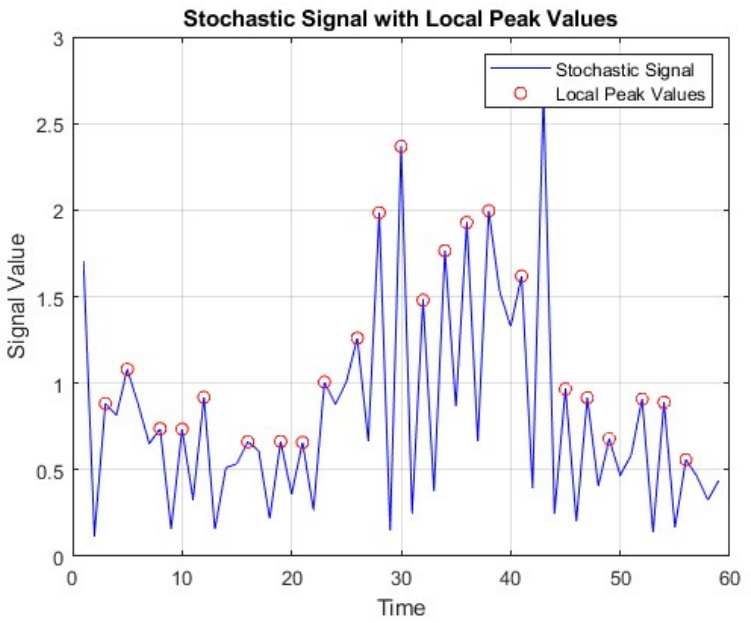


Figure 3.11 Local peak values for healthy EMG

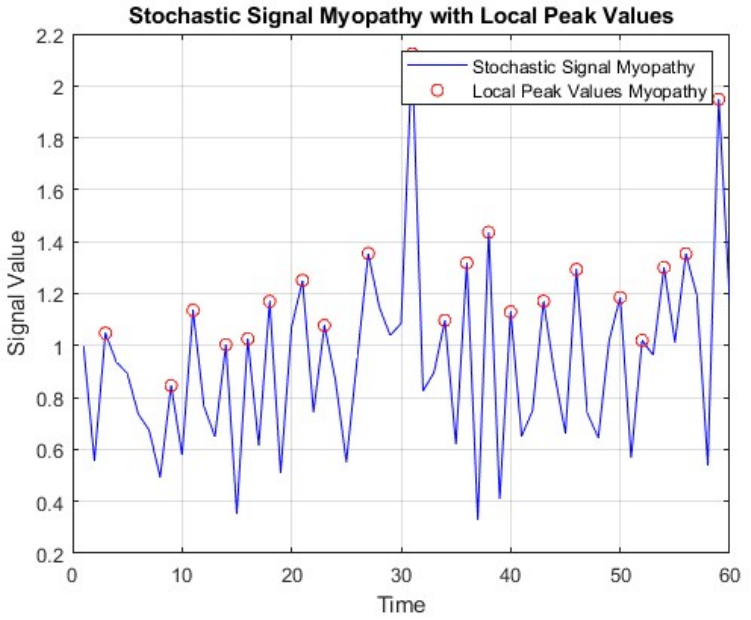


Figure 3.12 Local peak values for myopathic EMG

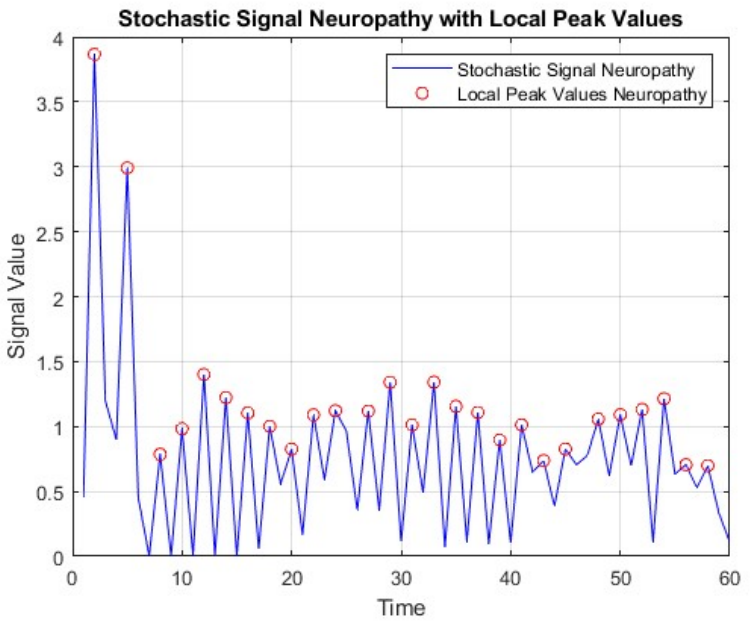


Figure 3.13 Local peaks values for neuropathic EMG

The following figure illustrates the sinusoidal interpolation of these local peak values, resulting in a more deterministic and continuous signal. This transformation makes it easier to apply artificial intelligence methods for classification purposes.

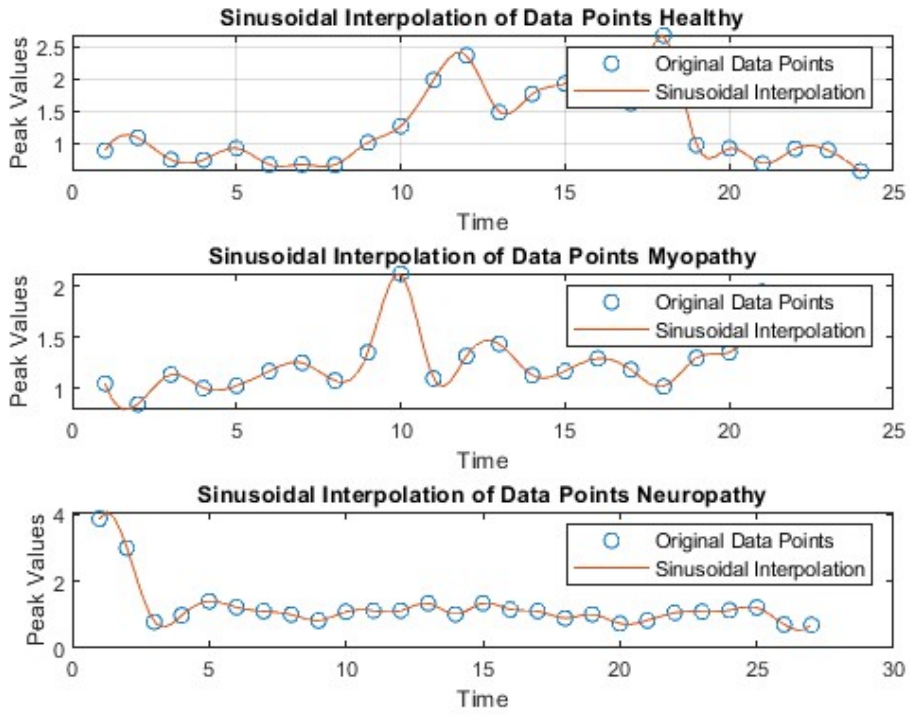


Figure 3.14 Interpolation results of each signal

To design the tremor cancelation mechanism, the frequency analysis of the shoulder movement data should be considered, As already mentioned before, 2-6 Hz tremors can be caused by Parkinson's disease and those tremors can affect the quality of the rehabilitation process negatively. In order to get rid of it during abduction and adduction movements, an interlock system has been proposed under the assumption that the displacement changes in the specified range of frequency brought about by such tremors. The system has been designed with some basic steps: a displacement sensor reads the displacement values then this signal is pre-processed by the similar time domain analysis steps that are explained in this section already. Then, through the Fourier Transform (FT) the frequency analysis is evaluated to detect the range of tremors, and after detection system locks itself through a tiny magnet for locking the passive joint on the shoulder which can perform abduction and adduction movements. Actually, as already told, these movements are not in the scope of this thesis in terms of rehabilitation. Therefore, tremors can be get rid of by this locking mechanism. For frequency analysis of the signals, there are several methods, one of the important methods is FT[40].

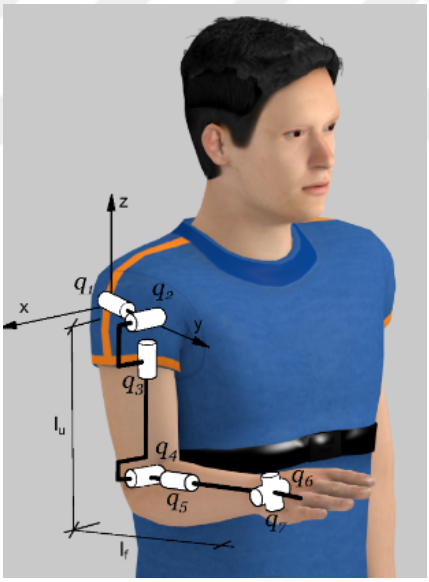


Figure 3.15 The axes of the human arm movements
[41]

Definition 3.1. The Fourier Transform is a mathematical operation that transforms a time-domain signal into its frequency-domain representation. The continuous Fourier Transform of a function $f(t)$ is defined as:

$$F(\omega) = \int_{-\infty}^{\infty} f(t) e^{-i\omega t}\, dt \tag{3.6}$$

where $F(\omega)$ is the Fourier Transform of $f(t)$, ω is the angular frequency, i is the imaginary unit, t is time.

According to the human arm axes, the x-axis analysis can be sufficient for the detection of the tremor movement. Abduction-adduction of the shoulder can be sensed directly on the x-axis. Therefore, the FT of the signal from this can evaluated for catching the 4-6 Hz tremors. The system may work for an on-off mechanism then once the signal in this range is detected, the system locks itself for this passive joint. To check this ability, the simulations have been realized on MATLAB. To do this, a signal was generated; it mimicked 20 degrees of abduction and adduction movement that immediately followed each other. Then through the threshold values, 4-6 Hz ranged movement could be detected and discriminated from the main movement signal.

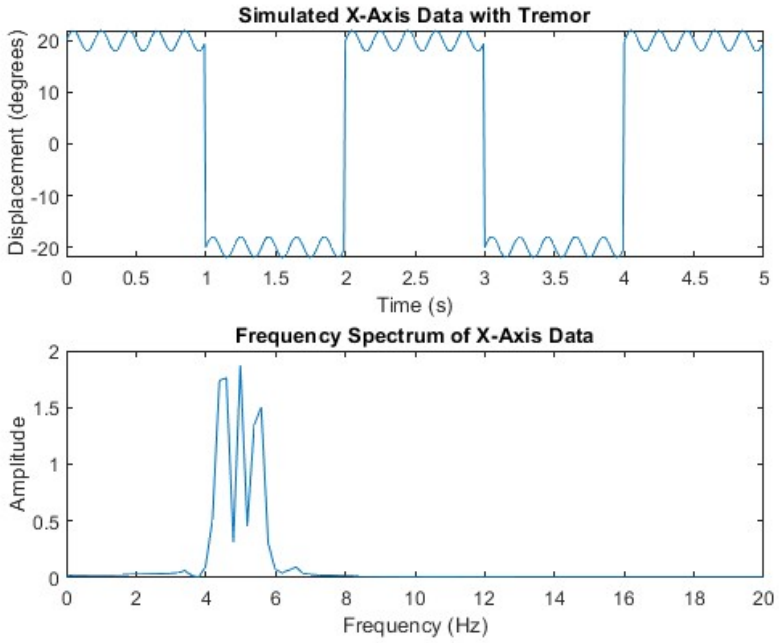
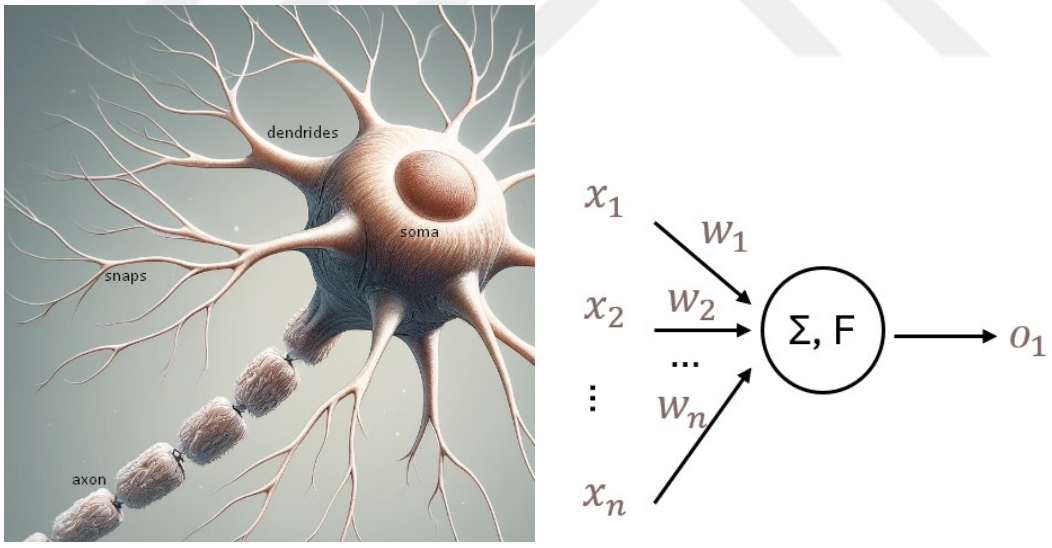


Figure 3.16 The simulation results on tremor detection

3.5 EMG Classification By The Neural Network

Artificial neural networks (ANNs) are computational models inspired by the human neural network structure of the nervous system. They are designed to mimic the ability of the neural network of the human body to recognize patterns and solve complex problems by training and learning. These networks consist of interconnected nodes which are basically neurons, which process data in layers to identify correlations and make predictions. There are so many variations of these artificial neural networks and many various algorithms have been proposed since

the last century in the machine learning field within ANNs[42]. One of the most frequently used ANNs in the literature is Multilayer Perceptron (MLP). An MLP is a type of artificial neural network that consists of multiple layers of nodes, including an input layer, one or more hidden layers, and an output layer. Each node, or neuron, in one layer, is connected to every node in the next layer, allowing the network to learn complex patterns in the data through a process called backpropagation (BP). The main purpose of an MLP is to approximate any function by learning from the input-output mappings provided during training, making it a versatile tool for various tasks such as classification, regression, and pattern recognition [43]. To interpret the MLPs, it should be said that they consist of three main layers: input, output, and middle layers. There are no connections between the nodes of the same layer, all the connections are made only for forward directions. The mid one is also called the hidden layer, generally taken as one while it can be more than this. An example of an MLP architecture can be seen in Figure 3.11. Here, p indicates the number of the inputs, while m stands for the number of the hidden layers. W and V matrices have the weights between the input-hidden layer and hidden layer-output, respectively. The Σ is a summation symbol and F symbolizes the activation functions in the nodes. Finally, bh_m indicates the hidden layer bias values.



(a) An illustration of a biological neuron **(b)** A basic node of a NN

Figure 3.17 The fundamental parts of the neural networks

The data used for the operation of the ANNs are input samples and target values. The main goal of the MLPs is determining the best-fitted values of weights and bias values in order to make outputs converge to the target values. The sub-work for determining these values is called training. By training, the system tries to find the most optimum ANN parameters in this structure with the smallest error values. For

the further predictions, the system must decide even without the inputs. This can be made by the generalizations which are various methods.

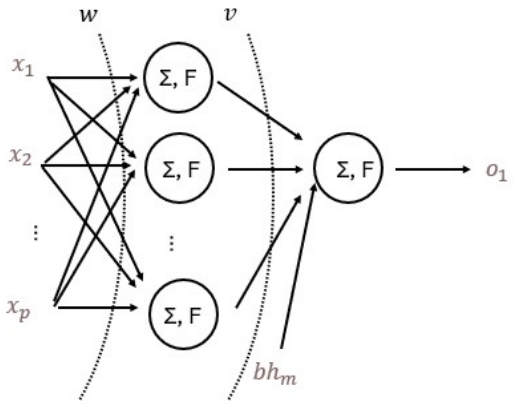


Figure 3.18 An example of an MLP structure

The elementary mechanism that enables an MLP to learn from the provided data is the backpropagation algorithm, a supervised learning technique widely used for training an ANN. The foundation of this algorithm can be evaluated as a kind of gradient descent method. In the BP learning algorithm, the error value, which is the difference between the network output obtained through forward computation for each learning example and the target value, is minimized by updating the weights backward within the structure of the network. The method has two main steps whose names are the forward pass and the backward pass to accomplish training. The input signals—in this example, time vs. voltage values from EMG datasets—propagate through the network to generate an output during the forward pass. The calculation of the error at the output layer is the first step in the backward pass, after which it is propagated backward across the network. The calculus chain rule is used to calculate gradients of the error with respect to the weights, which enables the weights to be adjusted in a way that minimizes the error [44]. Square error terms are taken into account when updating the weights. In the following equation, A indicates the target function, B is a vector that contains the bias values, and W and V are the weights that are mentioned above.

$$A(B, W, V) = \frac{1}{2} {e_n}^2 \tag{3.7}$$

Definition 3.2. The chain rule in calculus is a fundamental formula used to find the derivative of a composite function. If we have two functions f and g, where f is a function of g, and g is a function of x, then the chain rule states that the derivative of f with respect to x is:

$$\frac{d}{dx}f(g(x)) = f'(g(x))g'(x)$$

For partial derivatives, the chain rule is applied similarly. Suppose $z = f(x, y)$, where x and y are both functions of t. The chain rule for partial derivatives is given by:

$$\frac{\partial z}{\partial t} = \frac{\partial z}{\partial x}\frac{\partial x}{\partial t} + \frac{\partial z}{\partial y}\frac{\partial y}{\partial t}$$

First, the updating formulas for V can be calculated as follows:

$$v_j{}^{t+1} = v_j{}^t + \delta v_j{}^{t+1} = v_j{}^t + \lambda(-\frac{\partial A}{\partial v_j}) \tag{3.8}$$

$$\frac{\partial A}{\partial v_j} = \frac{\partial A}{\partial o_n}\frac{\partial o_n}{\partial net_o}\frac{\partial net_o}{\partial v_j} \tag{3.9}$$

The other bias and weight values of the hidden layers and input-output layers are updated according to the several formulas that are basically derived by this approach. Here, j is the hidden layer index, i is the input layer index, and k stands for the iteration counter. The flowchart of the BP algorithm is shown in the following figure. The efficiency and effectiveness of BP in training MLPs make it a fundamental technique in neural network-based signal processing, especially offering robust solutions for complex classification tasks in biomedical signal analysis [45].

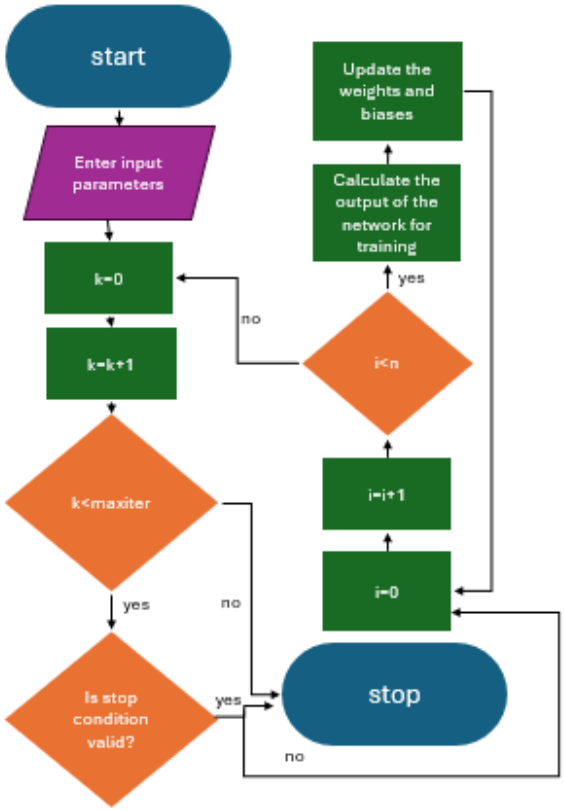


Figure 3.19 The flowchart of BP algorithm

To talk about its advantages it can be said that it provides a systematic way to compute the gradient of the loss function according to each weight by applying the chain rule of calculus, making the training process more efficient. Scalability which means its ability to scale the networks with multiple hidden layers and many numbers of neurons gives another power to this algorithm. However, it also has some drawbacks. One of its disadvantages is not being fast. It requires relatively higher iterations and it converges slowly to the error condition. To make the algorithm faster, there are some modifications in the literature. Generally, due to the bigger step size, the BP can fluctuate. The momentum term which is added to the algorithm weight update formula, helps to accelerate the gradient vectors [46]. It adds a fraction of the previous weight update to the current one. At the optimum point, fluctuation and so, sign changes can occur. The momentum term minimizes or maximizes the step size according to the change in weight. Another method for making the algorithm more rapid is the adaptive learning rate method. It is based on making the learning rate changeable instead of taking it being constant. It basically relies on adapting the learning rate according to the parameters, it performs larger updates for infrequent and smaller updates for frequent parameters [47]. Moreover, there is the Levenberg-Marquardt (LM) method that is one of the widest applied methods for training the ANN without such problems, and within the light of these updates.

3.5.1 The Levenberg-Marquardt Algorithm

The LM method is a kind of changed Newton method as an optimization approach for training. The proposed BP learning algorithm for MPs works by approximating the objective function with a linear function using the first term of the Taylor series expansion of the objective function. Therefore, BP is referred to as a first-order training algorithm, whereas the LM method is considered a second-order method. In the LM, the weights are changing once for each iteration. So, there is no updating operation for each training example. The objective function is equal directly to the total error. In the Newtonian method, W weight and bias included vector and J jacobian matrix are calculated. Here, W consists of the hidden layer, input and output layer weights, and all the layer bias parameters, in more detail, $W = [w_{11}, ..., w_{mm}, v_1, ..., v_m, bh_1, ..., bo_1$ and for the error vector e, J can be represented as partial derivatives of each term inside the e with respect to the W_1 to w_m. Therefore,

$$\nabla^2 f(W^t) = J^T \tag{3.10}$$

and finally, the weight updating can be achieved by the following equation:

$$W^{t+1} = W^t - [J^T J]^{-1} J^T e \tag{3.11}$$

The finding of the inverse of the $J^T J$ is sometimes impossible for some of the iterations and it can also be encountered that the convergence problem is due to the fixed step size. Therefore, they can be manipulative for better results. Actually, the LM or only Marquardt constant μ is inserted into the Newtonian algorithm. Finally, the Equation (3.10) can be converted into the following one:

$$W^{t+1} = W^t - (J^T J + \mu I)^{-1} J^T e \tag{3.12}$$

Training steps have been operated on MATLAB environment within the powerful machine learning tool. The data sets were already mentioned before and labeled as neuropathic, myopathic, and healthy. The following figures represent the parameters of the performance of the training process.

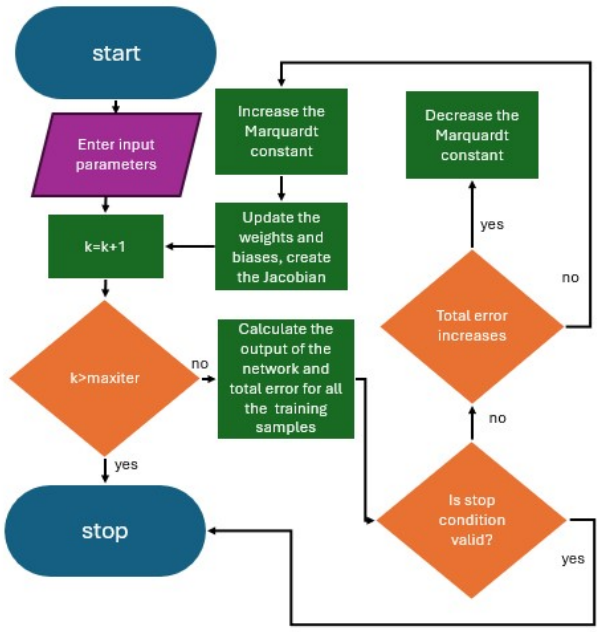


Figure 3.20 The flowchart of LM algorithm

The method has been applied successfully, even though it has shown better performance as it combines the rapidness of the Newtonian method and the robustness of the gradient descent; it has some drawbacks, as well. The LM requires the computation of the Jacobian matrix which can be burdened in the aspects of the process. The trained network has been used for the decision-making system on the real-time sEMG classification.

Before criticizing the results of the neural network training under these techniques, one should know about the neural network training success metrics [48]. For instance, accuracy is the proportion of correctly predicted instances out of the total instances. Besides, precision can be defined as the proportion of true positive

predictions out of the total positive predictions.

$$\text{Accuracy} = \frac{\text{Number of Correct Predictions}}{\text{Total Number of Predictions}}$$

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}}$$

More importantly, a confusion matrix is a matrix that is used to interpret the performance of a classification. It compares the actual target values with the predicted values by the algorithm. Accuracy, precision, and F! score are driven by this matrix [49]. It looks like:

	Predicted Positive	Predicted Negative
Actual Positive	True Positive (TP)	False Negative (FN)
Actual Negative	False Positive (FP)	True Negative (TN)

For the training process, the datasets of each EMG class have been divided into three groups which are test, validation, and training sets, In literature, Although there are different rates for splitting the data, 0.7, 0.2, and 0.1 rates have been applied for splitting, for training, test, and validation respectively [50]. Different hidden layers and epochs have been applied and it has been checked which numbers of epochs and hidden layers had the best accuracy.

The F1 score can be counted as another important metric for evaluating the quality of the training process numerically. It can be defined as the harmonic mean of precision and recall. It is used to balance the trade-off between precision and recall, especially when dealing with imbalanced datasets. The F1 scores for the training process with the best accuracy among the trials are depicted in the following table.

Table 3.2 F1 scores for different classes

Class	**Training**	**Validation**	**Test**	**All**
Healthy	0.899	0.667	0.857	0.881
Myopathy	0.887	0.769	0.857	0.879
Neuropathy	0.889	0.889	0.923	0.917

Table 3.3 Accuracy percentages of different trials of the training by LM

Number of epochs	**Number of hidden layers**	**Accuracy percentage**
50	1	53.6231
100	1	57.971
150	1	43.478
50	2	60.8695
100	2	63.7681
150	2	69.56521
50	3	71.0144
100	3	65.2173
150	3	69.56521
50	4	75.3623
100	4	69.5651
150	4	71.0144
50	5	65.2174
100	5	77.3623
150	5	77.3623
50	6	53.6231
100	6	83.6361
150	6	87.6831
50	7	79.9443
100	7	76.8431
150	7	74.2951
50	8	73.3622
100	8	77.7888
150	8	79.3101
50	9	77.6065
100	9	76.2349
150	9	79.3628
50	10	80.5264
100	10	75.7218
150	10	74.9096

According to the accuracy values, the best results have occurred for the 150 epochs, 6 hidden layers case. Then, after a training session of this one, the confusion matrix

has been determined as the following figure:

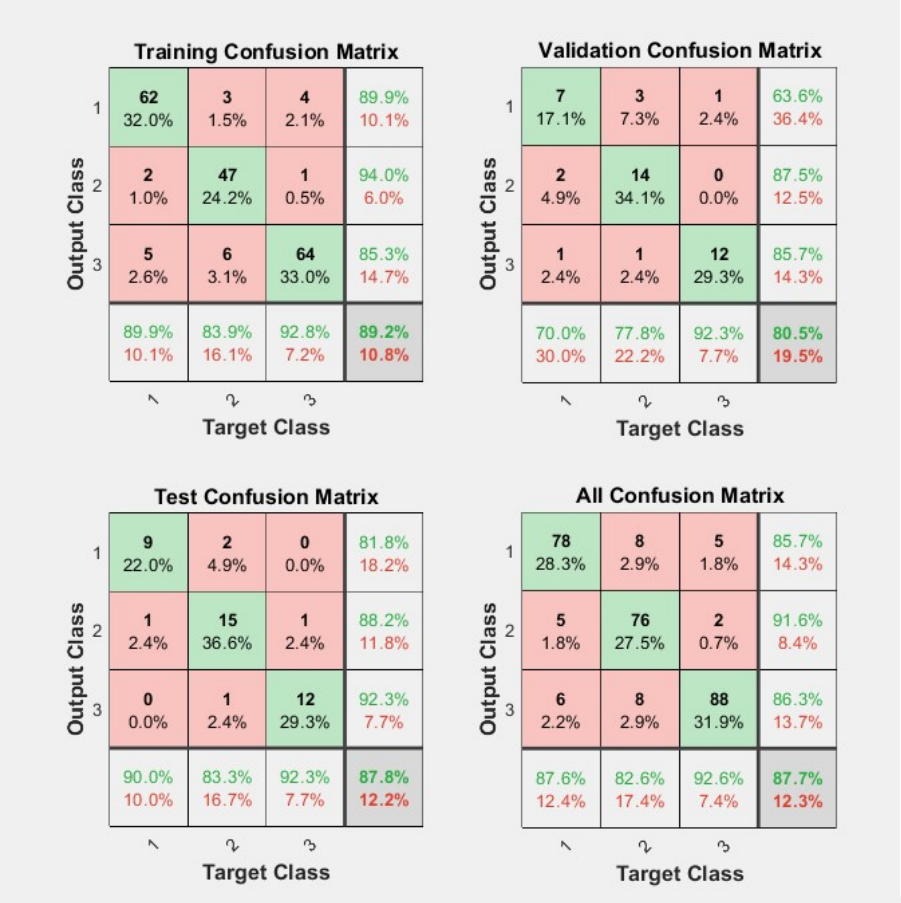


Figure 3.21 The confusion matrix for the training for 150 epochs, 6 hidden layers

The confusion matrices for the training, validation, test, and overall datasets reveal that the model performs reliably in classifying the given data. During training, the model achieves high accuracy across all classes, with an overall accuracy of 89.2%. The validation data shows a slight drop in accuracy, particularly for Class 1, which achieves 70.0%, while Classes 2 and 3 perform better at 77.8% and 92.3%, respectively, resulting in an overall accuracy of 80.5%. The test data maintains a high level of accuracy across all classes, with an overall accuracy of 87.8%. Combining all datasets, the model achieves an overall accuracy of 87.7%, indicating robust performance. Although the model demonstrates reliable classification capabilities, particularly for Classes 2 and 3, it should be improved, especially in enhancing the accuracy for Class 1 during validation.

Definition 3.3. The Mean Squared Error (MSE) is a common metric used to measure the average squared difference between actual target values and predicted

values by a model. It is calculated as follows:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{3.13}$$

where n is the number of observations, y_i is the actual value for the i-th observation, $\hat{y}_i$ is the predicted value for the i-th observation.

In light of this information, the following figure can evaluate the validation performance of the training process for the NN. The MSE is used to evaluate the performance of the model during training and validation. The best validation performance has been achieved at the point of the MSE is equal to 0.12404 at epoch 6, indicating the point at which the predictions of the models are the closest to the actual target values on the validation set. Monitoring the MSE over epochs helps in identifying the optimal point to stop training to prevent overfitting, as evidenced by the increase in validation error beyond epoch 6.

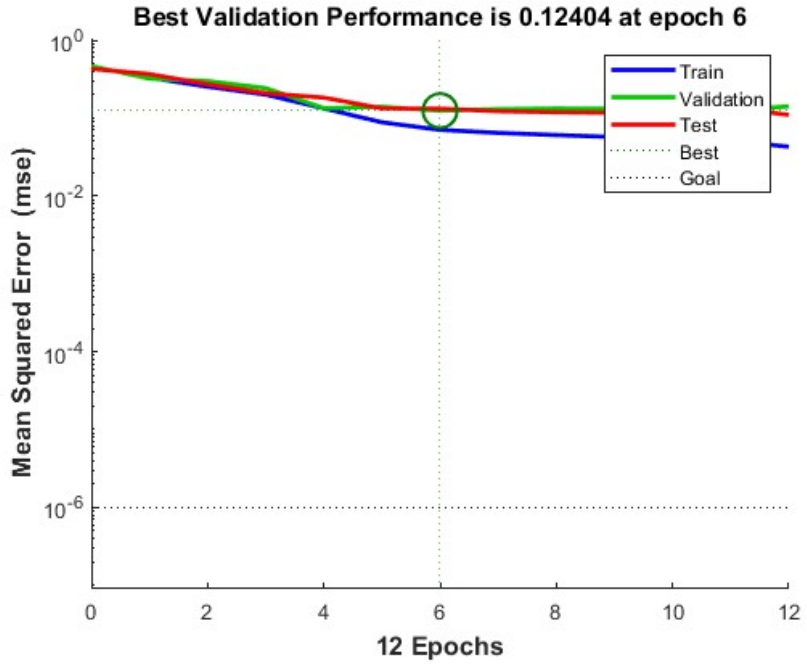


Figure 3.22 Performance metrics of the NN validation

The error histogram can depict the performance results as well. The error histogram provides a visual representation of the distribution of prediction errors made by the model. The highest bar is centered around zero error, indicating that most of the predictions are very close to the actual target values. The error distribution appears to be relatively symmetrical around zero. This symmetry suggests that the model

does not have a systematic bias towards overestimating or underestimating the target values. The histogram suggests that the model is generally accurate, with most predictions closely matching the actual target values.

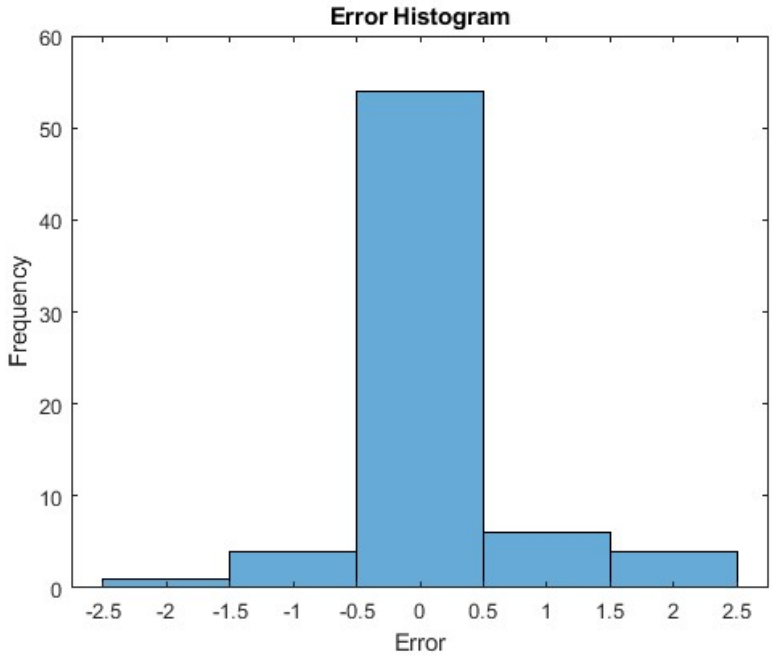


Figure 3.23 Error histogram of the best training sessions

The frequency in the error histogram plot represents the number of instances (or samples) that fall within each bin of error values. The x-axis shows the error values, which are the differences between the predicted and actual values. The y-axis shows the frequency, or the number of instances, corresponding to each error bin.

At last, plotting the True Positive Rate (TPR) against the False Positive Rate (FPR) at different threshold values allows for the creation of an ROC (Receiver Operating Characteristic) curve, a graphical representation that is used to assess the performance of binary classifier systems, including neural networks. The AUC (Area Under Curve) represents the degree or measure of separability achieved by the classifier. It indicates how well the classifier can distinguish between the positive and negative classes. AUC has been found as 0.94, this value suggests that the neural network model has a high ability to distinguish between the positive and negative classes. A value of 0.94 is considered excellent, indicating that there is a 94% chance that the model will correctly distinguish a randomly chosen positive instance from a randomly chosen negative one.

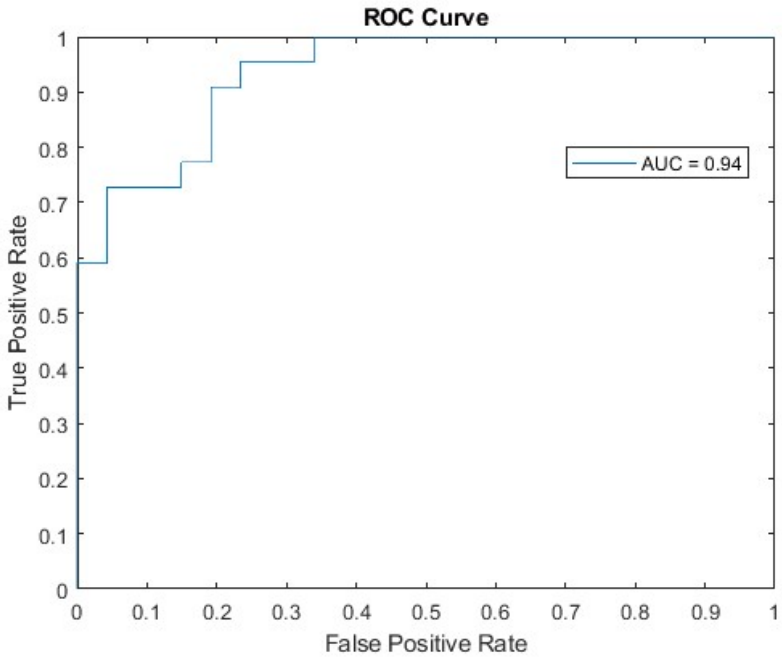


Figure 3.24 The ROC of the selected hidden layer-epochs training process

For the indicated training process, according to equation 3.13, MSE values can be seen as the following table.

Table 3.4 MSE values for the training process

Dataset	MSE
Training	0.247
Validation	0.341
Test	0.122
All Data	0.184
Combined	0.214

MSE is actually mostly used for comparing different methods; however, here it is a good indicator whose results are low for each dataset. A lower MSE relative to other models or previous iterations usually indicates better predictive performance.

3.6 From EMG to Muscle Force and Torque

After preprocessing, classifying, and then defining the EMG signal feature; it should be used for determining the force and torque for the selective movements of the arm. To do this, there are several different methods to generate the force from the EMG signals [51]. In this thesis, a Hill muscle model-based force-length and force-velocity relationships have been evaluated to propose a function that converts

the EMG signals into the predicted forces so torques that are assumed to be linearly dependent on that force.

3.6.1 Hill's Muscle Model

The Hill-based muscle model is a widely used representation of muscle mechanics in biomechanics [52, 53]. The model depicts muscle force generation through three key components: the contractile element (CE), series elastic element (SE), and parallel elastic element (PE). The contractile element represents the active force production of the muscle fibers, which depends on both neural activation and the intrinsic properties of the muscle, such as the force-length and force-velocity relationships. The series elastic element accounts for the elasticity of tendons and other connective tissues that transmit force to the skeleton. The parallel elastic element models the passive elastic properties of the muscle fibers. The overall muscle force is a combination of these components, allowing the Hill-based model to accurately simulate muscle behavior under various conditions of length, velocity, and activation.

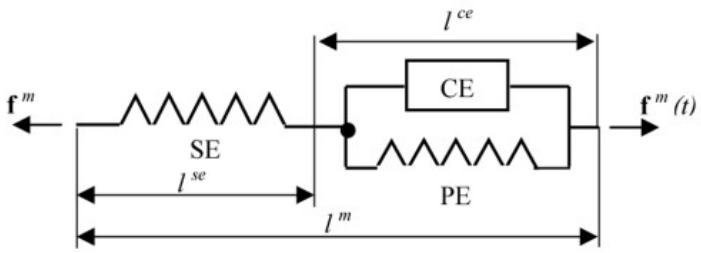


Figure 3.25 The Hill muscle model[54]

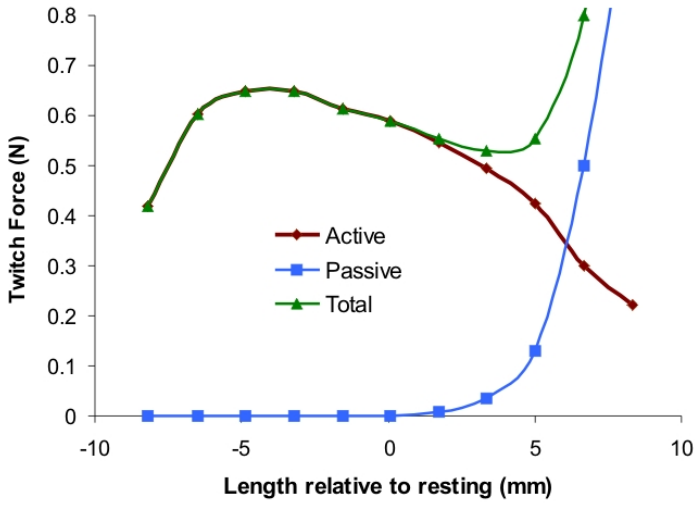


Figure 3.26 The Hill muscle model forces[55]

For EMG-to-force relationship the following equation can be used:

$$a_j(t) = \frac{e^{Au_j(t)} - 1}{e^A - 1} \tag{3.14}$$

where where $a_j(t)$ is the activation of muscle j; $u_j(t)$ the postprocessed EMG of muscle j at time t; A the nonlinear shape factor, constrained to $-3 < A_j < 0$; with 0 being a linear relationship [56].

Individual muscle-tendon length and activation data might be used as input to a modified Hill-type muscle model to calculate individual muscle forces. The force produced by each contractile element is estimated using a Hill-type muscle model with generic force-length $f(l)$, force-velocity $f(v)$, and parallel passive elastic force-length $f_p(l)$ curves. These curves are normalized to maximum isometric muscle force F_{max}, optimal fiber length L_0, and maximum muscle contraction velocity v_{max}. The general form of the equations for the force produced by the muscle-tendon unit ($F_{\text{mt}}(t)$) is given by:

$$F_{\text{mt}}(t) = F_t = F_{max}[f(l)f(v)a(t) + f_p(l)]\cos(\phi(t)) \tag{3.15}$$

where F_t is the tendon force. Pennation angle ($\phi(t)$) refers to the angle between the muscle fibers and the tendon they attach to. It describes the angle at which the muscle fibers are oriented relative to the direction of the tendon. Pennation angles can vary depending on the muscle and its function. It changes with instantaneous muscle fiber length by assuming the muscle belly has a constant thickness and volume. The following function is used to calculate the pennation angle at time t:

$$\phi(t) = \sin^{-1}(\frac{L_0^m \sin\phi_0}{L_m(t)}) \tag{3.16}$$

where $L_m(t)$ is the muscle fiber length at time t; and ϕ_0 is the pennation angle at muscle optimal fiber length, L_0^m.

The mathematical equations for the model elements can be written as:

$$F_{CE} = F_{\text{max}} \cdot a(t) \cdot f_L(L_{CE}) \cdot f_V(V_{CE}) \tag{3.17}$$

where F_{CE} is the force produced by the contractile element, F_{max} is the maximum

isometric force of the muscle, $f_L(L_{CE})$ is the force-length relationship function, $f_V(V_{CE})$ is the force-velocity relationship function, L_{CE} is the length of the contractile element, and V_{CE} is the velocity of the contractile element.

$$f_L(L_{CE}) = \exp\left(-\left(\frac{L_{CE} - L_0}{w}\right)^2\right) \tag{3.18}$$

where L_0 is the optimal length of the muscle, and w is a width parameter determining the range of lengths over which the muscle can produce force.

$$f_V(V_{CE}) = \begin{cases} (1 - \frac{V_{CE}}{V_{\text{max}}})/(1 + \frac{V_{CE}}{V_{\text{max}}}) & \text{if } V_{CE} \leq 0 \\ (1 - \frac{V_{CE}}{V_{\text{max}}})/(1 + \frac{V_{CE}}{V_{\text{max}}}) & \text{if } V_{CE} > 0 \end{cases} \tag{3.19}$$

where V_{max} is the maximum shortening velocity of the muscle.

$$F_{\text{total}} = F_{CE} + F_{PE} + F_{SE} \tag{3.20}$$

where F_{total} is the total force generated by the muscle, F_{PE} is the force from the parallel elastic element, and F_{SE} is the force from the series elastic element.

$$F_{PE} = k_{PE}(L_{CE} - L_0) \tag{3.21}$$

where k_{PE} is the stiffness coefficient of the parallel elastic element.

$$F_{SE} = k_{SE}(L_T - L_{0,T}) \tag{3.22}$$

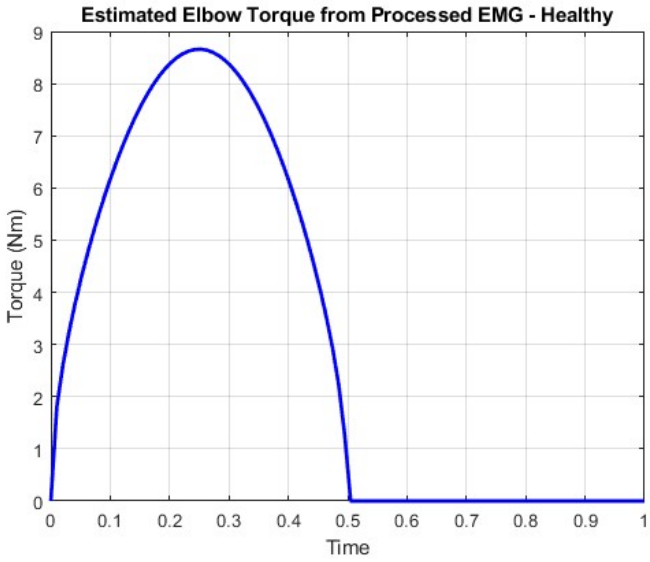
where k_{SE} is the stiffness coefficient of the series elastic element, L_T is the length of the tendon, and $L_{0,T}$ is the slack length of the tendon.

Finally, it can be said that: according to the Hill muscle model, first, the activation function has been determined by the preprocessed EMG signal then, the function approximation has been achieved with the other model parameters and by the assumed linear relationship between torque and force.

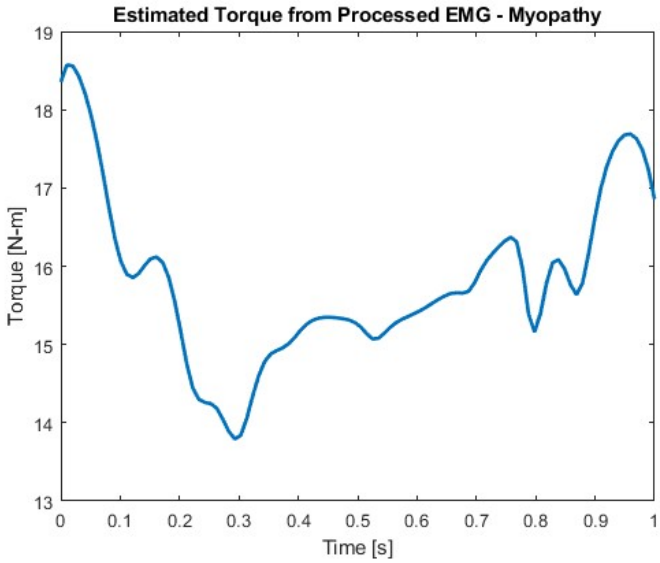
In conclusion, the torque estimations for myopathy and healthy cases that are from the preprocessed EMG signals can be seen in the following figures.

Table 3.5 Parameters used in the Hill-type muscle model for torque estimation from processed EMG signals.

Parameter	Description	Value or Range
$F_{\max}$	Maximum isometric muscle force	350 N
L_0	Optimal fiber length	0.15 m
L_{ts}	Tendon slack length	0.1 m
f_0	Pennation angle at optimal fiber length	$30°$ ($\frac{30\pi}{180}$ rad)
A	Shape factor for activation calculation	-2



(a) Healthy



(b) Myopathy

Figure 3.27 Torque estimations for healthy and myopathy cases for elbow flexion-extension scenario

As it can be seen clearly, fluctuations in even a small amount of time instance can

be seen for the problematic case, although both cases are in the logical ranges for the elbow flexion-extension movement scenario. Within the usage of many big data of the EMG signal, a neural network can be trained for this estimation process, as well. This time, those outcomes require the controller, and the related designs can be seen at the following chapter.

3.7 EMG Acquisition Procedure

The EMG signal quality directly depends on the procedure whose steps are mainly skin preparation and positioning of the surface electrodes. Stable electrode contact and low skin impedance are the key components of the skin preparation technique. The majority of contemporary EMG amplifiers are made to operate at skin impedance levels of 5 to 50kΩ between electrode pairs. Before applying the electrodes, some skin preparation needs to be done. First, removing the hair is required. In order to improve the adhesion of the electrodes, the sweat and hair should be removed.

To increase the signal-to-noise ratio (SNR) during the capture of EMG signals, amplification is necessary. Making use of the differential amplifier's common mode rejection capabilities is a major benefit of employing a differential (bipolar) or double differential electrode design. Common signals, like those resulting from deep muscle fibers or noise, are ignored by both electrodes. For dependable EMG acquisition, the common mode rejection ratio (CMRR) needs to be at least 95 dB [57].Since the average EMG signal has a peak-to-peak voltage of 6 mV, a gain of at least 1000 is suggested; however, low-amplitude signals can require a greater gain. The range of 10 Hz to 500 Hz comprises most of the EMG bandwidth.

To prevent distortion due to the input, the input impedance of the amplifier must be significantly greater than the skin impedance. It is advised to use an input impedance of at least 10 MΩ for Ag-AgCl electrodes. Dry electrodes can require an amplifier input impedance as high as 1 GΩ and will result in a substantially greater skin impedance.

For EMG, a sampling rate of at least 1000 Hz is required, but higher sampling rates are often used to account for the finite roll-off of the anti-aliasing filter and to avoid signal loss. The resolution of an ADC (Analog-to-Digital Converter) determines the number of discrete values it can produce over the range of analog input values, thereby limiting the accuracy of a measurement. Higher resolution, meaning more bits, results in more accurate measurements. An ADC with n bits has 2^n different discrete levels to represent the analog signal. Consequently, for a given dynamic

range of the input signal, more bits increase system accuracy. For instance, an 8-bit ADC can produce 256 discrete levels, whereas a 12-bit ADC can produce 4096 levels. For an input with a dynamic range of 10V (from 0 to 10 volts), an 8-bit ADC would be unable to resolve voltage differences smaller than 39mV, while a 12-bit ADC could resolve voltage differences as small as 2.4mV.

4
CONTROLLER DESIGN

To design a proper controller for the robot, the dynamics model of the robot must be examined first. In order to prepare a trajectory tracking controller for the rehabilitation mode of the robot, first the mathematical model has been interpreted, and then comparative controller schemes have been created and told in this chapter.

4.1 Mathematical Model of the Robot Dynamics

The exosuit basically consists of the two main revolute joints which are basic biomimetic outcomes of the shoulder and elbow joints of the human upper limb. Here, for the sake of simplicity, the movements on the sagittal plane are considered only. As a third degree of freedom, abduction/adduction of the shoulder is a passive part of the design. Therefore, it is only taken as a *checkpoint* for tremor canceling. The user can use the robotic arm for this movement freely if there does not exist a tremor which should be canceled out in the case of Parkinson's disease source is sensed. For the shoulder and elbow joint flexion/extension movements on the sagittal plane, the following figure can be followed to get the mathematical model of the robot in terms of its dynamics.

The system can be represented as a double pendulum model. The joint variable is defined as in Equation (4.1).

$$q = [\theta_1 \quad \theta_2]^T \tag{4.1}$$

Torques from the actuators for each joint are defined in Equation (4.2).

$$\tau = [\tau_1 \quad \tau_2]^T \tag{4.2}$$

Definition 4.1. ***Lagrange's equations***[58] To write the equations of motion, we define the Lagrangian, L, as the difference between the kinetic and potential energy of the system. Thus,

$$L(q, \dot{q}) = T(q, \dot{q}) - V(q), \tag{4.3}$$

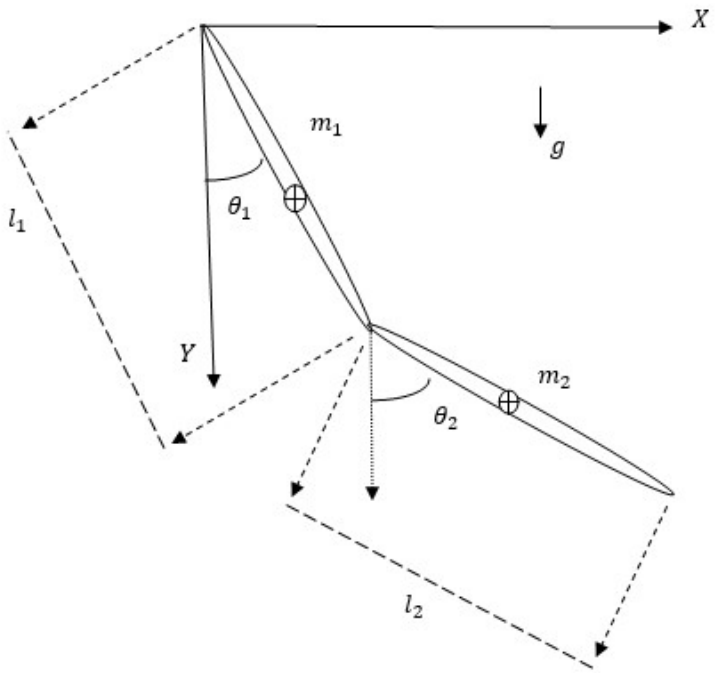


Figure 4.1 The double pendulum model representation of the robotic arm

where T is the kinetic energy and V is the potential energy of the system, both written in generalized coordinates.

Theorem 4.1. ***Lagrangian method*** *The equations of motion for a mechanical system with generalized coordinates $q \in \mathbb{R}^m$ and Lagrangian L are given by*

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{q}_i} - \frac{\partial L}{\partial \dot{q}_i} = \tau_i \quad i = 1, \ldots, n \tag{4.4}$$

where τ_i is the external force acting on the ith generalized coordinate.

According to the Lagrangian method, the equations of motion of the robot have been approximated to the standard form which can be written as Equation (4.5).

$$M(q, \ddot{q}) + C(q, \dot{q}) + G(q) = \tau \tag{4.5}$$

Here, $M(q)$ is the inertia matrix, $C(q, \dot{q})$ indicates the Coriolis/centripetal vector, and $G(q)$ stands for the gravity vector.

In real life, the exoskeletons are always affected by friction and external disturbances. So, the equation should be generalized as [59]

$$M(q,\ddot{q}) + C(q,\dot{q}) + G(q) + F(\dot{q}) + \tau_d = \tau \tag{4.6}$$

τ_d symbolizes disturbance and $F(\dot{q})$ is added as a friction term which is equal to:

$$F(\dot{q}) = F_v\dot{q} + F_d \tag{4.7}$$

whose F_v represents the viscous friction while F_d is a dynamic friction term. Before interpreting the properties of the elements of the dynamics modeling, some mathematical background should be given.

Definition 4.2. A symmetric matrix is a matrix that is equal to its transpose. For such a matrix, $A \in \mathbb{R}^{n\times n}$ it can be said that the matrix A is positive definite if $x^TAx > 0$ for all $x \in \mathbb{R}^n - \{0\}$.

Definition 4.3. Consider a vector x, defined in the X vector space. The mapping of the x vector to $\mathbb{R}$ with the conditions represented by the norm function $\|.\|$ is as follows

- $\|x\| \geq 0$ for $\forall x \in X$ and $\|x\| = 0$ if $x = 0$
- $\|\alpha x\| = \|\alpha\|\|x\|$ for $\forall x \in X$ and all scalars α

Definition 4.4. Kronecker product indicates that tensor production of the matrices which is generally denoted by $\otimes$. For two matrices $A \in \mathbb{R}^{n\times m}$ and $B \in \mathbb{R}^{p\times q}$ then,

$$A \otimes B = [a_{ij}B] \in \mathbb{R}^{np\times mq} \tag{4.8}$$

Definition 4.5. A skew-symmetric matrix is equal to the negative of its transpose which satisfies $x^TAx = 0$ and $A = -A^T$ for a matrix A.

The model satisfies some properties below [59].

- $M(q)$ is symmetric and positive definite.
- $M(q)$ is bounded which satisfies the following the condition

$$\mu_1 I \leq M(q) \leq \mu_2 I \tag{4.9}$$

  where μ_1 and μ_2 are some scalars.

- The boundedness property of the inertia matrix may also be expressed as

$$\mu_1 I \leq \|M(q)\| \leq \mu_2 I \tag{4.10}$$

- It is possible to define a matrix such that $C_m(q, \dot{q})$ which satisfies

$$C(q, \dot{q}) = C_m(q, \dot{q})\dot{q} \tag{4.11}$$

where C_m can be defined as U^T - $\frac{1}{2}U$ for a matrix U which satisfies the following equation

$$U(q, \dot{q} = (I_n \otimes \dot{q}^T)\frac{\partial M}{\partial \dot{q}} \tag{4.12}$$

then the dynamics can be written by the skew-symmetric matrix as

$$C_m(q, \dot{q}) = \frac{1}{2}(\dot{M} + U^T - U) \tag{4.13}$$

and finally, the dynamics can be expressed as

$$M(q)\ddot{q} + C_m(q, \dot{q})\dot{q} + G(q) = \tau \tag{4.14}$$

4.2 Computed Torque Control

The main goal of exoskeleton robots and generally of all robot manipulators is following the desired trajectory well. The computed torque control method combines the dynamic model of the robot and tunable motion to create a control that provides improved precision and stability. This method is capable of obtaining a predictable movement by recording the dynamic properties of complex robotic systems. In this way, robots can be used more efficiently and reliably in various applications.

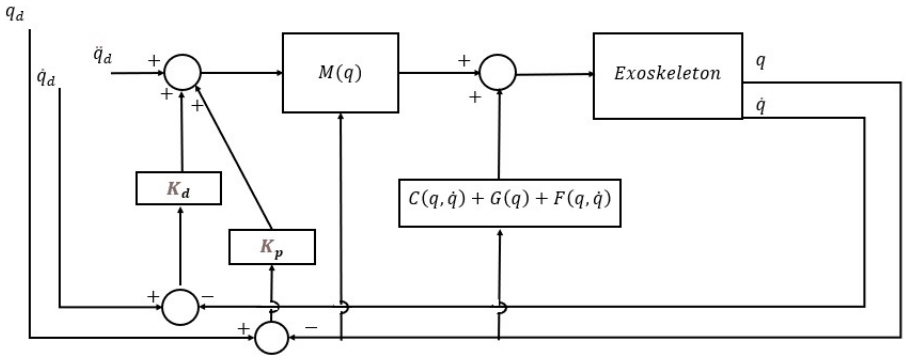


Figure 4.2 The block diagram of computed torque control

The dynamic model can be expressed as in Equation (4.6). This equation can be

redefined within a N matrix whose values represent gravity and other forces acting on the joints. The new equation:

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + N(q) = \tau \tag{4.15}$$

It can be also manipulated and reformed as:

$$\widehat{M}(q)\ddot{q} + \widehat{C}(q,\dot{q})\dot{q} + \widehat{N}(q) = \tau \tag{4.16}$$

where $\widehat{M}$, $\widehat{C}$, and $\widehat{N}$ are in their estimated forms.

$$\widehat{M}(q)\ddot{q} + \widehat{C}(q,\dot{q})\dot{q} + \widehat{N}(q) = \tau \tag{4.17}$$

The e, $\dot{e}$, and $\ddot{e}$ represent the error values for the distance, velocity and acceleration, respectively. Their equations might be written as in the following equations.

$$e = q_d - q \tag{4.18}$$

$$\dot{e} = \dot{q_d} - \dot{q} \tag{4.19}$$

$$\ddot{e} = \ddot{q_d} - \ddot{q} \tag{4.20}$$

PD control generates control signals by taking into account the position error of the system and the rate of error change. The proportional component K_p provides a proportional correction to the instantaneous position error and guides the system toward the target position. The derivative component K_d takes into account the error rate of change and reduces sudden movements and oscillations of the system, thus providing a smoother and more stable movement. PD computed torque control is especially effective in controlling robotic arms and manipulators with fast dynamics and shows superior performance in precise positioning and tracking tasks.

For defining the control input u to manipulate the acceleration of the joints explicitly.

$$\ddot{\theta} = u \tag{4.21}$$

$$\ddot{\theta}_d + K_d\dot{e} + K_p e = u \tag{4.22}$$

Therefore, PD computed torque can be represented as the following.

$$\tau = M(q)(\ddot{q}_d + K_d\dot{e} + K_p e) + N(q, \dot{q}) \tag{4.23}$$

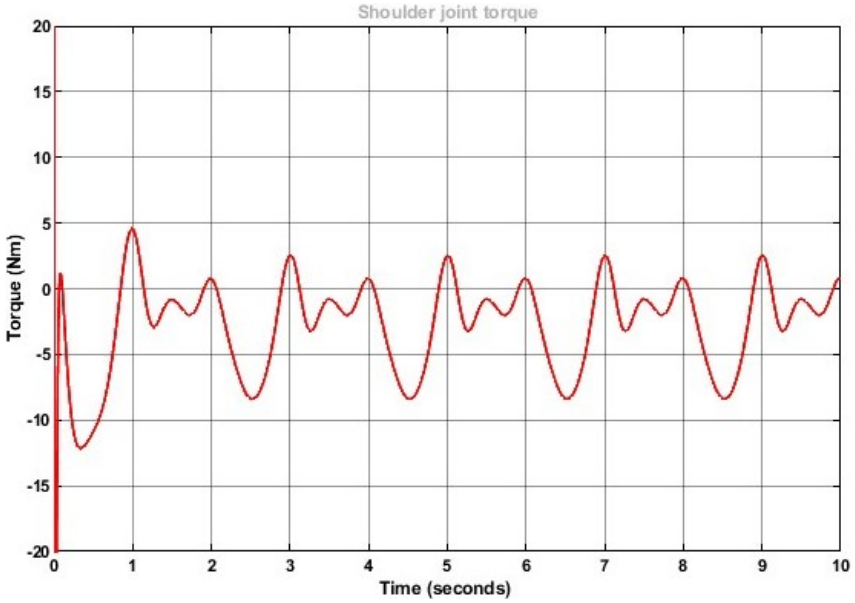
According to these equations, a PD-computed torque control can be simulated according to the arm parameters which are given on the following table. The simulations are prepared on $MATLAB/Simulink^{TM}$ environment. The block diagram for this environment and matrices in the equations above are both shown in *Appendix*.

Table 4.1 Parameters of the exoskeleton for computed torque control simulations

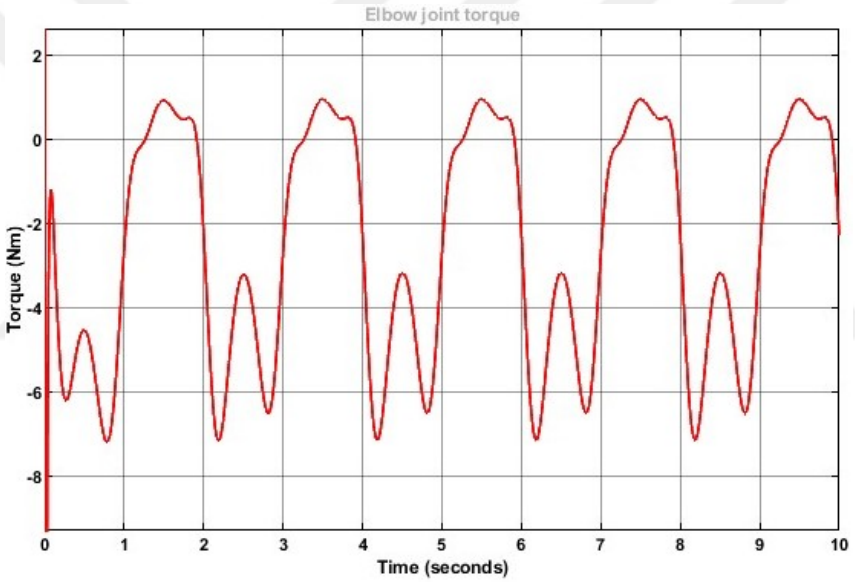
Variable	**Value**
K_p	100
K_d	50
Total mass	75 kg
Upper arm mass	1.98 kg
Lower arm mass	1.15 kg
Total height	1.75 m
Upper arm length	0.303 m
Lower arm length	0.275 m

The torque requirements according to the simulation can be depicted in Figure 4.3 for shoulder and elbow joints.

The torque values as the outcomes of the simulations have been seen with huge spikes at the very first interval. These spikes could occur since the initial conditions might have divergence from the desired trajectory, and then the simulation dynamics could give enormous torque values to get the initial desired conditions at the first time. Also, it has been depicted that each interval had local spikes. These were caused by the nature of the desired trajectory. If there are some repetitive or sudden changes in the desired trajectory, this can lead to spikes in the torques. To talk about controller performance, there are several metrics in the control theory. Although it cannot be feasible in the practical world, the step inputs can be used for the determination of such metrics. Two of those can be counted as overshoot and steady-state time. In control theory, overshoot refers to an output exceeding its final, steady-state value.



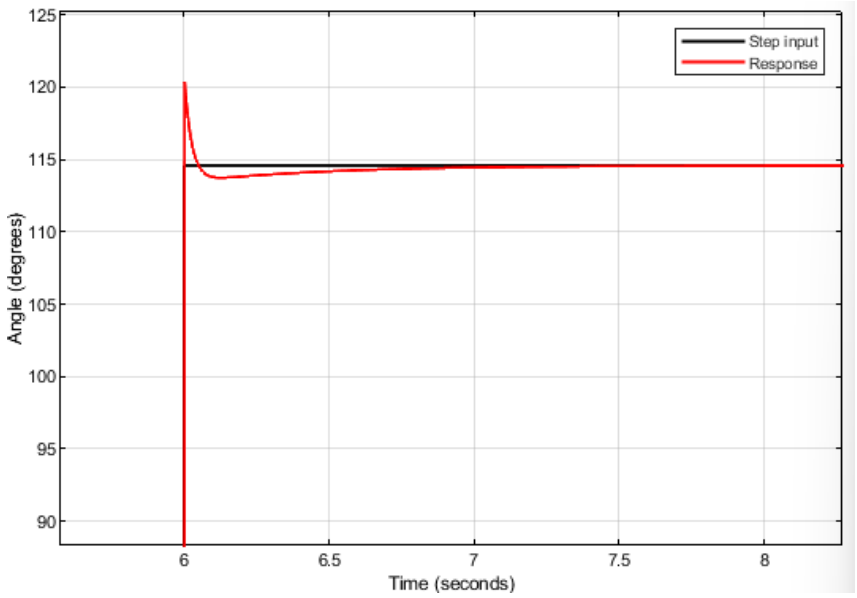
(a) Shoulder torque values



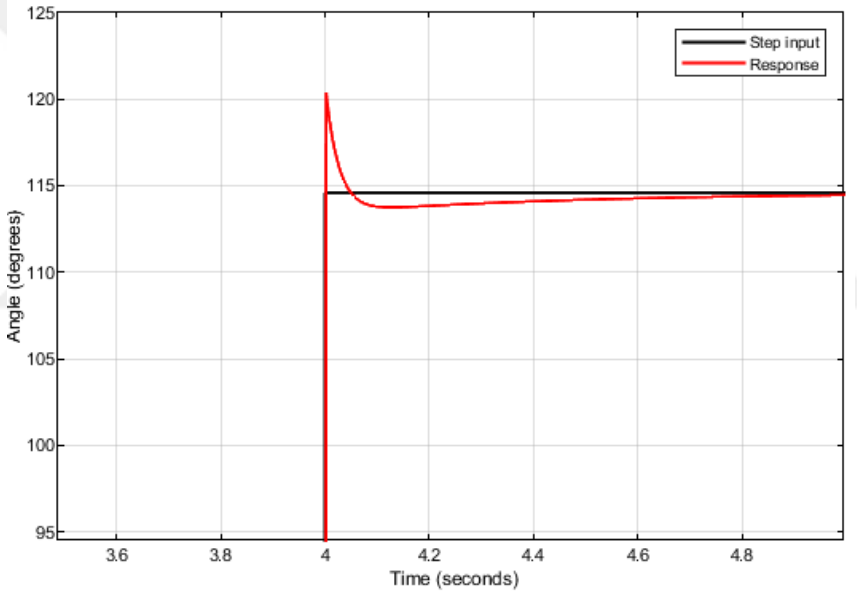
(b) Elbow torque values

Figure 4.3 Torque values for each joint

First, for the step time at 6th seconds during the 10 seconds simulation, a step input has been given into the shoulder joint, with K_p is 100 and K_d is 50. At this trial, the overshoot was determined and while tuning the PD controller gains, although the system has given better results in terms of less overshoot and more rapid steady-state response, the controller had still overshoot at the step time.



(a) Step response of the shoulder joint rotation at 6 seconds with lower PD gains



(b) Step response of the shoulder joint rotation at 4 seconds with higher PD gains

Figure 4.4 Step response of the shoulder rotation for different cases

The amount by which the response exceeds the final value, expressed as a percentage of the final value can be defined as overshoot. Overshoot value is useful for the comparison of the responses for the different control approaches.

$$OS = \left(\frac{out_{\max} - out_{\infty}}{out_{\infty}} \right) \times 100\% \tag{4.24}$$

Here, according to the step responses, even though different K_p and K_d gains have

been tried, the overshoot values vary between 4-5%.

The Integral of Squared Error (ISE) is a performance metric used to evaluate the quality of a control system. It is calculated by integrating the square of the error signal over time. The error signal $e(t)$ is the difference between the desired response and the actual response of the system. The ISE provides a measure of the total accumulated error over the entire response duration, giving more weight to larger errors.

$$\text{ISE} = \int_0^\infty e(t)^2 dt \tag{4.25}$$

Here, $e(t)$ is the error at time t.

For the higher PD gain case, first five seconds, the data can be extracted as follows:

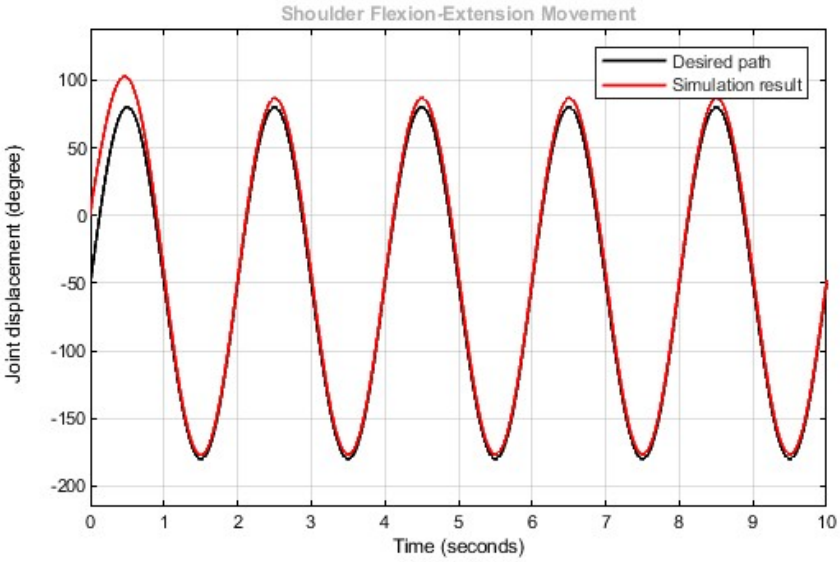
Table 4.2 Data points for step response ISE calculation

Time (s)	**Actual Response (deg)**	**Desired Response (deg)**	**Error (e(t))**	$e(t)^2$
0	0	115	115	13225
1	50	115	65	4225
2	90	115	25	625
3	110	115	5	25
4	114	115	1	1
5	115	115	0	0

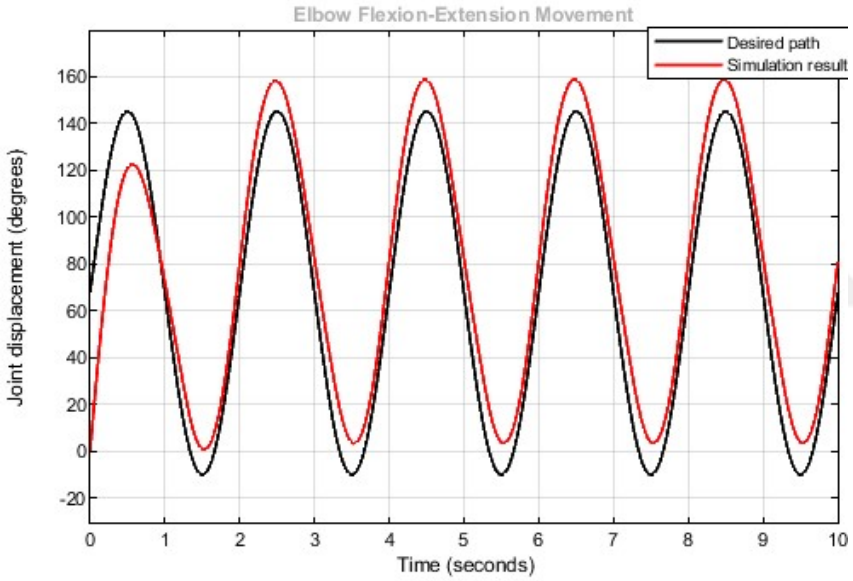
The trapezoidal rule can be used as a numerical integration method that approximates the definite integral of a function. After explaining the adaptive controller, ISE values will be depicted and compared.

The error values as the difference between desired and simulated trajectories are also shown in the following figure, for both joints. When these error values are interpreted, relatively high values can be seen at the first operation instances. These are caused by the torque adjustments for the desired path alignments. After all, the system depicted a small amount of error in rotation angles which are acceptable values for the repetitive rehabilitation mode. Furthermore, the assistive mode can operate well with these error values, since the position control should not be accurate but precise.

Here, another input has also been used for understanding the response of the system in the rapid or frequency-changed mode. When the elbow joint initializes with the sinusoidal signal which ranges in degrees -60 to 60 degrees, the frequency can be increased to imitate getting more rapid mode. 0.1 to 1 Hz frequency range, increased mode input, and response of the system can be seen in Figure 4.7 and the

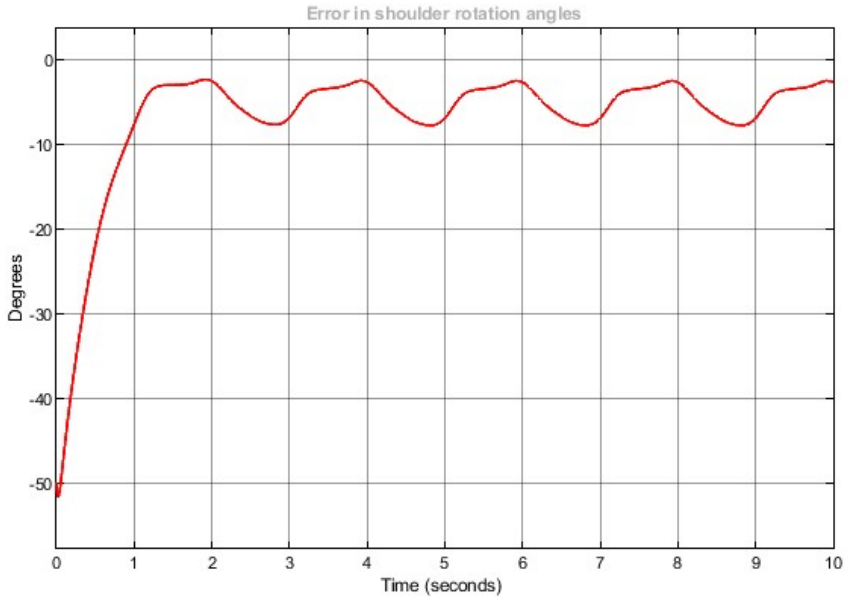


(a) The flexion-extension movement of the shoulder joint by the PD-computed torque controller

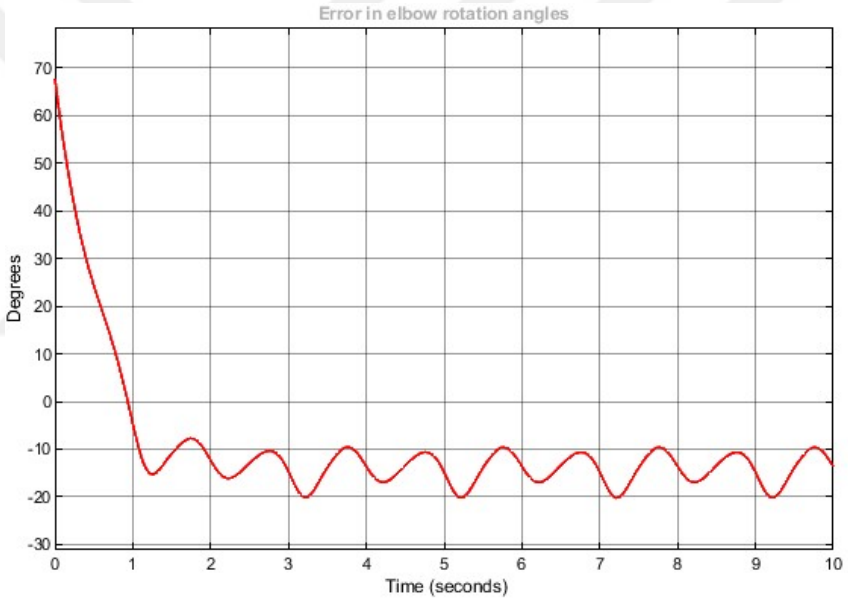


(b) The flexion-extension movement of the elbow joint by the PD-computed torque controller

Figure 4.5 Rotation angles for each joint by the PD-computed torque controller



(a) The shoulder joint error values for the PD-computed torque controller



(b) The elbow joint error values for the PD-computed torque controller

Figure 4.6 Error values in rotation angles for each joint for the PD-computed torque controller

related elbow joint rotation angle error for this increased frequency mode can be depicted in Figure 4.8

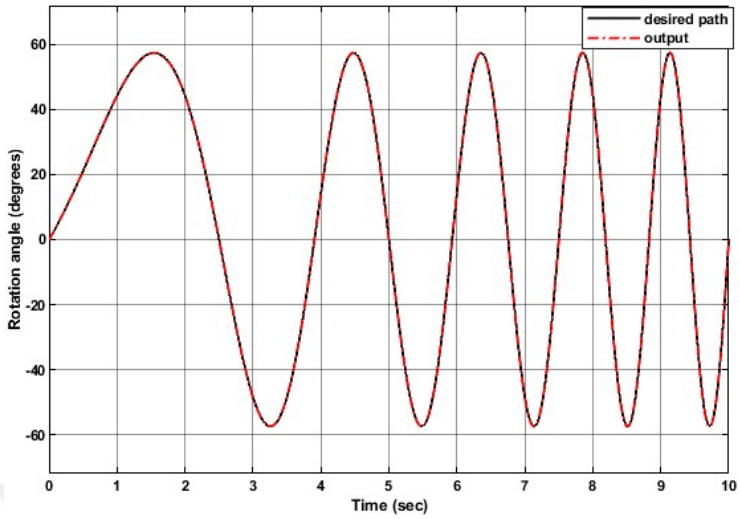


Figure 4.7 The increased frequency mode input on the elbow joint and the output of the model

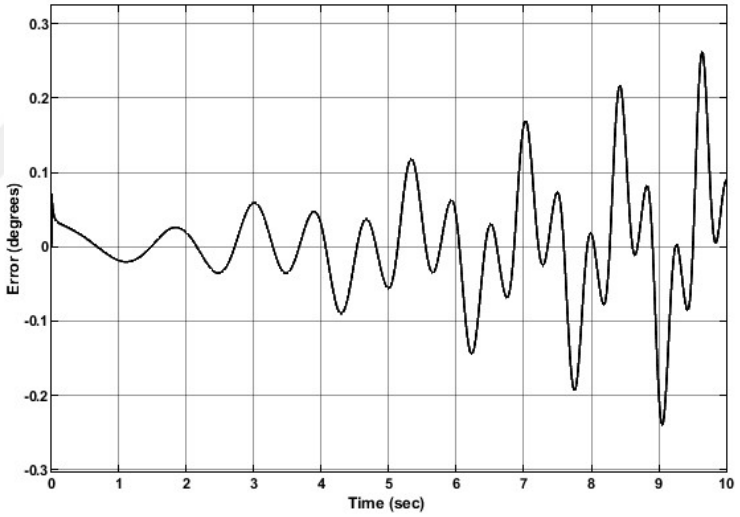


Figure 4.8 The increased frequency mode error value of the elbow joint path tracking

The error values can be minimized by adjusting the K_p and K_d values indeed. Even though, they can generally be determined by arbitrary trials, several methods to tune the gains have been proposed in the literature [60]. For instance, Ziegler and Nichols have proposed a pack of several rules for determining values of the proportional gain and derivative time based on the transient response characteristics of a given system [61]. At least, this method gives a starting point for the trials to determine proper gains for better results. Moreover, to reduce the number of trials, the dynamics have to be known to get limitations for the response, for example, limited overshoot, well-defined transient response time, etc.

In conclusion, although the error values are in the acceptable range, in real-time applications, the spikes on the first time intervals must be get rid of. Otherwise, the hardware, especially the actuators of the system might be damaged irreversibly. Therefore, the model reference adaptive control (MRAC) method has been applied to the system for real-time operations. The MRAC is a type of adaptive control where the controller parameters can be adjusted in real time to reach the reference model goals. This adjustable nature also makes the system applicable to different patient dynamics.

4.3 Model Reference Adaptive Control

Model Reference Adaptive Control (MRAC) is a control method to ensure optimal performance under uncertain conditions, especially for real-time applications. Through a dynamic comparison of the planned reference model and the actual system output, MRAC dynamically modifies control parameters to efficiently get rid of the disturbances. Even in the presence of unknown or unpredictable circumstances, the system can maintain stability and achieve the intended performance thanks to this adaptive method.

There are generally two main types of the adaptive controller: direct and indirect. A typical adaptive controller may be expressed as in Equation (4.26) [62].

$$u = k_x(t)x + k_r(t)r \tag{4.26}$$

In this equation; $k_x(t)$ and $k_r(t)$ are adjustable control gains and r indicates the reference trajectory, while x is output from the uncertain plant. MRAC is a type of direct adaptive approach. The general mechanism is based on adjusting the control gains by an adaptive law. In the presence of high system uncertainty, a direct adaptive control method effectively modifies the feedback control mechanism to cancel out any undesirable system uncertainty. A general overview of an MRAC can be depicted in the following figure.

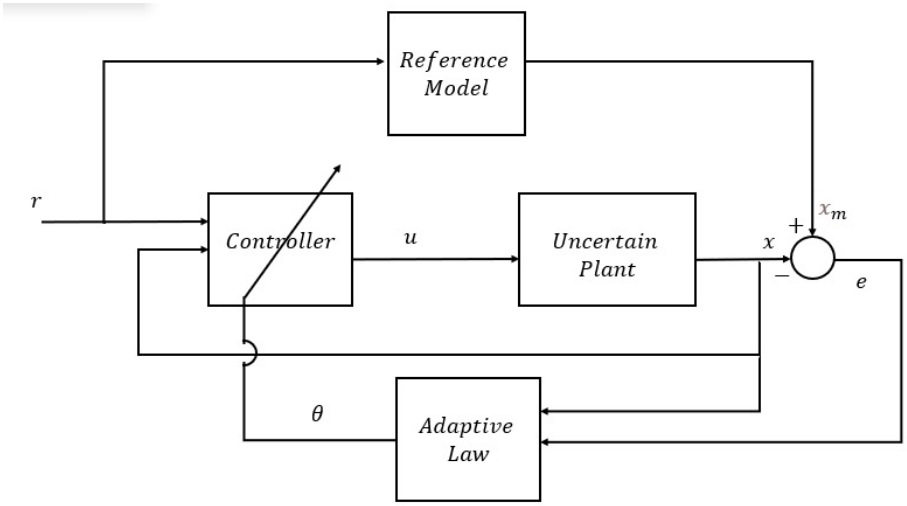


Figure 4.9 The block diagram of MRAC

Definition 4.6. ***Lyapunov-Based Stability Theorem***[63] If there exists a Lyapunov function $V(e, \theta)$ such that $\dot{V}(e, \theta) \leq 0$, then the error $e(t)$ is bounded and the adaptive control system is stable.

Proof. Consider a Lyapunov function $V(e, \theta)$ such that

$$V(e, \theta) > 0 \quad \text{and} \quad \dot{V}(e, \theta) \leq 0.$$

If $\dot{V}(e, \theta) \leq 0$, then $V(e, \theta)$ is non-increasing over time, ensuring that the error $e(t)$ and parameter deviations $\theta(t)$ remain bounded. Consequently, the system remains stable. ■

Lemma 4.1. ***Barbalat's Lemma****[64] If a function $f(t)$ is uniformly continuous and its time derivative $\dot{f}(t)$ is integrable and approaches zero as $t \to \infty$, then $f(t)$ approaches a finite limit as $t \to \infty$.*

Proof. Given that $\dot{f}(t)$ is integrable and approaches zero as $t \to \infty$, it implies that

$$\int_0^\infty \dot{f}(\tau)\, d\tau < \infty.$$

Since $f(t)$ is uniformly continuous, $f(t)$ does not exhibit unbounded oscillations and thus converges to a finite value as $t \to \infty$. ■

Theorem 4.2. ***Parameter Convergence*** *Given that the reference model and the plant share the same structure, and under the appropriate adaptive laws, the parameters $\theta(t)$ will converge to their true values as $t \to \infty$.*

To design an MRAC, it should be known that the left-hand side of Equation (4.15) can be linearly parameterized [65]:

Definition 4.7. The left-hand side of the dynamics equation can be linearly parameterized as the multiplication of a known regressor matrix $Y(q, \dot{g}, \ddot{q}) \in \mathbb{R}^{n \times r}$ with a parameter vector $p \in \mathbb{R}^n$

Finally, it can be said that Equation (4.15) can be rewritten as in Equation (4.27).

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) = Y(q, \dot{q}, \ddot{q})p \tag{4.27}$$

The controller based on the PD approach can be formed as in Equation (4.28).

$$\tau = \widehat{M}(\ddot{q}_d - K_d(\dot{q} - \dot{q}_d - K_p(q - q_d)) + \widehat{C}\dot{q} + \widehat{G} \tag{4.28}$$

Here, the $\widehat{M}, \widehat{C}$ and $\widehat{G}$ are the estimations of the M, C and G , respectively.

Then, the closed-loop system becomes Equation (4.29).

$$\widehat{M}(\ddot{e} + K_d\dot{e} + K_p e) = -\tilde{M}\ddot{q} - \tilde{C}\dot{q} - \tilde{G} \tag{4.29}$$

The estimation errors for each element can be expressed as in the following equation set.

$$\tilde{M} = M - \widehat{M} \tag{4.30}$$

$$\tilde{C} = C - \widehat{C} \tag{4.31}$$

$$\tilde{G} = G - \widehat{G} \tag{4.32}$$

If $\tilde{p} = p - \widehat{p}$ is valid and $\widehat{p} \in \mathbb{R}^r$ as an estimate of p in equation 4.2, and by assuming that $\widehat{M}$ is invertible for all time $t \geq 0$, the last equation to define the controller can be reformed into that:

$$\ddot{e} + K_d\dot{e} + K_p e = -\widehat{M}^{-1}Y(q, \dot{q}, \ddot{q})\tilde{p} \tag{4.33}$$

Here Y regressor matrix derivation is crucial for the derivation of the adaptive controllers. Conventional adaptive control methods have a common assumption that the uncertain parameters should be constant or slowly time-varying. The

determination of the regressor matrix can be complex and time-consuming, especially for the higher degree of freedom of robotic arms. For a two-link robotic arm, the parameter matrix p and the regressor matrix Y can be seen in the Appendix.

Due to the fact that the regressor depends on the joint position, velocity, and acceleration, updating it is mandatory for each cycle. These computational loads can be problematic for practical implementation. Moreover, the time-varying uncertainties without knowing their limits occurs often. Thus, a robust controller might not be feasible to eliminate them. The function approximation technique (FAT) without using a regressor has been proposed to overcome this issue [66]. This method also does not require acceleration measurements and inversion of the estimated inertia matrix. Basically, the method is for describing the system uncertainties using a finite number of linear combinations of orthonormal basis functions with some unknown constant weighting factors. By choosing suitable update laws for the weighting factors, it can be ensured that the time derivative of a chosen Lyapunov function candidate is negative semi-definite. This brings about analyzing the impact of approximation errors on system performance.

Definition 4.8. Orthonormal basis functions are a set of functions that satisfy two key properties: orthogonality and normalization.

To create such a controller, first, an error vector can be defined as:

$$s = \dot{e} + \Lambda e \tag{4.34}$$

where $e = q - q_d$ is the tracking error in the joint space and

$$\Lambda = diag(\lambda_1, \lambda_2, ..., \lambda_n) \tag{4.35}$$

with $\lambda_i > 0$ for all $i = 1, 2, ..., n$.

Then, the model can be rewritten as in Equation (4.36).

$$M\dot{s} + Cs + G + M\ddot{q}_d - M\Lambda\dot{e} + C\dot{q}_d - C\Lambda e = u \tag{4.36}$$

The closed loop becomes Equation (4.37).

$$M\dot{s} + Cs + K_d s = 0 \tag{4.37}$$

Now, a Lyapunov function candidate can be defined as

$$V = \frac{1}{2} s^T M s \tag{4.38}$$

while taking the derivative of the function along the trajectory and using the skew-symmetric property on $\dot{M} - 2C$ the following equation satisfies the tracking error converges asymptotically.

$$\dot{V} = -s^T K_d s \leq 0 \tag{4.39}$$

when $s \to 0$ as $t \to \infty$.

However, under the uncertainties, after some rearrangements, the closed-loop dynamics can be represented as the following.

$$M\dot{s} + Cs + K_d s = (\widehat{M} - M)(\ddot{q}_d - \Lambda\dot{e}) + (\widehat{C} - C)(\dot{q}_d - \Lambda e) + (\widehat{G} - G) \tag{4.40}$$

FAT presentations of the matrices M, C, and G with the basis functions are defined as Equation (4.41), Equation (4.42), and Equation (4.43).

$$M = W_M^T Z_M \tag{4.41}$$

$$C = W_C^T Z_C \tag{4.42}$$

$$G = W_G^T Z_G \tag{4.43}$$

Ws are weighting matrices and Zs are matrices of basis functions. The corresponding estimations should be defined as the following equation set.

$$\hat{M} = \hat{W}_M^T Z_M \tag{4.44}$$

$$\hat{C} = \hat{W}_C^T Z_C \tag{4.45}$$

$$\hat{G} = \hat{W}_G^T Z_G \tag{4.46}$$

Then finally, the closed-loop dynamics become Equation (4.47).

$$M\dot{s} + Cs + K_d s = \tilde{W}_M^T Z_M(\ddot{q}_d - \Lambda\dot{e}) + \tilde{W}_C^T Z_C(\dot{q}_d - \Lambda e) + \tilde{W}_G^T Z_G \tag{4.47}$$

The candidate Lyapunov function can be defined as in Equation (4.48):

$$V(s,\tilde{W}_M,\tilde{W}_C,\tilde{W}_G)=\frac{1}{2}s^TMs+\frac{1}{2}\text{Tr}(\tilde{W}_M^TQ_M\tilde{W}_M+\tilde{W}_C^TQ_C\tilde{W}_C+\tilde{W}_G^TQ_G\tilde{W}_G) \tag{4.48}$$

Here, $Q_D \in \mathbb{R}^{n^2\beta_D\times n^2\beta_D}$, $Q_C \in \mathbb{R}^{n^2\beta_C\times n^2\beta_C}$, and $Q_g \in \mathbb{R}^{n\beta_g\times n\beta_g}$ are positive definite matrices. The notation $\text{Tr}(\cdot)$ denotes the trace operation of matrices. This function also satisfies the relationship in Equation (4.37) [65].

In a nutshell, uncertainties have been represented by the Fourier series. As it is known, any periodic function can be represented by a Fourier series. Basically, any function can be derived with the summation of several cosine and sine functions.

Definition 4.9. A Fourier series is a way to represent a periodic function as a sum of sine and cosine functions. It decomposes a periodic function into its constituent frequencies. The Fourier series of a function $f(t)$ with period T is given by:

$$f(t)=a_0+\sum_{n=1}^{\infty}\left(a_n\cos\left(\frac{2\pi nt}{T}\right)+b_n\sin\left(\frac{2\pi nt}{T}\right)\right) \tag{4.49}$$

where the coefficients a_n and b_n are calculated as:

$$a_0=\frac{1}{T}\int_0^T f(t)\,dt$$

$$a_n=\frac{2}{T}\int_0^T f(t)\cos\left(\frac{2\pi nt}{T}\right)dt,\quad n\geq 1$$

$$b_n=\frac{2}{T}\int_0^T f(t)\sin\left(\frac{2\pi nt}{T}\right)dt,\quad n\geq 1$$

Fourier series can be used for generating basis functions. As already mentioned before, Z matrices for estimations and update laws, it can be said they might involve basis functions as Fourier series.

Finally, in light of this information, the controller equation of this regressor-free approach is:

$$\tau=\hat{M}v+\hat{C}v+\hat{G}+\hat{W}_M^TZ_Dv+\hat{W}_C^TZ_Cv+\hat{W}_G^TZ_G-K_ds \tag{4.50}$$

where v known signal vector, s error term are:

$$v = \dot{q}_d - \Lambda e$$

$$s = \dot{q} - \dot{q}_d + K_v(q - q_d)$$

and adaptive laws are finally obtained as:

$$\dot{\hat{W}}_M = -Q_M Z_M v^T s + \sigma_M \hat{W}_M \tag{4.51}$$

$$\dot{\hat{W}}_C = -Q_C Z_C v^T s + \sigma_C \hat{W}_C \tag{4.52}$$

$$\dot{\hat{W}}_g = -Q_g z_g s + \sigma_g \hat{W}_g \tag{4.53}$$

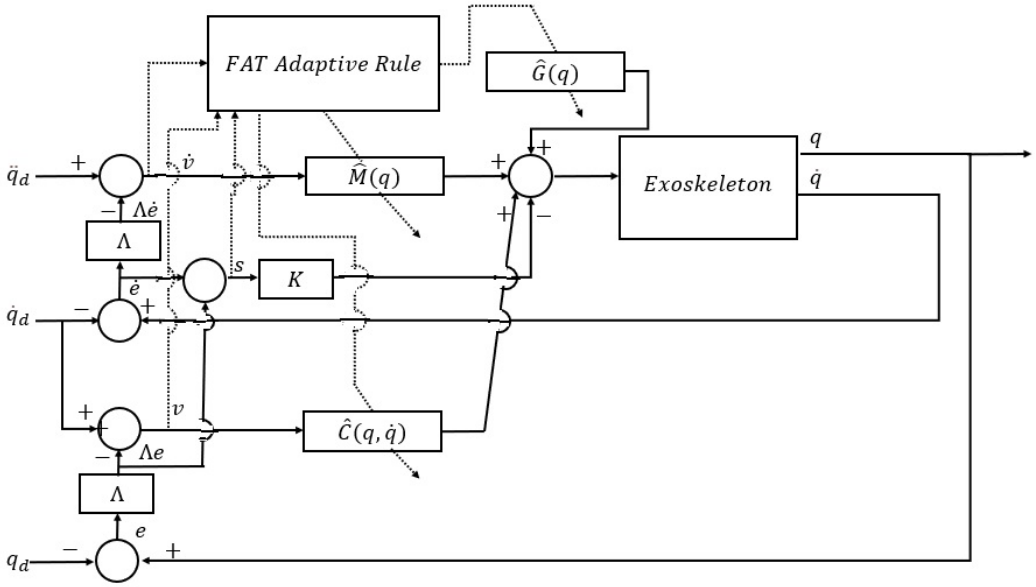
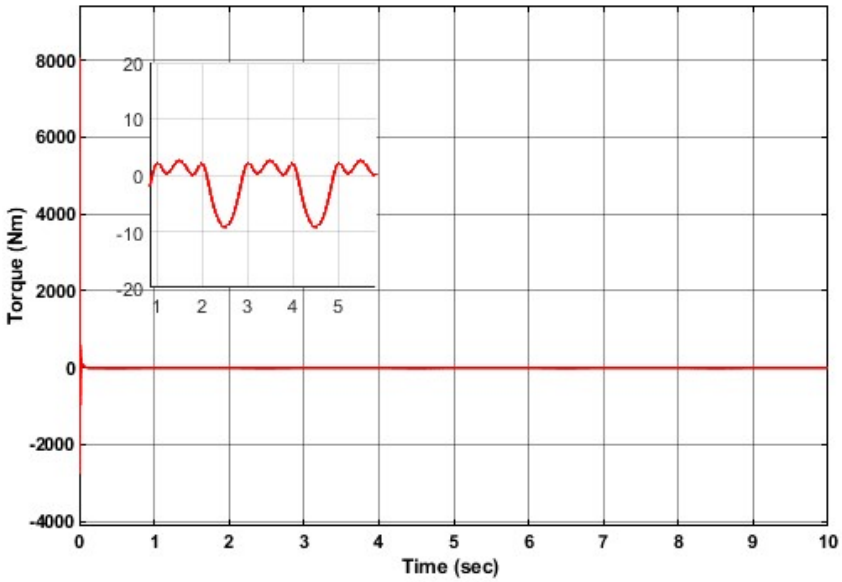
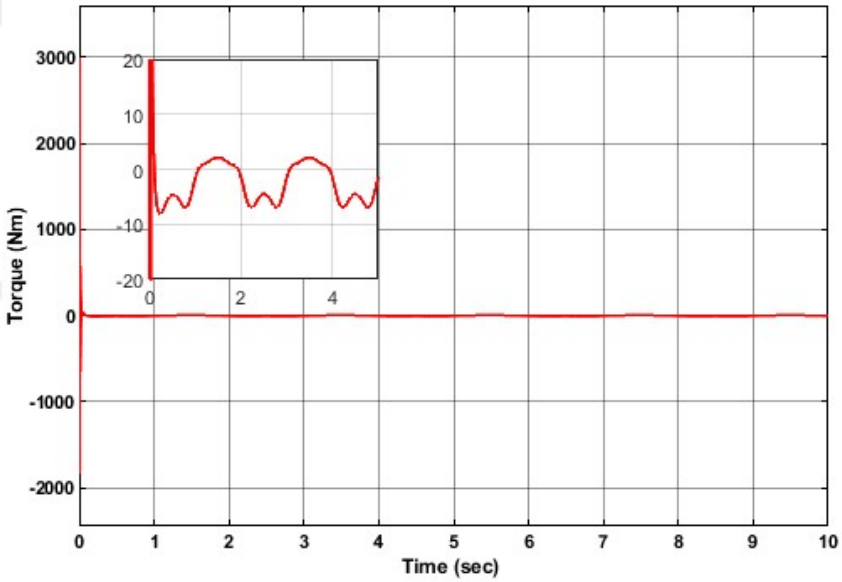


Figure 4.10 The block diagram of FAT-based adaptive controller

All related terms, matrices, and parameters have been mentioned in the Appendix. For the same parameters and their values of mass, length, etc. according to the matrices and equations on the adaptive controller which are shown in the Appendix, the simulation results can be seen in the following figures.



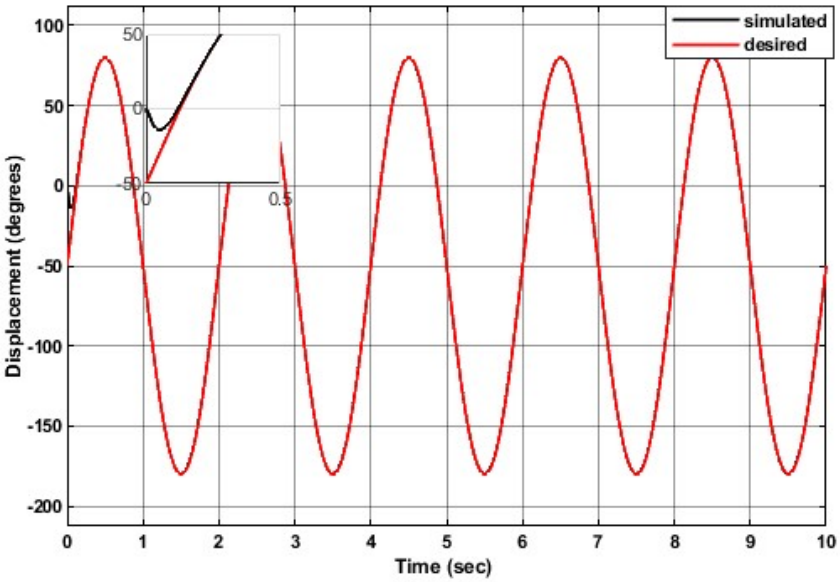
(a) The controller output as the required torque for the flexion-extension movement of the shoulder joint



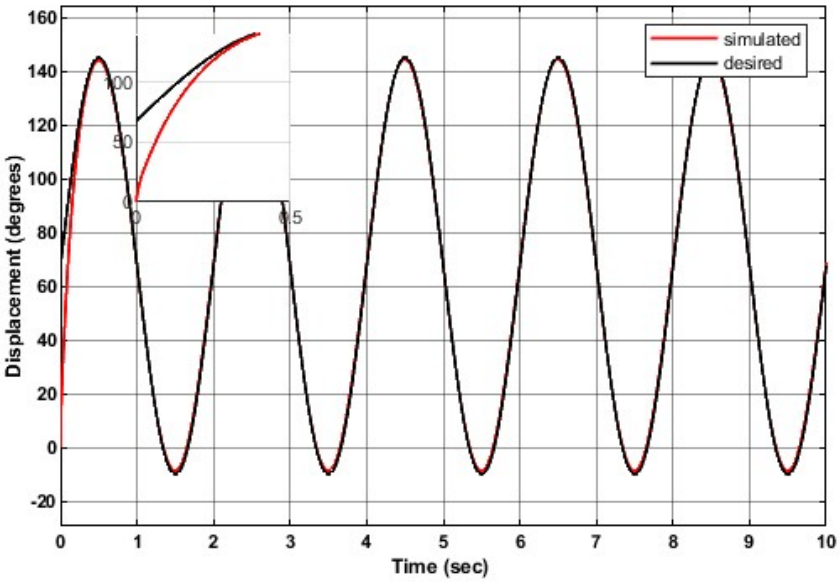
(b) The controller output as the required torque for the flexion-extension movement of the elbow joint

Figure 4.11 Torque values for each joint by the adaptive controller

In terms of tracking the desired path for flexion-extension movement for each joint:



(a) The flexion-extension movement of the shoulder joint by the adaptive controller



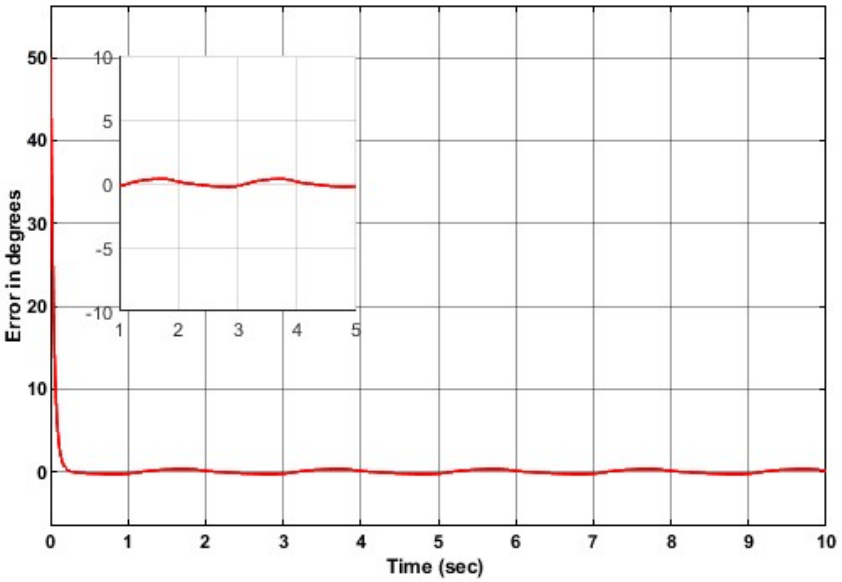
(b) The flexion-extension movement of the elbow joint by the adaptive controller

Figure 4.12 Rotation angles for each joint by the adaptive controller

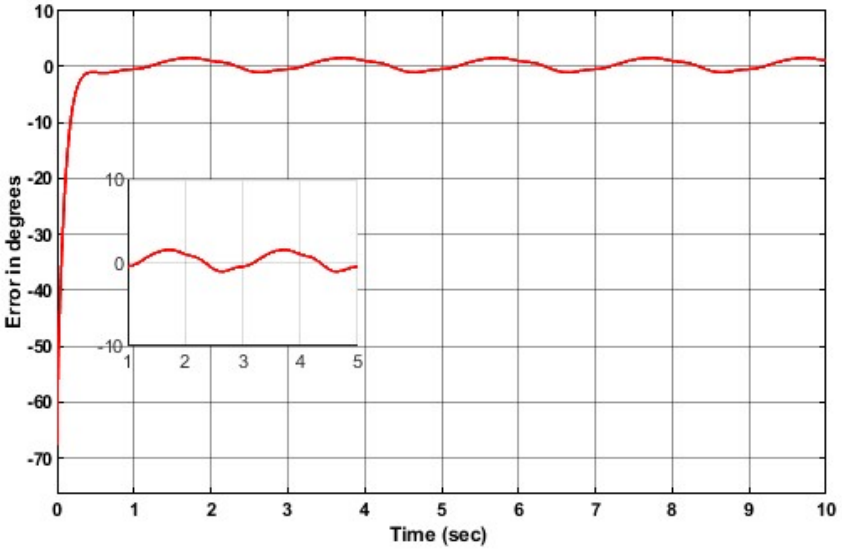
Likewise the previous simulation results of the other controller, the error values as the difference between desired and simulated trajectories are also shown in the following figure, for both joints. These error values converge to zero in small-time instances, which is the making goal of the controller. However, here again, ISE values are given in the following table to see the error convergence numerically. After ending the first second during the movement, error has been approached infinitesimally small error as zero so the table depicts the values between 0 and 2 seconds.

Table 4.3 ISE values for elbow movement under the adaptive controller

Time (s)	Actual Response (deg)	Desired Response (deg)	Error (e(t))	$e(t)^2$
0.05	0	90	90	8100
0.1	100	110	10	100
0.5	145	145	0	0
1	60	60	0	0



(a) The shoulder joint error values for the adaptive controller case



(b) The elbow joint error values for the adaptive controller case

Figure 4.13 Error values in rotation angles for each joint for the adaptive controller case

As the one applied for the computed torque controller block, here, the increased frequency case also has been examined. Again, under the same conditions, the

applied path and tracked trajectory can be seen in Figure 4.14. Besides, the error value for this elbow joint path tracking can be depicted in Figure 4.15.

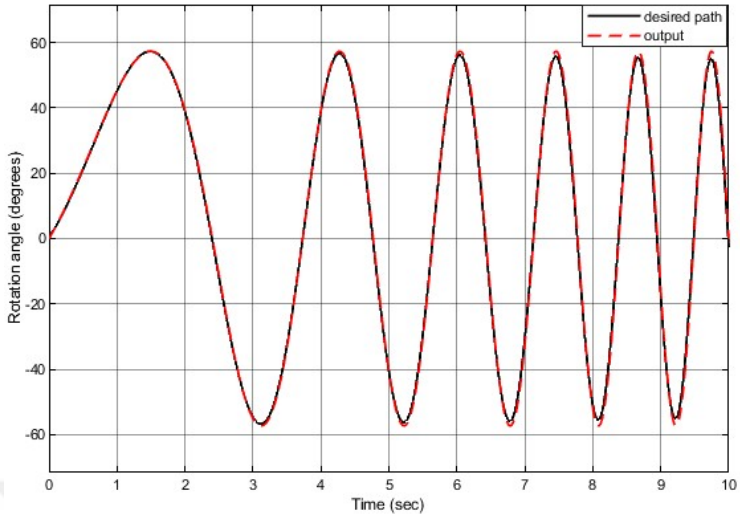


Figure 4.14 The increased frequency mode on the elbow joint and the output of the model for the adaptive controller case

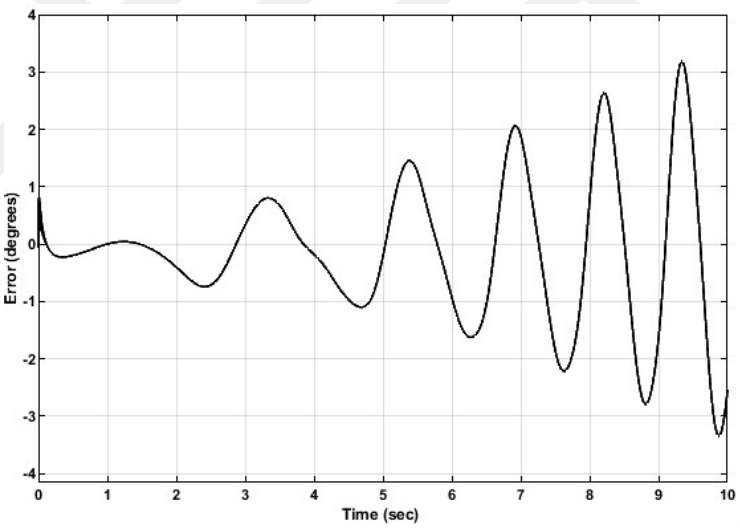


Figure 4.15 The increased frequency mode error on the elbow joint for the adaptive controller case

For upper limb exoskeletons, adaptive controllers provide a number of benefits over computed torque controllers, the most important one can be counted as being their resistance to parameter fluctuations, then, increased performance, increased safety, adaptability, and long-term appropriateness. Adaptive torque controllers are a better option for the changing and dynamic environment of human-exoskeleton interaction than computed torque controllers, even though the former can be useful in situations with well-understood and stable dynamics.

5
IMPLEMENTATION

This chapter details the real-time implementation of the upper limb exoskeleton driven by EMG sensor signal acquisition. Following successful simulations, the focus shifts to the practical realization of the system, encompassing the mechanical design of the entire system, signal acquisition by the sEMG sensors and its processing, adaptive control strategies, and integration with the exoskeleton framework. The aim is to achieve precise and responsive control of the exoskeleton through real-time EMG signal interpretation, ensuring rehabilitation tasks for flexion and extension movement of the shoulder and the elbow joint.

5.1 Mechanical Design and Electronic Components

The mechanical design of the upper limb exoskeleton involves a comprehensive approach to ensure functionality, durability, and user comfort for the rehabilitation tasks. Since one of the main goals for the thesis has been determined as designing and implementing an adjustable and financially accessible exosuit, the mechanical design has been achieved by 3D printing. Here, first of all, CAD (computer-aided design) based parts can be seen. Then, the figures of the 3D-printed parts are shown. After that, the torque requirements of targeted basic movements are interpreted and the designed transmission parts are depicted.

5.1.1 CAD of the Exosuit and Material Selection

The design and development of the upper limb exosuit began with the creation of detailed CAD models. These models served as the blueprint for the physical structure of the exosuit, allowing for precise planning and optimization of each component. The design has been operated in $Solidworks^{TM}$.

The design mainly has three parts: the shoulder complex, the upper arm, and the lower arm.



(a) The CAD of the shoulder complex

(b) The CAD of the upper arm

(c) The CAD of the elbow connection

(d) The CAD of the lower arm

Figure 5.1 The CAD parts of the system

In $Solidworks^{TM}$ environment, materials can be assigned manually or production material can be chosen automatically. So, the material specifications can be seen in the following figure:

Property	Value	Units
Shear Modulus	318.9	N/mm^2
Mass Density	1020	kg/m^3
Tensile Strength	30	N/mm^2
Compressive Strength		N/mm^2
Yield Strength	65	N/mm^2
Thermal Expansion Coefficient		/K
Thermal Conductivity	0.2256	W/(m·K)
Specific Heat	1386	J/(kg·K)

Figure 5.2 The material properties window on the CAD software

Polylactic Acid (PLA), a biodegradable polymer, has been selected as the production material. Because of its simplicity of use and environmental friendliness, it is one of the most often used materials for 3D printing. Because PLA melts at a lower temperature than many other polymers, it can be printed on the majority of 3D printers without the need for a heated bed. While not as strong as some other plastics like ABS (Acrylonitrile Butadiene Styrene), PLA offers sufficient strength for many applications, especially prototypes and parts that don't require high mechanical stress. PLA, while popular and easy to use, does have some drawbacks. It is more brittle compared to ABS and other more durable plastics, making it less suitable for parts that require high-impact resistance or flexibility. PLA also has a lower glass transition temperature, which means it can deform at relatively low temperatures, making it unsuitable for high-heat applications. Additionally, PLA is sensitive to moisture; it can absorb water from the air, which can negatively affect its print quality if not stored properly.

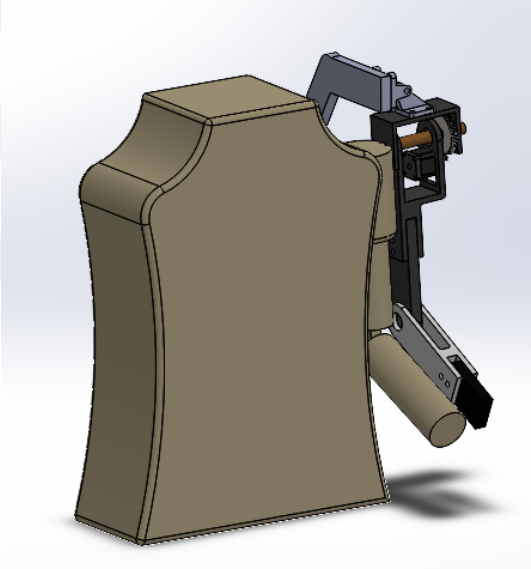


Figure 5.3 Overview of the CAD

5.1.2 Strength Test Simulation and Production

In $Solidworks^{TM}$ environment, the software allows to check the material strength under the mesh-force analysis. Since the shoulder complex shaft would be the most vulnerable place of the exoskeleton, the strength test has been only operated for it. Due to the weight of the arm itself, approximately 40 N might be applied to it at the resting position. The weight of an average adult arm is approximately 5% of the total body weight so for a 75 kg adult, the test has been evaluated under this amount of force. As a result, the morphologic changes have not occurred, therefore the material selection and design process have been achieved successfully.

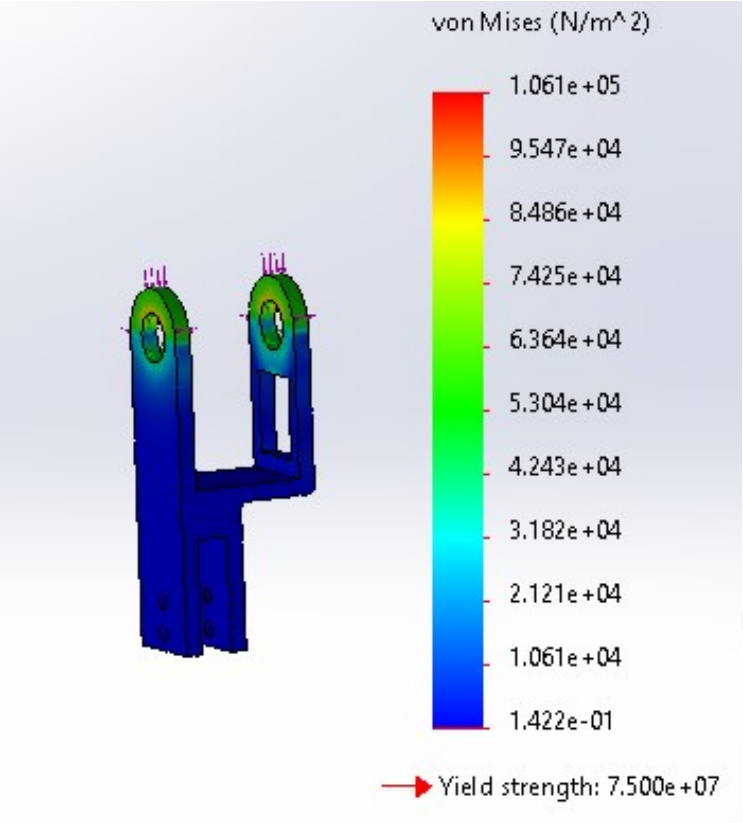


Figure 5.4 Strength test of the CAD

The 3D-printed parts of the exoskeleton can be seen in the following figures:

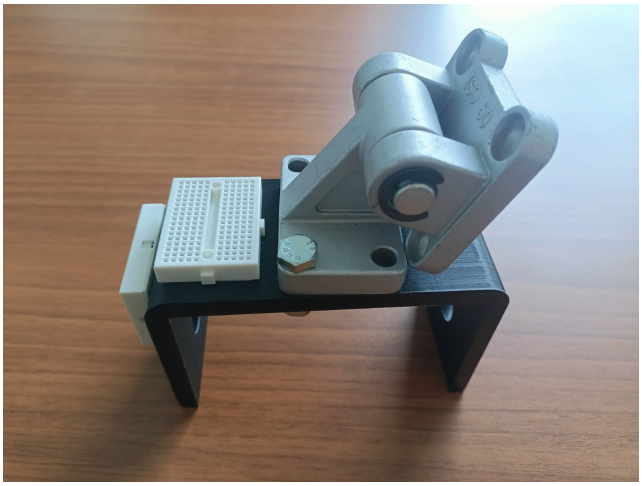


Figure 5.5 The produced shoulder complex



Figure 5.6 The produced upper arm

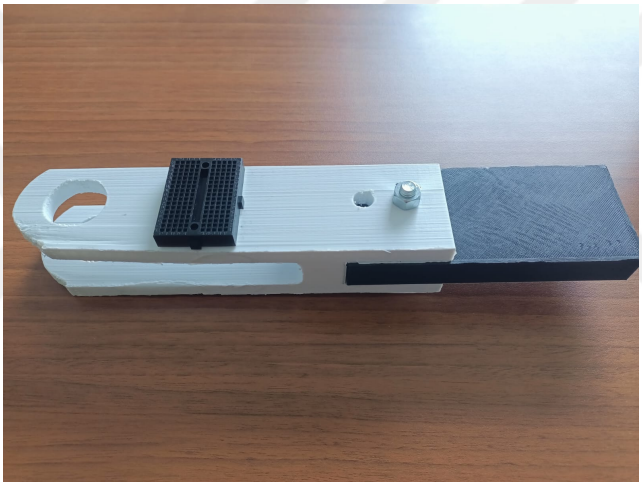


Figure 5.7 The produced lower arm

5.1.3 Torque Requirements and Actuator-Transmission System

Stepper motors are ideal for exoskeleton applications due to their precision, control, and reliability. They provide accurate positioning and repeatability of movement, which is crucial for the precise control required in exoskeletons to replicate human motion.

The Nema17 Step Motor BYGHW609 A, with a length of 40mm, is a compact and powerful step motor. It offers precise movement control with a step angle of 1.8 degrees, making it suitable for applications requiring accurate positioning. With a current rating of 1.7A, this motor delivers high performance and robust operation. Its strong torque output ensures the effortless handling of heavy loads, providing stable and reliable performance.



Figure 5.8 The stepper motor

The motor driver module is an A4988 2A stepper motor driver. It has dimensions of 20.32 mm x 15.24 mm and a pin spacing of 2.54 mm. It supports a motor supply current of less than 1A and a motor supply voltage range of 8 to 36V. The module features automatic current control and current decay control, ensuring efficient and stable operation of the motor by adjusting the current flow and managing the current reduction during motor operation.

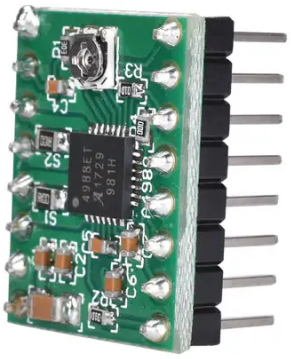


Figure 5.9 A4988 stepper motor driver

The typical holding torque for a Nema17 stepper motor like the BYGHW609 A is around 40 to 45 Ncm or 0.4 to 0.45 Nm. This holding torque ensures the motor can maintain its position effectively under load. The torque required for flexion and extension movements of the arm varies depending on the specific conditions and tasks being performed. Research indicates that for tasks involving lifting and placing during standing, the coordination of muscle activity is crucial for managing the required torque. Typically, during such activities, the torque needed for elbow

flexion can range from 2 to 8 Nm, depending on the speed and load [67]. Therefore, to enhance the torque ability of the actuator system, the transmission system has been designed. The spur gears have been designed and printed. Within the 70:10 ratio, almost 3.5 Nm has been achieved.



Figure 5.10 Gears for shoulder and elbow joint shafts

5.1.4 Microcontroller Board and Its Software

A powerful development tool, the STM32F407G-DISC1 Discovery board is made to facilitate quick prototyping and development for a variety of uses. This board has a Cortex-M4 core with FPU, 1 MB of Flash memory, and 192 KB of RAM, and it is powered by the powerful STM32F407 microcontroller. Specifically created for STM32 microcontrollers, STM32CubeIDE is a feature-rich integrated development environment (IDE) that provides a wide range of tools and functions for productive programming. Both have been used to develop EMG acquisition from the sensor card and to drive the motor.

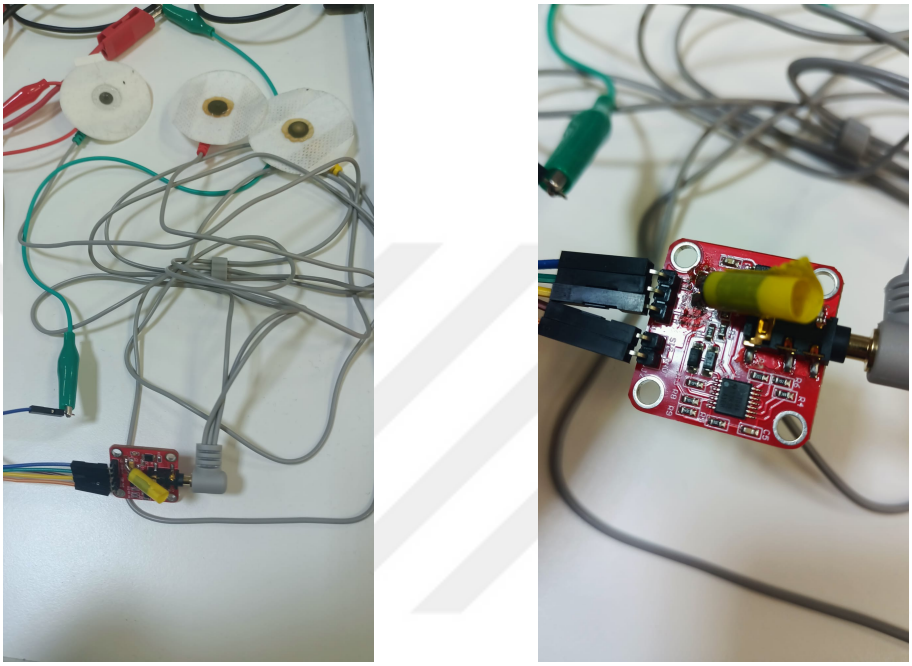


(a) An STM32F407G Board [68] **(b)** A screenshot from STMCubeIDE

Figure 5.11 The data acquisition board and its software

5.1.5 EMG Signal Acquisition Components

In this thesis, an AD8226-based EMG sensor to plot an EMG graph with the help of an STM32 board has been used. The AD8226 is an instrumentation amplifier commonly utilized in sensor design. It offers gain settings ranging from 1 to 1000, making it highly versatile. This amplifier is extensively employed in various fields including medical instrumentation, bridge amplifiers, industrial process control, and portable data acquisition systems.



(a) Sensor card with electrodes **(b)** Muscle signal card

Figure 5.12 EMG signal acquisition components

It is essential to implant electrodes on the upper limb in order to precisely detect muscle activation during flexion and extension motions. EMG signals, which are necessary for operating prosthetics and exoskeletons, must be reliably detected through proper placement. Electrodes are usually placed above the triceps brachii (TB) and biceps brachii (BB) muscles to record signals related to the flexion and extension of the arm. The triceps brachii, which are found on the back of the upper arm, aid in extension, while the biceps brachii, which are found on the front of the upper arm, are mostly in charge of flexion. The electrodes must be positioned precisely along the muscle fibers, staying away from joints and high adipose tissue (high body fat content) regions that may obstruct the signal.

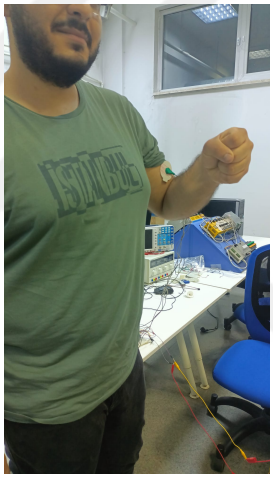


Figure 5.13 Electrode placement on the upper limb [21]

According to the literature, the data acquisition sessions have been realized. The healthy individual has made flexion and extension movements.



(a) Flexion of elbow



(b) Extension of elbow



(c) BB electrode placement



(d) TB and Ref electrode placement

Figure 5.14 EMG acquisition experiments

6
CONCLUSION

This thesis presents the design and implementation of an upper limb exoskeleton driven by EMG signals, highlighting significant advancements in rehabilitation technology. The project integrates various disciplines, including biomechanics, neural networks, control systems, and mechanical design, to create a robust and effective rehabilitation device.

The first critical aspect of this project is the acquisition and processing of EMG signals. EMG signals are bioelectrical signals generated by muscle activity, and their accurate acquisition is paramount for controlling the exoskeleton. The project employs advanced signal processing techniques to filter and amplify the raw EMG signals, ensuring that the data used for control is both accurate and reliable. The pre-processing steps, which include noise reduction and signal normalization, are crucial for minimizing errors and enhancing the quality of the data fed into the control algorithms.

A significant part of this thesis is the development and implementation of a neural network for classifying EMG signals. The Levenberg-Marquardt algorithm is employed for training the neural network, which proves to be an effective method for this application. The neural network successfully classifies EMG signals into different muscle activity patterns, which are then used to drive the exoskeleton.

The overall accuracy achieved is 87.7%, indicating the neural network's robustness in handling complex EMG data.

The mechanical design of the exoskeleton is another vital component of this project. The exoskeleton is designed to be lightweight, ergonomic, and customizable, with several parts 3D printed to reduce costs and enhance user-specific customization. The design also ensures that the device can accommodate a wide range of users, making it suitable for various rehabilitation scenarios. The exoskeleton's joints and links are meticulously designed to mimic the natural movement of the human arm,

providing effective support during rehabilitation exercises.

The thesis also introduces an adaptive controller based on Fourier series basis functions, which enhances the ability of the exoskeleton to adapt to individual users' anatomical variations. The controller's design ensures smooth and precise movements, which are essential for effective rehabilitation. The adaptive controller's performance is validated through extensive simulations demonstrating its superiority over traditional control methods. The ability to adapt to different users' needs is a significant advancement, making the exoskeleton more versatile and effective in various rehabilitation settings.

One of the key areas for future research is the further refinement and optimization of adaptive controllers for upper limb exoskeletons. While the current implementation demonstrates significant improvements in terms of adaptability and performance, additional work is needed to enhance the robustness of the control algorithms, especially in dynamic and unpredictable environments. Future studies could explore advanced machine learning techniques to improve the accuracy and responsiveness of the control system, as well as investigate the integration of additional sensors to provide more comprehensive feedback on user movements and exoskeleton performance. Moreover, real-time implementation could be enhanced by developing more efficient algorithms that reduce computational load and improve the system's responsiveness. Besides, expanding the dataset to include a wider range of muscle conditions and movements could improve the generalizability and robustness of the neural network model. Future research could also explore the real-time application of the neural network in a clinical setting, providing valuable feedback for rehabilitation and monitoring the progress of patients undergoing therapy with the exoskeleton.

To sum up, the creation of an upper limb exoskeleton powered by electromyography exhibits encouraging prospects for improving assistive technology and rehabilitation. Through the integration of complex control algorithms which is FAT-based MRAC, this research enhances the accuracy and functionality of exoskeleton systems, providing substantial advantages for people with upper limb limitations.

REFERENCES

[1] G. Britain, "Work-related musculoskeletal disorders statistics in great britain, 2021," *Hse. Gove. Uk,(March)*, pp. 1–22, 2021.

[2] S. Mendis, "Stroke disability and rehabilitation of stroke: World health organization perspective," *International Journal of stroke*, vol. 8, no. 1, pp. 3–4, 2013.

[3] S. Xie *et al.*, "Advanced robotics for medical rehabilitation," *Springer tracts in advanced robotics*, vol. 108, no. 1, p. 357, 2016.

[4] S. S. Hacısalihzade, *Biomedical applications of control engineering*. Springer, 2013, vol. 441.

[5] H.-Y. Hsu, H.-Y. Chiu, T.-S. Kuan, C.-L. Tsai, F.-C. Su, L.-C. Kuo, "Robotic-assisted therapy with bilateral practice improves task and motor performance in the upper extremities of chronic stroke patients: A randomised controlled trial," *Australian occupational therapy journal*, vol. 66, no. 5, pp. 637–647, 2019.

[6] M.-S. Kim, S. H. Kim, S.-E. Noh, H. J. Bang, K.-M. Lee, "Robotic-assisted shoulder rehabilitation therapy effectively improved poststroke hemiplegic shoulder pain: A randomized controlled trial," *Archives of physical medicine and rehabilitation*, vol. 100, no. 6, pp. 1015–1022, 2019.

[7] I. M. Barbosa, P. R. Alves, Z. d. C. Silveira, "Upper limbs' assistive devices for stroke rehabilitation: A systematic review on design engineering solutions," *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, vol. 43, pp. 1–16, 2021.

[8] J. Lee *et al.*, "Intelligent upper-limb exoskeleton integrated with soft bioelectronics and deep learning for intention-driven augmentation," *npj Flexible Electronics*, vol. 8, no. 1, p. 11, 2024.

[9] R. Altenburger, D. Scherly, K. S. Stadler, "Design of a passive, iso-elastic upper limb exoskeleton for gravity compensation," *Robomech Journal*, vol. 3, pp. 1–7, 2016.

[10] G. Vazzoler, P. Bilancia, G. Berselli, M. Fontana, A. Frisoli, "Analysis and preliminary design of a passive upper limb exoskeleton," *IEEE Transactions on Medical Robotics and Bionics*, vol. 4, no. 3, pp. 558–569, 2022.

[11] N. G. Tsagarakis, D. G. Caldwell, "Development and control of a 'soft-actuated'exoskeleton for use in physiotherapy and training," *Autonomous Robots*, vol. 15, pp. 21–33, 2003.

[12] K. Kiguchi, T. Tanaka, T. Fukuda, "Neuro-fuzzy control of a robotic exoskeleton with emg signals," *IEEE Transactions on fuzzy systems*, vol. 12, no. 4, pp. 481–490, 2004.

[13] J. C. Perry, J. Rosen, S. Burns, "Upper-limb powered exoskeleton design," *IEEE/ASME transactions on mechatronics*, vol. 12, no. 4, pp. 408–417, 2007.

[14] P. Geethanjali, "Myoelectric control of prosthetic hands: State-of-the-art review," *Medical Devices: Evidence and Research*, pp. 247–255, 2016.

[15] N. Lotti *et al.*, "Adaptive model-based myoelectric control for a soft wearable arm exosuit: A new generation of wearable robot control," *IEEE Robotics & Automation Magazine*, vol. 27, no. 1, pp. 43–53, 2020.

[16] M. W. Spong, M. Vidyasagar, *Robot dynamics and control*. John Wiley & Sons, 2008.

[17] H.-B. Kang, J.-H. Wang, "Adaptive control of 5 dof upper-limb exoskeleton robot with improved safety," *ISA transactions*, vol. 52, no. 6, pp. 844–852, 2013.

[18] S. Balasubramanian, J. He, "Adaptive control of a wearable exoskeleton for upper-extremity neurorehabilitation," *Applied Bionics and Biomechanics*, vol. 9, no. 1, pp. 99–115, 2012.

[19] A. M. Khan, D.-w. Yun, M. A. Ali, J. Han, K. Shin, C. Han, "Adaptive impedance control for upper limb assist exoskeleton," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, 2015, pp. 4359–4366.

[20] W. He, Z. Li, Y. Dong, T. Zhao, "Design and adaptive control for an upper limb robotic exoskeleton in presence of input saturation," *IEEE transactions on neural networks and learning systems*, vol. 30, no. 1, pp. 97–108, 2018.

[21] A. Toro-Ossaba, J. C. Tejada, S. Rúa, J. D. Núñez, A. Peña, "Myoelectric model reference adaptive control with adaptive kalman filter for a soft elbow exoskeleton," *Control Engineering Practice*, vol. 142, p. 105 774, 2024.

[22] S. Głowiński, T. Krzyżyński, S. Pecolt, I. Maciejewski, "Design of motion trajectory of an arm exoskeleton," *Archive of Applied Mechanics*, vol. 85, no. 1, pp. 75–87, 2015.

[23] H. S. Lo, S. Q. Xie, "Exoskeleton robots for upper-limb rehabilitation: State of the art and future prospects," *Medical engineering & physics*, vol. 34, no. 3, pp. 261–268, 2012.

[24] D. A. Neumann, *Kinesiology of the Musculoskeletal System-E-Book: Kinesiology of the Musculoskeletal System-E-Book*. Elsevier Health Sciences, 2016.

[25] M. C. Gash, P. F. Kandle, I. V. Murray, M. Varacallo, "Physiology, muscle contraction," in *StatPearls [Internet]*, StatPearls Publishing, 2023.

[26] W. T. Dempster, G. R. Gaughran, "Properties of body segments based on size and weight," *American journal of anatomy*, vol. 120, no. 1, pp. 33–54, 1967.

[27] R. Drillis, R. Contini, M. Bluestein, "Body segment parameters," *Artificial limbs*, vol. 8, no. 1, pp. 44–66, 1964.

[28] P. Grasgruber, J. Cacek, T. Kalina, M. Sebera, "The role of nutrition and genetics as key determinants of the positive height trend," *Economics & Human Biology*, vol. 15, pp. 81–100, 2014.

[29] V. Gohel, N. Mehendale, "Review on electromyography signal acquisition and processing," *Biophysical reviews*, vol. 12, no. 6, pp. 1361–1367, 2020.

[30] J. B. Reece, L. A. Urry, M. L. Cain, *Campbell biology*. Pearson, 2017.

[31] X. Chen, "Neuronal membrane and mechanisms appeared on the membrane surface," in *BIO Web of Conferences*, EDP Sciences, vol. 55, 2022, p. 01 024.

[32] J. A. Davies, *Human Physiology: A Very Short Introduction*. Oxford University Press, 2021.

[33] S. Gollapudi, J. Michael, M. Chandra, "Striated muscle dynamics," 2014.

[34] S. Shams, M. Asif, S. Hussain, "Muscle mechanics and electromyography," *Recent advances in biomechanics*, 2020.

[35] A. L. Goldberger *et al.*, "Physiobank, physiotoolkit, and physionet: Components of a new research resource for complex physiologic signals," *circulation*, vol. 101, no. 23, e215–e220, 2000.

[36] D. Bağci, "Biyonik el kontrolü için emg işaretlerininin makine öğrenmesi yöntemiyle sınıflandırılması," M.S. thesis, Fen Bilimleri Enstitüsü, 2016.

[37] S. Solnik, P. Rider, K. Steinweg, P. DeVita, T. Hortobágyi, "Teager–kaiser energy operator signal conditioning improves emg onset detection," *European journal of applied physiology*, vol. 110, pp. 489–498, 2010.

[38] X. Li, A. S. Aruin, "Muscle activity onset time detection using teager-kaiser energy operator," in *2005 IEEE Engineering in Medicine and Biology 27th Annual Conference*, IEEE, 2006, pp. 7549–7552.

[39] J. R. Potvin, S. H. Brown, "Less is more: High pass filtering, to remove up to 99% of the surface emg signal power, improves emg-based biceps brachii muscle force estimates," *Journal of Electromyography and Kinesiology*, vol. 14, no. 3, pp. 389–399, 2004.

[40] A. Phinyomark, H. Hu, P. Phukpattaranont, C. Limsakul, "Application of linear discriminant analysis in dimensionality reduction for hand motion classification," *Measurement Science Review*, vol. 12, no. 3, pp. 82–89, 2012.

[41] A. Bertomeu-Motos *et al.*, "Estimation of human arm joints using two wireless sensors in robotic rehabilitation tasks," *Sensors*, vol. 15, no. 12, pp. 30 571–30 583, 2015.

[42] Y. LeCun, Y. Bengio, G. Hinton, "Deep learning," *nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[43] S. Haykin, *Neural networks: a comprehensive foundation*. Prentice Hall PTR, 1998.

[44] C. M. Bishop, *Neural networks for pattern recognition*. Oxford university press, 1995.

[45] I. Goodfellow, Y. Bengio, A. Courville, *Deep Learning*. MIT Press, 2016, `http://www.deeplearningbook.org`.

[46] I. Sutskever, J. Martens, G. Dahl, G. Hinton, “On the importance of initialization and momentum in deep learning,” in *International conference on machine learning*, PMLR, 2013, pp. 1139–1147.

[47] J. Duchi, E. Hazan, Y. Singer, “Adaptive subgradient methods for online learning and stochastic optimization.,” *Journal of machine learning research*, vol. 12, no. 7, 2011.

[48] A. Tharwat, “Classification assessment methods,” *Applied computing and informatics*, vol. 17, no. 1, pp. 168–192, 2021.

[49] D. M. Powers, “Evaluation: From precision, recall and f-measure to roc, informedness, markedness and correlation,” *arXiv preprint arXiv:2010.16061*, 2020.

[50] A. Géron, *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow*. " O’Reilly Media, Inc.", 2022.

[51] D. Staudenmann, K. Roeleveld, D. F. Stegeman, J. H. Van Dieën, “Methodological aspects of semg recordings for force estimation–a tutorial and review,” *Journal of electromyography and kinesiology*, vol. 20, no. 3, pp. 375–387, 2010.

[52] A. V. Hill, “The mechanics of active muscle,” *Proceedings of the Royal Society of London. Series B-Biological Sciences*, vol. 141, no. 902, pp. 104–117, 1953.

[53] A. V. Hill, “The dynamic constants of human muscle,” *Proceedings of the Royal Society of London. Series B-Biological Sciences*, vol. 128, no. 852, pp. 263–274, 1940.

[54] S. E. Rodrigo, J. A. Ambrósio, M. P. Tavares da Silva, O. H. Penisi, “Analysis of human gait based on multibody formulations and optimization tools,” *Mechanics Based Design of Structures and Machines*, vol. 36, no. 4, pp. 446–477, 2008.

[55] Y.-c. Fung, *Biomechanics: Mechanical Properties of Living Tissues*. Springer Science & Business Media, 2013.

[56] D. G. Lloyd, T. F. Besier, “An emg-driven musculoskeletal model to estimate muscle forces and knee joint moments in vivo,” *Journal of biomechanics*, vol. 36, no. 6, pp. 765–776, 2003.

[57] P. Konrad, “The abc of emg,” *A practical introduction to kinesiological electromyography*, vol. 1, no. 2005, pp. 30–5, 2005.

[58] R. M. Murray, Z. Li, S. S. Sastry, *A mathematical introduction to robotic manipulation*. CRC press, 2017.

[59] F. L. Lewis, D. M. Dawson, C. T. Abdallah, *Robot manipulator control: theory and practice*. CRC Press, 2003.

[60] R. P. Borase, D. Maghade, S. Sondkar, S. Pawar, “A review of pid control, tuning methods and applications,” *International Journal of Dynamics and Control*, vol. 9, pp. 818–827, 2021.

[61] K. Ogata, Y. Yang, *Modern control engineering*. Prentice hall India, 2002, vol. 5.

[62] N. T. Nguyen, N. T. Nguyen, *Model-reference adaptive control*. Springer, 2018.

[63] R. R. Costa, L. Hsu, A. K. Imai, P. Kokotović, "Lyapunov-based adaptive control of mimo systems," *Automatica*, vol. 39, no. 7, pp. 1251–1257, 2003.

[64] M. Naeijian, A. Khosravi, "Stability analysis of model reference adaptive control with the new theorem of stability," in *2020 28th Iranian Conference on Electrical Engineering (ICEE)*, IEEE, 2020, pp. 1–6.

[65] A.-C. Huang, M.-C. Chien, *Adaptive control of robot manipulators: a unified regressor-free approach*. World scientific, 2010.

[66] A.-C. Huang, S.-C. Wu, W.-F. Ting, "A fat-based adaptive controller for robot manipulators without regressor matrix: Theory and experiments," *Robotica*, vol. 24, no. 2, pp. 205–210, 2006.

[67] M. Xiloyannis, D. Chiaradia, A. Frisoli, L. Masia, "Physiological and kinematic effects of a soft exosuit on arm movements," *Journal of neuro-engineering and rehabilitation*, vol. 16, pp. 1–15, 2019.

[68] STMicroelectronics. "Stm32f4discovery." Accessed: 2024-06-17. (), [Online]. Available: `https://www.st.com/en/evaluation-tools/stm32f4discovery.html`.

A EQUATIONS

A.1 Computed Torque Control

Let $g = 9.81\,\text{m/s}^2$ be the gravitational acceleration constant. q_1 and q_2 are the angular positions of the two links. m_1 and m_2 are the masses of the links. l_1 and l_2 are the lengths of the links. And finally, $\mathbf{u}$ is the input matrix (vector).

The matrix $\mathbf{B}$, which is a function of q_2, m_1, m_2, l_1, and l_2, is given by the following.

$$M^* = \begin{bmatrix} (m_1+m_2)l_1^2 + m_2 l_2^2 + 2m_2 l_2 l_1 \cos(q_2) & m_2 l_2^2 + m_2 l_1 l_2 \cos(q_2) \\ m_2 l_2^2 + m_2 l_1 l_2 \cos(q_2) & m_2 l_2^2] \end{bmatrix} \tag{A.1}$$

The mass/inertia matrix $\mathbf{M}$ is then computed as Equation (A.2).

$$M = M^* u \tag{A.2}$$

Here, $\mathbf{u}$ is the input matrix (vector).

The Coriolis and centrifugal forces matrix can be expressed as the Equation (A.3).

$$C = \begin{bmatrix} -m_2 l_1 l_2 \sin(q_2)(2\dot{q}_1\dot{q}_2 + \dot{q}_2^2) \\ -m_2 l_1 l_2 \sin(q_2)\dot{q}_1^2 \end{bmatrix} \tag{A.3}$$

The gravitational forces matrix $\mathbf{G}$ is given by the following equation.

$$G = \begin{bmatrix} (m_1+m_2)g l_1 \cos(q_1) + m_2 g l_2 \cos(q_1+q_2) \\ m_2 g l_2 \cos(q_1+q_2) \end{bmatrix} \tag{A.4}$$

A.2 Model Reference Adaptive Control

The parameter matrix p can be expressed as in Equation (A.5). [65]:

$$\begin{bmatrix} m_1 l_{c1}^2 \\ m_2 l_1^2 \\ m_2 l_{c2}^2 \\ m_2 I_1 l_{c2} \\ I_1 \\ I_2 \\ m_1 l_{c1} g \\ m_2 l_1 g \\ m_2 l_{c2} g \end{bmatrix} \tag{A.5}$$

The regressor matrix Y [65]:

$$Y(q, \dot{q}, \ddot{q} = \begin{bmatrix} \ddot{q_1} & \ddot{q_2} & \ddot{q_1} + \ddot{q_2} & y_{14} & \ddot{q_1} & \ddot{q_1} + \ddot{q_2} & cos(q_1) & cos(q_1) & cos(q_1 + q_2) \\ 0 & 0 & \ddot{q_1} + \ddot{q_2} & y_{24} & \ddot{q_2} & \ddot{q_2} & 0 & 0 & cos(q_1 + q_2) \end{bmatrix} \tag{A.6}$$

Here, the following expressions are valid: $y_{14} = 2cos(q_2)\ddot{q_1} + cos(q_2)\ddot{q_2} - 2sin(q_2)\dot{q_1}\dot{q_2} - sin(q_2)\dot{q_2}^2$ and $y_{24} = cos(q_2)\ddot{q_1} + sin(q_2)$.

The 11-term Fourier series basis functions are used as the following equation set.

$$\phi_1(x) = 1 \tag{A.7}$$

$$\phi_2(x) = \cos(x) \tag{A.8}$$

$$\phi_3(x) = \sin(x) \tag{A.9}$$

$$\phi_4(x) = \cos(2x) \tag{A.10}$$

$$\phi_5(x) = \sin(2x) \tag{A.11}$$

$$\phi_6(x) = \cos(3x) \tag{A.12}$$

$$\phi_7(x) = \sin(3x) \tag{A.13}$$

$$\phi_8(x) = \cos(4x) \tag{A.14}$$

$$\phi_9(x) = \sin(4x) \tag{A.15}$$

$$\phi_{10}(x) = \cos(5x) \tag{A.16}$$

$$\phi_{11}(x) = \sin(5x) \tag{A.17}$$

The gain matrices in the update laws can be mathematically formed as follows:

$$Q_M = I_{44} \tag{A.18}$$

$$Q_C = I_{44} \tag{A.19}$$

$$Q_G = 100 I_{22} \tag{A.20}$$

The control gain matrix K_d is determined as in Equation (A.21).

$$K_d = \begin{bmatrix} 100 & 0 \\ 0 & 100 \end{bmatrix} \tag{A.21}$$

The adaptive gain matrix Λ is taken as the following equation.

$$\Lambda = \begin{bmatrix} 20 & 0 \\ 0 & 20 \end{bmatrix} \tag{A.22}$$

The initial weighting vectors are determined as the following equation set.

$$\hat{W}_{M,11}(0) = \begin{bmatrix} 0.05 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \tag{A.23}$$

$$\hat{W}_{M,12}(0) = \hat{W}_{M,21}(0) = \begin{bmatrix} -0.05 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \tag{A.24}$$

$$\hat{W}_{M,22}(0) = \begin{bmatrix} 0.1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \tag{A.25}$$

$$\hat{W}_{C,11}(0) = \begin{bmatrix} 0.05 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \tag{A.26}$$

$$\hat{W}_{C,12}(0) = \hat{W}_{C,21}(0) = \begin{bmatrix} -0.05 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \tag{A.27}$$

$$\hat{W}_{C,22}(0) = \begin{bmatrix} 0.1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \tag{A.28}$$

PUBLICATIONS FROM THE THESIS

Conference Papers

1. Vural, B., & Yaşar, C. F. (2023, September), Ekstremum Arama Metodu Kullanılarak Düzeltilen EMG İşareti ile Kontrol Edilen Dış İskelet Robot Tasarımı Design of an Upper Limb Exoskeleton Controlled by Extremum Seeking Control Based EMG Signal Alteration. 24. Otomatik Kontrol Ulusal Konferansı (TOK 2023).