Design, Dynamics, and Control of Active-passive Upper-limb Exoskeleton Robots

by

Ali Nasr

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Doctor of Philosophy
in
Systems Design Engineering

Waterloo, Ontario, Canada, 2022

© Ali Nasr 2022

Examining Committee Membership

The following served on the Examining Committee for this thesis. The decision of the Examining Committee is by majority vote.

External Examiner: Dr. Maxime Raison
Professor, Dept. of Mechanical Engineering,
Polytechnique Montreal

Internal-External Member: Dr. Clark Dickerson
Professor, Dept. of Kinesiology and Health Sciences,
University of Waterloo

Internal Member: Dr. Ning Jiang
Associate Professor, Dept. of Systems Design Engineering,
University of Waterloo

Dr. Arash Arami
Assistant Professor, Dept. of Mechanical and Mechatronics Engineering,
University of Waterloo

Supervisor: Dr. John McPhee
Professor, Dept. of Systems Design Engineering,
University of Waterloo

Author's Declaration

This thesis consists of material all of which I authored or co-authored: see Statement of Contributions included in the thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Statement of Contributions

I declare that all the human-robot models and the control structures presented in this thesis were developed by me. The experimental trials were also designed and conducted by me.

This thesis contains some material that was taken from multi-author papers including:

[17] Nasr, A., Hunter J., Dickerson, C. R., and McPhee, J. Evaluation of a machine learning-driven active-passive upper limb exoskeleton robot: Experimental human-in-the-loop study. (Under review).
Contribution of J. Hunter: assisted with polishing the paper.

[16] Nasr, A., Inkol, K. A., and McPhee, J. Comparison of computed-torque assist-as-needed and model predictive control of robotic upper-limb exoskeletons: An experimental and computational study of human-robot interaction. (Under review).
Contribution of K. A. Inkol: assisted with thechnical review of the paper and fast model predictive control implementation.

[15] Nasr, A., Hashemi, A., and McPhee, J. Scalable musculoskeletal model for dynamic simulations of upper body movement. (Under review).
Contribution of A. Hashemi: assisted with polishing the paper and technical help with the experiments.

[14] Nasr, A., Bell, S., and McPhee, J. (2022). Optimal design of active-passive shoulder exoskeletons: A computational modeling of human-robot interaction. Multibody System Dynamics. (under press).
Contribution of S. Bell: assisted with polishing the paper.

[13] Febrer-Nafria, M., Nasr, A., Ezati, M., Brown, P., Font-Llagunes, J. M., and McPhee, J. (2022). Predictive multibody dynamic simulation of human neuromusculoskeletal systems: A review, Multibody System Dynamics. (under press).
Contribution of A. Nasr: assisted with the review of multibody biomechanical models.

[12] Nasr, A., and McPhee, J. (2022). Multibody constrained dynamic modelling of human-exoskeleton: Toward optimal design and control of active-passive wearable robots. In Proceedings of the 6th Joint International Conference on Multibody System Dynamics and 10th Asian Conference on Multibody System Dynamics. Springer.

[11] Nasr, A., and McPhee, J. (2022). Biarticular MuscleNET: A machine learning model of biarticular muscles. In Proceedings of the North American Congress of Biomechanics (NACOB).

[10] Nasr, A., Hashemi, A., and McPhee, J. (2022). Model-based mid-level regulation for assist-as-needed hierarchical control of wearable robots: A preliminary study of human-robot adaptation. Robotics 11 (1) 20. DOI: 10.3390/robotics11010020
Contribution of A. Hashemi: technical help with predictive control implementation.

[9] Nasr, A., Ferguson, S., and McPhee, J. (2022). Model-based design and optimization of passive shoulder exoskeletons. ASME. Journal of Computational and Nonlinear Dynamics. DOI: 10.1115/1.4053405
Contribution of S. Ferguson: assisted with polishing the paper.

[8] Nasr, A., Inkol, K. A., Bell, S., and McPhee, J. (2021). InverseMuscleNET: Alternative machine learning solution to static optimization and inverse muscle model. Frontiers in Computational Neuroscience. DOI: 10.3389/fncom.2021.759489
Contribution of S. Bell: assisted with polishing the paper.
Contribution of K. A. Inkol: assisted with thechnical review of the paper.

[7] Nasr, A., Bell, S., He, J., Whittaker, R. L., Jiang, N., Dickerson, C. R., and McPhee, J. (2021). MuscleNET: Mapping electromyography to kinematic and dynamic biomechanical variables. Journal of Neural Engineering. vol. 18, no. 4, p.0460d3. DOI: 10.1101/2021.07.07.451532.
Contribution of S. Bell: assisted with review and polishing the paper.
Contribution of J. He: technical help with filtering sEMG signals.
Contribution of R. L. Whittaker: technical help with data acquisition.

[6] Nasr, A., Ferguson, S., and McPhee, J. (2021). Model-based design and optimization of passive shoulder exoskeletons. In Proceedings of the ASME 2021 Virtual International Design Engineering Technical Conferences and Computers and Information in Engineering Conference (IDETC-CIE), vol. 85468, p. V009T09A029. DOI: 10.1115/DETC2021-69437
Contribution of S. Ferguson: assisted with review and polishing the paper.

[5] Nasr, A., Laschowski, B., and McPhee, J. (2021). Myoelectric control of robotic leg prostheses and exoskeletons: A review. In Proceedings of the ASME 2021 Virtual International Design Engineering Technical Conferences and Computers and Information in Engineering Conference (IDETC-CIE), vol. 85444, p. V08AT08A043. DOI: 10.1115/DETC2021-69203
Contribution of B. Laschowski: assisted with the review of myoelectric control systems.

[4] Nasr, A., He, J., Jiang, N., and McPhee, J. (2021). Muscle modelling using machine learning and optimal filtering of sEMG signals. In Proceedings of the 45th American Society of Biomechanics (ASB) Annual Conference, pp. 85.
Contribution of J. He: technical help with filtering sEMG signals.

[3] Nasr, A., He, J., Jiang, N., and McPhee, J. (2020). Activation torque estimation of muscles by forward neural networks (Forward-MuscleNET) for sEMG-based control of assistive robots. In Proceedings of the 7th International Conference of Control, Dynamic Systems, and Robotics. DOI: 10.11159/cdsr20.146.
Contribution of J. He: technical help with filtering sEMG signals.

[2] Nasr, A., and McPhee, J. (2020). Control-oriented muscle torque (COMT) model for EMG-based control of assistive robots. In Proceedings of the 7th International Conference of Control, Dynamic Systems, and Robotics. DOI: 10.11159/cdsr20.144.

[1] Nasr, A., Arami A., and McPhee J. (2019). Optimal cost function for predicting upper-limb movement with external load. In Proceedings of the 16th annual Ontario Biomechanics Conference.

Abstract

Modern power assistive technologies, neuromuscular rehabilitation engineering, and haptic research have all entered into a new age thanks to wearable exoskeleton robotics. Exoskeleton robots, which range from fully passive to fully active-assisted movements, have become an important instrument for assisting industrial employees and stroke rehabilitation therapy. Further technological advancements in this realm should continue to increase the upper limb functionality needed for activities of daily living (ADLs). Although exoskeleton technology is rapidly developing, interdisciplinary research is still required to address technological issues such as designing efficient power sources, evaluating human-robot interaction, estimating user intent, and developing control systems.

This thesis presents: 1) a scalable musculoskeletal (MapleSim Biomechanics) model for the dynamic simulations of movement, 2) an advanced high-fidelity human-exoskeleton multibody biomechatronic model, 3) a novel design for an efficient active-passive power source for wearable portable robots, 4) a human intent machine learning estimator using surface electromyography (sEMG) biological signals as input, 5) an efficient assist-as-needed (AAN) hierarchical control scheme of upper-limb power assist exoskeletons, and 6) a short-term experimental human-in-the-loop (HITL) evaluation of sEMG-based machine learning-driven hierarchical control of active-passive exoskeleton.

A musculoskeletal (MSK) model is a valuable tool for assessing complex biomechanical problems, estimating joint torques during motion, optimizing motion in sports, and designing exoskeletons and prostheses. This study proposes an open-source upper body MSK model that supports biomechanical analysis of human motion. The MSK model of the upper body consists of 8 body segments (torso, head, left/right upper arm, left/right forearm, and left/right hand). The model has 20 degrees of freedom (DoFs) and 40 muscle torque generators (MTGs), which are constructed using experimental data. The model is adjustable for different anthropometric measurements and subject body characteristics: sex, age, body mass, height, dominant side, and physical activity. Joint limits are modeled using experimental dynamometer data within the proposed multi-DoF MTG model. The model equations are verified by simulating the joint range of motion (ROM) and torque; simulation results have a good agreement with previously published research.

As an alternative to static optimization for resolving the redundancy issue in inverse muscle models, we proposed a new machine learning method. A recurrent neural network (RNN) was optimally configured, trained, and tested to estimate the pattern of muscle activation signals. Five biomechanical variables (joint angle, velocity, acceleration, torque, and activation signal) were used as inputs to the RNN, while the outputs were sEMG signals. The proposed approach permits a fast and direct estimation ability instead of iterative solutions for the inverse muscle model. It raises the possibility of integrating such a model in a device for functional rehabilitation and sports evaluation devices with real-time estimation and tracking. This method provides clinicians with a means of estimating sEMG activity without an invasive electrode setup.

Designing the exoskeleton actuation is challenging and time-consuming due to closed-kinematic loops in the 3D human-exoskeleton multibody model, complicated interactions, and interdependent selection of power transmission features. This research proposes a process for dynamic syntheses of passive and active assistive shoulder exoskeletons. First, a multibody model was developed using six components: an upper-body musculoskeletal model, an optimal controller, the exoskeleton's structure/mechanism, a passive mechanism, a powered actuator, and an assistance model. The system design was optimized by choosing features of the passive mechanism and exoskeleton motor such that the human joint active torque, power, muscle metabolic energy expenditure, and actuator electricity consumption were minimized. The resultant optimized active-passive exoskeletons allow for the creation of lighter and smaller wearable robots that reduce the user's muscular activation torque for the tasks being studied.

To estimate the human intent, motion, and force/torque in the high-level controller, a machine learning model for the regression of interrupted sEMG signals to future control-oriented signals (e.g., robot's joint angle and assistive torque) of an active biomechatronic device is proposed. The regression models, collectively known as MuscleNET, take one of four forms: artificial neural network (ANN), RNN, convolutional neural network (CNN), and recurrent convolutional neural network (RCNN). Inspired by conventional biomechanical muscle models, delayed kinematic signals were used along with sEMG signals as the machine learning model's input. The results indicated that the RNN (with filtered sEMG signals) and RCNN (with raw sEMG signals) models, both with delayed kinematic data, can extract underlying motor control information (such as joint activation torque or joint angle) from sEMG signals for pick-and-place tasks. The CNNs and RCNNs were able to filter

raw sEMG signals. The RNN with a 0.1-second prediction horizon could predict the control-oriented joint angle and assistive torque with 92% and 86.5% regression accuracy, respectively, on the test dataset. This proposed approach permits a fast, predictive, and direct estimation of control-oriented signals, instead of an iterative process that optimizes assistive torque in the inverse dynamic simulation of a multibody human-robot system. Training with these interrupted input signals significantly improved the regression accuracy in the case of sEMG signal disconnection. This robust predictive control-oriented model (RobustMuscleNET) can support volitional high-level myoelectric-based control of biomechatronic devices such as exoskeletons, prostheses, and assistive/resistive robots. The RNN model with filtered sEMG and delayed kinematic signals is particularly appropriate for applications in musculoskeletal simulation and biomechatronic device control.

A closed-loop human-robot system requires the development of an effective robotic controller that considers models of both the human and the robot, as well as the human adaptation to the robot. Doing so requires optimal coordination with the central nervous system (CNS) that manages the MSK. To develop an AAN upper-limb controller and assess human-robot interactions, we compared AAN-proportional rule, AAN-computed-torque method (CTM), AAN-fuzzy logic rule, and AAN-model predictive control (MPC) frameworks. The human-robot adaptation was simulated using a nonlinear model predictive control (NMPC) as the human CNS for the three conditions of initial (the initial session of wearing the robot, without any previous experience), short-term (the entire first session, e.g., 45 minutes), and long-term experiences. The results showed that the desired strength of the robot should be increased gradually to suppress unexpected human-robot interactions (e.g., robot vibration, human spasticity). Although both controllers derived the necessary torques for following the target shoulder joint trajectories and desired assistance strength, from experimental testing with a human participant, model-predictive control was able to outperform the CTM control in the presence of a substantial control loop delay, but at a greater computation cost.

This research studied exoskeleton control challenges, developed a test framework for evaluating a new active-passive shoulder exoskeleton, and proposed an sEMG-based human-robot cooperative control method to carry out the wearer's movement intentions. The active-passive assistance was compared with fully-passive and fully-active exoskeletons using the following criteria: 1) post-test personal survey, 2) load tolerance duration, and 3) computed human active torque, power, computed metabolic energy expenditure using sEMG signals and inverse dynamic simulation. The experiments outcomes showed that active-passive exoskeletons required less muscular activation torque (50%) from the user, and reduced fatigue duration indicators by a factor of 3, compared to fully-passive ones.

Acknowledgements

First and foremost, I would like to express my sincere gratitude to my benevolent supervisor, Dr. John McPhee, who graciously helped me carry out this project. Over the years, John has helped me develop both personally and professionally.

I also want to acknowledge my committee members - Drs. Clark R. Dickerson, Arash Arami, and Ning Jiang - for creating an environment at Waterloo Engineering that supports exoskeleton research.

I would like to thank Dr. Clark R. Dickerson and Rachel L. Whittaker for providing me with their experimental data.

I sincerely appreciate current and former members of the Motion Research Group for their encouragement and support, especially Sydney Bell, Jason Hunter, Keaton Inkol, Arash Hashemi, Spencer Ferguson, Brokoslaw Laschowski, Mahdokht Ezati, and Merwa Al-Rasheed.

I also want to thank Ekso Bionics Holdings Inc. for providing the Ekso EVO passive shoulder exoskeleton.

I would like to thank the Natural Sciences and Engineering Research Council of Canada (NSERC) and the Canada Research Chairs (CRC) program for funding my research.

Table of Contents

List of Figures xiv

List of Tables xxii

1 Introduction 1

1.1 Motivations and objective . . . 1
1.2 Challenges . . . 2
1.3 Thesis outline . . . 4

2 Literature Review 6

2.1 Musculoskeletal models . . . 6
2.2 Inverse muscle models . . . 7
2.3 Passive and active exoskeletons . . . 9
2.4 Neuromuscular sensing and estimation . . . 11
2.5 Assist-as-needed control of active exoskeleton . . . 13
2.6 Chapter summary . . . 16

3 MapleSim Biomechanics Model 17

3.1 Anthropometric and subject characteristics . . . 17
3.2 Skeletal dynamic model . . . 18
3.2.1 Locations of anatomical landmarks . . . 18
3.2.2 Body segment inertia parameters . . . 19
3.2.3 Degrees of freedom . . . 21
3.3 Biomechanical joint torque model . . . 22
3.3.1 Passive torque function . . . 25
3.3.2 Peak joint strength . . . 25
3.3.3 Active-torque-angle scaling function . . . 26
3.3.4 Active-torque-angular-speed scaling function . . . 27
3.3.5 Excitation-to-activation signal function . . . 28
3.3.6 Multi-DoF joint implementation . . . 28
3.4 General structure of dynamic equations . . . 29

3.4.1 Joint-torque-driven model . . . 29
3.4.2 Active-torque-driven model . . . 29
3.4.3 Activation-signal-driven model . . . 30
3.4.4 Excitation-signal-driven model . . . 30
3.5 Simulation method . . . 30
3.5.1 Inverse dynamic simulation . . . 30
3.5.2 Forward dynamic simulation . . . 32
3.6 Model validation . . . 35
3.6.1 Isometric test . . . 35
3.6.2 Isokinetic test . . . 37
3.6.3 Shoulder workspace volume . . . 37
3.7 Discussion . . . 38
3.7.1 Applications . . . 41
3.7.2 Limitations . . . 42
3.8 Concluding remarks . . . 42

4 Inverse Muscle Modelling (Inverse MuscleNET) 43
4.1 Data collection and preparation . . . 43
4.1.1 Subjects . . . 43
4.1.2 Instrumentation . . . 44
4.1.3 Task protocol . . . 44
4.1.4 Kinematic data estimation . . . 45
4.1.5 Filtered sEMG signals . . . 45
4.2 Recurrent neural network model . . . 46
4.2.1 Model inputs and outputs . . . 47
4.2.2 Model training and evaluation . . . 48
4.3 Optimization and analysis method . . . 48
4.3.1 Preliminary RNN configuration optimization . . . 48
4.3.2 sEMG signal estimation . . . 49
4.3.3 Input biomechanical signal sensitivity analysis . . . 49
4.3.4 RNN configuration optimization . . . 49
4.4 Results . . . 50
4.4.1 Preliminary RNN configuration . . . 50
4.4.2 Evaluation of sEMG signal estimation . . . 50
4.4.3 Input biomechanical signal sensitivity analysis . . . 51
4.4.4 Final RNN configuration optimization . . . 51
4.5 Discussion . . . 54
4.6 Concluding remarks . . . 57

5 Human-exoskeleton System Model **58**

5.1 Human model . . . 58

5.1.1 Upper-body dynamic model . . . 58

5.1.2 Optimal controller model of the CNS . . . 60

5.2 Exoskeleton model . . . 60

5.2.1 Dimensions and inertia properties . . . 61

5.2.2 Joints configuration . . . 61

5.2.3 Passive mechanism model . . . 63

5.2.4 Powered actuator model . . . 64

5.2.5 Hierarchical assistance model . . . 66

5.3 Closed-kinematic system . . . 68

5.3.1 Exoskeleton-human interaction . . . 68

5.3.2 Multibody model . . . 68

5.3.3 Multibody analysis . . . 69

5.3.4 Simulation requirement . . . 69

5.4 Concluding remarks . . . 70

6 Active-passive Exoskeleton Design **71**

6.1 Data acquisition and processing . . . 71

6.1.1 Tasks . . . 71

6.1.2 Data recording . . . 72

6.1.3 Kinematic process . . . 73

6.2 Design optimization . . . 73

6.2.1 Overall method . . . 73

6.2.2 Design variables . . . 75

6.2.3 Design cost function . . . 75

6.2.4 Design conditions . . . 76

6.3 Results and discussion . . . 76

6.3.1 Human performance analysis . . . 76

6.3.2 Motor selection . . . 78

6.3.3 Exoskeleton actuation comparison . . . 78

6.3.4 Custom passive mechanism . . . 79

6.3.5 Simulation example . . . 80

6.4 Concluding remarks . . . 80

7 Machine Learning Interpretation of Myoelectric Signals (MuscleNET) 81
7.1 Machine learning models . . . 81
7.1.1 Dividing ratio and data . . . 82
7.1.2 Artificial neural network model . . . 83
7.1.3 Recurrent neural network model . . . 83
7.1.4 Convolutional neural network model . . . 85
7.1.5 Recurrent convolutional neural network model . . . 88
7.2 Output signals . . . 89
7.2.1 Biomechanical variables . . . 90
7.2.2 Biarticular output . . . 90
7.2.3 Control-oriented output . . . 90
7.2.4 Future control-oriented output . . . 90
7.3 Input signals . . . 91
7.3.1 Delayed kinematics . . . 91
7.3.2 Filtered sEMG signals . . . 91
7.3.3 Optimizing sEMG filtering steps . . . 92
7.3.4 Interrupted sEMG signals . . . 92
7.4 Results and discussion . . . 93
7.4.1 Machine learning model . . . 93
7.4.2 Output signals . . . 100
7.4.3 Input signals . . . 102
7.5 Application . . . 105
7.5.1 Neuromusculoskeletal model simulation . . . 106
7.5.2 Myoelectric-based control of biomechatronic devices . . . 106
7.6 Concluding remarks . . . 107

8 Assist-as-needed Mid-level Control of Active Exoskeleton 108
8.1 Human-exoskeleton multibody model . . . 108
8.2 Central nervous system control . . . 110
8.3 Active exoskeleton control . . . 111
8.3.1 High-level control . . . 111
8.3.2 Mid-level controller . . . 111
8.3.3 Low-level controller . . . 117
8.4 Condition of simulation comparison . . . 117
8.5 Comparison of ANN proportional, CTM, and fuzzy-logic mid-level controller . . . 118
8.5.1 Full strength (first scenario) . . . 118
8.5.2 Variable strength (second scenario) . . . 120
8.5.3 Comparison . . . 121

8.5.4 Outcome and future work . . . 122
8.6 Comparison of ANN CTM and MPC mid-level controller . . . 123
8.6.1 Computational evaluation . . . 123
8.6.2 Experimental evaluation . . . 126
8.6.3 Outcomes . . . 127
8.7 Concluding remarks . . . 129

9 Experimental Evaluation of Active-passive Exoskeleton 131
9.1 Equipment . . . 131
9.1.1 Shoulder exoskeleton . . . 131
9.1.2 Motor and encoder . . . 131
9.1.3 Surface electromyography (sEMG) and inertial measurement unit (IMU) . . . 133
9.1.4 Calibration . . . 134
9.1.5 Data preparation . . . 135
9.2 Hierarchical controllers . . . 135
9.2.1 High-level controller . . . 136
9.2.2 Mid-level controller . . . 137
9.2.3 Low-level controller . . . 138
9.3 Participants and test protocol . . . 138
9.4 Evaluation criteria . . . 139
9.5 Results and discussion . . . 140
9.5.1 High-level controller accuracy . . . 140
9.5.2 Low-level controller gains . . . 141
9.5.3 Exoskeleton evaluation . . . 142
9.5.4 sEMG-based control . . . 144
9.6 Practical challenges and future directions . . . 145
9.7 Concluding remarks . . . 145

10 Conclusions 147
10.1 Summary of contributions . . . 147
10.2 Recommendations for future research on control of active-passive exoskeletons . . . 149
10.3 Exoskeleton requirements and challenges . . . 150
10.4 Future research directions . . . 152

References 153

Glossary 171

List of Figures

1.1 The diagram of human-robot-environment interaction (HREI) structure. The force/torque and the signals are in blue and black, respectively. The dashed line may connect or not, based on the situation of object handling, installed sensors, type of sensors, and place of sensor installation. . 3

2.1 Inverse simulation of a musculoskeletal system with multiple muscles, using (a) conventional two optimization loops for estimation of muscle forces and activation signals, (b) machine learning model for muscle force estimation and one optimization loop for activation signals, and (c) machine learning model for activation signals estimation. 8

2.2 The essential components of the human-robot system. The human consists of the CNS (modeled using NMPC [166]), neural control (previously modeled by MuscleNET [38, 170]), muscle (modeled using the MTGs of human model), and skeletal system. The robot consists of mechanical and electrical hardware. The essential electrical components are wearable wireless sEMG sensors and the receiver (Delsys Inc, Boston, MA), controller, adaptor, brushless direct current (BLDC) motor, emergency stop, and power supply. The EVO (Ekso Bionics Holdings Inc, California, USA) upper limb exoskeleton is used as a wearable device with three levels of passive assistance. 11

2.3 Schematic of HREI system with hierarchical strategy control. $\hat{\tau}_h$ is estimated human joint torque and $\hat{\theta}_{t+1}$ is predicted/desired joint angle computed by the high-level controller. 14

3.1 Locations of selected anatomical landmarks and the consequent orientations of the segment coordinate system according to international society of biomechanics (ISB) standard [68, 278]. . . . 18

3.2 Depiction of MSK model DoF and directions for (a) torso flexion/extension (TFE), (b) torso right/left lateral flexion (TRLLF), (c) torso right/left axial rotation (TRLAR), (d) neck flexion/extension (NFE), (e) neck right/left lateral flexion (NRLLF), (f) neck right/left axial rotation (NRLAR), (g) shoulder flexion/extension (SFE), (h) shoulder adduction/abduction (SAA), (i) shoulder medial/lateral rotation (SMLR), (j) elbow flexion/extension (EFE), (k) forearm pronation/supination (FPS), (l) wrist flexion/extension (WFE), and (m) wrist radial/ulnar deviation (WRUD). (n) Side and Front view with the initial posture (zero joint angles). 21

3.3 Schematic of the shoulder, elbow, and wrist joint, actuated by (a) Hill-type muscles and (b) MTGs. Note the complex muscle route geometry and redundancy encountered when using muscles. 23

3.4 Forward simulation of a musculoskeletal system with multiple muscles. a_i and F_i are the activation signal and the muscle force in the dynamic muscle model, respectively. 23

3.5 Schematics of forward simulation of a musculoskeletal system with MTGs. The MTG model consists of 2 active components for two excitation signals (positive and negative direction) and one passive component. 23

3.6 Example of MTG curves to mimic constraints of tendon and muscle forces: activation signal curve (a), the active-torque-angular-speed scaling curve (b), the active-torque-angle scaling curve (c), and the passive torque curve (d). 24

3.7 The schematic for extrapolating the joint torque for x-direction rotation (SAA) as a function of the z-direction orientation (SFE). The minus sign of torque in the second column of the extrapolation table is neglected for normalized functions, for example, active-torque-angle and active-torque-angular-speed scaling functions. 29

3.8 A depiction of object manipulation task performed for kinematic data collection. The participants were expected to lift and lower an object between two target locations in the sagittal plane. The joint trajectories were measured using an upper body Xsens MVN suit (Xsens Technologies, Enschede, Netherlands). 31

3.9 The trajectories of 20 joint angles (^{o}) acquired from the motion recording process and 5 Hz low pass filtering. 31

3.10 The plot of absolute active torque of 20 joints and 40 MTGs optimized for minimum active torque in the pick-and-place movement. 32

3.11 The activation signal of the 20 joints and 40 MTGs for activation of the MTGs driving the optimized active torque in the pick-and-place movement. 33

3.12 The excitation signal of 40 joint MTGs for the pick-and-place movement. 34

3.13 Elements of a typical musculoskeletal simulation in MapleSim Biomechanics. Movement occurs from a complex composition of the neural, muscular, and skeletal systems. MapleSim Biomechanics comprises computational models of these systems, allowing for human movement prediction and analysis. Neural commands to MTGs, in the form of excitations, can be estimated from optimal controller models or experimental data (e.g., sEMG) with MuscleNET (a machine learning framework that maps electromyography (EMG) excitations to kinematic/dynamic variables) [38]. MapleSim Biomechanics's MTGs models, which convert excitations into biomechanical joint torque, contain the passive torque, active torque angle scaling, and active torque angular velocity scaling features of the biomechanical joint. MapleSim Biomechanics MTG are based on experimental data. MapleSim Biomechanics's peak joint strength depends on age, height, body mass, dominant side, and athletic activity. 35

3.14 Maximum isometric torque generated by (a) torso flexors (+) and extensors (-) versus torso flexion, (b) torso right (+) and left (-) lateral flexors versus torso lateral flexion, (c) torso left (+) and right (-) axial rotator versus torso axial rotation, and (d) neck flexors (+) and extensors (-) versus neck flexion. The model estimates are highlighted in thick solid lines and are compared to experimental data from Keller and Roy [284], Kumar [67], and Jordan et al. [74]. M: Male, F: Female, Height is in meter (m), Weight is in kilograms (kg), number of session workout per week showed with W. 36

3.15 Maximum isometric torque generated by (a) shoulder flexors (+) and extensors (-) versus shoulder flexion rotation, (b) shoulder adductors (+) and abductors (-) versus shoulder abduction rotation, (c) shoulder horizontal adductors (+) and abductors (-) versus shoulder horizontal abduction rotation, and (d) shoulder medial (+) and lateral (-) rotators versus shoulder rotation. The model estimates are highlighted in thick solid lines and are compared to experimental data from our test (mentioned in section 3.3.3), Garner and Pandy [292], Winters and Kleweno [334], Otis et al. [77], Hughes et al. [276], and Engin and Kaleps [291]. SFE, shoulder vertical adduction/abduction (SVAA), and shoulder horizontal adduction/abduction (SHAA) were determined with a straight arm in the neutral position. SMLR torque was determined for the elbow in 60^o flexion, and 60^o abduction. M: Male, F: Female, Height is in meter (m), Weight is in kilograms (kg), number of session workout per week showed with W. 38

3.16 Maximum isometric torque generated by (a) elbow flexors (+) and extensors (-) versus elbow flexion angle, (b) forearm pronators (+) and supinators (-) versus pronation angle, (c) wrist flexors (+) and extensors (-) versus wrist flexion angle, and (d) wrist ulnar (+) and radial (-) deviators versus deviation angle. The model estimates are highlighted in thick solid lines and are compared to experimental data from Garner and Pandy [292], Buchanan et al. [336], Amis et al. [335], Gonzalez [337], and Delp et al. [297]. EFE, FPS, WFE, and WRUD were determined with the shoulder in 60^o flexion, and 30^o abduction and all other DoF in neutral position. M: Male, F: Female, Height is in meter (m), Weight is in kilograms (kg), number of session workout per week showed with W. 39

3.17 Maximum isokinetic torque generated by (a) shoulder flexors (+) and extensors (-) versus shoulder flexion angular velocity, (b) shoulder medial (+) and lateral (-) rotators versus shoulder angular velocity, (c) shoulder horizontal adductors (+) and abductors (-) versus shoulder horizontal abduction angular velocity, (d) elbow flexors (+) and extensors (-) versus elbow flexion angular velocity. The model estimates are highlighted in thick solid lines and are compared to experimental data from our test (mentioned in section 3.3.3), Brown [339], Alderink and Kuch [338], Timm [341], Silva et al. [304], Dvir [340]. M: Male, F: Female, Height is in meter (m), Weight is in kilograms (kg), number of session workout per week showed with W. 40

3.18 The visualization of a male right shoulder ROM for (a) Flexion/Extension from the side view, (b) vertical abduction/adduction from the front view, (c) horizontal abduction/adduction from the top view, and (d) 3D reachable surface of a unit (1-meter length) upper-arm. 41

4.1 A depiction of the data collection process, sEMG electrodes and the retroreflective position marker placements by Whittaker et al. [349], where participants delivered and returned a weighted object between target locations in the sagittal plane. 44

4.2 Sample of a raw in (a) and filtered in (b) sEMG signal. The frequency domain of relevant sEMG signal is shown in (b, d). 46

4.3 Schematic of a two-layer nonlinear autoregressive with external input neural (NARX) network as an RNN learning model for sEMG signal prediction. n_u and n_y are the input signal former values and the output signal former values, respectively. **IW**, **LW**, and **b** are the input weight, layer weight, and bias of one neuron, respectively. 47

4.4 Schematic of the sequential backward elimination algorithm for evaluating the sensitivity analysis and dominant biomechanical signals for estimating the sEMG channels. 50

4.5 The performance of the preliminary configuration of InverseMuscleNET on test data to estimate sEMG signals for 11 upper limb muscles. 52

4.6 The mapping performance of 11 sEMG signals for the general-model. All regression (training and validation) accuracies are shown on the top, and the K-fold cross-validation (CV) regression accuracies (average, maximum, and minimum) are shown on the bottom. 53

4.7 The RNN training (blue), validation (green), and test (red) performance (mean squared normalized error (MSE)) for 134 iterations. The exported model is based on epoch 34. The training stops after 100 iterations after the best validation performance. 54

4.8 Error histogram indicating the difference between actual sEMG signal and estimated output. . . 54

4.9 Regression or scatter plot of training, validation, test, and all data to RNN model. 55

4.10 The performance of final optimized InverseMuscleNET configuration on test data for the estimation of sEMG signals from 6 upper limb muscles. 56

5.1 (a) Two views of 20-DoF MapleSim Biomechanics model of a human upper body and EVO upper limb exoskeleton. The essential human joints are the lumbar joint center (LJC) with a 3-DoF spherical joint (TFE, TRLLF, and TRLAR), shoulder joint center (SJC) with a 3-DoF spherical joint (SFE, SAA, and SMLR), EFE, FPS, a 2-DoF wrist joint (WFE, and WRUD). (b) Schematic diagram of the human and exoskeleton closed kinematic loop models and their interaction with hinge (HG), elevation (E), and virtual spring (VS) joints. The human and exoskeleton systems are connected at the armrest (AR) and LJC. 59

5.2 An image of the actual active-passive shoulder exoskeleton. The important mechanical parts (e.g., belt, shafts, slices, link, and mechanism) and the electrical components (e.g., adaptor, motor, and sEMG sensor) are shown in the image. 61

5.3 (a) Plot of the first 5 revolute joints' passive torques with rotational spring (K) and maximum ROM (φ_{max}). (b) The normalized passive joint torque to the maximum of fitted data, (c) the measured passive joint torque for spring system levels 1 to 3, and (d) a polar indication of joint torque. All the measurements were done with zero rotational shift adjustment for (b-d). (e) The relation of the motor torque (T_m) and armature current (i_a) with motor torque constant (k_m). (f) The smooth curve of the human elevation torque using the dead zone torque (T_δ) and the threshold torque (T_γ). 63

5.4 Views of the passive elevation joint mechanisms assembled using a compressed spring (a) or stretched spring (b). (a) Evo providing an assistive torque to the upper arm of a wearer via a compressed spring [363]. (b) Our designed custom mechanism (adopted from Nasr et al. [26]) delivering passive torque via a stretched spring that wraps around a point P_B. 65

5.5 Block diagram of the mid-level controller of the hierarchical AAN-CTM control architecture of active-passive exoskeleton robots. The mid-level controller defines the potential driving torque using the human and exoskeleton model [28]. 67

6.1 Illustration of six industrial and daily living activity tasks for designing, optimizing, and evaluating the shoulder exoskeleton. The first three motions are arm-reaching tasks, and the last three motions are object-handling tasks. 72

6.2 Schematic of designing, synthesizing, analyzing, and optimizing the inverse scenario. 73

6.3 Schematic of designing, synthesizing, analyzing, and optimizing the forward scenario. 74

6.4 The average of absolute active torque for the right shoulder, left shoulder, and the torso (a-c). The reaction torque and the reaction force at the right armrest, left armrest, and the belt (d-i). The metabolic energy expenditure of the human wearing the exoskeleton is normalized by the human not wearing the exoskeleton for 6 tasks (j). H: human, IE: inactive exoskeleton, FP: fully-passive exoskeleton, FA: fully-active exoskeleton, AP: active-passive exoskeleton. 77

6.5 (a) The custom passive mechanism torque-angle profile for the stretched arm motion designed by the first loop of the inverse scenario. (b) The depiction of the custom mechanism posture changes (without the attachment link). The 1-6 postures in (b) are marked with relevant generated passive torque in (a). 78

6.6 The joint angle changes of the right SFE, SAA, and the exoskeleton's elevation (a-c). The activation torque of the right shoulder for the human-only condition (dashed) and the active-passive exoskeleton condition (solid) (d and e). The exoskeleton assistive torque of the passive mechanism and motor #4 (f). The excitation signal of the mentioned joint (g and h). The motor consumption current versus time is shown at (i). 79

7.1 Schematic of machine learning models configuration optimization loop. 81

7.2 Schematic of training models with training, validation, and testing sets of the input and output signal. 82

7.3 Schematic of the RNN MuscleNET for combined input of 11 filtered sEMG and three delayed kinematic signals. 84

7.4 A sample of the input image made from sEMG signals and delayed kinematic signals used as the input of the CNN or RCNN configuration of MuscleNET. The delayed shoulder elevation joint angle, velocity, and acceleration are $\theta(t')$, $\dot{\theta}(t'')$, and $\ddot{\theta}(t''')$, respectively (t', t'', and t''' are the delayed times). 86

7.5 Schematic of the convolutional MuscleNET for raw sEMG signal input only, including convolutional layers, average pooling layers, and fully connected (FC) layer. The batch normalization and the rectified linear unit after the convolutional layers in each group have not been shown for simplicity. The convolutional layers and the average pooling layers have the same padding. . . . 87

7.6 Schematic of the convolutional MuscleNET for combined input of raw sEMG and delayed kinematic signals, including convolutional layers, average pooling layers, and FC layer. The batch normalization and the rectified linear unit after the convolutional layers in groups 2 and 3 have not been shown for simplicity. 88

7.7 Schematic of the RCNN MuscleNET for combined input of raw sEMG and delayed kinematic signals, including convolutional layers, average pooling layers, and a long short-term memory (LSTM) layer. The batch normalization and the rectified linear unit after the convolutional layers in groups 2 and 3 have not been shown for simplicity. 89

7.8 Schematic of interrupting sEMG input signal algorithm. 93

7.9 Visual representation of dividing data base into training, validation, and testing sets for subject-specific and general models. 94

7.10 The machine learning training in (a), validation in (b), and testing in (c) regression of a subject-specific model of RNN with filtered sEMG and delayed kinematic signals input and joint angle output. The test subject did not participate in training and validation. 95

7.11 The machine learning training in (a), validation in (b), and testing in (c) regression of a general model of RNN with filtered sEMG and delayed kinematic signals input and joint angle output. The test subject did not participate in training and validation. 96

7.12 The RCNN training (solid) and validation (dashed) rates for 1000 iterations. However, the training procedure could be stopped at 500 iterations. 98

7.13 The performance of RNN and RCNN MuscleNET on random test data for users' joint angle in (a) and joint torque trajectory in (b). 98

7.14 The test of RNN and RCNN MuscleNET for second random non-used user data for training users' joint angle in (a) and joint torque trajectory in (b). 99

7.15 A section of four subjects' time-series data estimation for normalized future joint angle in (a-d) and assistive torque in (e-h) using a prediction horizon of 0.1 seconds. These subjects had similar anthropometric data (body height and mass) as the average subjects. 99

7.16 The regression accuracy of training, validation, and test data set for angle prediction in (a), acceleration prediction in (b), and assistive torque prediction in (c). 101

7.17 Evaluation of normal model and robust model with interrupted sEMG signals. The target output is the normalized future robot's joint angle in (a) and assistive torque in (b). 103

7.18 Comparison of robust and normal model with the number of interrupted sEMG signals input evaluated for all test subjects' data in (a). Signal impact versus maximum regression with sequential backward elimination signal analysis for 11 signals in (b). 104

7.19 Evaluation of normal model and robust model with interrupted sEMG signals for a random subject. The normalized angle prediction is in (a), and normalized assistive torque prediction is in (b). 104

7.20 Schematic of a neuromusculoskeletal (NMSK) model simulation (for example, for sport engineering purposes or MSK analysis). MuscleNET acts as a neural control and MTG act as a muscle model using the filtered sEMG, joint position, velocity, and acceleration. MuscleNET's output is the joint activation or excitation signal supplied to the MSK dynamic model for forward dynamic simulation. 106

7.21 Schematic for controlling an exoskeleton, a prosthesis, or a rehabilitation robot using MuscleNET as a model of mapping the raw sEMG, joint feedback position θ_f, velocity $\dot{\theta}_f$, and acceleration $\ddot{\theta}_f$ to future assistive torque $\tau_{t+\Delta t}$ and future joint angle $\theta_{t+\Delta t}$. The mid-level and low-level controller is responsible for controlling the mechatronic device. 107

8.1 Main control system components and connectivity protocols. The high, mid, and low-level controllers are described in section 8.3. The connection protocols/methods are highlighted as blue arrows. 109

8.2 (a) Two views of 20-DoF MapleSim model of a human upper body and EVO upper limb exoskeleton. (b) Depiction of a healthy subject wearing an actual EVO passive system with augmented BLDC motor on the elevation joint. (c & d) Delsys Trigno wireless sEMG system (Delsys Inc, Natick, MA, USA) was used to measure muscle activations over the area of #1 upper trapezius (UTRA), #2 middle trapezius (MTRA), #3 middle deltoid (MDEL), #4 posterior deltoid (PDEL), #5 anterior deltoid (ADEL), and #6 brachioradialis (BRD) of the right forearm, shoulder, and upper trunk muscles. 109

8.3 Schematic representation of the active-passive robotic exoskeleton's hierarchical control structure, including high, mid, and low-level controllers. The high-level component predicts future values of the control-oriented joint angle, angular velocity, and assistive torque. The mid-level controller determines the driving torque using the human and exoskeleton models. The low-level controller then ensures that the state of the motor matches these descending commands. 112

8.4 Schematic of (a) AAN-proportional rule, (b) AAN-CTM rule, (c) AAN-fuzzy logic rule, and (d) AAN-model predictive control as mid-level controller. 113

8.5 The Mamdani type fuzzy membership function and degree of membership for the two models with three inputs (a-c) and two outputs (d & e). The surface function plots of the outputs of the models for all input variables (f & g). The defuzzification technique is centroid-based. These membership functions are designed by the trial and error method. 115

8.6 The motion capture system markers (Vicon Motion Systems Ltd, UK) and wireless sEMG sensors (Delsys Inc, MA, USA) are attached to the subject who is wearing an inactive EVO exoskeleton (Ekso Bionics Holdings Inc., CA, USA). 118

8.7 The sagittal pick and place task with active shoulder and elbow joint (a). The desired trajectory of the shoulder and elbow joint angles (b). The external vertical load of the object at the palm of the hand (c). The required active torque of the shoulder joint with and without an inactive robot using inverse dynamic simulation (d). 118

8.8 The 1st scenario (full strength) plot of the shoulder actuation torques (solid line), the desired shoulder actuation torque (dash line), the extra assistance (red area), and insufficient torque (blue area). Shown are the three mid-level optimized controllers of the AAN-proportional (a-c), AAN-CTM (d-f), AAN-fuzzy logic rule (bottom-rule) for three different conditions of initial (a, d, & g), short-term (b, e, & h), long-term experience (c, f, & i) of wearing a powered robot. . . . 119

8.9 The integral of absolute, negative, and positive actuation error for AAN-proportional, AAN-CTM, and AAN-fuzzy logic controller (a); The integral of absolute velocity error for three CNS conditions of initial, short-term, and long-term experience (b); The integral of absolute tracking error for three CNS conditions of initial, short-term, and long-term experience (c). 120

8.10 The integral of absolute, negative, and positive actuation error for AAN-proportional, AAN-CTM, and AAN-fuzzy logic controller (a, d, & g); The integral of absolute velocity error for three CNS conditions of initial, short-term, and long-term experience (b, e, & h); The integral of absolute tracking error for three CNS conditions of initial, short-term, and long-term experience (c, f, & i). 121

8.11 The desired robot torque is computed by the simulated inverse dynamic model (dashed), as well as the robot torque for AAN-proportional (red), AAN-CTM (black), and AAN-fuzzy logic (blue) mid-level controllers. The first scenario with full strength and optimized weights for the short-term experience (a-c), and the second scenario with varying strength of 30% for initial experience (d), 50% for short-term experience (e), and 100% for the long-term experience (f). 122

8.12 Desired shoulder (a) and elbow (b) joint trajectories for the sagittal pick-and-place task, as well as (c) the downward vertical load of the object applied at the palm of the hand. Recall that the exoskeleton provides assistive torques about the shoulder joint only. 123

8.13 The results of AAN-CTM (with $\Omega = 100\%$) simulations with the human-robot system, where the human (CNS-NMPC model) performs an upper-limb pick-and-place task with assistance as follows: (a-c) under regular lifting conditions; (d-f) for the task with a heavier object, and (g-i) with a faster motion. Columns, from left to right, show the shoulder elevation joint angle, shoulder human joint torque, and robot assisting joint torque, respectively. 124

8.14 The results of AAN-basic-MPC (with $\Omega = 100\%$) simulations with the human-robot system, where the human (CNS-NMPC model) performs an upper-limb pick-and-place task with assistance as follows: (a-c) under regular lifting conditions; (d-f) for the task with a heavier object, (g-i) with a faster motion, and (j-l) with double the prediction horizon. Columns, from left to right, show the shoulder elevation joint angle, shoulder human joint torque, and robot assisting joint torque, respectively. 125

8.15 The results of fast AAN-MPC (with $\Omega = 100\%$) simulations with the human-robot system, where the human (CNS-NMPC model) performs an upper-limb pick-and-place task with assistance as follows: (a-c) under regular lifting conditions; (d-f) for the task with a heavier object, and (g-i) with a faster motion. Columns, from left to right, show the shoulder elevation joint angle, shoulder human joint torque, and robot assisting joint torque, respectively. 126

8.16 The participant wearing the active exoskeleton and sensors while doing pick-and-place task in the sagittal plane. 127

8.17 CTM experimental results; (a-f) the raw and filtered sEMG signals, (g-l) the raw Euler angles measured by inertial measurement units (IMUs), (m) predicted exoskeleton elevation angle and (n) the estimated human contribution torque for the elevation motion by MuscleNET [38], (o) the robot joint angle, and (p) robot feedforward torque. 128

8.18 Fast MPC experimental results; (a-f) the raw and filtered sEMG signals, (g-l) the raw Euler angles measured by IMUs, (m) predicted exoskeleton elevation angle and (n) the estimated human contribution torque for the elevation motion by MuscleNET [38], (o) the robot joint angle, and (p) robot feedforwarded torque are shown. 129

9.1 (a) Depiction of a healthy subject wearing an actual EVO passive system with augmented BLDC motor on the elevation joint. (c & d) The depiction of the sEMG electrode and IMU placements using Delsys Trigno wireless sEMG system (Delsys Inc, Natick, MA, USA), over the area of #1 UTRA, #2 MTRA, #3 MDEL, #4 PDEL, #5 ADEL, and #6 BRD of the right forearm, shoulder, and upper trunk muscles. 132

9.2 Block diagram of main control system components and connection protocols. The high, mid, and low-level controllers are described in section 9.2. The connection protocols/methods are highlighted as blue arrows. 133

9.3 The motor torque function of code (measured and fitted) (a), the motor angle function of code (b), the motor velocity function of code (c). 135

9.4 Block diagram of hierarchical control architecture of the active-passive exoskeleton robots, including high, mid, and low-level controllers. The high-level unit estimates the future values of control-oriented joint angle, angular velocity, and assistive torque. The mid-level controller defines the potential driving torque using the human and exoskeleton model. The low-level controller ensures the commanded data matches the motor states. 136

9.5 The participant is wearing the active exoskeleton and sensors while doing a weight-lifting task in the sagittal plane. 138

9.6 The regression accuracy of MuscleNET in MSE for (a) training, (b) validation, (c) test, and (d) all data. (e) The training performance of MuscleNET versus epoch number. 140

9.7 The quantitative evaluation values for the four exoskeleton actuation types (inactive, fully-passive, fully-active, and active-passive). Note that the maximum ($\triangle$), minimum (∇), median ($-$), and Standard deviation (STD) (blue line) are shown on the average of all subjects (bar charts). (a-c) are calculated from sEMG signals; (d-g) are calculated using the inverse dynamic MapleSim Biomechanics model and measured kinematic data. (a & d) shows average fatigue; (b & e) are showing average human flexion active torque; (c & f) shows average absolute power; (g) displays the average muscle energy expenditure; (h) shows the maximum duration time of load tolerance; and (j) displays the result of the post-test personal survey on the effectiveness of the exoskeletons. 141

9.8 The muscle fatigue evaluations for the four exoskeleton actuation types (inactive, fully-passive, fully-active, and active-passive). (a) Mean sEMG signals amplitude percentage; (b & c) Mean frequency drop and power increase of initial raw sEMG signal. The STD (blue line) is also shown on the average of all subjects (bar charts). 142

9.9 The sex impact and quantitative evaluation values for the four exoskeleton actuation types (inactive, fully-passive, fully-active, and active-passive). Note that the median ($-$), the STD, and the Tukey honestly significant difference (Tukey HSD) letters are shown on the average of all subjects for same-sex type (bar charts). (a-c) are calculated from sEMG signals; (d-g) are calculated using the inverse dynamic MapleSim Biomechanics model and measured kinematic data. (a & d) shows average fatigue; (b & e) are showing average human flexion active torque; (c & f) shows average absolute power; (g) displays the average muscle energy expenditure; (h) shows the maximum duration time of load tolerance; and (j) displays the result of the post-test personal survey on the effectiveness of the exoskeletons. 144

9.10 A sample of raw, filtered, estimated, and computed data for active-passive exoskeleton control test. (a-f) the raw and filtered sEMG signals; (g-l) the raw Euler angles measured by IMUs; (m) predicted and measured exoskeleton elevation angle; (n) predicted and measured exoskeleton elevation angular speed; (o) the estimated and calculated human contribution torque for the elevation motion, (p) the passive robot torque, (q) the active robot torque, and (r) the total assisted torque. 146

List of Tables

3.1 Abbreviations and descriptions for anatomical coordinate systems used in the upper body kinematic model according to ISB standard [68, 69, 278]. . . . 19

3.2 Scaled 3D positions of anatomical landmarks and segment centers of mass (COMs) relative to segment anatomical coordinate systems for female and male (right limb) [68, 69]. . . . 20

3.3 The female and male segment scaled mass, scaled length, and scaling factors for the inertia tensor [68, 69]. i is the unit imaginary number. . . . 20

3.4 Abbreviations and definitions of DoF, the mean ROM for female and male, the mean peak angular velocity of the upper-body joints, and the polynomial coefficients of torque-angle scaling functions (normalized isometric torque functions). . . . 22

3.5 Peak joint torque for non-athletic females and males (mean of dominant and non-dominant limb) and estimated parameters of peak isokinetic torque for non-athletic subjects with the anthropometric measures: body age, height, mass, and sex. . . . 26

4.1 The optimization variables, limits, and steps of RNN configuration. . . . 49

4.2 Top performance of RNN model configurations. . . . 51

4.3 Sensitivity analysis of 5 input biomechanical signals 53

5.1 The coordinates location, mass, COM, and inertia matrix of the mechanical system's links (the right tower of the exoskeleton). . . . 61

5.2 The hinge's passive joint parameters for the right tower of the exoskeleton. . . . 62

5.3 The electrotechnical specification of BLDC motor with or without a gearbox from CubeMars Co. (Nanchang, China). . . . 66

5.4 The coordinates location, mass, COM, and inertia matrix of the actuator cover. . . . 66

6.1 The details of 5 conditions of human and exoskeleton systems optimization. . . . 76

7.1 Detail of ANN model configuration. . . . 83

7.2 Detail of RNN model configuration. . . . 84

7.3 The optimization variables of RNN configuration (90 different configurations as a full combination of all variables). . . . 85

7.4 Comparison of regression accuracy for 80 models: 16 MuscleNET configuration times 5 output signals. . . . 97

7.5 The top 5 filtering combinations with the lowest MSE. . . . 103

8.1 The cost function weights and internal model are used for three simulated conditions of (I) initial, (II) short-term, and (III) long-term experience. The 1st and 2nd scenarios are implemented with full and variable strength of the mid-level controllers, respectively (sections 8.5.1 and 8.5.2). . . . 111

8.2 AAN-fuzzy logic rule of three functions of the mid-level controller. 116

9.1 The properties of the AK80-9 KV100 BLDC motor (Cubemars, Jiangxi Xintuo Enterprise Co., China). 132

9.2 Motor driver controller area network (CAN-bus) communication instruction. 134

9.3 The transmission control protocol (TCP)/IP server 5 port names, numbers, and functions. 134

9.4 The 6 phases of exoskeleton calibration, data-gathering, and test protocol. 139

9.5 Statistical measures for different criteria such as root mean squire of error (RMSE), p-values, and R-Squared. 143

9.6 A sample of Tukey HSD letters of metabolic energy expenditure for exoskeleton condition times sex factor. Levels not connected by the same letter are significantly different. 145

Chapter 1

Introduction

1.1 Motivations and objective

One-third of all work-related injuries and disorder cases involve musculoskeletal diseases (MSDs) [1]. The upper extremities in particular constitute one out of every five MSD [1] cases. Work-related MSDs of the upper extremities, particularly the shoulders, are common amongst Canadian workers [2–6]. According to reports, the prevalence of shoulder and arm pain is around 30%, making it one of the most prevalent regional pain syndromes [1]. One reason for this prevalence is that long-term muscular contractions tend to reduce blood flow and worsen exhaustion, in which static positions and manual force applications are necessary when lifting the weight of hand tools while performing overhead activity.

Due to the human shoulder's intricacy and the demands that can be placed on it, they are vulnerable to sprains and strains, especially when there is an excessive amount of overload placed on them [5, 7, 8]. Moreover, injury risks and MSDs at the shoulder location can have a causal link with cumulative exposure to overhead postures and shoulder joint elevations greater than 60 degrees [5, 9]. Object manipulation or manual handling (heavy lifting, pushing, tugging, holding, and carrying), working overhead, maintaining awkward postures, repetitive motions, and vibration are a few of these significant occupational exposures [7, 10–12]. Multiple occupational exposures raise the likelihood of shoulder issues [12]. The hands, elbows, shoulders, and back are the areas most frequently impacted by occupational exposures. For instance, in one US state, the total medical and insurance costs of all shoulder MSD claims with more than 7 days of lost work over 5 years were projected to be $457 million [13]. Upper extremity assistance is becoming increasingly important in a variety of industries, especially in regards to reducing work-related physical stresses [5, 14].

Muscle weakness, lack of specific joint control, joint stiffness, and aberrant synergy are all the main contributors to upper limb impairment following stroke [15]. The elbow and shoulder joints' ability to move in flexible unison is also compromised. For the rehabilitation of weak or stroke-affected limbs, movement disorders, or even just to increase muscular strength, a person may need an assistive device [16]. This is because when a person has a mobility issue, their capabilities are still restricted, which thus lowers their quality of life even more [17]. Physical rehabilitation involves using orthoses and physiotherapy to enhance the functionality of the injured limb [15, 16]. Although physiotherapy sessions may be beneficial, there is still a high likelihood for the incorrect movement of body parts if these movements are conducted manually in physiotherapy [16, 17].

Some animals, like turtles and crabs, have an external structure that is often termed as an exoskeleton. Their exoskeleton often serves as a protection mechanism from its environment and other species. It can also serve as a desiccation-preventing watertight barrier, a sensory organ, and a point of attachment for muscles [18]. Scientists were inspired by this animal biological structure and so, they designed a structure with joints derived from actuators or passive mechanisms, and linkages similar to the human body [19, 20]. Upper extremity exoskeletons, as biomechatronic systems, are interfaced with the user to supplement, replace, or amplify motor function [21]. By design, exoskeletons can supplement weak, ineffective, and injured muscles, joints, and skeletal parts caused by MSDs or a neurological disorder [19, 22]. In effect, an exoskeleton system is a brand-new form of intelligent human-machine system centered on a human [18]. It blends the abilities of both humans and

machines, hence enhancing their capabilities. As a result, the human-exoskeleton system performs at a higher level than either can do alone. Human-exoskeleton systems have broad areas of application, such as power amplification in industrial settings, compensation for neuromuscular dysfunction or post-stroke rehabilitation, and assistance with ADLs for persons with disabilities.

The concept of an exoskeleton dates back to the late 19th century. However, it was not until the 1960s that a practical prototype was operated in a master-slave configuration for use in the military to improve the wearer's strength and performance [23]. Afterward, upper limb exoskeletons with a human-robot interaction (HRI) concept were developed. The various biomechanic engineering fields have paid close attention to the upper limb exoskeletons utilized for different applications, throughout the past two decades. As a result, exoskeleton systems are becoming more significant as a potential tool for physically disabled or weak persons. However, despite their high utility and rising demand, HRI, mechanism designs, and control systems are still difficult to implement in upper limb exoskeletons [19].

Extensive study and development have gone into exoskeletons that facilitate the movement of different human joints. So far, rapid advancements in mechanical and electronic engineering, automation technology, biology, and material science have all contributed to the development of exoskeleton systems. These devices can either have 1-DoF, such as to help facilitate elbow flexion/extension, or multiple-DoF, such as to help facilitate shoulder joint movement. The kinematic configuration, ROM, control system, and power transmission are some of the variables that affect the exoskeleton's complexity and physical capabilities [24]. The selection of these components give the exoskeleton characteristics that must be assessed in its design. Any incorrect configuration would have a detrimental impact on the user, such that it could prevent extended and continuous use of the device. The most important considerations in an ergonomic exoskeleton system are mechanical design and kinematic analysis [25]. Weight, safety, portability, and comfort are also a few notable characteristics [24].

Exoskeleton robots are well known to be highly intricate wearable systems, and several engineering techniques have been employed to enhance the HRI. The three main areas of study that profoundly influence HRI are: 1. For better HRI, exoskeleton kinematic structures are created by taking into account complex biological properties and activities [26]; 2. The compliance/stiffness factor must be considered while selecting the actuation mode and power transmission [27]; 3. Most of all, the HRI is impacted by the control method used [28].

1.2 Challenges

- **Portability and weight:**

  The exoskeleton's portability increases with decreasing exoskeleton weight [29]. The device must be small, simple to use, and maneuverable, in order for the user to move around without restriction and excessive fatigue. Consequently, only the necessary DoF for the required mobility should be included in the system, which would then lessen its complexity. Some actuators and the power source must be placed far away from the injured joint [30]. Combining the device components in easy-to-wear clothing is another critical step to achieving portability. A lightweight power source with more battery life is also essential for extended usage periods [24]. Researchers are trying to implement regeneration technology into the exoskeleton system to restore energy from negative power motion [31].

- **Power source selection:**

  A crucial factor that designers and manufacturers should take into account during the design and building phases is the suitable choice of actuators for the upper limb exoskeleton. Although it also relies on the intended application, the suitable choice of actuators in the design is crucial because it tends to add weight to the exoskeleton structure. Therefore, one should choose an actuator with a high power-to-weight ratio (PWR) and the capacity to generate high torque with accurate movement [32].

  Any joint in the body requires a significant amount of energy to function. To build lightweight exoskeletons with sufficient force/torque to move the joints, it is crucial to use actuators with a high PWR. Actuators typically need to be optimized in order to increase the PWR due to the constrained space in an exoskeleton and the high torques needed for its functioning [33]. If possible, the inertia in operation and the weight of the actuator and power supply should be decreased to prevent uncomfortable loads for the user. Thus, it is important to incorporate passive components into the design to reduce the number of actuators.

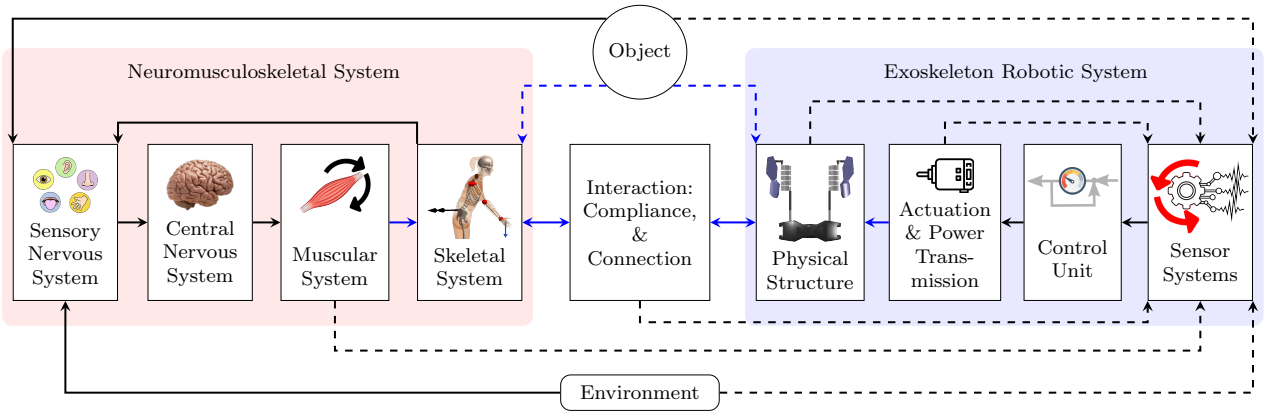


Figure 1.1: The diagram of HREI structure. The force/torque and the signals are in blue and black, respectively. The dashed line may connect or not, based on the situation of object handling, installed sensors, type of sensors, and place of sensor installation.

Based on the sources of force/torque, the exoskeleton design is divided into three categories: fully passive, fully active, and active-passive. A fully passive mechanism system usually stores potential energy from human motion [26, 34]. Compared to other categories, it is less sophisticated in terms of sensing and control. However, it is portable, safe, economical, and requires less maintenance [26, 34, 35]. Instead of supplying an assistive torque through a fixed function of joint angle (fully passive), active (powered) exoskeletons supply torque as a variable function of time using actuators such as an electric motor, a hydraulic cylinder, or a pneumatic artificial muscle [34, 36]. Since the net torque depends on the active component, the energy source and the used actuator must be powerful. As a result, active (powered) exoskeletons are big, heavy, fixed in place, expensive, and have a low PWR [37].

- **Sensing, sensor fusion, and interpretation:**

  To effectively collect the status information of the full human body and the exoskeleton system in real-time, various critical issues must be addressed. Kinematic, kinetic, and bioelectric feedback signals that measure the characteristics of human motion are the three broad categories of perceptual signals [19, 38, 39]. Every sensor has drawbacks. Numerous studies have attempted the fusion of multiple sensors as a means of intention acquisition to overcome their limitations.

  Integratable sensor modules are now readily available on the market for practically all types of electrical actuators, and can accurately predict the characteristics of the robot. However, identifying the exact human intent is crucial for controlling active exoskeletons [40]. Control failure is heavily influenced by the human operator's inability to produce real-time input to the machine while it is operating. Establishing a relationship between physiological and physical parameters and the bioelectrical signal is vital [19, 38, 39]. It is thus necessary to use sophisticated algorithms to extract helpful brain information [41]. So far, the biomimetic bioelectrical-based controller shows smooth and natural movements [42–44].

- **Design and optimization using models:**

  Designing exoskeletons using human biomechanics could result in safe and obedient HRIs [45]. In order to analyze the impact of the upper limb exoskeleton and ultimately evaluate and perhaps optimize the robot design, a thorough MSK model is needed. The MSK model enables researchers to investigate, assess, and possibly optimize the mechanical design by incorporating HRI. The human MSK system has been studied and modeled in an effort to comprehend it better. The human MSK system can be analyzed kinematically and dynamically using a variety of techniques, which provide access to pertinent biological information such as the response forces of the MSK system [46].

- **Performance assessment using models:**

  So far, it is unclear how a robotic exoskeleton would affect the human body, especially how it will affect safety, ergonomics, and health care [47]. The robotic exoskeleton system cannot only be assessed using

common performance indicators or testing procedures [19]. It is necessary to establish special testing processes that consider safety and HRI.

In addition, the wearer can adapt to the robot dynamics and the controller in the long term. The exoskeleton system's performance inherently depends on how the human may be adapting to the device. It would be useful to consider the human adaptation to the robot in the exoskeleton controller [28].

- **Human–robot interaction:**

  Humans and robots have a symbiotic relationship that goes beyond primary physical contact [48]. By implementing new mechanisms, modalities of actuation, and control strategies, several solutions have been implemented to enhance the connection between the human and exoskeletons [49]. In contrast to autonomous robots, the HITL control technique is the main focus of exoskeleton systems (Figure 1.1). However, it is difficult to fully or partially implement such a control loop [28, 50]. The challenges in developing such a control approach include cooperation and function allocation, human-exoskeleton information interchange, real-time motion planning, and safety control.

1.3 Thesis outline

Motivated by these challenges, this thesis research aims to support the active-passive actuation and automated control of next-generation robotic exoskeletons for real-world occupational assistance. Chapter 2 reviews the state-of-the-art in 1) the development of a scalable biomechanical MSK model according to anthropometric values and subject body characteristics (section 2.1); 2) the requirement(s) of developing a method to estimate EMG signals from kinematic and dynamic data for biomechanical analysis (inverse dynamic musculoskeletal simulation), post-rehabilitation analysis, sports engineering/optimization, functional electrical stimulation control, and assessment and biofeedback (section 2.2); 3) recent model development, design, and computational evaluation of active-passive actuators (section 2.3); 4) neuromuscular sensing and the machine learning intent estimation methods as high-level controllers of the active component of exoskeletons (section 2.4); 5) recent AAN automated mid-level control of power assist exoskeletons (section 2.5).

Chapter 3 presents the development details of the MSK (MapleSim Biomechanics) model, which is a valuable tool for assessing complex biomechanical problems, analyzing injuries, estimating joint torques during motion, optimizing motion in sports engineering, as well as designing exoskeletons and prostheses. This chapter proposes an open-source upper body MSK model that performs biomechanical human motion analysis. The MSK model of the upper body consists of 8 body segments (torso, head, left/right upper arm, left/right forearm, and left/right hand). This model has 20 DoFs and 40 MTGs, which are constructed using experimental data. This model is also adjustable for different anthropometric measurements and the subject body characteristics of sex, age, body mass, height, dominant side, and physical activity. Joint torques are modeled using experimental dynamometer data within the proposed multi-DoF MTG model. The model equations are verified by simulating the joint ROM and torque; simulation results have good agreement with previously published research.

Chapter 4 presents a machine learning model that is proposed as an alternative to static optimization for resolving the redundancy issue in inverse muscle models. A RNN was optimally configured, trained, and tested to estimate the pattern of muscle activation signals. Five biomechanical variables (joint angle, velocity, acceleration, torque, and activation signal) were used as inputs to the RNN. The proposed approach permits a fast and direct estimation ability instead of iterative solutions for the inverse muscle model. It also raises the possibility of integrating such a model in a real-time device for functional rehabilitation and sports evaluation devices with estimation and tracking. Hence this method allows clinicians to estimate sEMG activity without an invasive electrode setup.

Chapter 5 presents a multibody model that was developed using six components: an upper-body musculoskeletal model (chapter 4), an optimal controller, the exoskeleton's structure/mechanism, a passive mechanism, a powered actuator, and an assistance model. The inverse muscle model in chapter 4 was used for the experimental validation of the developed multibody model.

Chapter 6 proposes a process for the dynamic syntheses of passive and active assistive shoulder exoskeletons, which was modeled in chapter 5. The system design was optimized by choosing features of the passive mechanism

and exoskeleton motor, such that the human joint active torque, power, muscle metabolic energy expenditure, and actuator electricity consumption were minimized. The desired motion was experimentally measured from six tasks: frontal reaching, left to right reaching, overhead reaching, sagittal-plane object handling, frontal-plane object handling, and over-head object handling. The dynamic synthesis processes resulted in lighter and smaller wearable robots that reduced the user's muscular activation torque for the tasks being studied.

Chapter 7 presents a machine learning model for the regression of interrupted sEMG signals to future control-oriented signals (e.g., robot's joint angle and assistive torque) of an active biomechatronic device. High-level myoelectric-based hierarchical control of an active-passive exoskeleton (designed in chapter 6) is also proposed. A RNN was trained using output data, which was obtained from the offline optimization of the biomechatronic (human-robot) device and then shifted by the prediction horizon. The input of the RNN consisted of interrupted sEMG signals (to mimic signal disconnections) and previous kinematic signals of the assistive system. This robust predictive control-oriented model (RobustMuscleNET) can support volitional high-level myoelectric-based control of biomechatronic devices such as exoskeletons, prostheses, and assistive/resistive robots.

Chapter 8 presents the development of an AAN upper-limb controller and assesses human-robot interactions. We compare AAN-proportional rule, AAN-CTM, AAN-fuzzy logic rule, and AAN-MPC frameworks in this chapter. The goal of AAN is to provide the required extra torque because of the robot's dynamics and external load compared to the human limb's free movement. As a result of the AAN framework, the robot provides assistive torques that yield the movements desired by the user by measuring sEMG biomechanical signals. The human-robot adaptation is simulated using an NMPC as the human CNS for three conditions of initial (the initial session of wearing the robot, without any previous experience), short-term (the entire first session, e.g., 45 minutes), and long-term experiences.

Chapter 9 presents practical control challenges, the development of a test framework for evaluating a new active-passive shoulder exoskeleton, and the proposal of a sEMG-based human-robot cooperative control method to carry out the wearer's movement intentions. This hierarchical control used control-oriented sEMG-based intention estimation, a mid-level strength regulation, and low-level actuator control. It was applied to the exoskeleton's shoulder joint elevation movement experiments to verify the controller's effectiveness. The active-passive assistance was compared with the fully-passive and fully-active exoskeleton using the following criteria: 1) post-test personal survey, 2) load tolerance duration, and 3) computed human active torque, power, and metabolic energy expenditure using sEMG signals and inverse dynamic simulation. The experiments showed that active-passive exoskeletons required less muscular activation torque from the user, and was 3 times more helpful in terms of fatigue and operating duration.

Finally, chapter 10 reviews the contribution of this research, proposes some recommendations on improving this study path, reviews current exoskeleton development challenges, and discusses potential future research directions.

Chapter 2

Literature Review

This thesis combines the fields of human motion (i.e., the MSK upper extremity model) and mechatronics (i.e., the wearable robot) into a system known as biomechatronic systems. In this chapter, the state-of-the-art in the following areas are reviewed: 1) development of a scalable biomechanical MSK model according to anthropometric values and subject body characteristics (section 2.1); 2) development of a method to estimate EMG signals from kinematic and dynamic data for biomechanical analysis (inverse dynamic musculoskeletal simulation), post-rehabilitation analysis, sports engineering/optimization, functional electrical stimulation control, and assessment and biofeedback (section 2.2); 3) recent model development, design, and computational evaluation of active-passive actuators (section 2.3); 4) neuromuscular sensing and the machine learning intent estimation methods as high-level controllers for the active component of exoskeletons (section 2.4); 5) recent AAN automated mid-level control of power assist exoskeleton (section 2.5).

2.1 Musculoskeletal models

An MSK model is a valuable tool for evaluating complicated biomechanical problems [51], simulating and evaluating injuries [52, 53], designing and controlling exoskeletons [19, 26, 27], prostheses [31], and rehabilitation robots [54], estimating joint torques in a certain posture or during motion [52], predicting the outcome of orthopedic procedures [52, 55], optimizing motion or tools in sports engineering [56], and studying the CNS for optimal control [52, 57, 58]. The mechanical complexity of the musculoskeletal system necessitates the use of computer models that are both sufficiently accurate and reasonably efficient [53, 54, 59].

Skeletal and muscular elements are the two subsystems that make up the MSK system [43, 53]. Specifically, the bones and the connective tissues (such as cartilages and ligaments) make up the skeletal system. Skeletal muscles are similarly organized in layers over the bones, with tendons connecting them to the bones so that the forces created by the contractile components of the muscle fibers drive the body motions [51, 52, 60]. The activation of each muscle group causes one or more joints to move [58].

Most MSK models are mainly used for biomechanical simulation [53, 55] and not biomechatronics, i.e. they cannot be readily utilized for mechanical tool design problems that involve human-tool interaction (e.g., golf clubs [56], robotic mechanisms [26], exoskeletons [27], or bicycles [61]). The objective of this utilization is the biomechanical analysis and synthesis stage of designed mechanisms [52, 55], deriving a muscle-driven model for developing control frameworks for wearable robotic systems [19, 54], and integrating sensor information (e.g., sEMG, electroencephalography (EEG), IMU, camera) into control algorithms [19]. In addition, it is critical to have an MSK model for real-time data collection, simulation, and visualization in virtual reality applications with the patient in the loop [62]. One of the objectives of this paper is to develop a user-friendly adjustable model to aid biomechanical analysis, incorporate external sensory data, investigate various optimal control algorithms for the CNS, and optimize the design of sports tools, rehabilitation robots, and assistive devices.

Biomechanical MSK models are limited by measurement processes and the population on which they are built [53, 63–65]. As a result, the model should not be used or studied outside of its intended use [66]. So far,

multivariate analyses have been performed to assess the effect of subject characteristics on peak isometric and isokinetic strength. Consequently, multiple regression equations have been proposed for the strength prediction from anthropometric values [67]. These values, along with subject body characteristics, on which the biomechanical musculoskeletal model relies, consist of sex, mass, stature, age, dominant side, and physical activity; in particular, sex, body mass, and body height impact the anatomical locations, the COM, and inertia parameters [68, 69]. Furthermore, sex [70], age [70–73], body mass [70, 74, 75], body height [70, 74, 76], dominant side [71, 77–79], and physical activity [80] impact isometric and isokinetic strength of muscles and consequently joints.

Apart from these anthropometric values and subject body characteristics, the amount of torque generated at a joint is limited by the number of muscles, tendons, and joint constraints. Force-velocity scaling, force-length scaling, nonlinear passive force production owing to ligaments and connective tissue around a joint, and muscle activation dynamics are among the fundamental muscular constraints. These biomechanical constraints can be either modeled with an anatomically-detailed muscle tension [81–83] or MTGs [84]. Complicated wrapping geometry and redundancy should be taken into consideration when employing muscles [52, 53]. MTGs allow for the reduction of multibody system models by directly approximating force-length-velocity curves and passive components at the joint level (scaling torques using the joint kinematic biomechanical variable) [84]. We have adjusted the MTG models to consider anthropometric values and subject body characteristics.

So far, MTG models have only been implemented on 1-DoF joints [56, 84–86]. In 2018, McNally and McPhee [56] used a weighted approach to combine the SAA with SFE torque [56, 87]. Recently, Bell and McPhee [88] evaluated a coupling effect of shoulder motions and developed a 2-DoF shoulder model. To date, multi-DoF joints, e.g., torso, neck, or hip, have not been comprehensively modeled by MTGs.

To model and simulate human MSK systems, several commercial (e.g., AnyBody, ADAMS-LifeMOD, MuJoCo, SIMM, SimMechanics, MapleSim) and open-source (e.g., OpenSim, MSMS, RBDL, BiomechZoo) multibody dynamics simulation software have been used [52]. Except for SimMechanics and MapleSim, it is challenging to use the above software for augmenting the human model with sports tools, active prostheses, or exoskeleton models [52], especially when using hydraulic, pneumatic, or flexible components such as bike tires [61] or golf shafts [87]. MapleSim is based on symbolic computing and provides analytic derivatives when necessary, such as for optimization or sensitivity analyses, whereas the other software mentioned are based on numerical computation.

2.2 Inverse muscle models

Muscle contractions generate tension and, as a result, a moment about the human joint. Knowing the muscle forces or activation signals during human movement can assist in understanding the underlying biomechanical systems [89]. These analyses can improve movement performance, especially for athletes and patients [90]. Forward dynamic simulations of a given musculoskeletal model, driven via muscle activations, yield calculated motions [57] (Figure 3.4). As depicted in Figure 3.4, first, the raw sEMG goes through initial filtering steps and a muscle activation dynamic model [91, 92]. Second, the resultant activation converts to muscle forces using muscle contraction dynamics [91, 93, 94]. Third, the muscle forces convert to joint torque using the musculoskeletal geometry model [95]. Finally, the joint torque is used in the forward dynamic simulation of the torque-driven skeletal model to obtain the motion [96].

Conversely, the inverse dynamic problem yields either (I) muscle tensions or (II) sEMG signals from a predefined kinematic motion, introducing a redundancy resolution problem [97–101]. This redundancy is due to the number of participating muscles being greater than the number of joint degrees of freedom [102]. The redundancy resolution problem is commonly formalized as an optimization approach to find multiple muscle tension combinations for a given joint torque [97, 98, 101]. This optimization minimizes a function of muscle forces, muscle activations, and/or metabolic energy expenditure [101, 103]. Various static and dynamic optimization formulations have been proposed by [57, 97, 100, 101, 104] (Figure 2.1(a)). Generally, for model simplification of the optimization problem, four assumptions have been considered [105]: (A) there is no predefined history of the co-contraction pattern for the modeled muscles (assuming previous values have no impact), (B) the maximum muscle forces are unaltered by the muscle geometry, (C) muscles do not have dynamics and may produce any tension immediately, and (D) the musculoskeletal system satisfies a cost [97, 101, 103] (which is a function of

muscle stresses, muscle activations, or metabolic energy expenditure). However, the mentioned assumptions do not support the observed nature of muscles [82, 106–109] as muscles follow a co-contraction pattern [110], are defined by a dynamic model [111], and have a geometry that impacts the maximum muscle forces [91, 93].

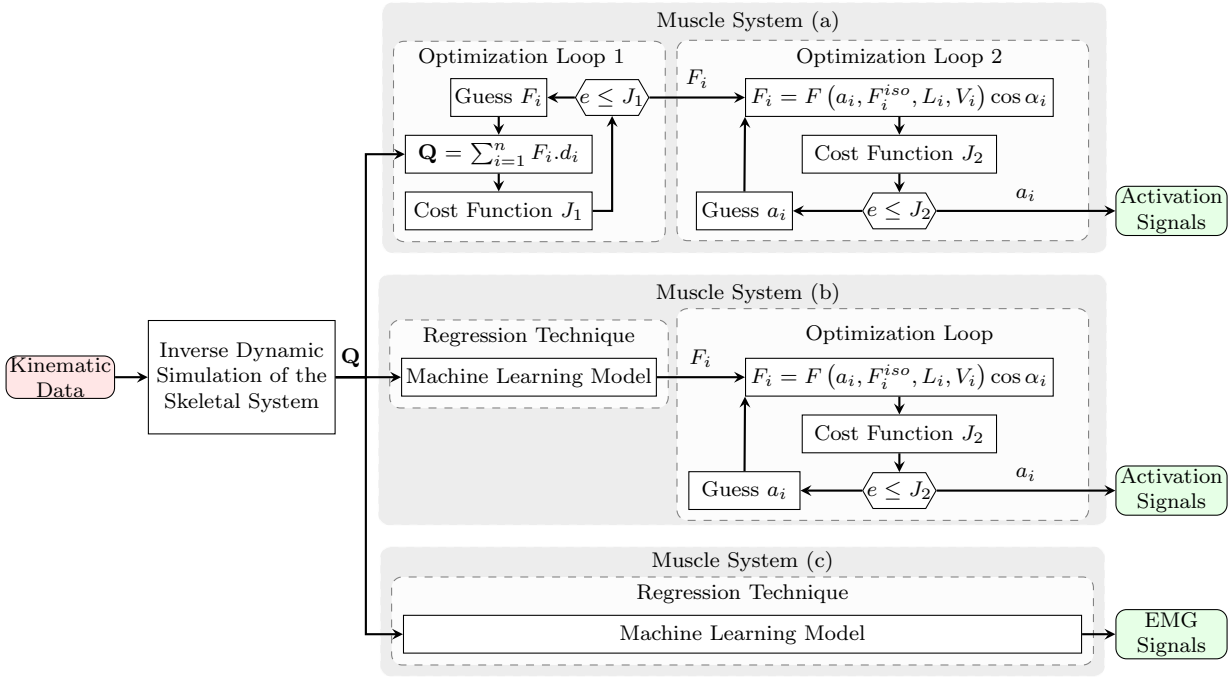


Figure 2.1: Inverse simulation of a musculoskeletal system with multiple muscles, using (a) conventional two optimization loops for estimation of muscle forces and activation signals, (b) machine learning model for muscle force estimation and one optimization loop for activation signals, and (c) machine learning model for activation signals estimation.

In conventional dynamic approaches, muscle activations or EMG signals are determined by an optimization algorithm (Figure 2.1(a)) [97, 100, 101]. Concurrently, the muscle tensions and joint moments are calculated by an EMG-driven model simulation. The optimization cost function is computed using a comparison of the joint moments with desired joint moments, calculated by the skeletal model's inverse dynamic simulation [110]. No inverse muscle model is used, and optimization tries to find the inputs of the forward dynamic muscle model to achieve specific joint moments [112]. The issues here lie in our approximation of complex skeletal muscle mechanics, which often requires modeling assumptions and sophisticated geometry/wrapping models [95, 105, 113]. Sensitivities and difficulties in fitting and identifying the relevant muscle parameters [114, 115], in addition to uncertainties related to experimental data, add further challenges to this process [116]. Due to the mentioned assumptions and challenges, the simulation results in estimated muscle activations that may not be correct [117].

Machine learning is an instrument or model for solving complicated mathematical problems without knowing the analytical relationship between the inputs and outputs [118]. Supervised machine learning offers an alternative solution to the optimization, leading to the easing of muscle redundancy, complexity, and pattern estimation of muscle tensions (or EMG signals) within a musculoskeletal model [118, 119]. This methodology substitutes the mathematical equations in musculoskeletal modeling with a network of interconnected artificial neurons imitating CNS function. Experimental data is used to train the model to find the mapping from joint moment time-series to a pattern of muscle tensions or EMG signals. This approach offers a computationally efficient method; however, it requires collecting and processing a significant amount of experimental data as well as optimizing the machine learning model configuration for the highest prediction accuracy, mapping efficiency, and real-time estimation ability.

Recently, machine learning models have been developed to estimate skeletal muscle tensions without explicit modeling of the physical muscle behaviors [106–109, 118]. Arjmand et al. [106] constructed ANNs to predict the trunk muscle forces required during static lifting. They used five input variables: thorax flexion angle,

load magnitude, the anterior and lateral distances of the reaching task, and the load handling technique. They disseminated that ANNs are more accurate than a regression-based mapping of input-output relationships. Cecchini et al. [107] also used ANNs to predict muscle force patterns for an athlete cycling. They validated the ANNs against an independent validation set and compared two alternative approaches using the Bland-Altman method [107]. Vilimek [109] used an ANN model for estimating the musculotendon forces around the elbow joint during a flexion/extension movement. Here, the network model's input parameters were the morphological and anatomical musculotendon parameters as well as the measured activation level. The author used that ANN for one specific muscle, which showed better results than the ANN for general muscles [109]. Instead of the previous ANN models [106, 107, 109] for considering the dynamic temporal relationships of the muscle forces, which is a time-series forecasting problem, recurrent deep neural networks (RDNNs) have been used by Dao [108]. Dao [108] used a long short-term memory model as an RDNN to approximate muscle forces from kinematic data. Rane et al. [118] trained a convolutional neural network on a set of kinematic, kinetic, and electromyographic measurements from 156 subjects to predict the muscle internal force magnitudes.

Many previous studies [106–109, 118] have predicted muscle tensions only and not activations in their solutions to the muscle redundancy problem. However, the inverse muscle model was not evaluated using the machine learning model in these previous studies (Figure 2.1(b)). Commonly, using the previous solution of the estimated muscle tensions, a second optimization problem should be used to estimate the muscle activations since there is no inverse dynamic model of the muscles. Providing an inverse dynamic model of the muscles with inputs of joint moments and outputs of activation variables can offer the ability to compare the subject's EMG signals via estimated activation signals. This inverse model completes the inverse dynamic simulation of the skeletal model for biomechanical analysis, post-rehabilitation analysis, and sports engineering/optimization.

A successful study by Gonzalez-Vargas et al. [120] developed a predictive model to generate muscle-specific excitation patterns for a given locomotion condition (i.e., speed and elevation) and a set of weightings characterizing the condition. However, the proposed pattern regression was used for cyclic gait and required subject-specific muscle weightings. To date, a couple of studies predict muscle activations using ANN machine learning and different kinds of kinematic or kinetic input signals [121–124]. Since the ANN's performance deteriorates with small variations in data, Sekiya et al. [125] used the linear logistic regression model, but its generalization performance is limited. Recently, Nasr et al. [38] demonstrated that RNNs could outperform ANNs (forward networks) for direct muscle modeling (EMG to biomechanical signal). We plan to use an RNN to predict muscle activities from biomechanical signals (inverse to the forward model in [38]). To the best of our knowledge, the inverse muscle model using the recurrent machine learning method is novel and proposed for the first time (Figure 2.1(c)).

2.3 Passive and active exoskeletons

The exoskeleton design splits into three categories based on sources of assistance: fully passive, fully active, and active-passive.

Passive exoskeletons are equipped with a mechanism to store potential energy from human motion for future use [26, 34]. For example, upper limb passive exoskeletons are mostly meant to unload the weight of the arm and any lifted tool while the arm is elevated, decreasing muscular strain [126]. The mechanism is assembled from a spring [26, 127], an elastic band [128], a counterweight [129, 130], or composed of a pneumatic cylinder for gravity compensation [131, 132]. The main advantages of entirely passive exoskeletons include being lightweight, portable, safe, affordable, high torque-weight ratio (TWR), less complex concerning sensing and control, and requiring less maintenance than other categories [26, 34, 35, 133, 134]. Consequently, a significant interest in studying, developing, commercializing, and utilizing passive industrial exoskeletons does exist [34, 37, 126]. On the other hand, passive devices are limited to providing a consistent torque profile for gravity-concerned movements only [14, 26, 34, 135]. However, the user may exert even more effort with a passive exoskeleton than without for some problematic dynamic activities [47].

On the contrary, active (powered) exoskeletons supply an assistive torque through a flexible function of time instead of the constant function of joint angle [26, 34]. The assistive force/torque is achieved through a controllable component such as an electric motor, a hydraulic cylinder, or a pneumatic artificial muscle [19, 35, 36]. This actuator controls via a sophisticated control algorithm that requires various types of sensors and

energy sources [19, 28, 36, 131, 136–140]. Since the total torque source relies on an active component, the utilized actuator and the energy source should be powerful and, consequently, are bulky, heavy, irremovable, expensive, and have a lower PWR [37]. Increased device bulk and inertia resulting from heavier motors and battery packages would necessitate more effort from the human body, lowering motion efficiency through increased metabolic power use [141].

To mitigate the issues of active and passive exoskeletons, some devices use a combination of active and passive power transmissions [129, 142–146]. As a definition, an active-passive exoskeleton simultaneously has both active and passive power sources for at least one joint. Essentially, passive mechanisms contribute to lowering the size, weight, and required active torque, advancing the portability [14, 129, 147]. The motor primarily provides extra assistance and may be used in negative work harvesting like electric vehicles [31, 148]. Specifically, the operating time of an active exoskeleton robot can be increased by more than 20% thanks to an energy harvesting module [148]. Recently, this combination transmission system has been conceptually studied as semi-active, semi-passive, or quasi-passive on lower-limb [131, 142, 143, 148–152], upper-limb [127, 145, 147, 153, 154], and spinal [136] wearable robots. Using the combined systems (especially the rotational inertia) may decrease the angular velocity and acceleration of the user by more than 60% [154]. Controlling the active component of this combined system requires considering the interaction with the human [28], the motor control (activation) delay [129], and understanding the dynamics of both systems. We aim to evaluate a fully passive, fully active, and active-passive shoulder exoskeleton and compare them for the specific task.

Designing the power transmission source for an exoskeleton requires a deep understanding and broad consideration of both human and robot systems via different criteria. In designing exoskeletons, several research studies employed multibody dynamic models of human limbs that are kinematically constrained to the exoskeletons' structure [26, 155]. Regarding the redundant actuation in these multibody human-exoskeleton models, various cost functions were optimized in the designing process [14, 26, 126]. These optimization variables are classified into (I) kinematic, (II) dynamic, and (III) biomechanics parameters.

Examples of the kinematic variables in the optimization are the dimensions of the exoskeleton and the kinematic fit to subjects via geometric parameters [156], the linkage dimensions of the exoskeleton to maximize the force transmission to the wearer [157], or motion/task-based optimization [158, 159]. The dynamic parameter optimization for active exoskeletons is motor torques, or for passive exoskeletons, the characteristics of the spring within the passive mechanism [14]. In this scenario, the optimization goal was to reduce human effort in the form of joint (cumulative and peak) torques [8, 26, 160, 161], powers [8, 162], joint reaction force [163], and muscle forces [8, 164]. An example of a biomechanics parameter in optimization is muscle fatigue, which is reduced [127, 152, 165] or, specifically, muscle activity measured through electromyography [142, 149, 151, 152]. We aim to design the system by concerning human variables (e.g., joint torque, active torque, activation signal, and power) and robot properties (e.g., the power, the inertia, and the dimensions of the actuator).

The model-based optimization utilizes a dynamic model expressed by nonlinear problems using forward dynamics simulations of the multibody human-exoskeleton model [164] and an optimal controller as the human CNS [166]. Note that the exoskeleton's controller is introduced in section 5.2.5. The forward and inverse dynamic simulations utilize two equations classifications according to exoskeletons' joints/axis configuration and requirements. The first classification is the ordinary differential equation (ODE) that are used for multibody models in which the exoskeleton joints/axis coincide with the human joints [155]. The exoskeleton link can be grouped with human segmental bodies and simplify the multibody model [155]. The second classification is the differential-algebraic equations (DAEs) that are used for the multibody system with misaligned human-exoskeleton respective joints (primarily for higher than 1-DoF joints, for example, 3-DoF shoulder joints) [14, 26, 167, 168]. Using DAE instead of ODE can provide the opportunity to measure and evaluate human-robot interaction forces/torques. Due to the formed closed-loop kinematic chains in the multibody human-exoskeleton model, the governing equations are DAEs within additional interaction dynamic and the kinematic constraints model than the motion equation [14, 26, 46, 169]. However, solving the DAEs is challenging, time-consuming, and not straightforward. We aim to introduce the framework for designing exoskeletons using DAEs.

To the best of our knowledge, the literature has not supplied clear guidelines to design and optimize of complex fully passive, fully active, and active-passive systems alongside using a complicated human-exoskeleton interaction model. First and foremost, this research aims to create dynamic analysis techniques that may be used to construct and customize an upper limb exoskeleton for a specific worker or patient. The second objective is to decrease the user's effort by choosing motor and passive element characteristics. For task-based designing

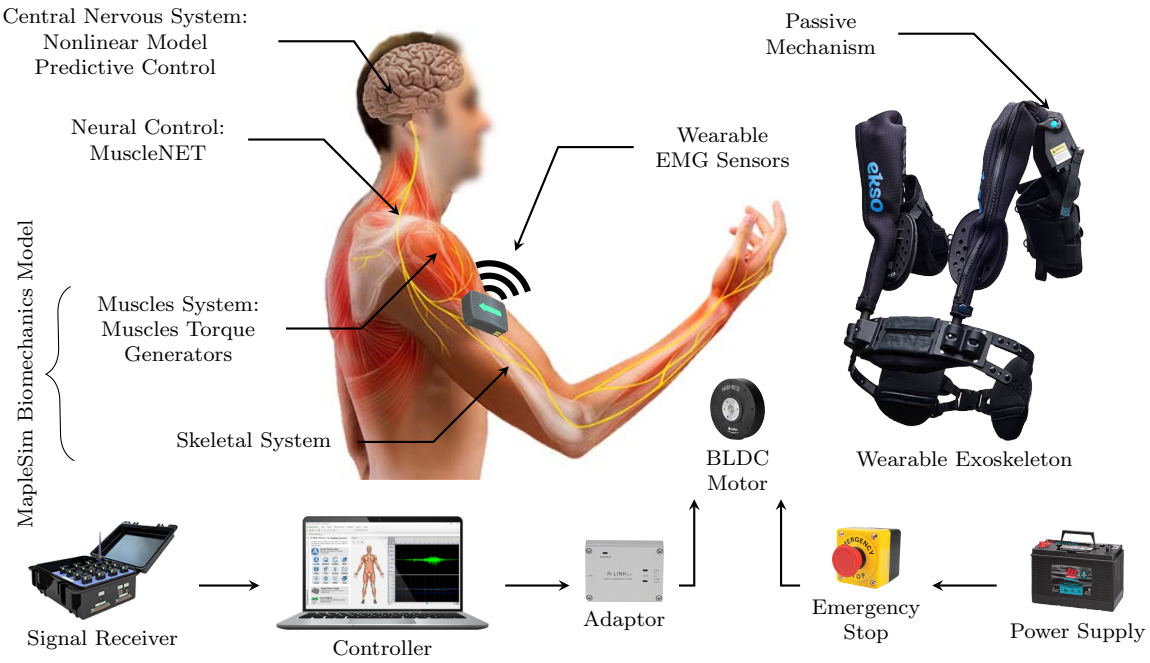


Figure 2.2: The essential components of the human-robot system. The human consists of the CNS (modeled using NMPC [166]), neural control (previously modeled by MuscleNET [38, 170]), muscle (modeled using the MTGs of human model), and skeletal system. The robot consists of mechanical and electrical hardware. The essential electrical components are wearable wireless sEMG sensors and the receiver (Delsys Inc, Boston, MA), controller, adaptor, BLDC motor, emergency stop, and power supply. The EVO (Ekso Bionics Holdings Inc, California, USA) upper limb exoskeleton is used as a wearable device with three levels of passive assistance.

and optimization, the data were recorded from participants doing functional activities.

2.4 Neuromuscular sensing and estimation

The state-of-the-art control of biomechatronic devices, such as robotic exoskeletons, prosthesis, and rehabilitation systems, employs a hierarchical design (high, mid, and low-level controllers [171, 172]) for rehabilitation, power augmentation, and artificial replacements of body segments [19, 38]. The high-level controller is in charge of interpreting the user's motion [54] or the wrench employed by the user (torque and force) [38, 171, 173]. The mid-level controller considers the determined intended motion or wrench from the high-level controller and regulates the biomechatronic actuators according to the human-robot interaction model and assistive/resistive regulations [38, 172]. The low-level controller is in charge of decreasing the actuator error.

The high-level control system may use human motion prediction as one of the feedback signals [54, 171]. To predict human motion, the three main categories of signals used are kinematic (position, velocity, or acceleration) [174], kinetic (force or torque) [175], and bioelectric signals (cognitive human-robot interaction) [19, 54, 176]. Bio-signals are one of the principal inputs of high-level controllers embedded within biomechatronic devices, rehabilitation robots, prostheses, and exoskeletons [19, 39, 54, 177, 178]. Amongst bio-signals, EMG, the measurement of the electric potential difference triggered by the nervous system to command muscle contraction, has demonstrated potential in illuminating human motion intention for controlling biomechatronic devices [19, 38, 39, 179]. Myoelectric control of biomechatronic devices has encouraged the study of more advanced sEMG control algorithms [180]. Accordingly, two main strategies have been used: (a) detailed muscle modeling and (b) machine learning methods.

One way to convert the sEMG signal to the muscle joint torque is to use a biomechanical muscle model. The most commonly used model is the three-component model, which takes inspiration from the lumped-parameter model developed by A.V. Hill for active and passive muscle tension behavior [82]. However, including a general

muscle model within a multibody model introduces various shortcomings such as muscle redundancy, a need to specify complex musculoskeletal geometry such as intricate muscle wrapping pathways and, difficult-to-fit parameters for each muscle, along with sensitivity to these additional parameters. One method that alleviates the muscle geometry, complexity, redundancy, and interpretation of sEMG signals within a control framework uses a machine learning model trained by experimental data.

The output of the machine learning method can be categorized into two groups: (A) a set of decisions, classes, or conditions [181–188], and (B) a data-driven kinematic or kinetic prediction value (regression-based) [189, 190]. The set of classes is usually used for state control and requires a large window of data to make the precise prediction class. Specifically, classification is a post-processing technique that requires the motion data after the motion task is completed. Thus, the regression method has superiority over classification method in terms of real-time processing. The regression-based schemes provide the capability of independent, simultaneous, real-time, and volitional control of each DoF. Recently, the regression of sEMG signal to biomechanical signals (human joint angle, velocity, acceleration, and torque) has been studied using machine learning models [191–194].

From the general input perspective, the machine learning models fall into two groups: (1) the single instantaneous value of the sEMG signal [195] and (2) pattern recognition techniques (discovering regularities in the pattern of the data) [39, 187, 196, 197]. Due to the sEMG signals' stochastic and time-varying nature, sEMG signal windows are utilized for pattern recognition instead of a single instantaneous value. The insufficient and stochastic nature of raw sEMG signals may be solved partially by using a considerable amount of training data and a complex machine learning model.

Transformation of a set or a window of sEMG data into a more readily implemented decreased set of features often uses two methods: (1) feature engineering and (2) machine learning. Feature engineering method's performance relies on the choice of features for extracting discriminative information from the sEMG data [185, 198–203]. Since the sEMG signal contains temporal or spectral information, the learning algorithms should not rely on specific and limited features. On the other hand, the deep-learning solution uses a hierarchy representation, which learns complex features by configuring and extracting stacks of features [204]. Deep learning architectures, like RNNs, CNNs, and RCNNs, are primarily used in image analysis or processing [205], speech recognition [206], and recently, bioinformatics [207].

Owing to the CNN's broad feature learning capability, they have become the most popular deep learning architectures that can perform classification or regression using multi-dimensional data [208]. Lately, CNNs have been applied in sEMG-based classification of hand or wrist gestures [181, 182, 188, 197, 209, 210], and limitedly in sEMG-based regression [211, 212]. Bao et al. [211] used a CNN for wrist multi-DoFs kinematics estimation. Ameri et al. [213] introduced a regression-based CNN that was developed for real-time sEMG based estimation of simultaneous wrist motions.

The RCNN is a particular type of CNN and can mine sequential data or time-frequency information. Chen et al. [189] used an RCNN as sEMG-to-Force mapping for multi-DoFs finger force prediction. They made significant improvements to the prediction results with an RCNN consisting of LSTM networks. Moreover, an RCNN has been utilized for offline estimation of upper-limb motions using sEMG frequency bands [214, 215] and wrist motion intention recognition using the time-frequency spectrum of sEMG signals [216].

The main similarity of the previous studies is that the estimators depend only on the sEMG signal as an input. However, by considering the biomechanical muscle model, the muscle tension relies on the joint's kinematics in addition to the sEMG signals that function as an activation. The joint angle, velocity, and acceleration partially define the muscle's tension, with the joint angle specifically defining the muscle fiber orientation. The fiber orientation indicates the direction of tension applied on the bone, and consequently the joint torque. Thus, we propose that in addition to the sEMG signal, the joint's kinematics be used as input signals to a machine learning model.

Using wearable myoelectric sensors for daily living activities has challenges and limitations discussed in the following paragraphs. One challenge with using myoelectric sensors is that signal disconnection or the wearable sensors losing contact from the skin is inevitable [217]. sEMG signals may have signal loss due to electrode disconnection, muscle artifacts, or impractical for deep muscles [217, 218]. In these cases, the myoelectric-based control would likely fail. Thus, preparing a robust prediction model for signal disconnection is necessary.

The second challenge relates to the delay of mapping the sEMG signals to the outputs in the control algorithm. Specifically, recording, transmitting, digitizing, filtering raw signals, and employing complex equations or

mapping via a multilayer neural network to the output signal is time-consuming. Having a model that can predict future human intent can help in predictive control. Theoretically, the electromechanical delay phenomenon is defined as the time delay between the onset of the EMG and the onset of the human motion [219, 220]. In other words, since the muscle activation signals are generated before the production of torque at a joint, it is possible to use EMG signals to predict motion intent [221, 222]. Thus, it is hypothetically possible to predict future motion from the current sEMG signal. A solution to predict motion intention or future joint torque using sEMG signals is beneficial in myoelectric control.

The third challenge emerges from differences in the configuration of robotic joints compared with human joints and a requirement of extra conversion. For example, a shoulder exoskeleton may have one joint for the elevation angle, although a human shoulder joint may be defined by SFE, SAA, and SMLR [14, 26]. Therefore, transferring the human joint variables to robot joint variables is necessary and sometimes uses time-consuming iterative calculations [26]. To satisfy the mentioned objective of the control-oriented model, we propose to have a model with outputs of the control-oriented variables (robot assistive torque or joint angle) instead of human variables (human joint torque or joint kinematics). The conversion should be done offline, and the resultant variables are the mapping target of a machine learning model.

Determining the optimal assistive wrench profile supplied by biomechatronic devices throughout human motion is an ongoing challenge [223, 224]. Most researchers have used a time-consuming optimization method in offline simulation [157, 160, 164]. For example, Zhang et al. [225] introduced an iterative process that optimizes the assistive torque profile for one stride of walking (in one hour of computation) to minimize human energy cost. Additionally, the human-robot system has redundant actuation, and finding the optimum torque profile is complicated. Although fast optimization methods have been suggested [166, 224, 226], they struggle to run real-time with sufficient model fidelity. Thus, the control loop of active biomechatronic devices should preferentially use the offline optimized result of an iterative optimization process. Consequently, inefficient online optimization for assistive control loops is not necessary.

The myoelectric-based control should be fast in terms of computation costs [177] and robust in terms of responsiveness to myography challenges [177, 227, 228]. The controller should predict the future state [229], define the assistive torque profile, and require no extra conversion of human to robot metrics.

Combining the mentioned solutions produces MuscleNET (defined as the machine learning models of sEMG signals to regression of kinematic and dynamic variables), a robust predictive control-oriented model for mapping sEMG to biomechanical or robot's variables using for myoelectric control of biomechatronic devices. This model representing muscles in a musculoskeletal simulation, sports biomechanics simulation, controlling active exoskeletons and prostheses, developing model-based assistive/resistive robots, and post-rehabilitation analysis. To the best of our knowledge, using interrupted EMG signals to train a machine learning model, along with the offline optimization of the required assistive torque profile as the output, has not been studied yet.

2.5 Assist-as-needed control of active exoskeleton

The human CNS and the robot control system simultaneously modify the behavior of the closed-loop system with different approaches of robot's hierarchical control [19, 137] and human's optimal control [166]. Desirable performance of wearable devices entails good coordination between these two controllers [230]. This can be accomplished by developing an effective robot controller that considers both the wearer and the robot in its design. Additionally, the robot controller should be appropriately modified as the human adapts to the device to account for human-robot coadaptation. For this purpose, researchers have either conducted HITL studies or personalized computer simulations [231–234]. Since safety is a major consideration in biomechatronics research, this work adopts the both approach; the results of this research pave the way for future experimental implementations.

The control strategy that assists users according to their capabilities is widely known as AAN [28, 235]. Ideally, AAN control supports the user only when further assistance is necessary to complete a given task. To do so, the underlying intention of the user, i.e., what movement the CNS desires, should be considered, especially for power-assist wearable robots [28]. To elaborate, the human operator is both the control supervisor as well as a component of the human-robot complex system [18]. Said user makes the decisions on which movement to

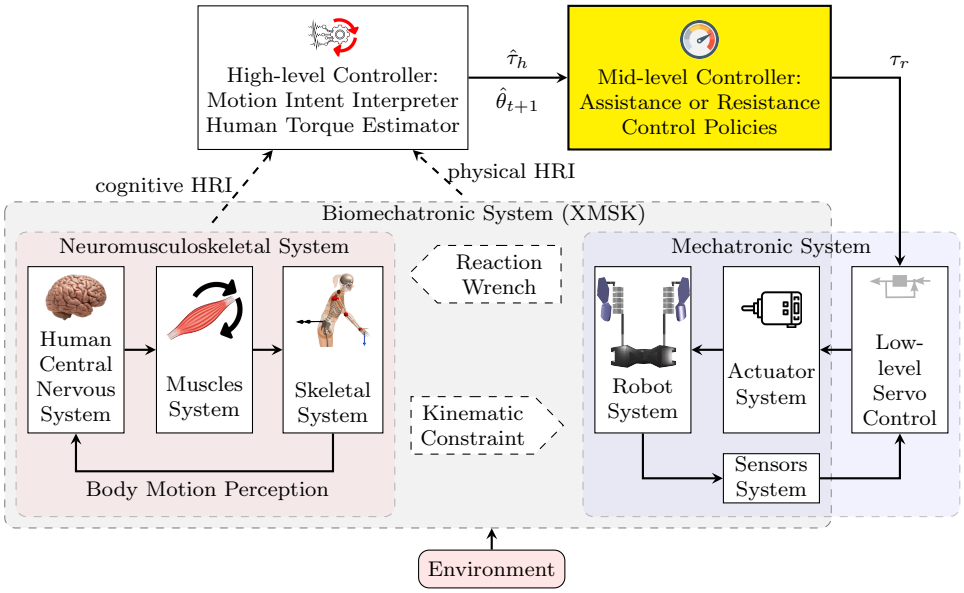


Figure 2.3: Schematic of HREI system with hierarchical strategy control. $\hat{\tau}_h$ is estimated human joint torque and $\hat{\theta}_{t+1}$ is predicted/desired joint angle computed by the high-level controller.

execute, and the exoskeleton follows suit. Difficulties arise as the robot and the human operator continuously exchange feedback information bidirectionally [28, 45]. As a result, the exoskeleton system's performance inherently depends on how the human may be adapting to the device, e.g., increased strength or slacking. From a control viewpoint, human-exoskeleton systems must coordinate two control units in parallel (shown in Figure 8.1): the human CNS and the exoskeleton controller [28].

The state-of-the-art control of robotic rehabilitation, assistive, exoskeleton, and prosthesis systems utilizes a hierarchical design: high, mid, and low level controllers [28, 137, 172] (Figure 2.3). The high-level controller is responsible for interpreting the motion intent of the user (primarily for robotic prosthesis systems) or the exerted torque and force by the user (mainly for wearable robotic systems); the two approaches are categorized into cognitive human-robot interaction (cHRI) [236, 237] and physical human-robot interaction (pHRI) [238, 239]. The mid-level controller transforms the determined intended motion or wrench (force/torque) from the high-level controller into the desired motion or wrench trajectories for the low-level controller [240]. Finally, the low-level controller is responsible for following the desired states (from the mid-level controller) by considering the measured device states (from the robotic sensors) and directly commands the robotic actuators [137, 172]. The current chapter focuses on the mid-level controller.

Apart from the quality of the sensors, the accuracy of the interpreted intent/wrench, the actuator, and the low-level feedback control, the mid-level controller plays a vital role in users' experience [241]. It governs the HREI [40]; the HREI depends on the robot's mid-level cooperation control. Developing effective robotic systems requires the consideration of application targets (clinical or industrial), power augmenting purposes (resistive rehabilitation or power-assistive fixture), and the human interaction adaption level [40, 241]. An inefficient control policy may limit the normal human range of motion or increase discomfort due to unnatural motion [242, 243]. Executing natural motions from human intent is challenging, applying volitional natural human adjustment is difficult, and the overall applicability is limited [19]. Recently, the AAN strategy, which operates based on enhancing motor activity but not replacing it, shows promising results [233, 238, 244–246].

So far, common strategies have been used for controlling wearable robots: finite state machines (prerecorded motion) [247, 248], master-slave [40], proportional gain [28, 236, 249, 250], proportional-integral-derivative (PID) [251], decision-based fuzzy mapping [28, 252–254], CTM [28], impedance [138], MPC [255–259], and combination control strategies [240, 251]. Finite state machines use prerecorded motions usually purposed for cyclic [248] (e.g., walking, running, and stair climbing) or categorized movements (e.g., hand/finger posture). Problems include that the maneuver evoked is non-volitional once triggered, and that the controller does not use corrective feedback at any stage. Although the master-slave control uses human motions and simultaneously

commands the same motions, the robot wearer feels an uncomfortable contact force if there is a human-robot size inconsistency [40]. A proportional controller was established to create a proportional command from the high-level input, often a biological signal [236, 237, 249, 250, 260]. However, as the the biological signal-torque ratio is not proportionate, this method causes the exoskeleton user to move uncomfortably [19, 236]. To address this issue, novel controllers recruit kinematic/kinetic signals in addition to the biological signal. Still, tuning the gains of PID controller is time-consuming in addition to task- and subject-specific [261]. Developing a decision-based fuzzy controller system avoids the need for a complex MSK dynamic model [252–254]; however, generalizing this fuzzy model is still a challenge as it reduces a wide range of biomechanical signals to just a few variables [28]. The CTM control strategy uses a complex human limb model within the control unit to maneuver the exoskeleton [28]. However, these methods are not efficient and robust in the presence of model uncertainties, external disturbances, sensor noise, control delay, and do not consider the energetic cost in the actuators. Researchers have used a hybrid assistive control since signal-based control strategies are inefficient for different motions [262]. Depending on the motion requirements, the controller was segmented into various control phases [262].

MPC is well-regarded as a control method as it is capable of considering the input and output signal limitations as constraints [259, 263, 264]. Predictive control approaches are receiving increasing attention from researchers with frequent use in a variety of dynamic systems [255–257, 257, 258, 265, 266]. Reducing the cost function in the MPC control formulation guarantees the tracking error and the smoothness of the assistive device behavior [264]. The exoskeleton user's comfort may be significantly and favorably impacted by this smoothness [263]. The MPC law is also known for its capacity to anticipate future disruptions and to implicitly take into account constraints on the kinematic outputs and wrench inputs, which may ensure the user's safety and overall control stability [263]. Likewise, it can counteract external disturbances and parametric uncertainties [267]. However, two of the main drawbacks of MPC are the computational costs associated with iterative processes [268] and the dependency on model accuracy [264].

Overall, the inefficiency, application limitation, uncomfortable movement, and usage anxiety challenges are still important issues to resolve [241, 242]. They result from the following: ignoring the limb dynamics or assuming constant limb dynamics, neglecting the robot stiffness and inertia, adapting to an overly simplified strategy that does not capture the mathematical relationships between variables, and struggling with tuning/adjusting the controller for participants. The system dynamics play an essential role since the applied torque to robots, external wrench (from disturbance or manipulation object), and the reaction wrench on the user lead to the motion according to the system dynamics [243]. The sources of human-robot-environment system dynamics consist of inertial properties of human segments and robot links (proportional to the joint acceleration, according to Newton's law), centripetal force (proportional to the joint velocity squared and toward the center of rotation), Coriolis effect (interaction of two rotating links/limb), gravity wrench (caused by gravity), and external wrench (caused by friction, environment, or heavy object handling).

For example, in the presence of an external load, the amount of necessary active torque increases compared to the free human limb movement. The robot's dynamics (stiffness and inertia) and the environment/external load result in extra necessary torque. The AAN mid-level control of wearable robots should be designed to provide this extra torque [244, 246]. As a result, the primary goal of this project is to create a model for a mid-level controller that includes two distinct parts: the robot and the environment. This method is similar to a combination of torque and gravity compensation control approaches [240, 269]. Previous researches have converted the HREI system to an overly simplified rule. Consequently, adjustments are necessary for any changes in the dynamics of limb, robot, and environment [19, 249, 253, 262]. To overcome these challenges, or at least mitigate the inefficiencies and limitations, considering the human-robot-environment dynamic interaction is a viable solution.

The important controller in the integrated human-exoskeleton system is the human CNS (comprising the brain and spinal cord). Among other things, it is in charge of regulating and coordinating MSK kinematics and kinetics despite redundant actuators (muscles) [166]. Driven by experiments, a common and well-acknowledged theory in the literature suggests that the CNS coordinates movement in an optimal fashion [270, 271]. This coordination relies on commands that combine features of (I) motion prediction (feed-forward control) and (II) error correction (feedback control) [270]. Essentially, states are projected forward in time using an internal model (IM) to estimate measures like feasibility and effort. An internal model acts to encode complicated MSK system dynamics, usually with simplifications. Consistent with studies, the CNS uses these projections to determine muscle excitations that minimize some cost function compromised of kinematic, kinetic, energy, and

muscle terms [166, 270]. These feedforward commands are then combined with available sensory information to correct any errors caused by model uncertainty or external disturbance. In simulations, NMPC is capable of achieving this dual-purpose control feature (prediction and correction) [166]. NMPC with a finite horizon formulation predicts the optimal motion (feed-forward) using an IM that represents human MSK dynamics, and it leverages sensory data to correct prediction mistakes (feedback) [166]. The quality of IM and the gain of NMPC can be changed to mimic short or long-term usage of assistive exoskeleton robots in which the CNS learns the IM over time [28, 272].

However, in most mid-level controller assessment studies, the central neural system is modeled with a PID controller [273, 274]. The wearer can adapt to the robot dynamics and the controller in the long term, yet the adaptation level is low in the initial and short-term experiences. We aim to test a mid-level controller by modeling the CNS as an optimal controller [166] for the initial, short-term, and long-term experiences.

As mentioned prior, it is inherently challenging to control robotic exoskeletons due to the need for simultaneous cooperation or competition with the CNS. In particular, it is not a clear how a more traditional mid-level AAN approach with CTM control (AAN-CTM) might perform against AAN using optimization methods, i.e., MPC. Using an optimal strategy should provide better assistance (e.g., less tracking error), though that may be outweighed by excess computation times if iterative algorithms are necessary. Likewise, if the actions of the mid-level controller are already designed to follow the CNS, which performs predictive optimal control, then the use of a separate optimal control scheme for the exoskeleton may be redundant. Given its similarities to the CNS, it is therefore of interest to assess how these mid-level control schemes interact with an NMPC controller that mimics neural function prior to testing with a human. This interest aligns well with the recent push to incorporate computational simulations into medical and assistive device design [275]. With that said, to our knowledge, it is still unclear how the CNS considers the exoskeleton in the control problem, e.g., minimizing energy consumption in both muscles and robotic actuator.

Therefore, the current work aims to simulate performance of an AAN-MPC mid-level exoskeleton controller in assisting a typical human user, simultaneously controlled via a CNS-NMPC model, complete an upper limb task. This performance was then compared to simulations that use a mid-level AAN-CTM approach. An experiment of the same task was also completed with a single participant to assess whether simulated performance is consistent with the real-world.

2.6 Chapter summary

In summary, this chapter reviewed the latest advances in modelling, design, optimization, control, and evaluation for a wearable active-passive exoskeleton robot.

Section 2.1 reviewed the necessities of a generic MSK model for assessing complex biomechanical problems, analyzing injuries, estimating joint torques during motion, optimizing motion in sports engineering, as well as designing exoskeletons and prostheses. Section 2.2 reviewed the requirement of a machine learning model as an alternative to static optimization for resolving the redundancy issue in inverse muscle models. Section 2.3 reviewed the current literature on designing the exoskeleton actuation, which is challenging and time-consuming due to the closed-kinematic loops in the 3D human-exoskeleton multibody model, the complicated interactions, and the interdependent selection of power transmission features.

Section 2.4 reviewed the development of estimation systems used for automated high-level control and intent recognition. Fusing data from kinematic sensors with sEMG, known as neuromuscular-mechanical data fusion, can improve intent recognition accuracies and decision times when compared to implementing either system separately. Section 2.5 reviewed the state-of-the-art hierarchical control, especially the AAN mid-level control for the active component of the exoskeletons.

Chapter 3

MapleSim Biomechanics Model

A MSK model is a valuable tool for assessing complex biomechanical problems, estimating joint torques during motion, optimizing motion in sports, and designing exoskeletons and prostheses. This study proposes an open-source upper body MSK model that supports biomechanical analysis of human motion. The MSK model of the upper body consists of 8 body segments (torso, head, left/right upper arm, left/right forearm, and left/right hand). The model has 20 DoFs and 40 MTGs, which are constructed using experimental data. The model is adjustable for different anthropometric measurements and subject body characteristics: sex, age, body mass, height, dominant side, and physical activity. Joint limits are modeled using experimental dynamometer data within the proposed multi-DoF MTG model. The model equations are verified by simulating the joint ROM and torque; all simulation results have a good agreement with previously published research.

First, this chapter presents the locations and inertia parameters of the upper body model scaled by anthropometric parameters in sections 3.2.1 and 3.2.2. Second, the biomechanical joint torque model is introduced in section 3.3 as a function of biomechanical kinematic variables, anthropometric parameters, and subject body characteristics. Then, the symbolic model extraction is presented in section 3.4, two simulation methods of inverse and forward are introduced in section 3.5, and the model is validated with the literature data in section 3.6. The method of model simulation is described and an example is provided. Finally, applications of the provided model are discussed in section 3.7.

3.1 Anthropometric and subject characteristics

The biomechanical MSK model relies on anthropometric values and subject body characteristics. These variables that adjust the MSK MapleSim Biomechanics model consist of:

- **Sex**: impacts muscle morphology and control, with men having bigger muscles that produce more consistent and stronger contractions than females [276]. Consequently, males have higher joint isometric and isokinetic torques than females [70]. However, the sex relation does not follow a rule of thumb formula for all joints and each-joint comparison should be reported separately. Furthermore, sex variables impact the anatomical locations, the COMs, and inertia parameters [68, 69].
- **Age**: has a significant impact on an individual's physical capabilities. The age variable is negatively correlated with all strength measures in both sexes [71, 276]. The muscle peak isokinetic strength and power in adults (more than 20 years old) gradually declines as age increases. However, muscle strength decline rates differ in individual muscles and, consequently, joints [70, 72, 73].
- **Body height**: has substantial power function connections to isometric force and torque measurements and positively impacts joint strength [70, 74, 76]. Furthermore, the body height impacts the anatomical locations and segment COMs [68, 69].
- **Body mass**: has a high positive connection to muscle mass and is positively associated with joint torque [70, 74, 75]. Additionally, body mass impacts the inertia parameters [68, 69].

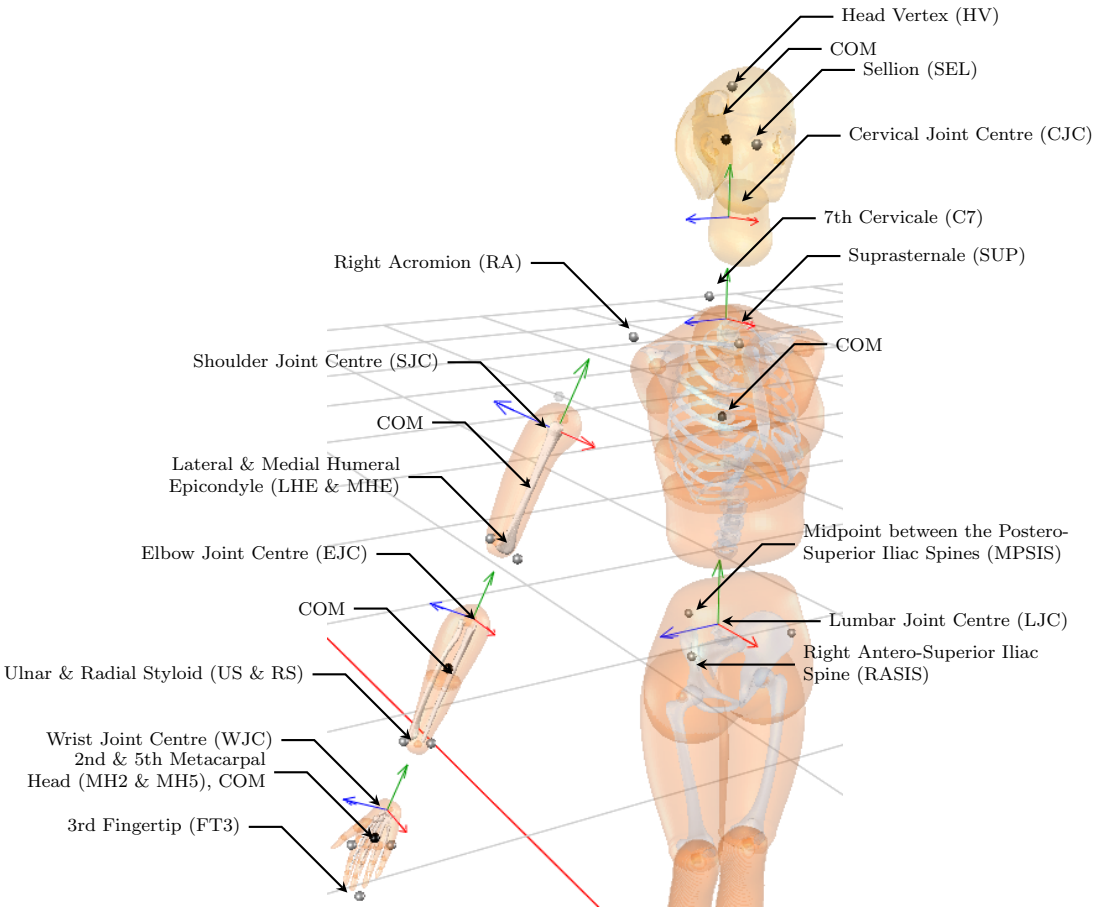


Figure 3.1: Locations of selected anatomical landmarks and the consequent orientations of the segment coordinate system according to ISB standard [68, 278].

- **Dominance**: appears to be more widespread in the upper extremities with minor variances [71, 77–79, 276]. The dominant side can provide a mean value of 9% more strength than the non-dominant side [77, 79].
- **Physical activity**: has an impact on muscle strength [277]. Specifically, adult athletes have significantly higher concentric and eccentric peak strength of most muscles [80].

The six variables of (I) sex, (II) age, (III) body mass, (IV) body height, (V) dominant side, and (VI) physical activity are used to adjust the subject MSK model in sections 3.2 and 3.3.

3.2 Skeletal dynamic model

3.2.1 Locations of anatomical landmarks

This section presents the location of anatomical landmarks and segment COMs of the upper body using anthropometric data: subject sex, body height. The pelvis and the lower limb are fixed to the ground, and the global frame is attached to the pelvis anatomical coordinate system according to ISB standard [278]. Because no suitable joint center estimation for the Thoracic joint center is available, the torso (a single segment) stands for Thorax and Abdomen [68].

The orientations of the segment coordinate systems are created, and the joint centers are determined according to the literature using specified anatomical landmarks (Figure 3.1) [68, 278]. The definitions and

Table 3.1: Abbreviations and descriptions for anatomical coordinate systems used in the upper body kinematic model according to ISB standard [68, 69, 278].

Segment	Origin /Axis	Description (refer to Figure 3.1)
Pelvis	O	Coincident with the LJC
	Z	The line connecting the LASIS to the RASIS
	Y	The line perpendicular to the plane through the LASIS, the RASIS, and the MPSIS, pointing cranially
	X	The cross product of the Y- and Z-axes
Torso	O	Coincident with the CJC
	Y	The line connecting the LJC to the CJC, pointing superiorly
	Z	The line perpendicular to the plane containing the LJC, the CJC, and the SUP, pointing laterally
	X	the cross product of the Y- and Z-axes
Head	O	Coincident with the CJC
	Y	The line connecting the CJC to the HV, pointing superiorly
	Z	the line perpendicular to the plane containing the HV, the CJC, and the SEL, pointing laterally
	X	the cross product of the Y- and Z-axes
Upper arm	O	Coincident with the SJC
	Y	The line connecting the EJC to the SJC, pointing proximally
	X	The line perpendicular to the plane containing the SJC, the LHE, and the MHE, pointing anteriorly
	Z	The cross product of the X- and Y-axes
Forearm	O	Coincident with the EJC
	Y	The line is connecting the WJC and the EJC, pointing proximally
	X	The line perpendicular to the plane through the EJC, the US, and the RS, pointing anteriorly
	Z	The cross product of the X- and Y-axes
Hand	O	Coincident with the WJC
	Y	The line connecting the midpoint between the MH2 and the MH5 to the WJC, pointing proximally
	X	The line perpendicular to the plane containing the WJC, the MH2, and the MH5, pointing anteriorly
	Z	The cross product of the X- and Y-axes

abbreviations of the segment anatomical landmarks are adopted from [68, 278]. Using these critical landmarks, the segment anatomical coordinate systems are defined by [68, 69, 278] and presented in Table 3.1.

The locations of segment anatomical landmarks and the COMs shown for the right body side in Figure 3.1 are adopted from [68]. Then, the raw position data was scaled to mean body height for each sex (male mean stature 1.77 m, female mean stature 1.61 m) and are presented in Table 3.2.

The right and left limbs are considered to be symmetrical. Transforming the position of anatomical landmarks and segment COMs from the right limb (reported in Table 3.2) to the left limb is done by the matrix in Equation (3.1).

$$R_{R2L} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{bmatrix} \tag{3.1}$$

3.2.2 Body segment inertia parameters

The biomechanical modeling and analysis of human movement and posture rely heavily on body segment inertial parameterss (BSIPs) [279]. BSIPs are often calculated using linear or non-linear regression equations published in the literature [68, 69, 280, 281]. This section presents the body segment inertia parameter estimation by anthropometric data: subject sex, body height, and body mass.

The mass of each segment (M), provided as a percentage of the total body mass, is presented in Table 3.3 [69]. The inertia tensor is calculated relative to the origin of segment coordinate axes. Explicitly, the segment inertia matrix (I) is expressed relative to the segment length (L) and the segment mass (M) with Equation (3.2) for the right limb. Scaled segment length L to body height, scaled segment mass (M) to body mass, and scaling factors r_{ij} for the inertia matrix are provided in Table 3.3.

Table 3.2: Scaled 3D positions of anatomical landmarks and segment COMs relative to segment anatomical coordinate systems for female and male (right limb) [68, 69].

Segment	Reference Coordinate System	Anatomical Landmark (refer to Figure 3.1)	Scaled 3D position to height (%)					
			Female			Male		
			X	Y	Z	X	Y	Z
Pelvis	LJC	LASIS (Left Antero Superior Iliac Spine)	5.40	-0.81	-7.39	4.41	0.40	-6.33
		RASIS (Right Antero Superior Iliac Spine)	5.40	-0.81	7.39	4.41	0.40	6.33
		MPSIS (Midpoint between the Postero-Superior Iliac Spines)	-6.71	-0.81	0.00	-5.76	0.40	0.00
Torso	CJC	C7 (7th Cervicale)	-3.54	2.11	0.00	-3.90	1.92	0.00
		SUP (Suprasternale)	2.80	-2.42	0.00	2.60	-2.54	0.00
		RA (Right Acromion)	-1.86	-1.37	11.24	-1.19	-1.75	11.81
		SJC (Shoulder Joint Centre)	0.81	-4.53	11.24	1.19	-4.12	11.81
		LJC (Lumbar Joint Centre)	0.00	-26.52	0.00	0.00	-27.01	0.00
		COM	-0.43	-11.55	-0.12	0.96	-11.36	-0.06
Head	CJC	HV (Head Vertex)	0.00	13.73	0.00	0.00	13.79	0.00
		SEL (Sellion)	4.91	7.89	0.00	4.69	7.74	0.00
		COM	-0.93	8.20	0.00	-0.85	7.68	0.00
Upper arm	SJC	RA (Right Acromion)	-2.17	3.48	0.25	-1.41	3.05	0.17
		LHE (Lateral Humeral Epicondyle)	0.00	-13.73	2.11	0.00	-14.58	2.32
		MHE (Medial Humeral Epicondyle)	0.00	-14.78	-2.11	0.00	-14.92	-2.32
		EJC (Elbow Joint Centre)	0.00	-14.25	0.00	0.00	-14.75	0.00
		COM	-0.99	-6.46	-0.37	0.28	-6.67	-0.40
Forearm	EJC	US (Ulnar Styloid)	0.00	-15.03	-1.68	0.00	-16.05	-1.86
		RS (Radial Styloid)	0.00	-15.65	1.68	0.00	-16.05	1.86
		WJC (Wrist Joint Centre)	0.00	-15.34	0.00	0.00	-16.05	0.00
		COM	0.31	-6.34	0.31	0.17	-6.67	0.23
Hand	WJC	MH2 (2nd Metacarpal Head)	0.00	-4.91	2.36	0.00	-4.86	2.60
		MH5 (5th Metacarpal Head)	0.00	-3.91	-2.36	0.00	-4.24	-2.60
		FT3 (3rd Fingertip)	0.87	-10.31	-0.81	0.40	-10.68	0.06
		COM	0.31	-3.42	0.19	0.40	-3.84	0.34

Table 3.3: The female and male segment scaled mass, scaled length, and scaling factors for the inertia tensor [68, 69]. i is the unit imaginary number.

Segment	Origin	Length definition	Scaled mass (%)		Scaled length (%)		Scaling factors for the inertia tensor $[r_{xx}, r_{yy}, r_{zz}, r_{xy}, r_{xz}, r_{yz}](\%)$	
			Female	Male	Female	Male	Female	Male
Torso	CJC	CJC to LJC	30.4	33.3	26.65	27	$[29, 27, 29, 22, 5, 5i]$	$[27, 25, 28, 18, 2, 4i]$
Head	CJC	CJC to HV	6.7	6.7	13.11	13.8	$[32, 27, 34, 6i, 1, 1i]$	$[31, 25, 33, 9i, 2i, 3]$
Upper arm	SJC	SJC to EJC	2.2	2.4	15.09	15.3	$[33, 17, 33, 3, 5i, 14]$	$[31, 14, 32, 6, 5, 2]$
Forearm	EJC	EJC to WJC	1.3	1.7	15.34	16	$[26, 14, 25, 10, 4, 13i]$	$[28, 11, 27, 3, 2, 8i]$
Hand	WJC	WJC to FT3	0.5	0.6	10.37	10.7	$[27, 18, 25, 12, 10, 12i]$	$[26, 16, 24, 9, 7, 8i]$

$$I_R = ML^2 \begin{bmatrix} r_{xx}^2 & r_{xy}^2 & r_{xz}^2 \\ r_{xy}^2 & r_{yy}^2 & r_{yz}^2 \\ r_{xz}^2 & r_{yz}^2 & r_{zz}^2 \end{bmatrix} \tag{3.2}$$

The segment inertia matrix of the left limb can be calculated by transforming the corresponding right limb inertia matrix of Equation (3.2) by Equation (3.3) using the matrix of Equation (3.1).

$$I_L = R_{R2L}^T I_R R_{R2L} \tag{3.3}$$

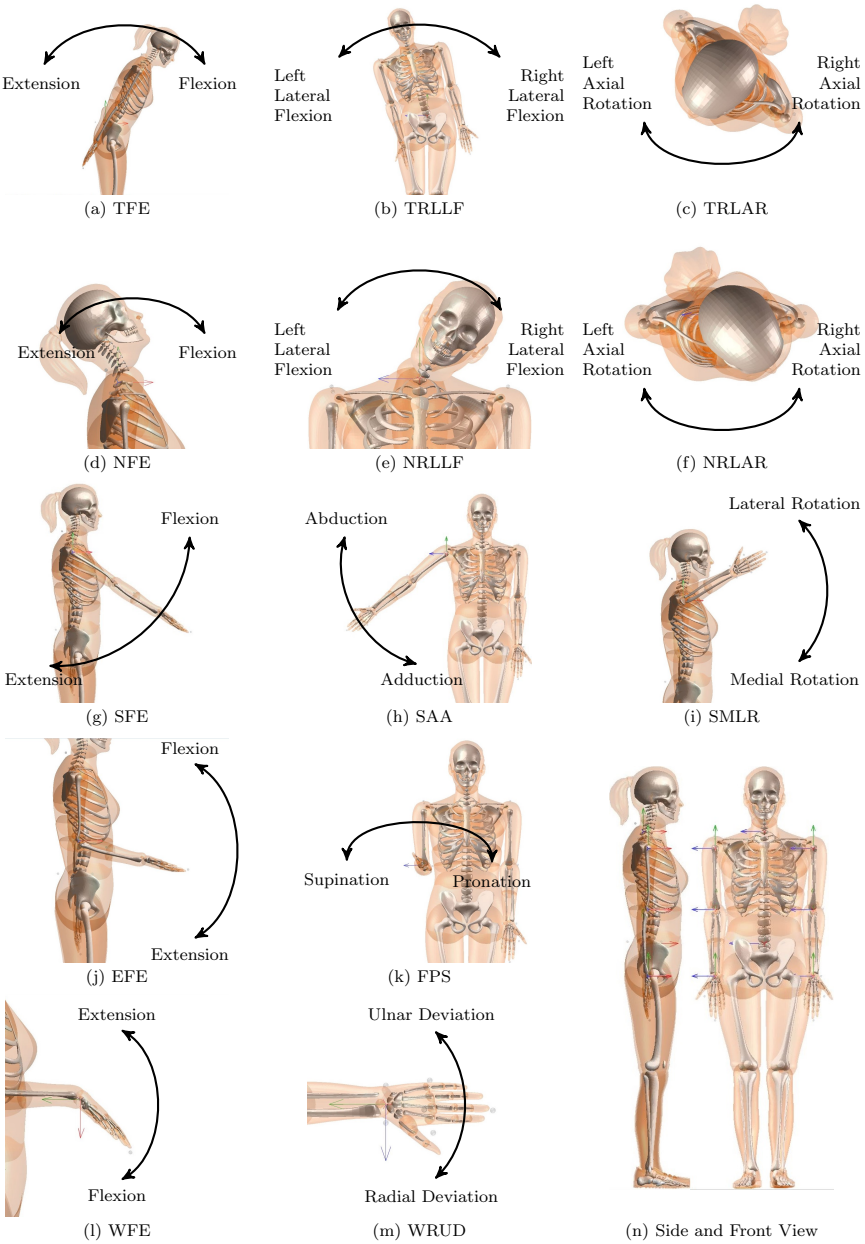


Figure 3.2: Depiction of MSK model DoF and directions for (a) TFE, (b) TRLLF, (c) TRLAR, (d) NFE, (e) NRLLF, (f) NRLAR, (g) SFE, (h) SAA, (i) SMLR, (j) EFE, (k) FPS, (l) WFE, and (m) WRUD. (n) Side and Front view with the initial posture (zero joint angles).

3.2.3 Degrees of freedom

The presented upper body kinematic model has 20 DoF: 3 DoF for the torso (TFE, TRLLF, and TRLAR), 3 DoF for the neck (NFE, NRLLF, and NRLAR), 3 DoF for each shoulder (SFE, SAA, and SMLR), 2 DoF for each elbow (EFE and FPS), and 2 DoF for each wrist (WFE and WRUD). The definitions and movements of DoF in the upper body kinematic model are presented in Table 3.4 (visualized in Figure 3.2). The left limb has reversed directions for WRUD, FPS, SAA, and SMLR compared to Table 3.4 (right limb information).

Table 3.4: Abbreviations and definitions of DoF, the mean ROM for female and male, the mean peak angular velocity of the upper-body joints, and the polynomial coefficients of torque-angle scaling functions (normalized isometric torque functions).

Joint	DoF	Axis	Direction	θ_{max} (rad)			$\lvert\omega_{max}\rvert$ (rad/s)		Polynomial Coefficient of Torque-angle Scaling Function in Equation (3.7)			
				Female	Male	Ref	Mean	Ref	p_1	p_2	p_3	Ref
Torso	TFE	Z-axis	- Flexion	0.547	0.620	[282]	2.531	[283]	0.9466	0.3969	-0.7370	[284]
			+ Extension	-0.470	-0.469	[282]	3.421	[285]	0.6093	0.7892	-0.2903	[284]
	TRLLF	X-axis	+ Right Lateral Flexion	0.354	0.412	[282]	1.344	[286]	0.9995	0.0485	-1.0856	[67]
			- Left Lateral Flexion	-0.354	-0.412	[282]	1.152	[286]	0.9671	-0.4460	-1.5055	[67]
	TRLAR	Y-axis	- Right Axial Rotation	-0.730	-0.868	[282]	3.857	[286]	0.9903	0.1555	-0.6199	[67]
			+ Left Axial Rotation	0.730	0.868	[282]	3.176	[286]	0.9740	-0.2235	-0.4801	[67]
Neck	NFE	Z-axis	- Flexion	0.972	1.004	[287]	1.134	[288]	0.7678	0.2558	-0.0337	[74]
			+ Extension	-1.288	-1.162	[287]	1.117	[288]	0.9834	0.0974	-0.1432	[74]
	NRLLF	X-axis	+ Right Lateral Flexion	0.753	0.680	[287]	0.384	[289]	0.9996	0.0266	-0.3572	
			- Left Lateral Flexion	-0.756	-0.770	[287]	0.384	[289]	0.9667	-0.2524	-0.4777	
	NRLAR	Y-axis	- Right Axial Rotation	-1.360	-1.326	[287]	0.541	[289]	0.9904	0.0492	-0.0630	
			+ Left Axial Rotation	1.331	1.252	[287]	0.541	[289]	0.9737	-0.0729	-0.0506	
Shoulder	SFE	Z-axis	+ Flexion	2.965	2.904	[282]	4.974	[286]	0.9992	0.0131	-0.0565	
			- Extension	-1.124	-1.092	[282]	3.491	[286]	0.6621	0.3796	-0.1066	
	SVAA	X-axis	+ Vertical Adduction	0.199	0.283	[290]	4.084	[286]	0.8255	-0.3164	-0.1433	
			- Vertical Abduction	-2.985	-2.989	[282]	4.538	[286]	0.8298	0.5322	0.1398	
	SHAA		+ Horizontal Adduction	1.018	0.948	[282]	4.272		0.9253	0.0919	-0.0182	
			- Horizontal Abduction	-2.393	-2.281	[282]	4.274		0.8988	-0.2512	-0.1557	
	SMLR	Y-axis	+ Medial Rotation	1.120	0.971	[282]	8.954	[286]	0.9390	-0.2040	-0.1702	[291]
			- Lateral Rotation	-1.571	-1.521	[282]	5.114	[286]	0.9676	-0.1777	-0.2430	[291]
Elbow	EFE	Z-axis	+ Flexion	2.402	2.344	[282]	6.231	[285]	0.4046	0.9405	-0.3714	[292]
			- Extension	0.093	0.101	[282]	4.451	[286]	0.4932	0.8047	-0.3194	[292]
	FPS	Y-axis	+ Pronation	2.859	2.774	[293]	5.184	[286]	0.7748	0.3948	-0.1731	[294]
			- Supination	-0.010	0.083	[293]	7.121	[286]	0.3414	0.7590	-0.2187	[294]
Wrist	WFE	Z-axis	+ Flexion	1.489	1.446	[282]	9.913	[295]	0.9996	0.0228	-0.3021	[292]
			- Extension	-1.391	-1.313	[282]	8.029	[286]	0.9860	0.1331	-0.3139	[292]
	WRUD	X-axis	+ Ulnar Deviation	0.281	0.284	[296]	2.740	[286]	0.9728	-0.4350	-1.7343	[297]
			- Radial Deviation	-0.401	-0.382	[296]	3.683	[286]	0.9608	-0.5288	-1.7812	[294]

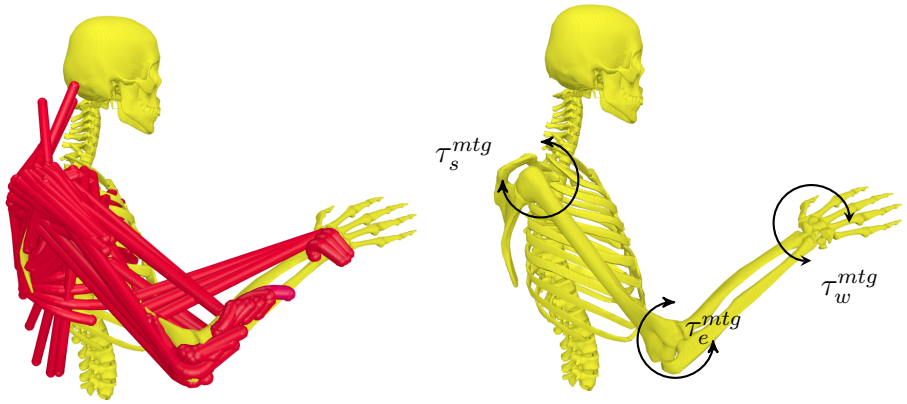
3.3 Biomechanical joint torque model

Human muscles have several recognized constraints that limit the amount of torque generated around each joint. The critical muscle's constraints can be listed as follows: force-velocity scaling, force-length scaling, nonlinear passive force generation due to ligaments and connective tissues surrounding a joint, and muscle activation dynamics.

Generally, a 3-element Hill-type muscle-tendon unit is used for muscle modeling [81, 82]. Furthermore, moving and wrapping points are used to simulate the constraints of the muscle-tendon geometry [82, 83]. Each muscle in Figure 3.4 is divided into two parts: an active contractile portion (creates active force) and a passive parallel-elastic element (with viscoelasticity). Moreover, a nonlinear series-elastic element describing the tendon's mechanical characteristics completes the model. These type of complex muscle models are difficult to obtain. Moreover, the model variables cannot be experimentally verified as they need in vivo human studies.

MTG model offers a joint torque that imitates the performance of muscles and joint constraints without the need for modeling muscle geometries and forces [84]. Explicitly, the MTG model mimics elements of muscle modeling: the velocity constraint and the joint angle constraint [56, 84] (shown in Figure 3.5). The MTG models, in torque-driven simulation, deliver torque at every joint that is a function of that joint's kinematics (monoarticular assumption) [298].

Each MTG model has a unique set of characteristics: an active component (generates the active torque limited by the joint's angle and angular velocity) and a passive component (mimicking the joint's small viscosity or friction and constraining the joint ROM). The human joint torque (τ_h) calculated by the MTG model is shown in Equation (3.4) [56, 84].



(a) Anatomically detailed muscles (b) Muscle torque generators

Figure 3.3: Schematic of the shoulder, elbow, and wrist joint, actuated by (a) Hill-type muscles and (b) MTGs. Note the complex muscle route geometry and redundancy encountered when using muscles.

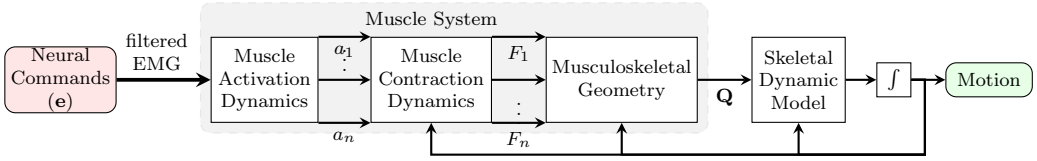


Figure 3.4: Forward simulation of a musculoskeletal system with multiple muscles. a_i and F_i are the activation signal and the muscle force in the dynamic muscle model, respectively.

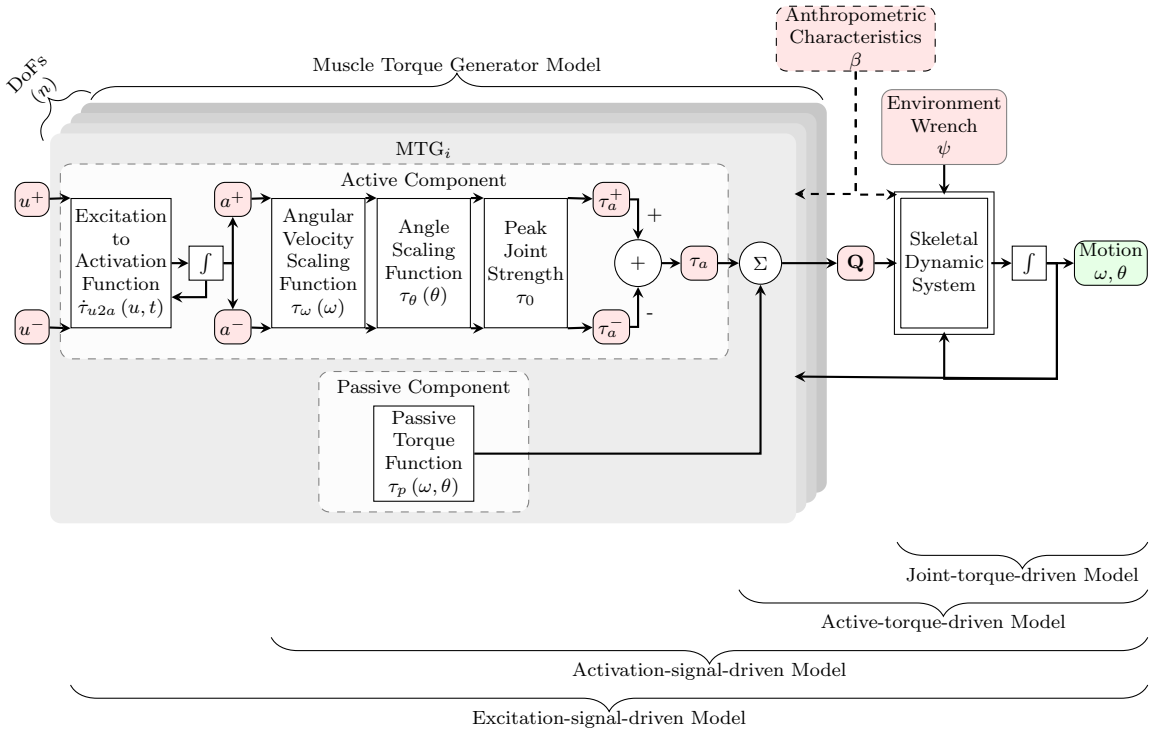


Figure 3.5: Schematics of forward simulation of a musculoskeletal system with MTGs. The MTG model consists of 2 active components for two excitation signals (positive and negative direction) and one passive component.

$$\tau_h(t) = \tau_{u2a}^+\left(u^+,t\right)\tau_\omega^+(\omega)\,\tau_\theta^+(\theta)\,\tau_0^+ + \tau_{u2a}^-\left(u^-,t\right)\tau_\omega^-(\omega)\,\tau_\theta^-(\theta)\,\tau_0^- + \tau_p(\theta,\omega) \tag{3.4}$$

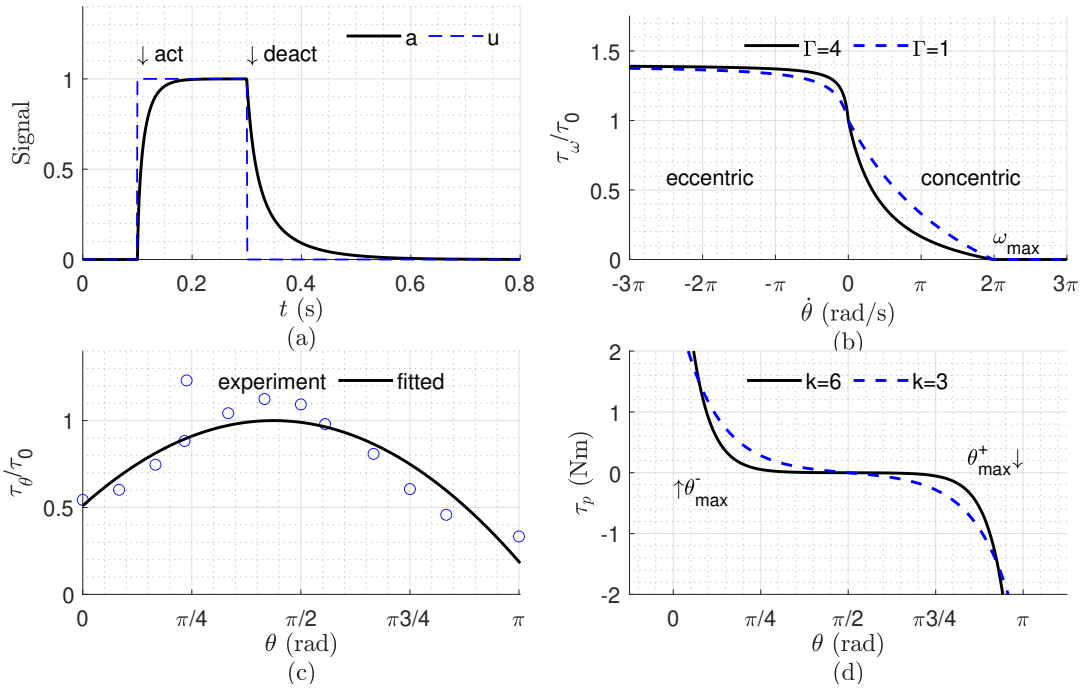


Figure 3.6: Example of MTG curves to mimic constraints of tendon and muscle forces: activation signal curve (a), the active-torque-angular-speed scaling curve (b), the active-torque-angle scaling curve (c), and the passive torque curve (d).

where

$\tau_{u2a}(u,t)$	the excitation-to-activation signal ODE function, which is introduced in section 3.3.5 and Equation (3.9)
$\tau_{\omega}(\omega)$	the active-torque-angular-speed-scaling function, which is introduced in section 3.3.4 and Equation (3.8)
$\tau_{\theta}(\theta)$	the active-torque-position-scaling function, which is introduced in section 3.3.3 and Equation (3.7)
$\tau_{p}(\theta,\omega)$	the passive torque function due to viscous damping and nonlinear stiffness, which is introduced in section 3.3.1 and Equation (3.5)
τ_0	the peak isometric joint strength producing capability

The product of $\tau_{u2a}(u,t)\,\tau_{\omega}(\omega)\,\tau_{\theta}(\theta)\,\tau_0$ is the net active joint torque that the biomechanical model generates. The control input to the MTG model is the activation signal $a(t)$ or the excitation signal $\varepsilon(t)$, which may take any value between 0 and 1. The examples of these four functions are shown in Figure 3.6.

Two active components of MTGs (42 total products of $\tau_{u2a}(u,t)\,\tau_{\omega}(\omega)\,\tau_{\theta}(\theta)\,\tau_0$) were used to control each DoF (joint axis of rotation): one for each direction (negative and positive) of joint rotation (two for each joint). One passive component of MTGs (20 total $\tau_{p}(\theta,\omega)$) were used for each DoF (joint axis of rotation): one for each joint. Consequently, the MSK system is over-actuated and requires actuation redundancy resolution, potentially with an optimization solution. The maximum torque range of each active component of MTGs was limited to τ_0^{+} for positive MTGs and τ_0^{-} for negative MTGs.

The parameters of Equations (3.4-3.8) can be calculated by fitting the functions to the experimental data. Accordingly, a computerized robotic dynamometer is generally utilized in sports and orthopedic medical applications for acquiring experimental data. The human dynamometer is a device that measures the net torque generated by an isolated joint in the body. It can be used in both isometric and isokinetic joint situations.

3.3.1 Passive torque function

Passive torques arise when the muscles, tendons, and ligaments surrounding the joints are strained [299]. Passive joint torques are intensified near the anatomical joint limits [299–301]. The double exponential function in Equation (3.5) is typically used to simulate the joint's viscous damping and nonlinear stiffness [302]. Consequently, this function limits the joint position within its ROM [302]. Specifically, when θ falls outside the ROM (those defined by breakpoints θ^{-}_{max} and θ^{+}_{max}), it creates a high restoring torque. In addition, the rotational damping term is included in Equation (3.5) to represent viscoelasticity.

$$\tau_p(\theta, \omega) = k_1 e^{-k_2\left(\theta - \theta^{-}_{max}\right)} - k_3 e^{k_4\left(\theta - \theta^{+}_{max}\right)} - c\omega \tag{3.5}$$

where

θ	the joint angle
k	the passive parameters determined by fitting the nonlinear function to experimental data
θ^{-}_{max}	the maximum ROM for negative direction, which is presented in Table 3.4
θ^{+}_{max}	the maximum ROM for positive direction, which is presented in Table 3.4
c	the rotational damping linear coefficient

The passive parameters (k_i) are generally determined by fitting the nonlinear function to experimental data and are between 0.5 to 6 [56]. k_i should be lower for the weaker joints like NRLLF, NRLAR, FPS, WFE, and WRUD, based on our experimental experience. The linear damping coefficient (c) is recommended to be about 0.1 Nm/rad [302].

In Table 3.4, θ_{max} for the WRUD, FPS, SVAA, and SMLR DoF of the left limb is negative.

3.3.2 Peak joint strength

Mean peak isokinetic torque for non-athletic females and males with their related anthropometric measures are presented in Table 3.5. The mean maximal isometric joint torque for any non-athletic adult male or female is estimated from parameters in Table 3.5. The intercept and sex parameters in Table 3.5 are extrapolated based on the anthropometric parameters and specific subjects reported in Table 3.5. The mean peak isokinetic torque for concentric contraction is regressed with Equation (3.6).

$$\tau_0 = (\beta_1 + \beta_2 \times s + \beta_3 \times a + \beta_4 \times h + \beta_5 \times m) \times \left(1 - \frac{\mu}{2} + \mu \times d\right) \times (1 + \varsigma \times w) \tag{3.6}$$

where

β_i	the torque-anthropometric parameters reported in Table 3.5
s	the sex parameter (for a male is 1 and for a female is 0)
a	the age of the subject in year
h	the height of the subject in meter
m	the body mass of the subject in kilogram
w	the physical activity or workout time per week
d	the dominant side (for a dominant side is 1 and for a non-dominant side is 0)
μ	the maximum strength difference of dominant with non-dominant side
ς	the effect of workout on peak muscle strength

In Table 3.5, the joint torque is reported for the average of both sides. The dominant side can provide a mean value of 9% more strength than the non-dominant side [77, 79]. Thus, the side parameter (μ) is selected as 0.09 for the arms. The central joints (torso and neck) are assumed to be symmetric in terms of maximum strength with $\mu = 0$. In Table 3.5, the joint torque is reported for healthy non-athletic subjects (w=0 for this group). For the upper-limb, the value of ς is equal to 0.065, which is a mean value reported by [277] for training time per week.

All the data reported in Table 3.5 are relevant to joint isometric torque except for the Horizontal Adduction and Abduction. Since the maximum isometric data are unavailable, the maximum isokinetic torque is reported.

Table 3.5: Peak joint torque for non-athletic females and males (mean of dominant and non-dominant limb) and estimated parameters of peak isokinetic torque for non-athletic subjects with the anthropometric measures: body age, height, mass, and sex.

Joint	DoF	Direction	Female						Male						β_1 (Nm)	β_2 (Nm)	β_3 (Nm/year)			β_4 (N)			β_5 (Nm/kg)		
			Torque (Nm)	Age (year)	Height (m)	Mass (kg)	Subject Numbers	Ref	Torque (Nm)	Age (year)	Height (m)	Mass (kg)	Subject Numbers	Ref			Mean	Range	Ref	Mean	Range	Ref	Mean	Range	Ref
Torso	TFE	-	109	30	1.65	60	43	[67]	162	31	1.75	75	59	[67]	-2.05	14.85	-1.6	-2.00 : -1.20	[73]	0.03	0.01 : 0.05	[76]	2.65	1.6 : 3.7	[75]
		+	189	30	1.65	60	43	[67]	288	31	1.75	75	59	[67]	77.95	60.85									
	TRLLF	+	101	30	1.65	60	43	[67]	141	31	1.75	75	59	[67]	-10.05	1.85									
		-	100	30	1.65	60	43	[67]	137	31	1.75	75	59	[67]	-11.05	-1.15									
	TRLAR	-	38	30	1.65	60	43	[67]	67	31	1.75	75	59	[67]	-73.05	-9.15									
		+	35	30	1.65	60	43	[67]	66	31	1.75	75	59	[67]	-76.05	-7.15									
Neck	NFE	-	15	30	1.64	65	5	[303]	30	32	1.77	77	11	[303]	22.41	13.95	-0.70	-1.05 : -0.34	[72]	0.36	0.20 : 0.53	[74]	0.2	0.06 : 0.33	[74]
		+	21	30	1.64	65	5	[303]	52	32	1.77	77	11	[303]	28.41	29.95									
	NRLLF	+	16	30	1.64	65	5	[303]	36	32	1.77	77	11	[303]	23.41	18.95									
		-	16	30	1.64	65	5	[303]	36	32	1.77	77	11	[303]	23.41	18.95									
	NRLAR	-	6	30	1.64	65	5	[303]	15	32	1.77	77	11	[303]	13.41	7.95									
		+	6	30	1.64	65	5	[303]	15	32	1.77	77	11	[303]	13.41	7.95									
Shoulder	SFE	+	34	25	1.62	62	12	[276]	61	25	1.72	75	12	[276]	-100.95	14.83	-0.17	-0.33 : -0.02	[70]	71	16 : 125	[70]	0.39	0.05 : 0.73	[70]
		-	45	25	1.62	62	12	[276]	95	25	1.72	75	12	[276]	-89.95	37.83									
	SVAA	+	43	25	1.62	62	12	[276]	84	25	1.72	75	12	[276]	-91.95	28.83									
		-	23	25	1.62	62	12	[276]	46	25	1.72	75	12	[276]	-111.95	10.83									
	SHAA	+	35	18	1.62	52	13	[304]	57	18	1.72	69	25	[305]	-97.24	8.27									
		-	31	18	1.62	52	13	[304]	52	18	1.72	69	25	[305]	-101.24	7.27									
	SMLR	+	27	25	1.62	62	12	[276]	51	25	1.72	75	12	[276]	-107.95	11.83									
		-	19	25	1.62	62	12	[276]	34	25	1.72	75	12	[276]	-115.95	2.83									
Elbow	EFE	+	33	23	1.66	67	24	[306]	63	24	1.80	81	30	[306]	-21.67	23.02	-0.12	-0.24 : -0.01	[70]	23.7	-6.5 : 54	[70]	0.27	0.09 : 0.46	[70]
		-	29	23	1.66	67	24	[306]	53	24	1.80	81	30	[306]	-25.67	17.02									
	FPS	+	6.3	28	1.68	62	22	[307]	12	30	1.78	80	21	[307]	-46.90	-1.29									
		-	5.8	28	1.68	62	22	[307]	11	30	1.78	80	21	[307]	-47.40	-1.79									
Wrist	WFE	+	15	25	1.68	59	78	[70]	25	24	1.79	74	90	[70]	-23.07	6.85	-0.04	-0.09 : 0.02	[70]	21.5	2.8 : 40.3	[70]	0.05	-0.08 : 0.18	[70]
		-	7	22	1.70	60	26	[78]	11	22	1.80	76	6	[78]	-31.67	1.05									
	WRUD	+	7	22	1.70	60	26	[78]	11	22	1.80	76	6	[78]	-31.67	1.05									
		-	8	22	1.70	60	26	[78]	12	22	1.80	76	6	[78]	-30.67	1.05									

3.3.3 Active-torque-angle scaling function

The tendon force-length relation is not linear, and accordingly, the joint isometric torque is not consistent in the full ROM. The active-torque-angle relationship is typically represented in the literature using second-order or fourth-order polynomial functions [85, 86, 308, 309] with the summation notation in Equation (3.7).

$$\tau_\theta(\theta) = ramp\left(\sum_{k=1}^{n+1} p_k \theta^{k-1}\right) \tag{3.7}$$

where

θ	the joint angle
p_k	the polynomial coefficient (for a 2nd-order polynomial function), which is presented in Table 3.4
n	the order of polynomial functions

and $\tau_\theta(\theta)$ is normalized by the maximal isometric torque of each joint. Consequently, the value of $\tau_\theta(\theta)$ at the optimum angle is 1.

For fitting the polynomial functions to the experimental data, the maximum isometric torque should be available for all ROMs. Since there is not enough published information for all ROMs, for example, the shoulder joint [292], the fitted polynomial function may go to zero or even negative for ranges out of the reported angles.

Thus, we introduce two solutions. The first solution is using a dynamometer to measure the joint isometric torque for all ROMs that we use for shoulder joint (Flexion, Extension, Vertical Adduction, Vertical Abduction, Horizontal Adduction, Horizontal Abduction) for one male subject (25 years, 75 kg mass, and 1.71 m height) with a System 4 Dynamometer from Biodex Medical Systems (Mirion Technologies, Inc., NY, USA). The second solution is introducing "edge torque". The edge torque is the estimated or assumed joint torque at the maximum joint angle, which is calculated with the average of fitted polynomial and interpolation with nearest joint torque.

As mentioned, the fitted polynomial function must have a maximum value of 1. For the normalization, in contrast to [84], which used the maximum of joint torque, we use the maximum of the fitted polynomial function.

For the neck joint (NRLLF and NRLAR), no experimental data is available. Thus, we have used the torso-relevant data with scaling the torso's ROM to match the neck's ROM.

The left limb has reverse positive and negative directions for WRUD, FPS, SVAA, and SMLR. In Table 3.4, the even index of p_k for the WRUD, FPS, SVAA, and SMLR DoF of the left limb has a reversed sign.

3.3.4 Active-torque-angular-speed scaling function

Muscle strength is a function of muscle velocity. As a result, the angular velocity (ω) of the movement scales the activation torque (τ_a) (active-torque–angular-speed correlation). A rational piecewise instantaneous isokinetic torque function, associated with human muscle and predicted from a linearized Hill model structure, is implemented to represent this behavior according to [310–313]:

$$\tau_\omega(\omega) = \begin{cases} 0 & \omega > \omega_{max} \\ \frac{|\omega_{max}| - \omega}{|\omega_{max}| + \Gamma\omega} & \omega_{max} \geq \omega \geq 0 \\ \frac{(1-R)|\omega_{max}| + S\omega R(\Gamma+1)}{(1-R)|\omega_{max}| + S\omega(\Gamma+1)} & \omega < 0 \end{cases} \tag{3.8}$$

where

ω	the joint's instantaneous angular velocity
$\lvert\omega_{max}\rvert$	the absolute value of joint's peak angular velocity corresponding to zero torque production, as reported in Table 3.4
Γ	the shape factor influencing the curvature of the torque-velocity hyperbola concentric relationship
S	the slope of eccentric and concentric functions at zero angular velocity (the transition of eccentric to concentric phases) defined by [312]
R	the ratio of the maximum eccentric isokinetic torque (τ_m) over the maximum isometric torque (τ_0)

In Equation (3.8), for the $\omega \geq 0$ condition, $\tau_\omega(\omega)$ has a potential singularity as ω proceeds toward $-|\omega_{max}|/\Gamma$ instantaneous angular velocity. Consequently, the force-velocity scaling relationship differs from Hill-type's model [82] when the force is greater than the isometric tension (τ_0); it also has different output when $\omega < 0$ [312, 314]. As a result of $\tau_\omega(\omega)$ normalization by the maximal isometric torque of each joint, the value of $\tau_\omega(\omega)$ is 1 at zero velocity.

The value for Γ is about 3 [87]. The value for S is about 2 [56, 312] to 3 [315]. In addition, the value for R is around 1.4 [311] to 1.5 [312]. For negative direction movements (e.g., flexion, adduction, or supination), negative ω was used in Equation (3.8).

The absolute value of the joint's peak angular velocity for the shoulder horizontal abduction/adduction is not available; we have used the mean value of shoulder flexion/extension and vertical abduction/adduction instead.

3.3.5 Excitation-to-activation signal function

The muscle fiber recruitment status is represented by an excitation-to-activation signal function or simply activation dynamics [316]. The activation dynamics introduce a time delay between input and output, or the time between receiving a signal (neural excitation) and converting those signals into action potentials to activate muscle fibers by electrochemical processes in the muscle tissue [316]. Post-processed EMG data from experiments can be used to generate a collection of activations [38].

Similarly, for muscle groups around a joint, the activation signal generated by a neural drive is mimicked utilizing an extension of the activation curve [315] (independent of joint angle). Theoretically, the activation signal function incorporates muscle activation and deactivation time when the torque generator engages and disengages, respectively. Practically, this excitation-to-activation signal function is analogous to the temporal lag caused by the electrochemical activity generated by the action potential that leads to muscle contraction.

The modified function of an excitation-to-activation signal is suggested in Equation (3.9) adopted from the muscle fiber Hill-type model [317]. The excitation-to-activation dynamic model consists of a linear first-order differential equation and a non-linear function [318, 319]. The first-order differential equation converts the excitation signal $u\,(t)$ (from the CNS) to activation signal $a\,(t)$ which indicates the muscle excitation strength level.

$$\dot{\tau}_{u2a} = \begin{cases} \frac{u-a}{t_{act}(0.5+1.5a)} & e > a \\ \frac{(u-a)(0.5+1.5a)}{t_{deact}} & e \leq a \end{cases} \tag{3.9}$$

where

u	the excitation signal
a	the activation signal
t_{act}	the muscle activation time constants
t_{deact}	the muscle deactivation time constants

t_{act} can have a value of $40+0.25\times$age ms for an adult subject [320] and t_{deact} is assumed to be equal to $4t_{act}$. The excitation signal is in the range of $[0, 1]$ for monoarticular muscles. However, even for monoarticular muscles, the antagonist and agonist muscles activate simultaneously. In addition, the biarticular muscle provides tension for two joints [321]. However, the MTG models assume a monoarticular joint. To overcome this assumption, the impact of the excitation signal of one joint on another joint should be investigated [88]. Studying the excitation signal of joints is out of the scope of this chapter.

3.3.6 Multi-DoF joint implementation

Transferring these MTG torques to human joint torque depends on the joint type: single-DoF and multi-DoF. Most of human joints have consistent parameters in different orientations, denoted as one-DoF MTGs, e.g., torso, neck, forearm, and wrist. Consequently, the modeled MTGs are not a function of the coordinates for other-DoF. The MTG model for a single-DoF joint can directly apply to the human joint torque [86, 322].

A few human joints have variable parameters in different orientations, for example, the shoulder and hip joint (except for the axial rotation since it is configuration-independent). Specifically, the MTG parameters of multi-DoF MTGs are based on the orientation of other-DoF. For example, there are different MTG variables for the shoulder joint for abduction/adduction movement based on the value of the flexion/extension. Note that this joint orientation is defined with ZXY Euler angles. More specifically, when the upper arm has 0^o SFE in the z-direction, then the x-direction follows the MTG for SVAA. When the upper arm has 90^o SFE in the z-direction, then the rotated new x-direction follows the MTG for SHAA. The shoulder x-direction MTG elements are calculated with the extrapolation of MTG elements of SVAA and SHAA. Specifically, the x-direction shoulder MTG elements are computed with an extrapolation of SVAA (for 0^o SFE in the z-direction), SHAA (for 90^o SFE in the z-direction), and the shifted and mirrored SVAA (for 180^o SFE in the z-direction). MTG elements are

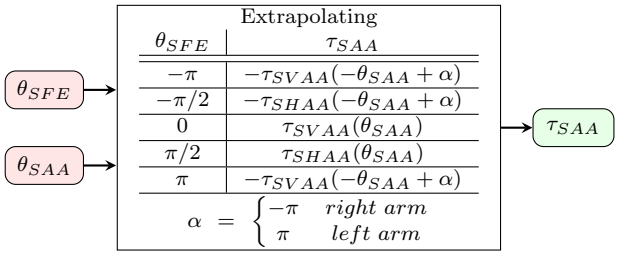


Figure 3.7: The schematic for extrapolating the joint torque for x-direction rotation (SAA) as a function of the z-direction orientation (SFE). The minus sign of torque in the second column of the extrapolation table is neglected for normalized functions, for example, active-torque-angle and active-torque-angular-speed scaling functions.

presented in Table 3.4 and Table 3.5. The extrapolated elements are the final values of $\tau_{\omega}^{-/+}(\omega)$, $\tau_{\theta}^{-/+}(\theta)$, and $\tau_{0}^{-/+}$ for both positive and negative directions as well as $\tau_{p}(\theta,\omega)$ functions. Based on our tests, the Piecewise Cubic Hermite Interpolating Polynomial (PCHIP) method provides good curve fitting to experimental results in section 3.6.3.

3.4 General structure of dynamic equations

The dynamic musculoskeletal model can be exported as (I) joint-torque-driven, (II) active-torque-driven, (III) activation-signal-driven, and (IV) excitation-signal-driven, as shown in Figure 3.5. The joint torque-driven level consists of only the dynamic model of the skeletal system. The active-torque-driven level consists of the skeletal system and the passive torque function. The activation-driven level consists of the dynamic model of the skeletal system and the MTG model without the excitation-to-activation dynamic model. The forth form of the model consists of the skeletal system and full MTG model with the excitations as input.

3.4.1 Joint-torque-driven model

The unconstrained dynamic equations governing the MSK system response are a set of ODEs since the chosen n generalized coordinates and generalized speeds are independent. By utilizing the "Multibody Dynamic Exports" and "Optimized Code Generation" module of MapleSim, the equations of motion are converted to simplified C code that can be used in the MATLAB/Simulink environment [323, 324].

The joint-torque-driven (MTG-excluded) dynamic model is in the form of Equation (3.10) with three matrices comprising the system ODEs.

$$\mathbf{M}_{n\times n}\left(\boldsymbol{\theta}_{n\times 1}\right)\dot{\boldsymbol{\omega}}_{n\times 1}=\mathbf{F}_{n\times 1}\left(\boldsymbol{\omega}_{n\times 1},\boldsymbol{\theta}_{n\times 1}\right)+\mathbf{Q}_{n\times 1} \tag{3.10}$$

where

n	the number of independent coordinates and speeds
$\boldsymbol{\theta}$	the column matrix of all joint angles
$\boldsymbol{\omega}$	the column matrix of all joint angular speeds
$\mathbf{M}$	the mass matrix
$\mathbf{F}$	the right-hand side of the dynamic equations, which consist of Coriolis, centrifugal, and gravitational effects
$\mathbf{Q}$	the applied joint torques, a column matrix containing $\tau_h(t)$ for all joints

3.4.2 Active-torque-driven model

The active-torque-driven dynamic model is similar to the form of Equation (3.10) with the inclusion of the passive torque function in section 3.3.1. The active torques are converted to human joint torques using Equation (3.11). The activation-driven model combines Equations (3.10, 3.11).

$$\mathbf{Q}_{n\times 1} = \left[\tau_a + \tau_p\left(\boldsymbol{\theta}, \boldsymbol{\omega}\right)\right]_{n\times 1} \tag{3.11}$$

3.4.3 Activation-signal-driven model

The activation-signal-driven (MTG-included) dynamic model is in the form of Equation (3.10) with three matrices comprising the system ODEs. The activation signals of positive and negative directions are converted to human joint torques using Equation (3.12), which is derived from Equation (3.4). The activation-driven model combines Equations (3.10,3.12).

$$\mathbf{Q}_{n\times 1} = \left[\boldsymbol{a}^{+}\boldsymbol{\tau}_{\omega}^{+}\left(\boldsymbol{\omega}\right)\boldsymbol{\tau}_{\theta}^{+}\left(\boldsymbol{\theta}\right)\boldsymbol{\tau}_{0}^{+} + \boldsymbol{a}^{-}\boldsymbol{\tau}_{\omega}^{-}\left(\boldsymbol{\omega}\right)\boldsymbol{\tau}_{\theta}^{-}\left(\boldsymbol{\theta}\right)\boldsymbol{\tau}_{0}^{-} + \boldsymbol{\tau}_{p}\left(\boldsymbol{\theta}, \boldsymbol{\omega}\right)\right]_{n\times 1} \tag{3.12}$$

where

$^{+}$	the positive direction of joint
$^{-}$	the negative direction of the joint

3.4.4 Excitation-signal-driven model

Including the first-order differential equation of excitation-to-activation, established in Equation (3.9), introduces another state to the system as shown in Equation (3.13). Some researchers bypass this extra state with an alternative formulation (e.g., exponential gain formula), but their alternative approach does not consider the delay from excitation to activation [317].

$$\begin{bmatrix} I_{2n\times 2n} & 0 \\ 0 & \mathbf{M}_{n\times n}\left(\boldsymbol{\theta}_{n\times 1}\right) \end{bmatrix} \begin{bmatrix} \dot{\boldsymbol{a}}_{2n\times 1} \\ \dot{\boldsymbol{\omega}}_{n\times 1} \end{bmatrix} = \begin{bmatrix} \boldsymbol{\tau}_{u2a}\left(\boldsymbol{a}_{2n\times 1}, \boldsymbol{u}_{2n\times 1}, t\right)_{2n\times 1} \\ \mathbf{F}_{n\times 1}\left(\boldsymbol{\omega}_{n\times 1}, \boldsymbol{\theta}_{n\times 1}\right) + \mathbf{Q}_{n\times 1} \end{bmatrix} \tag{3.13}$$

$$\mathbf{Q}_{n\times 1} = \left[\boldsymbol{a}^{+}\boldsymbol{\tau}_{\omega}^{+}\left(\boldsymbol{\omega}\right)\boldsymbol{\tau}_{\theta}^{+}\left(\boldsymbol{\theta}\right)\boldsymbol{\tau}_{0}^{+} + \boldsymbol{a}^{-}\boldsymbol{\tau}_{\omega}^{-}\left(\boldsymbol{\omega}\right)\boldsymbol{\tau}_{\theta}^{-}\left(\boldsymbol{\theta}\right)\boldsymbol{\tau}_{0}^{-} + \boldsymbol{\tau}_{p}\left(\boldsymbol{\theta}, \boldsymbol{\omega}\right)\right]_{n\times 1} \tag{3.14}$$

3.5 Simulation method

3.5.1 Inverse dynamic simulation

Inverse dynamics involves solving the torque-driven equations of motion (provided in section 3.4.1) using the kinematics and the body characteristics to compute the resultant forces and moments at specific joints [325].

The MTG models have different positive and negative torques for one joint. This over-actuation requires actuation redundancy resolution, primarily by optimization methods. Variations of two optimization techniques are used to compute positive and negative active MTG torques: static and dynamic optimization. Static optimization (SO) is considered as an inverse approach: MTG torques are optimized in SO using time-marching, which means that time frames are solved consecutively [57]. Once the positive and negative active MTG torques are calculated, the inverse dynamic simulation can compute the positive and negative MTG activation signals.

Example

The resultant torques and signals are evaluated for a given motion, as an example of model application. The pick and place motions depicted in Figure 3.8 were measured from one subject (Male, 29-years, 1.80 m, 85 kg, right-handed, 1 session workout per week). The object weight is 40 N, and we assume the weight is equally spread on both palms of hands. The raw measured joint angles are filtered with a 5 Hz low pass filter (Figure 3.8). Using inverse dynamic simulation (given joint angles, velocities, and accelerations), the required torques for 20 joints are calculated using equation (10). The activation torques for 40 MTGs are derived using an optimization

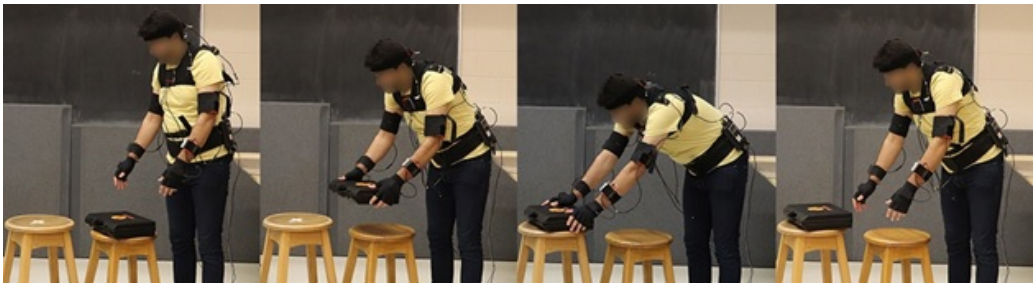


Figure 3.8: A depiction of object manipulation task performed for kinematic data collection. The participants were expected to lift and lower an object between two target locations in the sagittal plane. The joint trajectories were measured using an upper body Xsens MVN suit (Xsens Technologies, Enschede, Netherlands).

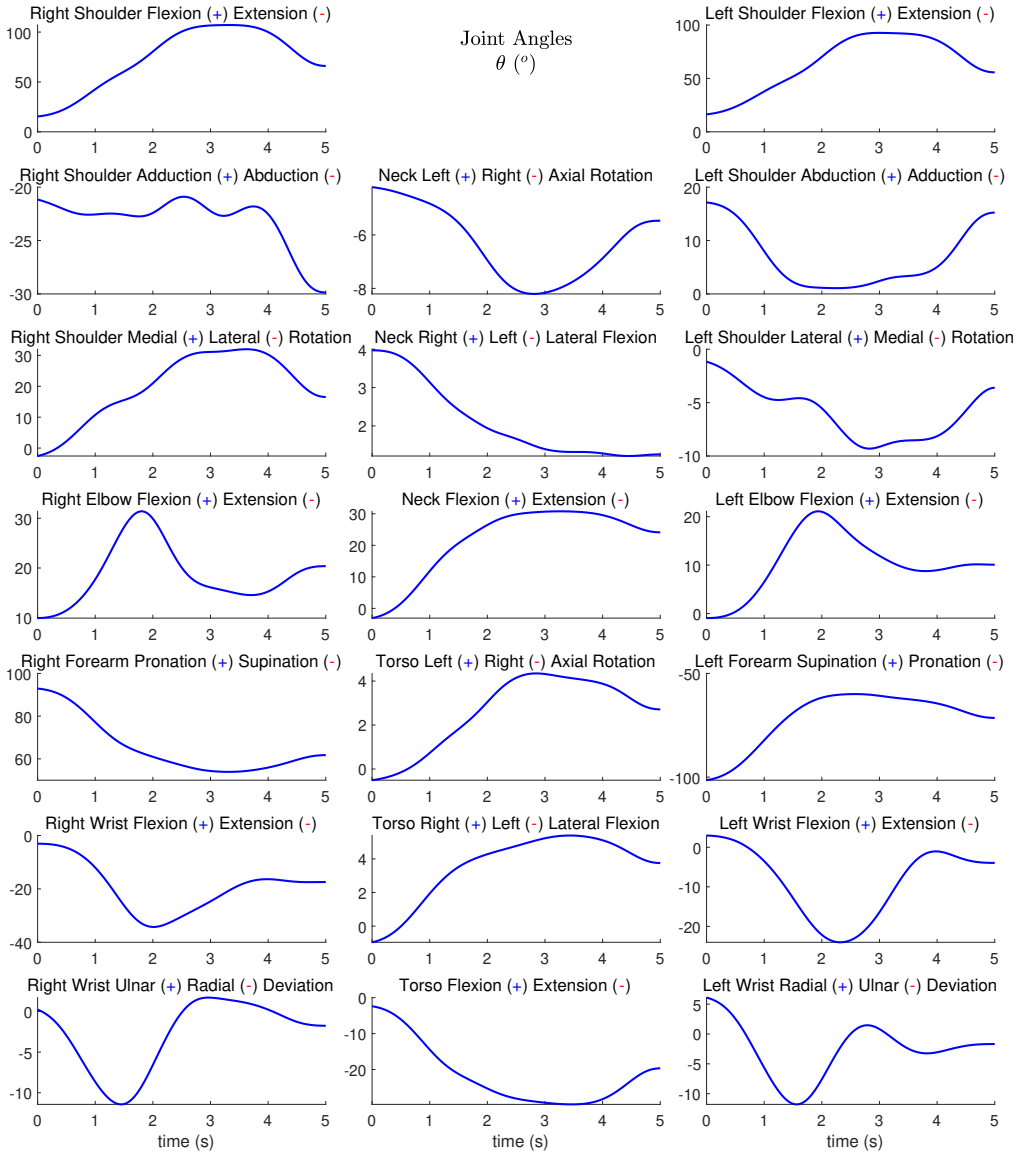


Figure 3.9: The trajectories of 20 joint angles (^{o}) acquired from the motion recording process and 5 Hz low pass filtering.

method and plotted in Figure 3.10. The activation torque is minimized to compute the joint torques using the "fmincon" function of MATLAB (R2021b. MathWorks, Natick, MA, USA). The activation and excitation

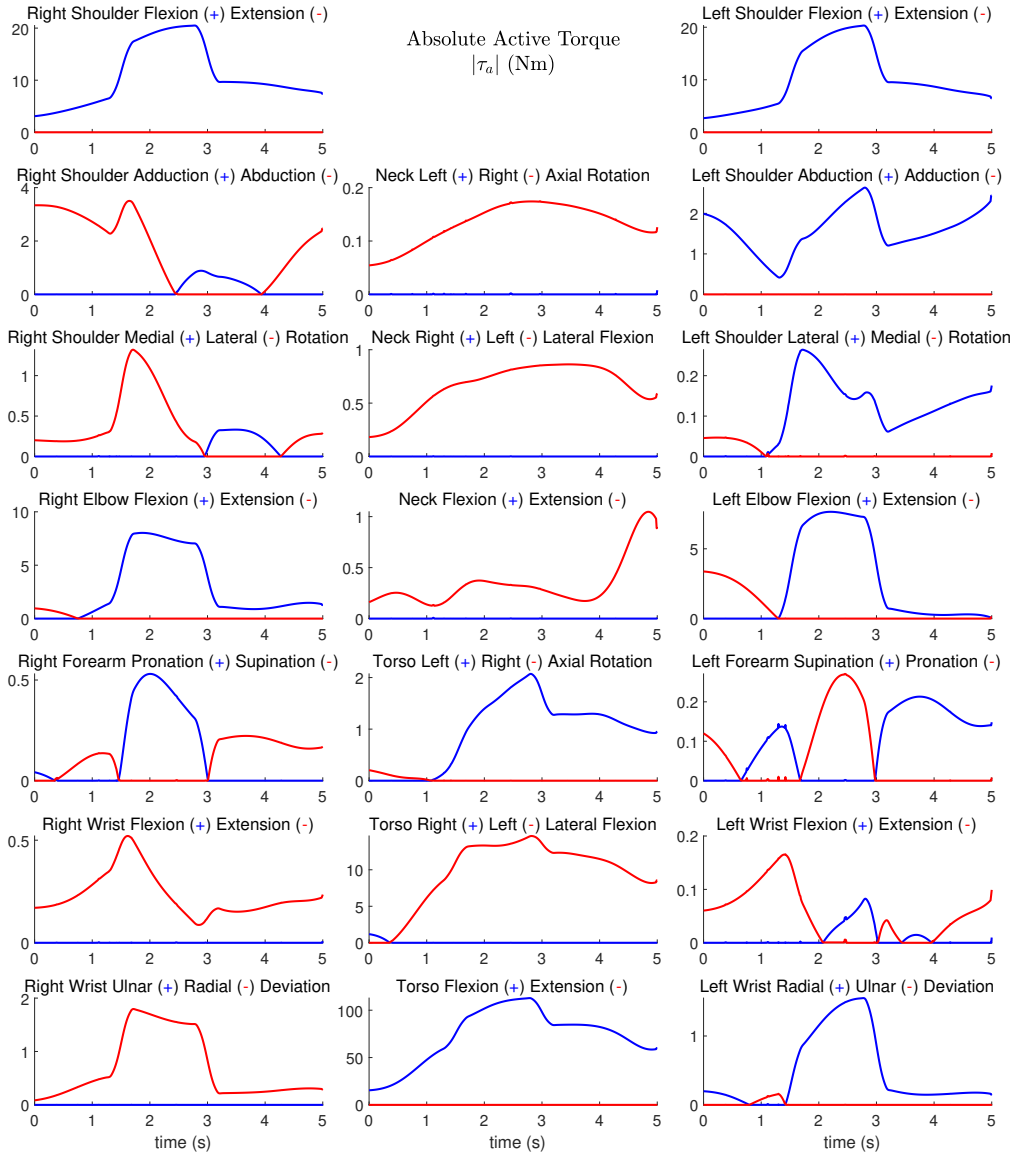


Figure 3.10: The plot of absolute active torque of 20 joints and 40 MTGs optimized for minimum active torque in the pick-and-place movement.

signals, from Equation (3.4) are shown in Figure 3.11 and Figure 3.12.

According to the results in Figure 3.11, SFEs for both arms activate with 50% of the signal, and the TFE activates with roughly 100% when the object is lifted from the furthest target. This activation pattern means that lifting in the pose shown in Figure 3.8 has an undesirable impact on the torso and shoulder joint.

3.5.2 Forward dynamic simulation

The human CNS, which includes the brain and the spinal cord, controls the human body's mobility [166, 326]. Despite uncertain/unknown trajectories and intricate muscle dynamics, the CNS simultaneously maintains the kinetics and kinematics [166, 327]. Experimental investigations of human mobility suggest that the CNS optimally coordinates body movements [271, 328]. According to many studies, the CNS regulates the human by minimizing a cost function that may include jerk [329], torque [330], muscle activation [331], metabolic energy [97], and muscle fatigue [332].

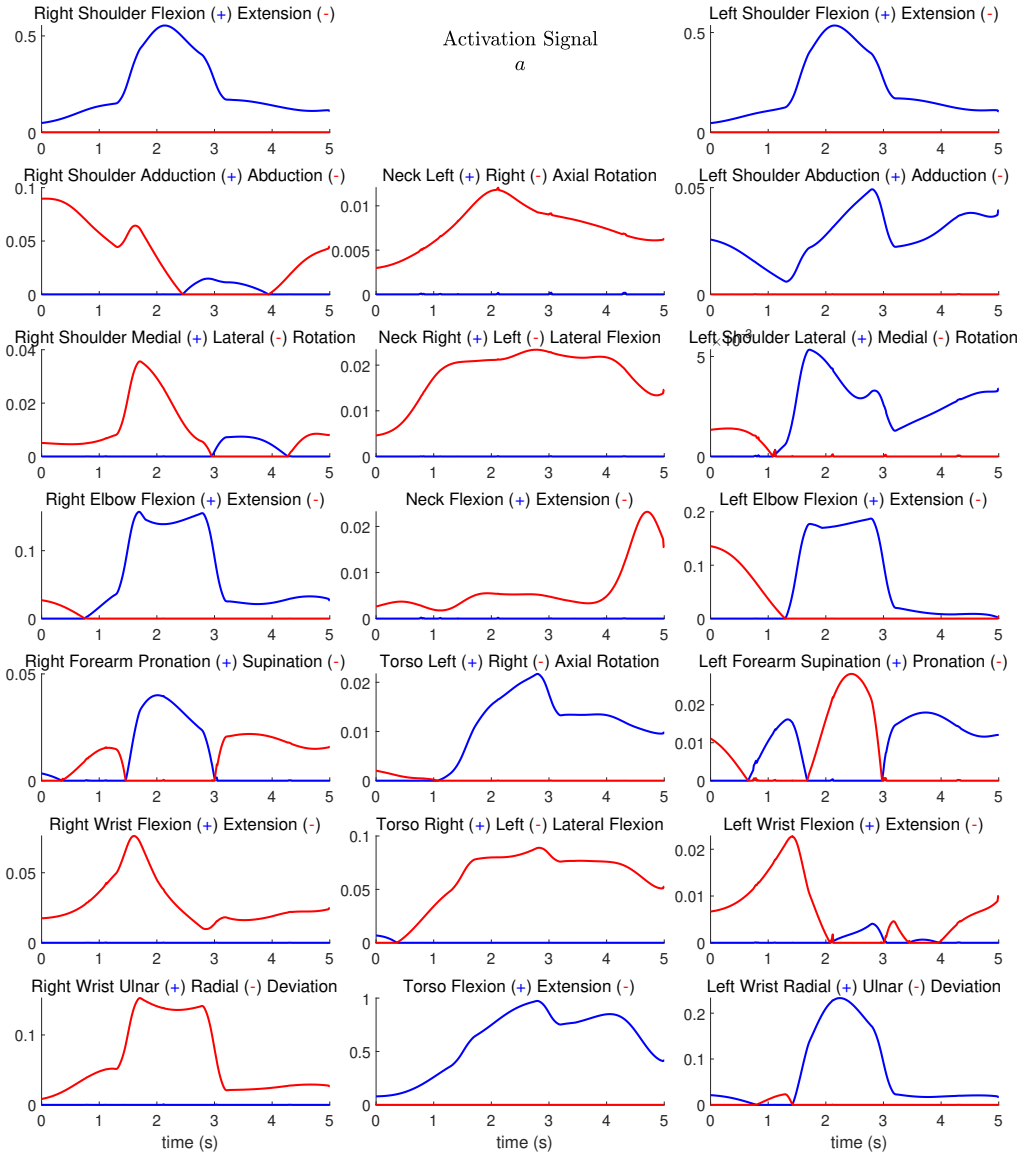


Figure 3.11: The activation signal of the 20 joints and 40 MTGs for activation of the MTGs driving the optimized active torque in the pick-and-place movement.

The CNS coordinates human movements with a complex signal that may be computed using (I) motion prediction and (II) corrective command [166]. An internal model or representation of the MSK system is used to compute motion prediction or feed-forward control. In addition, the sensory organs calculate a correction instruction or feedback control to remedy any mistakes caused by model uncertainty, external disruption, or an unfamiliar environment. NMPC emulate this multi-purpose control with a limited horizon (feed-forward and feedback control) [166] (Figure 3.13). The NMPC employs an IM that is learned over time [272, 333] to simulate human dynamics to forecast optimal motion (feed-forward) and correct prediction mistakes (feedback) [166]. It evaluates the ideal motion using a cost function. This cost function comprises joint angle errors ($\boldsymbol{\theta} - \boldsymbol{\theta}_{\mathbf{d}}$), joint angular velocity errors ($\boldsymbol{\omega} - \boldsymbol{\omega}_{\mathbf{d}}$), model inputs and their derivatives, as well as weights ($\mathbf{W}$). For the activation-signal-driven model (input $\boldsymbol{a}$) , this cost function can be formulated as Equation (3.15).

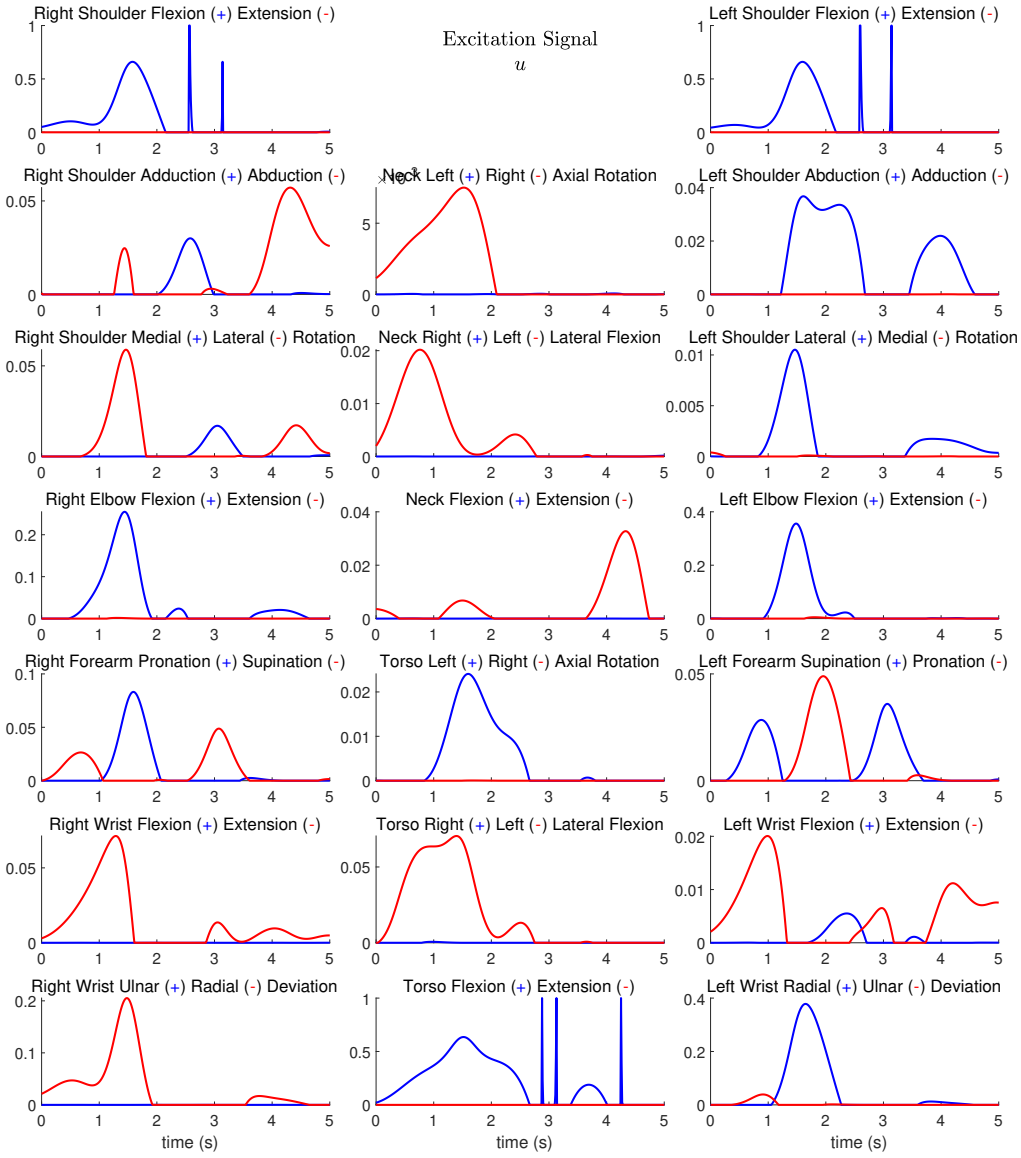


Figure 3.12: The excitation signal of 40 joint MTGs for the pick-and-place movement.

$$J = \int_0^{t_f} \left[\mathbf{W}_1^T \left(\boldsymbol{\theta} - \boldsymbol{\theta}_\mathbf{d} \right)^2 + \mathbf{W}_2^T \left(\boldsymbol{\omega} - \boldsymbol{\omega}_\mathbf{d} \right)^2 + \mathbf{W}_3^T \left(\boldsymbol{a} \right)^2 + \mathbf{W}_4^T \left(\dot{\boldsymbol{a}} \right)^2 \right] dt \tag{3.15}$$

$$\begin{aligned} \boldsymbol{\theta}_{min} \leq \boldsymbol{\theta} \leq \boldsymbol{\theta}_{max} \\ \boldsymbol{\omega}_{min} \leq \boldsymbol{\omega} \leq \boldsymbol{\omega}_{max} \\ \mathbf{0} \leq \boldsymbol{a} \leq \mathbf{1} \\ \dot{\boldsymbol{a}}_{min} \leq \dot{\boldsymbol{a}} \leq \dot{\boldsymbol{a}}_{max} \end{aligned} \tag{3.16}$$

Under the influence of system dynamics provided in section 3.4 and constraints in Equation (3.16), the NMPC controller optimizes the cost function (3.15).

The MTG excitations or activations in the musculoskeletal model may be generated by: (1) a forward static optimization (FSO), in which a musculoskeletal system behavior is approximated by static reduction of a physiological cost functional, or (2) by a forward dynamic optimization (FDO), in which a musculoskeletal system behavior is modeled through dynamic minimization of a physiological cost functional, or by (3) experimental

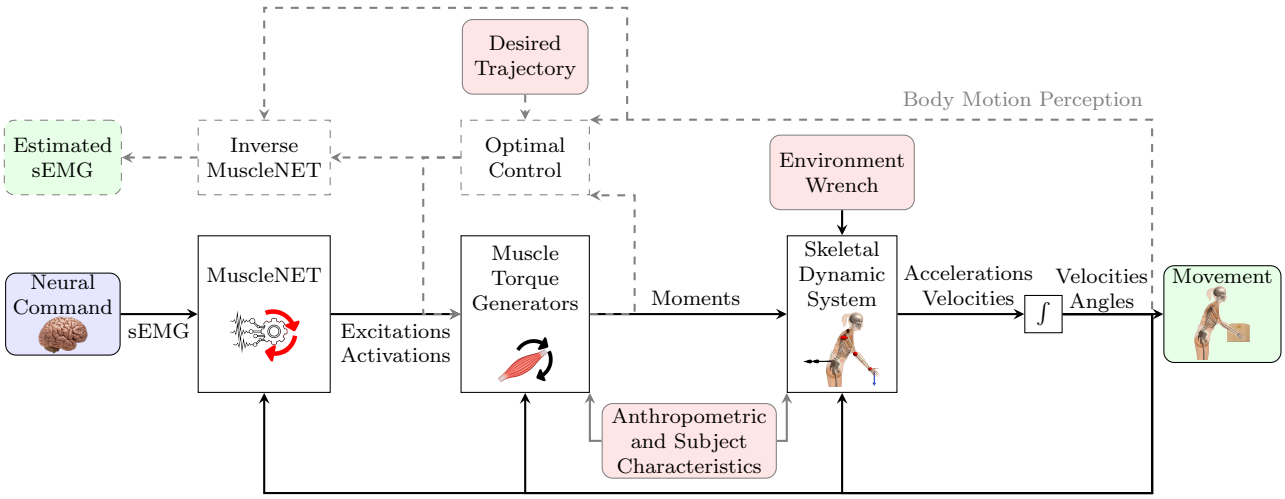


Figure 3.13: Elements of a typical musculoskeletal simulation in MapleSim Biomechanics. Movement occurs from a complex composition of the neural, muscular, and skeletal systems. MapleSim Biomechanics comprises computational models of these systems, allowing for human movement prediction and analysis. Neural commands to MTGs, in the form of excitations, can be estimated from optimal controller models or experimental data (e.g., sEMG) with MuscleNET (a machine learning framework that maps EMG excitations to kinematic/dynamic variables) [38]. MapleSim Biomechanics's MTGs models, which convert excitations into biomechanical joint torque, contain the passive torque, active torque angle scaling, and active torque angular velocity scaling features of the biomechanical joint. MapleSim Biomechanics MTG are based on experimental data. MapleSim Biomechanics's peak joint strength depends on age, height, body mass, dominant side, and athletic activity.

sEMG data from a subject. The sEMG data may be mapped to MTG inputs with a machine learning model [38].

3.6 Model validation

We compare the measured moments at the joints of interest to the model's estimated ones to assess the model's performance for a variety of different predicted movements. The experimental study uses dynamotor measurement devices that constrain joint motion to measure joint moments. In these tests, subjects are directed to flex or stretch maximally against the device's resistance, and the resulting isometric and isokinetic torques are recorded. The maximum isometric test is done with zero joint velocity. In contrast, the isokinetic test is done as the dynamotor measurement system controls the joint velocity. The isometric and the isokinetic tests are also simulated using the MapleSim Biomechanics model and forward dynamics. We consider maximum activation for MTG in the model when calculating maximum isometric moment-generating capability.

According to section 3.3.2, the joint peak isometric torque is a function of age [70, 72, 73], height [70, 74, 76], body mass [70, 74, 75], as well as, sex, dominant side [77, 79], and athletic activity [277].

3.6.1 Isometric test

The results of many published experiments are represented in the maximum isometric joint moments computed using the MapleSim Biomechanics model. Since the published experiments used for validation all had a limited number of subjects (less than 10), we focus on the trend and pattern of maximum isometric joint torques in our comparison with the experimental literature. Modeling and experimental results for the torso and neck are compared in Figure 3.14, for the shoulder in Figure 3.15, and for the elbow, forearm, and wrist in Figure 3.16.

The maximum isometric torques generated by torso flexors and extensors as a function of flexion angle successfully follow the trend of the data reported by Keller and Roy [284] and Kumar [67] in the ROM of torso flexion/extension (Figure 3.14(a)). Surprisingly, the data reported by Kumar [67] covers more than the

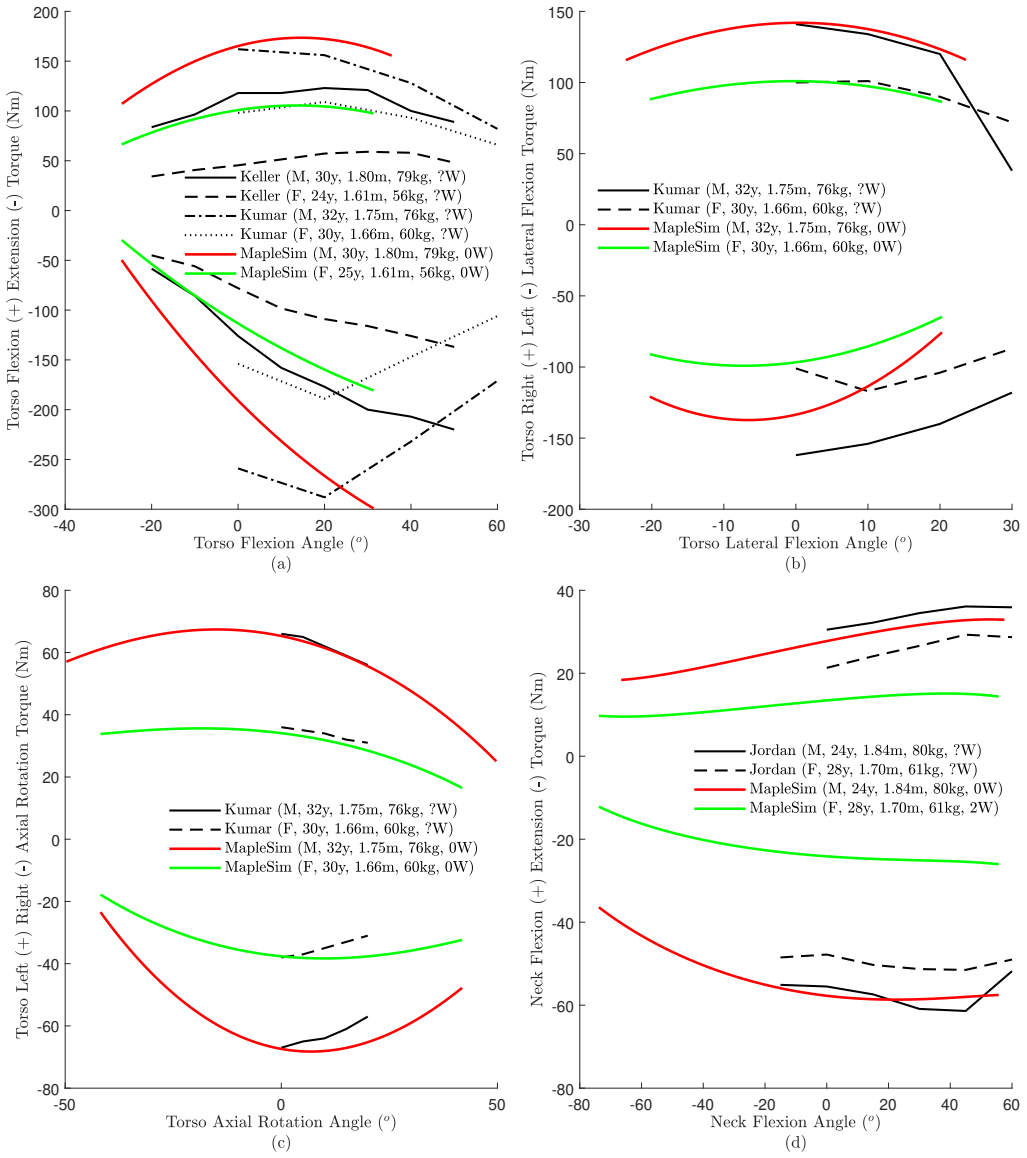


Figure 3.14: Maximum isometric torque generated by (a) torso flexors (+) and extensors (-) versus torso flexion, (b) torso right (+) and left (-) lateral flexors versus torso lateral flexion, (c) torso left (+) and right (-) axial rotator versus torso axial rotation, and (d) neck flexors (+) and extensors (-) versus neck flexion. The model estimates are highlighted in thick solid lines and are compared to experimental data from Keller and Roy [284], Kumar [67], and Jordan et al. [74]. M: Male, F: Female, Height is in meter (m), Weight is in kilograms (kg), number of session workout per week showed with W.

average possible ROM that is measured by most researchers [282, 283, 285] for the torso joint. The maximum isometric moment produced by torso right and left flexors in Figure 3.14(b) also follows the pattern reported in the study of Kumar [67] for the ROM of torso lateral flexion [282, 286]. As shown in Figure 3.14(b), the torso left axial rotator isometric torque matches the data by Kumar [67] well, although Kumar [67]'s experiments did not cover the full ROM. The torso's right axial rotator's maximum torque decreases when the torso axial angle increases, which is similar to the experimental data of Kumar [67] (Figure 3.14(c)). Similar trends between the experimental data of Jordan et al. [74] and the estimated isometric torque of neck flexion and extension by the MapleSim Biomechanics model are shown in Figure 3.14(d). The neck peak flexor torque is generated at approximately 60^o neck flexion angle by MapleSim Biomechanics, similar to the experimental data of Jordan et al. [74]. Other-DoF of neck has no experimental isometric torque versus neck joint angle data for validation.

The model's shoulder flexion moment-generating capacity diminishes at the same rate as in the literature [77, 292, 334] as shoulder flexion angle increases. The model's extension moment-generating strength follows the same pattern as the data in the literature [292], and our dynamometer experiment as mentioned in section 3.3.3, as shoulder flexion angle increases (Figure 3.15(a)). The model reproduces the empirical trend of literature data [77, 292] in shoulder vertical abduction and adduction strength as the shoulder abduction angle increases (Figure 3.15(b)). The vertical adduction torque peaks at approximately -60^o rotation similar to the experimental data of Garner and Pandy [292]. As the shoulder horizontal abduction angle changes, the model reproduces the experimental trend in shoulder horizontal abduction and adduction strength (Figure 3.15(c)). Horizontal abduction torque peaks at approximately -50^o rotation angle like the experimental data. The model's maximum isometric external/internal rotation moment follows the same pattern as that seen in the literature [77, 276, 291, 292] (Figure 3.15(d)).

As shown in Figure 3.16(a), the model's elbow flexion and extension moment-generating capacity follows the same trend as the research in [292, 335, 336], as shoulder flexion angle changes. In addition, the maximum elbow flexion and extension isometric torque occur at about 70^o and 80^o, respectively, which are similar to the literature [292, 335]. The forearm pronation isometric torque estimation has the same trend as Hutchins's experimental data [294] and has a slight difference in fully supinated arm with other studies [292, 337] (Figure 3.16(b)). The forearm supination torque profile has a similar trend and value as most of the reported values [292, 294, 337]. The estimated wrist flexion is similar to Garner and Pandy [292] and Delp et al. [297] (Figure 3.16(c)). The estimated wrist extension by MapleSim Biomechanics is similar to other research, and the joint angle in which the estimated maximum torque occurs is close to the values reported in the literature [292, 294, 297]. The wrist ulnar and radial deviation torque is similar to some sources [294, 297] and has a different trend with only Garner and Pandy [292] (Figure 3.16(d)).

3.6.2 Isokinetic test

Maximum isokinetic joint moments computed using the MapleSim Biomechanics model is compared with experimental data reported in the literature (shown in Figure 3.17). The experimental data used in most references are from elite athletes [338, 339] that generate torque even for higher angular velocities than the normal subject reported in Table 3.4 are capable. In addition to not comparing the amplitude of the isokinetic torque, the maximum possible torque for the highest reported angular velocity is not compared in this research due to the difference in subject types.

As shown in Figure 3.17, the experimental torques decrease as the angular velocity increases for the concentric isokinetic test of joints, similar to the estimation of MapleSim Biomechanics [304, 338–341]. For the eccentric isokinetic test, the experimental torques increases as the angular velocity increases [339] similar to the MapleSim Biomechanics simulation. In total, the maximum isokinetic torque of joints estimated by the MapleSim Biomechanics model follows the same pattern of the experimental data reported in the literature [304, 338–341].

3.6.3 Shoulder workspace volume

The shoulder workspace has not been evaluated in most biomechanical modeling studies. The ROM for the right shoulder joint of a male is shown in Figure 3.18. For calculating the ROM, the first and second shoulder Euler angles have been varied from $-\pi$ to π. Then, the passive torque of the shoulder joint has been calculated using the 2 MTG models of SFE and SAA. As described in section 3.3.6, for the second-DoF of the shoulder (SAA), the extrapolation of 2 MTG models of the SVAA and SHAA has been used. The ROM that has less passive torque than 5 Nm has been visualized in Figure 3.18. As described in section 3.3.1, the passive torque of the joint outside of the ROM increases exponentially. Thus, the 3D workspace in Figure 3.18(d) shows the reachable surface of the upperarm, which is quite similar to the 3D experimental data in [342, 343] and 2D experimental data in the literature [282, 290].

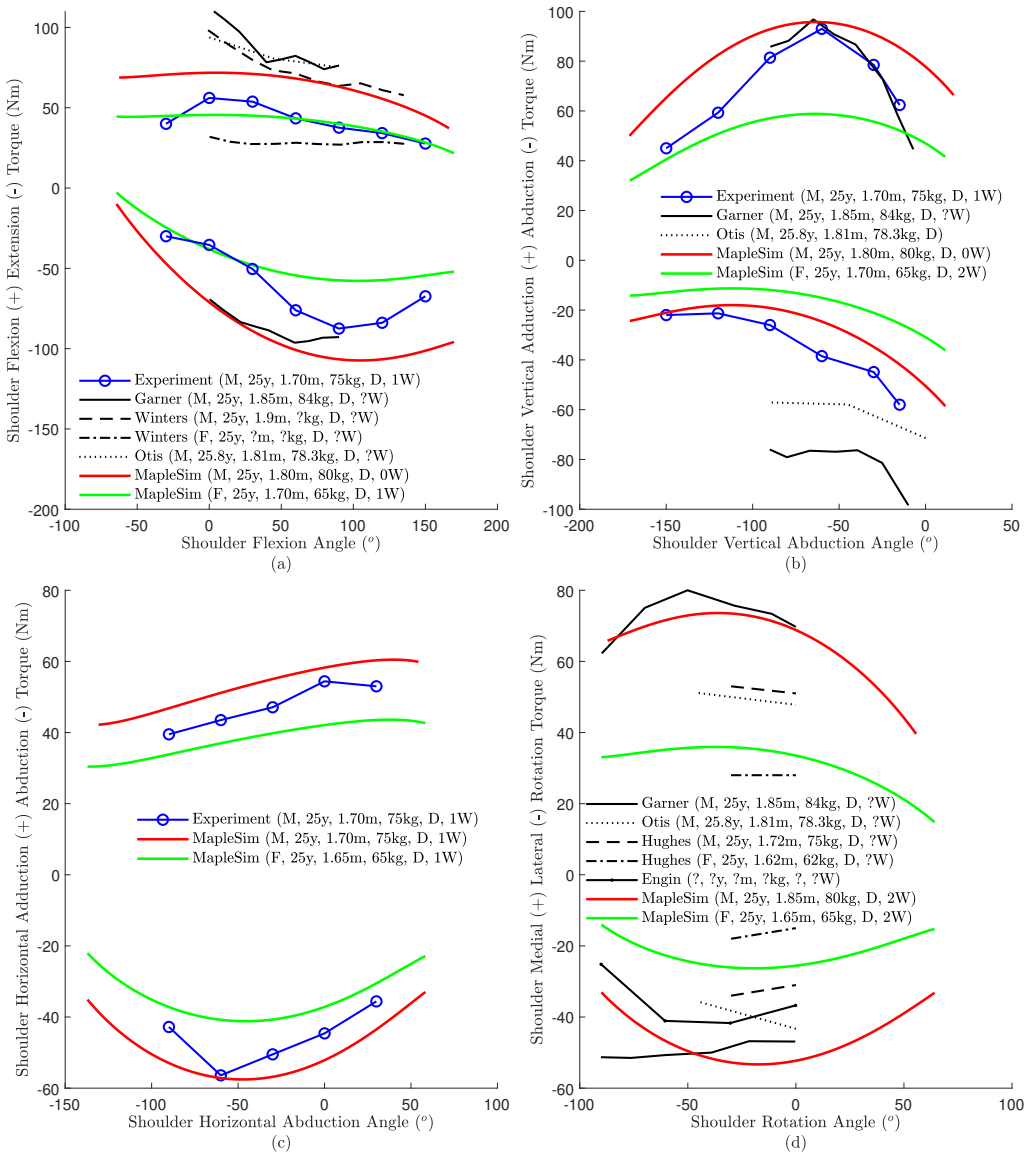


Figure 3.15: Maximum isometric torque generated by (a) shoulder flexors (+) and extensors (-) versus shoulder flexion rotation, (b) shoulder adductors (+) and abductors (-) versus shoulder abduction rotation, (c) shoulder horizontal adductors (+) and abductors (-) versus shoulder horizontal abduction rotation, and (d) shoulder medial (+) and lateral (-) rotators versus shoulder rotation. The model estimates are highlighted in thick solid lines and are compared to experimental data from our test (mentioned in section 3.3.3), Garner and Pandy [292], Winters and Kleweno [334], Otis et al. [77], Hughes et al. [276], and Engin and Kaleps [291]. SFE, SVAA, and SHAA were determined with a straight arm in the neutral position. SMLR torque was determined for the elbow in 60^o flexion, and 60^o abduction. M: Male, F: Female, Height is in meter (m), Weight is in kilograms (kg), number of session workout per week showed with W.

3.7 Discussion

This chapter presents a musculoskeletal model for torso, neck, shoulder, elbow, and wrist joints in the human upper body. The model is based on average experimental biomechanical data for segment properties (dimensions and inertia) as well as joint properties (maximum joint torque, maximum ROM, maximum angular velocity, and maximum isometric torque-angle profile). It can be adjusted based on subject body characteristics: sex, age, body mass, height, dominant side, and physical activity.

One of the contributions of this work is that it provides enough information to build a mathematical model.

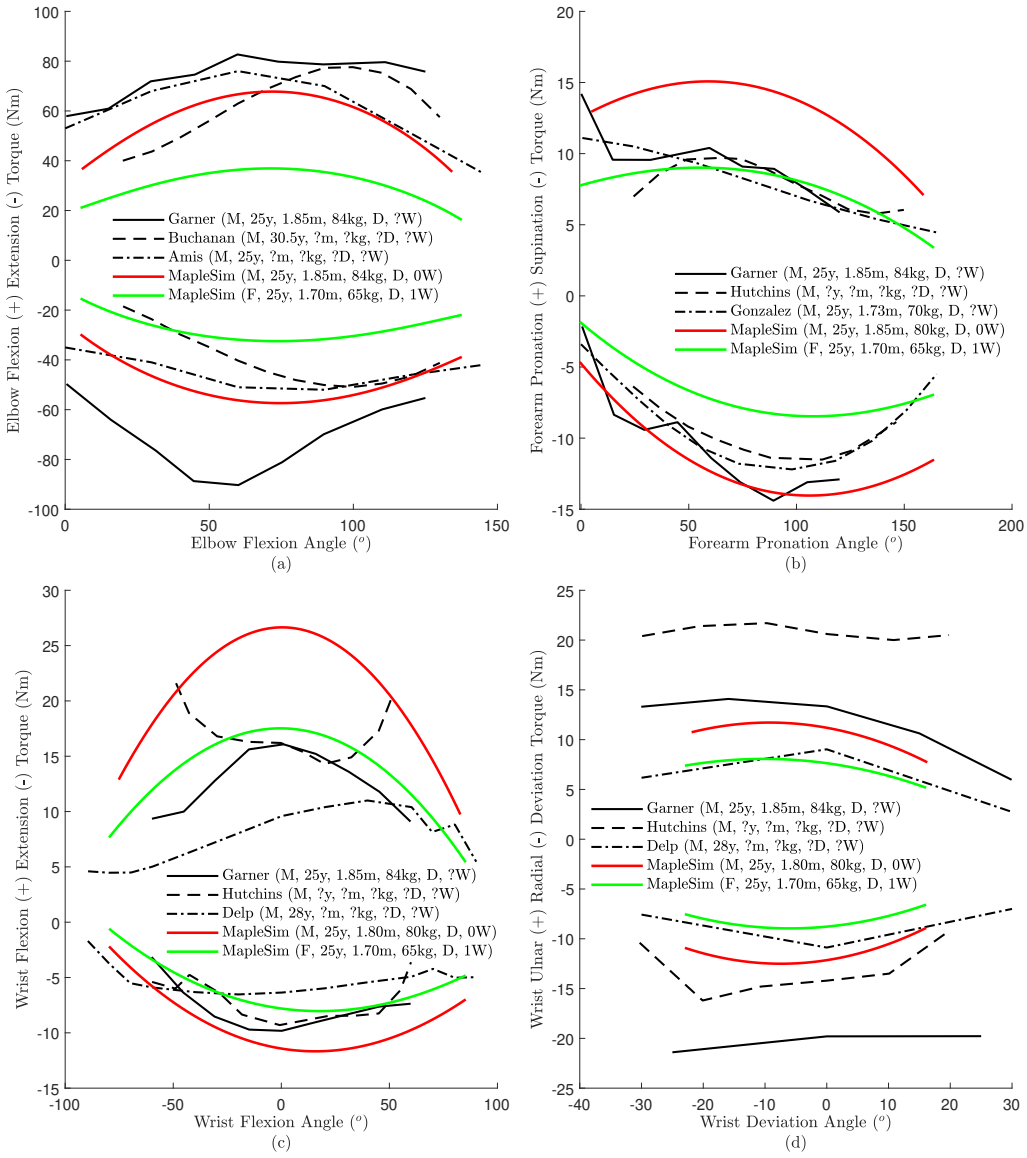


Figure 3.16: Maximum isometric torque generated by (a) elbow flexors (+) and extensors (-) versus elbow flexion angle, (b) forearm pronators (+) and supinators (-) versus pronation angle, (c) wrist flexors (+) and extensors (-) versus wrist flexion angle, and (d) wrist ulnar (+) and radial (-) deviators versus deviation angle. The model estimates are highlighted in thick solid lines and are compared to experimental data from Garner and Pandy [292], Buchanan et al. [336], Amis et al. [335], Gonzalez [337], and Delp et al. [297]. EFE, FPS, WFE, and WRUD were determined with the shoulder in 60^o flexion, and 30^o abduction and all other DoF in neutral position. M: Male, F: Female, Height is in meter (m), Weight is in kilograms (kg), number of session workout per week showed with W.

The data in Table 3.3, Table 3.4 and Table 3.5, as well as the presented equations, are sufficient to construct a dynamic model. Future users can focus on the targeted joint and ignore the non-targeted joint data.

To date, our model is more general and more complete in terms of being scalable with more anthropocentric variables than other presented MSK models; it is also directly verified and validated. The data for this general model is acquired from literature with multiple subjects in contrast to the subject-specific models in [63, 64, 292]. In contrast with most musculoskeletal models that are not validated for all ROMs [67, 77, 276, 291, 292, 334], our model was constructed and verified within the maximum ROM from the literature [282, 287, 290, 293, 296]. The average maximum angular velocity from the literature [283, 285, 286, 288, 289, 295] is used to limit the maximum isokinetic joint torque in concentric motion and verified against the experimental literature [304, 338–341]. The maximum joint torque is adopted from the literature for females and males [67, 70, 78, 276, 303–307].

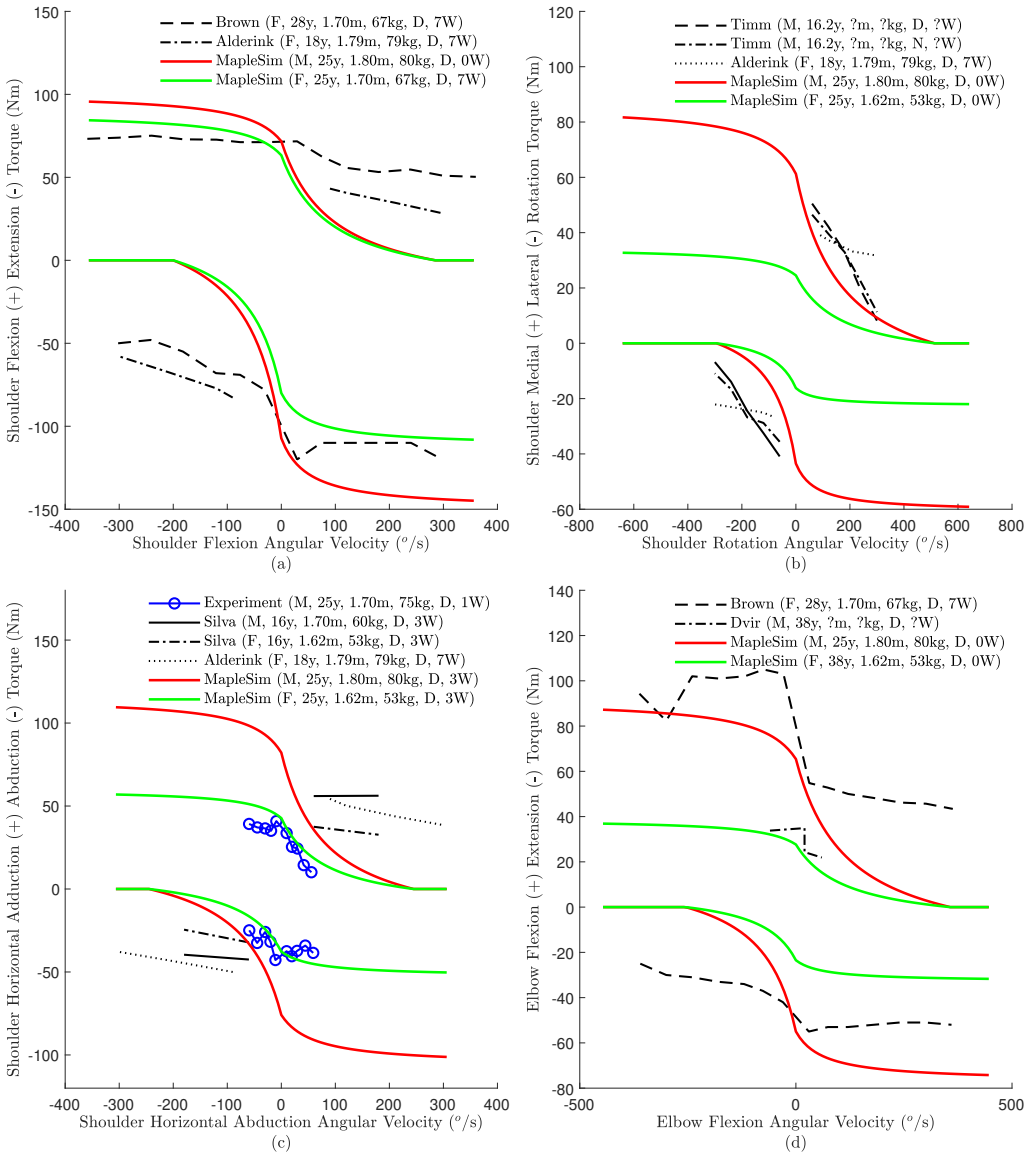


Figure 3.17: Maximum isokinetic torque generated by (a) shoulder flexors (+) and extensors (-) versus shoulder flexion angular velocity, (b) shoulder medial (+) and lateral (-) rotators versus shoulder angular velocity, (c) shoulder horizontal adductors (+) and abductors (-) versus shoulder horizontal abduction angular velocity, (d) elbow flexors (+) and extensors (-) versus elbow flexion angular velocity. The model estimates are highlighted in thick solid lines and are compared to experimental data from our test (mentioned in section 3.3.3), Brown [339], Alderink and Kuch [338], Timm [341], Silva et al. [304], Dvir [340]. M: Male, F: Female, Height is in meter (m), Weight is in kilograms (kg), number of session workout per week showed with W.

The isometric joint torque-angle pattern is modeled from the literature data [67, 74, 284, 291, 292, 294, 297] or our own experimental measurement. The isokinetic torque-angular-speed profile is modeled using the literature data of [56, 87, 311, 312, 315] in addition to the mentioned average maximum angular velocity. Since the shoulder abduction/adduction has different parameters in different orientations, we have created a novel 2D MTG model, in contrast to previous musculoskeletal models of the upper limb [77, 292, 334], and verified it in two vertical and horizontal planes. Verification of the isokinetic test of each joint, which is essential for simulation of high-speed motion (mostly sports analysis), is missing in previous musculoskeletal models of the upper limb [292, 297, 335–337]. The workspaces of the shoulder flexion/extension and abduction/adduction are evaluated and verified with the literature [282, 290, 342, 343], which is essential for a complex motion of the shoulder.

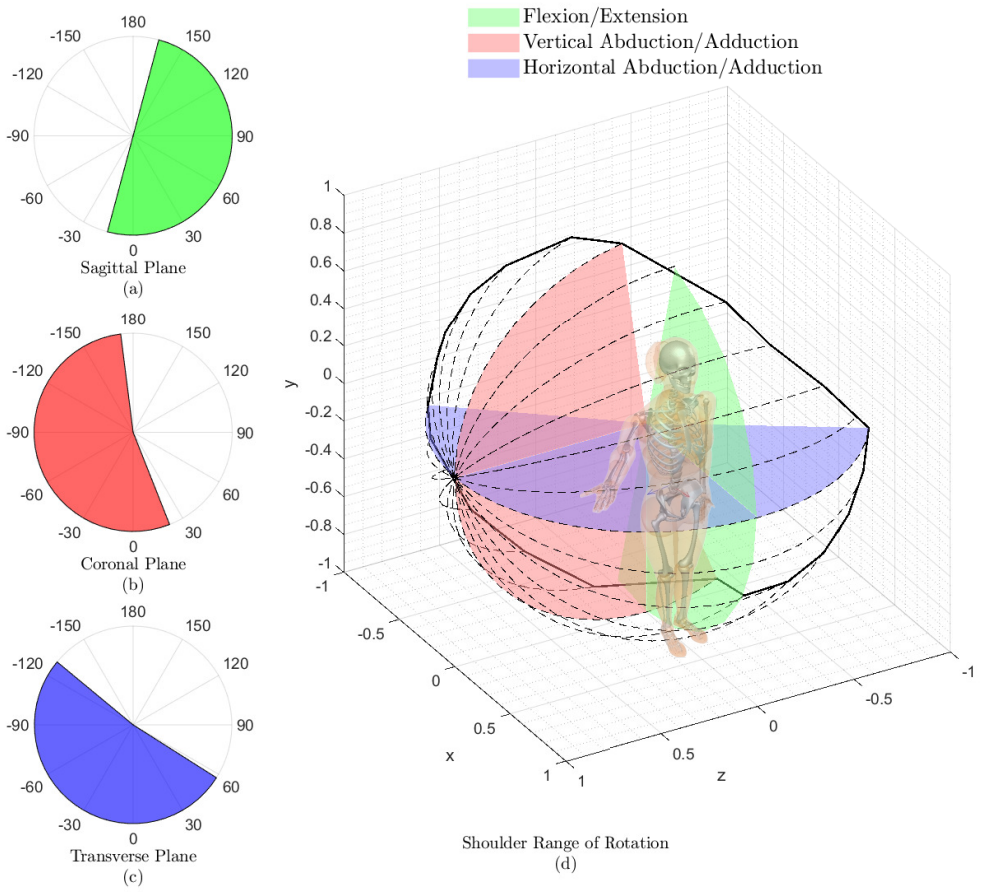


Figure 3.18: The visualization of a male right shoulder ROM for (a) Flexion/Extension from the side view, (b) vertical abduction/adduction from the front view, (c) horizontal abduction/adduction from the top view, and (d) 3D reachable surface of a unit (1-meter length) upper-arm.

The skeletal dynamic model can be adjusted for sex, body mass, and body height since they impact the anatomical locations, the COMs, and inertia parameters [68, 69]. The biomechanical joint torque is also a function of subject characteristics. The changes of peak torque relevant to age [70, 72, 73], height [70, 74, 76], body mass [70, 74, 75], dominant side [77, 79], and amount of activity [277] are modeled using the literature data. The maximum joint ROM can change according to sex differences in passive torque function. In addition, age impacts the excitation-to-activation delay, according to [320].

3.7.1 Applications

The provided biomechanical model can be used in various applications broadly categorized as: simulation of human movements (joint torques, ROM), biomechatronics, ergonomics and usability studies, predictive simulation of activities of daily living (reaching, walking, sitting, etc), clinical practice, human-device interactions, sports engineering, rapid and safe development/control of medical and assistive devices. Some of the mentioned applications are discussed in detail below.

Subject-specific torque-driven computer models of human movement have provided insight into optimal athletic technique [344]. Computer simulation is a valuable tool for investigating the athletic dynamics without repeating actual experiments [84]. These predictive simulation trials can provide insight into issues like equipment design [56], injury prevention [309], and optimum motor control methods [166], all of which increase sports performance [84], by providing researchers with an interface through which significant circumstances and parameters can be modified. Multibody computer simulation has been used for various activities (golf, track

cycling, etc.) to study athletic performance [84].

Predictive multibody dynamic simulations are typically useful for rapid and safe development of new medical and assistive devices. Recent applications have included orthoses, prostheses, and wearable exoskeleton robotics [52]. Models that include human interaction with biomechatronic devices (exoskeletons, prostheses, rehabilitation systems, etc.) are required for their design and control [26, 27, 345]; these models are utilized within computer simulations [19, 346]. In addition, the human biomechanical model can be used for model-based control of robotic devices [19, 38].

3.7.2 Limitations

Not providing information about muscle tension is the main weakness of the MTG models. Anatomically detailed muscle models are needed to evaluate joint reaction force/torque as well as the muscle tensions. Our model uses MTGs instead of an anatomical-detailed model because of their ability to mimic the muscle group torque impact on the targeted joint. Note that without direct verification of parameters for the anatomical-detailed model, the estimated muscle tension cannot be trusted. Our model has a wide range of applications in sports and biomechatronics analysis, but is limited for from muscle injury analysis.

Some parameters are not provided in the literature for all the joints; thus we considered the same constant value for all the them. In the future, measuring this information with a high number of subjects can increase the accuracy of our model, for example, the ratio of the maximum eccentric isokinetic torque (τ_m) over the maximum isometric torque (τ_0) and the impact of workout on the maximum joint angular velocity.

Using this model in EMG-driven simulation suffers from a gap of EMG to MTG excitation mapping, as shown in Figure 3.13. Having experimental EMG data for most upper body muscles can create new opportunities for developing a machine learning mapping according to the MuscleNET model [38, 347].

The joint torque is modeled with a monoarticular assumption [298] of muscles, which is not always accurate. Experimental investigation of the biarticular muscles and the impact on the MTGs model can increase the accuracy of these models [88, 344].

3.8 Concluding remarks

In this research, we created an adjustable MSK upper body model. The model has 20 DoF driven by 40 MTGs and is constructed from experimental literature data. The model is adjustable for different anthropometric measurements and subject body characteristics: sex, age, body mass, height, dominant side, and physical activity. The maximal isometric and isokinetic torque and workspace were compared with previously published research for model validation. This model contributes to sports analysis simulation, ergonomics and usability studies, predictive simulation of activities of daily living, human-device interaction evaluation, design, development, and control of medical, assistive devices, exoskeletons, and prosthesis.

Chapter 4

Machine Learning Solution to Static Optimization and Inverse Muscle Modelling (Inverse MuscleNET)

InverseMuscleNET, a machine learning model, is proposed as an alternative to static optimization for resolving the redundancy issue in inverse muscle models. A RNN was optimally configured, trained, and tested to estimate the pattern of muscle activation signals. Five biomechanical variables (joint angle, velocity, acceleration, torque, and activation signal) were used as inputs to the RNN. A set of sEMG signals, experimentally measured around the SFE, were used to train and validate the RNN model. The obtained machine learning model yields a normalized regression in the range of $88 - 91\%$ between experimental data and estimated muscle activation. A sequential backward elimination algorithm was used as a sensitivity analysis to discover the less dominant inputs. The order of most essential signals to least dominant ones was as follows: joint angle, activation signal, joint torque, joint velocity, and joint acceleration. The RNN model required 0.06 seconds of the previous biomechanical input signals and 0.01 seconds of the predicted feedback sEMG signals, demonstrating the dynamic temporal relationships of the muscle activation profiles. The proposed approach permits a fast and direct estimation ability instead of iterative solutions for the inverse muscle model. It raises the possibility of integrating such a model in a real-time device for functional rehabilitation and sports evaluation devices with real-time estimation and tracking. This method provides clinicians with a means of estimating sEMG activity without an invasive electrode setup.

In this chapter, firstly, data preparation is introduced. Secondly, the machine learning model, configuration optimization, and evaluation algorithms are described. Finally, results are discussed and evaluated.

4.1 Data collection and preparation

This section presents the data collection methods [348, 349] and data preparation steps. The processed data was used for training the machine learning models in section 4.2.

4.1.1 Subjects

Nine female and eight male healthy right-handed individuals (23 ± 4 years; 72.25 ± 29.85 kg mass; 1.66 ± 0.16 m height) free of upper extremity injury gave informed consent and performed proscribed object manipulation tasks. The experimental protocol was in accordance with the Declaration of Helsinki. The university office of research ethics approved the data collection study (ORE #: 21246).

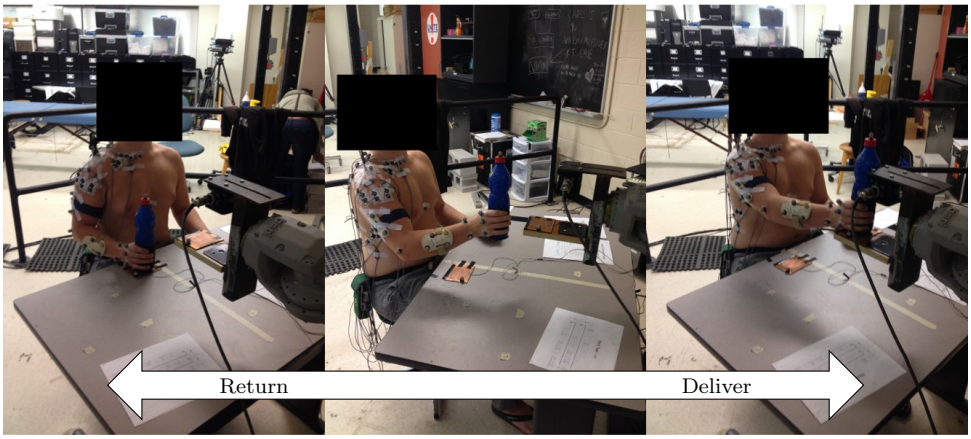


Figure 4.1: A depiction of the data collection process, sEMG electrodes and the retroreflective position marker placements by Whittaker et al. [349], where participants delivered and returned a weighted object between target locations in the sagittal plane.

4.1.2 Instrumentation

The 3D upper-limb motion and muscle sEMG signals were recorded at the same time. SEMG signals were measured from 11 sites over muscles of the right upper-limb, similar to those suggested by Hislop et al. [350]: supraspinatus (SUPR), infraspinatus (INFR), pectoralis major (PECC), latissimus dorsi (LATS), serratus anterior (SERR), MDEL, ADEL, PDEL, UTRA, MTRA, and lower trapezius (LTRA).

A ground electrode was positioned over the clavicle. Skin sites were shaved and swabbed with an isopropyl alcohol wipe prior to electrode placement. Noraxon Bipolar Surface Ag-AgCl circular electrodes (Noraxon Inc, Arizona, USA) with a fixed 2.0 cm inter-electrode distance were used for placement, and a Noraxon T2000 telemetered system, TeleMyo, (Noraxon Inc, Arizona, USA) was used for collecting signals.

We placed the single bipolar electrode over each muscle. Then, we verified signal quality by observing the signal as participants performed isometric contractions in postures that elicit activity in the muscle of interest. Then, the raw sEMG signals were amplified (common-mode rejection ratio $\geq$ 100 dB at 60 Hz, input impedance 100 MΩ), sampled at a rate of 1500 Hz, and finally, digitalized (16-bit A/D card, maximum $\pm$10 V).

Eight Vicon MX20+ cameras (Vicon Motion Systems, Oxford, U.K.) were used to record the 3D position of retroreflective markers attached as rigid clusters to the subjects' forearm, upper arm, and scapula segments at 50 Hz.

4.1.3 Task protocol

The data was collected during pick and place tasks [348, 349]. Consequently, the subjects were requested to gently lift an object from the desk, place it on the upper target, and contrariwise, 15 times in 60 seconds following the beat of a metronome (Figure 4.1). Subjects were asked to rest/pause at the lower and upper target for about 2 seconds. Participants were asked to sit with their right arm at roughly 30° for SFE, 45° for SMLR, 120° for EFE, and the hand grasping the object (the positions of the lower and upper target were adjusted for each participant). The object was a bottle with a 14.9 ± 6.6 N weight (each subject used a bottle scaled to their individual shoulder elevation strength, while resulting in different external forces and therefore a more general muscle model [348]). During the mentioned task, the motion of the upper-limb and sEMG signals of muscles were recorded.

During data collection, subjects had no interaction with an assistive biomechatronic device (Figure 4.1) since the device may impact human behavior, decrease measurement accuracy, and the human may adapt to the assistive wrench [351]. However, the exoskeleton dynamic model can be added to the human skeletal system via simulation (section 7.2.3).

4.1.4 Kinematic data estimation

The following sections describe how the upper-limb position, external load, and sEMG signals are used in the data preparation. These steps prepare the RNN inputs (joint angles, velocity, acceleration, torque, and activation signal) and the RNN outputs (filtered sEMG signals).

Kinematic data analysis and estimation

Position data was digitally low-pass filtered using a 2^{nd} order Butterworth filter with a cut-off frequency of 10 Hz. Then, re-sampled with the higher sample rate of the sEMG data, 1500 Hz, using 1-D data cubic interpolation of the neighboring grid's position values with four-position trajectory instances.

The joint angles of the thoracohumeral and forearm (section 4.1.1) were estimated using the cluster markers' 3D position as well as an anatomical calibration matrix that interprets the correlation between the cluster and coordinate system of the body segment [105]. The absolute angles of the torso, the relative angles of shoulder and elbow joint were determined according to the ISB recommendations (a ZXY rotation sequence [278], as mentioned in section 3.2).

The lifting and lowering motions were recognized using the position and velocity of the object. Precisely, the object was estimated at one of the resting targets when its absolute velocity was less than 10 mm/s for more than 40 milliseconds [352].

The joints' velocity was obtained using the numerical first derivative of joint angles. Then, using a low-pass second-order Butterworth filter with a cut-off frequency of 20 Hz.

The acceleration of joints was estimated by the numerical first derivative of joint velocities and a low-pass filter using a 2^{nd} order Butterworth filter with a cut-off frequency of 30 Hz.

Computed kinetic process

For the given system motion (kinematics data that was estimated and discussed in section 4.1.4), the required forces and torques were calculated by Equations (3.10), which is called inverse dynamic analysis (section 3.5.1). Precisely, the joint torque τ_h is the torque necessary for applying the elevation motion in the sagittal plane (SFE). In addition, the required activation signal (σ) were computed by using Equations (3.12).

4.1.5 Filtered sEMG signals

In this research, the outputs of the machine learning model are sEMG signals. The following steps achieved filtering of the raw signals:

i A second-order band-pass digital Butterworth filter with a normalized cut-off frequency of $20 - 500$ Hz was used to remove heart rate artifacts and high-frequency content [353, 354] (signals below 20 Hz were cleaned to remove the motion artifacts, and sEMG signals above 500 Hz were cleaned as they had minor power spectral density [355]).

ii A second-order band-stop digital Butterworth filter (notch filter) with a normalized cut-off frequency of $55 - 65$ Hz was applied to eliminate the 60 Hz noise from the measurement unit.

iii The absolute value of the signal amplitude or rectifying the signal was used for making the signal positive.

iv A second-order low-pass digital Butterworth filter with a normalized cut-off frequency of 7 Hz [354] was used to smooth the signal as evaluated and analyzed by Nasr et al. [356].

v Normalization to the trial maximum signal amplitude was used.

vi For using control-orinted output signal, the sampling rate of the sEMG, kinematic and kinetic signals was decreased to 50 Hz, thus decreasing the computational cost of the model.

vii A re-sampling function (with the 1-D data cubic interpolation method) to a 50 Hz rate was used for the InverseMuscleNET model.

viii A curving function or an exponential function (to mimic the muscle activation signals in biomechanical muscle models [82]) as provided in Equation (4.1) was used for the InverseMuscleNET model.

$$\frac{e^{-sEMG}-1}{e^{-1}-1} \tag{4.1}$$

Extra filtering was not required since MuscleNET is a machine learning model for mapping signals (not a mathematical muscle model). A sample of the raw and filtered sEMG signals in the time and frequency domain is shown in Figure 4.2.

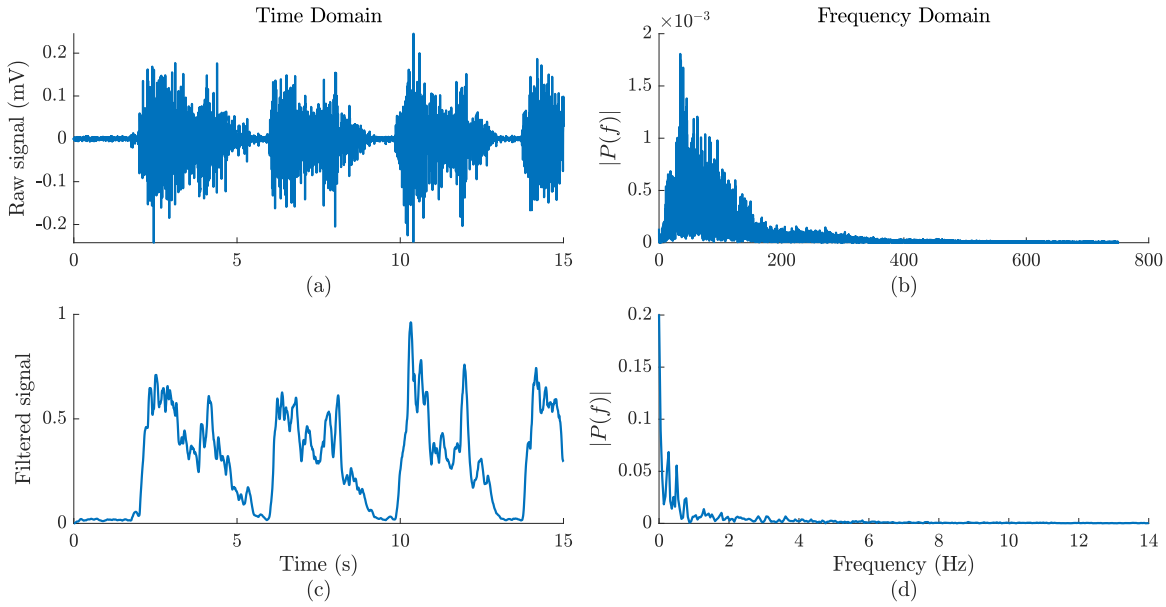


Figure 4.2: Sample of a raw in (a) and filtered in (b) sEMG signal. The frequency domain of relevant sEMG signal is shown in (b, d).

4.2 Recurrent neural network model

Machine learning is an adaptive system that trains by using interconnected nodes or neurons in a layered composition. RNNs can outperform forward networks for direct muscle modeling (sEMG to biomechanical signals) [38]. This machine learning configuration can learn the dynamic temporal relationships of a model due to its built-in recurrent signal (the feedback from the output) [357]. Essentially, an RNN is a deep machine learning network structure that uses information history (of inputs and outputs) to improve the network performance and estimate current outputs.

The nonlinear autoregressive with external input neural (NARX) network was selected as an RNN learning model. NARX is a recurrent network with feedback connections that are commonly used in time-series modeling. It predicts the next output signal value of a nonlinear dynamic system. A schematic of the NARX network is shown in Figure 4.3 within a two-hidden-layer feedforward network. This implementation allows for multidimensional inputs and outputs.

The primary equation of the RNN model is:

$$y(t) = f\left[y(t-1), y(t-2), \cdots, y(t-n_y), u(t), u(t-1), \cdots, u(t-n_u)\right] \tag{4.2}$$

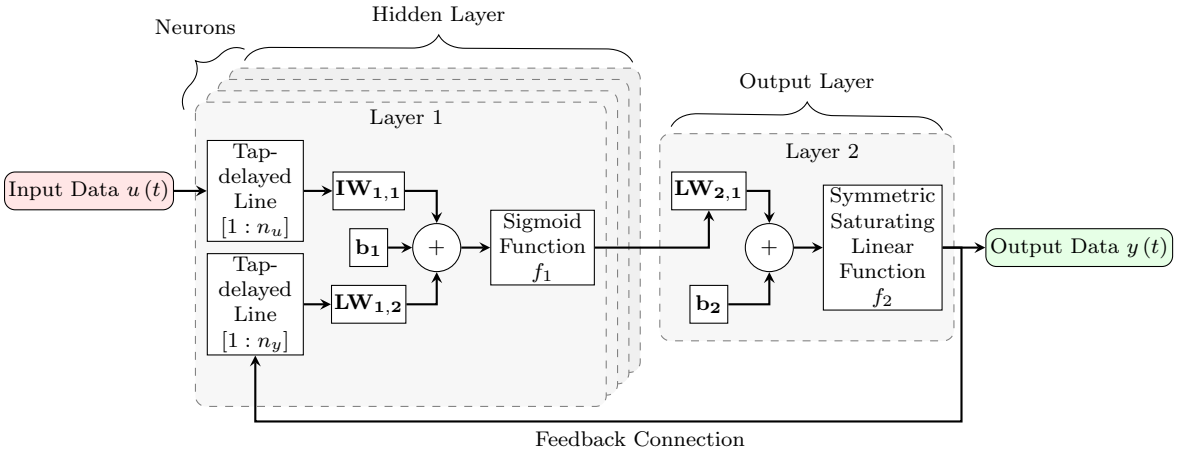


Figure 4.3: Schematic of a two-layer NARX network as an RNN learning model for sEMG signal prediction. n_u and n_y are the input signal former values and the output signal former values, respectively. **IW**, **LW**, and **b** are the input weight, layer weight, and bias of one neuron, respectively.

where

$y(t)$	the current output signal
$y(t-n_y)$	the former values of the output signal
$u(t-n_u)$	the former values of the input signal
n_u	the number of previous values of the input signal
n_y	the number of previous values of the output signal

As shown in Figure 4.3, the feedback structure permits the network to store past information in the hidden state $y(t-n_y)$ and operate on current sequences $y(t)$. Every single neuron has weights (W) that act as signal strength modifiers for that neuron. Every layer has a bias (b) that permits a shift in the activation function by adding a constant to the input. The weights and the bias are tuned throughout the learning process. The activation function in the network defines how the weighted sum of the inputs is converted into an output from the nodes in a network layer. We tested different activation functions and picked a sigmoid function for the hidden layer to provide the highest regression accuracy. Since the sEMG signals had been normalized during signal processing, we proposed to use a symmetric saturating linear transfer function for the output activation function. Accordingly, the estimated output signals are always between 0 and 1.

The four primary configuration parameters of the RNN model are the: (1) number of hidden layers, (2) number of nodes in each hidden layer, (3) number of input signal former values, and (4) number of output signal former values. Using many of these parameters would lead to a high variable model that would require a time-consuming training process on a large dataset. One of the goals of this chapter is to find the optimum configuration that can map kinematic and dynamic signals to sEMG signals with limited training time and a limited dataset.

4.2.1 Model inputs and outputs

The datasets consisted of the five input signals and one output channel for 17 subjects. The inputs of the RNN model are biomechanical signals such as (1) joint angle, (2) joint velocity, (3) joint acceleration, (4) joint torque, and (5) activation signal. The kinematic and dynamic signals are calculated in section 4.1.4. The outputs of the RNN model are the predicted sEMG signals. The filtered sEMG prepared in section 4.1.5 is used for the model output.

4.2.2 Model training and evaluation

There are two methods for assessing and evaluating a machine learning model: (I) subject-based estimation and (II) general-model performance. The subject-based estimation test is used as a full estimation model. It employs a data set from one subject that has not been utilized for training purposes. Thus, 14 subjects' data sets were used for training, two different subjects' data sets were used to validate the model, and one subject's data set was used in the estimation test. This method is used to optimize the RNN configuration in sections 4.4.1 and 4.4.4 and is used in Figure 4.7 and Figure 4.10. The general-model performance was used to evaluate the general estimation trend, general configuration, and signals' impact. The general-model utilized all subjects' data sets for training and validation purposes. This method is used for estimation analysis in section 4.4.2 and sensitivity analysis of input in section 4.4.3.

Primarily, out-of-sample (OOS) evaluation was used for the specific selection of training and validation sets. Specifically, CV is a method that comparatively divides the databank into two sets of training and validation. K–fold CV acquires k subdivisions having an equal amount of data from the databank. Each subdivision is used as a validation set and all other subdivisions as a training set for a regression process. So, the consequent regression accuracy performance is determined by the average k regression accuracy [358]. For 17 subjects' data sets, k was set to 17, and the training-validation process was repeated 17 times. The average, maximum, and minimum regression accuracy for each of the 17 training-validation processes were reported in Figure 4.6. The K–fold CV was used to assess sEMG signals estimation in sections 4.3.2 and 4.4.2.

To solve the nonlinear least-squares problem of the mapping model, the Levenberg–Marquardt backpropagation algorithm was used. The backpropagation algorithm calculates the Jacobian of the training performance concerning the weight and bias variables of the network. The initial adaptive mu, mu decrease factor, mu increase factor, and maximum mu value were 0.001, 0.1, 10, and 1e10, respectively. The training performance was evaluated using the MSE function (Equation (4.3)). Parallel processing with the MEX 4 workers was used to make the training process faster. The training time was limited to 1-hour, and the epoch (the number of passes of the entire training dataset to the model) was set to infinity. The linear regression of targets relative to outputs (R) (Equation (4.4)) was used as the mapping performance in this chapter.

$$MSE = \frac{1}{N}\sum_{i=1}^{M}(X_i - T_i)^2 \tag{4.3}$$

$$R = \left[\frac{D_{TRN}}{\sum D_i}\left(1-\sqrt{MSE_{TRN}}\right) + \frac{D_{VLD}}{\sum D_i}\left(1-\sqrt{MSE_{VLD}}\right) + \frac{D_{TST}}{\sum D_i}\left(1-\sqrt{MSE_{TST}}\right)\right] \times 100 \tag{4.4}$$

where

X	the output values of the trained machine learning model
T	the processed target values for training the machine learning model
N	the total number of datasets points
D_{TRN}	the training data volume
D_{VLD}	the validation data volume
D_{TST}	the testing data volume

4.3 Optimization and analysis method

4.3.1 Preliminary RNN configuration optimization

The estimation performance relies on the NARX network configuration, consisting of the number of hidden layers n_l, the nodes in each layer n_n, the input signal former values n_u, and the output signal former values n_y. These configuration parameters depend on the nature of the input signals (biomechanical signals) and the output signals (sEMG signals). This research tested different possible configurations and evaluated the estimation accuracy.

For preliminary optimization of the machine learning model's configuration, all available biomechanical signals (joint angle, velocity, acceleration, torque, and activation signal) were used as the input signals. All the sEMG signals were used as the output signals. The input signal sensitivity and the estimation possibility of the output signals were analyzed after the preliminary optimization of the RNN configuration.

The training condition was limited to a maximum of 1-hour and 200 maximum epochs. Different NARX network configurations were tested to evaluate the mapping or pattern estimation of the RNN model. The minimum and maximum optimization limits, along with steps, are presented in Table 4.1.

Table 4.1: The optimization variables, limits, and steps of RNN configuration.

Variable	Symbol	Minimum	Maximum	Step
number of input signal former values	n_u	1	10	1
number of output signal former values	n_y	1	10	1
number of hidden layers	n_l	1	3	1
number of nodes in each hidden layer	n_n	10	50	5

4.3.2 sEMG signal estimation

SEMG signals were measured from 11 sites over the right arm muscles. The goal of this research was to estimate sEMG signals from biomechanical signals. For the mentioned pick and place motion in the sagittal plane, some of the muscles may not be required for the mentioned motion. For evaluating the sEMG signals estimation possibility, each sEMG signal was selected as the output of the RNN model, and the estimation error was calculated as an assessment criterion. This assessment has been done using the general-model for comprehensive results and conclusions.

4.3.3 Input biomechanical signal sensitivity analysis

The joint angle, velocity, torque, and activation signal are the variables in the mathematical muscle model [82]. To evaluate the sensitivity of sEMG signals to input signals, the mapping model was an ANN with a similar configuration to the RNN (same number of hidden layers and neurons in each hidden layer). The ANN was used instead of the RNN since the evaluation of direct relationship is required for the sensitivity analysis. The direct relation means that the model is trained to map the input to the output without considering the input and output signal history. Using an RNN for sensitivity analysis has the same order of results but with higher regression accuracy. We have provided the results of sensitivity analysis with the RNN model for dominant signals in each phase.

The sensitivity analysis to input biomechanical signals was done using a sequential backward elimination algorithm for evaluating the sensitivity and dominant signals. The sensitivity analysis algorithm is shown in Figure 4.4 and consists of the subsequent steps:

i Ignore the signal index $i \in \{N\}$, then use $\{N - i\}$ of biomechanical signals as the input of the ANN model and record the mapping performance with index i.

ii If $i < dim_{\mathcal{R}} \{N\}$ is satisfied, new i=i+1 and then step (i) is repeated.

iii The signal index j with the lowest mapping error or highest mapping performance is labeled as the least dominant biomechanical signal and is deleted from the biomechanical input pool (new $\{N\} = \{N - j\}$).

iv If the biomechanical pool is not empty, go to step (i).

4.3.4 RNN configuration optimization

The number of input signals was reduced using the information from section 4.3.3. The number of output signals was decreased by employing the results of section 4.3.2. Since the input and the output signals were

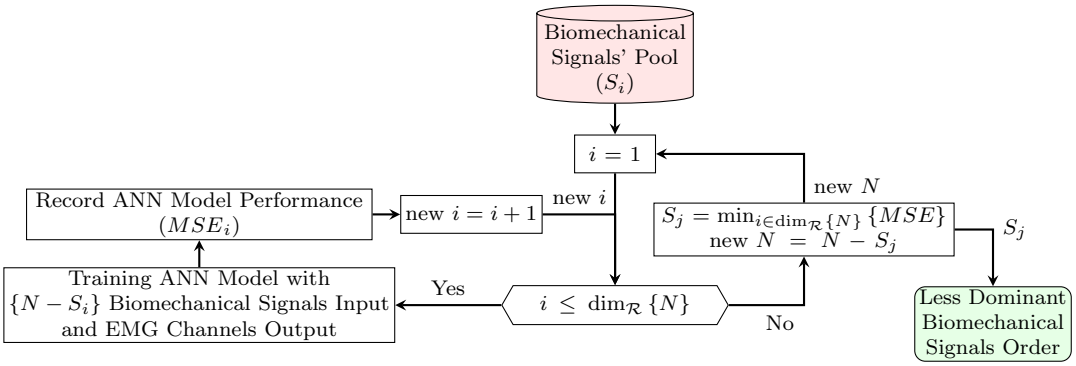


Figure 4.4: Schematic of the sequential backward elimination algorithm for evaluating the sensitivity analysis and dominant biomechanical signals for estimating the sEMG channels.

optimized to the dominant and non-negligible ones, the initial configuration was optimized according to the new conditions of the input and output signals. The final RNN configuration optimization was done using the best results of sections 4.3.1 and 4.4.4.

4.4 Results

The optimum configuration of the RNN model, the analysis of dominant input signals, and the estimation possibility of the sEMG signals are presented in the following sections. A PC with an Intel® Core™ i7-3370 CPU @ 3.40 GHz processor and 16.0 GB memory was used for training the RNN model.

4.4.1 Preliminary RNN configuration

The mentioned algorithm tested 1701 possible configurations, and the top 9 configurations are presented in Table 4.2. The best performance occurred when the number of layers (n_l) was 1, not 2 or 3. Thus, the mapping of biomechanical signals to sEMG signals should be limited to a linear function. Secondly, the optimal number of nodes in each layer (n_n) was approximately 20 to 30. Fewer or extra nodes in each layer lead to lower performance. Fewer nodes means a fewer number of weights and biases, leading to lower regression accuracy. In contrast, extra nodes in a model require more time and data to train [359]. Thirdly, the number of input signal former values (n_u) was between 3 to 6 which shows that the history of biomechanical signals is important for pattern regression of sEMG signals. Finally, the number of the output signal former values (n_y) was required to be between 4 to 7 to give the best performance for the mapping. The number of output signal former values tries to model the dynamics of the sEMG signals. It shows that the current sEMG signal value is related to the previous sEMG signal values. We have used the first RNN configuration in Table 4.2 for the following evaluation processes.

An example of using the RNN model with the preliminary optimized configuration for mapping the biomechanical signals to sEMG signals is shown in Figure 4.5. The validation subject has not participated in the training pool and represents a full prediction. As shown in Figure 4.5, the mapping has a good prediction of sEMG signals in terms of timing and pattern. The subtle "off-pattern" observed may be due to insufficient data regarding external forces, like the target reaction load, which were not measured during the resting period of the experiment [348, 349].

4.4.2 Evaluation of sEMG signal estimation

In Figure 4.6, the performance of biomechanical-sEMG mapping is shown for each sEMG signal. The performance of the general model has different results for each muscle. All regression (training-and-validation) and

Table 4.2: Top performance of RNN model configurations.

Configuration						Average Regression (%)			
n_l	n_n	n_u	n_y	Number of Weights	Inference Time (μs)	Training (14 Subjects)	Validation (2 Subjects)	Test (1 Subject)	All
1	20	3	5	1046	46	91.1	85.5	80.6	89.9
1	20	6	6	1466	45	90.9	85.5	80.0	89.6
1	25	5	5	1556	42	91.5	83.6	73.9	89.5
1	20	6	3	1106	40	92.3	76.1	75.5	89.4
1	25	6	5	1681	41	91.3	81.1	78.2	89.3
1	25	4	7	1731	40	90.2	83.4	83.7	89.0
1	20	4	4	1026	41	90.0	84.6	80.5	88.8
1	30	2	3	1056	41	90.0	80.2	79.1	88.2
1	20	6	5	1346	51	86.7	85.5	82.4	86.3

17-fold CV accuracies are shown in the top and the bottom of Figure 4.6, respectively. Although the regression accuracies have minor differences in both methods, the order of signal regression accuracy is the same. Signal predictions for each muscle (Figure 4.6) are shown from best to worst in the following order: ADEL, INFR, MDEL, PECC, SERR, UTRA, LTRA, MTRA, SUPR, PDEL, LATS. This result does not mean that the dominant muscles, i.e., those primarily responsible for movement, are in the mentioned order. Some dominant muscles are located deeper under the skin, and recording with a surface electrode is subject to error and inefficiency. Rather, the order of dominant sEMG contributions is based on the experimental data, motion, and measurements. For example, although the LATS sEMG signal estimation performance was the lowest, the RNN model still showed good temporal predictions, i.e., following the shape of measured pattern and timing.

4.4.3 Input biomechanical signal sensitivity analysis

The results of the sensitivity analysis of the biomechanical signals are shown in Table 4.3. The analysis started from phase 1, which consists of all the biomechanical signals as the inputs. In the second phase, the analysis used four input signals as the inputs. The mapping performance minorly changed the evaluated regression from 90.0% to 87.9% when the joint acceleration signal was ignored. The joint acceleration, therefore, has a minor impact on the accuracy. For phase 3, four biomechanical signals (all except joint acceleration) were used for the mapping, and the joint velocity signals had the most negligible impact. These phases continued until all signals had been ignored for mapping. The order of most essential signals to least dominant ones was established from this protocol: joint angle, activation signal, joint torque, joint velocity, and joint acceleration. According to the mapping performance in Table 4.3, having all the biomechanical signals for mapping is helpful but not necessary.

4.4.4 Final RNN configuration optimization

The final RNN configuration optimization utilized all five biomechanical signals and 6 of the 11 sEMG signals. The configuration optimization was similar to section 4.3.1, but the 6 sEMG signals consisted of ADEL, INFR, MDEL, PECC, SERR, and UTRA. These were selected according to the observations in section 4.4.2. Suppose the rest of the signals (those five with lower regression accuracy) are needed for specific clinical purposes/applications for different tasks. In that case, more data with different motions and tasks are required for training. The training properties were similar to the previous step. However, the maximum training time was set to 2 hours, and the maximum epoch was set to 200 times. The training could be stopped after 100 epochs if the validation performance had not changed.

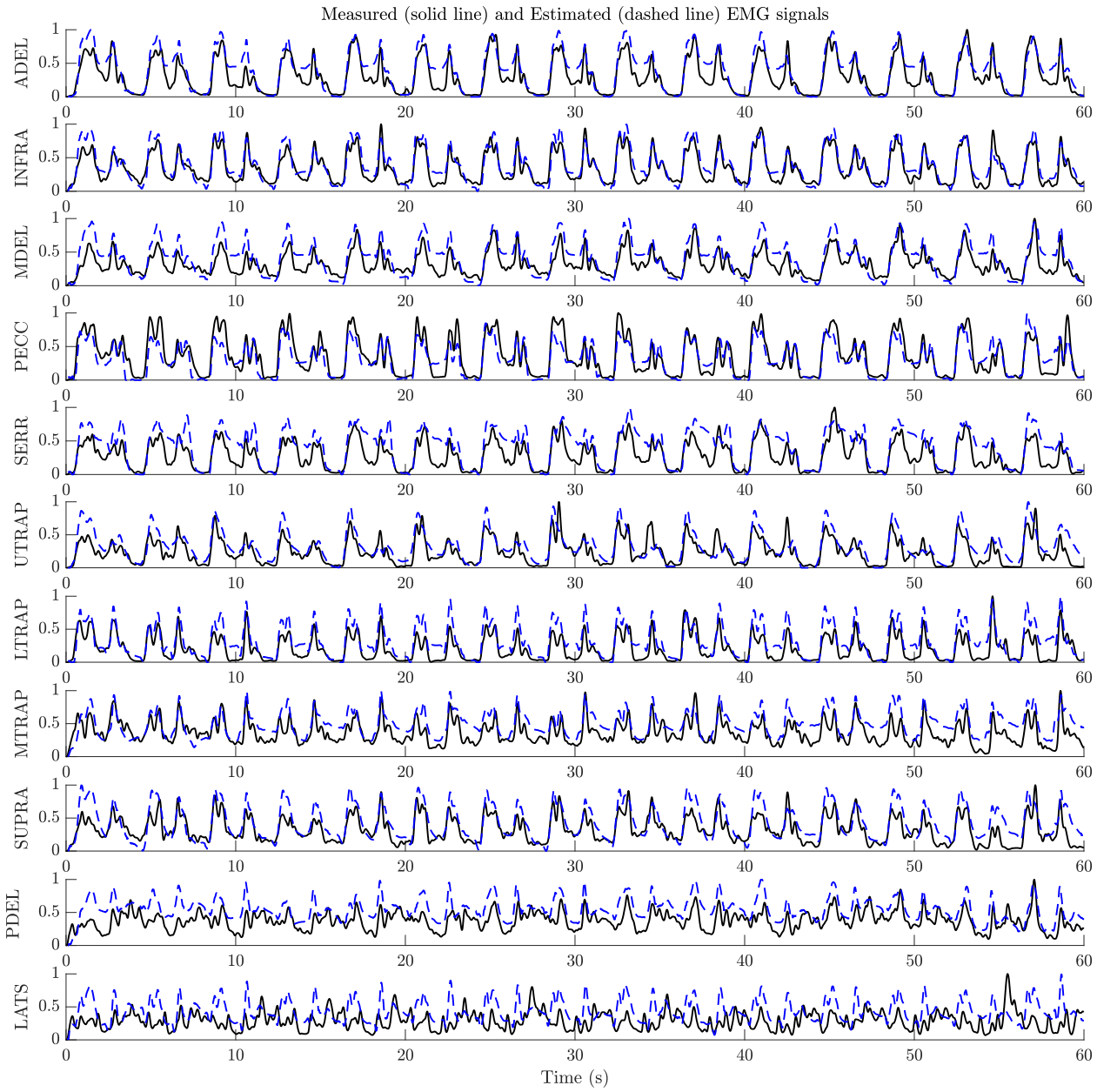


Figure 4.5: The performance of the preliminary configuration of InverseMuscleNET on test data to estimate sEMG signals for 11 upper limb muscles.

In this case, the training was stopped after 134 iterations instead of 200 iterations since the best performance occurred at epoch 34 (Figure 4.7). The regression results for epoch 34 have been shown in Figure 4.8. The error histogram in Figure 4.9 indicates the difference between the measured sEMG signal and the estimated output, which is primarily near zero and has a normal distribution. The regression accuracy for training, validation, test, and all sets are 91.0, 82.6, 88.9, and 89.2%, respectively.

From the optimization, the number of hidden layers (n_l) was 1. The machine learning model consisted of two layers, one hidden layer, and one output layer. The number of nodes in the hidden layer (n_n) was 20. The number of input signal former values (n_u) was 3 and number of the output signal former values (n_y) was 5. Since the sample rate of all input and output signals was 50 Hz, the input signals and the feedback signals should consist of 0.06 and 0.1 seconds of the vector input (containing current and previous states).

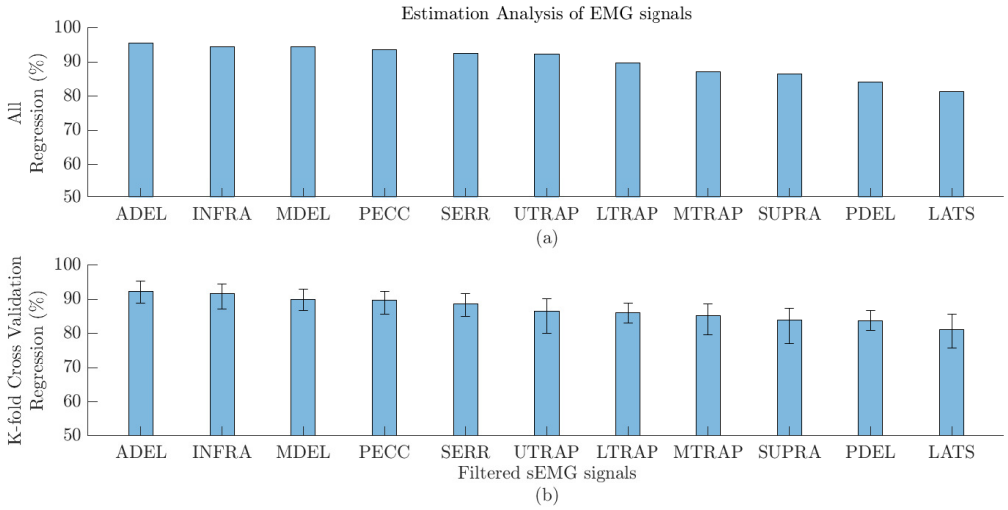


Figure 4.6: The mapping performance of 11 sEMG signals for the general-model. All regression (training and validation) accuracies are shown on the top, and the K-fold CV regression accuracies (average, maximum, and minimum) are shown on the bottom.

Table 4.3: Sensitivity analysis of 5 input biomechanical signals

Phase	Number of Inputs	Input Signals: Joint Angle	Joint Velocity	Joint Acceleration	Joint Torque	Activation Signal	ANN Regression (%)	RNN Regression (%)	Order in Phase	Signals Order	Dominant Signals
1	5	✓	✓	✓	✓	✓	83.9	91.0			
2	4	✓	✓	✓	✓		81.6	85.8	1	5	Joint Acceleration
	4	✓	✓	✓		✓	82.3	86.6	3		
	4	✓	✓		✓	✓	83.8	87.9	5		
	4	✓		✓	✓	✓	82.6	87.0	4		
	4		✓	✓	✓	✓	82.1	86.0	2		
3	3	✓	✓		✓		79.9	83.2	2	4	Joint Velocity
	3	✓	✓			✓	80.9	83.9	3		
	3	✓			✓	✓	81.0	84.4	4		
	3		✓		✓	✓	79.2	82.5	1		
4	2	✓			✓		75.8	82.0	2	3	Joint Torque
	2	✓				✓	76.0	82.9	3		
	2				✓	✓	69.3	79.0	1		
5	1	✓					65.5	77.8	2	2	Activation Signal
	1					✓	58.2	76.9	1	1	Joint Angle

For example, by feeding the biomechanical signals of the test subject to the model and having the initial value of sEMG signal at 0 seconds, the estimation of 6 dominant sEMG signals for 60 seconds has been presented in Figure 4.10. As can be seen, InverseMuscleNET with an optimized configuration could estimate the sEMG signals quite well. The different amplitude of signals shows that the different subjects' muscles have varying strengths. The similar pattern of the estimated and measured signal indicates that muscle activation follows a similar pattern. Since the sEMG signals have a stochastic behavior, time-varying nature, and are subject to measurement error, it is impossible to reach 100% accuracy for the estimation, and the achieved 90% regression

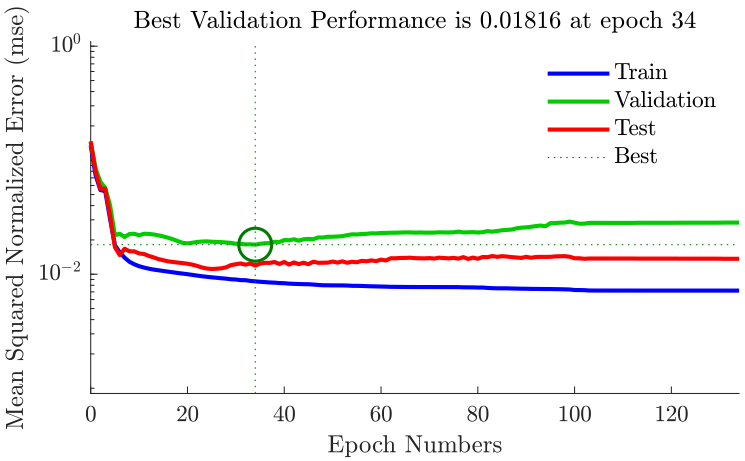


Figure 4.7: The RNN training (blue), validation (green), and test (red) performance (MSE) for 134 iterations. The exported model is based on epoch 34. The training stops after 100 iterations after the best validation performance.

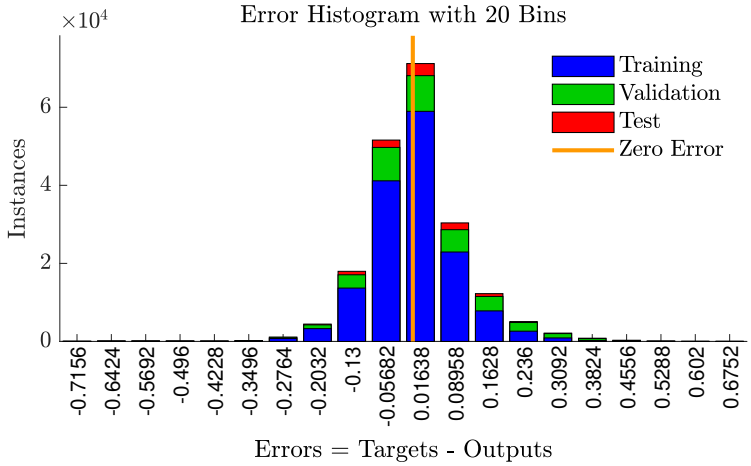


Figure 4.8: Error histogram indicating the difference between actual sEMG signal and estimated output.

is acceptable for biomechanical analysis.

The model's accuracy was slightly increased when using the elbow joint angle, velocity, and acceleration in addition to the previous shoulder joint angle, velocity, acceleration, torque, and activation signal. The training, validation, test, and all regression accuracy increased from 91.0, 82.6, 88.9, and 89.2% to 91.3, 83.3, 88.9, and 90.9%, respectively. The improvement is not much but can help the accuracy of the estimation model. We propose using the joint biomechanical variables relevant to the muscles that impact them.

4.5 Discussion

Optimal control and musculoskeletal modeling have been studied as potential tools for functional rehabilitation and sports evaluation. Despite significant progress, musculoskeletal modeling has encountered some significant problems. The muscle within a musculoskeletal system is complex and challenging to model, and the static optimization simulation requires a time-consuming iterative problem solution based on some assumptions [98, 360]. Altogether, model development and simulation procedure are time-consuming, costly, and prone to error [120]. Thus, it is challenging to use the model in clinical devices. Providing clinicians with a means of estimating sEMG activity without an invasive electrode setup is the primary objective of this research.

This study intended to discover how to predict muscle activations accurately and efficiently using a machine

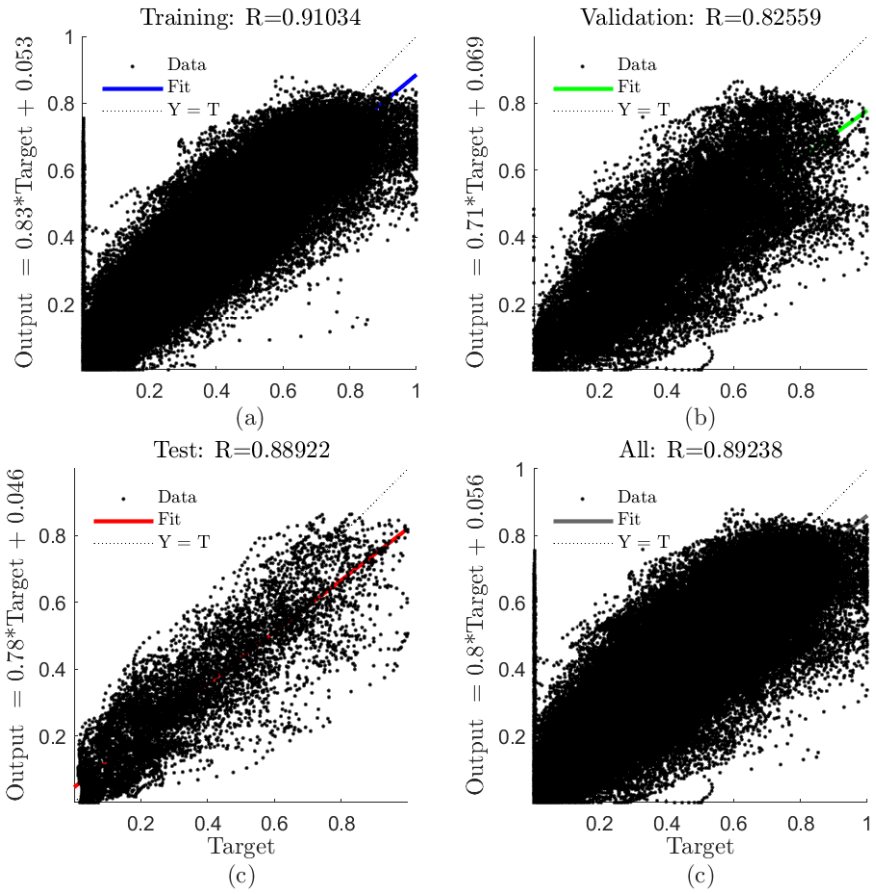


Figure 4.9: Regression or scatter plot of training, validation, test, and all data to RNN model.

learning model, specifically a recurrent neural network. Machine learning is an instrument or model for solving complicated mathematical problems without knowing the analytical relationship between inputs and outputs [118]. However, there are a few disadvantages: it is challenging to determine the optimum network configuration and train the model; obtaining a rich set of input signals to solve the problem is also time-consuming. The main goal was to obtain the muscle activation solution with minor estimation errors and the fastest training convergence.

Recently, machine learning models have been developed to estimate skeletal muscle tensions without explicit modeling of the physical behaviors of muscles [106–109, 118]. However, an inverse muscle model has not been developed using a machine learning model. An optimization problem should be used to estimate the muscle activations since there is no inverse dynamic model of the muscles. A new modeling approach, which provides an inverse dynamic model of muscles, is proposed. This RNN model can offer the ability to compare the subject's sEMG signals via estimated activation signals. Thus, clinicians can avoid sEMG electrode setups in a laboratory and use the InverseMuscleNET. This inverse model completes the inverse dynamic simulation of the skeletal model for biomechanical analysis, post-rehabilitation analysis, and sports engineering/optimization. In addition, InverseMuscleNET can be used within a forward dynamic simulation of the skeletal model. In a forward dynamic simulation, the human motor control can be implemented by nonlinear model predictive control [166]. Other applications include modeling, functional electrical stimulation control, assessment and biofeedback, and biped robotic control [120].

The inverse muscle model was a NARX network, selected as an RNN learning model. The RNN includes a feedback connection to provide memory capacity for time-varying data. Successfully, Dao [108] used an RNN for considering the dynamic temporal relationships of the muscle forces for the first time. We have shown that using an RNN could predict muscle activation by using joint kinematics and kinetics without additional resources.

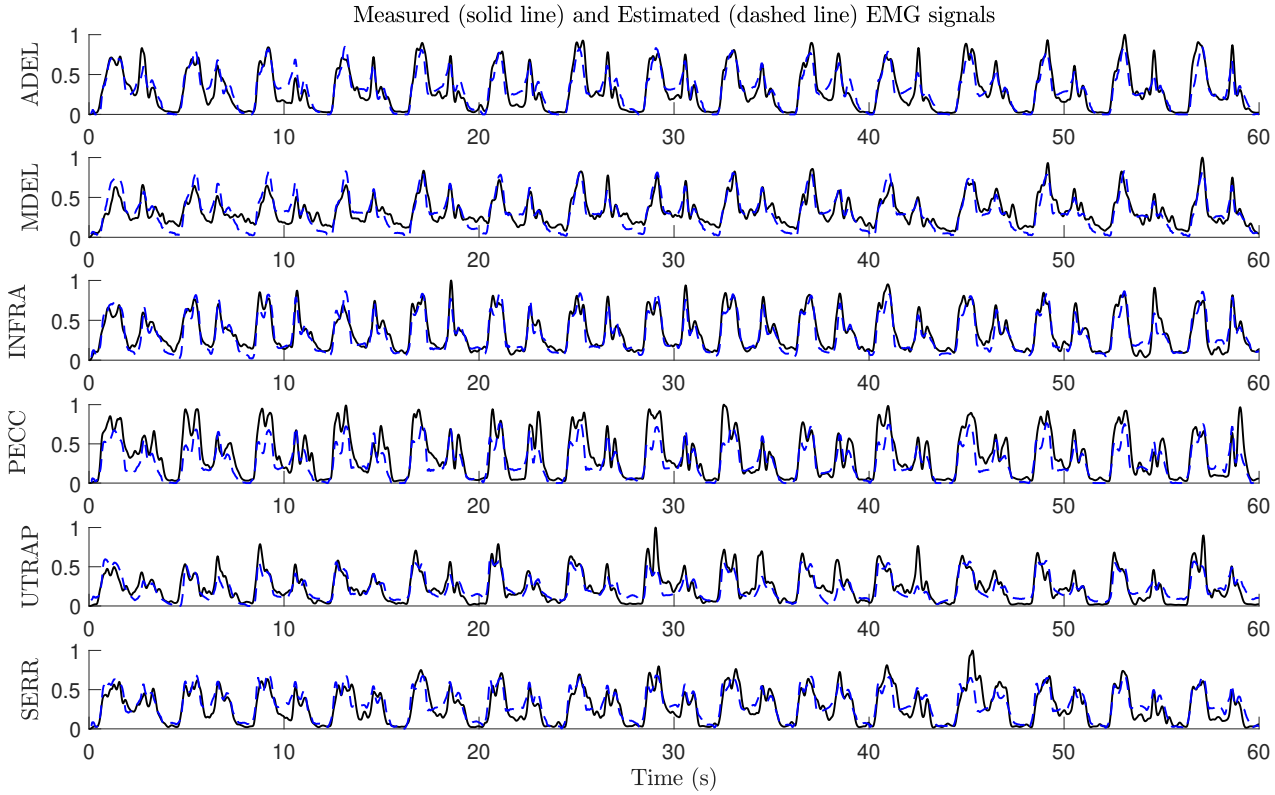


Figure 4.10: The performance of final optimized InverseMuscleNET configuration on test data for the estimation of sEMG signals from 6 upper limb muscles.

Moreover, the RNN model was used to estimate muscle activation with fewer input signals in the sensitivity analysis section. These fewer input signals demonstrated the flexibility of the RNN model in the estimation and muscle model. This model could be trained based on the availability of input data, consisting of joint angle, velocity, acceleration, torque, and activation signal. It is noteworthy to mention that this RNN flexibility could not be done with static optimization approaches.

As an alternative, this model can be resolved using the recently created machine-learning framework known as inverse reinforcement learning [361]. These models can be trained using the data and then reversed. But these models do not have the nonlinear functions, which is helpful for modeling nonlinear systems.

Developing a machine learning model has some challenging aspects, including (I) determining the optimal configuration and (II) requiring a rich and representative dataset for training. We have tested multiple topologies with the same training conditions to determine the model topology's optimal configuration. The number of hidden layers (1), the number of nodes in each layer (20), the number of input signal former values (3), and the number of the output signal former values (5) have been optimized to have the highest mapping accuracy. Although having a model with more parameters may have more accuracy, the model may have a delay and require more data for training. Having fewer parameters in the trained network allows the network to be saved and straightforwardly used in real-time.

Forming a rich database for model training was established on the data collection for a repetitive manual task from 17 healthy right-handed young individuals [348, 349]. Fourteen subjects' data was used for training, two subjects were used to validate the model, and one subject was used for testing the model. Since the validation dataset consisted of 2 subjects, we can be reasonably confident that the exported model was not overfitting the training dataset [362] and is general enough for assessing the upper limb motor task in question. The training dataset consisted of 14 subjects' datasets and is enough for training and finding the general-model since the full estimation has more than 90% regression accuracy. However, as the dataset increases, the final model becomes more general, reliable, and valuable. Since the data is recorded during object manipulation in the sagittal plane, it is not certain that the model can be used for other motions. The model is motion-based, and for a

general-model, a large amount of data is required.

Another strength of this study relates to the rich number of involved muscles. Having access to 11 muscles provides the opportunity to evaluate the estimation possibility. The dominant and most accurate estimation occurred with the ADEL, INFR, MDEL, PECC, SERR, and UTRA for the described task and motion.

Analysis of using five biomechanical signals for the estimation provides the possibility of sensitivity analysis. The order of most essential signals to least dominant ones was as follows: joint angle, activation signal, joint torque, joint velocity, and joint acceleration. This result demonstrated that the estimation of activation signals was relevant to joint angle, activation signal, and joint velocity. The results from the sensitivity analysis concur with previous classical muscle mathematical model studies. The muscle pennation angle and muscle-tendon length have been found to be sensitive to the joint angle and velocity [82]. The maximum muscle force was relevant to the joint torque [82]. Thus, the first three dominant signals correspond to the primary variables in the classical muscle mathematical model.

Since we found that the input signal and the feedback signal should consist of 0.06 and 0.1 seconds of the vector input (containing current and previous states) in the optimization of the RNN configuration process, the estimation of sEMG signals relies on the history of the previous value of sEMG signals and the biomechanical signals.

The estimated and the measured sEMG signals in Figure 4.10 showed that the designed RNN model, in some cases, had slight differences. Occasionally, the measured sEMG signals had small amplitude differences from the estimated muscle activation. However, the pattern of the estimated and measured signals is similar enough for the final applications, which are simulations or clinics. Sometimes, the measured sEMG had high changes while the estimated variable followed a curved path, but the two signals had similar patterns. It is noteworthy to mention that every single computational technique for muscle variable computations has limitations in analytical expressions and suffers from unrealistic assumptions in the muscle models. The model parameters identified with measurements that are subject to error, require an iterative optimization loop, and results in estimated muscle activations that may not be correct [117].

The main limitation of the mentioned model for inverse muscle stimulation is the amount of training data. The machine learning model depends significantly on training data, like the human brain that tunes the neurons' weights and bias using motion and response. As far as we are concerned, there were fourteen sets of input/target pairs from fourteen subjects, two to three subjects' data for validation, and one set for training. The data consisted of SFE movement (the combination of external load, slow, and a fast motion), each of them in twelve trials of the pick-and-placing task. Having more data for more ROMs would facilitate a more general model.

4.6 Concluding remarks

A model estimating the shoulder muscle activation was developed using an optimized recurrent neural network. This estimation was based on biomechanical (kinematic and kinetic) input signals. This study suggests that the classical static optimization of the dynamic model could be replaced with an inverse muscle model using a machine learning model. This approach could increase potential decision support tools for functional rehabilitation with real-time estimation and muscle activation or sEMG signal tracking.

Chapter 5

Human-exoskeleton System Model

As shown in Figure 5.1, the multibody model consists of two sub-models of the human (section 5.1) and the exoskeleton (section 5.2). The connection of these systems is described in more detail in section 5.3. The multibody system was modeled using MapleSim v2021 (Maplesoft, Waterloo, ON, Canada), a multibody dynamic system modeling and simulation software package, and exported as symbolic optimized C code to MATLAB (R2021b. MathWorks, Natick, MA, USA) for use in the design and optimization process.

5.1 Human model

The human body model and the CNS control loop are introduced in this section.

5.1.1 Upper-body dynamic model

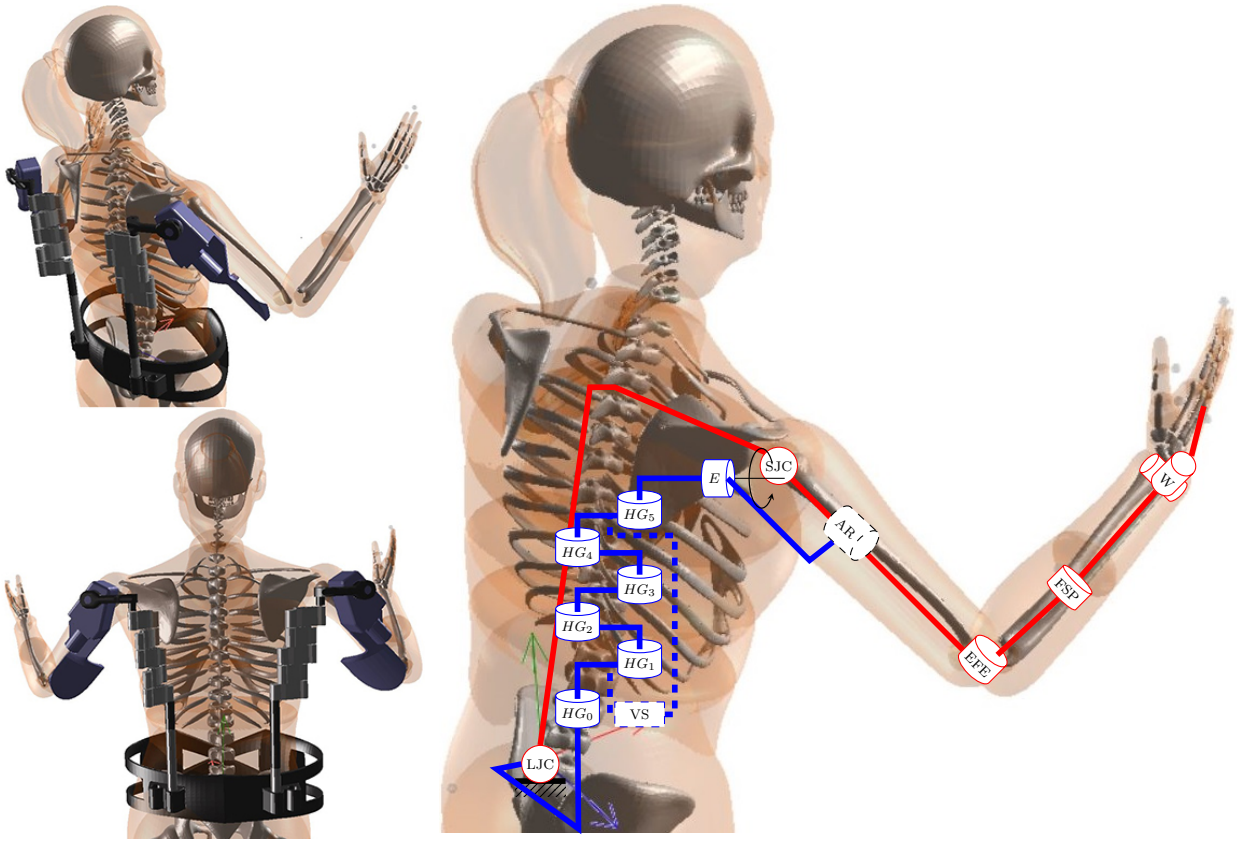
The MapleSim Biomechanics (scalable MSK) model developed and validated using experimental dynamometer data for dynamic simulations of movement was used for the upper body. The MapleSim Biomechanics upper body model is comprised of 8 body segments (torso, head, left/right upper arm, left/right forearm, and left/right hand) and has 20 DoF. The MapleSim Biomechanics model has two components of a skeletal dynamic and biomechanical joint torque model. The biomechanical joint torque model is developed with MTGs [84] that consists of 6 functions of (I) the passive torque, (II) peak joint strength, (III) active-torque-angle scaling, (IV) active-torque-angular-velocity scaling, and (V) excitation-to-activation signal. The MTG parameter values were identified for the simulated subject using a dynamometer device in isometric, isokinetic, and passive modes [28, 84]. The MapleSim Biomechanics model can be adjusted for various anthropometric measurements and body characteristics, including sex, age, body mass, height, dominant side, and physical activity. The adjustable MapleSim Biomechanics model allows for the evaluation of the exoskeleton on different subjects. There are 3 joints at torso (TFE, TRLLF, and TRLAR), and 3 joints at neck (NFE, NRLLF, and NRLAR). For each side of the body, there are 3 DoF at SJC (SFE, SAA, and SMLR), EFE, FPS, WFE, and WRUD.

The MapleSim Biomechanics dynamic model is presented as Equation (5.1). For the active-torque-driven model, Equation (5.3) is used instead of Equation (5.2) which is for the excitation-signal-driven model. Note that Equations (5.1-5.3) are only for the human body and the combined human-exoskeleton model is provided in Equations (5.18-5.20).

$$\begin{bmatrix} I_{2n\times 2n} & 0 \\ 0 & \mathbf{M}_{n\times n}\left(\boldsymbol{\theta}_{n\times 1},\beta\right) \end{bmatrix} \begin{bmatrix} \dot{\boldsymbol{a}}_{2n\times 1} \\ \dot{\boldsymbol{\omega}}_{n\times 1} \end{bmatrix} = \begin{bmatrix} \boldsymbol{\tau}_{u2a}\left(\boldsymbol{a}_{2n\times 1},\boldsymbol{u}_{2n\times 1},t,\beta\right)_{2n\times 1} \\ \mathbf{F}_{n\times 1}\left(\boldsymbol{\omega}_{n\times 1},\boldsymbol{\theta}_{n\times 1},\beta\right)+\mathbf{Q}_{n\times 1} \end{bmatrix} \tag{5.1}$$

$$\mathbf{Q}_{n\times 1} = \left[\boldsymbol{a}^{+}\boldsymbol{\tau}_{\omega}^{+}\left(\boldsymbol{\omega},\beta\right)\boldsymbol{\tau}_{\theta}^{+}\left(\boldsymbol{\theta},\beta\right)\boldsymbol{\tau}_{0}^{+}\left(\beta\right)+\boldsymbol{a}^{-}\boldsymbol{\tau}_{\omega}^{-}\left(\boldsymbol{\omega},\beta\right)\boldsymbol{\tau}_{\theta}^{-}\left(\boldsymbol{\theta},\beta\right)\boldsymbol{\tau}_{0}^{-}\left(\beta\right)+\boldsymbol{\tau}_{p}\left(\boldsymbol{\theta},\boldsymbol{\omega},\beta\right)\right]_{n\times 1} \tag{5.2}$$

$$\mathbf{Q}_{n\times 1} = \left[\boldsymbol{\tau}_{a}+\boldsymbol{\tau}_{p}\left(\boldsymbol{\theta},\boldsymbol{\omega},\beta\right)\right]_{n\times 1} \tag{5.3}$$



(a) EVO with MapleSim Biomechanics

(b) The human-exoskeleton closed kinematic chain

Figure 5.1: (a) Two views of 20-DoF MapleSim Biomechanics model of a human upper body and EVO upper limb exoskeleton. The essential human joints are the LJC with a 3-DoF spherical joint (TFE, TRLLF, and TRLAR), SJC with a 3-DoF spherical joint (SFE, SAA, and SMLR), EFE, FPS, a 2-DoF wrist joint (WFE, and WRUD). (b) Schematic diagram of the human and exoskeleton closed kinematic loop models and their interaction with hinge (HG), elevation (E), and virtual spring (VS) joints. The human and exoskeleton systems are connected at the armrest (AR) and LJC.

where

n	the number of independent coordinates and speeds
$\boldsymbol{\theta}$	the column matrix of all joint angles
$\boldsymbol{\omega}$	the column matrix of all joint angular speed
$\mathbf{M}$	the mass matrix
$\mathbf{F}$	the segments' Coriolis, centrifugal, and gravitational effects
$\mathbf{Q}$	the applied joint torques, a column matrix containing $\tau_h(t)$ for all joints
$\boldsymbol{\tau}_{u2a}$	the excitation-to-activation signal ODE function
$\boldsymbol{\tau}_{\omega}$	the active torque-angular-velocity scaling function
$\boldsymbol{\tau}_{\theta}$	the active torque-position scaling function
$\boldsymbol{\tau}_0$	the peak isometric joint strength
$\boldsymbol{\tau}_p$	the passive torque function due to viscous damping and nonlinear stiffness
$\boldsymbol{\tau}_a$	the vector containing the active torques at the joints
$+$	the positive direction of joint
$-$	the negative direction of the joint
β	the subject adjustment variables: sex, age, body mass, height, dominant side, and physical activity

5.1.2 Optimal controller model of the CNS

The CNS is in charge of the motion of the human body [166, 326]. The CNS synchronizes the kinetics and kinematics regardless of complex MSK dynamics and unknown/uncertain disturbances and trajectories [326]. Human mobility studies suggest that the CNS coordinates body movements based on optimization [271].

The CNS uses a complicated command to coordinate human motions, composed of two sectors: motion prediction and correction command [166]. To compute motion prediction (model-based feed-forward control), an IM as a representation of the complicated system is used. Furthermore, the sensory organs calculate a correction instruction or feedback control to correct any errors caused by model uncertainty, external disruption, or a new environment. NMPC with a short time horizon is shown to be able to attain this multi-purpose control complexity (feed-forward and feedback control) [166]. Precisely, to simulate human dynamics, the NMPC uses an IM that is learned, identified, and tuned over time [333] to forecast optimal motion (feed-forward) and correct prediction errors (feedback) [166]. It uses a cost function to evaluate the ideal motion. Here, we construct this cost function with biomechanical variables such as joint angles and velocity errors, as well as model inputs and their derivative. This cost function is similar to Equation (5.4) for the active-torque-driven model. NMPC controller optimizes the cost functional (Equation (5.4)) under the impact of system dynamics (section 5.3.2) and limitations (Equation (5.5)).

$$J_{CNS} = \int_0^{t_f} \left[\mathbf{W}_1^T \left(\boldsymbol{\theta} - \boldsymbol{\theta}_d\right)^2 + \mathbf{W}_2^T \left(\boldsymbol{\omega} - \boldsymbol{\omega}_d\right)^2 + \mathbf{W}_3^T \left(\boldsymbol{\tau}_a\right)^2 + \mathbf{W}_4^T \left(\dot{\boldsymbol{\tau}}_a\right)^2 \right] dt \tag{5.4}$$

$$\begin{gathered} \boldsymbol{\theta}_{min}\left(\beta\right) \leq \boldsymbol{\theta} \leq \boldsymbol{\theta}_{max}\left(\beta\right) \\ \boldsymbol{\omega}_{min}\left(\beta\right) \leq \boldsymbol{\omega} \leq \boldsymbol{\omega}_{max}\left(\beta\right) \\ -R\boldsymbol{\tau}_0^-\left(\beta\right) \leq \boldsymbol{\tau}_a \leq R\boldsymbol{\tau}_0^+\left(\beta\right) \\ \dot{\boldsymbol{\tau}}_{min} \leq \dot{\boldsymbol{\tau}}_a \leq \dot{\boldsymbol{\tau}}_{max} \end{gathered} \tag{5.5}$$

where

$\mathbf{W}$	the vector of cost function weights
$\boldsymbol{\theta}$	the vector containing the joint angles
$\boldsymbol{\omega}$	the vector containing the joint instantaneous angular velocities
$\boldsymbol{\tau}_a$	the vector containing the active torques at the joints
$\dot{\boldsymbol{\tau}}_a$	the vector containing the instantaneous active torque rates at the joints
$\boldsymbol{\tau}_0$	the peak isometric joint strength
R	the ratio of the maximum eccentric isokinetic torque over the maximum isometric torque
t_f	the prediction horizon
$_d$	the desire value or trajectory, recorded in section 6.1.2 and processed in section 6.1.3
$_{min}$	the minimum variable
$_{max}$	the maximum variable
$^+$	the positive direction of joint
$^-$	the negative direction of the joint
β	the subject adjustment variables: sex, age, body mass, height, dominant side, and physical activity

The cost function weights were selected for long-term experience and robust trajectory tracking according to the study in [28]. The IM is the human model mentioned in section 5.1.1. In addition, the NMPC has full awareness of the interaction force/torque of the exoskeleton and the external force/torque of the manipulated object.

5.2 Exoskeleton model

Here, the rigid links, the joint configuration, the passive torque, and the active torque of the exoskeleton model are described. The topology and kinematic structure of the exoskeleton are already shown on Figure 5.1(b) and highlighted in the Figure 5.2 for further consideration.

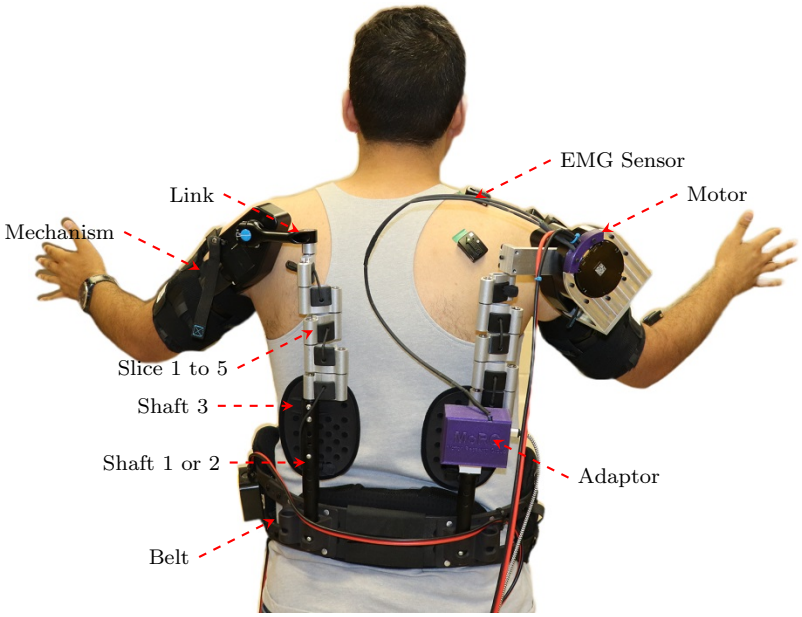


Figure 5.2: An image of the actual active-passive shoulder exoskeleton. The important mechanical parts (e.g., belt, shafts, slices, link, and mechanism) and the electrical components (e.g., adaptor, motor, and sEMG sensor) are shown in the image.

Table 5.1: The coordinates location, mass, COM, and inertia matrix of the mechanical system's links (the right tower of the exoskeleton).

#	Part Name	Coordinate		Center of Mass (m)	Mass (kg)	Inertia Matrix (kgm2)					
		Parent	Child			I_{xx}	I_{yy}	I_{zz}	I_{xy}	I_{xz}	I_{yz}
P1	Belt	LJC	RH1: -0.063, 0.000, 0.135 RH2: -0.067, 0.000, 0.100	0.059, 0.000, 0.000	0.711	0.01	0.017	0.008	-4e-6	0	0
P21	Shaft 1	H	L1: 0.000, 0.145, 0.000	0.000, 0.072, 0.000	0.036	64e-6	3e-6	64e-6	0	0	0
P22	Shaft 2	H	L2: 0.000, 0.220, 0.000	0.000, 0.110, 0.000	0.052	210e-6	3e-6	210e-6	0	0	0
P3	Shaft 3	L	HG0: 0.000, 0.133, 0.000	0.000, 0.071, 0.000	0.031	50e-6	2e-6	50e-6	0	0	0
P4-P8	Slice	HG	HG1: 0.000, 0.036, 0.057	0.000, 0.016, 0.030	0.075	50e-6	46e-6	9e-6	0	0	-1e-6
P9	Link	HG5	E: -0.018, 0.024, 0.080	-0.011, 0.022, 0.040	0.083	116e-6	118e-6	13e-6	1e-6	21e-6	-5e-6
P10	Mechanism	E	AR: 0.126, -0.170, 0.000	0.049, -0.072, -0.020	0.850	0.005	0.001	0.006	0.002	433e-6	-0.001

5.2.1 Dimensions and inertia properties

We used an EVO (Ekso Bionics Holdings Inc, California, USA) upper limb exoskeleton as the wearable device with passive assistance. However, the described method can be applied to any upper-limb exoskeleton. All robot linkages were reverse-engineered utilizing a mixture of physical measurements and reference photos available from the manufacturer to create computer-aided geometric models. A digital caliper, tape meter, and scale were used to measure the dimension of each link, the joint center locations, and the mass of the robot. The center mass and inertia matrix were calculated using the design software with the assumption that the part has constant density. The joint attachment coordinates, the COM, mass, and inertia matrix are provided in Table 5.1.

5.2.2 Joints configuration

The exoskeleton has 2 adjustment links and 7 joints that form a tower on each side. Here, the adjustment options and the joints of the exoskeleton are described. The exoskeleton-human attachment, interaction, and closed-kinematic loop are described in section 5.3.1.

Table 5.2: The hinge's passive joint parameters for the right tower of the exoskeleton.

Joint	K (Nm/rad)	φ_0 (rad)	k_1 (Nm)	k_2 (1/rad)	k_3 (Nm)	k_4 (1/rad)	$\varphi^-_{\max}$ (rad)	$\varphi^+_{\max}$ (rad)	c (Nms/rad)
HG0	0.1	0	1	10	1	10	0	$-\pi$	0.001
HG1	0.1	π	1	10	1	10	$\pi/2$	$3\pi/2$	0.001
HG2	0.1	$-\pi$	1	10	1	10	$-\pi/2$	$-3\pi/2$	0.001
HG3	0.1	π	1	10	1	10	$\pi/2$	$3\pi/2$	0.001
HG4	0.1	$-\pi$	1	10	1	10	$-\pi/2$	$-3\pi/2$	0.001

Each tower can be mounted on 2 different holes on the same belt as a first adjustment link. These holes have different distances from the belt center in the Sagittal axis. The hole should be chosen based on the subject's hip or torso span. The locations of the holes are provided in Table 5.1 with RH1 and RH2 indicating the right side. The second adjustment is a prismatic link in the transverse axis between Shaft 1 or Shaft 2 and Shaft 3. It should be adjusted to minimize the vertical distance of the human shoulder with the exoskeleton's elevation joint. Shafts 1 and 2 are usually used for short and tall subjects, respectively. These two adjustments (hole selection and prismatic vertical adjustment) are not considered joints and should be fixed for simulation and usage. The prismatic adjustment between Shaft 1 or Shaft 2 and Shaft 3 is relevant to the vertical distance of the SJC with the LJC. According to the anthropometric data reported in the literature [68], the adjustment is calculated from Equation (5.6).

$$d = h\left[0.2289s + 0.2199\left(1 - s\right)\right] - l \tag{5.6}$$

where

d	the adjustment distance in meters of Shaft 1 or Shaft 2 with Shaft 3
h	the height of the subject in meters
s	the sex parameter (for a male is 1 and for a female is 0)
l	the exoskeleton maximum vertical distance for Shaft 1 (0.4820 m) and Shaft 2 (0.5570 m)

The joints are connected serially from Shaft 3 to the mechanism segments. The first 5 revolute joints in the vertical direction, like hinges (HG), connect 5 Slices and Shaft 3 together like an accordion. The vertical direction of the sixth revolute joint connects the last Slice in HG5R and the Link. The last exoskeleton joint is the revolute joint in the direction of the shoulder elevation (E) that connects the Link to the mechanism. The last joint has a limited ROM. The assistive torque of this elevation joint is applied with a passive mechanism or actuator (controlled motor).

The first 6 revolute joints are in the vertical direction (in the transverse plane) and are mainly responsible for keeping the exoskeleton's elevation joint at the desired location. The minimum number of required joints to provide this function is 3 revolute joints. These revolute joints can move the last joint (elevation joint) in the traverse and the frontal direction and pose as a desired orientation for the Link segment. If only 3 joints are used, the number of Slices is 3 (as opposed to 6), and they should be longer to cover the full horizontal SAA ROM. By increasing the number of these joints from 3 to 6 and increasing the number of Slice segments from 3 to 6, the dimension of these slices can be smaller. Using smaller slice segments helps in achieving a compact exoskeleton. However, increasing the number of the Slice segments and the connective joints leads to redundancy problems during simulation. Calculating the state of a redundant system requires extra calculation, relying on guessing the joint angles and depending on the initial joint angles.

The first 5 revolute joints in the EVO upper limb exoskeleton have a novel elastic rope that goes through holes on the slices. One end of the rope is connected to the last slice, and the other end is connected to the Shaft3. This elastic rope creates weak springs to keep the accordion-like system in its original location. In addition, the rope introduces a constraint to the 5 joints that inhibit the hinges from rotating in the undesired direction. The hinge's rotational springs and constraints are modeled using Equation (5.7) (symbolically shown in Figure 5.3(a)). Equation (5.7) was originally adopted from [302] to model the stiffness. The equation constrains the maximum range of rotation and is equipped with a rotational spring model of the elastic rope.

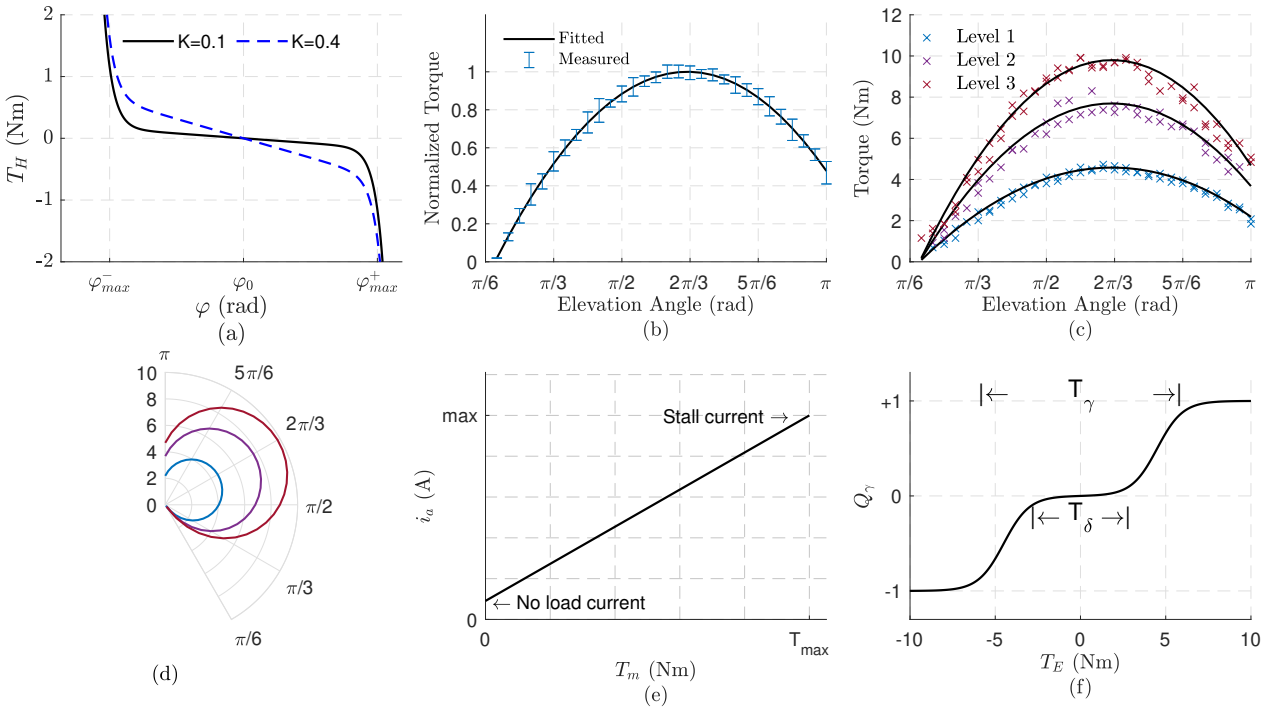


Figure 5.3: (a) Plot of the first 5 revolute joints' passive torques with rotational spring (K) and maximum ROM (φ_{max}). (b) The normalized passive joint torque to the maximum of fitted data, (c) the measured passive joint torque for spring system levels 1 to 3, and (d) a polar indication of joint torque. All the measurements were done with zero rotational shift adjustment for (b-d). (e) The relation of the motor torque (T_m) and armature current (i_a) with motor torque constant (k_m). (f) The smooth curve of the human elevation torque using the dead zone torque (T_δ) and the threshold torque (T_γ).

The passive joint parameters of the hinge were assumed by the authors as listed in Table 5.2. These assumed parameters have no impact on the elevation mechanism and the assistive torque, which are the main focus of this research.

$$T_{Hinge} = -K\left(\varphi - \varphi_0\right) + k_1 e^{-k_2\left(\varphi - \varphi^-_{\max}\right)} - k_3 e^{k_4\left(\varphi - \varphi^+_{\max}\right)} - c\dot{\varphi} \tag{5.7}$$

where

T_{Hinge}	the passive torque of the hinge joints
K	the spring constant, in Nm/rad
φ	the joint angle in radian
φ_0	the unstretched angle of the rotational spring, in radian
k	the passive parameters of the constraint
$\varphi^-_{\max}$	the maximum ROM for negative direction
$\varphi^+_{\max}$	the maximum ROM for positive direction
c	the rotational linear damping coefficient
$\dot{\varphi}$	the joint's instantaneous angular velocity

5.2.3 Passive mechanism model

As mentioned in the introduction, the passive exoskeleton is primarily equipped with a spring mechanism that stores potential energy from human motion. The assistive joint torque profile varies based on the task [26]. The EVO upper limb exoskeleton can aid in positive elevation direction (flexion or vertical abduction) using a

compressed spring and a cam mechanism (Figure 5.4(a)). In addition, it can have different spring strengths and minor rotational shift adjustments for the torque profile.

We have simultaneously measured the assistance force relevant to the elevation angle using a digital gram scale positioned perpendicular to the armrest and a digital inclinometer. The torque is computed by the measured force and length between the scale and the rotation axis. Finally, the joint torque was fitted with a 2-degree polynomial presented in Equation (5.8). The resultant torque-angle profile is shown in Figure 5.3(b-d).

$$T_{Passive} = T_{max}\left[p_2\left(\varphi+\varphi_0\right)^2 + p_1\left(\varphi+\varphi_0\right) + p_0\right] \tag{5.8}$$

where

p_i	the polynomial coefficients ($p_2 = -0.4582$ 1/rad2, $p_1 = 1.9003$ 1/rad, and $p_0 = -0.9704$)
T_{max}	the maximum passive joint torque, which is 4.578, 7.693, 9.798 Nm for the Level 1, 2, 3 springs of EVO, respectively
φ	the exoskeleton's elevation angle in radian
φ_0	the passive mechanism rotational shift (between -0.175 to 0.175 rad)

We created the custom mechanism depicted in Figure 5.4(b) (previously presented in [131]). The assistive passive torque of the mechanism (at the elevation shaft) is given by Equation (5.9). In Equation (5.9), $|\mathbf{P}_W - \mathbf{P}_B|$ and $|\mathbf{P}_A - \mathbf{P}_W|$ are the length from the base point to the wrapping point and the wrapping point to the attachment point, respectively. Thus, the spring deflection is the absolute value of the base-wrap-attachment path minus the initial spring length (L_0). In addition, the direction of the $\mathbf{P}_A - \mathbf{P}_W$ vector defines the direction of the spring tensile force to the attachment point. Finally, the consequent torque is defined as the cross-product of the position vector to the attachment point (from the rotation center) and the spring force. We assume the cylinder radius to be negligible; thus the wrapping point is the center of the cylinder in Figure 5.4(b). The length of the spring decreases as the angle of the elevation joint increases. Simultaneously, as the angle of the joint increases, so does the moment arm of the spring force. As shown in the custom mechanism (Figure 5.4(b)), as the angle of the elevation joint increases, the moment arm increases while the spring length decreases. The delivered passive torque is produced by the combination of decreasing the force of the spring (due to shorter spring length) and increasing the moment arm of that force. Even though the spring is linear, the mechanism assembly is designed to produce a nonlinear torque-angle relationship (plotted in Figure 6.5(a)).

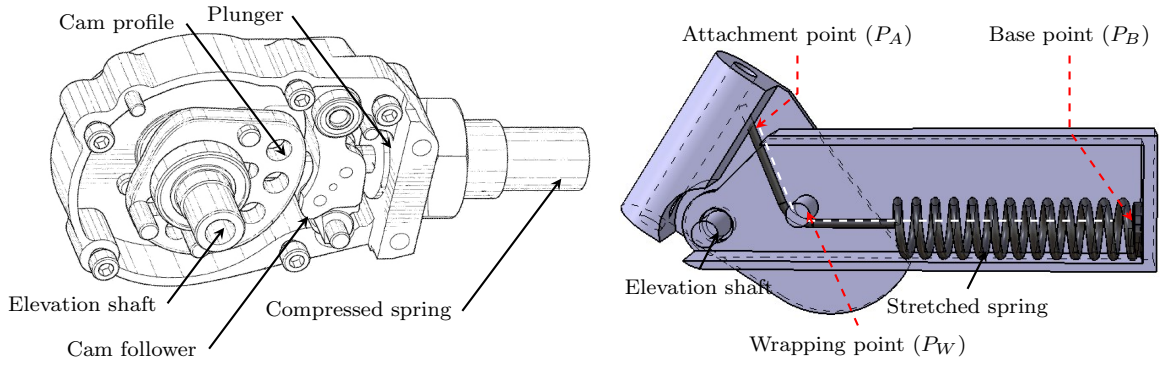
$$T_{Passive}\left(\varphi\right) = \left|\mathbf{P}_A \times\ k\left[\frac{\mathbf{P}_A - \mathbf{P}_W}{|\mathbf{P}_A - \mathbf{P}_W|}\left(|\mathbf{P}_A - \mathbf{P}_W| + |\mathbf{P}_W - \mathbf{P}_B| - L_0\right)\right]\right| \tag{5.9}$$

where

k	the spring stiffness proportional constant
L_0	the initial length of the spring (equilibrium position)
$\mathbf{P}_A$	the coordinates of the attachment point relative to the frame at the elevation shaft
$\mathbf{P}_W$	the coordinates of the wrapping point relative to the frame at the elevation shaft
$\mathbf{P}_B$	the coordinates of the base point relative to the frame at the elevation shaft

5.2.4 Powered actuator model

Preliminary robotic exoskeletons and prostheses employed pneumatic and hydraulic actuators tethered to off-board bulky air compressors and fluid pumps resource stations, respectively [364, 365]. However, research and industry have moved to electromagnetic actuators, particularly brushless and brushed direct current motors [364, 365]. Motors are most efficient at high speeds and minimal torques, with rough torque densities of 15 Nm/kg and power densities of near 200 W/kg, and a 90% efficiency [366]. On the other hand, human muscle groups have torque and power densities of roughly 20 Nm/kg and 50 W/kg, respectively, with a concentric contraction efficiency of around 30% [366]. To increase the motor torque output to the robot's joints, electric motors are



(a) EVO mechanism [363]

(b) Custom mechanism

Figure 5.4: Views of the passive elevation joint mechanisms assembled using a compressed spring (a) or stretched spring (b). (a) Evo providing an assistive torque to the upper arm of a wearer via a compressed spring [363]. (b) Our designed custom mechanism (adopted from Nasr et al. [26]) delivering passive torque via a stretched spring that wraps around a point P_B.

combined with a high-ratio transmission (mostly harmonic gearing); this design allows for precision position control and causes the robotic actuator to have a high mechanical impedance (i.e., high inertia and mechanically stiff) [367]. However, utilizing actuators with high output impendence in wearable robots generally increases energy consumption and electric motor peak power requirements [367]. Thus, it is evident that designing a wearable robot actuator requires considering the primary actuator dynamic, which was ignored in designing active-passive exoskeletons to the best of our knowledge.

The following design requirements should be achieved to reduce system inefficiencies: low electric power consumption, low inertia and impedance, and high torque density motors with little gearing. To attain low mechanical impedance, high energy efficiency, and back-drivability, torque-dense motors with low ratio transmissions, also called direct-drive actuators, have been deployed in recent years [368]. Developments in high torque-density (torque per unit mass) BLDC motors by the drone industry have significantly fueled the use of low transmission ratios in wearable robots. The design of BLDC motors increases delivered torque by rising motor torque-density instead of the transmission ratio, decreasing the impacts of high-gearing (i.e., backlash, torque-dependent Coulomb friction, increased weight, and the actuator's effective inertia and damping, which scale with the square of transmission ratio) [366, 367]. Thus, to overcome the resistance and back drive the actuator, significant external loads are required when the active component in an active-passive exoskeleton is off. The properties of high-gearing systems introduce dynamic stiffness in physical interactions between humans and active machines. Recently, following the mentioned actuator design concepts, the MIT Cheetah legged robot was built using pancake-style BLDC motors with encapsulated windings [369].

The electromechanical characteristics of commercial BLDC motors were gathered in Table 5.3. The most critical motor specifications are effective inertia and nominal torque. In active-passive exoskeletons, when the actuator is turned off, and only the passive component is utilized, the actuator should have the lowest output inertia. Furthermore, the needed torque should always be smaller than the nominal torque. The achievable speed lowers as the required torque grows and the motor heats up.

A motor's torque-producing ability is usually expressed as Equation (5.10) and Figure 5.3(e). Here, we are not considering an electrical equation such as the motor-armature inductance. The gear ratio (N) generates a rise in torque at the load and consequently a decrease in the load's speed. Without considering the loaded dynamic, the net output torque can be determined by considering the motor rotor inertias and the rotator viscous friction coefficients as seen in Equation (5.11) [370].

$$T_m = k_m i_a \tag{5.10}$$

$$T_{Output} = NT_m - \left(N^2 I_m \ddot{\varphi} + N^2 b_m \dot{\varphi}\right) \tag{5.11}$$

where

Table 5.3: The electrotechnical specification of BLDC motor with or without a gearbox from CubeMars Co. (Nanchang, China).

#	Motor Model	Gear Ratio	Nominal Torque (Nm)	Rotor Inertia (kgm2)	Viscous Damping Coefficients (Nms/rad)	Motor Weight (kg)	Nominal Rotary Speed (rad/s)	Nominal Input Current (A)	Torque Constant (Nm/A)	Input Voltage (v)
1	AK80-64	64	48	564e-7	0.065	0.850	2.93	6.3	0.119	24-48
2	AK10-9 V2.0	9	18	1002e-7	0.130	0.960	23.67	11.8	0.16	24-48
3	AK10-9 V1.1	9	15	1002e-7	0.101	0.820	46.60	18.5	0.095	24-48
4	AK80-9	9	9	607e-7	0.054	0.485	25.66	12.0	0.091	24-48
5	AK80-6	6	6	607e-7	0.054	0.485	38.22	12.0	0.091	24-48
6	AK70-10	10	8.3	414e-7	0.034	0.521	41.89	8.8	0.095	24-48
7	AK60-6 V1.1	6	3	243e-7	0.021	0.368	23.04	4.5	0.113	24
8	R100 KV90	1	3.7	2505e-7	0.221	0.731	397.94	35.0	0.106	48
9	R80 KV110	1	1.3	758e-7	0.060	0.354	481.71	15.0	0.0868	48

Table 5.4: The coordinates location, mass, COM, and inertia matrix of the actuator cover.

Part Name	Parent Coordinate	Center of Mass (m)	Mass (kg)	Inertia Matrix (kgm2)					
				I_{xx}	I_{yy}	I_{zz}	I_{xy}	I_{xz}	I_{yz}
Motor	E	-0.042, 0.000, 0.000	0.669	75e-5	51e-5	50e-5	0	-1e-5	0
Housing	E	0.015, -0.050, 0.000	0.144	32e-5	17e-5	25e-5	4e-5	0	0
Arm	HG5	-0.005, 0.024, 0.061	0.069	4e-5	4e-5	1e-5	0	1e-6	0

T_m the torque applied to the rotor
k_m the motor torque constant
i_a the armature current
T_{Output} the net output torque
N the gear ratio
I_m the motor rotor inertias
b_m the rotator viscous friction coefficients

These pancake BLDC motors have average 10 cm diameter, 7 cm height, and 0.669 kg mass. The covering and fixture inertia properties are provided in Table 5.4 for an average motor dimension.

5.2.5 Hierarchical assistance model

High, mid, and low-level controllers are used in advanced hierarchical control of robotic exoskeletons, prosthesis, assistive, and rehabilitation systems [19, 28]. Primary, the high-level controller is in charge of understanding the user's motion intent for robotic prosthesis systems or the torque and force exerted by the user for wearable robotic systems [38, 170]. Here we assume that the human active torque ($\hat{\tau}_a$) is predicted accurately by the high-level controller. On the other hand, the low-level controller is in charge of directly commanding the robotic actuators considering the sensed robotic joint variable, following the desired torque (τ_d) that is generated by the mid-level controller [19, 28, 137]. Here, we assumed the low-level controller has a one-to-one model and can apply the desired torque (τ_d) accurately. Finally, the mid-level controller uses the high-level controller's determined intended motion or force/torque and defines the low-level controller's command [28]. The details of the mid-level controller is described in the following paragraph.

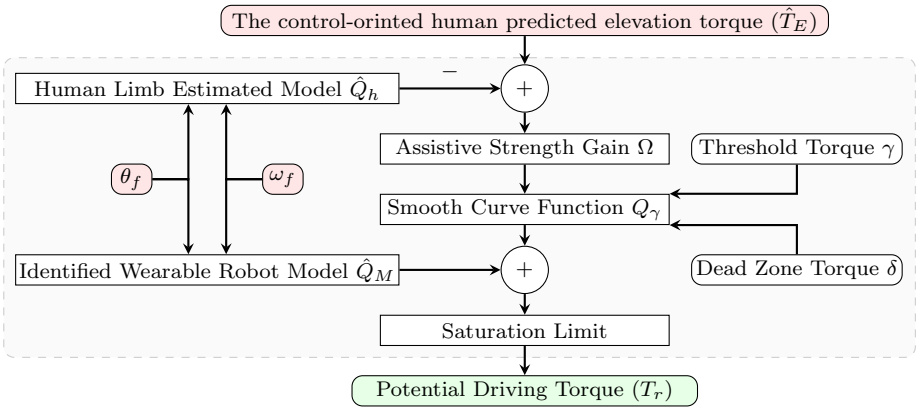


Figure 5.5: Block diagram of the mid-level controller of the hierarchical AAN-CTM control architecture of active-passive exoskeleton robots. The mid-level controller defines the potential driving torque using the human and exoskeleton model [28].

Recently, the AAN-CTM technique, which boosts rather than replaces motor activity, has shown encouraging outcomes for the mid-level controller [28]. Here, we adopted the controller in [28] (Equations (5.12-5.17)), considered the full AAN approach, and designed it to provide the required assistance torque (Figure 5.5). This unique approach finds the desired actuator torque by boosting the human active torque via the desired strength variable (Ω) and combining the robot's powered actuator model in Equation (5.15). The comprehensive model was presented using Equations (5.12, 5.13). The hyperbolic Equation (5.13) (symbolically shown in Figure 5.3(f)) eliminates the controller chattering effect in the presence of low human joint torque. In the simulation, the desired elevation torque (T_E) was calculated using the human shoulder joint torque and Equation (5.14). In practice, the elevation torque ($\hat{T}_E$) is estimated from the high-level controller using human data, for example, with MuscleNET [38, 170] and a wireless sEMG sensor, as shown in Figure 2.2. The maximum motor nominal torque limits the desired assistive output torque to mimic the practical actuation limit in Equation (5.16). The desired input current (i_a) calculated with Equation (5.17) should be driven by the hierarchical low-level actuator controller, which is primarily a servo controller.

$$T_d = \Omega Q_\gamma \hat{T}_E + \hat{Q}_M \tag{5.12}$$

$$Q_\gamma = \left| \frac{1}{1 + e^{-\frac{4}{T_\gamma - T_\delta}\left(\hat{T}_E - \frac{T_\gamma + T_\delta}{2}\right)}} - \frac{1}{1 + e^{\frac{4}{T_\gamma - T_\delta}\left(\hat{T}_E + \frac{T_\gamma + T_\delta}{2}\right)}} \right| \tag{5.13}$$

$$T_E = \left({}^{G}R_{UA} \begin{bmatrix} \tau_{SAA} \\ \tau_{SMLR} \\ \tau_{SFE} \end{bmatrix} \right) \bullet \left({}^{G}R_{E} \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \right) \tag{5.14}$$

$$\hat{Q}_M = N^2 \hat{I}_m \ddot{\varphi}_f + N^2 \hat{b}_m \dot{\varphi}_f \tag{5.15}$$

$$T_r = min\left(max\left(T_d, -NT_{max}\right), NT_{max}\right) \tag{5.16}$$

$$i_a = \frac{1}{N\hat{k}_m} T_r \tag{5.17}$$

where

T_d	the desire assistive output torque
Ω	the robot's desired assistive strength variable, which is positive for assistive and negative for resistive (rehabilitation) control
$\hat{T}_E$	the human predicted elevation torque by the high-level controller
Q_γ	the smooth curve of the human elevation torque between the dead zone and the threshold torque
$\hat{Q}_M$	the identified wearable robot actuator model, the result of the rotor inertia and the viscous damping
T_γ	the positive threshold torque

T_δ	the positive dead zone torque
${}^G R_{UA}$	the rotation matrix of the upper arm frame relevant to the global reference frame
${}^G R_E$	the rotation matrix of the mechanism frame attached to the elevation joint relevant to the global reference frame
τ_{SAA}	the human SAA joint torque
τ_{SFE}	the human SFE joint torque
τ_{SMLR}	the human SMLR joint torque
$\dot{\varphi}_f$	the actuator joint angular velocity that is calculated with the derivative of the joint angle
$\ddot{\varphi}_f$	the actuator joint angular acceleration that is calculated with the derivative of the joint angular velocity
T_r	the potential driving torque, which is saturated by the maximum motor torque
T_{max}	the maximum nominal motor torque
$\hat{k}_m$	the identified torque constant

5.3 Closed-kinematic system

Here, the connection of the exoskeleton to the human, the combined model, and the requirements of simulation are described.

5.3.1 Exoskeleton-human interaction

The belt of the exoskeleton fastens around the hip or connects to the LJC. The LJC is estimated as 26.4%, 0% and 12.6% (for male) of the Pelvis width on the X-, Y- and Z-axes, or 28.9%, 0% and 17.2% (for female), respectively. The Z-axis is in the direction of the left-to-right anterior superior iliac spine (the most prominent point on the anterior superior spine of the right and left ilium). The X-Z plane is formed with the Z-axis and the midpoint between the postero-superior iliac spines (the prominent points on the right and the left ilium's posterior superior spine). In total, the assistive torque reaction on the shoulder conveys to the exoskeleton belt and, consequently, to the hip. The belt of the exoskeleton and the LJC of the human are rigidly connected. For future studies, the connection can be modeled with a spring and damper [371].

The exoskeleton's armrest is attached to the human upper arm with a specific joint. The specific joint is a 1-DoF revolute joint, which allows for one rotational DoF about the common axis. The common axis is the Y-direction of the upper arm (the line connecting the elbow joint center (EJC) to the SJC, pointing proximally). The connected 1-DoF revolute joint can easily rotate around the Y-axis, but has 5 constraints that prevent rotation in the X- and Z-direction and translation in all directions of the human upper arm coordinate system. The mentioned 5 constraints convey the interaction forces and torques to the human upper arm from the exoskeleton's armrest. The interaction forces and torques result from the configuration of the human-exoskeleton connections and the assistance torque by the passive mechanism and powered actuator.

5.3.2 Multibody model

The stated two connections at the exoskeleton's belt and armrest to the human hip and upper arm form the closed kinematic loop. The closed kinematic loop and the joint system configuration cause some exoskeleton joints to be kinematically dependent on the human joints [26]. Since the generalized coordinates (q) and generalized speeds (p) are dependent, the multibody system dynamic equations can be derived with the constraint reactions occurring explicitly in Equation (5.18), which are DAEs model.

$$\mathbf{M}_{n\times n}\dot{\mathbf{p}}_{n\times 1} + \mathbf{C}_{n\times m}\boldsymbol{\lambda}_{m\times 1} = \mathbf{F}_{n\times 1} + \mathbf{Q}_{n\times 1} \tag{5.18}$$

$$\Phi(\mathbf{q}_{n\times 1}, t) = 0 \tag{5.19}$$

$$\Psi(\mathbf{p}_{n\times 1}, \mathbf{q}_{n\times 1}, t) = 0 \tag{5.20}$$

$$\chi(\dot{\mathbf{p}}_{n\times 1}, \mathbf{p}_{n\times 1}, \mathbf{q}_{n\times 1}, t) = 0 \tag{5.21}$$

where

$\mathbf{M}$	Mass matrix, a combination of the human model, the exoskeleton, and the mass matrixes of the assistive components
$\mathbf{C}$	Coefficient matrix of constraint reactions
$\boldsymbol{\lambda}$	Reaction forces and torques in the cotree joints that enforce the kinematic constraint of human-exoskeletons connection
$\mathbf{F}$	The right-hand side of dynamic equations
$\mathbf{Q}$	The vector containing the applied wrench (torque/force) at the joints, which consists of the human joint torque (τ) and the robot joint torque (T)
Φ	Position-level kinematic constraint equations
Ψ	Velocity-level kinematic constraint equations; if the generalized speeds are equal to the coordinate derivatives, then $\Psi = \dot{\Phi}$
χ	Acceleration-level kinematic constraint equations
$\mathbf{q}$	Generalized coordinates of the multibody system, which consists of the human joint angles (θ) and the robot joint states (φ)
$\mathbf{p}$	Generalized speeds of the multibody system, which consists of the human joint angular velocities (ω) and the derivative of robot joint states ($\dot{\varphi}$)
n	number of total joints
m	number of constraint reactions

5.3.3 Multibody analysis

The human-exoskeleton closed-loop system has 10 algebraic constraints (5 for each side). These 5 constraints are due to the human-exoskeleton interaction forces and torques described in section 5.3.1. The human body model itself has 20-DoF (section 5.1.1). The exoskeleton should have at least 10-DoF (5-DoF for each side) to avoid decreasing, resisting, or locking the human DoF. In total, the multibody system was modeled using 30 generalized coordinates coupled with 10 algebraic constraints. Specifically, there are 20 human joint angles (θ), 10 robot joint states (φ), and 10 reaction forces and torques (λ).

5.3.4 Simulation requirement

The human upper limb with the shoulder exoskeleton forms a closed-kinematic loop. In contrast to an open-kinematic chain, a combined exoskeleton produces residual forces/torques if the exoskeleton DoF are inadequate to align with the motion axis of a human joint, or if there are any human-robot kinematic differences [372, 373]. A limited number of passive or self-adjusting joints are utilized to address this issue [238]. For example, the mentioned EVO exoskeleton has utilized 6 passive joint configurations (section 5.2.2).

Theoretically, the human joint angles are determined by the measured human joint trajectory (section 6.1) or the CNS (section 5.1.2). The exoskeleton joints angle, angular velocity, and angular acceleration should be defined by the human kinematic using Equations (5.19-5.21). The initial conditions of the exoskeleton joints are ignored or guessed by the software at the start of the simulation.

Simulating the redundant configuration introduces hardships when calculating the initial joint angles in simulation software like MapleSim. Calculating the initial joint angles has no unique solution and has infinite solutions. One solution is to apply a virtual prismatic joint with liner springs parallel to the system of the second and fifth revolute Hinges joints (shown in Figure 5.1(b)). Using the virtual prismatic joint increases the chance of choosing a realistic solution from infinite options.

As mentioned in section 5.2.2, there are 6 joints in the Y-direction. When the five Slices are aligned or fully extended, the closed kinematic chain experiences a singularity, introducing an impulsive force internally. Particularly, the required torques to move from that specific configuration are infinite. Using the mentioned elastic rope in practice or using the hyperbolic function of Equation (5.7) in simulation, introduce a constraint. The constraint inhibits the singularity configuration from happening.

5.4 Concluding remarks

In this chapter, the human-exoskeleton multibody model was presented. The multibody model consisted of two sub-models: the human and the exoskeleton. The exoskeleton system is connected to the human on the belt and armrest. The multibody system was modeled using MapleSim, and exported as symbolic optimized C code to MATLAB for use in the design and optimization process.

Chapter 6

Active-passive Exoskeleton Design

Exoskeleton robots, which range from fully passive to fully active-assisted movements, have become an essential instrument for assisting industrial employees and stroke rehabilitation therapy. Designing the exoskeleton actuation is challenging and time-consuming due to closed-kinematic loops in the 3D human-exoskeleton multibody model, complicated interactions, and interdependent selection of power transmission features. This chapter proposes a process for dynamic syntheses of passive and active assistive shoulder exoskeletons. The desired motion was experimentally measured from six tasks: frontal reaching, left to right reaching, overhead reaching, sagittal-plane object handling, frontal-plane object handling, and over-head object handling. The system design was optimized by choosing features of the passive mechanism and exoskeleton motor such that the human joint active torque, power, muscle metabolic energy expenditure, and actuator electricity consumption were minimized. The dynamic synthesis processes were found to be successful, and the resultant optimized active-passive exoskeletons allow for the creation of lighter and smaller wearable robots that reduce the user's muscular activation torque for the tasks being studied.

First, the six recorded industrial tasks or motions were introduced in section 6.1. Second, the design variable, targeted cost function, the procedure of actuator designing, and the model-based optimization method are introduced (section 6.2). Finally, the results for different tasks, cost functions, and actuators are presented and discussed.

6.1 Data acquisition and processing

6.1.1 Tasks

Occupational tasks are generally used for evaluating the efficient design of industrial exoskeletons that reduce muscular demands [126]. For example, overhead work and carrying a heavy load can be a challenging risk to eliminate in specific industries, such as automotive production and hand assembly [126]. So far, a couple of industrial and daily living tasks are used for evaluating upper limb exoskeletons, such as (1) overhead raising, maintaining, operating, and moving [9, 146, 374–377], (2) lifting a load from the ground and transferring [331, 378], (3) general raising, reaching, and lowering arms [14, 165], (4) static holding in stooped posture [145, 331], (5) plugging needle into holes and groove [379], (6) surgery operation simulation [380], (7) daily living tasks (e.g., eating with a spoon, opening a door, zipping a coat, and combing hair) [14]. Six tasks were designed to simulate industrial and daily activities to guarantee that each joint had a wide range of amplitudes. As depicted in Figure 6.1, tasks are categorized into two groups: arm moving and object handling. The tasks are frontal reaching, left to right reaching, overhead reaching, sagittal-plane object handling, frontal-plane object handling, and over-head object handling.

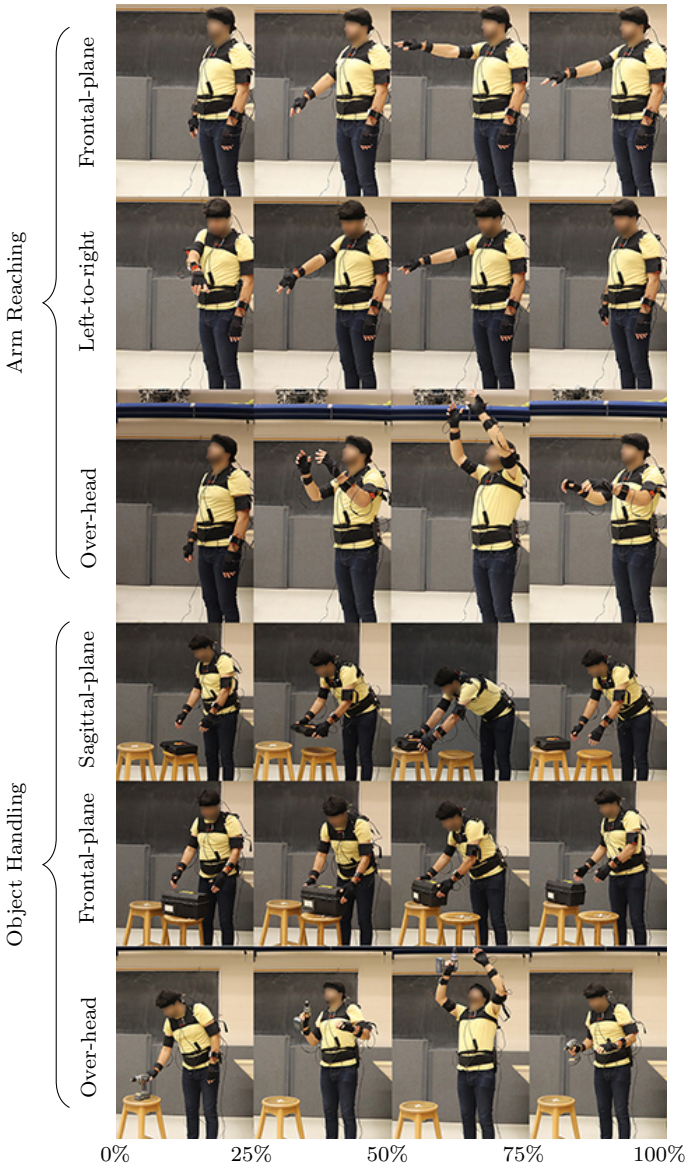


Figure 6.1: Illustration of six industrial and daily living activity tasks for designing, optimizing, and evaluating the shoulder exoskeleton. The first three motions are arm-reaching tasks, and the last three motions are object-handling tasks.

6.1.2 Data recording

One volunteer (male, age: 29 years; mass: 76 kg; height: 1.80 m; right-handed; 1 session workout per week), free of upper limb injury, provided informed consent and was recruited for this study. The experimental protocol followed the Declaration of Helsinki's guidelines. The University of Waterloo's Office of Research Ethics approved the data collecting study (ORE #: 31194).

An MVN Suit (Xsens Technologies, Netherlands) based on the IMU technology was used to quantify the angular human joint kinematics (without exoskeleton). The subject conducted tasks with 2 minutes of rest between tasks after a standard calibration of the IMU system. The data were captured at a rate of 100 Hz.

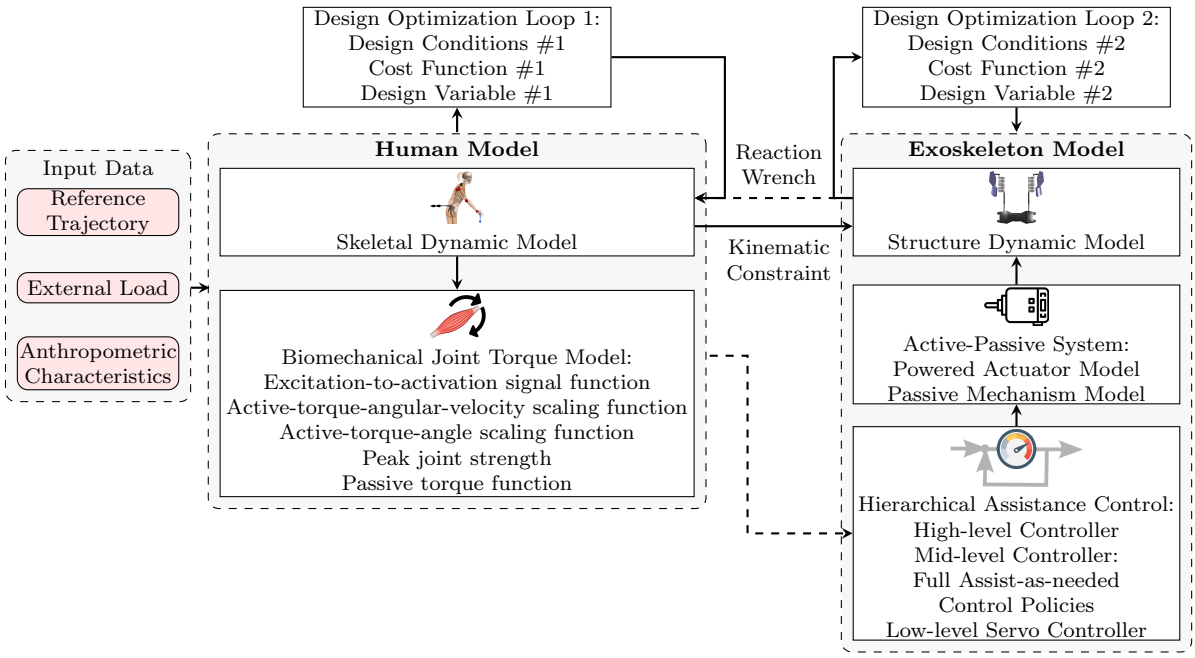


Figure 6.2: Schematic of designing, synthesizing, analyzing, and optimizing the inverse scenario.

6.1.3 Kinematic process

Similar to the kinematic process used [26, 347], the following steps were applied to filter the joint angles and estimate the joint angular velocity, acceleration, and the raising and lowering of the object.The processed kinematic data were used as the reference trajectory in section 6.2.1.

A 10 Hz low-pass filter was used to ignore the high-frequency noises. Then, the position data were re-sampled with a rate of 50 Hz using 1-D data cubic interpolation of the neighboring values. The velocity and acceleration of the joints were initially calculated using the numerical first and second derivatives of joint angles. Then, with a cut-off frequency of 20 and 30 Hz, two low-pass second-order Butterworth filters were used to ignore minor variations.

The object's position and velocity were used to distinguish the raising and lowering processes. The object was estimated at the resting targets when the absolute velocity of the object was less than 1 cm/s for more than 30 milliseconds.

6.2 Design optimization

6.2.1 Overall method

The exoskeleton actuator designing process is achieved through forward or inverse simulation of the multibody dynamic system (depicted in Figure 6.2 and Figure 6.3). So far, the inverse scenario (section 6.2.1) was used to find the passive mechanism's optimum design [26], which is not controlled by the system and is a function of joint angle, not time. The forward scenario (section 6.2.1) is used to evaluate the optimum actuator parameters for motor control, which depends on the human intention prediction. Broadly, the active component is controlled as a function of time. The desired motion trajectory (generalized coordinates, speeds, and acceleration), the human, and the exoskeleton information are necessary for both scenarios.

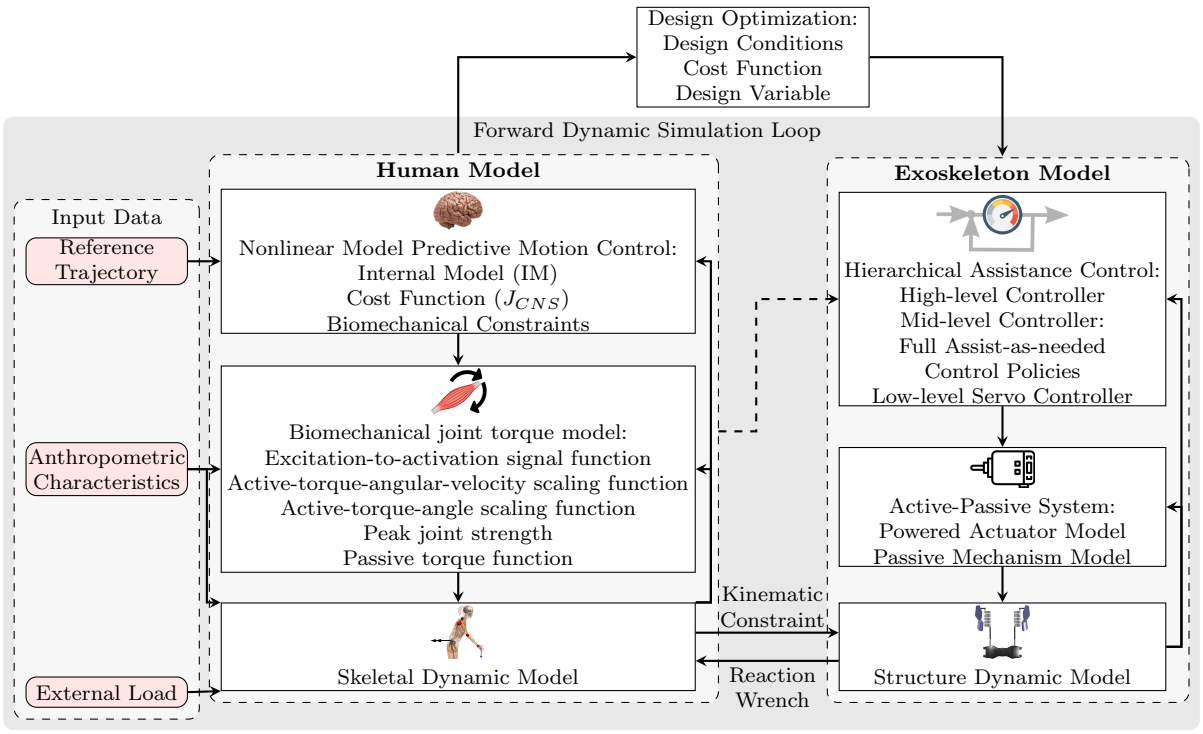


Figure 6.3: Schematic of designing, synthesizing, analyzing, and optimizing the forward scenario.

Inverse scenario

The inverse scenario benefits from computationally efficient simulation [26]. However, in the inverse simulation, the impact of the human actuation torque is ignored in the active exoskeleton control. The inverse scenario is similar to forward simulation when high assistance strength is selected. As shown in Figure 6.2, initial optimization is done with the reaction wrench variable having minimal human activation torque. This first optimization loop is constructed from the human dynamic system without an actuator dynamic model to guess the optimum assistive exoskeleton torque. The second optimization loop is constructed to design the actuator parameter. The second loop produces the output torque similar to the optimum assistive torque from the first optimization loop.

Here, the inverse scenario was used to design the custom passive mechanism. Precisely, the first loop optimized the passive torque-angle profile and the second loop optimized the mechanism parameters outlined in Equation (5.9).

Forward scenario

The forward scenario benefits from a comprehensive and similar simulation to the human-robot interaction. However, the forward simulation is time-consuming. As shown in Figure 6.3, the primary step is designing, synthesizing, analyzing, and optimizing the actuator model via the optimal controller, which acts as the CNS and contributes to the forward dynamic simulation of the human-robot model. The procedure starts with an optimal controller that follows the desired trajectory and commands the human (excitation or activation) signal or (active or joint) torque. Both the actuator command (defined by the assistance model [28]) and human input are injected into the forward dynamic simulation of the multibody system. The optimization loop evaluates the objective function and finally guesses and selects the actuator parameter.

6.2.2 Design variables

A design variable is a numerical input that can be changed during design optimization. Optimization of the actuator can be achieved by changing the properties of the passive mechanism and selecting the BLDC motor.

As mentioned in section 5.2.2 and modeled in Equation (5.8), there are two variables in the passive mechanism. First, the maximum passive joint torque (T_{max}), which is 4.578, 7.693, 9.798 Nm for the Level 1, 2, 3 springs of EVO, respectively. The second variable is the passive mechanism rotational shift (φ), which can be adjusted between -0.175 and 0.175 rad.

The addition of an actuator, although it introduces additional inertia (Table 5.4) and viscous rotator friction to the system, can provide active torque with the required actuator control system. The active torque is modeled in Equation (5.11), and the maximum possible nominal torque is provided in Table 5.3 for 9 motors. The optimization design variable is the selection of the BLDC motor from Table 5.3.

6.2.3 Design cost function

Industrial wearable exoskeletons are designed to assist and improve human performance to reduce user effort, fatigue, and, therefore, the risk of work-related MSDs [8, 14, 34, 152]. The performance metric that aims to minimize the cost function is one or a combination of the following variables.

The first variable is the human joint applied torque (Q) in Equation (5.1), which is the net torque applied to the skeletal system. This variable shows the required torque for the human motion without considering the active and passive components of the MTGs [84] in Equation (5.2).

The second variable is the human active torque (τ_a) in Equation (5.2) that only consider the passive torque that occurs when the muscles, tendons, and ligaments near the joints are strained [299]. In the MapleSim Biomechanics model, the human active torque is relevant to the active tension of the muscle.

The third variable is the human joint power. Theoretically, the quantity of energy converted in one unit of time is called power. In rotational joints, power is the product of the joint active torque (τ_a) and the angular velocity (ω).

The fourth variable is the muscles metabolic energy expenditure (MMEE). We have used the MMEE model [381] as a critical performance measure of human motion with or without wearing of the exoskeletons. The MMEE model is a general joint-space numerical model of metabolic energy expenditure obtained by combining the laws of thermodynamics and multibody system dynamics principles [381]. The equations of the energy model for zero co-contraction MTGs are as follows:

$$E = \int_0^{t_{max}} \left[\dot{h}_i + p_i dt\right] dt \tag{6.1}$$

$$\mathbf{p} = \boldsymbol{\tau}_a \boldsymbol{\omega} \tag{6.2}$$

$$\dot{\mathbf{h}} = \dot{h}_M \left|\boldsymbol{\omega}\right|_{(max)} \left|\boldsymbol{\tau}_a\right| + \dot{h}_{SL} \left|\boldsymbol{\tau}_a \boldsymbol{\omega}\right| \tag{6.3}$$

where

p	the mechanical power at the joint
$\dot{h}$	the heat rate
$\dot{h}_M$	the dimensionless heat rate for activation and maintenance, determined to be 0.054
$\dot{h}_{SL}$	the dimensionless shortening lengthening heat rate, 0.283 for positive power, and 1.423 for negative power
$\boldsymbol{\omega}_{(max)}$	the maximum angular velocity over the entire motion

In addition to the human-based design cost function, the electricity consumption of the actuator should be as low as possible. Precisely, the exoskeleton should provide the maximum portion of assistance using the passive mechanism. The actuator should have the role of adjusting or providing extra assistance whenever needed. Consequently, the electrical consumption of the actuator should be one element of the design cost function.

Table 6.1: The details of 5 conditions of human and exoskeleton systems optimization.

Symbol	Condition	Human	Exoskeleton Inactive	Design Variable				
				Passive	Active	T_{max}	φ_0	Motor
H	only human	✓						
IE	human with inactive exoskeletons	✓	✓					
FP	human with fully passive exoskeletons	✓		✓		✓	✓	
FA	human with fully active exoskeletons	✓	✓		✓			✓
AP	human with active-passive exoskeletons	✓		✓	✓	✓	✓	✓

6.2.4 Design conditions

For a fair comparison, five conditions were considered to study the effect of exoskeletons on the human body. The exoskeleton can be inactive (provide no active and passive torque), and the human should be able to carry its mass around. The EVO exoskeleton's novel passive mechanism can be activated with a tiny leveler. The five conditions in Table 6.1 are constructed by combining the sub-models in chapter 5.

6.3 Results and discussion

The optimization checks the 966 possible simulations (6 Tasks (section 6.1.1), 5 conditions (section 6.2.4), 3 maximum passive torques, 5 passive mechanism rotational shifts (section 5.2.3), and 9 motors (section 5.2.4)), and the optimum results are presented in the following.

6.3.1 Human performance analysis

According to the simulation results, only the shoulders (SFE and SAA) and the torso (TFE, TRLLF, and TRLAR) variables change in the conditions mentioned in section 6.2.4. This result is compatible with the fact that the shoulder exoskeleton directly provides an assistive torque for the shoulder joint and indirectly increases the torque of the torso joint. Thus, here, Figure 6.4 provides the results of the design cost function in section 6.2.3 for the torso joints, the right, and the left shoulder.

The average of absolute active torque of the right shoulder joint (summation of SFE and SAA) for 5 conditions and 2 groups of tasks are shown in Figure 6.4(a). When the exoskeleton is inactive, the average absolute active torque increases by approximately 1 Nm compared to not wearing the exoskeleton. This is due to the inactive exoskeleton weighing 3.461 kg, which the user should carry. By employing a passive exoskeleton with $T_{max} = 9.798$ Nm and $\varphi_0 = -0.175$ rad, the average of absolute active torque of the shoulder joint decreased by 13% and 53% for heavy object handling and arm-reaching tasks, respectively. Using motor #3 and motor #1 for arm-reaching and heavy object handling tasks reduced the average of absolute active torque by 79% and 49%, respectively. Using $T_{max} = 9.798$ Nm, $\varphi_0 = -0.175$ rad, and motor #4 for arm-reaching task as well as $T_{max} = 9.798$ Nm, $\varphi_0 = -0.175$ rad, and motor #1 for heavy object handling task, the average of absolute active torque drops by 80% and 57%, respectively.

According to Figure 6.4(b), the left shoulder has the same trend as the right side. However, the average of absolute active torque amount is lower than the right shoulder (Figure 6.4(a)) since the left arm has not contributed to lifting a heavy object in the overhead work task.

The average absolute active torque for the torso joint (Figure 6.4(c)) has not changed considerably. Using an inactive exoskeleton can decrease the average absolute active torque since it adds weight to the subject's back and helps in torso extension. If the subject bends with a larger angle, the exoskeleton's added weight would increase the required torque for torso extension. Since in a fully active and active-passive exoskeleton, the motor, housing, and shaft are added to the system near the elevation mechanism of the exoskeleton, the torso's torque increased by 1%.

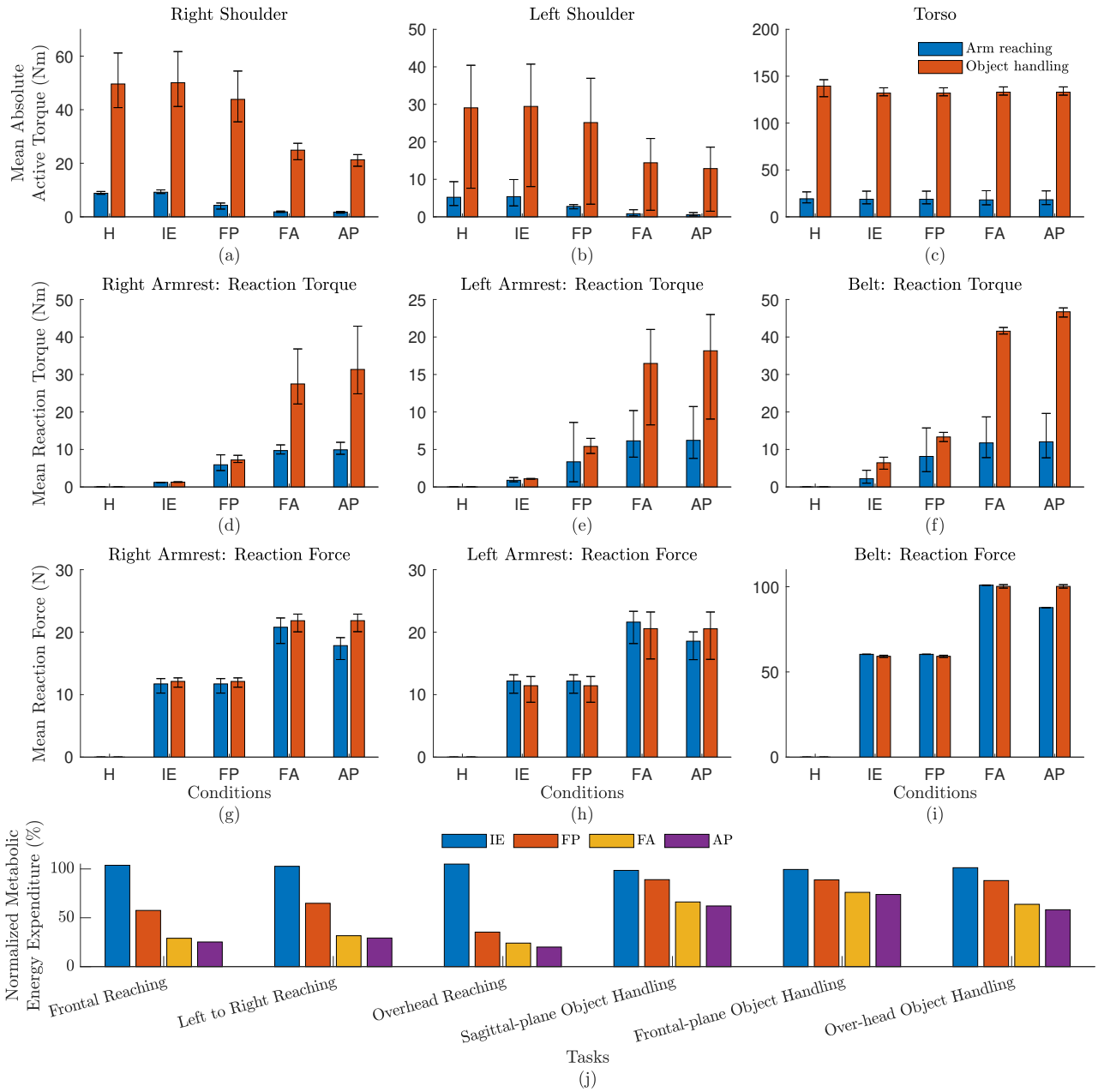


Figure 6.4: The average of absolute active torque for the right shoulder, left shoulder, and the torso (a-c). The reaction torque and the reaction force at the right armrest, left armrest, and the belt (d-i). The metabolic energy expenditure of the human wearing the exoskeleton is normalized by the human not wearing the exoskeleton for 6 tasks (j). H: human, IE: inactive exoskeleton, FP: fully-passive exoskeleton, FA: fully-active exoskeleton, AP: active-passive exoskeleton.

The reaction force and torque at the attachments of the exoskeleton to the human (belt, right, and left armrest) are shown in Figure 6.4(d) to Figure 6.4(i). According to these results, the reaction increases by providing a more assistive torque profile at the exoskeleton's elevation joint. This value should be considered for study to improve human comfort.

According to Figure 6.4(j), the fully active and active-passive exoskeleton can lead to a metabolic energy expenditure drop. The metabolic energy expenditure decreases more for free motion and less for heavy object manipulation. One of the reasons behind this is that the motors are weak or limited by the maximum motor torque as seen in Equation (5.16). The other reason is that the active assistance augments power with the

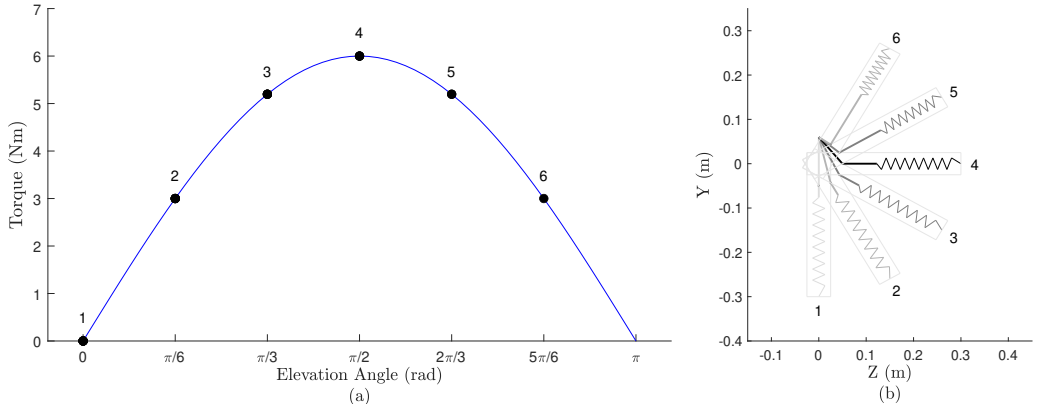


Figure 6.5: (a) The custom passive mechanism torque-angle profile for the stretched arm motion designed by the first loop of the inverse scenario. (b) The depiction of the custom mechanism posture changes (without the attachment link). The 1-6 postures in (b) are marked with relevant generated passive torque in (a).

strength variable according to Equation (5.12).

6.3.2 Motor selection

According to the optimization results, three motors from Table 5.3 are the design candidates. Motor #1, which can provide the highest torque of 48 Nm, is suitable for heavy object handling assistance with or without the passive mechanism. The drawback of motor #1 is the high gear ratio and rotator inertia, which is not suitable for back-driving the motor while the motor is off.

Motor #3 is another suitable actuator that provides the highest assistance for the arm-reaching task without adding as much back-driving difficulty as motor #1. In the case of a passive mechanism within the active component, motor #3 can be replaced with motor #4, which has lower rotator inertia but the same gear ratio. The other positive point of motor #4 over motor #3 is its lighter body which is included in the modeling, simulation, and optimization section. The motor weight can be lighter when a passive mechanism is incorporated since a portion of the assistance torque is provided by the passive mechanism. It is unnecessary to use a heavy motor with the highest amount of torque.

As shown in Figure 2.2, the BLDC motor should connect to the computer using an adaptor. For safety, an emergency shutdown switch may be handed to the user and should be located between the motor and power supply.

6.3.3 Exoskeleton actuation comparison

According to the optimization outcomes, the order of the highest to lowest assistance is provided by the following systems: active-passive, fully active, fully passive, and inactive exoskeletons. The fully active can provide the same assistance as the active-passive exoskeleton if a stronger motor was available, but it would reduce the back drivability of the wearable robots. Sometimes, a stronger motor is unavailable with the desired back drivability, and consequently, the wearable robot is weaker than expected, requiring additional assistance. This fact leads to the user requiring crutches with a lower-limb exoskeleton or a technician group carrying the lower-limb exoskeleton worn by the patient [155].

The active-passive exoskeleton has advantages over a fully active exoskeleton from a weight perspective. The passive mechanism is 0.850 kg (Table 5.1) and can provide 9.798 Nm (section 5.2.3). Its TWR is, therefore, 11.52 Nm/kg. At the same time, motor #3 can provide 15 Nm when its weight is only 0.820 kg and requires, for example, a 2.4 kg battery to operate as used in the Technaid Exo-H3 exoskeleton. Consequently, motor #3 has a 4.65 Nm/kg battery TWR. Note that as the TWR increases, the portability and user comfort improve proportionally. Thus using the passive mechanism has advantages over the actuator, but it is impossible to

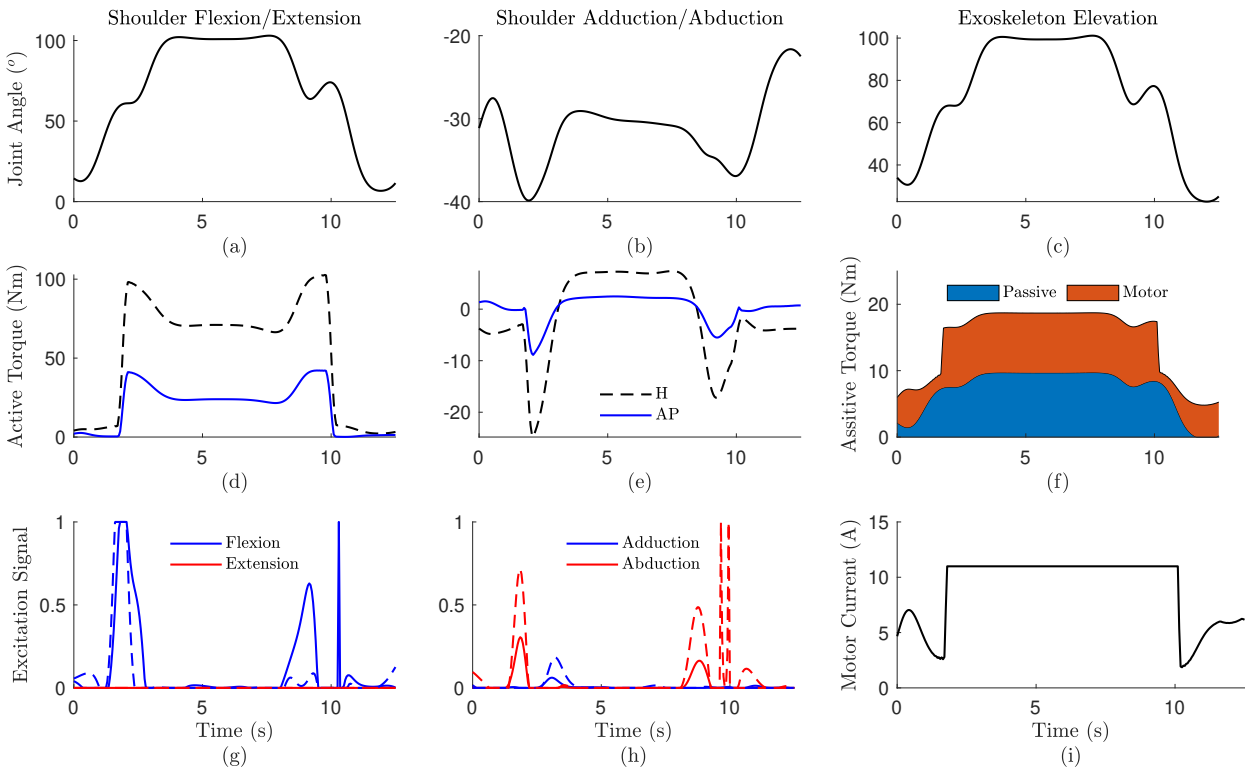


Figure 6.6: The joint angle changes of the right SFE, SAA, and the exoskeleton's elevation (a-c). The activation torque of the right shoulder for the human-only condition (dashed) and the active-passive exoskeleton condition (solid) (d and e). The exoskeleton assistive torque of the passive mechanism and motor #4 (f). The excitation signal of the mentioned joint (g and h). The motor consumption current versus time is shown at (i).

provide the torque profile required as a function of time within the passive mechanism [26, 34]. Using active and passive assistance is prescribed in this condition as it benefits from a higher TWR of passive mechanism and flexible torque profile of actuator. This conclusion is similar to the literature results, which describe the contribution of the passive mechanisms to lowering the size, weight, and required active torque, advancing the portability [14, 129, 147].

From an energy consumption viewpoint, a weaker motor lowers the energy required to drive motors in Table 5.3. Using the passive mechanism along with the motor in the active-passive exoskeleton helps reduce the required electrical energy. The lower required electrical energy leads to a more lightweight energy storage unit (battery) in the portable exoskeleton [32, 134]. Alternatively, using the same battery, the working duration of the active-passive exoskeleton would be higher than the fully active exoskeleton.

To recap, the AAN control requires minimal motion from the human to estimate the contributed torque and the exoskeleton strengthens the human motion. Moreover, when the human motion is small, and the contributed torque is low according to the AAN, the exoskeleton's motor should not activate and should not restrict the motion (i.e. low stiffness). In other words, there should be a direct drive with small inertia and stiffness, without a large gear ratio. When the gear ratio is low, the produced torque from electrical motors is low. Thus, the fully active exoskeleton can produce limited torque because of the direct drive. However, the added passive mechanism has no inertia and stiffness that resists human motion, while providing torque in addition to the motor. Thus, the torque provided by passive mechanism works in harmony with the active motor torque.

6.3.4 Custom passive mechanism

The passive torque-angle profile for a general stretched arm motion (without heavy object manipulation) is shown in Figure 6.5(a). To design this profile, the first loop of the inverse scenario (section 6.2.1) was used. For the

mentioned assistive torque-angle profile, specific passive mechanism variables (Equation (5.10)) are necessary. Using the second loop of the inverse scenario (section 6.2.1), the spring stiffness proportional constant (k) was determined to be 2000 N/m and the initial length of the spring (L_0) was 0.25 m. The distance between the wrapping point and base point ($|\mathbf{P}_W - \mathbf{P}_B|$) was 0.25 m. The attachment point is located from the elevation joint at the π rad direction of the gravity (vertical line). When the elevation angle is 0, the distance between the attachment point and the wrapping point ($|\mathbf{P}_A - \mathbf{P}_W|$) was 0.06 m. The passive elevation joint mechanism at different angles is shown in Figure 6.5(b).

6.3.5 Simulation example

As an example, a 10-second movement of an overhead heavy object (100 N weight) manipulation task was used with and without the active-passive exoskeleton condition, and the result is shown in Figure 6.6. The joint angle of the right SFE and SAA as well as the exoskeleton elevation, is shown in Figure 6.6(a-c). The relevant active torque decreased using the exoskeleton according to Figure 6.6(d-e). As shown in Figure 6.6(f), the passive torque of the exoskeleton is a function of joint angle [26], and the active or motor torque is a function of time. The motor torque provides extra assistance whenever needed, which was expected [26]. The excitation signal for positive and negative directions of the MTGs is shown in Figure 6.6(g-h). The level of the excitation signal decreases when using the exoskeleton, i.e. less effort is required from the user. The motor's current is displayed in Figure 6.6(i), which indicates that the motor is not overloaded according to the data on motor #4 in Table 5.3.

6.4 Concluding remarks

This research proposes a dynamic synthesis of passive and active assistive shoulder exoskeletons by incorporating a model with six components: the MapleSim Biomechanics upper-body dynamic model, an optimal controller, the exoskeleton rigid body, a passive mechanism, a powered actuator, and an assistance model. The six practical tasks for human motion used were frontal reaching, left to right reaching, overhead reaching, sagittal-plane object handling, frontal-plane object handling, and over-head object handling. The design optimization aimed to reduce the human-based and the exoskeleton-based criteria. According to the results, the active-passive exoskeleton provided more assistance to the human than the fully active and fully passive exoskeleton, while the weight of the exoskeleton did increase compared to the fully active exoskeleton. Additionally, the optimization offered a back-drivable motor with lower inertia, gear ratio, and viscous damping coefficients instead of a powerful motor that is not back-drivable. The active-passive exoskeleton would consume less electricity than the fully active exoskeleton, which helps design a lighter portable exoskeleton. In total, the optimized active-passive exoskeletons create lighter and smaller exoskeletons that lessen the user's muscular activation torque for the tasks being studied.

We expect future wearable exoskeletons to adopt components that incorporate passive and active assistance systems. Future studies should focus on the control design and propose a practical algorithm to control such a complicated but efficient actuation system as mentioned by relevant studies [28]. In addition, researchers may focus on energy harvesting by using the active motor to utilize positive power [32]. The last point to consider is safety and human comfort, which should be thoroughly examined before producing a commercial wearable exoskeleton.

Chapter 7

Machine Learning Interpretation of Myoelectric Signals (MuscleNET)

In this work, firstly, the configuration of the machine learning models is introduced in section 7.1. Secondly, the output and input signals of the machine learning model are evaluated in sections 7.2 and 7.3. Finally, the training results of models are discussed and compared. Note that the data collection steps are described in section 4.1.

7.1 Machine learning models

This section describes in detail the configuration of four different machine learning models: ANN, RNN, CNN, and RCNN. The initial architecture of machine learning models was adopted from prior research [189, 211, 213–215, 382]. The number of hidden layers, neurons in each hidden layer, feedback delay, input delay, convolution filtering size, pooling size, and strides were selected by trial and error (Figure 7.1) to have the highest regression accuracy. The CNN and RCNN models with delayed kinematics have a novel configuration, which are introduced in section 7.3.1.

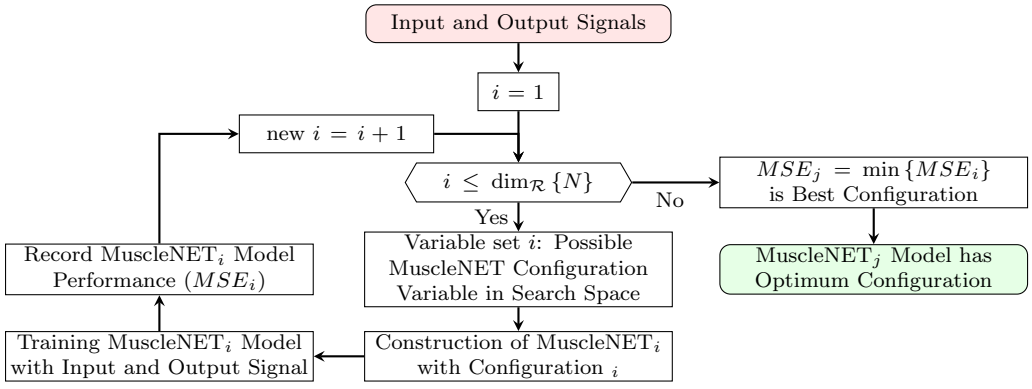


Figure 7.1: Schematic of machine learning models configuration optimization loop.

The machine learning models' output are described in section 7.2. The outputs were normalized by the maximum value for each subject. The machine learning models' input are described in section 7.3.

The performance was reported with the computation of regression or MSE using Equations (7.1) or (7.2).

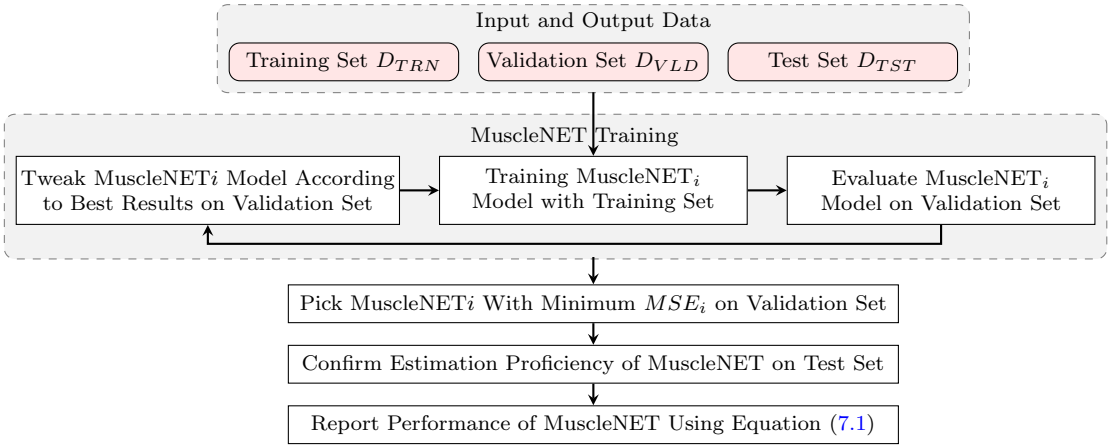


Figure 7.2: Schematic of training models with training, validation, and testing sets of the input and output signal.

$$R = \left[\frac{D_{TRN}}{\sum D_i} \left(1 - \sqrt{MSE_{TRN}}\right) + \frac{D_{VLD}}{\sum D_i} \left(1 - \sqrt{MSE_{VLD}}\right) + \frac{D_{TST}}{\sum D_i} \left(1 - \sqrt{MSE_{TST}}\right) \right] \times 100 \tag{7.1}$$

$$MSE = \frac{1}{N} \sum_{i=1}^{M} (X_i - T_i)^2 \tag{7.2}$$

where

D_{TRN}	the training data volume
D_{VLD}	the validation data volume
D_{TST}	the testing data volume
T	the processed target values (section 7.2) for training the machine learning model
X	the output values of the trained machine learning model
N	the total number of datasets points

7.1.1 Dividing ratio and data

All 17 subjects contributed data for training, validating, and testing. Specifically, 1,560,000 pairs of inputs and outputs ($1,560,000 = 17$ subjects $\times$ 60 seconds $\times$ 1500 Hz sampling rate) were used for training, validating, and testing the models.

To evaluate the amount of experimental data required for the machine learning model, a complete estimation was done on the test subject who did not participate in the training and validation of the model (section 7.4.1). In situations where the model had more than 85% regression accuracy, the capability of tuning the general model to estimate a test subject's output was demonstrated.

The regression proficiency of the machine learning model depends on the training algorithm (Figure 7.2). Dividing data for training, validation, and testing is controversial and depends on the nature of the datasets, the number of variables, and the machine learning model architecture. There is no strict rule for dividing, and therefore each method should be evaluated separately as intrinsic characteristics vary (model architecture, data nature, and model purpose). Using more data sets for training and less data for validation and testing can increase the risk of overfitting [362]. We have added a discussion of this issue in section 7.4.1.

Table 7.1: Detail of ANN model configuration.

#	Inputs			Configuration				Outputs	
	Filtered sEMG	Raw sEMG	Delayed Kinematic	Size	Number of Hidden Layers	Number of Neurons in Each Hidden Layer	Activation Function in Hidden Layer	Activation Function in Output Layer	Size	Number of Neurons
1	✓			11×1	1	55	Tan-sigmoid	Tan-sigmoid	1×1	716
2		✓		11×1	2	55	Tan-sigmoid	Tan-sigmoid	1×1	3796
3	✓		✓	14×1	1	56	Tan-sigmoid	Tan-sigmoid	1×1	897
4		✓	✓	14×1	2	56	Tan-sigmoid	Tan-sigmoid	1×1	4089

7.1.2 Artificial neural network model

One of the first and simplest machine learning models for modeling a muscle is the forward artificial learning model. According to Nasr et al. [382], this multilayer model should be wide and shallow to map the sEMG to activation signal (σ) with minimal error and high accuracy. Their recommended architecture was adopted, and different configurations (number of hidden layers and neurons in each hidden layer) were tested on the data sets to obtain the optimized configuration (for maximum regression accuracy). The optimized configuration had the highest regression accuracy in the same training condition and is described in the following.

If the sEMG signal was raw, the number of hidden layers was set at 2, and if the sEMG was filtered, it was set at 1. One additional layer was added to the Nasr et al. [382]'s model for the described approach as more layers filtered the raw signal. Since the delayed kinematics provided valuable data for the model, the number of neurons in each layer was set to 4 times the number of input signals (for example, for 11 sEMG signals and 3 delayed kinematic signals, the number of neurons is 4 times 14 or 56). When the inputs were purely sEMG signals, the number of neurons in each layer was set at 5 times the number of input signals (for example, for 11 sEMG signals, the number of neurons is 5 times 11 or 55). The ANN model configuration details are summarized in Table 7.1.

The training method was a Levenberg-Marquardt backpropagation. The maximum epoch was set to 2000. The maximum elapsed time was set to 6 hours.

7.1.3 Recurrent neural network model

Since the muscle dynamics and joint motion rely on motion history, we hypothesized that RNNs might have better accuracy in muscle modeling. This model has feedback from the output to the inputs. After training the NARX networks, the output time series was predicted with the past output values (the feedback input) and the external input time series (sEMG signal and delayed kinematics).

The configuration of the RNN model plays an essential role in modeling performance and regression accuracy. The variables of the RNN configuration contain n_u, n_y, the number of layers (n_l), the number of nodes in each layer (n_n), and the activation functions. This configuration should be tuned for the nature of inputs and outputs signals. The general architecture was adopted from Chen et al. [189], and different configurations (number of hidden layers, neurons in each hidden layer, feedback delays, and input delays) were tested using the data sets to obtain the optimized configuration that had the highest regression accuracy for the same training condition.

Initial configuration

The number of hidden layers was 2. The number of neurons was 3 times the number of input signals when the inputs had delayed kinematic signals and 4 times the input signals when the inputs did not consist of the delayed kinematics. The feedback delays were then set to the last 7 signal values. Since different input delays

Table 7.2: Detail of RNN model configuration.

#	Inputs			Configuration						Outputs	
	Filtered sEMG	Raw sEMG	Delayed Kinematic	Size	Number of Hidden Layers	Number of Neurons in Each Hidden Layer	Input Delays	Feedback Delay	Activation Function in Hidden Layer	Activation Function in Output Layer	Size	Number of Neurons
5	✓			11×1	2	44	1	7	Tan-sigmoid	Tan-sigmoid	1×1	2024
6		✓		11×1	2	44	1	7	Tan-sigmoid	Tan-sigmoid	1×1	2024
7	✓	✓		14×1	2	42	1	7	Tan-sigmoid	Tan-sigmoid	1×1	1848
8		✓	✓	14×1	2	42	1	7	Tan-sigmoid	Tan-sigmoid	1×1	1848

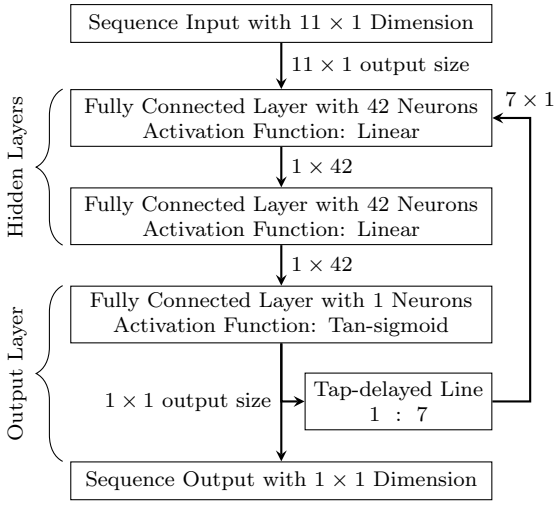


Figure 7.3: Schematic of the RNN MuscleNET for combined input of 11 filtered sEMG and three delayed kinematic signals.

did not impact accuracy, we set the input delays to zero or used the current input value. The details of the RNN MuscleNET configuration are summarized in Table 7.2, and an example of RNN MuscleNET for 11 filtered sEMG signals and 3 delayed kinematic signals is depicted in Figure 7.3. The maximum epoch training method and the maximum elapsed time were the same as for the ANN.

Configuration optimization

This research tested 90 different RNN configurations, assessed the estimation accuracy, and adopted the superior configuration. The minimum variables, maximum variables, and step changes are presented in Table 7.3. Since the output signals are normalized by the maximum signal value and are always between 0 and 1, we have employed a symmetric saturating linear transfer function for the output activation function (Figure 7.3).

The Levenberg–Marquardt backpropagation algorithm was used to train the RNN model. The backpropagation algorithm calculates the Jacobian of the training performance regarding the bias and weight variables of the RNN network. The initial adaptive mu, mu decrease factor, mu increase factor, and maximum mu values were 0.001, 0.1, 10, and 1e10. The epoch was set to 1000, and the training time was limited to less than one hour. The training was terminated when the epoch or the time reached the maximum values. Although the network is not deep, the model was configured to be closed-loop instead of open-loop for the training purposes of time-series data, which requires ten times the amount of training time

We found the training time selection relevant to the computation capability of the PC to be reasonable.

Table 7.3: The optimization variables of RNN configuration (90 different configurations as a full combination of all variables).

Variable (Symbol)	Minimum	Maximum	Step
number of input signal previous values (n_u)	1	5	2
number of output signal previous values (n_y)	1	5	2
number of hidden layers (n_l)	1	2	1
number of nodes in each hidden layer (n_n)	10	50	10

The training performance was assessed using the MSE function in Equation (7.2). The linear regression R in Equation (7.1) of targets relative to outputs was used for reporting and evaluating the regression performance.

7.1.4 Convolutional neural network model

We initially adopted the CNN and RCNN configurations from prior work [209, 211, 215] and tested different configurations (number of convolutional layers, convolution filtering size, pooling size, strides, neurons in FC layer, activation function) using the data sets to obtain the following configuration, which had the highest regression accuracy for the same training condition. The configurations of CNN and RCNN MuscleNET with a delayed kinematic signal are novel in terms of parallel structures for delayed kinematic signal input.

Since there were two different input signals (sEMG signals and delayed kinematic signals), the CNN models' configurations were categorized into 4 different shapes based on the input's conditions. First, we describe the CNN generic configuration and then the different configurations in the following paragraph.

Generally, the typical configuration consists of an input group, the convolution groups, and an output group. First, the input group is a sequence input layer that gets an image from the input signals as described in section 7.1.4. Secondly, the first layer of convolution groups is a folding sequence layer, and the last one is an unfolding sequence layer. Finally, the output group consists of 4 layers: the flatten layer, a FC layer with an output size of 50 neurons, a FC layer with an output size of 1 neuron, and finally, the regression layer.

The training method was Adam (adaptive moment estimation) optimizer with a unit threshold for the gradient. The learning rate used for training was set to 0.001. The maximum epoch was set to 500. The mini-batch size (a subset of the training set) for each training iteration was set to 2000.

The convolution groups were unique for the different input signals (with or without delayed kinematics and raw or filtered sEMG signals). In sections 7.1.4 and 7.1.4, the CNN models are explained in detail for each input signal combination.

Input image

The number of signals made the image's height (for example, for 11 sEMG signals and 3 delayed kinematic signals, the image's height was 14). The image's width is relevant to the history of sEMG signals that we want to filter by CNN or RCNN. Since the maximum human motion frequency is roughly 6 Hz [383], we recommend using a maximum of 7 Hz. We did not propose to use less than 5 Hz since the number of data points increases for the CNN or RCNN, slowing model processing. For data with a 1500 Hz sample rate, we used 250 points for the image's width, which is 6 Hz of the data. In summary, the input image size was 11 by 250 for the sEMG signals input and was 14 by 250 for inputs consisting of sEMG signals plus delayed kinematics (Figure 7.4). The current data is located on the image's right, and the 249 previous data is located at the left of the image. The sequential input layer handled importing this 1D image. It is noteworthy that signals should not have the same rate as the sampling rate when making the images from the signals. Since the model's delay might be around 0.05 seconds or 20 Hz, we used a slower rate of making images. We used 0.1 seconds for making the images, and at each time, must use the current signals and 249 of the previous signals to make the image, then wait for 0.1 sec to make the following image. This way, we considered the model's delay and system to use the model for model-based control of biomechatronic devices. Using an image composed of sEMG signals and delayed kinematic signals is a novel approach.

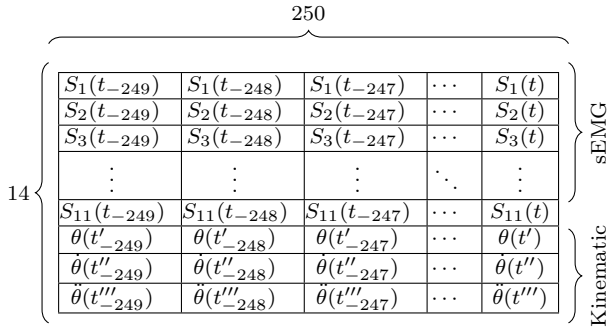


Figure 7.4: A sample of the input image made from sEMG signals and delayed kinematic signals used as the input of the CNN or RCNN configuration of MuscleNET. The delayed shoulder elevation joint angle, velocity, and acceleration are $\theta(t')$, $\dot{\theta}(t'')$, and $\ddot{\theta}(t''')$, respectively (t', t'', and t''' are the delayed times).

Without delayed kinematic signals

When only the raw sEMG signals were the input image's constructor, there were 8 layers in the convolutional groups, and they were taken serially (Figure 7.5). All groups had a 2-D convolutional layer that was applied to slide convolutional filters with a filter size of height 1 and width 3. The output of this layer had the same size as the input when the stride equaled 1. The step size for traversing the input was 1 vertically and was 1 horizontally. Groups 1, 2, 7, and 8 had 16 neurons, and groups 3 to 6 had 64 neurons in the convolutional layer (the number of channels or feature maps). All 8 groups had a batch normalization layer that normalized each input channel across a mini-batch size. We used this batch normalization layer to accelerate the networks' learning and reduce the network initialization sensitivity. Moreover, all 8 groups had a rectified linear unit (ReLU) layer that implemented a threshold operation to each input element, where any negative value was set to zero. Only the first 5 groups had average pooling layers for down-sampling by dividing the input into rectangular pooling areas and calculating each area's average values. The pooling region's dimensions had a height of 1 and a width of 3. The vertical step size was 1, and the horizontal step size was 3. The vertical unit size for the pooling layer's stride was utilized because the sEMG signals should be filtered separately.

All in all, the first 5 groups had a 2-D convolution layer, a batch normalization layer, a ReLU layer, and a 2-D average pooling layer. The 6 to 8 groups had a 2-D convolution layer, a batch normalization layer, and a ReLU layer.

For input images constructed with the filtered sEMG signals, group 7 was removed, and group 6 was connected to group 8. The convolutional neural network configuration for raw sEMG signal and without the delayed kinematics signal was quite similar to the model proposed by Ameri et al. [213], which allowed for a direct comparison with our new model.

With delayed kinematic signals

Since the origin of the delayed kinematic and sEMG signals are different, they first should be filtered with different convolutional layers separately. To this end, we innovated a parallel structure with two separate bridges. The first bridge handles filtering the sEMG signals, and the second one relates to delayed kinematic usage. Finally, both bridges intersect, and all data was used for modeling the mapping of the input (sEMG and delayed kinematic signals) to the output signals. This novel configuration for separating different signals from the image is shown in Figure 7.6.

The left bridge, dedicated to filtering the sEMG signals, had 4 groups for raw sEMG signals. The first group had a transposed 2-D convolution layer with an additional feature of cropping the image. This new feature shaved the input image (that is, the combination of sEMG and delayed kinematic signals). The transposed 2-D convolution layer's output size was reduced by cropping from the bottom of the input by 3 units since the last 3 pixels are relevant to the delayed kinematic signals. The transposed 2-D convolution layer and 2-D convolutional layers at groups 2 and 3 had a filter size of the height of 1 and the width of 1. All convolutional layers had 40

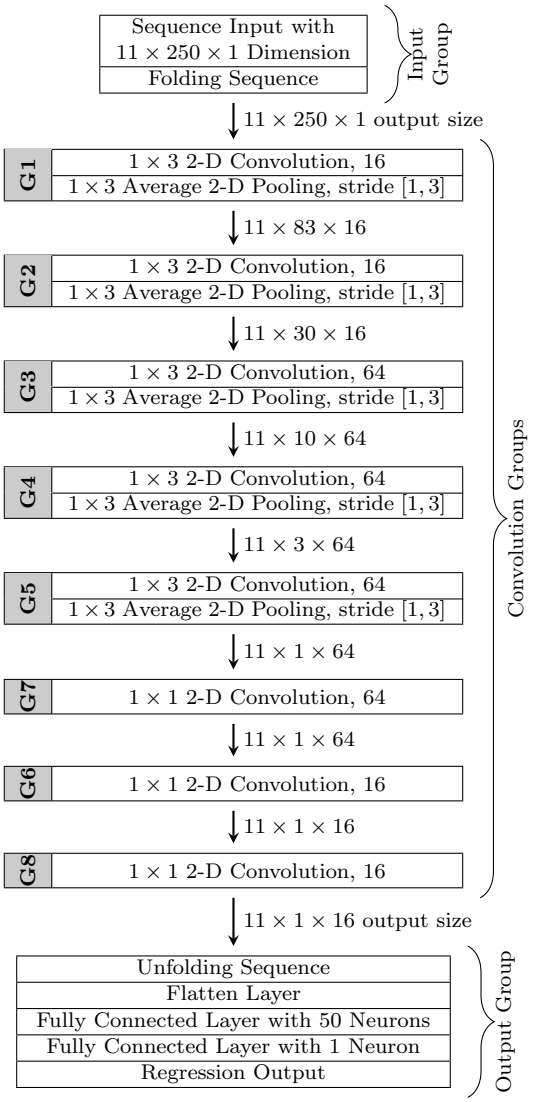


Figure 7.5: Schematic of the convolutional MuscleNET for raw sEMG signal input only, including convolutional layers, average pooling layers, and FC layer. The batch normalization and the rectified linear unit after the convolutional layers in each group have not been shown for simplicity. The convolutional layers and the average pooling layers have the same padding.

neurons each. The step size for traversing the input was 1 vertically and was 1 horizontally for all convolutional layers. The first two groups had a ReLU layer and average pooling layers for down-sampling. Both pooling layers had a pooling region's dimensions with a height of 1 and a width of 25. However, the first one's stride at the first layer was 10, and the second one at the second layer was 5 in the horizontal direction. The vertical unit size for stride of both pooling layers arose because the sEMG signals should be filtered separately. The last group only had a 2-D convolutional layer with different filter sizes of the height of 11 (since there are 11 sEMG signals) and the width of 5 with the same stride. As shown in Figure 7.6, since the output size of group 3 was 11x5x40, it is evident that the 250 pixels were converted to 5 pixels. We have tried to have these 5 pixels for pattern detection purposes. Since the sEMG signal was not converted to one signal and had 5 pixels, it can provide more timing information.

For input images constructed with the filtered sEMG signals, group 3 was removed, and group 2 connected to group 4.

The right bridge, which is dedicated to acquiring the delayed kinematic signals, had 2 layers. The first layer had a transposed 2-D convolution layer with a cropping feature from the top of the input image by 11 units

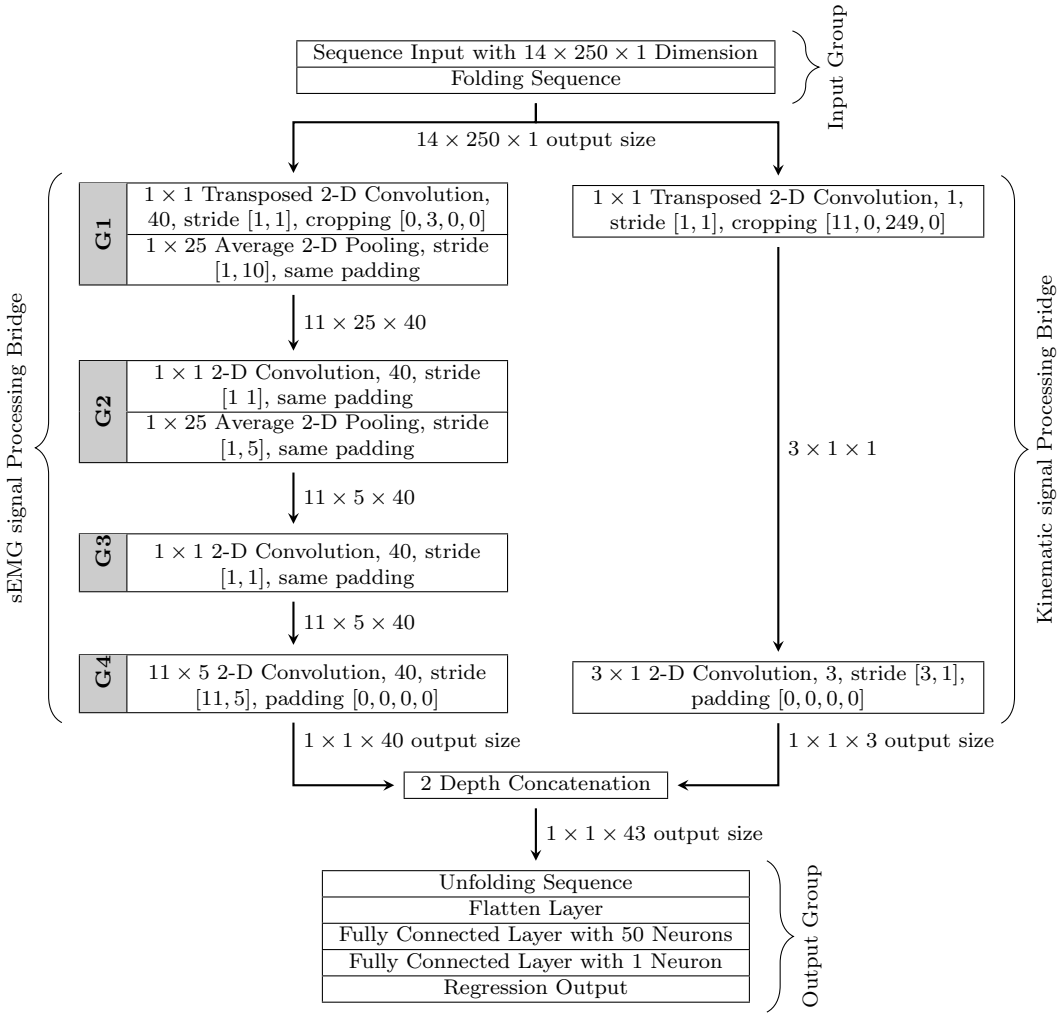


Figure 7.6: Schematic of the convolutional MuscleNET for combined input of raw sEMG and delayed kinematic signals, including convolutional layers, average pooling layers, and FC layer. The batch normalization and the rectified linear unit after the convolutional layers in groups 2 and 3 have not been shown for simplicity.

(since the first 11 pixels are relevant to the sEMG signals) and from the left of the input image by 249 units (since the last column is the recent signals). Precisely, the cropping converted the 14 by 250 image size to a 3 by 1 image size. The transposed 2-D convolution layer had a filter size of the height of 1 and the width of 1 with 1 neuron. The second layer had a regular 2-D convolutional layer with a filter size of the height of 3 and the width of 1.

The signals from two bridges intersected with a depth concatenation layer. The depth concatenation layer received inputs with the same height of 1 and width of 1 and concatenated them alongside the depth dimension. In this way, both signals of sEMG and delayed kinematics were stacked to an image with 1 width, 1 height, and 43 depth.

7.1.5 Recurrent convolutional neural network model

The RCNN's configuration approximates the CNN; the only difference is in the output group's layers. The FC layer with an output size of 50 neurons in the CNN was changed to a LSTM layer with 50 hidden units. Additionally, the output of the LSTM was the last time step of the sequence in the RCNN. The activation function to update the cell and hidden state was set to be the hyperbolic tangent function, and the activation function to apply to the gates was the sigmoid.

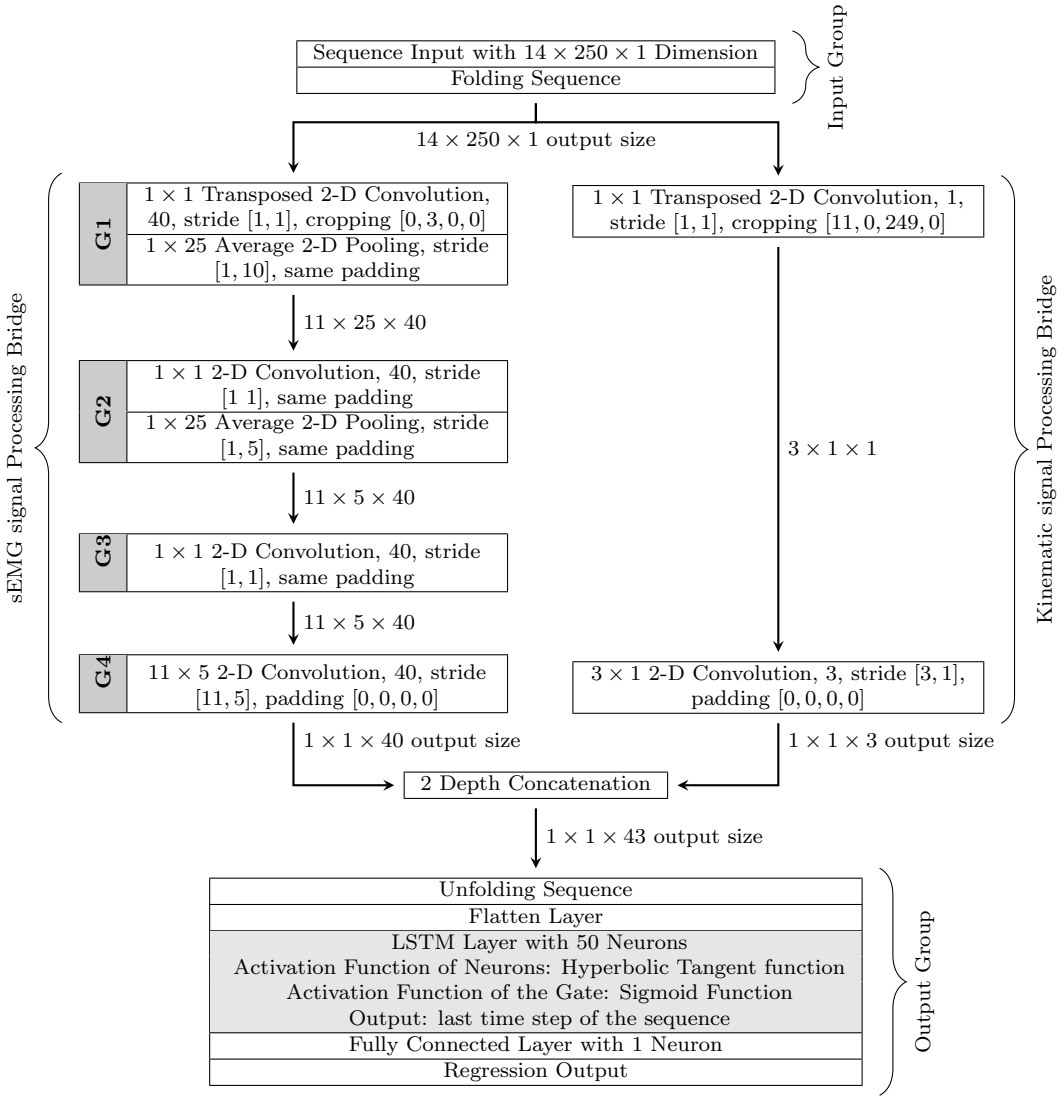


Figure 7.7: Schematic of the RCNN MuscleNET for combined input of raw sEMG and delayed kinematic signals, including convolutional layers, average pooling layers, and a LSTM layer. The batch normalization and the rectified linear unit after the convolutional layers in groups 2 and 3 have not been shown for simplicity.

The schematic of the RCNN MuscleNET for the combined input of raw sEMG and delayed kinematic signals is shown in Figure 7.7. The general output group were introduced in section 7.2. The details of the sEMG signal processing and kinematic signal processing were presented in section 7.3. The input image was introduced in section 7.1.4.

7.2 Output signals

Here, the output signals of the machine learning model are evaluated. First, the biomechanical variables for the SFE joint are assessed in section 7.2.1. Second, multiple joints (SFE and EFE) data are evaluated for the outputs of the machine learning model in section 7.2.2. Third, control-oriented (the robot's variable) signals are used for the output of the machine learning model in section 7.2.3. Finally, the future variables are used, and the prediction ability of the mapping is assessed in section 7.2.4.

7.2.1 Biomechanical variables

Processing the biomechanical variables (joint angle, velocity, acceleration, torque, and activation signal) for SFE joint out of raw kinematic data is described in section 4.1.4. These variables were assessed as an output of the machine learning model, and the results are in section 7.2.1 and Table 7.4.

7.2.2 Biarticular output

So far, the monoarticular machine learning model estimates only one activation signal, independent of other joint variables. In contrast, the RNN outputs can be multiple joints (SFE and EFE) activation signals of MTGs simultaneously. Hence, the machine learning model estimates multiple dependent MTG activation signal simultaneously and uses the estimated variables as feedback for the RNN configuration. For this end, the RNN consisted of one hidden layer with 40 hidden neurons, five input signal former values, and five output signal former values.

7.2.3 Control-oriented output

One of the model's objectives for high-level controlling biomechatronic devices is to provide a less time-consuming mapping. From a control viewpoint, it is better to have minimal kinematic and dynamic computation in real-time controlling. Thus, providing a control-oriented model instead of a human-oriented model plays a vital role in decreasing the computation cost. These extra kinematic and dynamic computations are due to the kinematic structure, joints, links, and constraints of biomechatronic devices, leading to biomechatronic joints that are not co-axial with human joint axes. Since the human-mechatronic devices may have closed kinematic chains in the multibody system model, we must solve DAEs, which requires time-consuming optimization loops.

The conversion from human joint variables to robot control command should not included in assistive robots' real-time high-level control. However, the conversion should occur a prior via an offline simulation to prepare the control-oriented model's training data. The offline simulation can be obtained by optimizing the robot's assistive torque through the multibody system dynamic model simulation (section 4.3).

The control-oriented output signals of machine learning model were prepared through following steps.

i. the recorded human kinematic data is used to find the control-oriented joint angles, velocity, and acceleration by using Equations (5.19-5.21);

ii. the optimized control-oriented torque (assistive torque) is calculated using an optimization loop (more detail in section 6.2 as well as Figure 6.3)

7.2.4 Future control-oriented output

The electromechanical negative delay phenomenon [219] can compensate for the control system computation or actuation delay. We parameterized this phenomenon to predict the joint angle or motion intention using sEMG signals and a machine learning method. This section describes the method of shifting the training output data and provides the prediction horizon assessment.

The predictive output signal can be obtained through two possible scenarios: (A) the future joint angle in the $t + \Delta t$ time or (B) the future joint acceleration using the Euler series. Both possible outputs are defined in the following paragraph.

For the current sEMG set in the time of t, the future joint angle $\theta_t + \Delta t$ in the time of $t + \Delta t$ was calculated by straightforward 1-D data extrapolation.

Similarly, since human motion (acceleration) is actuated by muscle contractions measured by the sEMG technique, the reference joint or task space acceleration can also be calculated from sEMG. The predicted acceleration $\ddot{\theta}'_h$ can be calculated by the Euler series for $t + \Delta t$ time in Equation (7.3) by using the current joint angle θ_t and joint velocity $\dot{\theta}_t$.

$$\theta_{t+\Delta t} \approx \theta_t + \dot{\theta}_t \Delta t + \frac{1}{2!}\ddot{\theta}'_h \Delta t^2 \tag{7.3}$$

The prediction horizon Δt plays an essential role in the accuracy of motion prediction. Therefore, we studied different values for the prediction horizon along with the possibility of the future joint angle $\theta_{t+\Delta t}$ or the predicted acceleration $\ddot{\theta}'_h$ as an output of the machine learning model.

The assistive torque was shifted with $t + \Delta t$ as the prediction horizon to obtain the future assistive torque $\tau_t + \Delta t$.

7.3 Input signals

Here, the inputs of the machine learning model are discussed. These inputs are delayed kinematic signals (section 7.3.1), raw or filtered sEMG signals (section 7.3.2), optimized filtered sEMG signals (section 7.3.3), and interrupted sEMG signals (section 7.3.4).

7.3.1 Delayed kinematics

Since the muscle model relies on the kinematic values as well as the sEMG signal, the kinematic data was used as one of the inputs to the machine learning model. No prior research has combined the kinematic signals with the sEMG signals.

Practically acquiring the kinematic values such as angle, velocity, and acceleration has a delay. Thus, we considered a specific amount of delay for these signals. The delay itself has two primary sources: electrical delay and computation delay. The electrical delay relates to the speed of connecting, sampling rate, and computer delay. The computation delay relates to the calculation method delay. For example, the robot's joint angle can be measured through rotary encoders or a rotary potentiometer, and commonly, a low-pass filter with a computation delay of 0.025 seconds is used for filtering joint angle noises. The joint velocity and acceleration are estimable via the first and second derivatives using the current joint angle values, which have computation delays. In other words, the velocity value is not real-time and is based on the previous angle value.

Thus, we selected 0.025 seconds for the electrical delay modeling, and 0.025 s, 0.05 s, and 0.075 seconds for the computation delay of the joint angle, velocity, and acceleration signal. In total, the joint angle, velocity, and acceleration had delays of 0.05 seconds (0.025 electrical + 0.025 computation delay), 0.075 seconds (0.025 electrical + 0.05 computation delay), and 0.1 seconds (0.025 electrical + 0.075 computation delay), respectively, from the SFE joint angle may be one set of inputs for the machine learning models. In the biomechatronic control application, the mentioned delay is automatically applied to the signal; thus, it is unnecessary to add additional delay.

The assumed total delay is more than a typical experimental delay because we wanted to analyze the signals in more adverse conditions. The delay for biomechanical simulation is not necessary because real-time performance or processing is not required.

The delayed joint angle, velocity, and acceleration are $\theta\,(t')$, $\dot{\theta}\,(t'')$, and $\ddot{\theta}\,(t''')$, respectively.

7.3.2 Filtered sEMG signals

In this research, the inputs of the machine learning model are sEMG signals. For evaluating the inputs of the machine learning model, raw and filter sEMG signals were tested and compared. The initial filtering method are introduced in section 4.1.5.

7.3.3 Optimizing sEMG filtering steps

Developing a machine learning muscle model that maps sEMG signals to biomechanical variables is critically dependent on signal quality. By optimizing the structure and cut-off frequencies of the sEMG signal processing algorithm, the machine learning muscle model accuracy can be maximized. This study proposes a novel filtering-and-mapping evaluation algorithm, which considers different topologies and cut-off frequencies for the filtering steps, for maximizing the accuracy of sEMG mapping to the output. The algorithm is inspired by training methods for CNN: finding the minimum error for filtering (convolutional layers) and mapping (FC layers).

Firstly, the ANN model (section 7.1.2) with wide and shallow layers has been used to map the filtered sEMG to activation signal (σ) similar to the proposed evaluation by Nasr et al. [382]. Secondly, for finding the optimum cut-off frequency and filtering combinations for minimum mapping error, the method illustrated in Fig.1 has been proposed. The pool of possible steps includes:

- the low-pass filter
- the high-pass filter
- the normalization to the maximum value of signals
- the full wave rectification step
- the white step (a non-filtering step)

We have drawn from the pool with the following conditions:

i. The minimum and maximum number of steps selected from the pool should be one and four, respectively;

ii. All steps except the white one have one chance of being chosen;

iii. The white step can only be at the end of the previous steps;

iv. If the filtering steps of high-pass or low-pass are selected, they may have 10, 40, 70, 100, 200, or 500 Hz cut-off frequency

After drawing from the pool, the filtering and manipulation steps construct 1504 possible unique methods, categorized into 63 unique steps without considering the cut-off frequency. The particle swarm method has been used to infer the cut-off frequency from the mentioned range for minimizing the mapping error cost function. Each permutation and cut-off frequency has been used to filter the raw sEMG channels and produced filtered sEMG signals.

Then the machine learning model has been trained using the input (filtered sEMG) and the output. The cost function of optimization is MSE calculated by the machine learning model's output $\hat{\sigma}$, target activation signal σ, start time t_s, and finish time t_f of the task.

7.3.4 Interrupted sEMG signals

As mentioned, one challenge with using myoelectric sensors is that signal disconnection or the wearable sensors losing contact from the skin is inevitable [217]. sEMG signals may have signal loss due to electrode disconnection, muscle artifacts, or the impracticality of measuring deep muscles [217, 218]. In these cases, the sEMG-based control would likely fail. Thus, preparing a robust prediction model for signal disconnection is necessary.

The intention for producing interrupted sEMG signals is to simulate this sEMG signal disconnection. Although in some situations, the signals' amplitude may only be reduced, we have considered the worst-case condition, which is complete disconnection. In this section, each input sEMG signal disconnects, and the output signals are intact. The volume of data samples is equal to the volume of the recorded data multiplied by the number of disconnections.

The dataset for training the robust model was produced according to the algorithm in Figure 7.8 and using the following steps:

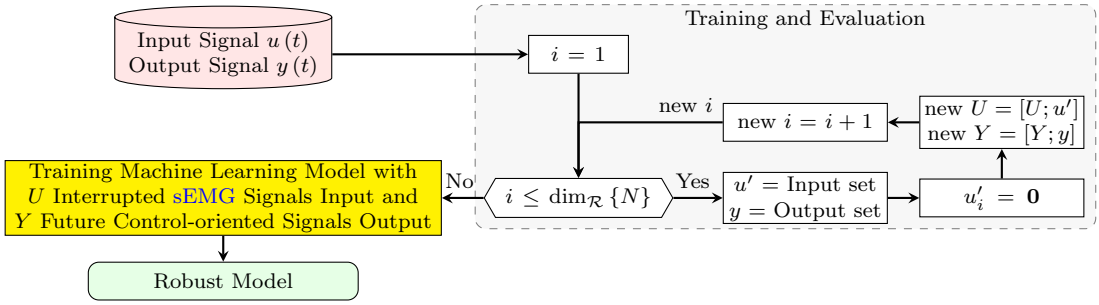


Figure 7.8: Schematic of interrupting sEMG input signal algorithm.

i. Initially set $i = 1$.

ii. For the new input signal, multiply the input channel index i by zero.

iii. If $i \leq N$ is satisfied, the index will be changed ($new\ i = i + 1$), and step (ii.) will be repeated; if not, go to step (iv.).

iv. Train machine learning model using the interrupted input signals and intact output signals.

Repeating steps (ii.-iii.) is similar to finding all combinations of N sEMG numbers, taking $N-1$, and setting one signal at a time to a zero value (mimicking a disconnected signal). The zero value persist for each signal for the whole trial, and consequently, the volume of training data increases by the number of sEMG channels. Multiplying more channels (e.g., the index i and j) by zero and applying the same strategy of assuming more sensors are disconnected is possible.

After completing the described strategy, we have all common conditions which would likely happen to the input signals. As the machine learning model was trained with this dataset, theoretically, it can estimate the output signals using less than N input signals. Note that the data volume is multiplied by the number of signal disconnection possibilities. For example, for 1, 2, and 3 disconnections, the data volume is multiplied by 11, 110, and 990, respectively.

7.4 Results and discussion

For training of the machine learning models, a personal computer with an Intel® Core™ i7-3370 CPU @ 3.40 GHz processor and 16.0 GB memory was used.

7.4.1 Machine learning model

This section presents the data ratio analysis for the training, validation, and testing of subject-specific and general models. Second, the volume of experimental data for complete estimation with a general model was assessed. Third, eighty different configurations of MuscleNET with different input conditions (raw or filtered sEMG signal (section 7.3.2), without or with delayed kinematic signals (section 7.3.1)) and different outputs (5 different biomechanical signals (section 7.2.1)) were compared in terms of the performance, training time, and maximum inference time. Finally, samples of complete estimation for random subjects were presented.

Dividing ratio and data volume

To study the division of data for training, validation, and testing concerning maximum estimation accuracy without overfitting, we trained the models with different ratios of training, validation, and test data. We have assessed the different ratios, and two combinations produced the best results for the subject-specific and general models (Figure 7.9).

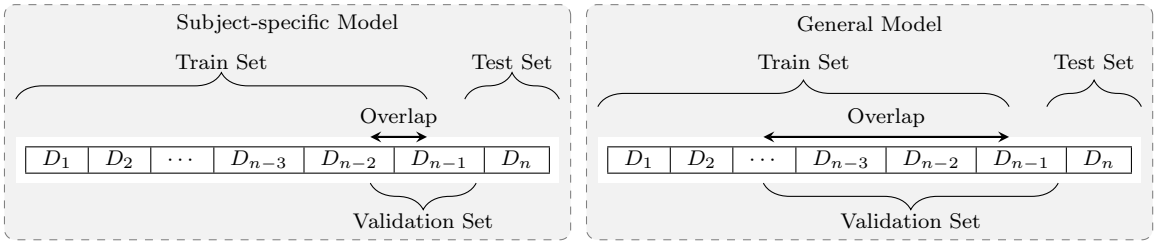


Figure 7.9: Visual representation of dividing data base into training, validation, and testing sets for subject-specific and general models.

The subject-specific model was tuned for maximum estimation accuracy of validation data with 94.1% of data for training, 5.9% of data for exporting-and-validation (note that validation data overlaps with training data), and 5.9% unique data for testing. The general model was achieved with 94.1%, 23.5%, and 5.9% of data for training, exporting-and-validation, and testing, respectively. The overlap of validation and training sets may decrease in the case of reach experimental data availability.

For the subject-specific model, 16 subjects (94.1% of total data) were selected for training, and one of them (5.9% of total) was present in training and validation simultaneously. This subject plays the role of stopping the training process and signaling the export of the final model with the lowest MSE. The other subject who did not participate in training and validation was selected as the test set. We observed that the regression accuracy for the validation subject reached 99.7% (Figure 7.10). However, the test result was not as successful (97.7%). In other words, since the training stopped and was exported based on the regression of one subject validation, the model is subject-specific for the validation subject and a less general model for testing data sets. The machine learning model has less regression accuracy for testing with different data sets (any subjects other than those used for validation). Although this trained MuscleNET with one validation subject's data has high regression accuracy for the validation subject, the model is subject-specific and should not be used for general purposes. If the sEMG signals were not stochastic and time-varying, the regression would be greater than the amount reached already. In addition, from a biomechanical perspective, the maximum muscle force, muscle attachment configuration, and muscle passive and active functions (biomechanical muscle features) are different between different subjects. Thus, selecting a small amount of data for validation (for example, only one subject) decreases the breadth of biomechanical muscle features for different subjects and increases the validation accuracy for the subject-specific model.

For the general model, we found that 23.5% of data (4 subjects) for validation gave better results than 5.9% of data (1 subject) used for subject-specific cases. The general model estimated the biomechanical signals of the test data set (1 subject) with a regression of 99.1% (Figure 7.11). In other words, less data for validation and more data for training decreased the regression of the general model for testing. This issue is called overfitting in supervised machine learning and prevents generalized machine learning models from properly fitting observed training data, along with unobserved data on validation and testing sets [362]. Thus, we needed validation data for more than one subject in order to have a general machine learning model that did not overfit the data.

For stochastic and time-varying sEMG signal inputs to the machine learning model, it is essential to have a rich data set to extract and validate the model and configure the model architecture. Designing a machine learning model with many layers and millions of variables requires very large datasets. The combination of training, validation, and testing datasets with 1,530,000 pairs of inputs and outputs were sufficient for MuscleNET (a shallow neural network with three layers and a deep neural network with eight convolutional layers and a small filtering size).

Using a subject that has not participated in the training and validation process for estimation evaluation purposes provided the opportunity for electrode shift evaluation, which is another challenge for daily prosthesis use [177, 384–386]. Electrode placement, human body characteristics, muscle strength, and skin thickness differ from one subject to another. Generally, when using different subjects for validation and testing in a machine learning model with a wide configuration network, the electrode shift is assessed between subjects, and the full estimation accuracy is estimated via data of a random subject.

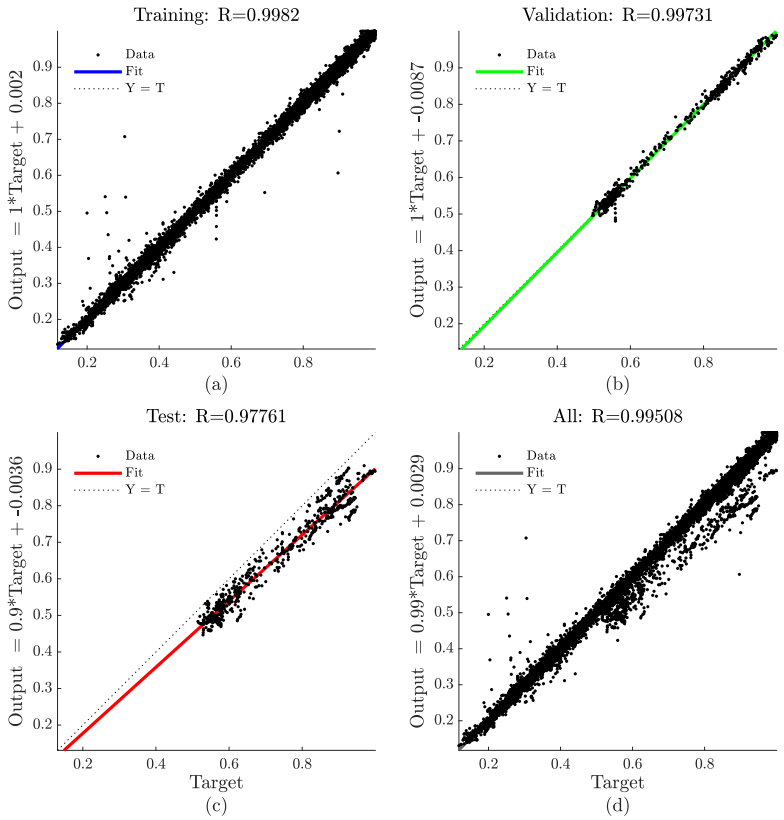


Figure 7.10: The machine learning training in (a), validation in (b), and testing in (c) regression of a subject-specific model of RNN with filtered sEMG and delayed kinematic signals input and joint angle output. The test subject did not participate in training and validation.

Models comparison

From a topology viewpoint (Table 7.4), the machine learning method with the recurrent topology or the feedback outperforms the feed-forward mapping (RNN and RCNN had average regression accuracy of 89.5% and 95.7%, respectively, while ANN and CNN had lower average regression accuracy of 85.7% and 90.0%, respectively). Previously, Chen et al. [189] reported that the recurrent-based models, RCNN consisting of LSTM networks, were much better than a CNN consisting of a FC network, which agrees with our conclusion based on current results (Table 7.4). The reason behind this is associated with the relation of sEMG to the motion. According to the active and passive muscle tension behavior utilized for the Hill-type muscle dynamic model [82], the relationship is not straightforward and has a second-order differential equation. The recurrent topology of the RCNN uses the previous output signals as the inputs; therefore, it can model a higher-order differential equation.

Models' delay

Since real-time control of biomechatronic devices requires a minimal delay, this factor should be considered. Comparison of the system's delay is controversial since the signal filtering delay should be considered as well. Subsequently, filtering the raw sEMG signal has a specific delay due to low-pass filtering. The delay in filtering the raw signals should be considered with the model delay to compute the total delay. In this regard, the models' delay with the filtered sEMG signals exceeds expectations. All the control methods had an operation delay of less than 2 milliseconds (Table 7.4) because of shallow configuration (not having too many layers), which is quick for real-time control purposes.

Compared to sEMG frequency-based input [214, 215], the MuscleNET is much faster and works in real-time since prior research [214, 215] initially converted sEMG signal to sEMG frequency bands, which is an offline and

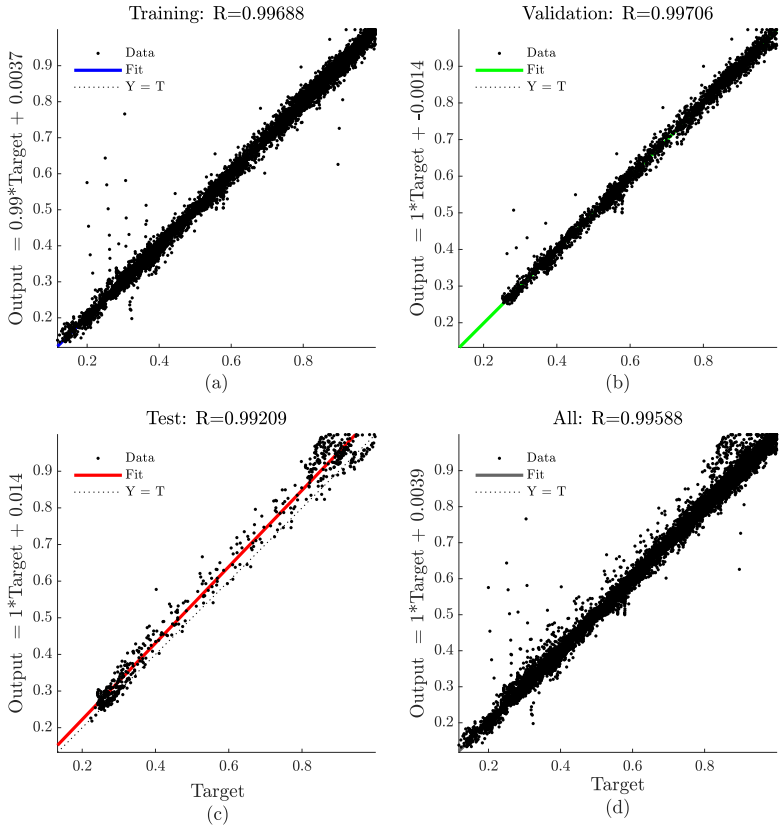


Figure 7.11: The machine learning training in (a), validation in (b), and testing in (c) regression of a general model of RNN with filtered sEMG and delayed kinematic signals input and joint angle output. The test subject did not participate in training and validation.

time-consuming conversion. Moreover, sEMG-based classification of hand or wrist gestures [181, 182, 188, 197, 209, 210] requires additional sEMG history compared to RNN MuscleNET which mainly uses the signal value and a faster regression. The Ameri et al. [213] model also has five times more variables than CNN MuscleNET with delayed kinematic input, indicating that MuscleNET incurs less computation cost and has less delay than that model [213].

Model configuration optimization

The RNN model was trained, validated, and tested with the 90 different configurations for control-oriented output (section 7.2.3) and interrupted sEMG input (section 7.3.4). Firstly, the top regression accuracy was obtained when the number of hidden layers (n_l) was 1. While having more hidden layers is possible, the training was limited to one hour, and the mapping slowed and required more memory as the number of hidden layers increased. Secondly, the optimal number of nodes in each layer (n_n) was approximately 30. With inadequate or spare nodes in each layer, the regression accuracy reduced or required more time for training, respectively. Thirdly, the number of input signal former values (n_u) was 3, which indicates that the history of biomechanical signals is essential for mapping. Finally, the number of the output signal former values (n_y) was required to be 3 for the optimum configuration. With more of the output signal's former values, the output becomes smoother. This result is quite similar to the configuration of the RNN model proposed as a machine learning muscle model [38].

Table 7.4: Comparison of regression accuracy for 80 models: 16 MuscleNET configuration times 5 output signals.

Model Properties						Regression (%)						
#	Configuration Identity	Inputs: Filtered sEMG / Raw sEMG / Delayed Kinematic	Size	Number of Neurons or Variables	Maximum Inference Time (microsecond)	Normalized Output Signal with a 1×1 Size: Kinematics: Joint Angle	Kinematics: Joint Velocity	Kinematics: Joint Acceleration	Dynamics: Joint Torque	Dynamics: Activation Signal	Average ± STD	
1	ANN	✓	11×1	716	6	91.1	81.7	79.2	87.4	89.9	85.9 ± 5.2	85.7 ± 6.4
2		✓	11×1	3796	10	81.1	69.2	65.9	80.5	85.0	76.3 ± 8.3	
3		✓ ✓	14×1	897	6	96.9	93.7	87.2	92.2	93.9	92.8 ± 3.6	
4		✓✓	14×1	4089	10	95.1	88.9	78.7	86.2	90.0	87.8 ± 6.0	
5	RNN	✓	11×1	2024	12	94.9	87.7	81.3	90.4	91.6	89.2 ± 5.1	89.5 ± 6.4
6		✓	11×1	2024	12	86.7	71.4	69.3	81.4	85.4	78.8 ± 8.0	
7		✓ ✓	14×1	1848	12	98.5	98.2	95.0	94.6	95.6	96.4 ± 1.8	
8		✓✓	14×1	1848	12	97.6	97.1	92.9	88.6	91.2	93.5 ± 3.9	
9	CNN	✓	11×250	62437	1400	92.2	85.5	85.5	88.5	90.2	88.4 ± 2.9	90.0 ± 2.2
10		✓	11×250	62485	1500	93.8	89.6	86.6	91.4	91.8	90.6 ± 2.7	
11		✓ ✓	14×250	10035	1200	92.7	87.7	86.6	89.4	90.7	89.4 ± 2.4	
12		✓✓	14×250	13825	1300	94.6	90.1	88.4	91.6	92.6	91.5 ± 2.4	
13	RCNN	✓	11×250	125387	1900	96.3	96.1	94.9	94.7	95.7	95.5 ± 0.7	95.7 ± 0.6
14		✓	11×250	125435	2000	96.4	96.2	94.9	94.9	95.9	95.7 ± 0.7	
15		✓ ✓	14×250	108785	1300	96.5	96.2	94.9	94.8	95.8	95.6 ± 0.8	
16		✓✓	14×250	110425	1350	97.2	96.6	95.0	94.9	95.8	95.9 ± 1.0	
			Average ± STD			93.9 ± 4.5	89.1 ± 8.8	86.0 ± 9.2	90.1 ± 4.6	91.9 ± 3.5		

Testing and demonstration

These findings motivated re-learning of two specific models and comparison of their results. In those models, the outputs were the joint angle and activation signal. The two models were (I) RNN with filtered sEMG signals plus delayed kinematics and (II) RCNN with raw sEMG signals plus delayed kinematics. For re-training of the RNN, the maximum elapsed time was set to 12 hours. For re-training of the RCNN MuscleNET, the maximum epoch was set to 1000. The rest of the training options were the same as the previous training step. As an example, the RCNN training and validation error rates for 1000 iterations are presented (Figure 7.12).

The performance of RNN and RCNN MuscleNET for (a) joint angle and (b) activation signal trajectory with a user dataset that had not been used for training is shown in Figure 7.13. The RCNN MuscleNET followed the target values, which were the normalized joint angle and joint torque.

Another random subject data was used to evaluate the model (Figure 7.14). This complete estimation of the biomechanical signals of random subjects, who did not participate in training and validation of the model, had 96.9% accuracy. The result showed that the novel model configuration, using delayed kinematic signals, the optimum model configuration, the given amount of data, and the optimum division for training, validation, and testing successfully achieved the goal. The amount of experimental data was sufficient to reach this regression accuracy for estimation tests.

Another point to consider regarding the RNN model application and evaluation is that this recurrent machine learning model relies highly on input and output data history. Specifically, for an RNN configuration with input signal former values (n_u) of 3 and output signal former values (n_y) of 5, starting the estimation requires some previous data that is not available at $t = 0$ seconds. This study feeds the RNN model with random data at the

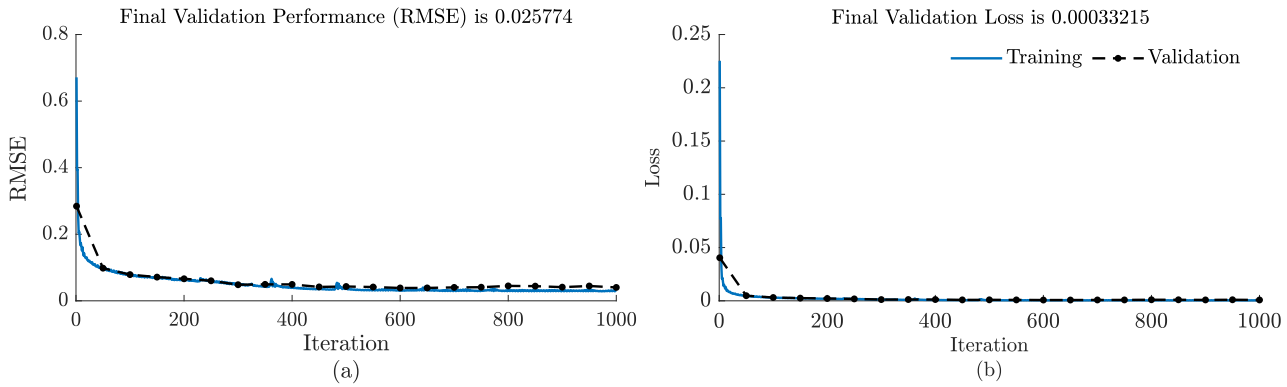


Figure 7.12: The RCNN training (solid) and validation (dashed) rates for 1000 iterations. However, the training procedure could be stopped at 500 iterations.

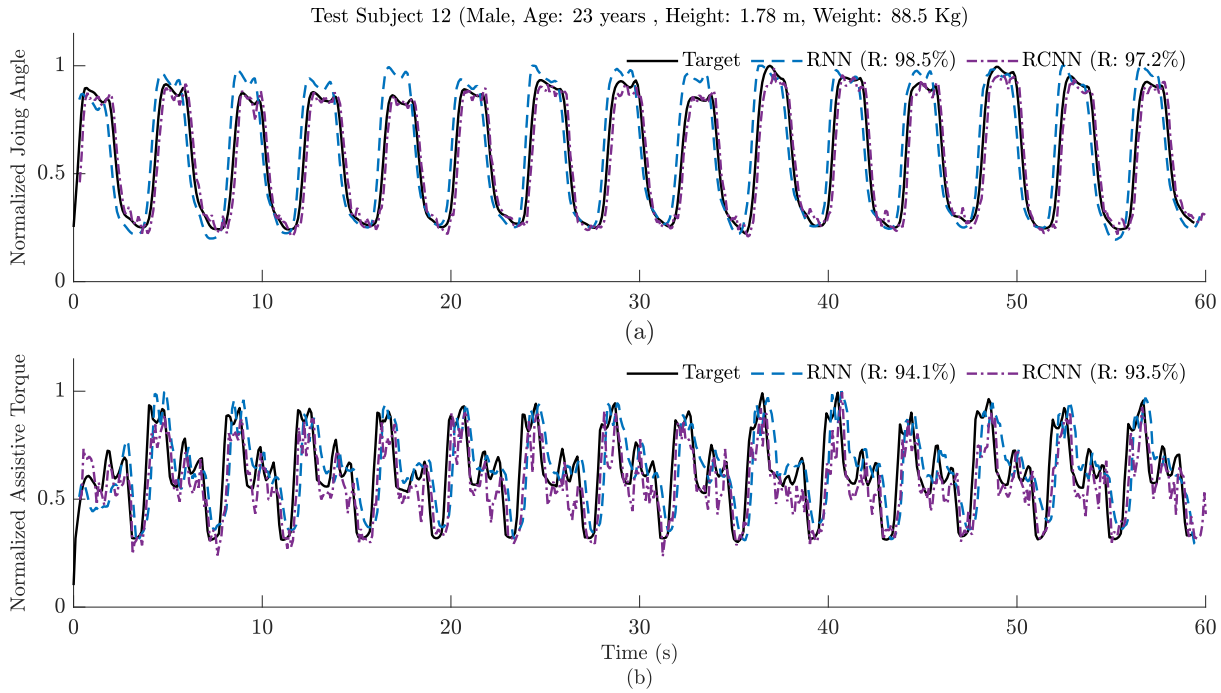


Figure 7.13: The performance of RNN and RCNN MuscleNET on random test data for users' joint angle in (a) and joint torque trajectory in (b).

start phase. It then continues with experimental data from when the data was available. Random input data at the beginning leads to inaccurate estimation at the beginning phase. This estimation was used as the output signal former values for further estimation. The RNN model converges in less than 2 seconds. For evaluation, a section of the estimation and target data for four subjects were shown in Figure 7.15.

Conclusion

An alternative solution to mathematical muscle modeling using regression RNN and CNN-based sEMG signals is proposed. The model's performance and quality metrics were compared across 80 different machine learning regression-based schemes with various conditions for different kinds of neural network configurations: ANN, RNN, CNN, and RCNN. The two models of (A) RNN with delayed kinematic signal and filtered sEMG signals inputs and (B) RCNN with delayed kinematic and raw sEMG signal inputs outperformed the other schemes

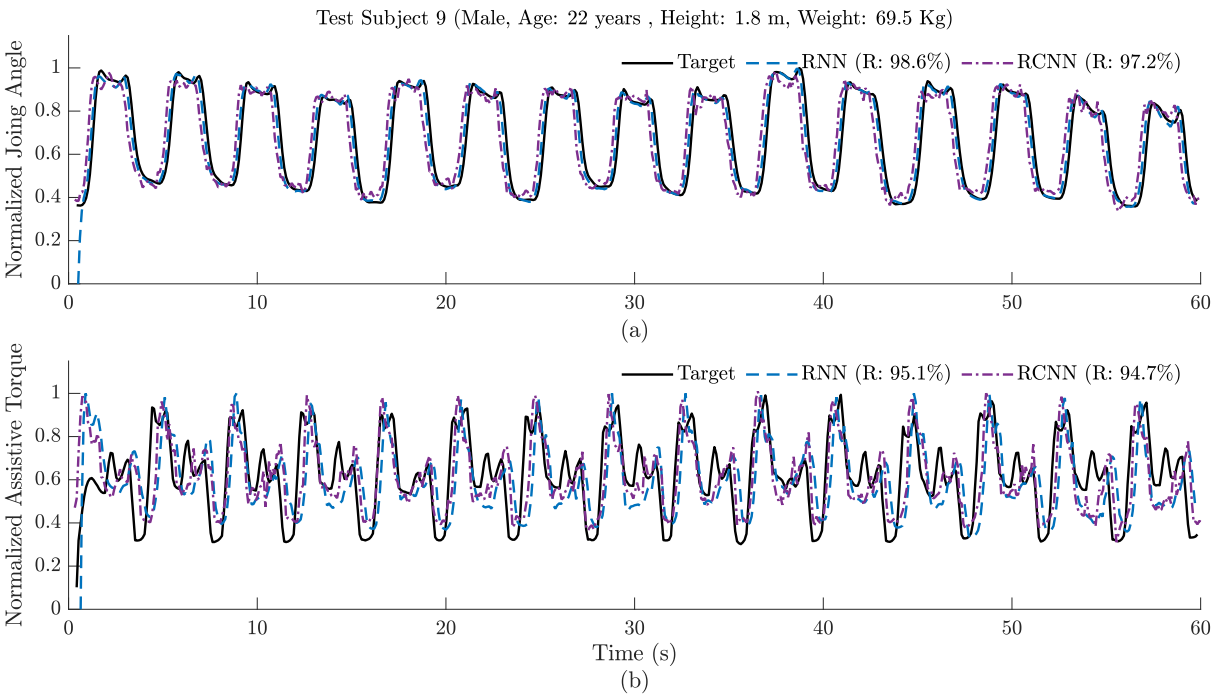


Figure 7.14: The test of RNN and RCNN MuscleNET for second random non-used user data for training users' joint angle in (a) and joint torque trajectory in (b).

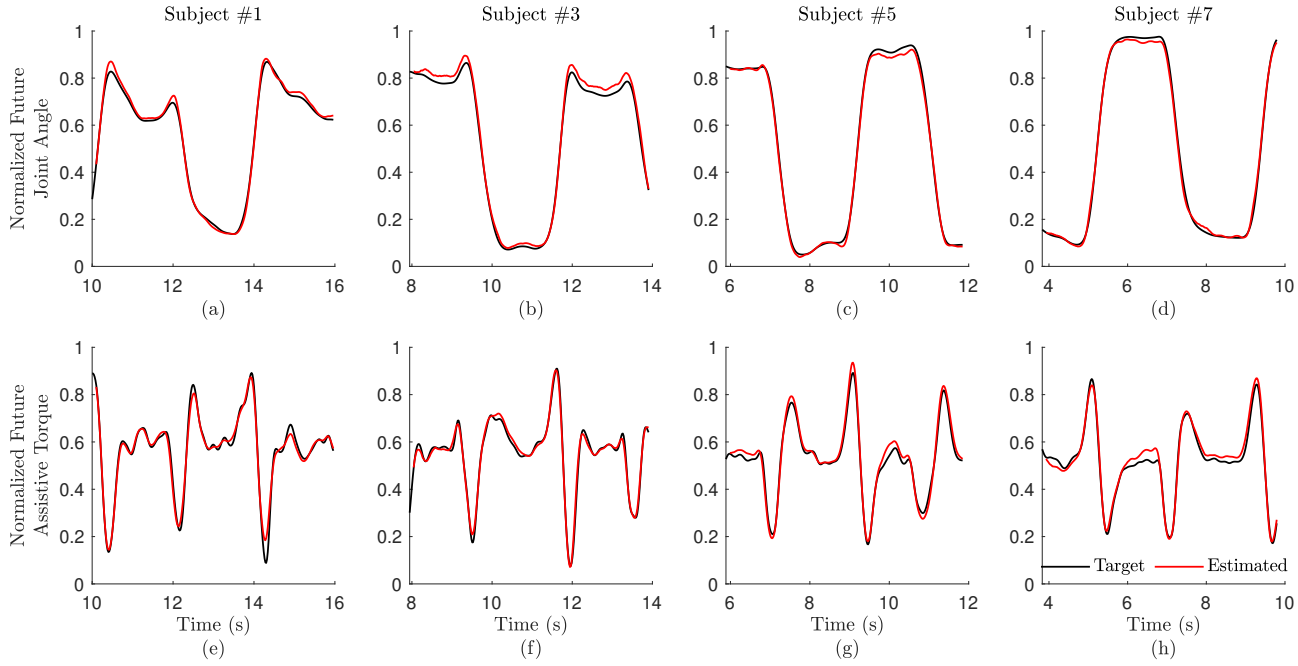


Figure 7.15: A section of four subjects' time-series data estimation for normalized future joint angle in (a-d) and assistive torque in (e-h) using a prediction horizon of 0.1 seconds. These subjects had similar anthropometric data (body height and mass) as the average subjects.

in terms of estimation accuracies. The high performance of the recurrent-based models demonstrated their ability to learn necessary motor information from sEMG signals. Further, the signal filtering capability of the CNN-based models was established.

7.4.2 Output signals

Biomechanical variables

The mentioned biomechanical variables were used as output of the machine learning model and the results are provided in Table 7.4.

In terms of the models' outputs (Table 7.4), accuracy varied. In comparison to Bao et al. [211] which used a CNN for regression of only wrist joint angle, we have evaluated different output signals (Kinematic and Dynamic Biomechanical Variables) and calculated relative accuracy, which helps improve the future application of MuscleNET for myoelectric-based control.

The general accuracy of the outputs in ascending order is: output joint angle, activation signal, joint torque, output joint velocity, and output joint acceleration (with average regression accuracy of 93.9%, 91.9%, 90.1%, 89.1%, and 86.0%, respectively). This order concurs with mathematical muscle models. The muscle pennation angle, muscle wrapping, muscle-tendon length, and passive muscle torque are functions of the joint angle [82]. In addition, the relation of muscle velocity and activation dynamics of muscle has been seen in the MTG models for activation signal [84]. The maximum muscle force was relevant to the joint torque [82]. Thus, the first three dominant output signals correspond to the primary variables in classical muscle models. Consequently, we propose using the sEMG signal to control biomechatronic devices using the joint angle or activation signal.

In this regard, since most robotic servo motors have a PID position control, using the sEMG as the input of the control algorithm and the joint angle as a command to the low-level actuator control is more straightforward. Another method is using a force/torque-based control algorithm for biomechatronic devices. With the sEMG used as the input, the machine learning model can estimate the activation signal using an MTG model [56]. The joint torque can be calculated and commanded to the low-level control loop or the robot actuator. The activation signal's superiority over joint torque arises from consideration of the passive and active torque impact of muscle in the MTG model [56, 84].

The machine learning models accurately estimated joint angles and activation signals more than other biomechanical variables such as joint velocity, acceleration, and torque. In MSK model simulation and myoelectric-based control of biomechatronic systems, the mentioned biomechanical variables can be used.

Biarticular output

Previously, the regression accuracy of the RNN Biarticular machine learning model achieved 94.3, 86.3, 89.9, and 92.9% regression accuracy percentage for training, validation, testing, and all data sets, respectively. Coupling the joints' activation signals for Biarticular machine learning model's outputs can accomplish 3% more regression accuracy than the average monoarticular MuscleNET reported in section 7.2.1 and Table 7.4.

The proposed combination of Biarticular machine learning model and monoarticular MTGs for NMSK simulation predicted torques for an adult's pick-and-place motion comparable with experimental data. As a result, using the proposed models in NMSK simulations (Figure 7.21) alleviates the difficulties associated with anatomically precise muscle models.

Control-oriented output

Using the control-oriented signals (the exoskeleton's joint angle and assistive torque) as the output of machine learning model assisted in decreasing the calculation cost compared to using human biomechanical variable outputs. The conversion of the human biomechanical variables to the robot's kinematic and kinetic variables alternatively would require iterative time-consuming variables calculation with 0.1 seconds per frame of data for an assistive torque. This time-consuming iterative calculation is helpful for offline design [387].

When compared to using human biomechanical variable outputs, using control-oriented signals (the exoskeleton's joint angle and assistive torque) as the output of the machine learning model assisted in lowering the calculation cost.

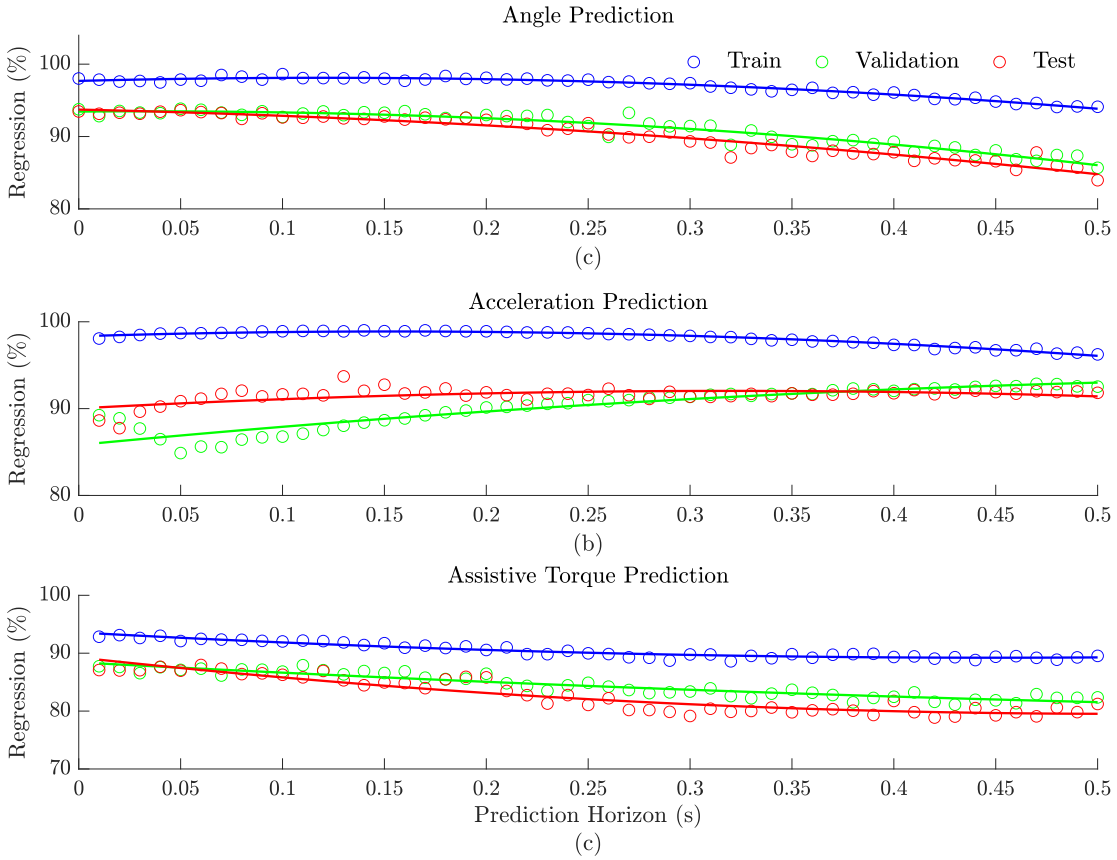


Figure 7.16: The regression accuracy of training, validation, and test data set for angle prediction in (a), acceleration prediction in (b), and assistive torque prediction in (c).

Future control-oriented output

The accuracy of the predictive model was assessed for two methods: angle prediction and acceleration prediction. The angle prediction $\theta_t + \Delta t$ used a time-shifted joint angle, and the acceleration prediction $\ddot{\theta}'_h$ used Equation (7.3). The shift, the positive delay range, or the prediction horizon was assessed between $0-0.5$ seconds (Figure 7.16).

For the angle prediction in Figure 7.16, the regression accuracy dropped under 92% for a prediction horizon of more than 0.1 seconds. Thus, we recommend a prediction horizon of less than 0.1 seconds for the joint angle. This result approximates the electromechanical delay phenomenon [219].

For the acceleration prediction in Figure 7.16, the accuracy of the acceleration estimation was insufficient. Besides that, Equation (7.3) requires an accurate joint angle and velocity to predict the future joint angle. Thus, we do not propose to use Equation (7.3) for the acceleration prediction. If the biomechatronic device's current joint angle and velocity were measured with the highest accuracy and the estimation criteria were less than 92% regression accuracy for the validation and test datasets, Equation (7.3) could be used successfully.

The joint angle prediction was successful and had adequate accuracy (more than 90%) for less than 0.1 seconds (Figure 7.16). The amount is similar to the electromechanical delay phenomenon, defined as the time delay between the onset of the sEMG and the onset of the human motion [219]. Using this positive delay, the machine learning model could successfully predict the output signals for the next 0.1 seconds. The amount of electromechanical delay or prediction horizon may differ depending on individual body characteristics [220] and different tasks and can be evaluated using the same method.

For the assistive torque prediction in Figure 7.16, the regression accuracy dropped under 86.5% for a prediction horizon of more than 0.1 seconds. This assistive torque prediction can be used in high-level predictive

control of assistive devices.

Using the future signals as the output of machine learning allows for real-time volitional high-level control of the biomechatronic system, and having the future intent provides the mechatronic device with the opportunity to be ready [19]. For example, the actuation system can run to serve in advance whenever the mapping system indicates a high amount of assistive torque for future steps.

Conclusion

In this section, different outputs of the machine learning model were assessed, such as human biomechanical variables (human joint angle, velocity, acceleration, torque, activation signal) for single SFE joint, multiple activation signals for SFE and EFE, control-oriented signals (the exoskeleton's joint angle and assistive torque), and shifted control-oriented signals. According to the results, we proposed using shifted control-oriented signals for myoelectric-based control of the assistive device. In addition, multiple activation signals can be selected as the machine learning output for the NMSK model simulations.

7.4.3 Input signals

Here, results of evaluating different inputs are provided in details. The inputs are: delayed kinematic signals (section 7.4.3), raw or filtered sEMG signals (section 7.4.3), optimized filtered sEMG signals (section 7.4.3), and interrupted sEMG signals (section 7.4.3).

Delayed kinematics

From the kinematic input standpoint, using the delayed kinematic signals along with the sEMG signals positively affected the model's accuracy as provided in Table 7.4 (using delayed kinematic signals resulted in an average regression accuracy of 92.9% while not using those signals resulted in a lower average regression accuracy of 87.6%). It is noteworthy to mention that the Hill-type muscle model's inputs [82] are the joint angle, velocity, and acceleration, along with the sEMG signal. Thus, using the kinematic signals along with the sEMG signal, for the first time, is novel and increased model accuracy. Indeed, incorporating the delayed kinematic signals decreased the regression error from 50% to 20%. For example, using the delayed kinematic signals in row 12 in Table 7.4 provides more accuracy than row 10, which results from the model proposed by Ameri et al. [213]. Using the delayed signals instead of the current signals reflects the need to consider electrical and computational delays. Moreover, for real-time control of an actual bio-robot, the delay must be considered in the modeling. For the simulation of the musculoskeletal model, delayed signals are unnecessary.

Raw or filtered sEMG signal

From the sEMG input perspective, the result (Table 7.4) provided two distinct conclusions regarding other input conditions: raw or filtered sEMG signal. First, for the ANN and RNN models, the models' accuracy was much better when the inputs were filtered signals (filtered sEMG signals yielded average regression accuracy of 91.1%, while raw signals yielded average regression accuracy of 84.4%). Hence, the ANN and RNN models should not be used for filtering the raw signals. The results revealed the capability of the convolutional models in filtering the signals and confirm the superiority of using raw sEMG signals as the inputs of the CNN and RCNN models.

Optimized sEMG filtering steps

The algorithm described in section 7.3.3 tested 1504 input groups in the training procedure of the machine learning model with the same training conditions and saved the best MSE for each unique step. After evaluating the MSE of the steps, we found that 5 of them had the least MSE (Table 7.5), and the rest had considerably higher MSE. One of the interesting results of this evaluation was that the low-pass filtering step appeared after

Table 7.5: The top 5 filtering combinations with the lowest MSE.

#	Combination and Cut-off Frequency	MSE %
1	70 Hz High-pass > Rectifying > 10 Hz Low-pass	7.95
2	500 Hz High-pass > Rectifying > 10 Hz Low-pass > Norm to maximum	7.96
3	Norm to maximum > 10 Hz High-pass > Rectifying > 10 Hz Low-pass	8.33
4	Rectifying > Norm to maximum > 10 Hz Low-pass	8.50
5	10 Hz High-pass > Rectifying > Norm to maximum > 10 Hz Low-pass	8.58

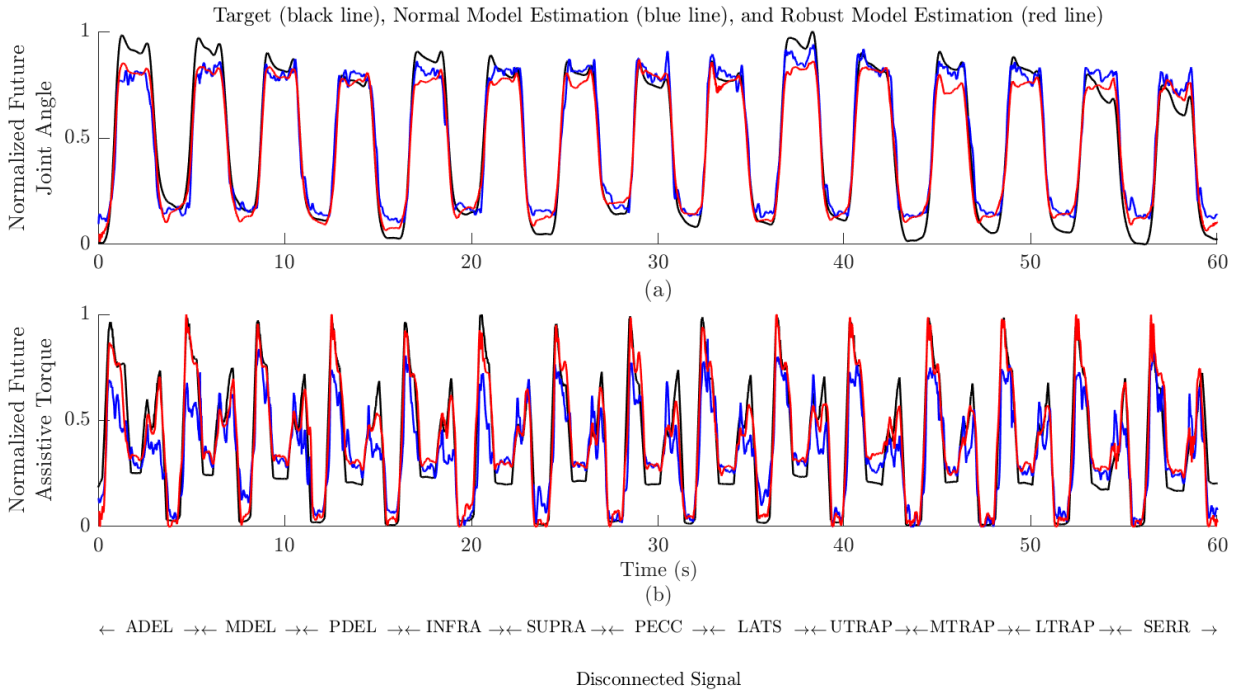


Figure 7.17: Evaluation of normal model and robust model with interrupted sEMG signals. The target output is the normalized future robot's joint angle in (a) and assistive torque in (b).

the full wave rectification step in all seven best steps. Furthermore, the 10 Hz low-pass filter had a better result than all other (higher) cut-off frequencies.

By using three filtering steps of 70 Hz high-pass filter, then full wave rectification, and finally, 10 Hz low-pass filter, the key sEMG information is preserved for a machine learning muscle model; additional filtering steps or different filtering frequencies are not needed for the object manipulation task (section 4.1.3). Combining the optimized filtering algorithm with the machine learning model and MTG, one obtains an accurate sEMG to activation signal mapping that can replace traditional muscle models.

Robustness

The estimation ability of normal and robust models were also evaluated using interrupted signals (Figure 7.17). The input into the models was the interrupted sEMG (section 7.3.4) and delayed kinematic signals (section 7.3.1). For the evaluation, eleven sEMG signals were disconnected one by one. Two normal and robust models were trained using different input signals and the same target signals. Specifically, the normal RNN model was trained with the ideal input signals without any disconnection. Subsequently, the robust model was trained using the interrupted input signals.

The normal model had more error and sometimes misestimated the target signals (Figure 7.17), especially for the normalized assistive torque (regression of 71%). The robust model handled the disconnection of input

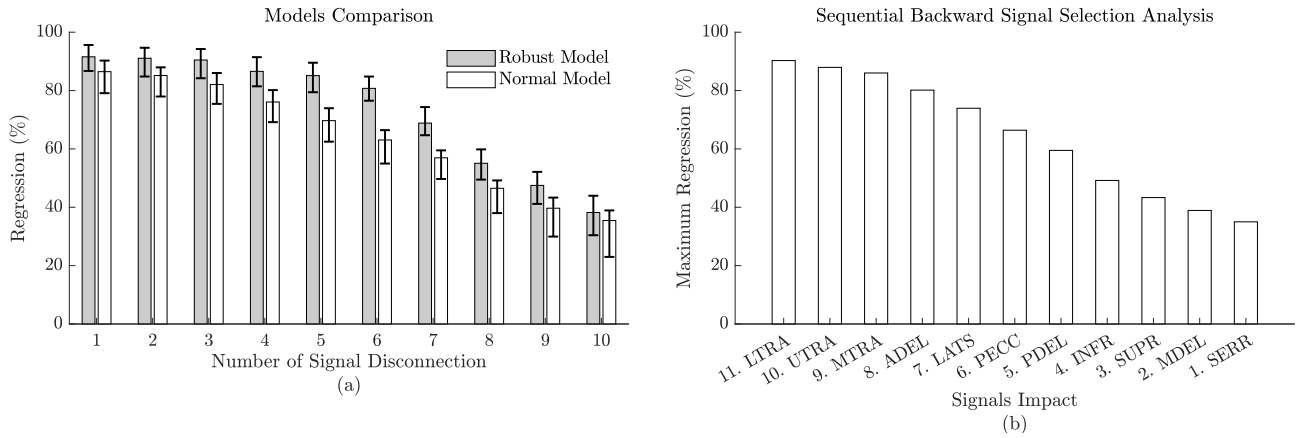


Figure 7.18: Comparison of robust and normal model with the number of interrupted sEMG signals input evaluated for all test subjects' data in (a). Signal impact versus maximum regression with sequential backward elimination signal analysis for 11 signals in (b).

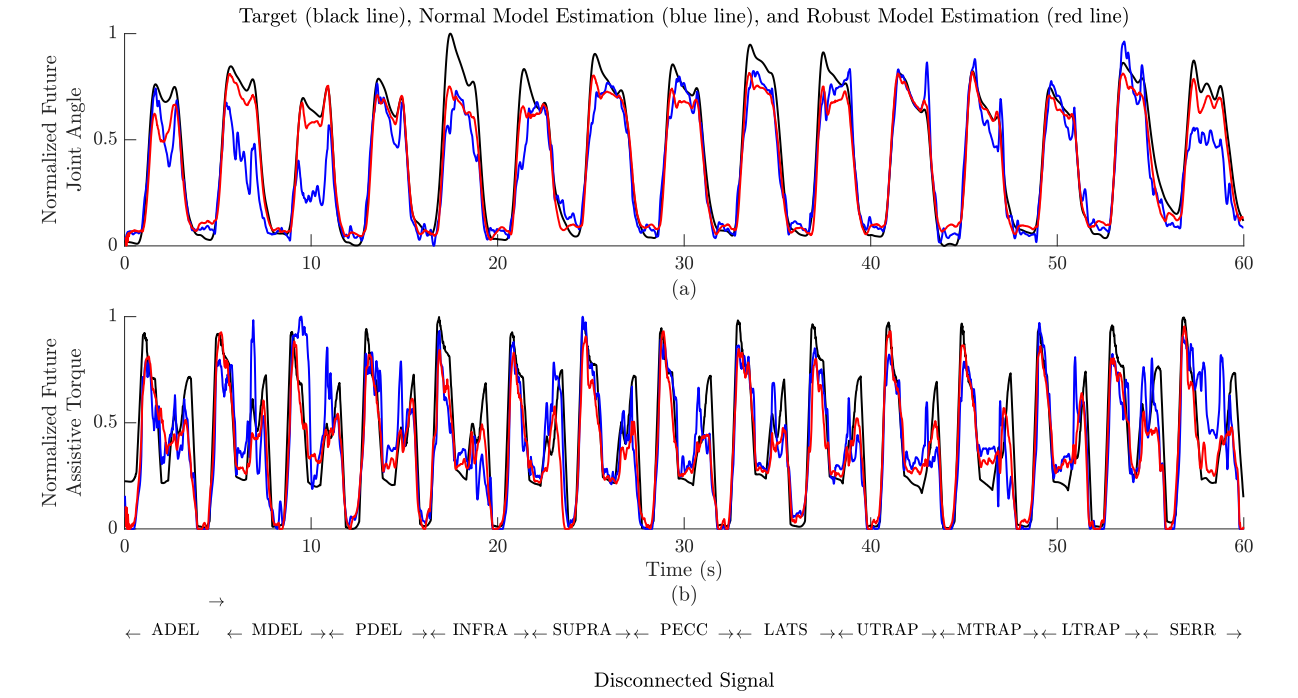


Figure 7.19: Evaluation of normal model and robust model with interrupted sEMG signals for a random subject. The normalized angle prediction is in (a), and normalized assistive torque prediction is in (b).

signals and estimated the output signals with regression accuracy of 92.3%. The higher regression accuracy of the robust model indicates that the robust model is more suitable for practical application.

The normal and robust model with interrupted sEMG signal were also compared (Figure 7.18). All test subjects' data was used to assess the estimation proficiency of the normal and robust model in the presence of sEMG disconnection. The robust model had $5-15\%$ more regression accuracy than the normal model when there is an interruption. This robust model thus addressed the issue of sEMG signal disconnection [217, 218]. An example of the estimation ability of normal and robust models for a random subject is displayed (Figure 7.19).

The sEMG signal disconnections in section 7.3.4 are extended from 1 to 10 disconnections (for 11 sEMG channels in total), and the robust model was trained and compared to the normal model performance in the top part of Figure 7.18. The regression of the robust model with more disconnections generally decreases, but

still is greater than the normal model. The regression accuracy for both robust and normal models drops below 50% for eight signal disconnections. This means that at least 4 sEMG signals should maintain connected for the shoulder exoskeleton device for more than 50% regression accuracy. The shoulder exoskeleton assists the 3-DoF shoulder joint and requires 4 sEMG signals from at least 4 muscles to control the joint. This result of a minimum of 4 signals for a 3-DoF joint is similar to what is seen in muscle synergies [58].

In the bottom part of Figure 7.18, the signals' impact was evaluated using sequential backward elimination signal analysis. Each signal was deleted from the signal bank in the analysis, and the machine learning model was trained. Then, the deleted signal with the maximum regression was reported as the non-important signal. For example, the maximum regression was achieved with the LTRA deletion from the 11 signals database for the first analysis loop. The second analysis achieved the maximum regression with the LTRA and UTRA ignoration from the 11 signals database.

In total, the order of least to most essential signals is: 11. LTRA, 10. UTRA, 9. MTRA, 8. ADEL, 7. LATS, 6. PECC, 5. PDEL, 4. INFR, 3. SUPR, 2. MDEL, 1. SERR. For the mentioned task of sagittal pick-and-place with the mentioned EMG recording unit (section 4.1), the results show that SERR and MDEL directly relate to the output of the machine learning model. This analysis should be done for different tasks and different measurement units before controlling wearable robots.

Since the output data (robot's joint angle and assistive torque) was normalized in Figure 7.17 and Figure 7.19 to show fair estimation assessment, it may seem as if the data strictly follows a specific pattern and that it is possible to predict with time series analysis. However, there are several explanations as to why the data is not following an exact pattern with constant frequency. First, although the 30 cm^2 area of the lower and upper targets were similar for all subjects, the subjects had different body heights and segment lengths, which required different postures to complete the task. In addition, the subjects were not restricted to exact trajectory patterns. Further, the motion timing was not dictated to all subjects, and participants had complete control of triggering the task.

As a further depiction of estimation ability, Figure 7.15 shows a section of estimation with higher resolution. The prediction horizon in Figure 7.15 is 0.1 seconds into the future, which means that the RNN model estimated the target that was shifted by 0.1 seconds. This RNN estimation has 0.1 seconds of shifted data compared to the synchronized mapping of biomechanical variables with sEMG signals [38].

In addition, the assistive torque is highly reliant on anthropometric data of subjects, external wrench due to the object mass, and trajectory of motion. Without having information, especially the object mass, it is impossible to solve the dynamic motion Equation (5.21) and predict the assistive torque with time series analysis of patterns. Using the sEMG signals provides information on the activation of muscles and how the subject controls the arms according to the external object mass. For the volitional control of biomechatronic devices for non-periodic motion [19, 388], e.g., pick-and-place an object, it is practical to use sEMG signals for estimation instead of predicting using only previous kinematic data.

Additionally, the estimation was accomplished with the strict conditions of normalized input data (e.g., Figure 7.17 and Figure 7.19) for all subjects and no sEMG information provided for the validation subject. Specifically, the maximum and minimum joint angle, velocity, acceleration, and assistive torque differed from one subject to another. The RNN model had no information on a subject's exact data (only normalized data was used) and had no information about the muscle strength and sEMG pattern of the validation subject, which differs from one person to the other. However, the RNN model could successfully estimate the data with regression accuracy similar to previous work [38].

7.5 Application

The evaluated machine learning models, input signals, and output signals have two specific applications in NMSK model simulation and myoelectric control of biomechatronic devices.

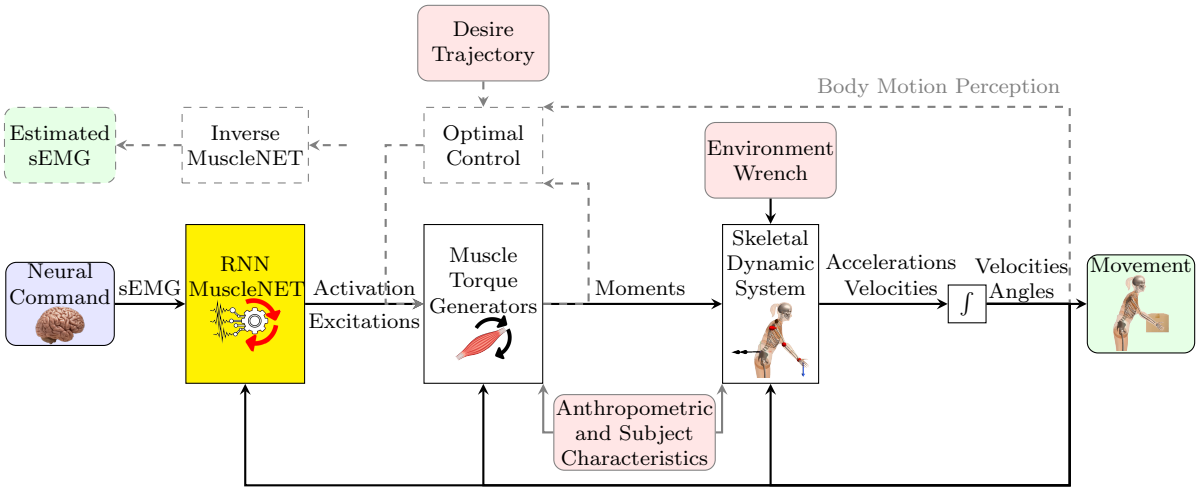


Figure 7.20: Schematic of a NMSK model simulation (for example, for sport engineering purposes or MSK analysis). MuscleNET acts as a neural control and MTG act as a muscle model using the filtered sEMG, joint position, velocity, and acceleration. MuscleNET's output is the joint activation or excitation signal supplied to the MSK dynamic model for forward dynamic simulation.

7.5.1 Neuromusculoskeletal model simulation

Using RNN machnine learning model (section 7.1.3) with inputs of optimized filtered sEMG singals (section 7.4.3) and delayed kinematic signals (section 7.3.1) could replace neural control in NMSK models.

The outputs of the MuscleNET is multiple activation signal of MTG, then it should be used as the input of the MSK dynamic model. MuscleNET and the MSK (consists of MTG) model's total system can be used in the forward dynamic simulation of a NMSK model (for example, for clinical or sport engineering purposes) (Figure 7.21). MuscleNET with multiple activation output signals and MTG can simulate a biarticular muscle model.

7.5.2 Myoelectric-based control of biomechatronic devices

The robust predictive control-oriented model can be used for the high-level control of biomechatronic devices, such as exoskeletons, prostheses, and assistive/resistive robots. This high-level controller predicted the user intent based on the sEMG signal and the robot's kinematics [178, 229]. Since sEMG signals may disconnect in practice, the robust model supersedes a normal model in a human-in-loop control system. Since the model output was normalized to the maximum value, it is recoverable by multiplying by the maximum value. This proposed volitional myoelectric-based control loop using the robust mapping (as high-level control) is displayed in Figure 7.21. For the preparation of this model, offline sEMG, kinematic, and kinetic data was used.

Although this model has been evaluated for a 1-DoF exoskeleton, it can be used for multi-DoF biomechatronic systems. In this case, the model's input should be enriched with more signals, data, and complex tasks relevant to more degrees of freedom when training the RNN.

For a prosthesis, since the amputated limb cannot be used for offline modeling and training of the robust model, the intact mirrored limb may be used for data recording and training the model given that the use of sEMG signals from the remaining limb has been studied further [384, 385]. However, the idea of training the robust model with interrupted input sEMG signals and future control-oriented output signals still contributes to prosthetics control.

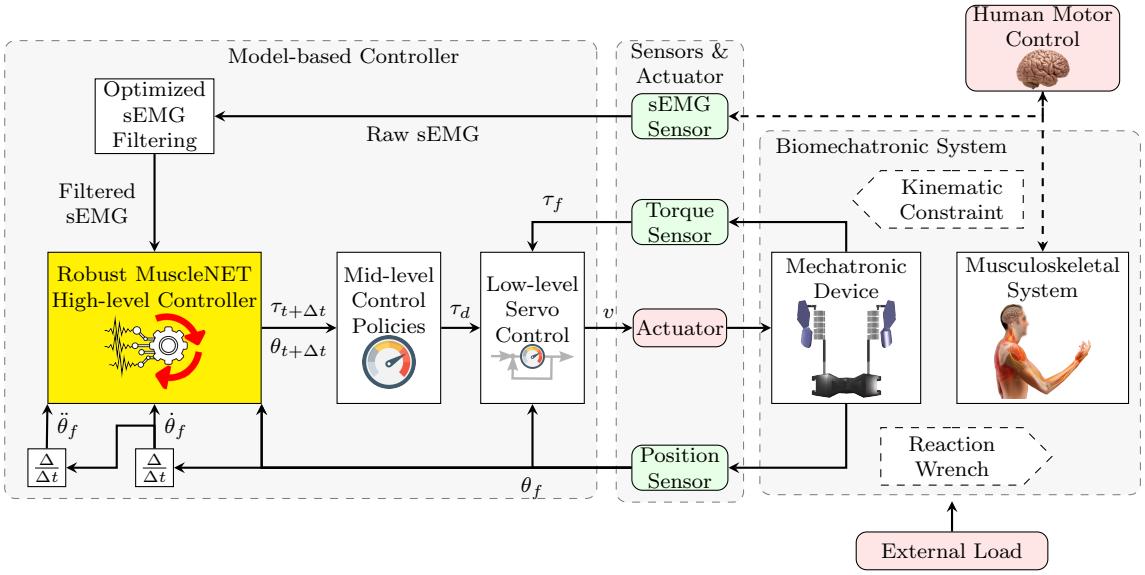


Figure 7.21: Schematic for controlling an exoskeleton, a prosthesis, or a rehabilitation robot using MuscleNET as a model of mapping the raw sEMG, joint feedback position θ_f, velocity $\dot{\theta}_f$, and acceleration $\ddot{\theta}_f$ to future assistive torque $\tau_{t+\Delta t}$ and future joint angle $\theta_{t+\Delta t}$. The mid-level and low-level controller is responsible for controlling the mechatronic device.

7.6 Concluding remarks

Different machine learning models (ANN, RNN, CNN, and RCNN) were evaluated for mapping sEMG to kinematic and kinetic variables. Different input of delayed kinematic, raw or filtered sEMG, optimized filtered sEMG, and interrupted sEMG signals were evaluated. In addition, different output of one joint biomechanical variables, multiple joints biomechanical variables, control-oriented (the robot's variable), and the future variables were assessed. Finally, the RNN model developed a successful robust predictive regression-based model for mapping interrupted sEMG signals to future control-oriented signals. The output of the machine learning model implemented control-oriented biomechatronic signals instead of human biomechanical signals. This control-oriented model decreases computational cost and eliminates the optimization loop for assistive torque computation. Besides that, using future variables instead of the current variable raises the opportunity of high-level controlling the system without delay. Using an interrupted sEMG signal for model training increases the robustness of modeling in the case of disconnection and associated signal loss during online sEMG measurement. The robust predictive control-oriented regression model offers the potential to transform practical volitional myoelectric-based control of biomechatronic devices, such as exoskeletons, prostheses, and assistive/resistive robots.

Chapter 8

Assist-as-needed Mid-level Control of Active Exoskeleton

In this work, the multibody human-exoskeleton dynamic model is first introduced. Second, the control units of the simulated human and exoskeleton are discussed. Third, the results and evaluation of 3 controllers (AAN-proportional rule, AAN-CTM, and AAN-fuzzy logic) on the system of human-robot-environment are presented. Then, the effectiveness of the controllers for the AAN goals and initial, short, and long-term experiences is discussed. Finally, the simulation and experimental results of 2 controllers (AAN-MPC and AAN-CTM) on the human-exoskeleton system are provided, compared, and discussed.

8.1 Human-exoskeleton multibody model

As shown in Figure 8.2(a), the physical system consists of two sub-systems: an adult human body and a wearable robot. The high and mid-level robot controllers operate from a separate computer that represents a third subsystem (Figure 8.1). The EVO (Ekso Bionics Holdings Inc, California, USA) upper limb exoskeleton model, experimentally verified in [27], was used as the wearable robotic system model. The structure of EVO consists of 14 passive revolute joints. The original passive exoskeleton (which was powered by spring mechanism [26]) was equipped with a 9 Nm BLDC motor that was tailored for occupational tasks with the active-passive exoskeleton. Specifically, motor torques were applied about the shoulder orthogonal to the plane of elevation, i.e., they act to increase or decrease the angle of arm elevation. The BLDC motor is connected through a CAN-bus connection to an Arduino adaptor first, and then to a personal/desktop computer. The five variables related to the BLDC motor state that can be changed through real-time control in Matlab are as follows: desired angular position (θ_{ref}), desired angular velocity ($\dot{\phi}_{ref}$), feedforward desired torque (T_{ref}), proportional gain (K_p), and derivative gain (K_d).

A 20-DoF scalable MSK MapleSim upper body model was created. This 8 segment model was developed and validated using experimental dynamometer data [27]. The MapleSim model can be further subdivided into a skeletal dynamic model and a biomechanical joint torque model. The biomechanical joint torque model was created using MTGs and includes five sub-functions [84]: (I) the passive torque, (II) peak joint strength, (III) active-torque-angle scaling, (IV) active-torque-angular-velocity scaling, and (V) excitation-to-activation signal.

The closed kinematic loop comprises the two connections at the exoskeleton's belt and armrest to the human hip and upper arm, respectively. The connection of the exoskeleton's armrest with the human upper arm forms 10 scalar constraint equations (5 each for the left and right side; 2 revolute and 3 prismatic constraints per arm). Equation (8.1) represents the multibody system dynamics as a system of DAEs, since the generalised coordinates

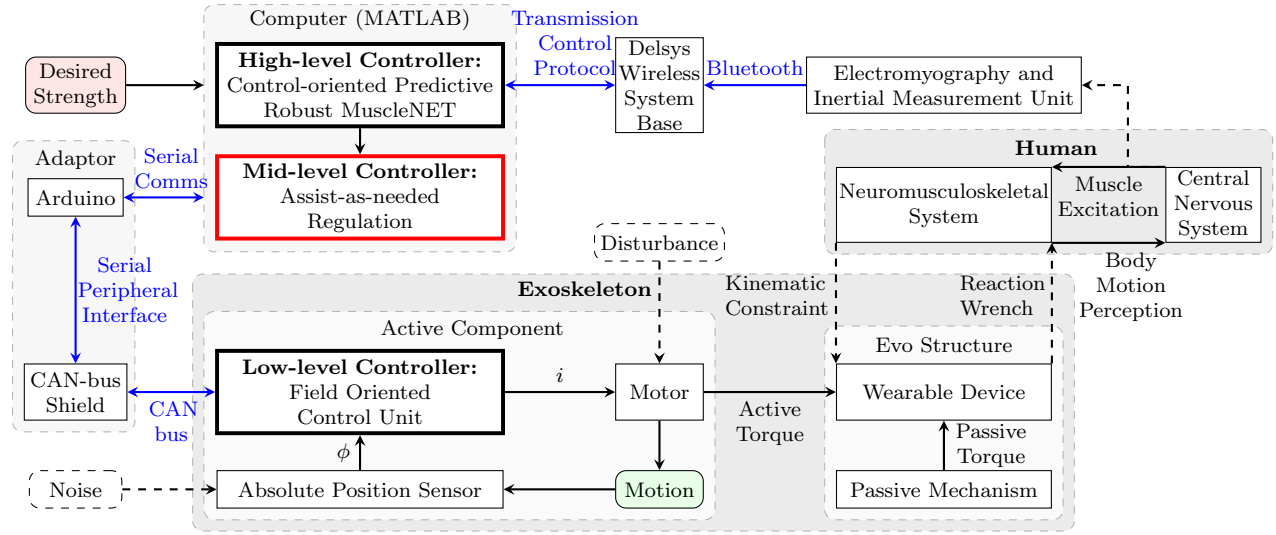
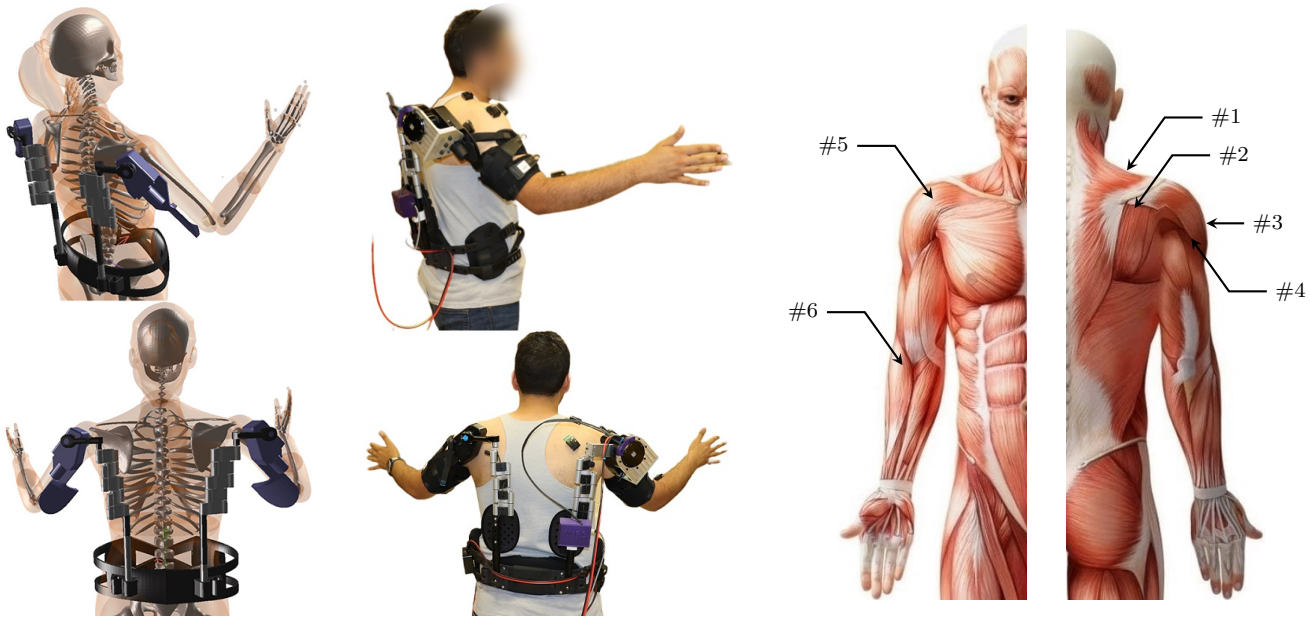


Figure 8.1: Main control system components and connectivity protocols. The high, mid, and low-level controllers are described in section 8.3. The connection protocols/methods are highlighted as blue arrows.



(a) MapleSim model (b) Actual human-robot system (c) Anterior view (d) Posterior view

Figure 8.2: (a) Two views of 20-DoF MapleSim model of a human upper body and EVO upper limb exoskeleton. (b) Depiction of a healthy subject wearing an actual EVO passive system with augmented BLDC motor on the elevation joint. (c & d) Delsys Trigno wireless sEMG system (Delsys Inc, Natick, MA, USA) was used to measure muscle activations over the area of #1 UTRA, #2 MTRA, #3 MDEL, #4 PDEL, #5 ADEL, and #6 BRD of the right forearm, shoulder, and upper trunk muscles.

(q) and generalised speeds (p) are dependant. The actual robot with a subject is shown in Figure 8.2(b).

$$\mathbf{M}_{n\times n}\dot{\mathbf{p}} + \mathbf{C}^T\boldsymbol{\lambda} = \mathbf{F} + \mathbf{Q} \tag{8.1}$$

$$\Phi(\mathbf{q}, t) = 0 \tag{8.2}$$

$$\Psi(\mathbf{p}, \mathbf{q}, t) = 0 \tag{8.3}$$

$$\chi(\dot{\mathbf{p}}, \mathbf{p}, \mathbf{q}, t) = 0 \tag{8.4}$$

where

n	number of joints
m	number of constraint reactions
$\mathbf{M}$	$n \times n$ mass matrix, a combination of the human model, the exoskeleton, and the mass matrices of the assistive components
$\mathbf{C}$	$m \times n$ constraint Jacobian (maps reaction forces/torques to generalized forces)
$\boldsymbol{\lambda}$	$m \times 1$ reaction forces and torques in the cotree joints that enforce the kinematic constraint of human-exoskeletons connection
$\mathbf{F}$	The $n \times 1$ right-hand side of dynamic equations
$\mathbf{Q}$	The $n \times 1$ generalized force vector containing the applied wrench (torque/force) at the joints, which consists of the human joint torque (τ) and the robot joint torque (T)
$\boldsymbol{\Phi}$	$m \times 1$ position-level kinematic constraint equations
$\boldsymbol{\Psi}$	$m \times 1$ velocity-level kinematic constraint equations; if the generalized speeds are equal to the coordinate derivatives, then $\Psi = \dot{\Phi}$
χ	$m \times 1$ acceleration-level kinematic constraint equations; if the generalized velocities are equal to the coordinate second derivatives, then $\chi = \ddot{\Phi}$
$\mathbf{q}$	n generalized coordinates of the multibody system, which consists of the human joint angles (θ) and the robot joint states (φ)
$\mathbf{p}$	n generalized speeds of the multibody system, which consists of the human joint angular velocities (ω) and the derivative of robot joint states ($\dot{\varphi}$)

8.2 Central nervous system control

The motion of the human body is coordinated by CNS-NMPC. To perform motion prediction, an IM represents the complex system dynamics of a human body with full awareness of the interaction force/torque of the exoskeleton and the external force/torque of the manipulated object. The CNS-NMPC controller then optimizes the cost function (Equation (8.5)) subject to biomechanical limitations (Equation (8.6)). According to the study in [28], the cost function weights ($\mathbf{W}_1 = 40000\ \mathbf{I}_{20\times20}\ 1/\text{rad}^2$, $\mathbf{W}_2 = 1000\ \mathbf{I}_{20\times20}\ \text{s}^2/\text{rad}^2$, $\mathbf{W}_3 = 1/200\ \mathbf{I}_{20\times20}\ 1/\text{Nm}^2$, and $\mathbf{W}_4 = 2000\ \mathbf{I}_{20\times20}\ \text{s}^2/\text{Nm}^2$) and the prediction horizon ($t_f = 20$ ms) were chosen to provide reliable trajectory tracking and a replication of long-term experience.

$$J_{CNS} = \int_0^{t_f} \left[\mathbf{W}_1^T \left(\boldsymbol{\theta} - \boldsymbol{\theta}_d\right)^2 + \mathbf{W}_2^T \left(\boldsymbol{\omega} - \boldsymbol{\omega}_d\right)^2 + \mathbf{W}_3^T \left(\boldsymbol{\tau}_a\right)^2 + \mathbf{W}_4^T \left(\dot{\boldsymbol{\tau}}_a\right)^2 \right] dt \tag{8.5}$$

$$\begin{gathered} \boldsymbol{\theta}_{min}\left(\beta\right) \leq \boldsymbol{\theta} \leq \boldsymbol{\theta}_{max}\left(\beta\right) \\ \boldsymbol{\omega}_{min}\left(\beta\right) \leq \boldsymbol{\omega} \leq \boldsymbol{\omega}_{max}\left(\beta\right) \\ -R\boldsymbol{\tau}_0^-\left(\beta\right) \leq \boldsymbol{\tau}_a \leq R\boldsymbol{\tau}_0^+\left(\beta\right) \\ \dot{\boldsymbol{\tau}}_{min} \leq \dot{\boldsymbol{\tau}}_a \leq \dot{\boldsymbol{\tau}}_{max} \end{gathered} \tag{8.6}$$

where

$\mathbf{W}$	cost function weights
$\boldsymbol{\theta}$	joint angles
$\boldsymbol{\omega}$	joint instantaneous angular velocities
$\boldsymbol{\tau}_a$	active torques at the joints
$\dot{\boldsymbol{\tau}}_a$	instantaneous active torque rates at the joints
$\boldsymbol{\tau}_0$	peak isometric joint strength
R	ratio of the maximum eccentric isokinetic torque over the maximum isometric torque
t_f	prediction horizon
$_d$	desired value or trajectory
$_{min}$	minimum variable
$_{max}$	maximum variable
$^+$	positive direction of joint

Table 8.1: The cost function weights and internal model are used for three simulated conditions of (I) initial, (II) short-term, and (III) long-term experience. The 1st and 2nd scenarios are implemented with full and variable strength of the mid-level controllers, respectively (sections 8.5.1 and 8.5.2).

	Experiences	**Initial**	**Short-Term**		**Long-Term**
			1st Scenario	**2nd Scenario**	
Weights	$\boldsymbol{\omega}_1^T$ (angle)	$[39, 183, 39, 183]$	$[39, 183, 39, 183]$		$[40,000, 40,000]$
	$\boldsymbol{\omega}_2^T$ (angular velocity)	$[1234, 1234]$	$[1234, 1234]$		$[1000, 1000]$
	$\boldsymbol{\omega}_3^T$ (torque)	$[1/120, 1/70]$	$[1/120, 1/70]$		$[1/200, 1/200]$
	$\boldsymbol{\omega}_4^T$ (torque derivative)	$[1360, 1360]$	$[2720, 2720]$	$[2040, 2040]$	$[2000, 2000]$
IM	human limb dynamic	Yes	Yes		Yes
	known robot's assistive torque	No	No		Yes

$^-$ negative direction of the joint

β subject adjustment variables: sex, age, body mass, height, dominant side, and physical activity

The CNS learns and identifies the IM in the long term [272, 333], and the muscles can adapt in the short-term [272, 389, 390]. Clearly, after the initial use of a wearable robot, the CNS only has information about the IM of the human and the old weights for the limb, and has no model for interacting with the wearable robot. In short experiences, the human can adjust and tune the weights for the new condition, subject to the wearable robot's assistive or resistive torque [391]. We use the human limb model as the IM for the initial and short experiences. The IM incorporates the robot dynamics in the long-term experience. In addition, the weights ($\boldsymbol{\omega}_i$) are tuned for free limb motion only and are used for the initial experience. The weights are adjusted for the short-term and long-term experience.

The weights in the cost function (Equation (8.5)) are manually adjusted for three conditions of (I) initial, (II) short-term, and (III) long-term experience (Table 8.1). The initial experience shows the human-robot interaction but has the same weights as the human-only mode. The short-term experience is when the IM is not updated, and the weights are adjusted for the optimal motion. The long-term experience is when the robot dynamics and mid-level controller are incorporated into the IM, i.e., the human has adapted to the robot.

8.3 Active exoskeleton control

As shown in Figure 8.3, the AAN hierarchical control of the exoskeleton constructed with three levels, which are described as follows.

For simulation and mid-level assessment, the high-level and low-level controllers were assumed to be perfect and have a 1-to-1 relation. However, for experimental evaluation of the controller, they have configured as mentioned in sections 8.3.1 and 8.3.3.

8.3.1 High-level control

To estimate the human contribution torque and the future elevation joint angle, MuscleNET [38] was used. MuscleNET is a machine learning model that maps the sEMG signals, Euler angle data of IMUs (both from the measurement box/unit of Delsys Trigno wireless sEMG system), and delayed kinematics (from the motor's embedded sensors) to future joint torque ($\hat{\tau}_h$) and angle ($\hat{\phi}_d$).

8.3.2 Mid-level controller

The input variables to the mid-level controller are from the high-level controller, such as estimated human joint torque $\hat{\tau}_h$, and the system kinematic feedback measured by sensors, such as predicted/desired joint angle $\hat{q}_d$,

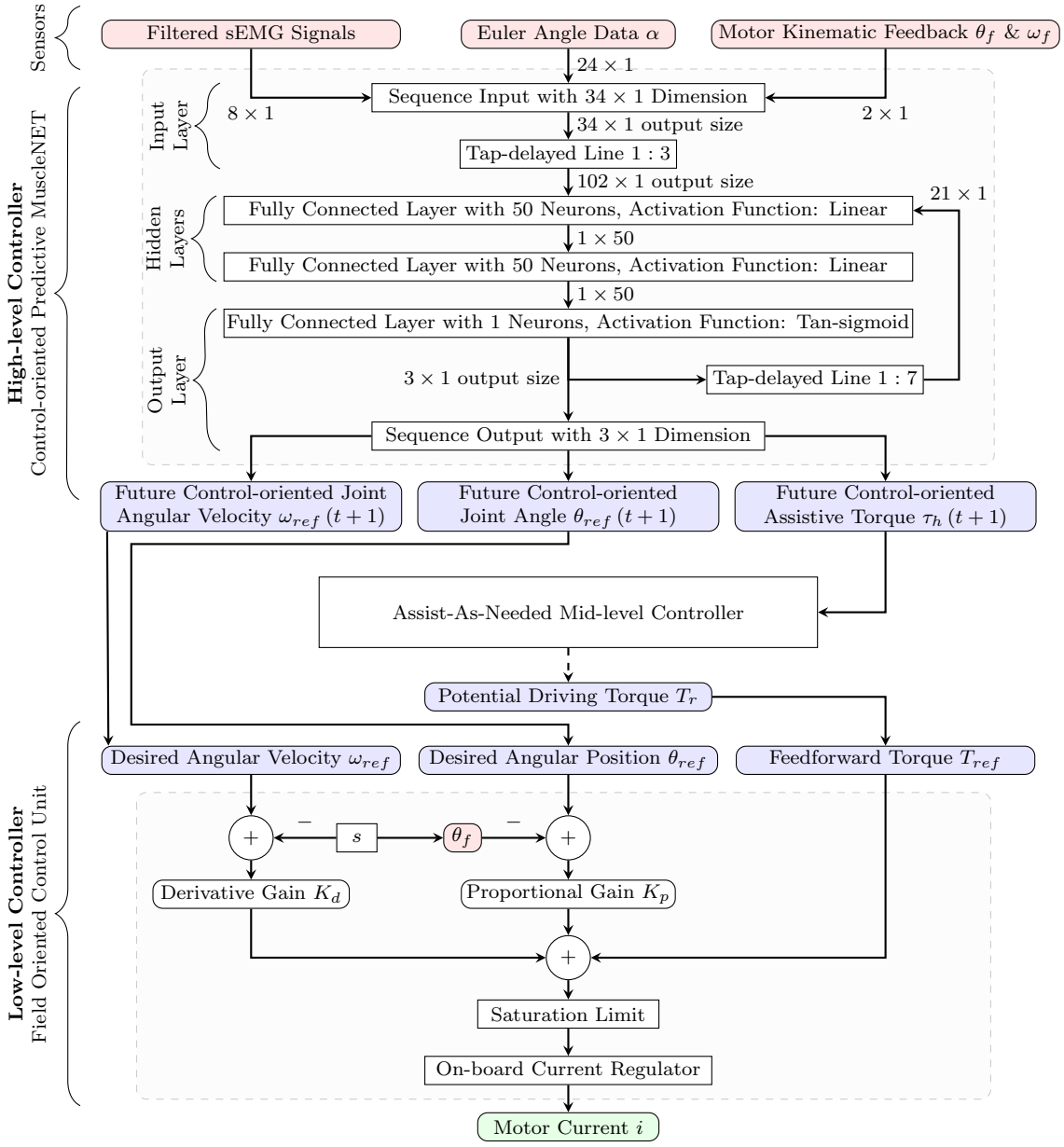


Figure 8.3: Schematic representation of the active-passive robotic exoskeleton's hierarchical control structure, including high, mid, and low-level controllers. The high-level component predicts future values of the control-oriented joint angle, angular velocity, and assistive torque. The mid-level controller determines the driving torque using the human and exoskeleton models. The low-level controller then ensures that the state of the motor matches these descending commands.

velocity $\hat{p}_d$, acceleration $\frac{d\hat{p}_d}{dt}$, feedback/measured joint angle q_f, velocity p_f, and acceleration $\frac{dp_f}{dt}$. The strength and direction of the applied wrench are defined in the mid-level controller. The user can define the strength of assistance or resistance Ω. The output of this controller is the desired robot torque T_r, which is commanded to the low-level controller.

In this research, two new AAN-CTM and AAN-fuzzy logic based mid-level controllers are introduced, considering the distinctive effect of the human and the robot's dynamic models. The common AAN-proportional rule is also applied for comparison with the above two approaches. In order for a fair comparison, the gain of all controllers is optimized.

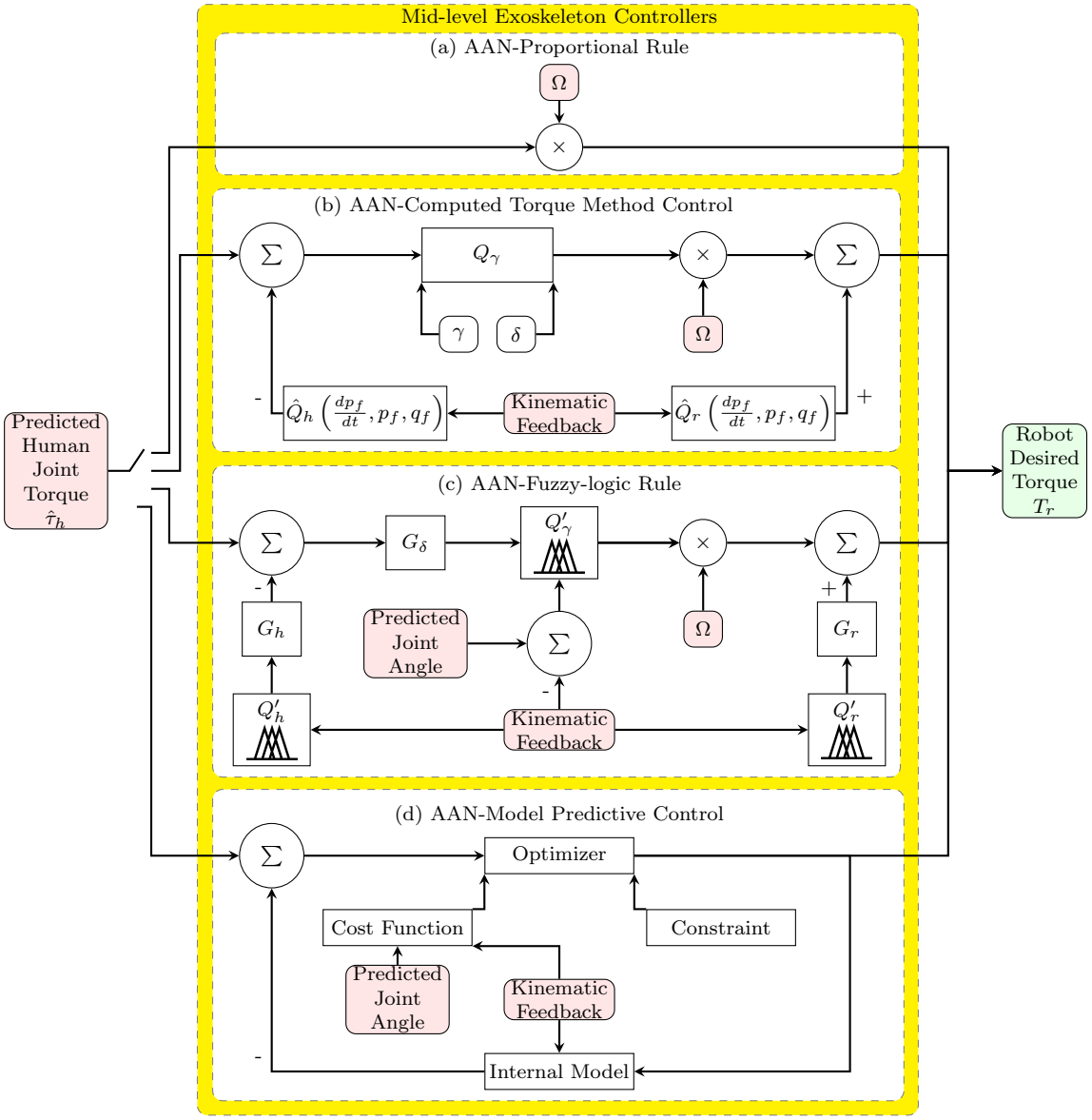


Figure 8.4: Schematic of (a) AAN-proportional rule, (b) AAN-CTM rule, (c) AAN-fuzzy logic rule, and (d) AAN-model predictive control as mid-level controller.

ANN-proportional rule

Researchers have commonly used the AAN-proportional gain method [19, 249, 250] (Figure 8.4(a)). The desired robot torque T_r is calculated as a proportion of the estimated human joint torque $\hat{\tau}_h$ (Equation (8.7)). This conventional controller does not consider distinctive human, robot, and environment system models. Directly using the human torque as the command of the robot torque causes subject discomfort in exoskeletons, as described in section 2.5.

$$T_r = \Omega \hat{\tau}_h \tag{8.7}$$

AAN-computed torque method

This novel approach uses the identified limb model and the robot model to find the net torque of the human joint without the shared portion for the limb and the robot fixture; this approach resolves the previously mentioned issues by considering the human and robot dynamics. The net torque may multiply to the desired strength variable to further increase or decrease assistance or resistance (Figure 8.4(b)). The detailed model has been presented in Equations (8.8)-(8.10).

$$T_r = \Omega Q_\gamma \tau_\delta + \hat{Q}_r \left(\frac{dp_f}{dt}, p_f, q_f \right) \tag{8.8}$$

$$Q_\gamma = \frac{1}{1 + e^{-\frac{4}{\gamma - \delta}\left(\tau_\delta - \frac{\gamma+\delta}{2}\right)}} + \frac{1}{1 + e^{\frac{4}{\gamma - \delta}\left(\tau_\delta + \frac{\gamma+\delta}{2}\right)}} \tag{8.9}$$

$$\tau_\delta = \hat{\tau}_h - \hat{Q}_h \left(\frac{dp_f}{dt}, p_f, q_f \right) \tag{8.10}$$

where

$\hat{Q}_r$	the calculated torque from the wearable robot model
$\hat{Q}_h$	the calculated torque from the human limb model (without the external load impact)
Q_γ	the AAN-CTM threshold function that eliminates the controller chattering effect. It is defined by a hyperbolic relation in Equation (8.9).
γ	the threshold value; The threshold γ should not be more than the maximum torque of the robot actuator.
δ	the dead zone; The effect of the dead zone becomes more significant as the accuracy decreases for the $\hat{\tau}_h$ and $\hat{Q}_h$ estimates.
Ω	the desired strength variable, which is negative for resistive and positive for assistive control.
τ_δ	the difference between the estimated human joint torque and the torque of the human limb model

We used a 1-DoF limb model with an elevation joint variable for $\hat{Q}_r$ and $\hat{Q}_h$ for simplicity. The 1-DoF limb model is selected so that our simulations are better aligned with our exoskeleton physical setup, which only has a shoulder joint kinematic sensor and cannot measure the elbow joint angle.

AAN-fuzzy logic rule

The AAN-CTM control method (section 8.3.2) can be implemented by AAN-fuzzy logic rules that regulate the actuator. The detailed novel model is presented in Equation (8.11) and is shown in Figure 8.4(c).

$$T_r = \Omega \acute{Q}_\gamma \left(\acute{\tau}_\delta, q_d - q_f \right) + G_r \acute{Q}_r \tag{8.11}$$

$$\acute{\tau}_\delta = G_\delta \left[\hat{\tau}_h - G_h \acute{Q}_h \right] \tag{8.12}$$

where

Ω	the desired strength variable
$\acute{Q}_\gamma$	the threshold
$\acute{Q}_r$	the robot gravity function
$\acute{\tau}_\delta$	the difference between the estimated human joint torque and the torque of the human limb model
G_r	the gain of the robot AAN-fuzzy logic model
G_h	the gain of the human AAN-fuzzy logic model
G_δ	the gain of the environment AAN-fuzzy logic model

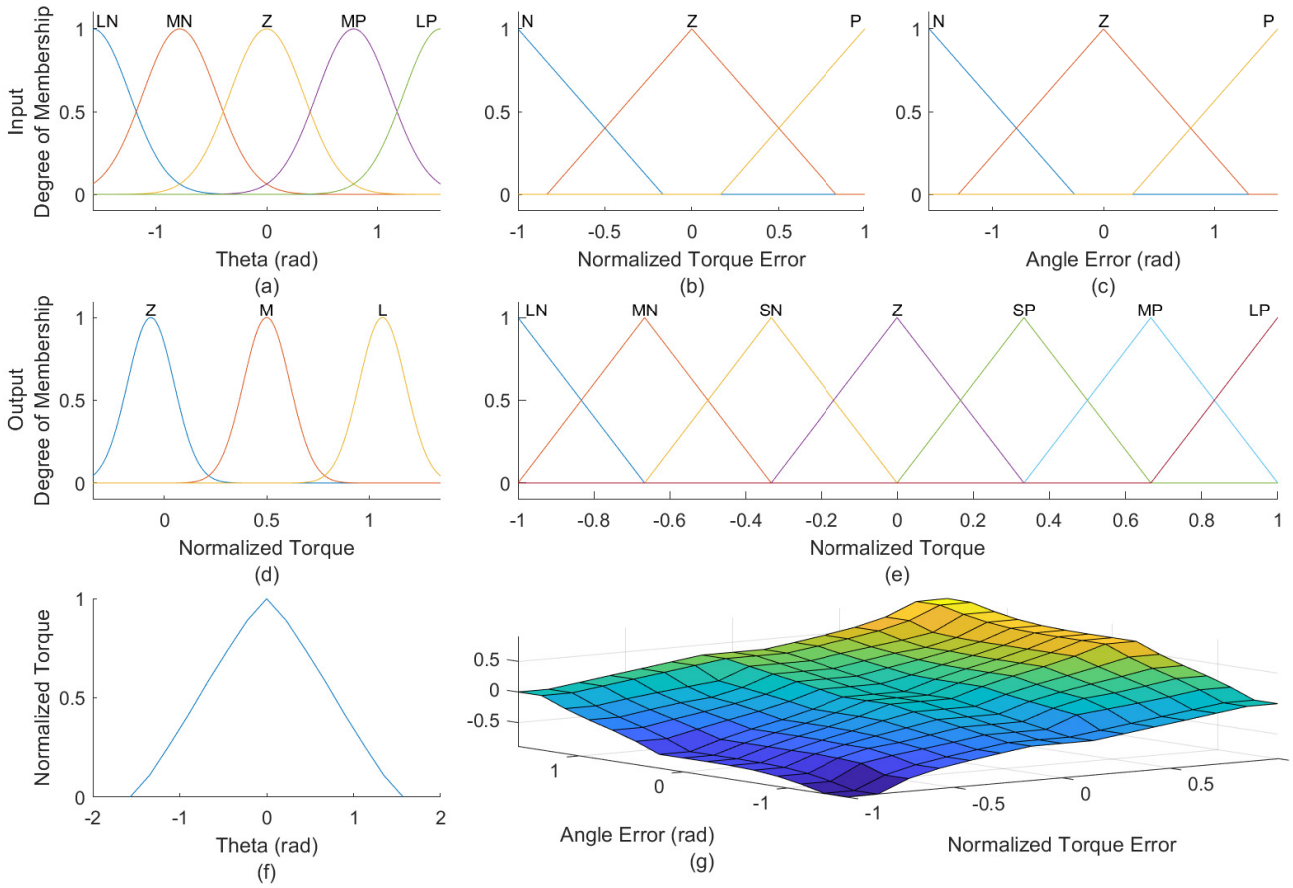


Figure 8.5: The Mamdani type fuzzy membership function and degree of membership for the two models with three inputs (a-c) and two outputs (d & e). The surface function plots of the outputs of the models for all input variables (f & g). The defuzzification technique is centroid-based. These membership functions are designed by the trial and error method.

The three functions are defined in Table 8.2, using the AAN-fuzzy logic condition-statement rules. The rules can be more complex and use more feedback, such as velocity p_f or acceleration $\frac{dp_f}{dt}$.

The degree of membership and membership function for the two models (the three inputs and two output), as well as the surface plot of the input-output functions are shown in Figure 8.5.

ANN-model predictive control

The initial idea of AAN-model predictive mid-level control was adopted from the study by Teramae et al. [258] and then adjusted to fit the strength level of the exoskeleton. Similar to Teramae et al. [258], Equation (8.13) is used to represent the exoskeleton robot's inverse dynamic model in the continuous time domain and the elevation plane. The optimal robot's input torque (T_r) is then derived by AAN-MPC. To expand, the cost to be optimized in Equation (8.14) involves the kinematic tracking error between the desired trajectory and the measured joint angle [257], as well as the kinetic tracking error between the weighted MuscleNET-estimated human torque $\hat{\tau}_h$ and the measured robot control torque. Equation (8.15) presents the inequality that represents the constraints of the optimal controller. Thus, the controller's goal is not to lower the energetic cost of action. Rather it aims

Table 8.2: AAN-fuzzy logic rule of three functions of the mid-level controller.

#	Condition				Statement
1		Large Negative (LN)			Zero (Z)
2		Medium Negative (MN)			Medium (M)
3	If q_f is	Zero (Z)		, then $\acute{Q}_r$ or $\acute{Q}_h$ is	Large (L)
4		Medium Positive (MP)			Medium (M)
5		Large Positive (LP)			Zero (Z)
6		Negative (N)	Negative (N)		Large Negative (LN)
7		Negative (N)	Zero (Z)		Medium Negative (MN)
8		Negative (N)	Positive (P)		Zero (Z)
9		Zero (Z)	Negative (N)		Small Negative (SN)
10	If $\acute{\tau}_\delta$ is	Zero (Z)	& $q_d - q_f$ is Zero (Z)	, then $\acute{Q}_\gamma$ is	Zero (Z)
11		Zero (Z)	Positive (P)		Small Positive (SP)
12		Positive (P)	Negative (N)		Zero (Z)
13		Positive (P)	Zero (Z)		Medium Positive (MP)
14		Positive (P)	Positive (P)		Large Positive (LP)

to provide some portion (Ω) of the human effort during the mentioned task to satisfy "assistance-as-needed".

$$I\ddot{\phi} + D\dot{\phi} + \Gamma_1 \tanh\left(\Gamma_2\dot{\phi}\right) + mgd_{com}\sin\left(\phi\right) - \hat{\tau}_h = T_r \tag{8.13}$$

$$J_{AAN} = \int_0^{t_f} \left[w_1\left(\hat{\phi}_d - \phi_f\right)^2 + w_2\left(T_r - \Omega\hat{\tau}_h\right)^2\right] dt \tag{8.14}$$

$$\begin{aligned} \boldsymbol{\phi}_{min} \leq \boldsymbol{\phi} \leq \boldsymbol{\phi}_{max} \\ \boldsymbol{T}_{min} \leq \boldsymbol{T}_r \leq \boldsymbol{T}_{max} \end{aligned} \tag{8.15}$$

where

I	moment of inertia of the exoskeleton's armrest around the elevation joint (0.009 kgm^2)
m	combined mass of the exoskeleton elevation mechanism, the motor's fixture, and the BLDC motor's body (1.756 kg)
d_{com}	distance of mass center from the elevation joint (0.09 m)
g	gravitational acceleration (9.81 m/s^2)
D	coefficient of viscous (dynamic) friction
Γ_i	static friction parameters
T_r	robot feedforward torque
$\hat{\tau}_h$	estimate of the human joint torque using MuscleNET and sEMG signal; it is constant over the prediction horizon
$\hat{\phi}_d$	estimate of the human joint elevation angle using MuscleNET and sEMG signal
ϕ_f	measured motor angle
Ω	desired strength variable: positive for assistive and negative for resistive control
w	cost function weights

To improve the fairness of our comparison between AAN-MPC and AAN-CTM, we simulate the pick-and-place task using a fast AAN-MPC. Briefly, the control horizon and the prediction horizon were set to 3 and 7 nodes, respectively (the time steps was 10 ms). The active-set algorithm "KWIK" was then used instead of "fmincon" to solve the optimization problem. The symbolic/analytical Jacobian of the plant was used so that the controller would not compute the Jacobians via numerical perturbations. This switch can save substantial time when iterative Jacobian calculations take place.

8.3.3 Low-level controller

We used a proportional-derivative (PD) and feed-forward controller, which is detailed in Equation (8.16), adopted from [392] and driving the BLDC servo motor.

$$T = K_p(\phi_{ref} - \phi_f) + K_d(\dot{\phi}_{ref} - \dot{\phi}_f) + T_r \tag{8.16}$$

$$\bar{T} = min\left(max\left(T, -T_{max}\right), T_{max}\right) \tag{8.17}$$

$$i_a = \frac{\bar{T}}{NK_m} \tag{8.18}$$

where

K_p	proportional gain: set to 0 $\frac{\text{Nm}}{\text{rad}}$ for torque-control mode
K_d	derivative gain: set to 1% of $\frac{\text{sNm}}{\text{rad}}$
ϕ_{ref}	desired angular position
ϕ_f	measured absolute joint angle sensor
$\dot{\phi}_{ref}$	desired angular velocity
$\dot{\phi}_f$	calculated angular speed (first derivative of the joint angle)
T_{max}	maximum nominal motor torque
$\bar{T}$	actual motor torque
N	gear ratio
k_m	motor torque constant
i_a	armature commanded current

8.4 Condition of simulation comparison

As mentioned in section 8.3.2, the exoskeleton is providing assistance and measuring the joint angle only for the human SFE and SAA. The human elbow joint has no assistance and no sensor for the robot's controller; this is to make the case study model similar to our physical setup.

An instance of the task motion and the desired trajectory (consisting of two different tasks of free motion and lifting) for an object pick-and-place in the sagittal plane subject to an external load is shown at Figure 8.6 and in Figure 8.7(a). In addition to the task-space path of the palm of the hand in Figure 8.7(a), the shoulder and elbow joint angles are obtained using the inverse kinematic geometric calculation and shown in Figure 8.7(b). The vertical force that is transferred to the palm of the hand due to gravity and the object weight is shown in Figure 8.7(c). As depicted, the force is -20.0 N starting from the grasping time and goes back to zero at release.

Different active torques of the shoulder joint in different conditions are shown using the inverse dynamic simulation of the MapleSim Biomechanics model in Figure 8.7(d). The different conditions consist of (I) not wearing an assistive shoulder exoskeleton and no presence of the external load (motion-only), (II) wearing an inactive exoskeleton with no presence of the external load, and (III) using an inactive exoskeleton subject to the external load of an object. Evidently, wearing an inactive robot requires more joint torque since the robot joint has stiffness and the robot link has inertia (the lower highlighted area in Figure 8.7(d)). Additionally, the amount of the required active torque increases in the presence of the external load. Thus, the sources of extra required torque are the dynamics of the robot (stiffness and inertia) and the environment/external load. The goal of the AAN mid-level control of wearable robots is to provide this extra required torque. Consequently, the main reason for this work is to provide a model with two distinct parts of the robot and the environment.

The criterion for assessing the proficiency of each mid-level controller is how accurately they can provide the required extra torque with the presence of the robot and environment load. If the robot provides an excessive assistive torque, the human joint will provide less torque than the no-robot condition (when the skeletal dynamics is performing the motion without the robot and the external load). This lower than normal human joint torque results in discomfort; for example, it suggests to the CNS that the inertia of the human limb has decreased from the usual amount [242].

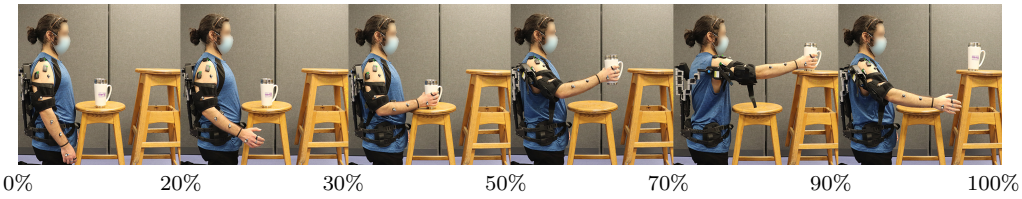


Figure 8.6: The motion capture system markers (Vicon Motion Systems Ltd, UK) and wireless sEMG sensors (Delsys Inc, MA, USA) are attached to the subject who is wearing an inactive EVO exoskeleton (Ekso Bionics Holdings Inc., CA, USA).

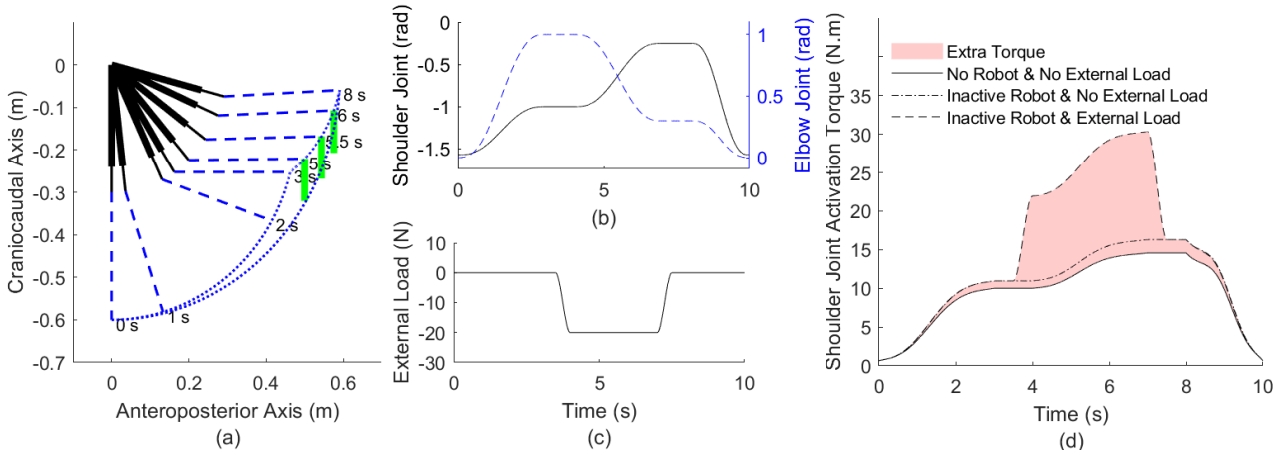


Figure 8.7: The sagittal pick and place task with active shoulder and elbow joint (a). The desired trajectory of the shoulder and elbow joint angles (b). The external vertical load of the object at the palm of the hand (c). The required active torque of the shoulder joint with and without an inactive robot using inverse dynamic simulation (d).

For a fair comparison of the mid-level controllers, each mid-level controller's gains and parameters should be optimized. The cost function of this optimization is decreasing the highlighted area in Figure 8.7(d), or simply the difference between the extra required torque in the presence of the external load and inactive robot and the condition of not wearing the inactive robot and no external load. The particle swarm optimization method has been used for this nonconvex optimization with a swarm size of 200 for each variable. The CNS-NMPC adjusts for long-term experience in this optimization.

8.5 Comparison of ANN proportional, CTM, and fuzzy-logic mid-level controller

8.5.1 Full strength (first scenario)

The three mid-level optimized controllers of (A) AAN-proportional, (B) AAN-CTM, (C) AAN-fuzzy logic rule for three different conditions of (I) initial, (II) short-term, (III) long-term experience of wearing a powered robot have been simulated (9 simulations in total). The active torques of the shoulder joint using an inverse dynamic simulation for three conditions (1. No robot and external load, 2. Inactive robot and no external load, and 3. Inactive robot with external load) are shown in Figure 8.7. The goal is to decrease both the red (extra assistance) and blue (insufficient torque) highlighted areas in Figure 8.7.

As shown in Figures 8.7 and 8.8, the desired actuation torque is equal to the condition that the human does not wear the robot and moves without experiencing the external load. Both AAN-CTM and AAN-fuzzy logic mid-level controllers have a small error with the desired actuation torque (Figure 8.9). Particularly, the actuation torques of the AAN-proportional, AAN-CTM and AAN-fuzzy logic mid-level controllers have 0.817,

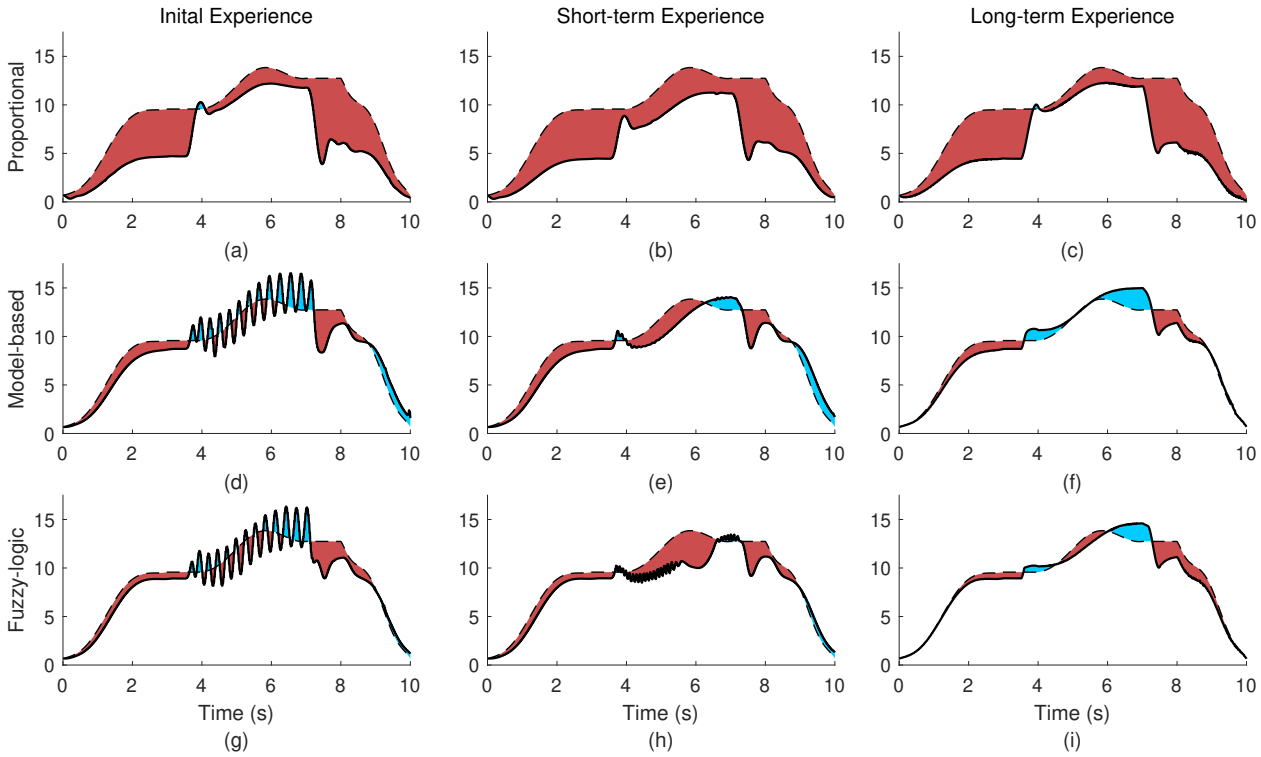


Figure 8.8: The 1st scenario (full strength) plot of the shoulder actuation torques (solid line), the desired shoulder actuation torque (dash line), the extra assistance (red area), and insufficient torque (blue area). Shown are the three mid-level optimized controllers of the AAN-proportional (a-c), AAN-CTM (d-f), AAN-fuzzy logic rule (bottom-rule) for three different conditions of initial (a, d, & g), short-term (b, e, & h), long-term experience (c, f, & i) of wearing a powered robot.

0.311, and 0.268 Nm root-mean-square error, respectively. They consider the distinctive effects of the human's dynamic model, robot's dynamic model, and the required wrench of the environment. The AAN-proportional controller only has a specific gain to the human identified torque and provides more than necessary torque. This unnecessary torque can be seen in Figure 8.8; specifically, the human shoulder actuation torque decreases too much from the normal condition (highlighted in red). This extra torque means that humans tolerate less torque than expected in the same motion without the external load. After using the AAN-proportional controller for the long term and adapting to this controller, the user feels different dynamics after taking off the robot. Thus, it takes a while for the user to adapt to the normal condition. On the contrary, if the robot only provides part of the required torque, the human should provide the remaining amount, which leads to fatigue in muscles and decreases the efficiency of the wearable assistive robot.

The errors in the AAN-CTM rule result from the 1-DoF $\hat{Q}_h\left(\frac{dp_f}{dt}, p_f, q_f\right)$ model in Equation (8.10) without having the velocity and acceleration feedback, instead of a 2-DoF human arm, to mimic the practical complexity of robotic sensing. Specifically, as mentioned in section 8.3.2, since the robot assists only the shoulder and has a built-in shoulder joint sensor, we only use the shoulder joint angle for mid-level control input. Then, $\hat{Q}_h\left(\frac{dp_f}{dt}, p_f, q_f\right)$ is considered as a 1-DoF limb with the shoulder joint kinematic data as an input variable. Practically, the 2-DoF human arm cannot be simplified to a 1-DoF system without error.

The robot's mid-level controllers should not have a negative impact on the CNS and IM. Providing too much assistive torque suggests to the CNS that the upper-limb dynamics have been changed, for example, less inertia or less stiffness at the joint [242]. This problem can cause discomfort in industry workers every time they wear the exoskeleton [243]. Additionally, this issue can cause changes in the human sensation of inertia/stiffness even after the workers remove the exoskeleton.

The AAN-CTM and AAN-fuzzy logic controllers have oscillation or vibration (velocity error) since they are

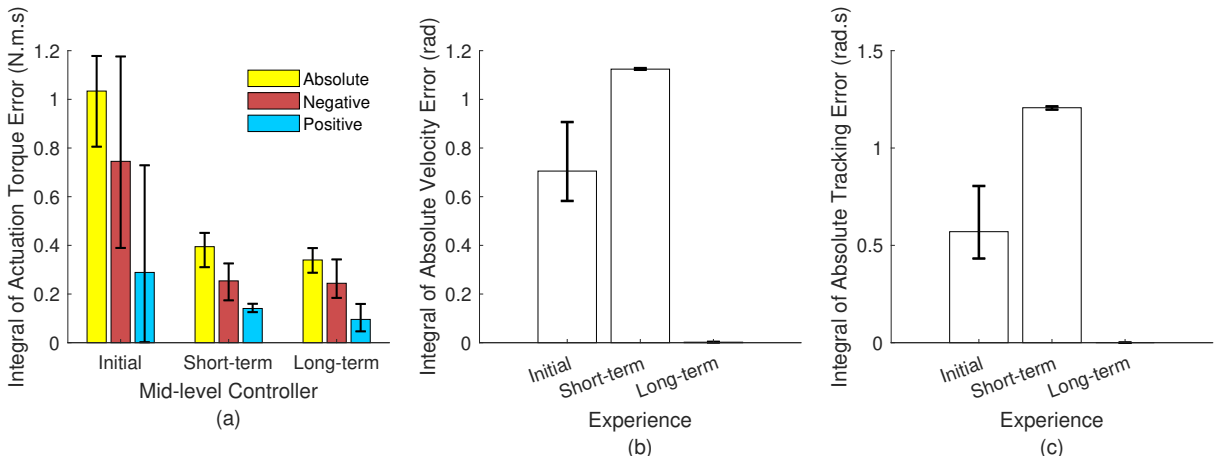


Figure 8.9: The integral of absolute, negative, and positive actuation error for AAN-proportional, AAN-CTM, and AAN-fuzzy logic controller (a); The integral of absolute velocity error for three CNS conditions of initial, short-term, and long-term experience (b); The integral of absolute tracking error for three CNS conditions of initial, short-term, and long-term experience (c).

based on the joint angle and velocity (Figures 8.8 and 8.9). The AAN-proportional controller is based on the identified torque and has no chattering. The initial experience is evaluated with the CNS-NMPC-controlled CNS, and the weights are manually tuned for a no-robot condition. As shown in Figure 8.8, assisting when the human has not learned the interaction leads to vibration in motion (here, oscillation occurs about the shoulder joint frontal axis). The short-term experience has been simulated with the same weights for the CNS-NMPC except for the $\boldsymbol{\omega}_4$ (weight of the torque derivative), which has been doubled (Table 8.1). Doubling the weight of the torque derivative means the human has activated both agonist and antagonist's muscles to have a robust motion (with increasing muscle stiffness) [393–395]. This condition results in muscle fatigue since both kinds of muscles are activated [396]. In addition, when the weight of torque derivative increases, the impact of the other weight decreases in the cost function, so the tracking error increases in the short-term experience (Figure 8.9) [397–399].

The weights in Equation (8.5) have been manually adjusted, and the IM knows the robot assistive torque in the long-term experience (Figure 8.8). Consequently, the vibration does not occur in the long-term experience. This awareness or adaption requires using the robot and motor learning for the long-term experience. One solution to mitigate the vibration in the initial experience is using a lower desired strength Ω for the robot controller in the first moments of using the robot and then increasing it after a while. This approach, i.e., changing the strength from less (initial experience) to high amount (long-term), has been experimentally evaluated in the literature [19, 400, 401], and the results have reported that the method of increasing the strength over time is preferred for human adaptation to robots. Many wearable biomechatronic device users have preferred to initiate with lower controller strength [400, 401]. Then, whenever users adapt to the wearable device or become confident with practice after a while, they gradually tend to increase the controller gains/strength.

8.5.2 Variable strength (second scenario)

To decrease the vibration about the frontal axis of the shoulder in the initial experience (shown on the left section of Figure 8.8), the simulation of controllers with 30% strength for the initial experience and 50% strength for the short-term experience (2nd scenario) is shown in Figure 8.10 (unlike the previous scenarios in which the strength was 100%). The long-term experience uses the full strength optimized for the long-term. As shown in Figure 8.10, unlike the initial experience section of Figure 8.8, the vibration does not happen. Since the vibration in the initial experience still has a minor impact, $\boldsymbol{\omega}_4$ (weight of the torque derivative) for the short-term experience is 1.5 times of the weight for the initial experience. This value was used to increase 100% (from initial to short-term experience) for the condition in which full strength has been used in Figure 8.8. Thus, muscles are relaxed, and increasing muscle robustness is not required as in the previous scenario. Note that the assistance with 30% and 50% strength does not satisfy the goal of decreasing the area in Figure 8.7(d).

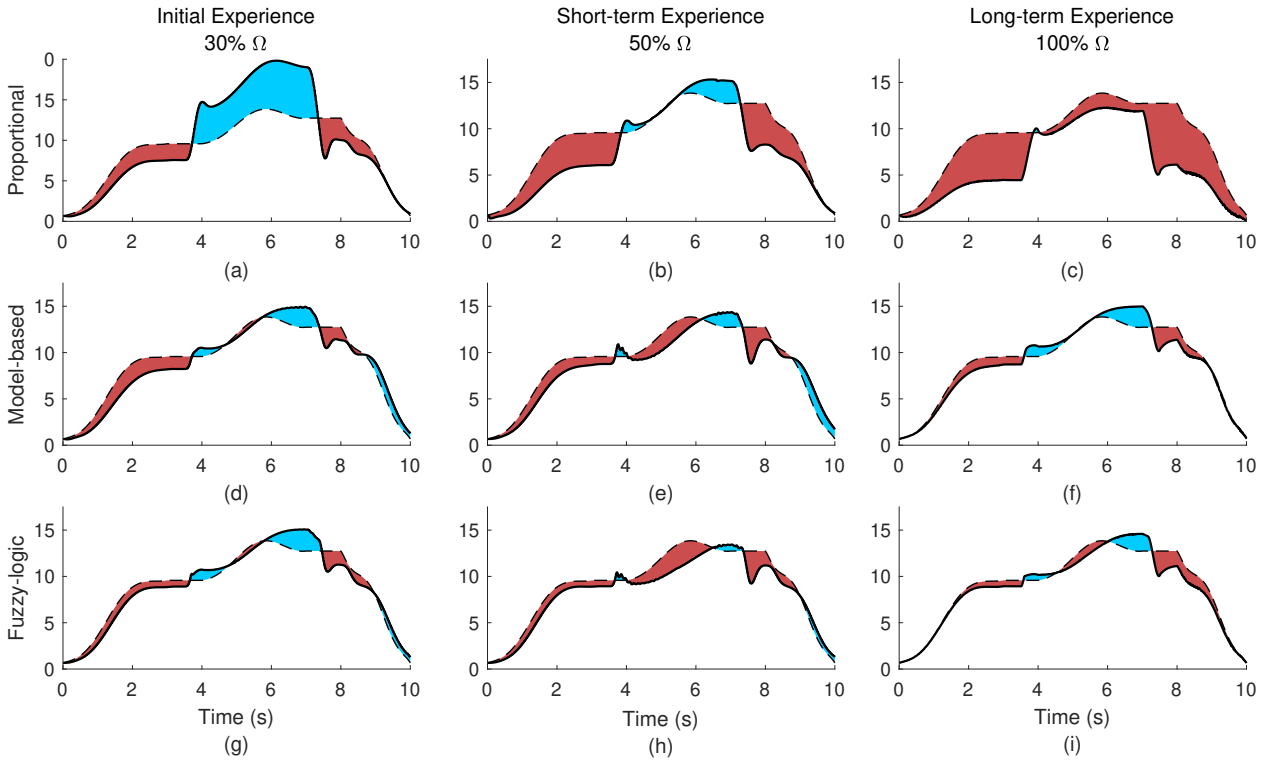


Figure 8.10: The integral of absolute, negative, and positive actuation error for AAN-proportional, AAN-CTM, and AAN-fuzzy logic controller (a, d, & g); The integral of absolute velocity error for three CNS conditions of initial, short-term, and long-term experience (b, e, & h); The integral of absolute tracking error for three CNS conditions of initial, short-term, and long-term experience (c, f, & i).

For example, in the initial experience for the AAN-proportional mid-level controller with 30% strength, the AAN is not satisfied with the mentioned criteria (the highlighted area from the desired value) 80% worse than a long-term experience for this mid-level controller. The simulated controller/model is required to provide most of the torque for handling the object for the period of $4 - 7$ s.

8.5.3 Comparison

The robot torque for both scenarios (full strength and varying strength) is shown in Figure 8.11. The desired robot torque is computed with the inverse dynamic model simulation that is highlighted in Figure 8.7(d). The AAN-proportional mid-level controller provides too much assistance for the free motion phase ($0 - 3.5$ and $7.5 - 10$ s). In the case of 30% and 50% strengths, the AAN-proportional mid-level controller's efficiency (following the desired torque) is much lower than the desired robot torque. Although the strength is variable in the second scenario, the AAN-CTM and AAN-fuzzy logic mid-level controllers' efficiencies do not decrease as much as the AAN-proportional mid-level controller for the lifting phase (3.5-7.5 s).

Another approach to reduce vibrations is relaxing the threshold value (γ) and the dead zone (δ) for the initial and short-term experiences. The result of increasing γ and δ lead to a similar simulation to Figure 8.10, which makes a new advantage for this AAN-CTM mid-level control. Instead of decreasing the strength of the mid-level controller, the threshold and the dead zone can be increased, but the strength can remain the same. A third possible solution to reducing the vibrations may be using an adaptive controller [273] or using a MPC with dual purposes of decreasing the joint error and the human-provided torque for the motion.

The AAN-CTM and AAN-fuzzy logic controllers are based on the joint angle and velocity. Both can successfully provide the assistive torque required due to external load and the robot's dynamic. In terms of comparison, the AAN-CTM controller with the capability of tuning the threshold value and the dead zone

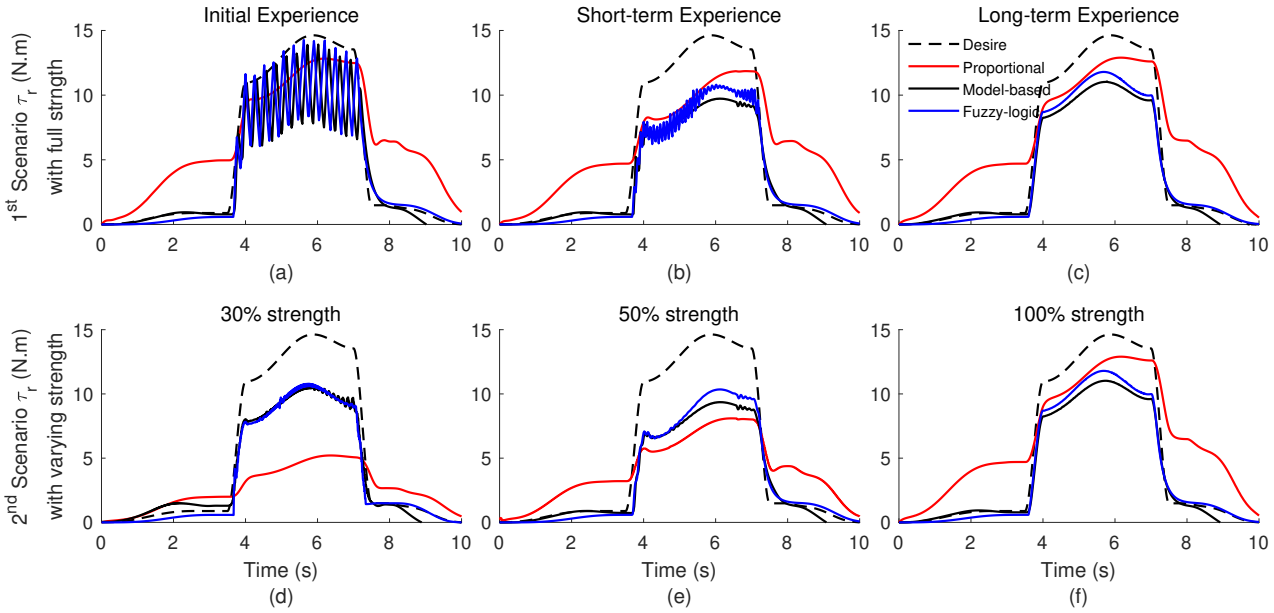


Figure 8.11: The desired robot torque is computed by the simulated inverse dynamic model (dashed), as well as the robot torque for AAN-proportional (red), AAN-CTM (black), and AAN-fuzzy logic (blue) mid-level controllers. The first scenario with full strength and optimized weights for the short-term experience (a-c), and the second scenario with varying strength of 30% for initial experience (d), 50% for short-term experience (e), and 100% for the long-term experience (f).

can slightly outperform the AAN-fuzzy logic controller. In the case of a nonlinear limb model or higher-speed motion, the AAN-CTM controller can have better results than the AAN-fuzzy logic controller, which uses linear regulation. In comparison to [252–254], our AAN-fuzzy logic controller in Equations (8.11) and (8.12) is straightforward and simpler to implement.

8.5.4 Outcome and future work

In this study, the novel AAN-CTM and novel AAN-fuzzy logic controllers are compared to AAN-proportional controllers. The idea behind the proposed mid-level controllers is to consider both the human and the robot models. Taking into consideration models of the human and the robot, as well as using a nonlinear controller (e.g., AAN-CTM, AAN-fuzzy logic, impedance control [138, 402, 403], haptic/admittance control [260, 404], and adaptive control [273, 405]) improves the efficiency of AAN control of wearable robots during two different tasks of free motion and lifting. The nonlinear controllers should be prioritized over AAN-proportional controllers for controlling the nonlinear human limb model.

So far, one major challenge of real-time control is the controller speed and computation delay [138, 406]. The proposed controllers capture the interaction between the robot and the subject, and are also computationally tractable and applicable for real-time control.

Overall, this work's distinguishing novelties and contributions are categorized into three points. (I) The effect of the human's and robot's dynamic model and the required wrench of the environment are considered in the design of the mid-level controller, for example, in Equations (8.8)-(8.10). (II) the strengths, variables, weights, and gains of an AAN wearable robot are optimized for two tasks of free motion and lifting (decreasing the area in Figure 8.7(d)). They are evaluated for three AAN-proportional, AAN-CTM, and AAN-fuzzy logic mid-level controllers. (III) the three mid-level controllers for three phases of (A) initial, (B) short-term, (C) long-term experiences of wearing a powered robot are compared and assessed. Two solutions to the vibration of controllers and muscle fatigue are evaluated: decreasing the mid-level controller strength and relaxing the threshold value and the dead zone of the AAN-CTM mid-level controller.

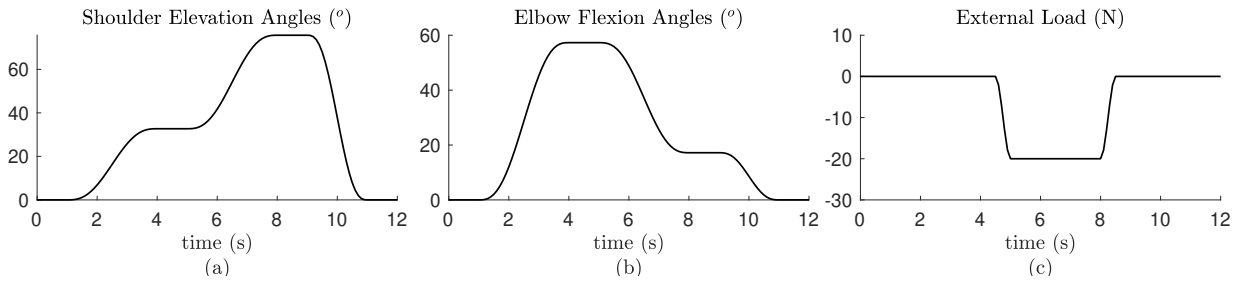


Figure 8.12: Desired shoulder (a) and elbow (b) joint trajectories for the sagittal pick-and-place task, as well as (c) the downward vertical load of the object applied at the palm of the hand. Recall that the exoskeleton provides assistive torques about the shoulder joint only.

To use this controller, an interface should be designed to capture the subject data and the desired strength variable Ω. Subject anthropometric data, including limb measurement and inertia parameters, will be incorporated in the Q_h term in Equations (8.10) and (8.12), in a future implementation. In addition, there should be a calibration scenario to adjust the high-level controller model and parameters for each subject. On-board joint angle sensor and sEMG sensor, as well as vision-based sensors, will be used to estimate the human joint torque or intention [19, 226, 407].

In a practical experiment, the noise and drift of the sensors may degrade the performance of the AAN-CTM and AAN-fuzzy logic controllers. Consequently, it is essential to make the control robust to any undesired motion [247]. In the future, the nonlinear mid-level controllers should be evaluated and enriched with a robust and safe term for human usage goals [243]. For example, to prevent rapid assistive torque changes due to noisy feedback signals, a low-pass filter or limiter can increase the robot's safety by preventing rapid assistive torque changes due to noisy feedback signals.

In rehabilitation therapy applications, the desired strength variable Ω should gradually change from a small negative value to a bigger one. In addition, visual tracking has recently been utilized to evaluate the patient's motor function recovery and the adaptability level to the therapist rehabilitation exercises [408, 409].

8.6 Comparison of ANN CTM and MPC mid-level controller

8.6.1 Computational evaluation

In the following section, the performance of the proposed mid-level AAN controllers during dynamic simulations of an upper-limb task are compared. The upper limb task involved the following sequence: free motion, reaching to object, grasping, lifting, holding, lowering, releasing, and free motion in the sagittal plane. Figure 8.12a-b depicts an example of the task motion and desired trajectory.

AAN controller gains

Unlike the model-predictive controller disseminated in Teramae et al. [258] i.e., primarily a trajectory follower, the AAN-MPC controller proposed herein mainly relies on kinetic regulation. Equation (8.14) is written so as to decrease the difference between the robot torque and Ω times the human torque. Effective use of the AAN-MPC controller should thus be reliant on the value of both function gains (w_1 and w_2), i.e., weights. In Teramae et al. [258], even with normalization, the position tracking term is weighed 100 times higher than the torque tracking term. Prior to our final simulations, we tuned the MPC gains to limit chattering in the results, while still permitting the robot to strong reasonable assistance for the human. In fact, the results of our gain tuning experiment for AAN-MPC were inverse to those reported in Teramae et al. [258]: the torque tracking (w_1) shoulder gain is approximately 100 times the magnitude of the position tracking gain (w_2).

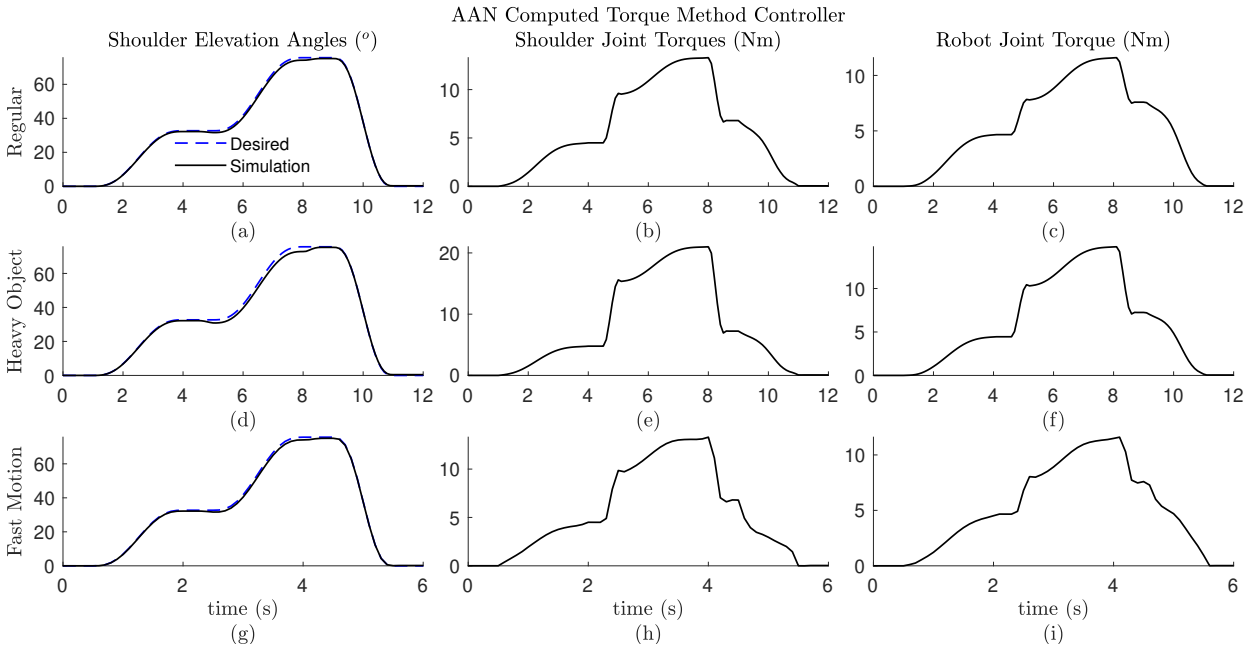


Figure 8.13: The results of AAN-CTM (with $\Omega = 100\%$) simulations with the human-robot system, where the human (CNS-NMPC model) performs an upper-limb pick-and-place task with assistance as follows: (a-c) under regular lifting conditions; (d-f) for the task with a heavier object, and (g-i) with a faster motion. Columns, from left to right, show the shoulder elevation joint angle, shoulder human joint torque, and robot assisting joint torque, respectively.

Simulation comparison

The AAN-CTM (Figure 8.13), AAN-basic-MPC (Figure 8.14), and fast AAN-MPC controllers (Figure 8.15) show relatively similar performance in the regular/baseline condition of the lifting tasks, although achieving this did require manual gain tuning for the basic and fast AAN-MPC.

To assess the performance of the proposed basic and fast AAN-MPC methods across a range of operating conditions, we performed additional simulation experiments. Each experiment involved one of the following conditions: addition of a disturbance force (object weight increased); the simulated human user wishes to move with greater speed; the MPC prediction horizon is increased (for basic MPC only).

- **Different object weight:** Although the object weight acts as an unknown disturbance for the basic and fast AAN-MPC, it can be inferred from the human contribution torque. According to Figure 8.14, the basic AAN-MPC controller tries its best to optimize the cost function, but has chattering in the torque when the human contribution torque is decreasing ($t \geq 8$ s). We suggest that the delay of the basic AAN-MPC controller is responsible for this situation. Figure 8.14d-f shows that when the user intends to lower their arm, the controller is still providing too much assistance and has not decreased its output because it thinks that the participant still desires an old set point. In Molazadeh et al. [259], the authors claim that MPC is robust to uncertainties for exoskeleton systems, which appears to stand in opposition to our own observations. In their case, they ignore issues like hardware limitations and tested their controller in an ideal situation, i.e., simulation with a powerful computer. By increasing the prediction horizon and slightly adjusting the weights of AAN-MPC (Figure 8.14j-l), the basic AAN-MPC can accomplish the goal of providing effective assistance, especially, when lowering the arm. Chen et al. [266] mentioned the requirement of gain tuning for different weights.

- **Different trajectory speed:** During high-speed motions, the AAN controller should be faster in responding to the inputs. Thus, we tested each controller with a faster reference trajectory; the simulation results are provided in Figure 8.13g-i, Figure 8.14g-i and Figure 8.15g-i. The basic AAN-MPC controller could not provide smooth assistance torque due to slow response times. One potential solution for this issue may

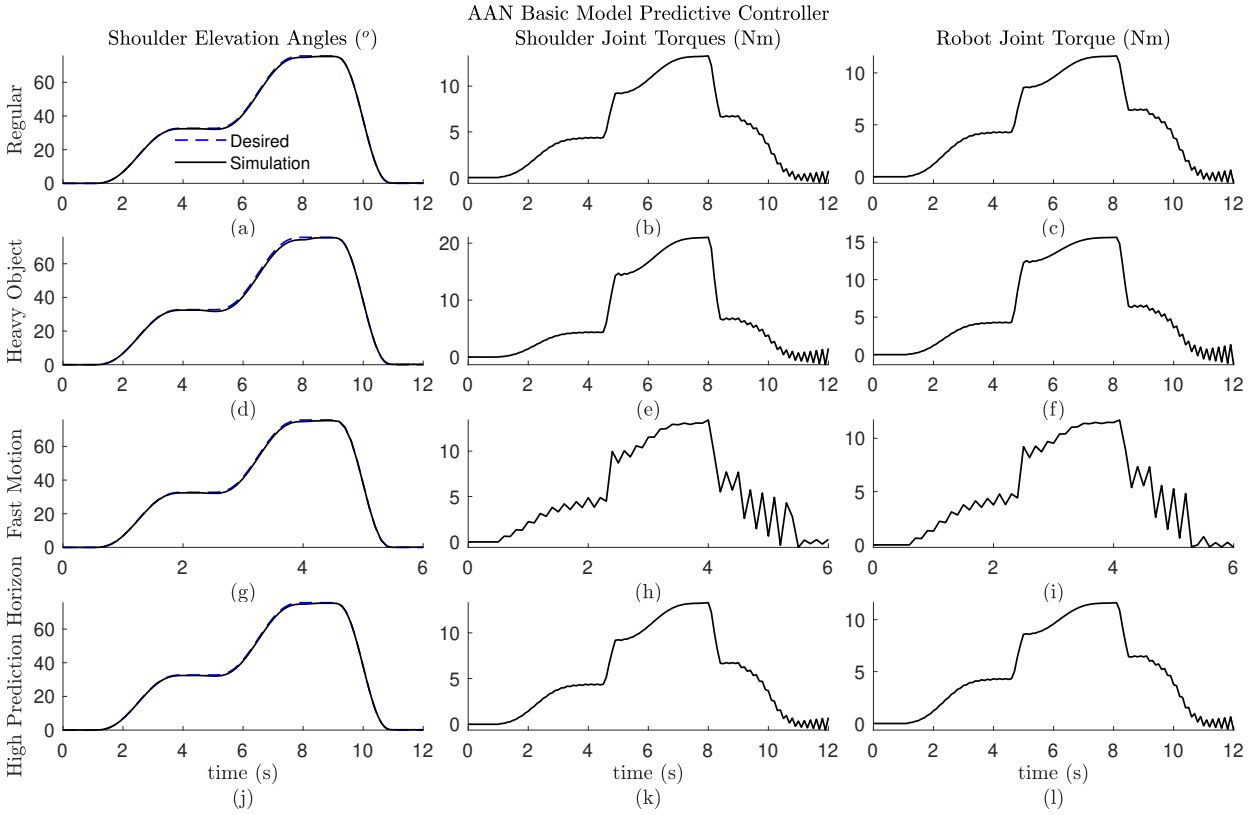


Figure 8.14: The results of AAN-basic-MPC (with $\Omega = 100\%$) simulations with the human-robot system, where the human (CNS-NMPC model) performs an upper-limb pick-and-place task with assistance as follows: (a-c) under regular lifting conditions; (d-f) for the task with a heavier object, (g-i) with a faster motion, and (j-l) with double the prediction horizon. Columns, from left to right, show the shoulder elevation joint angle, shoulder human joint torque, and robot assisting joint torque, respectively.

be adding velocity tracking or rotatum (the first time derivative of torque) to the cost function (Equation (8.14)). However, adding terms to an optimal controller carries the risk of increasing computation time. Although Rodriguez et al. [257] claimed that basic MPC is good at position tracking a slow motion, our results show that fast motions necessitate considerations like additional gain tuning to achieve this. Additionally, studies involving MPC in exoskeletons, like Cao and Huang [265], tend to test only during slow motions like walking at one stride per 10 seconds. In this case, the extremely low speeds seemingly shift the control focus from gait/propulsion to more quasi-static or dynamic balance. In Cao and Huang [265], the authors used a pneumatic actuator that does not operate at a high enough frequency to show any chattering. Likewise, the authors in [255] tested their optimal controller in simulations of sit-to-stand; however, the motion required 10 seconds (far slower than real human sit-to-stand [117]), so their basic MPC results were inherently smooth.

- **Different prediction horizon:** As seen in Figure 8.14j-l, the vibrations almost vanish when the basic MPC is predicting the IM response to commanded torques over a longer horizon. With that said, this change does introduce a greater computational cost to the controller unit. This cost subsequently introduces "lag" to the control system and may require a separate lag compensator if severe enough. To summarize, using the basic AAN-MPC for smooth real-time control requires a more powerful computer in comparison to the AAN-CTM controller.

The performance of the fast AAN-MPC in Figure 8.15 is very similar to the AAN-CTM in Figure 8.13. The only problem that still persists is the need for substantial weight tuning for fast motions. The MPC provides a balance between position tracking error and tracking torque; when the exoskeleton needs to move quickly, priority should be shifted to position error to achieve reasonable performance. Thus, in practice, a weight

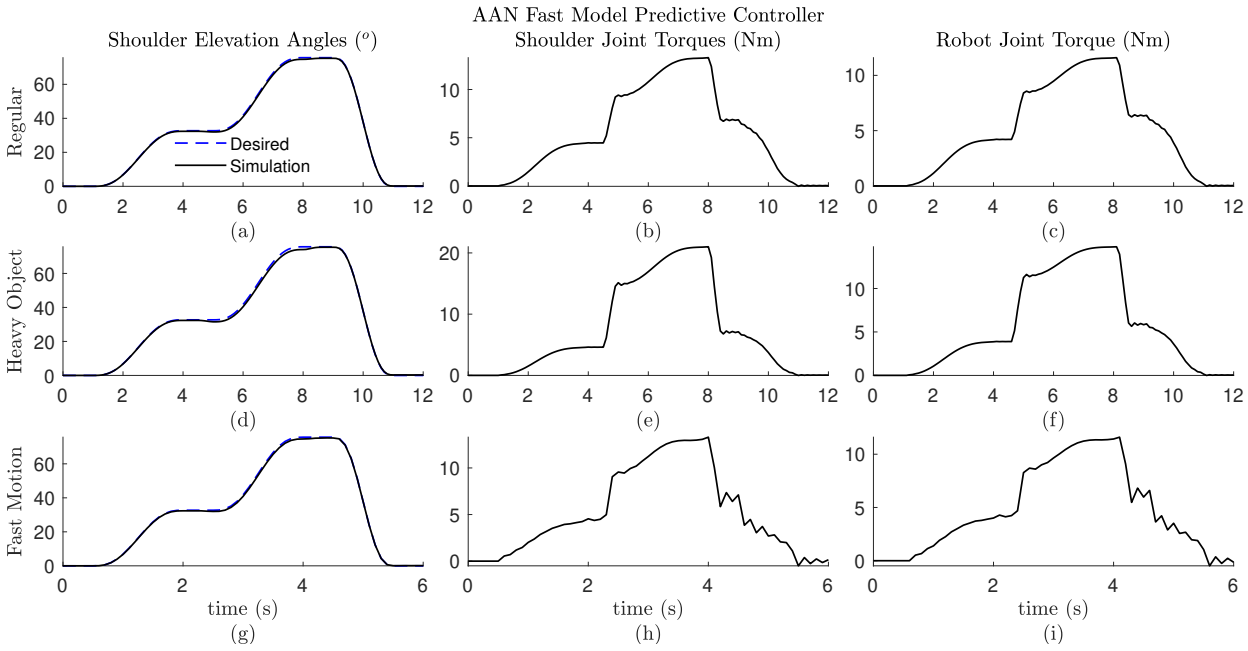


Figure 8.15: The results of fast AAN-MPC (with $\Omega = 100\%$) simulations with the human-robot system, where the human (CNS-NMPC model) performs an upper-limb pick-and-place task with assistance as follows: (a-c) under regular lifting conditions; (d-f) for the task with a heavier object, and (g-i) with a faster motion. Columns, from left to right, show the shoulder elevation joint angle, shoulder human joint torque, and robot assisting joint torque, respectively.

adjustment protocol may need to take place before a session of wearing the robotic system.

Computation costs comparison

From a computation cost viewpoint, AAN-MPC and AAN-CTM methods have notable differences. The computation cost of a controller has an inverse relation to its speed. The basic AAN-MPC operates 600 times slower than the AAN-CTM controller on the same personal computer. However, this basic method was reliant on numerical Jacobian generation within its iterative solver. The slow down due to these matrices are evident when we note that the fast MPC was at least 55 times faster than the basic MPC. In the basic AAN-MPC, the prediction horizon was set to 10 in order to capture the main arm dynamics sufficiently, but the high problem dimensionality makes this controller slower than regular controllers. Reducing the basic AAN-MPC control horizon to 5 nodes (time step = 10 ms) is long enough for the controller to manage the unstable mode without adding an excessive number of decision variables. Inherently, the dimension of the problem, and thus its complexity, scales with these horizons. The authors in Tahamipour Zarandi et al. [268] highlighted the high computation cost of basic MPC and decreased the prediction horizon to less than 7 nodes with a simpler IM than that used herein. Similar problems with high computation cost and limited available memory for basic AAN-MPC were observed in Jammeli et al. [263], even though the authors omitted the optimal controller's constraints, so they experienced difficulties with real-time implementation.

8.6.2 Experimental evaluation

A healthy young individual (Male; 25 years; 1.7 m height; 64 kg weight; right hand; 2 exercise sessions per week) free of any self-reported upper extremity injuries, signed the informed consent sheet and voluntarily donned the exoskeleton. They then performed the same pick-and-place task used in simulations, in accordance with the Research Ethics Board of the University of Waterloo (ethics clearance: #43980). The participant was instructed to pick up a 5 lbs kettlebell from a stool situated beside them, lift it, hold it, and lastly put it back on the desk; no restrictions were placed on timing (Figure 8.16). As shown in Figure 8.2(b), a Delsys Trigno wireless sEMG system (Delsys Inc, Natick, MA, USA) was used to measure muscle activations over the area of #1 UTRA, #2



Figure 8.16: The participant wearing the active exoskeleton and sensors while doing pick-and-place task in the sagittal plane.

MTRA, #3 MDEL, #4 PDEL, #5 ADEL, and #6 BRD. In real-time, the raw sEMG signals were filtered as discussed in [38, 356]. The filtered sEMGs were then used to predict the human joint angle and required torque (both in the sagittal plane) required to help the subject, using a machine learning method called MuscleNET [38]. The prediction was used as the input to one of two controllers: a CTM controller or the fast MPC. To protect the user, the BLDC motor torque was limited to ± 5 Nm.

AAN-CTM

Figure 8.17 shows the sEMGs, predicted states, actual states (indicated by subscript f in Equations (8.14,8.8,8.10, and 8.16)), and the feedforward motor torque for the CTM. As seen from Figure 8.17m&o, the prediction state by MuscleNET overlaps the actual state reasonably well. The CTM used the human joint contribution and the predicted state to define the robot assistance torque. The participant stated that they noticed the robot helping them lift the object. As a result, they reported becoming less fatigued compared to doing the same task not wearing the exoskeleton. However, the participant also stated that the robot had a small delay and felt "lagg" when they wished to lower their arm. We went through the control loop highlighted in Figure 8.1 and estimated this total delay to be 900 ms. Most of the delay was attributed to streaming sEMG data from the Delsys base, commanding BLDC motor, and monitoring real-time data. The computational delay arising from the filtering stage, machine learning, and the control framework was minor (90 ms).

Fast AAN-MPC

Figure 8.18 shows the sEMGs, prediction states, current state, and the feedforwarded motor torque for the fast MPC control method, which used the human joint contribution and the predicted state to define the robot torque. The participant mentioned that this robot strategy felt identical to the previous one (CTM) in terms of assistance. Additionally, they felt that the robot's lag was slightly smaller than that experienced with CTM.

From a control point of view, the primary source of delay is not due to actions taken by the mid-level control method. Rather, it is mainly due to the sEMG and IMU sensors recording, communication dead times, the different communication system protocols (Figure 8.1), and communications with the low-level controller embedded in the BLDC motor. Our experiments suggest that *any* substantial delay (≥ 100 ms) will make the human user feel uncomfortable, especially when decreasing the required torque. For example, when lowering the arm, the participant herein required less assistance. However, the delayed system believed that the participant still required a high amount of assistance for holding the arm against gravity. With that said, one of the advantages to predictive control is the ability to compensate for a part of the delay, especially, as we experienced, when the delay arises from the hardware used.

8.6.3 Outcomes

From a control engineering perspective, a controller that follows user intent quickly, with less lag and disturbances to the users' intended motion, while using less computations and minimizing gain tuning for different motions/conditions, is desirable. The results of our simulation experiments comparing AAN-MPC with AAN-CTM control indicate similar performance once the gains of the MPC have been tuned. Not surprisingly, the AAN-CTM controller herein shows more advantages over the optimal controller in term of base computation cost; both methods require gain tuning regardless. Once the CNS controls the human body with an optimal

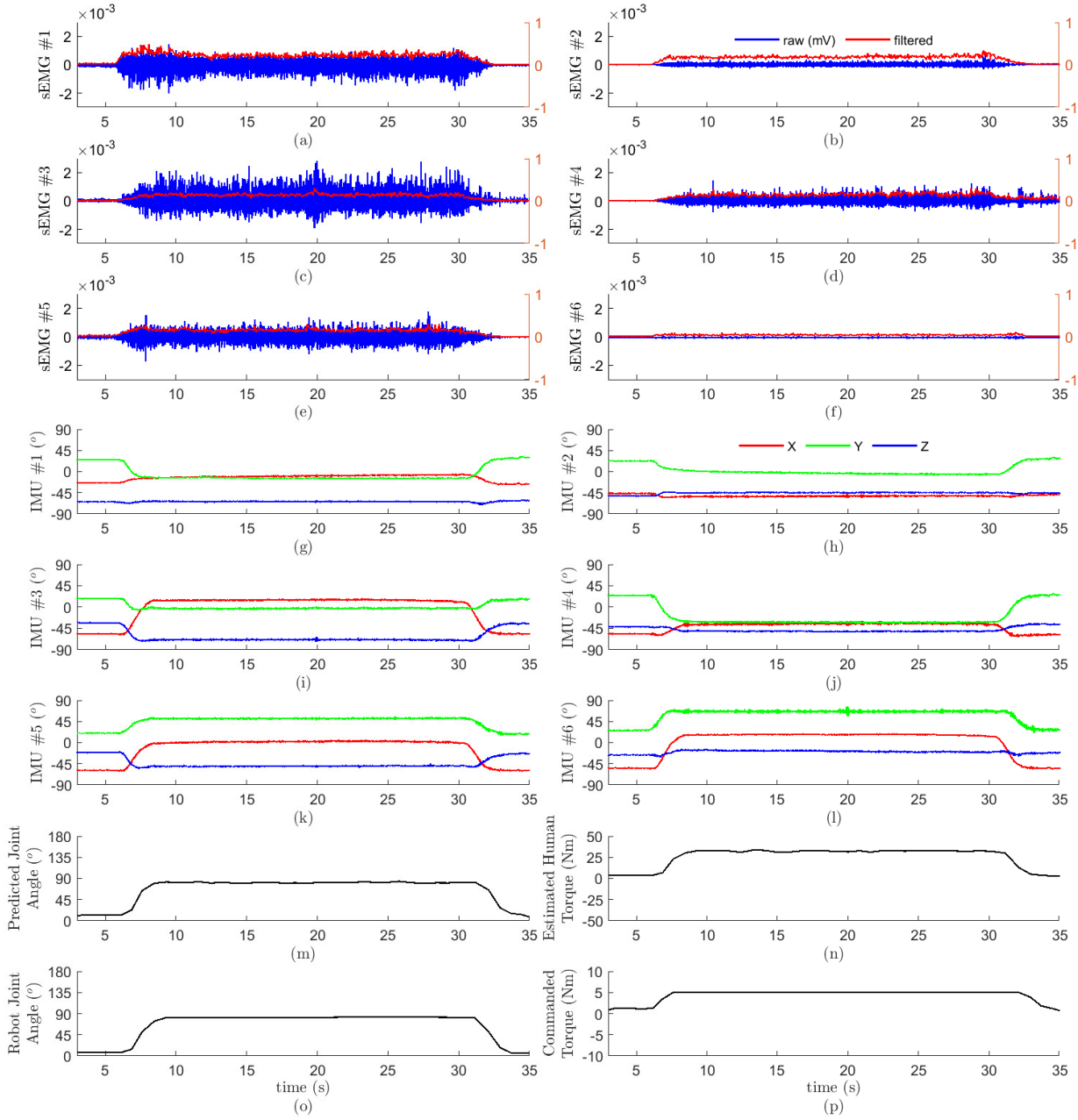


Figure 8.17: CTM experimental results; (a-f) the raw and filtered sEMG signals, (g-l) the raw Euler angles measured by IMUs, (m) predicted exoskeleton elevation angle and (n) the estimated human contribution torque for the elevation motion by MuscleNET [38], (o) the robot joint angle, and (p) robot feedforward torque.

controller (NMPC), it seems that using another optimal controller (MPC) to control a wearable robot to follow user-intent is redundant. Note that this notion applies solely for the human body, which has relatively slow but complex dynamics, and a CNS trained to work with it. Unlike the human body, a vehicle has considerably faster motions yet a simpler dynamic model; thus, an optimal controller embedded in the vehicle can control it better relative to the human CNS.

As observed in the current study, when the control delay exists primarily within the high-level controller (responsible for estimating the desired human intent and muscle contributions in a task) or within the sensor and actuator communications, the fast AAN-MPC controller managed to outperform the AAN-CTM controller. Once there is a delay in system, especially in the measurements by sensors or actuation, the MPC is able to

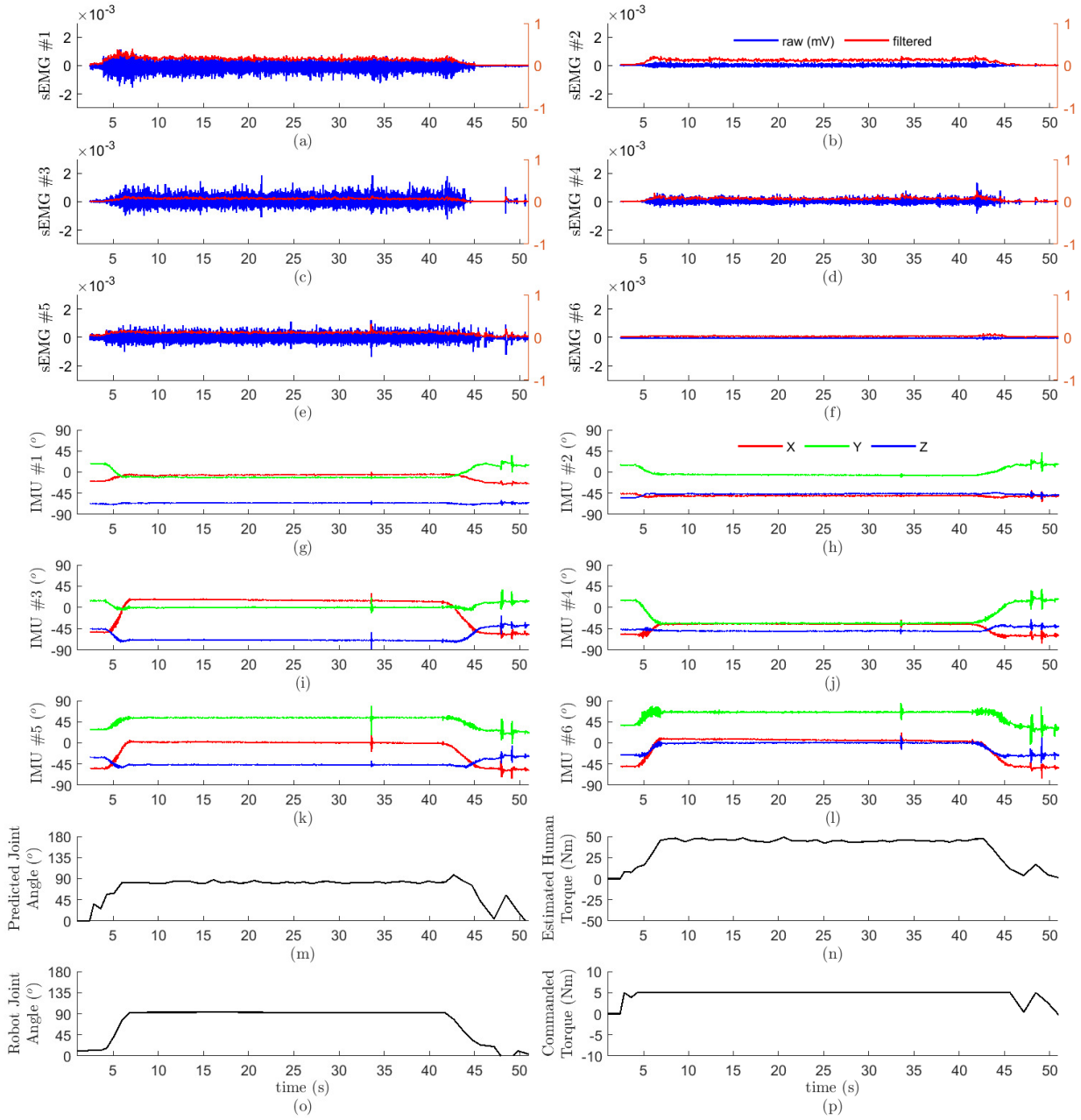


Figure 8.18: Fast MPC experimental results; (a-f) the raw and filtered sEMG signals, (g-l) the raw Euler angles measured by IMUs, (m) predicted exoskeleton elevation angle and (n) the estimated human contribution torque for the elevation motion by MuscleNET [38], (o) the robot joint angle, and (p) robot feedforwarded torque are shown.

leverage the predictions it makes over the operating time to provide more responsive intent recognition during transitions between actions. This is despite the IM assuming no delays in the input-output response.

8.7 Concluding remarks

In this chapter, intially, AAN-CTM and AAN-fuzzy logic mid-level controllers in hierarchical control of a 1-DoF active shoulder exoskeleton robot had been simulated. Then, we compared CTM and model predictive control frameworks to design an AAN upper-limb controller and evaluate HRI in simulation and experiment. The AAN

framework successfully enables the robot to offer supportive torques by detecting sEMG biomechanical signals to produce the movements desired by the user.

In designing AAN-CTM and fuzzy logic mid-level controllers, the distinctive effects of the human's dynamic model, robot's dynamic model, and the required wrench of the environment (the difference between the estimated torque in the presence of external load and the human dynamic model for free motion) were considered. The controller gains and variables had been optimized for an AAN wearable robot during a task of free motion and lifting in the sagittal plane. The CNS had been simulated with NMPC, and the three mid-level controllers for the three phases of (I) initial, (II) short-term, (III) long-term experience of wearing a powered robot, had been assessed. The results have shown that AAN-CTM and AAN-fuzzy logic controllers could outperform the AAN-proportional controller for AAN goals. Based on the results, we propose to use less desired strength in the initial usage of the wearable robots.

The outcomes of our simulation investigation indicate that when the NMPC is used to govern the user's biomechanics as a model of the CNS, both MPC and CTM control are capable of offering assistance. Both controllers were able to calculate the required torque from the subject's voluntary movements in order to follow the desired shoulder joint trajectories and provide assistance to the subject. In doing so, MPC required a high computation cost and additional weight tuning. However, MPC outperformed CTM control in the presence of delays. According to the HITL experimental test, delays in the control loop have the potential of making the users of wearable robots feel uncomfortable.

This hierarchical controller can be used in wearable exoskeletons and rehabilitation robots. In future, researchers should focus on decreasing the delay of sensors, the delay of communication, the delay of filtering/control computations, and implementation of direct drive actuators to improve the wearable devices.

Chapter 9

Experimental Evaluation of Active-passive Exoskeleton

Exoskeleton power transmission sources must be assessed using various criteria that consider both human and robot systems. To the best of our knowledge, the literature has not offered specific instructions for evaluating active-passive systems in HITL experiment. The primary goal of this project is to develop a practical test for hierarchical control of an upper limb exoskeleton designed for a particular worker or patient. The secondary goal of this project is to reduce user effort by tuning the control gains and selecting the correct strength level of the passive mechanism. Participants who were engaged with the robot provided the data for evaluation.

First, the required equipment, calibration steps, and data preparation are introduced in section 9.1. Second, hierarchical schemes are introduced, modeled, and implemented (section 9.2). Third, test protocol and phases, as well as the exoskeleton evaluation criteria, are introduced in sections 9.3 and 9.4. Finally, the results for different criteria are presented, compared, and discussed in section 9.5.

9.1 Equipment

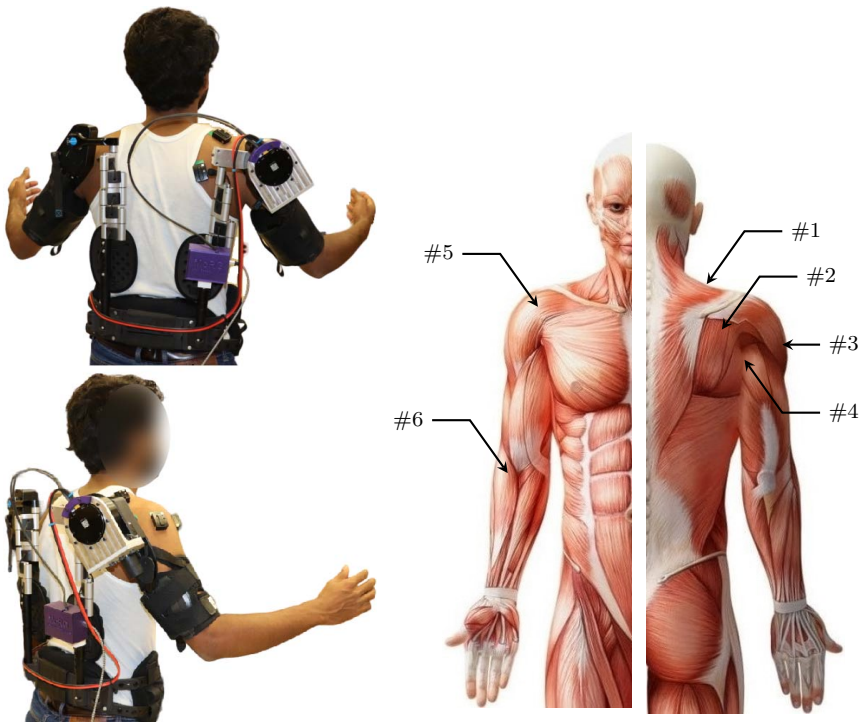
Here, the wearable passive device, the augmented active component, the sensors, and the connection protocols/methods used to assess the performance of the active-passive exoskeleton robot, are introduced.

9.1.1 Shoulder exoskeleton

We have added a motor to an EVO (Ekso Bionics Holdings Inc, California, USA) upper limb exoskeleton with built-in passive assistance (Figure 8.2(a)). The method of combining active and passive components can be applied to any upper limb exoskeleton. The EVO upper limb exoskeleton has 1-DoF for each side (left and right) of the shoulder elevation joint that rotates with a passive mechanism. Level 1, 2, and 3 springs provide a maximum torque of 4.578, 7.693, and 9.798 Nm, respectively. The exoskeleton passive joint is attached to a hip belt using passive accordion-like joints and links that allow movement in the horizontal plane. The exoskeleton armrest attaches to the user's upper arm. The exoskeleton was modeled, including its passive torque-angle function, and validated in [27].

9.1.2 Motor and encoder

An AK80-9 KV100 BLDC motor (Cubemars, Jiangxi Xintuo Enterprise Co., China) with the properties in Table 9.1 has been used for the active component. The built-in motor driver communicates with controllers through a CAN-bus and calibrates through serial communication.



(a) Human-robot system (b) Anterior view (c) Posterior view

Figure 9.1: (a) Depiction of a healthy subject wearing an actual EVO passive system with augmented BLDC motor on the elevation joint. (c & d) The depiction of the sEMG electrode and IMU placements using Delsys Trigno wireless sEMG system (Delsys Inc, Natick, MA, USA), over the area of #1 UTRA, #2 MTRA, #3 MDEL, #4 PDEL, #5 ADEL, and #6 BRD of the right forearm, shoulder, and upper trunk muscles.

Table 9.1: The properties of the AK80-9 KV100 BLDC motor (Cubemars, Jiangxi Xintuo Enterprise Co., China).

Physical		Electrical		Ratios		Driver	
Weight (kg)	0.485	Voltage (V)	24	Kt (Nm/A)	0.091	Rated voltage (V)	48
Outside diameter (m)	0.098	Rated torque (Nm)	9	Kv (rpm/V)	100	Rated current (A)	20
Height (m)	0.0385	Peak torque (Nm)	18	Ke (V/rpm)	0.009524	Current consumption (mA)	50
Number of pole-pair	21	Max output speed at rated torque (rpm)	245	Km ($Nm/w^{0.5}$)	0.2206	CAN-bus bit rate (Mbps)	1
		Rated current (A)	12	Max torque weight ratio (Nm/kg)	37	Working environment temperature (C)	-20 -+65
		Peak current (A)	24	Gear reduction ratio	9:1	Absolute encoder accuracy (bit/rev)	14
		Resistance phase to phase ($m\Omega$)	170				
		Inductance phase to phase (uH)	57				

The actuators driver board utilizes a field-oriented control (FOC) unit and is supplied with an active disturbance rejection control loop to control the speed and angle, according to Figure 9.2. This low-level controller

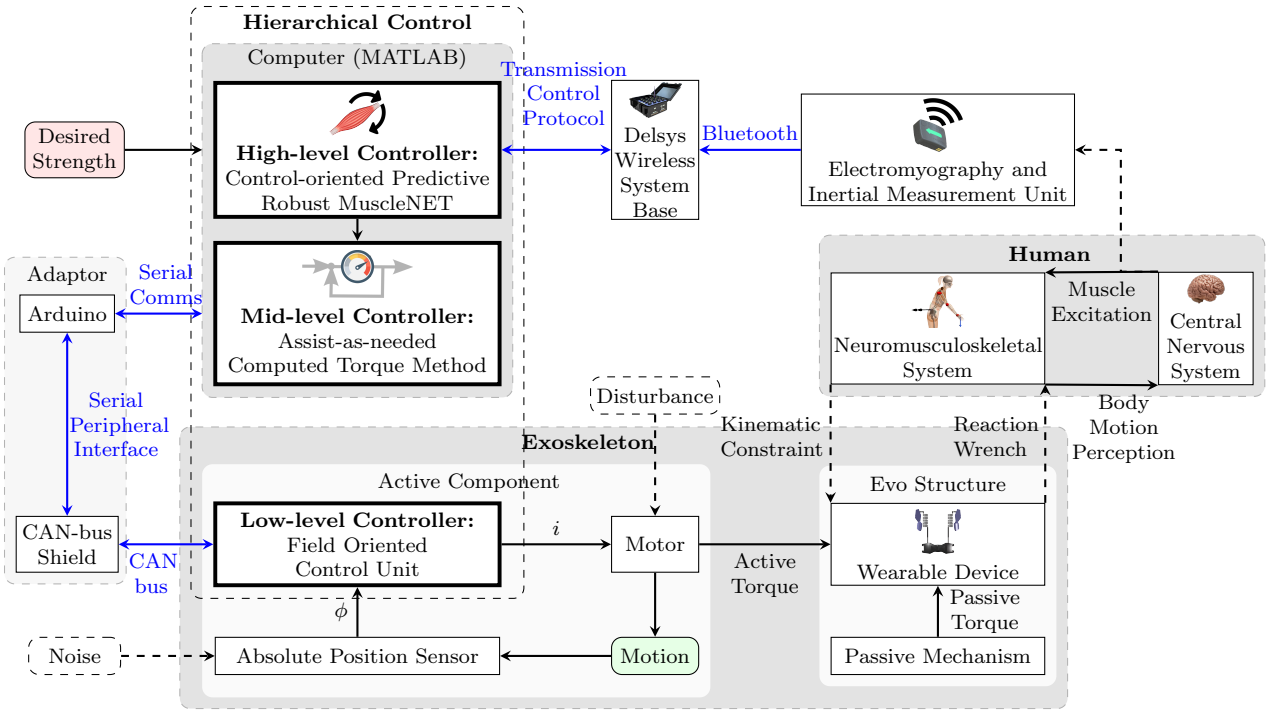


Figure 9.2: Block diagram of main control system components and connection protocols. The high, mid, and low-level controllers are described in section 9.2. The connection protocols/methods are highlighted as blue arrows.

is responsible for rejecting disturbances and applying the desired kinematics and kinetics that are commanded by the mid-level controller. This low-level controller consists of proportional-derivative feedback and a torque feed-forward controller. In total, 5 variables should be commanded to the built-in controller via CAN-bus from MATLAB. These 5 variables are: desired angular position (ϕ_{ref}), desired angular velocity ($\dot{\phi}_{ref}$), feed-forward desired torque (T_r), proportional gain (K_p), and derivative gain (K_d).

To command the variables, the motor should be entered into control mode (Table 9.2). Then, the motor can be commanded via an 8-byte data code (Table 9.2). Upon receiving a command the motor reports its sensor data via another 8-byte data code (Table 9.2).

We have designed an intermediary adaptor to connect MATLAB to the motor using an Arduino Uno and a CAN-bus Shield V2.0 (Seeed Technology Inc., Shenzhen, China). MATLAB connects to the Arduino with a serial communication port (115200 $\frac{\text{bits}}{\text{s}}$). The Arduino converts the received string to an integer for commanding the motor. To increase the robustness of the conveyed data, the data that consists of specific start and end frames, as well as fixed length, is given a special pattern. The Arduino connects to the CAN-bus board through a serial peripheral interface (SPI). The CAN-bus board includes a MCP2515 CAN-bus controller and a MCP2551 CAN-bus transceiver.

9.1.3 Surface electromyography (sEMG) and inertial measurement unit (IMU)

For the connection and streaming of real-time data between a computer and a Delsys Trigno wireless sEMG system (Delsys Inc, Natick, MA, USA), the TRIGNO software development toolkit (SDK) has been used. Delsys's wireless system base should be connected to the computer using a USB cable. The TCP, a standardized protocol for transmitting information over a network, receives the data from the base and provides the data over a virtual TCP/IP server with the trigno control utility (TCU). The Trigno SDK utilizes five TCP ports to communicate with client applications (Table 9.3). The TCU application listens for incoming connections and manages data routing to any applications that connect. All sEMG channels stream through the 50043 port, and all additional data channels stream through the 50044 port. Additionally, any sensors with 4 or fewer data

Table 9.2: Motor driver CAN-bus communication instruction.

Data fields		Data [0]	Data [1]	Data [2]	Data [3]	Data [4]	Data [5]	Data [6]	Data [7]
Special command	Enter motor control mode	255	255	255	255	255	255	255	252
	Exit motor control mode	255	255	255	255	255	255	255	253
	Set current position of the motor to 0	255	255	255	255	255	255	255	254
Control command	Data bits	7-0	7-0	7-0	7-4	3-0	7-0	7-0	7-4	3-0	7-0
	Data contents	Motor position 8 bit high	Motor position 8 bit low	Motor speed 8 bit high	Motor speed 4 bit low	Kp value 4 bit high	Kp value 8 bit low	Kd value 8 bit high	Kd value 4 bit low	Current value 4 bit high	Current value 8 bit low
Feedback data	Data bits	7-0	7-0	7-0	7-0	7-4	3-0	7-0	7-0	7-0
	Data contents	Driver board ID code	Motor position 8 bit high	Motor position 8 bit low	Motor speed 8 bit high	Motor speed 4 bit low	Current value 4 bit high	Current value 8 bit low	Motor temperature	An error code

Table 9.3: The TCP/IP server 5 port names, numbers, and functions.

Name	Number	Function		Channels	Sampling	
		Receives	Sends		Rate (Hz)	Period (s)
Command	50040	Commands	Replies to control commands	1		
sEMG Data	50043	N/A	sEMG and primary non-sEMG data from all sensor types	16	1111.111	0.0135/15
AUX Data	50044	N/A	Auxiliary non-sEMG data from all sensor types	144	148.148	0.0135/2
Legacy sEMG Data	50041	N/A	sEMG and primary non-sEMG data from select sensor types	16	1925.925	0.0135/26
Legacy AUX Data	50042	N/A	Auxiliary non-sEMG data from select sensor types	48	148.148	0.0135/2

channels, or an IMU, will have data duplicated on the 50041 and 50042 ports. A "START" command to the command port or a start trigger to the Trigno base station initiates data acquisition.

As illustrated in Figure 9.1, sEMG signals were measured from 6 sites over muscles of the right limb and trunk, similar to those suggested by Hislop et al. [350]. The locations are #1 UTRA, #2 MTRA, #3 MDEL, #4 PDEL, #5 ADEL, and #6 BRD. No sensors were attached to the upper arm area since the exoskeleton armrest covered the area.

9.1.4 Calibration

The motor command and feedback signals are defined as unsigned integer data, but we have converted them to real numbers using linear regression. The motor angle (ϕ_{ref}), the motor speed ($\dot{\phi}_{ref}$), the FOC proportional gain (K_p), the FOC derivative gain (K_d), and the motor torque (T_{ref}) were defined between 0 and 65535, 4095, 4095, 4095, 4095 bits, respectively. We have mapped the motor torque to the input code using Figure 9.3(a). According to the manufacturer manual, the motor absolute position sensor is calibrated for 4 full revolutions (Figure 9.3(b)). Additionally, the motor speed is calibrated for ± 50 $\frac{\text{rad}}{\text{s}}$ (Figure 9.3(c)).

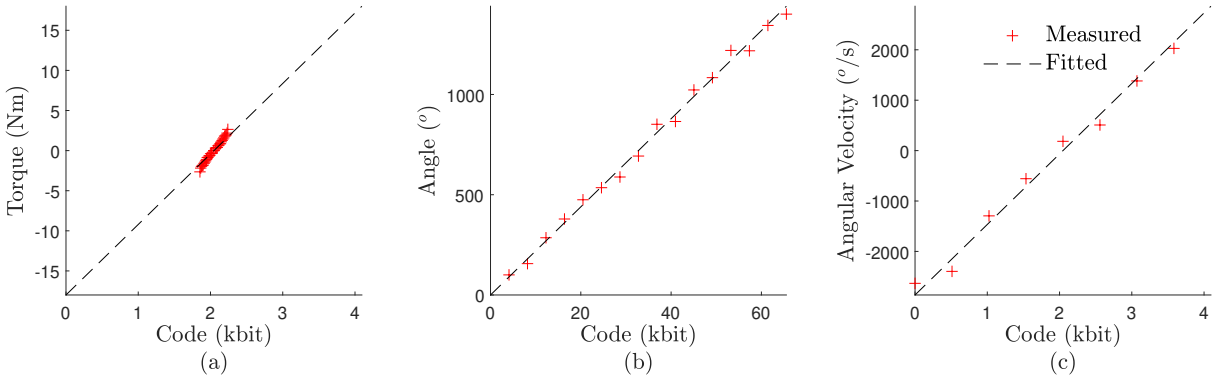


Figure 9.3: The motor torque function of code (measured and fitted) (a), the motor angle function of code (b), the motor velocity function of code (c).

9.1.5 Data preparation

The machine-learning model inputs to the high-level controller are the filtered sEMG signals, the Euler angle data from a built-in IMU in the sEMG sensors, and the motor kinematic data. These data are filtered or rectified in the preparation stage, as described below, before feeding to the high-level controller.

- **sEMG data:** The raw signals were filtered using the methods described below. In contrast to a mathematical muscle model, which requires more filtering steps and analysis, these steps are provided for MuscleNET, a machine-learning mapping method.

  i A band-pass filter with a cut-off frequency of $20 - 500$ Hz was applied [353, 354] (signals with less than 20 Hz frequency were cleaned to remove the motion artifacts, and sEMG signals with frequency more than 500 Hz were removed as they had insignificant power spectral density [355]).

  ii A band-stop filter, known as a notch filter, with a cut-off frequency of $55 - 65$ Hz was used (to clear the 60 Hz noise from the measurement unit).

  iii An absolute function was used to make the signal positive.

  iv A low-pass filter with a cut-off frequency of 7 Hz [354] was used to smooth the sEMG signal [356].

  v A normalization function to the subject's maximum signal amplitude was used.

  vi A re-sampling function using the 1-D data cubic interpolation method to a 20 Hz rate was used (only for the data preparation phase of MuscleNET, not for real-time control).

- **Euler angle data:** A low-pass filter with 30 Hz cut-off frequency was used to clear the high-frequency noise of Euler angles measured by IMU components.

- **Motor kinematic data:** To remove the high-frequency noise of the motor velocity $\dot{\phi}_f$ calculated as the numerical derivative of the joint angle, a low-pass filter with 20 Hz cut-off frequency was used. The joint angle of the motor ϕ_f was used without any data adjustment.

9.2 Hierarchical controllers

The wearable device controller adopts a hierarchical control architecture [19]. The controllers for each level are described in the following sections.

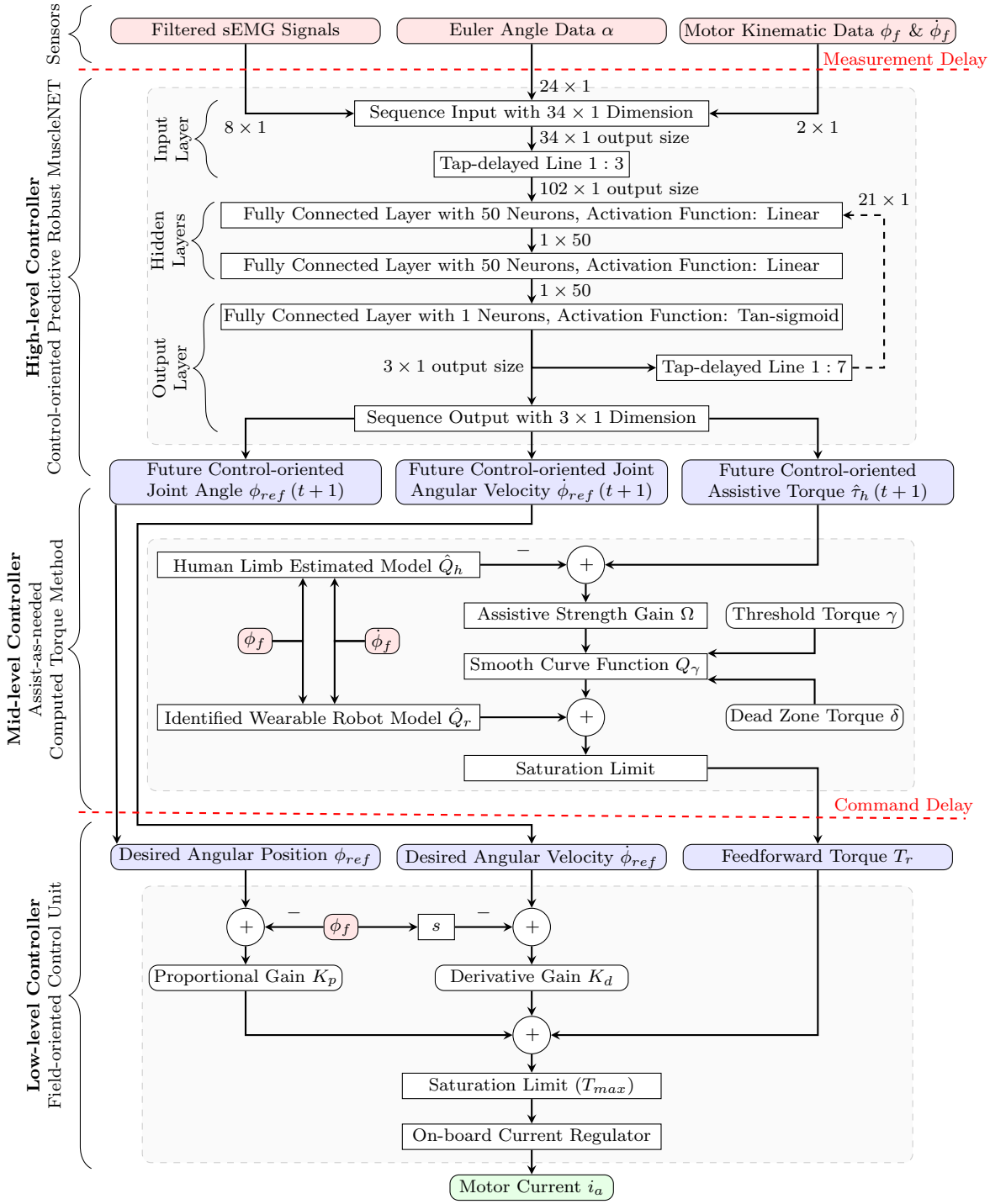


Figure 9.4: Block diagram of hierarchical control architecture of the active-passive exoskeleton robots, including high, mid, and low-level controllers. The high-level unit estimates the future values of control-oriented joint angle, angular velocity, and assistive torque. The mid-level controller defines the potential driving torque using the human and exoskeleton model. The low-level controller ensures the commanded data matches the motor states.

9.2.1 High-level controller

The high-level unit estimates the user's state and predicts the user's motion or wrench intent [38, 382]. Researchers are working on developing automated estimation and prediction systems that use machine-learning

models and information from IMUs and sEMG wearable sensors. This is in contrast to commercial devices, which use both direct and manual commands from the user for decision-making [19].

The machine-learning model used here as the high-level controller is called "MuscleNET", and its configuration is optimized for mapping sEMG to kinematic and kinetic biomechanical signals [38]. A RNN configuration (shown on top of Figure 9.4) has 34×1 dimension input, 3×1 dimension output, 3 input signal former values, 7 output signal former values, 2 hidden layers, and 50 nodes in each hidden layer.

Here, the input of the RNN is a fusion of three types of signals: muscle activation from filtered sEMG, Euler angles from IMU, and the exoskeleton joint angle (and its derivative to define joint angular velocity) by built-in angle position sensors. The output of the RNN consists of future control-oriented joint angle, angular velocity, and assistive torque. Shifting and dynamic model-based optimization loops were used to prepare the output training data. The motor joint angle and velocity are shifted in order to find two of the output training data. The last output training data was calculated using a validated exoskeleton and human MSK model [27] and an optimization loop. The optimization loop tries to find the actuator's assistive torque to decrease the human joint torque [26, 27].

9.2.2 Mid-level controller

The mid-level unit transforms the high-level estimation and prediction into mode-specific reference trajectories, or the desired wrench using the biomechatronic system's dynamic equations. In the past, this level of control was one of the conventional control methods: finite state machines/prerecorded motion [247], master-slave [40], proportional [250], CTM [28], fuzzy-logic [28, 252], impedance control [402], haptic/admittance control [404], and adaptive control [273].

Recently, positive results for the mid-level controller have been demonstrated using the AAN-CTM approach, which augments rather than replaces muscular activity [28]. Here, we consider the complete AAN-CTM strategy intended to deliver the necessary assistance torque. This novel method determines the desired actuator torque (T_r) by increasing the human active torque (τ_h) using the desired strength variable (Ω), and integrating the powered actuator model for the robot in Equation (9.1). In the presence of low human joint torque, the hyperbolic Equation (9.2) eliminates the controller chattering effect. The exoskeleton dynamic model of actuator torque (Q_r) is presented in Equation (9.3), and sums with the desired torque to cancel the dynamics of the robot structure and the actuator.

$$T_r = \Omega Q_\gamma \left(\hat{\tau}_h - \hat{Q}_h\right) + \hat{Q}_r \tag{9.1}$$

$$Q_\gamma = \left| \frac{1}{1 + e^{-\frac{4}{\gamma-\delta}\left(\hat{\tau}_h - \frac{\gamma+\delta}{2}\right)}} - \frac{1}{1 + e^{\frac{4}{\gamma-\delta}\left(\hat{\tau}_h + \frac{\gamma+\delta}{2}\right)}} \right| \tag{9.2}$$

$$\hat{Q}_r = N^2 \hat{I}_m \ddot{\phi}_f + N^2 \hat{b}_m \dot{\phi}_f + \hat{m}_r g \hat{d}_{com} \sin\left(\phi_f\right) \tag{9.3}$$

where

T_r	desired/feedforward assistive torque
$\hat{\tau}_h$	human predicted elevation torque by the high-level controller
$\hat{Q}_h$	human limb estimated model
Ω	robot's desired strength variable, which is positive for assistive and negative for resistive control between 0 and ∞
Q_γ	smooth curve of the human elevation torque between the dead zone and the threshold torque
$\hat{Q}_r$	wearable robot actuator model, the result of the rotor inertia and the viscous damping
γ	the positive threshold torque
δ	the positive dead zone torque
ϕ_f	actuator joint angle
$\dot{\phi}_f$	actuator joint angular velocity that is calculated with the derivative of the joint angle
$\ddot{\phi}_f$	actuator joint angular acceleration that is calculated with the derivative of the joint angular velocity



Figure 9.5: The participant is wearing the active exoskeleton and sensors while doing a weight-lifting task in the sagittal plane.

N	the gear ratio
I_m	moment of inertia of the motor rotor around the elevation joint
b_m	rotator viscous friction coefficient
$\hat{m}_r$	combined mass of the exoskeleton elevation mechanism, housing, and fixture of the motor
g	gravitational acceleration
$\hat{d}_{com}$	distance of mass center from the rotational joint

9.2.3 Low-level controller

The low-level agent computes the error between the measured states (from the kinematic and kinetic sensors) and the desired device states (from the mid-level controller) using common control algorithms like PID control. It then instructs the robotic actuators to use reference tracking and closed-loop feedback control to minimize the error [19]. Here we used PD-based and feed-forward controllers adopted from [392], and provided in Equation (9.4). The maximum motor nominal torque limits the desired assistive output torque in Equation (9.5). A motor's torque-producing ability relevant to its current is usually expressed as Equation (9.6).

$$T = K_p(\phi_{ref} - \phi_f) + K_d(\dot{\phi}_{ref} - \dot{\phi}_f) + T_r \tag{9.4}$$

$$\bar{T} = min\left(max\left(T, -T_{max}\right), T_{max}\right) \tag{9.5}$$

$$i_a = \frac{\bar{T}}{NK_m} \tag{9.6}$$

where

T_{max}	the maximum nominal motor torque
$\bar{T}$	the actual motor torque
k_m	the motor torque constant
i_a	the armature commanded current

Since the angular velocity is calculated using the first numerical derivative of the joint angle and it has no filter in the FOC, the controller is consequently noisy. Thus, it is reasonable to have a small K_d for a robust controller, but it should nonetheless be more than zero to inhibit any small speed errors. As the proportional gain K_p increases, the robot controller acts as a motion controller, and as it decreases, it acts as a torque controller.

9.3 Participants and test protocol

- **Participants:**

  21 participants (12 male and 9 female; 25 ± 3.2 years; 64 ± 12.9 kg mass; 1.75 ± 0.09 m height; 2.5 ± 1.9 workout session per week; 15 right-handed and 5 left-handed) free of upper extremity injury gave informed consent and performed the following tasks (Figure 9.5) in different phases. The test protocol has been given ethics clearance by the Research Ethics Board of the University of Waterloo (REB: #43980). Since the results are consistent, the task is a single axis, we have not invited more volunteers.

Table 9.4: The 6 phases of exoskeleton calibration, data-gathering, and test protocol.

Phase	Name	Exoskeleton Structure	Assistance Source		Tasks	
			Passive	Active	Weight Lifting	Free Motion
1	Sensor Calibration	✓				✓
2	Data-gathering	✓			✓	✓
3	Inactive Exoskeleton	✓			✓	
4	Fully-passive	✓	✓		✓	
5	Active-passive	✓	✓	✓	✓	
6	Fully-active	✓		✓	✓	

- **Phases:**

  Participants did tasks during the phases detailed in Table 9.4. In the first phase, the participant freely moved their upper arm from the natural pose to overhead, as well as contracted the muscles individually. This data was used for motor home angle calibration and calculating the maximum sEMG signal amplitude. In the second phase, participants made the free motion. The data in this phase was used to optimally calculate the assistive torque [27] and provide training data for the MuscleNET model [38]. The subject would repeat the task for phases 3-6 to assess the different sources of assistance (inactive/none, fully-passive, fully-active, and active-passive assistance).

- **Tasks:**

  Participants performed two tasks of weight lifting and free motion in the sagittal plane: 1) while wearing the exoskeleton and 2) not wearing the exoskeleton. For the weight lifting task, participants lifted a 2.26 kg dumbbell off a stool to about a 90^o elevation angle for the shoulder joint. The participants kept the weight at the shoulder height level until they became fatigued. The duration for tolerating the weight was recorded for each phase. Moreover, in phase one, at which data was gathered for training MuscleNET, participants were requested to keep the elbow angle at 0^o or 90^o.

9.4 Evaluation criteria

The main goal of this study was to compare the performance of the active-passive exoskeleton with the traditional passive or active exoskeleton. To this end, we applied the following three methods: survey questionnaire, experimental data, and computed data. For the computational criteria, we used the MapleSim Biomechanics (scalable MSK) model, which was developed and validated using experimental dynamometer data [27].

- Post-test survey:
  This survey aims to determine the user's experience with different actuation situations. Each participant completed a survey relevant to muscle fatigue for each phase once the test had been finished. The user reported the level of fatigue, location of fatigue, and comfort level with the exoskeleton. The results were qualitatively compared.

- Load tolerance duration:
  The participants were required to lift dumbbells as frequently as possible. The duration of this weight tolerance was recorded for phases, and then quantitatively compared.

- Mean power frequency of initial raw sEMG:
  The purpose of the sEMG frequency domain analysis is to detect the variation characteristics via short-fast Fourier transform decomposition of the sEMG signal into signal components at various frequencies. Initial sEMG mean power frequency is used to compare muscles in different phases.

- Average fatigue of motion using sEMG signals:
  Muscle fatigue accumulated during motion is given by $\sum_m \int_T \sigma_m^2 dt$, where the square of muscle activations σ is a fatigue measure widely used in neuromechanical models [410, 411]. Here, sEMG signals are assumed to be the muscle activations.

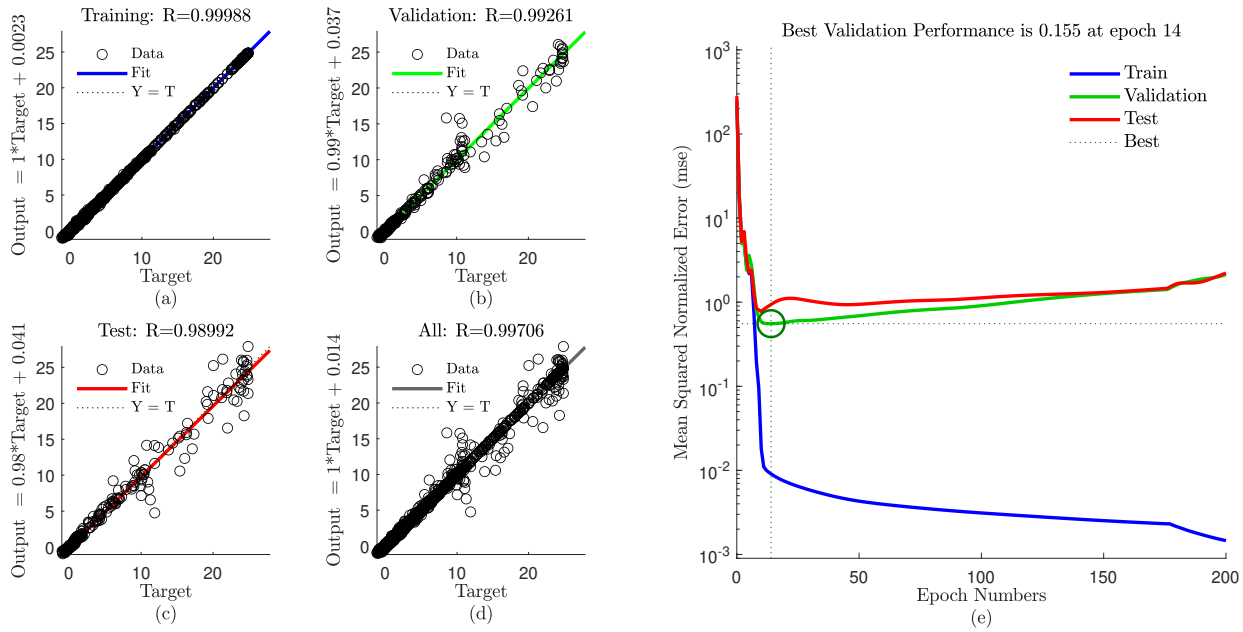


Figure 9.6: The regression accuracy of MuscleNET in MSE for (a) training, (b) validation, (c) test, and (d) all data. (e) The training performance of MuscleNET versus epoch number.

- Average computed fatigue of motion using inverse dynamic simulation and static optimization:
  Muscle activation can be calculated using the static optimization, the dynamic model's inputs of recorded joint kinematics and external force/weight.

- Average computed human active torque and absolute power using sEMG signals and MuscleNET:
  The amplitude and pattern of sEMG channels differed for each phase and each participant. In addition to mentioning their pattern difference for each phase, we used the trained machine-learning model, MuscleNET, to estimate the current shoulder elevation torque. In addition to active joint torque, the absolute joint power was calculated and compared.

- Average computed human joint torque and power using inverse dynamic simulation:
  Another method of calculating joint torque and power is using a dynamic model. The joint torque and power can be computed using the recorded joint kinematics and external force/weight as inputs to the dynamic model.

- Average computed metabolic energy expenditure using muscle torque generators:
  The last evaluation benchmark is computed MMEE. The MMEE model verified by [381] helps in the performance measurement of human motion. Kim and Roberts [381] developed a joint-space numerical model of metabolic energy expenditure by merging thermodynamic laws with principles of multibody system dynamics.

9.5 Results and discussion

9.5.1 High-level controller accuracy

We found that the accuracy of the high-level machine-learning model was higher when we used a subject-based model instead of a general model since there were differences between the NMSK of each participant, as highlighted in [38]. The accuracy of subject-based MuscleNET for all subjects was more than 99.5%, 99.2%, and 98.9% MSE regression accuracy for training, testing, and validation using randomly separated 70, 15, and 15% of data, respectively (Figure 9.6(a-d)). Using a variety object weights in the data-gathering phase helps increase

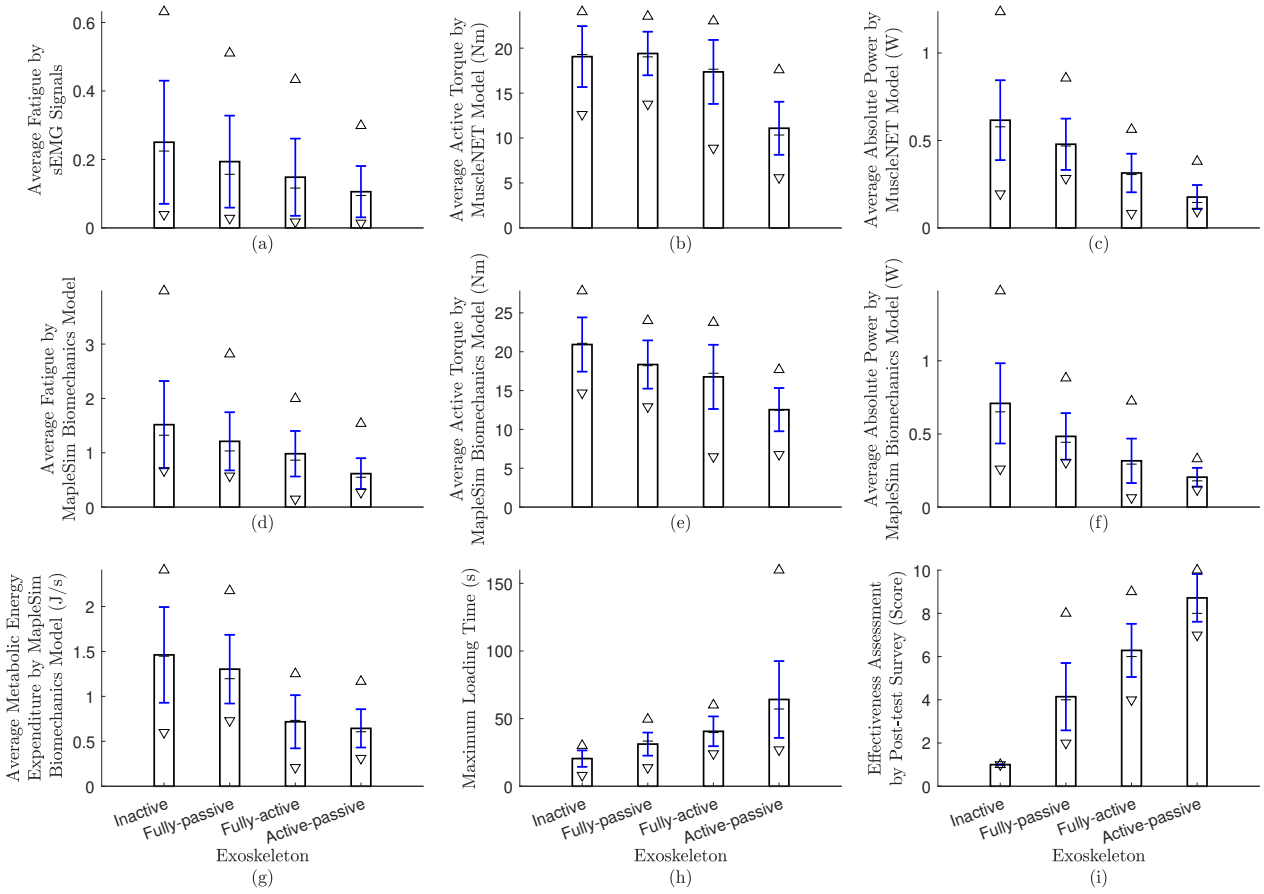


Figure 9.7: The quantitative evaluation values for the four exoskeleton actuation types (inactive, fully-passive, fully-active, and active-passive). Note that the maximum ($\triangle$), minimum (∇), median ($-$), and STD (blue line) are shown on the average of all subjects (bar charts). (a-c) are calculated from sEMG signals; (d-g) are calculated using the inverse dynamic MapleSim Biomechanics model and measured kinematic data. (a & d) shows average fatigue; (b & e) are showing average human flexion active torque; (c & f) shows average absolute power; (g) displays the average muscle energy expenditure; (h) shows the maximum duration time of load tolerance; and (j) displays the result of the post-test personal survey on the effectiveness of the exoskeletons.

the high-level controller's accuracy. However, the tests were limited by the maximum 1-hour battery lifetime of the sEMG sensors and the users exhibited fatigue before the actual exoskeleton test. The best MuscleNET performance occurred with epoch #14 and too much training results in over-fitting (Figure 9.6(e)).

9.5.2 Low-level controller gains

Since the participant lifted a weight during their motion, the low-level controller should be more concerned about the assistive torque than the position. In other words, the task was more interested in providing the assistive torque instead of having an accurate tracking error. In addition, the accuracy of the high-level controller is better for future control-oriented assistive torque in comparison to the joint angle and angular velocity. Thus, by trial and error with one of the participants, we determined that the proportional gain (K_p) should be set as small as 5 % of the maximum value. Setting the derivative gain (K_d) to a value of 20 % of the maximum value helped decrease the actuator's vibration because of the angle tracking error. In total, these gains changed the low-level controller to a direct torque plus a vibration cancellation controller.

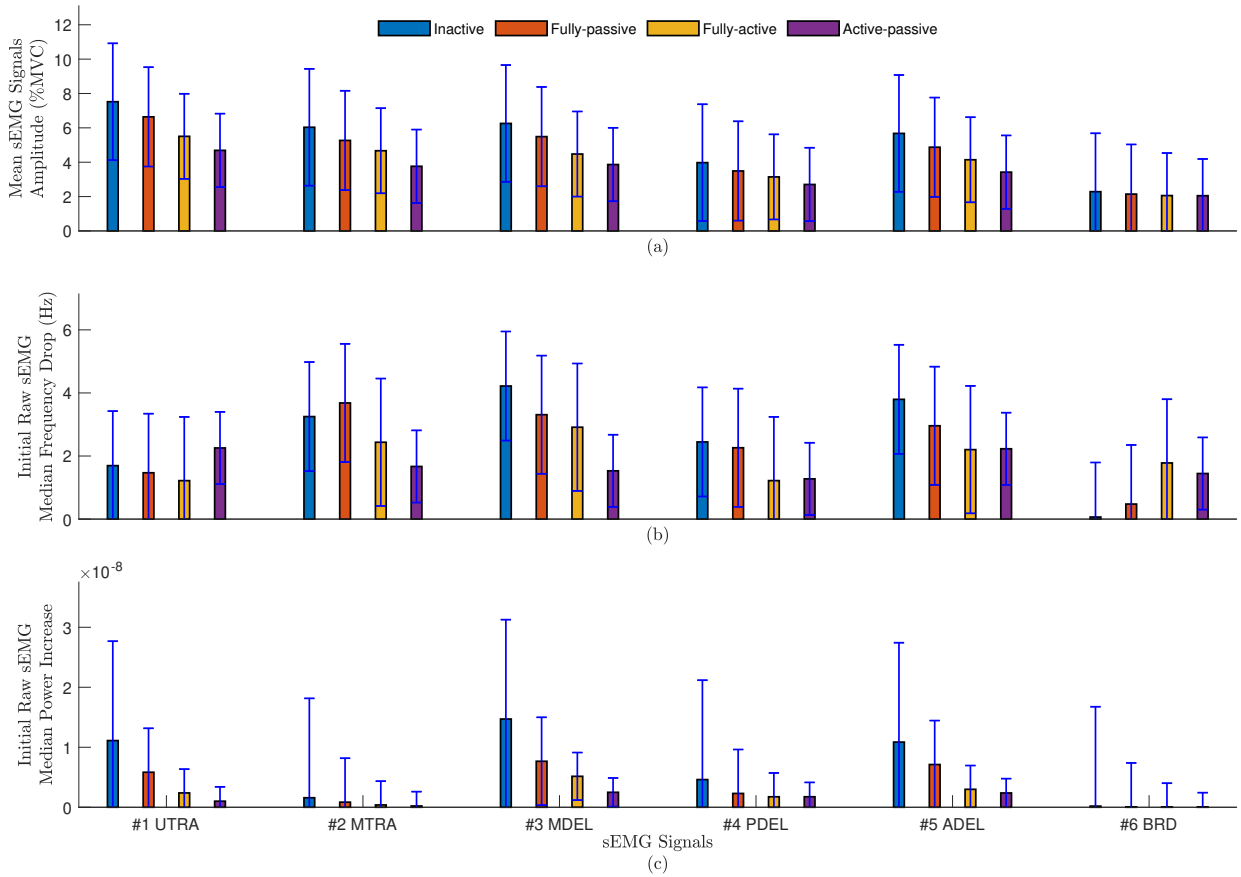


Figure 9.8: The muscle fatigue evaluations for the four exoskeleton actuation types (inactive, fully-passive, fully-active, and active-passive). (a) Mean sEMG signals amplitude percentage; (b & c) Mean frequency drop and power increase of initial raw sEMG signal. The STD (blue line) is also shown on the average of all subjects (bar charts).

9.5.3 Exoskeleton evaluation

- **Quantitative:**
  The results of quantitative evaluation criteria are shown in Figures 9.7 and 9.8. The criteria for the average of all subjects are shown in the solid bar, as well as the maximum and minimum ($\triangle$ and ∇) with the STD (blue line) on the bar charts. According to Figure 9.7, the most efficient exoskeleton types from powerful to weak are: active-passive, fully-active, fully-passive, and inactive. According to Figure 9.8(b), 9.8(c), 9.7(a), and 9.7(d), using the active-passive exoskeleton can decrease the fatigue level compared to using the fully-passive exoskeleton.

  As seen in Figure 9.7(h), participants could lift the object for more time (until they became fatigued) by using the active-passive exoskeleton. The active-passive exoskeleton can be helpful for almost 2 times longer than the fully-passive exoskeleton. In fact, the active-passive exoskeleton is adaptive to the task and can provide variable assistive torque instantly, as compared to the fully-passive torque that can only provide fixed assistive torque at one specific angle, as mentioned in [26]. The participants scored the active-passive exoskeleton to be more effective than other exoskeletons (Figure 9.7(j)).

  For the majority of muscles, according to Figures 9.8(a, b, and d), the muscles were activated, the median frequency was decreased, and the median power was increased from low to high for the following exoskeletons: active-passive, fully-active, fully-passive, and inactive. This means that the muscles became less fatigued with the active-passive exoskeleton. However, a couple of muscles showed not to have similar trends with most of the other muscles. For example, the sEMG signals from the BRD location showed no signs of fatigue. Obviously, BRD is for EFE, and the exoskeleton had no effect on these muscles and the

Table 9.5: Statistical measures for different criteria such as RMSE, p-values, and R-Squared.

Criteria	Figure 9.7	Signal		Model		Statistical Metric		
		Kinematic	sEMG	MuscleNET	MapleSim Biomechanics	Normalized RMSE by Mean value (%)	Subjects Wald P-value	R-Squared
Average fatigue	(a)		✓			30.5	0.0036	0.90
Average active torque	(b)	✓	✓	✓		12.1	0.0280	0.85
Average absolute power	(c)	✓	✓	✓		28.5	0.0260	0.81
Average fatigue	(d)	✓			✓	22.6	0.0041	0.89
Average active torque	(e)	✓			✓	8.1	0.0056	0.93
Average absolute power	(f)	✓			✓	30.0	0.0197	0.82
Average metabolic energy expenditure	(g)	✓			✓	27.8	0.0318	0.76
Maximum loading time	(h)					32.0	0.0286	0.77
Post-test survey	(i)					18.5	0.0394	0.93

joints they cross.

- **Qualitative:**
  All participants ranked the active-passive exoskeleton double that of the fully-passive exoskeleton and 1.5 times that of the fully-active exoskeleton. They stated the following sentences for the active-passive ones in the post-test survey: "I feel my arm was on a table and became less fatigued". "The active-passive was much more helpful than other kinds". "I feel it was holding my arm and had no shoulder pain and fatigue with active-passive and finished the experiment because of my elbow fatigue". "The active-passive phase was significantly effective, and I finished due to wrist fatigue". "I feel I was completely at rest and could hold the object longer". "Felt good and no interference. Overall it was pretty smooth".

- **Statistical analysis:**
  For statistical analyses, we used JMP 16.0 (SAS Inc., NC, USA). To determine normality, the histograms of the data were visually examined for skewness and kurtosis. In Table 9.5 the criteria in Figure 9.7 were evaluated in terms of predicted RMSE, p-value (a statistical metric that calculates the likelihood of getting the outcomes that were observed, supposing that the null hypothesis is correct), and R-Squared (a statistical fit metric that quantifies the proportion of a dependent variable's variance accounted for by the independent variables in a regression model). According to Table 9.5 the general trend and the mentioned discussion in the previous session are valid since the p-value is less than 0.05. The p-value is less than 0.001 considering the sex as a random variable, since obviously each subject tested exoskeletons with one sex type and has not repeated it with another sex type.

  To evaluate the effects of the fixed factors exoskeleton and exoskeleton condition times sex, we conducted Least-Squares Mean Differences and computed Tukey HSD test (using $\alpha = 0.050$ and $Q = 3.14619$). For example, for metabolic energy expenditure in Figure 9.9(g) the Tukey HSD is provided in Table 9.6. As can be seen, levels in Table 9.6 that are not connected by the same letter differ greatly, which means FP-F, FA-M, and AP-M are filled in the same category and the differences are minor but definitely, they are better than IE and worse than AP-F.

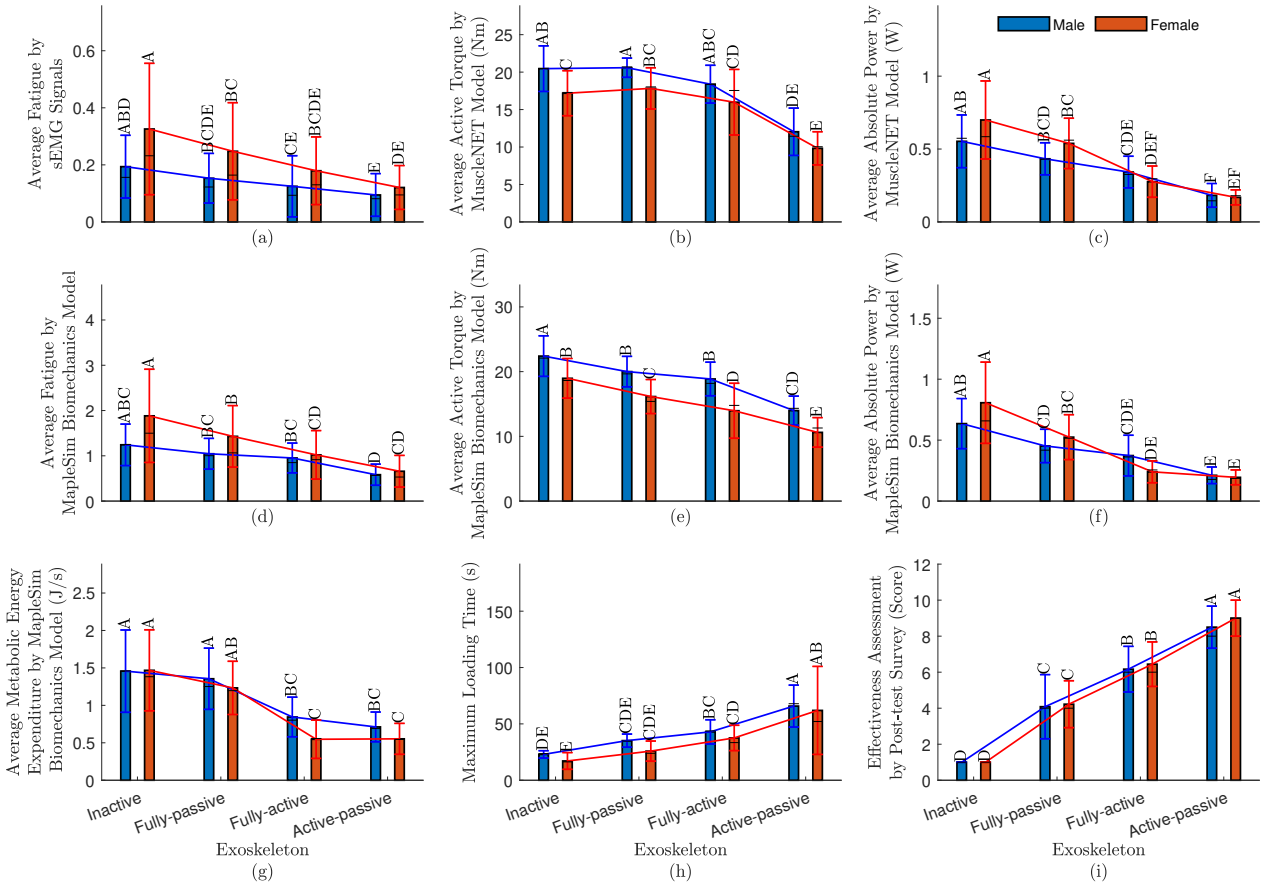


Figure 9.9: The sex impact and quantitative evaluation values for the four exoskeleton actuation types (inactive, fully-passive, fully-active, and active-passive). Note that the median (−), the STD, and the Tukey HSD letters are shown on the average of all subjects for same-sex type (bar charts). (a-c) are calculated from sEMG signals; (d-g) are calculated using the inverse dynamic MapleSim Biomechanics model and measured kinematic data. (a & d) shows average fatigue; (b & e) are showing average human flexion active torque; (c & f) shows average absolute power; (g) displays the average muscle energy expenditure; (h) shows the maximum duration time of load tolerance; and (j) displays the result of the post-test personal survey on the effectiveness of the exoskeletons.

9.5.4 sEMG-based control

A sample of the active-passive exoskeleton test is shown in Figure 9.10. The raw measured and filtered sEMG signals are shown in Figure 9.10(a-f). The predicted joint angle, angular velocity, and the human joint torque contribution to the motion by MuscleNet, as well as the measured and post-calculated values, are shown in Figure 9.10(m-o). The exoskeleton's passive torque, the commanded torque, and total torque are shown in Figure 9.10(p-r). The mid-level AAN-CTM controller defined the active torque according to the human contribution.

As observed in Figure 9.10, the volitional control of the active component was successful. However, one minor challenge was the time delay when the participants wanted to lower their arm at the end of each phase, which is discussed in section 9.7. In addition, there was a minor error in the calculation of the joint angle, angular velocity, and human joint torque. Nonetheless, the minor human joint torque error was deemed feasible and was considered in designing the mid-level controller with the positive dead zone torque (δ). The hyperbolic Equation (9.2) successfully ignored minor differences from the estimated human torque and expected human upper-limb dynamic without the external wrench. Basically, Equation (9.1) strengthens the estimated human joint torque that is higher than the positive threshold torque (γ), and tries to ignore the lower human joint torque than the positive dead zone torque (δ).

Table 9.6: A sample of Tukey HSD letters of metabolic energy expenditure for exoskeleton condition times sex factor. Levels not connected by the same letter are significantly different.

Category Exoskeleton setup	Sex	Connections			Least Sq Mean
IE	M	A			1.506
FP	M	A			1.389
IE	F	A			1.388
FP	F	A	B		1.163
FA	M		B	C	0.852
AP	M		B	C	0.735
FA	F			C	0.498
AP	F			C	0.496

9.6 Practical challenges and future directions

We found the following challenges in the experimental test. Researchers in wearable robotics should address these challenges in the future.

- **Measurement and command delay:** The delay is the main challenge in the HITL experiment. The delay for fetching sEMG and IMU data was 190 ms, since the Delsys Trigno wireless is not real-time equipment and is primarily used for recording and offline analysis. The second primary source of delay was BLDC motor communication with the amount of 185 ms. The reason is because there are 3 connection protocols between MATLAB and BLDC motor, as shown in Figure 9.2. Actually, serial connection, which is responsible for delivering a string or character, is not efficient and fast. This is because once the data is sent from MATLAB to Arduino, it should be converted to a number and then be commanded to the motor. The last delay source is the computation delay with the amount of 90 ms. Filtering sEMG signal, using the machine-learning model, using the CTM, calculating the motor communication instruction, and displaying data in MATLAB, are altogether time-consuming tasks and should be optimized or downgraded to a low-level or embedded controller.
- **Sensor fusion:** Although using these different sensors (sEMG, IMU, and joint angle) is helpful for the estimation of human motion and muscle contribution, this process requires synchronization. Any delay in one sensor should be considered in the model, and the sensor fusion scheme should compensate for synchronization problems.
- **Kinematic sensor inaccuracy:** Currently, an incremental-type sensor measures the motor joint angle. The absolute joint angle measurement requires knowing a global value. We asked the participant to hold the arm downward to calibrate the joint angle sensor. As a result, using an accurate absolute joint angle sensor helped cancel the initial home address setting.
- **sEMG probe and motor built-in sensor disconnection:** Sometimes, the sEMG sensors become disconnected due to detachment or power loss due to a weak battery. The onboard motor sensors sometimes disconnected due to differences in the connection band rate. The HITL controller should be robust and safe regarding any sensor disconnections.

9.7 Concluding remarks

Exoskeleton robots have become a valuable tool for supporting industrial workers and stroke rehabilitation therapy. The movements of these robots can range from being entirely passive to being fully active-assisted. Due to the complex HRIs, realistic sensor and actuator selection, and connection challenges, evaluating exoskeleton actuation designs and building an efficient controller are different and time-consuming. To execute the wearer's movement goals, this research presented a sEMG-based human-robot cooperative control method. It also

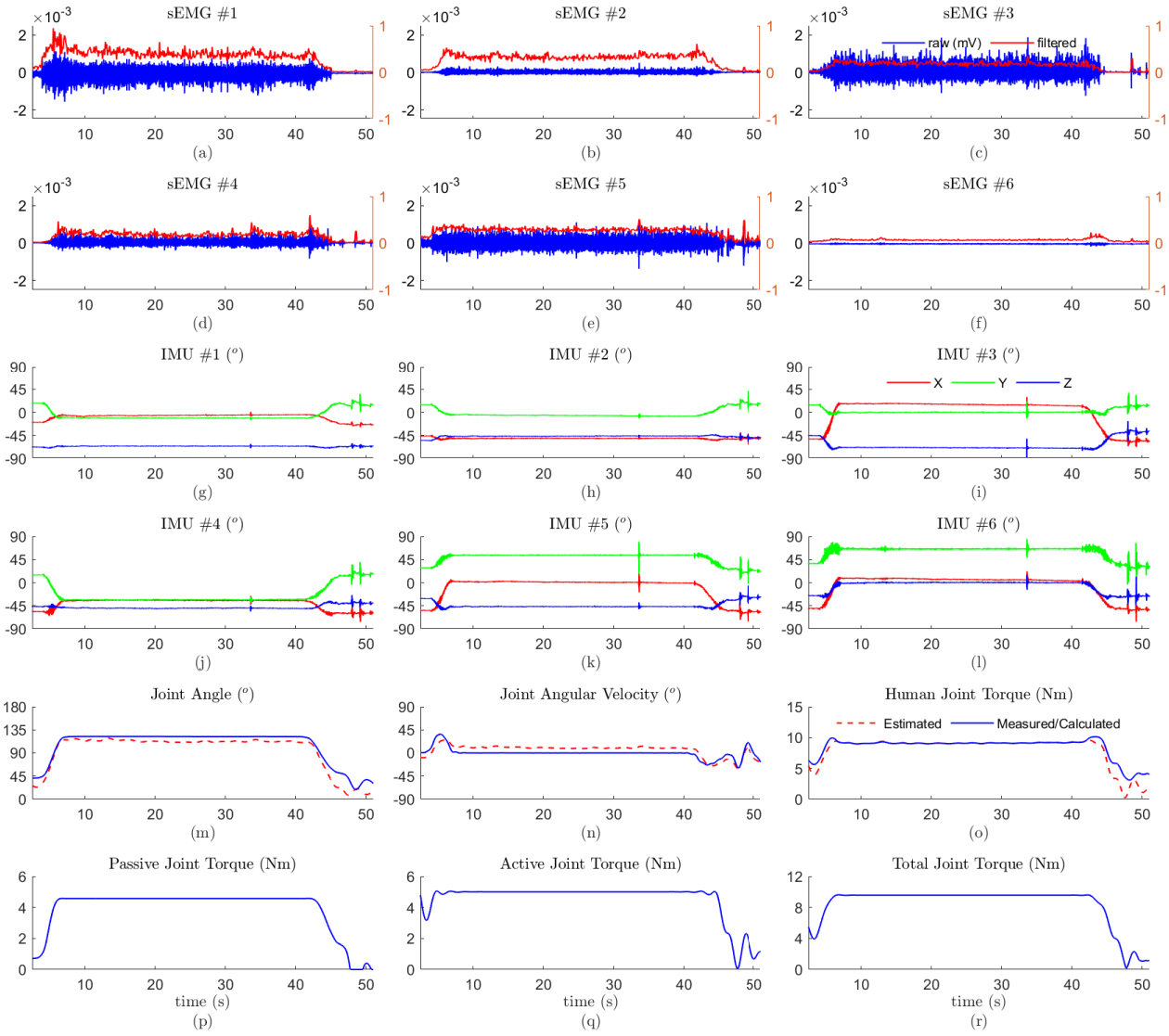


Figure 9.10: A sample of raw, filtered, estimated, and computed data for active-passive exoskeleton control test. (a-f) the raw and filtered sEMG signals; (g-l) the raw Euler angles measured by IMUs; (m) predicted and measured exoskeleton elevation angle; (n) predicted and measured exoskeleton elevation angular speed; (o) the estimated and calculated human contribution torque for the elevation motion, (p) the passive robot torque, (q) the active robot torque, and (r) the total assisted torque.

established a test framework for evaluating a new active-passive shoulder exoskeleton. A mid-level strength regulation, low-level actuator control, and control-oriented sEMG-based intention estimate, were all utilized in the hierarchical control. It was then used to test the controller's efficiency during shoulder joint elevation movement studies with the exoskeleton.

According to the results, the active-passive exoskeleton was more efficient in comparison to the fully-passive and fully-active exoskeleton with the following evaluation criteria: 1) post-test personal survey, 2) load tolerance duration, and 3) computed human active torque, power, computed metabolic energy expenditure using both sEMG signals and inverse dynamic simulation. The results showed that the active-passive exoskeleton was better than the fully-passive and fully-active exoskeleton, since the participants could hold the weighted object for about 2 times longer in duration until they became fatigued.

Future improvements should focus on decreasing the communication delay, and proposing more robust and safe controllers.

Chapter 10

Conclusions

10.1 Summary of contributions

The development and evaluation of an active-passive upper-limb exoskeleton robot using mathematical, computational, machine learning, and experimental methods to support the actuation and automated control of next-generation wearable robots for real-world power assistance was the focus of this thesis research. The following contributions were made:

- **Chapter 3: MapleSim Biomechanics Model**
  i. Development of an adjustable, open-source, user-friendly, and easily augmentable (as an external tool) biomechanical MSK (MapleSim Biomechanics) model capable of extracting the model equations symbolically;
  ii. Modeling of a 20-DoF upper body skeletal dynamic model that is adjustable with anthropometric values: (I) sex, (II) body mass, and (III) body height;
  iii. Modeling of biomechanical joint torques with adjusted MTGs according to anthropometric values and subject body characteristics: (I) sex, (II) age, (III) body mass, (IV) body height, (V) dominant side, and (VI) physical activity;
  iv. Multi-DoF joint implementation of biomechanical joint torques using an MTG model;
  v. Providing a general model useful in various applications broadly categorized as motion analysis, sports engineering, design and control of biomechatronic devices.

- **Chapter 4: Inverse Muscle Modelling (Inverse MuscleNET)** [347]
  i. Development of an optimized recurrent neural network configuration to estimate EMG signals from kinematic and dynamic data;
  ii. Evaluation and acquisition of necessary biomechanical data for the estimation, including joint angle, velocity, acceleration, torque, and activation signal;
  iii. Assessment of the estimation accuracy of the EMG channels for a given joint;
  iv. Evaluation of the general and subject-based estimation model;
  v. Explanation of how InverseMuscleNET can be used in applications for biomechanical analysis (inverse dynamic musculoskeletal simulation), post-rehabilitation analysis (comparison of subject EMG signals against those of a healthy individual via estimated activation signals), sports engineering/optimization, functional electrical stimulation control (application of the estimated healthy limb muscle activation to the functional electrical stimulation probe), and assessment and biofeedback [120].

- **Chapter 5 and 6: Human-exoskeleton System Model and Active-passive Exoskeleton Design** [27, 170, 387]

  i. Modeled the human-exoskeleton multibody system (using six components: the MapleSim Biomechanics upper-body dynamic model (consisting of 20 DoF and 40 MTGs), an NMPC, the exoskeleton's rigid body mechanism, a passive mechanism, a powered actuator (BLDC motor), and an AAN model) and used DAEs simulation to study the interaction force/torque;

  ii. Proposed a guideline to design fully active, fully passive, and active-passive transmission system components;

  iii. Optimized the actuation properties using multiple criteria such as human joint variables and signals as well as practical manufacturing criteria, including the power, inertia, and dimension of the actuator;

  iv. Proposed a model of integrating passive mechanisms with motors that leads to smaller and more lightweight motor selection (Figure 2.2).

- **Chapter 7: Machine Learning Interpretation of Myoelectric Signals (MuscleNET)** [19, 38, 170, 356, 382]

  i. Regression-based mapping for human intent recognition and robot motion prediction, which is useful for muscle modeling and volitional control of biomechatronic devices such as exoskeletons, prostheses, and assistive/resistive robots;

  ii. Real-time mapping of sEMG signals by using an optimum number of layers in the machine learning model (ANN, RNN, CNN, and RCNN);

  iii. Proposing a novel structure of CNN and RCNN for filtering raw sEMG signals and feature learning muscle dynamics;

  iv. Using delayed joint kinematics as a second input type, inspired by biomechanical muscle models;

  v. Estimating the control-oriented signal (the robot's joint angle and joint assistive torque) without additional time-consuming iterative conversion in the control loop;

  vi. Using a data preparation algorithm for the robust mapping of interrupted sEMG signals in order to anticipate signal loss, which is a common challenge in sEMG signal recording;

  vii. Proposing a volitional myoelectric-based control of biomechatronic devices such as exoskeletons, prostheses, and assistive/resistive robots.

- **Chapter 8: Assist-as-needed Mid-level Control of Active Exoskeleton** [28, 52]

  i. Consideration of the distinctive effect of the human's, robot's dynamic models and their interactions, as well as the wrench of the environment;

  ii. Optimization, evaluation, and comparison of an AAN-proportional controller, a novel AAN-CTM controller, a novel AAN-fuzzy logic, and an AAN-MPC mid-level control of a wearable robot during two free motion and lifting tasks;

  iii. Assessment of the three mid-level controllers for the three phases of (A) initial, (B) short-term, and (C) long-term experiences, when wearing a powered robot;

  iv. Comparing performance of an AAN-MPC and AAN-CTM scheme in providing mid-level exoskeleton control. Performance was evaluated in terms of computation cost, error tracking, actuator energy consumption, and metabolic muscle energy consumption;

  v. Experimental evaluation of both proposed mid-level exoskeleton controllers with a HITL setup.

- **Chapter 9: Experimental Evaluation of Active-passive Exoskeleton**

  i. Implementing a hierarchical scheme for real-time sEMG-based control of a shoulder exoskeleton;

  ii. Evaluating active-passive in HITL experiments assistance and comparing it with fully-passive and fully-active exoskeletons using the following criteria: 1) post-test survey, 2) load tolerance duration, 3) mean power frequency, 4) motion fatigue, and 5-8) computed human active torque, joint torque, power, and metabolic energy expenditure using an experimentally-validated and scalable NMSK model;

  iii. Suggesting current real-time implementation challenges to address in future studies and analysis.

10.2 Recommendations for future research on control of active-passive exoskeletons

- **Band or sleeve electromyography:**

  Sometimes, the sEMG sensors become disconnected due to a poor attachment or lose power due to a weak battery. In addition, the onboard motor sensors sometimes disconnect due to differences in the connection band rate. The HITL controller should be robust and safe in response to any sensor disconnections. Compared to traditional electrodes that require skin preparation in sEMG applications, electronic textile (e-textile) electrodes offer numerous advantages, including the ability to be integrated into clothing for continuous muscle activity tracking. Softness, flexibility, air permeability, multi-use, and self-administration without medical assistance are also advantages of e-textile electrodes [412]. By augmenting e-textile EMG on a wearable sleeve, the time-consuming attachment of sEMG, probe shifting, and probe disconnection may be resolved. A second solution may be using sEMG on a band or the armrest of the exoskeleton. This way, the sEMG would be integrated to the device.

- **Sensor fusion:**

  Data from sEMG should supplement rather than replace user-specific measurements, such as mechanical, inertial, and/or environment recognition via vision processing methods for automated control. Images from a wearable smartphone camera system could be combined with measurements from the onboard sEMG and IMU to improve high-level control performance. For instance, when an arm exoskeleton user wants to move an object, the vision processing system indicates the movement and estimates the object's weight. Although using these different sensors is helpful for the estimation of human motion and muscle contribution, this process requires synchronizing. Any delay in one sensor should be considered in the model, and the sensor fusion scheme should compensate for synchronizing problems. This idea in system integration presents exciting challenges and opportunities for future research, such as determining the optimal multi-sensor data fusion for environment-adaptive control of robotic exoskeletons.

- **Real-time implementation:**

  Delays are the main challenge in the HITL setup. The delay for fetching sensor data, systems communication, and different connection protocols between systems should be addressed and optimized, or some control calculations should be downgraded to a low-level or embedded controller.

- **Energy regeneration:**

  An exoskeleton with a heavy battery is not very portable; it may even be tethered with a limited workspace. Energy regeneration from the BLDC motor to the battery makes it possible to lighten the bulky exoskeleton. This presents an opportunity to study both the negative and positive work of this motion, such that a storing energy source can be developed.

- **Actuator design and prototype:**

  The current active-passive exoskeleton is developed by augmenting a BLDC motor to a manufactured passive exoskeleton. The current motor connection to the exoskeleton can be replaced with a new design to save the extra weight of the fixture and housing. The active-passive power source system can also be efficiently re-designed to be lightweight and compact.

- **Long-term clinical evaluations:**

  Future research could study the potential active-passive exoskeleton performance. So far, the calculations in chapter 6 were based on data from a couple of tasks from which frequency can be increased. In addition, the experimental evaluation in chapter 9 was only for short-term usage of the exoskeleton. It would be helpful to test, evaluate, and study the performance in an occupational working environment with actual industrial workers using the exoskeleton. In conclusion, studying HITL in the long term may reveal issues that may not have been revealed in the short term. In addition, further study of gender impact may be considered in the long-term evaluation [413].

10.3 Exoskeleton requirements and challenges

Exoskeleton technology has established its value in a variety of application sectors. Still, there are many difficulties and unanswered scientific questions that need to be addressed. The following list includes some general design criteria directly related to exoskeleton development.

- **Safety and reliability:**

  A flaw in the device's design or functionality could seriously hurt the user. This is why it is crucial to include security and control components that disable or fix the device in the event of unforeseen circumstances. Continuous device parameter monitoring is thus required. The actuators should stop or issue a warning if any reading falls outside of the acceptable range. Fault-tolerant control techniques must be used to reduce human error and unanticipated structural mistakes. To keep the device from exceeding the intended ROM, mechanical stops should be added at the various joints. Additional risks and/or safety issues with exoskeleton use include the potential for skin irritation, allergic responses, hygienic problems, or even chemical burns from the batteries of an active exoskeleton leaking corrosive compounds [414]. Any ADLs that require mobility must be carried out effectively to avoid stumbling or making a mistake.

- **Kinematic compatibility and misalignment:**

  Getting the exoskeleton and the user's anatomical joints to line up correctly is still quite tricky [415]. The physiological joints exhibit different behaviors than typical mechanical joints because of their intricate design. As a result, the rotational joint axes frequently shift positions during movement. A wearable device's kinematic compatibility with the anthropomorphic parameters of the user is essential for a comfortable fit, efficient torque transfer, preservation of maneuverability within the full physiological ROM, and the reduction of unwanted force on the human skeletal system. Minor aspects of human-robot compatibilities, such as discomfort and exercise conditions, will inevitably be accentuated. Therefore, a thorough examination of human upper limb anatomy helps evaluate the design's logic from the user's standpoint. As a reminder, the location of the glenohumeral joint changes due to the very complicated multi-joint cooperative movement of the shoulder, which has the most complex joint structure in the human body. Misalignment is expected as a result of significant skin-to-bone migration or unanticipated forces. As a result, it is crucial to ensure that the device is aligned by using the proper fixation or by including a control mechanism that corrects misalignment.

- **Comfort and fitting:**

  The exoskeleton must be designed with several qualities such as flexibility, breathability, adaptability, softness, lightweight, and ease of use, when taking on and taking off the equipment. Users can utilize the device for extended periods of time if they are more comfortable using it. Robotic components may also be incorporated into soft-material clothing that is simple to put on and take off, in an effort to improve user comfort. However, the durability of these devices should also not be sacrificed.

- **Workspace limitation:**

  Comparatively speaking, wearable exoskeletons cannot offer the same ROM as a human arm. Although the impact is acceptable in some situations, current exoskeleton designs tend to decrease the actual upper limb workspace. To date, a number of studies have attempted to solve this problem of the limited workspace in their design by taking into account upper limb movements at the expense of complicated and heavier mechanisms [416]. To accomplish the desired workspace, studying the mutual ROM between the human and exoskeleton is necessary, but also achievable by examining both systems' kinematic and dynamic models. However, a complicated and heavier mechanism is often required to obtain the large ROM.

- **Joint singularity problem:**

  When two joints of an exoskeleton are lined up with one another, singularity occurs, which is a significant problem. As a result, the exoskeleton loses one DoF, becomes unstable, may vibrate, and may even crash into adjacent objects. Adjusting the exoskeleton out of this configuration requires infinite torque [417]. There are two ways to handle this problem: using a control method to inhibit reaching the singular configuration, or by adding mechanical constraints.

- **Portability and weight:**

  The exoskeleton's portability increases with decreasing exoskeleton weight. The device must be small, simple to use, and maneuverable, in order for the user to move around without restriction and excessive fatigue. Consequently, only the necessary DoF for the required mobility should be included in the system, which would then lessen its complexity. Some actuators and the power source must be placed far away from the injured joint. Rods and cables are reasonable solutions, and cable drives have additional advantages. Combining the device components in easy-to-wear clothing is another possible step to achieving portability. A lightweight power source with more battery life is essential for extended usage periods. Researchers are trying to implement regeneration technology into the exoskeleton system to restore energy from negative power motion.

- **Accessibility and affordability:**

  Due to cost, weight, and safety issues, many of today's exoskeletons are unavailable outside of a clinical or research environment. Additionally, the handling and installation of the device are frequently complicated and need ongoing supervision from certified technicians. To improve accessibility, the exoskeleton must should be portable, easy to use, and affordable. By ensuring components are as simple, light, and inexpensive as possible by lowering prices using inexpensive materials, exoskeletons become more accessible. Making a modular device also allows users to add and remove components based on their needs.

- **Actuation selection and low power consumption:**

  Any joint in the body requires a significant amount of energy to function. To build lightweight exoskeletons with sufficient force/torque to move the joints, it is crucial to use actuators with a high PWR. Actuators typically need to be optimized in order to increase the PWR due to the constrained space in an exoskeleton and the high torques needed for its functioning. If possible, the inertia in operation and the weight of the actuator and power supply should be decreased to prevent uncomfortable loads for the user. Thus, it is important to incorporate passive components into the design to reduce the number of actuators.

- **Human–robot interaction:**

  Humans and robots have a symbiotic relationship that goes beyond primary physical contact. By implementing new mechanisms, modalities of actuation, and control strategies, several solutions have been implemented to enhance the connection between the human and exoskeletons [49]. In contrast to autonomous robots, the HITL control technique is the main focus of exoskeleton systems. However, it is difficult to fully or partially implement such a perfect control loop. The challenges in developing such a control approach include cooperation and function allocation, human-exoskeleton information interchange, real-time motion planning, safety control, and long-term adaptation to robot.

- **Sensing, sensor fusion, and interpretation:**

  Integratable sensor modules are now readily available on the market for practically all types of electrical actuators, and can accurately predict the characteristics of the robot. However, identifying the exact human intent is crucial for controlling active exoskeletons [40]. Control failure is heavily influenced by the human operator's inability to produce real-time input to the machine while it is operating. Establishing a relationship between physiological and physical parameters and the bioelectrical signal is vital. It is thus necessary to use sophisticated algorithms to extract helpful brain information [41]. So far, a biomimetic bioelectrical-based controller shows smoother and natural movements [42–44].

- **Performance assessment using models:**

  So far, it is unclear how a robotic exoskeleton would affect the human body, especially how it will affect safety, ergonomics, and health care. The robotic exoskeleton system cannot only be assessed using common performance indicators or testing procedures. It is necessary to establish special testing processes that consider safety and HRI.

- **Design and optimization using models:**

  Designing exoskeletons using human biomechanics could result in safe and obedient HRIs. To analyze the impact of the upper limb exoskeleton, and evaluate and optimize the robot design, a thorough MSK model is needed. The MSK model enables researchers to investigate, assess, and optimize the mechanical design

by incorporating HRI. The human MSK system has been studied and modeled in an effort to comprehend it better. The human MSK system can be analyzed kinematically and dynamically using a variety of techniques, and facilities can provide access to pertinent biological information such as the response forces of the MSK system [46].

10.4 Future research directions

Undoubtedly, the use of exoskeletons in rehabilitation, occupation, and haptic fields has boosted their development in recent years. Exoskeletons are a viable and practical solution; however, most existing technologies have some drawbacks. For instance, the majority still place a significant load on users and are inefficient for daily use. They are challenging to put on and take off, and the majority are still not very portable. The rotating axis can also be easily misaligned, and the equipment cost is high. Currently, many research teams are developing tools to aid in ADLs and rehabilitation procedures.

Rehabilitation does not always result in a full return of sensory-motor abilities. As a result, some patients are forced to live with disability forever, despite receiving therapy. This has resulted in patients who frequently require assistance with ADLs, having a lower quality of life. Therefore, it is crucial to provide tools that lessen the disability of the user. Exoskeletons are a potential alternative for consistent help. The creation of the ideal exoskeleton, however, is still tricky.

The lack of mobility of exoskeletons is rarely seen as a significant issue in rehabilitation because therapies are frequently administered in regulated settings with set boundaries. However, the portability limitation for assistive equipment prevents users from moving freely for their ADLs. Exoskeleton development that uses active-passive actuators is beneficial in the field of assistive technology. Compared to a pneumatic or hydraulic system with the same characteristics, an electric battery and motor system is significantly lighter, more accurate, smaller, and better suited for portable assistance.

Exoskeletons are undoubtedly necessary for long-term or permanent daily care. The features needed for continuous use are not yet present in all current devices. To enable prolonged usage of these devices, more study is therefore required. The system can be improved by taking proactive measures when the control is based on the movement's intention. However, more research must be done to pinpoint the human movement's intended purpose.

A primary area of study for exoskeleton design methodology is biomechanical design, which is based on the anatomy of human limb movement. Even though a variety of exoskeleton constructions have been developed, mechanically simulating a human's natural motion has proven to be challenging. Therefore, exploring the biomechanics of exoskeletons in terms of their shape, function, and distribution of DoF is essential.

In jobs requiring persistent and excessive effort, an augmented exoskeleton helps reduce numerous injuries. Its development is complex, however, since it needs to boost the user's inherent potential, while also requiring less physical effort. The energy lost due to the retention stresses placed on the joints to keep them in the appropriate position is a significant additional factor to consider. To avoid retention pressures and conserve energy, it is advised that the robotic equipment have restraints that fix the desired position.

In conclusion, considerable advancements are still needed in exoskeleton development to reduce restrictions and create a highly effective tool that gives the user comfort and portability. Exoskeletons must be customized for each user to reduce the danger of damage and improve mechanical effectiveness. Weight should be decreased, and actuator power should be increased. Control systems that are reliable and efficient must also be studied.

References

[1] J. J. Luime, B. W. Koes, I. J. Hendriksen, A. Burdorf, A. P. Verhagen, H. S. Miedema, and J. A. Verhaar, "Prevalence and incidence of shoulder pain in the general population; A systematic review," *Scandinavian Journal of Rheumatology*, vol. 33, no. 2, pp. 73–81, 2004.

[2] S. L. Fischer, S. Koltun, and J. Lee, "A cross-sectional survey of musculoskeletal disorder hazard exposures and self-reported discomfort among on-shore wind turbine service technicians," *Ergonomics*, vol. 64, no. 3, pp. 383–395, 2021.

[3] I. Khansa, L. Khansa, T. S. Westvik, J. Ahmad, F. Lista, and J. E. Janis, "Reply: Work-related musculoskeletal injuries in plastic surgeons in the United States, Canada, and Norway," *Plastic and Reconstructive Surgery*, vol. 141, no. 1, pp. 165e–175e, 2018.

[4] G. Nazari, J. MacDermid, and H. Cramm, "Prevalence of musculoskeletal disorders among Canadian firefighters: A systematic review and meta-analysis," *Journal of Military, Veteran and Family Health*, vol. 6, no. 1, pp. 83–97, 2020.

[5] R. M. Van Rijn, B. M. Huisstede, B. W. Koes, and A. Burdorf, "Associations between work-related factors and specific disorders of the shoulder - A systematic review of the literature," *Scandinavian Journal of Work, Environment and Health*, vol. 36, no. 3, pp. 189–201, 2010.

[6] D. Schopflocher, P. Taenzer, and R. Jovey, "The prevalence of chronic pain in Canada," *Pain Research and Management*, vol. 16, no. 6, pp. 445–450, 2011.

[7] C. H. Linaker and K. Walker-Bone, "Shoulder disorders and occupation," *Best Practice and Research: Clinical Rheumatology*, vol. 29, no. 3, pp. 405–423, 2015.

[8] X. Zhou and L. Zheng, "Model-based comparison of passive and active assistance designs in an occupational upper limb exoskeleton for overhead lifting," *IISE Transactions on Occupational Ergonomics and Human Factors*, vol. 9, pp. 167–185, 2021.

[9] J. R. Grieve and C. R. Dickerson, "Overhead work: Identification of evidence-based exposure guidelines," *Occupational Ergonomics*, vol. 8, no. 1, pp. 53–66, 2008.

[10] A. Descatha, J. F. Chastang, D. Cyr, A. Leclerc, Y. Roquelaure, and B. Evanoff, "Do workers with self-reported symptoms have an elevated risk of developing upper extremity musculoskeletal disorders three years later?" *Occupational and Environmental Medicine*, vol. 65, no. 3, pp. 205–207, 2008.

[11] J. Beach, A. Senthilselvan, and N. Cherry, "Factors affecting work-related shoulder pain," *Occupational Medicine*, vol. 62, no. 6, pp. 451–454, 2012.

[12] H. Miranda, L. Punnett, E. Viikari-Juntura, M. Heliövaara, and P. Knekt, "Physical work and chronic shoulder disorder. Results of a prospective population-based study," *Annals of the Rheumatic Diseases*, vol. 67, no. 2, pp. 218–223, 2008.

[13] K. K. Dunning, K. G. Davis, C. Cook, S. E. Kotowski, C. Hamrick, G. Jewell, and J. Lockey, "Costs by industry and diagnosis among musculoskeletal claims in a state workers compensation system: 1999-2004," *American Journal of Industrial Medicine*, vol. 53, no. 3, pp. 276–284, 2010.

[14] L. Blanchet, S. Achiche, Q. Docquier, P. Fisette, and M. Raison, "A procedure to optimize the geometric and dynamic designs of assistive upper limb exoskeletons," *Multibody System Dynamics*, vol. 51, no. 2, pp. 221–245, 2020.

[15] S. Sirawattanakul and W. Sanngoen, "Review of upper limb exoskeleton for rehabilitation and assistive application," *International Journal of Mechanical Engineering and Robotics Research*, vol. 9, no. 5, pp. 752–758, 2020.

[16] A. Gupta, A. Singh, V. Verma, A. K. Mondal, and M. K. Gupta, "Developments and clinical evaluations of robotic exoskeleton technology for human upper-limb rehabilitation," *Advanced Robotics*, vol. 34, no. 15, pp. 1023–1040, 2020.

[17] M. R. Islam, B. Brahmi, T. Ahmed, M. Assad-Uz-Zaman, and M. H. Rahman, "Exoskeletons in upper limb rehabilitation: A review to find key challenges to improve functionality," *Control Theory in Biomedical Engineering*, pp. 235–265, 2020.

[18] C. J. Yang, J. F. Zhang, Y. Chen, Y. M. Dong, and Y. Zhang, "A review of exoskeleton-type systems and their key technologies," *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, vol. 222, no. 8, pp. 1599–1612, 2008.
[19] A. Nasr, B. Laschowski, and J. McPhee, "Myoelectric control of robotic leg prostheses and exoskeletons: A review," in *Proceedings of the ASME International Design Engineering Technical Conferences & Computers and Information in Engineering Conference*, vol. 85444. Online, Virtual: ASME, 2021, pp. 2021–69 203.
[20] J. L. Pons, *Wearable robots: Biomechatronic exoskeletons*. John Wiley & Sons, 2008.
[21] S. Bai, G. S. Virk, and T. G. Sugar, *Wearable exoskeleton systems: Design, control and applications*. Institution of Engineering and Technology, 2018.
[22] K. Anam and A. A. Al-Jumaily, "Active exoskeleton control systems: State of the art," *Procedia Engineering*, vol. 41, pp. 988–994, 2012.
[23] G. Bao, L. Pan, H. Fang, X. Wu, H. Yu, S. Cai, B. Yu, and Y. Wan, "Academic review and perspectives on robotic exoskeletons," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 27, no. 11, pp. 2294–2304, 2019.
[24] P. Xu, D. Xia, J. Li, J. Zhou, and L. Xie, "Execution and perception of upper limb exoskeleton for stroke patients: a systematic review," *Intelligent Service Robotics*, pp. 1–22, 2022.
[25] N. Sylla, V. Bonnet, F. Colledani, and P. Fraisse, "Ergonomic contribution of ABLE exoskeleton in automotive industry," *International Journal of Industrial Ergonomics*, vol. 44, no. 4, pp. 475–481, 2014.
[26] A. Nasr, S. Ferguson, and J. McPhee, "Model-based design and optimization of passive shoulder exoskeletons," *Journal of Computational and Nonlinear Dynamics*, vol. 17, no. 5, p. 051004, 2022.
[27] A. Nasr, S. Bell, and J. McPhee, "Optimal design of active-passive shoulder exoskeletons: A computational modeling of human-robot interaction," *Multibody System Dynamics*, p. in press, 2022.
[28] A. Nasr, A. Hashemi, and J. McPhee, "Model-based mid-level regulation for assist-as-needed hierarchical control of wearable robots: A computational study of human-robot adaptation," *Robotics*, vol. 11, no. 1, p. 20, 2022.
[29] I. Galiana, F. L. Hammond, R. D. Howe, and M. B. Popovic, "Wearable soft robotic device for post-stroke shoulder rehabilitation: Identifying misalignments," in *Proceedings of the IEEE International Conference on Intelligent Robots and Systems*. Vilamoura-Algarve, Portugal: IEEE, 2012, pp. 317–322.
[30] Y. Mao and S. K. Agrawal, "Design of a cable-driven arm exoskeleton (CAREX) for neural rehabilitation," *IEEE Transactions on Robotics*, vol. 28, no. 4, pp. 922–931, 2012.
[31] B. Laschowski, R. S. Razavian, and J. McPhee, "Simulation of stand-to-sit biomechanics for robotic exoskeletons and prostheses with energy regeneration," *IEEE Transactions on Medical Robotics and Bionics*, vol. 3, no. 2, pp. 455–462, 2021.
[32] B. Laschowski, J. McPhee, and J. Andrysek, "Lower-Limb Prostheses and Exoskeletons With Energy Regeneration: Mechatronic Design and Optimization Review," *Journal of Mechanisms and Robotics*, vol. 11, no. 4, p. 040801, 2019.
[33] J. Huang, W. Huo, W. Xu, S. Mohammed, and Y. Amirat, "Control of upper-limb power-assist exoskeleton using a human-robot interface based on motion intention recognition," *IEEE Transactions on Automation Science and Engineering*, vol. 12, no. 4, pp. 1257–1270, 2015.
[34] M. P. de Looze, T. Bosch, F. Krause, K. S. Stadler, and L. W. O'Sullivan, "Exoskeletons for industrial application and their potential effects on physical work load," *Ergonomics*, vol. 59, no. 5, pp. 671–681, 2016.
[35] A. W. de Vries and M. P. de Looze, "The effect of arm support exoskeletons in realistic work activities : A review study," *J Ergonomics*, vol. 9, no. 4, pp. 1–9, 2019.
[36] R. Gopura, D. S. V. Bandara, K. Kiguchi, and G. K. I. Mann, "Developments in hardware systems of active upper-limb exoskeleton robots: A review," *Robotics and Autonomous Systems*, vol. 75, pp. 203–220, 2016.
[37] R. Bogue, "Exoskeletons – a review of industrial applications," *Industrial Robot*, vol. 45, no. 5, pp. 585–590, 2018.
[38] A. Nasr, S. Bell, J. He, R. L. Whittaker, N. Jiang, C. R. Dickerson, and J. McPhee, "MuscleNET: Mapping electromyography to kinematic and dynamic biomechanical variables," *Journal of Neural Engineering*, vol. 18, no. 4, p. 0460d3, 2021.
[39] Z. Lu, A. Stampas, G. E. Francisco, and P. Zhou, "Offline and online myoelectric pattern recognition analysis and real-time control of a robotic hand after spinal cord injury," *Journal of Neural Engineering*, vol. 16, no. 3, p. 036018, 2019.
[40] H. D. Lee, W. S. Kim, J. S. Han, and C. S. Han, "The technical trend of the exoskeleton robot system for human power assistance," *International Journal of Precision Engineering and Manufacturing*, vol. 13, no. 8, pp. 1491–1497, 2012.
[41] B. Siciliano, O. Khatib, and F. Groen, "Exoskeletons in rehabilitation robotics," in *Springer Tracts in Advanced Robotics*. Springer, 2011, vol. 69, pp. 1–150.
[42] A. F. Ruiz, E. Rocon, and A. Forner-Cordero, "Exoskeleton-based robotic platform applied in biomechanical modelling of the human upper limb," *Applied Bionics and Biomechanics*, vol. 6, no. 2, pp. 205–216, 2009.
[43] L. L. Ren, Z. Qian, and L. L. Ren, "Biomechanics of musculoskeletal system and its biomimetic implications: A review," *Journal of Bionic Engineering*, vol. 11, no. 2, pp. 159–175, 2014.

[44] Y. Hasegawa, Y. Mikami, K. Watanabe, and Y. Sankai, "Five-fingered assistive hand with mechanical compliance of human finger," in *Proceedings of the IEEE International Conference on Robotics and Automation*. Pasadena, CA, USA: IEEE, 2008, pp. 718–724.
[45] J. C. Perry, J. Rosen, and S. Burns, "Upper-limb powered exoskeleton design," *IEEE/ASME Transactions on Mechatronics*, vol. 12, no. 4, pp. 408–417, 2007.
[46] M. Laitenberger, M. Raison, D. Périé, and M. Begon, "Refinement of the upper limb joint kinematics and dynamics using a subject-specific closed-loop forearm model," *Multibody System Dynamics*, vol. 33, no. 4, pp. 413–438, 2015.
[47] E. P. Luque, D. Högberg, A. Iriondo, and P. Thorvald, "Evaluation of the use of exoskeletons in the range of motion of workers," Ph.D. dissertation, University of Skovde, Skovde, 2019.
[48] T. Lenzi, S. De Rossi, N. Vitiello, A. Chiri, S. Roccella, F. Giovacchini, F. Vecchi, and M. C. Carrozza, "The neuro-robotics paradigm: NEURARM, NEUROExos, HANDEXOS," in *Proceedings of the 31st Annual International Conference of the IEEE Engineering in Medicine and Biology Society: Engineering the Future of Biomedicine*. Minneapolis, MN, USA: IEEE, 2009, pp. 2430–2433.
[49] S. Bai and J. Rasmussen, "Modelling of physical human-robot interaction for exoskeleton designs," in *Multibody Dynamics,CCOMAS Thematic Conference*, Brussels, Belgium, 2011, pp. 4–7.
[50] B. K. Dinh, M. Xiloyannis, C. W. Antuvan, L. Cappello, and L. Masia, "Hierarchical cascade controller for assistance modulation in a soft wearable arm exoskeleton," *IEEE Robotics and Automation Letters*, vol. 2, no. 3, pp. 1786–1793, 2017.
[51] A. M. Willson, A. J. Anderson, C. A. Richburg, B. C. Muir, J. Czerniecki, K. M. Steele, and P. M. Aubin, "Full body musculoskeletal model for simulations of gait in persons with transtibial amputation," *Computer Methods in Biomechanics and Biomedical Engineering*, 2022.
[52] M. Febrer-Nafría, A. Nasr, M. Ezati, P. Brown, J. M. Font-Llagunes, and J. McPhee, "Predictive multibody dynamic simulation of human neuromusculoskeletal systems: A review," *Multibody System Dynamics*, p. in press, 2022.
[53] M. S. Shourijeh, N. Mehrabi, J. J. McPhee, and B. J. Fregly, "Advances in musculoskeletal modeling and their application to neurorehabilitation," *Frontiers in Neurorobotics*, vol. 14, p. 65, 2020.
[54] B. Ghannadi, R. S. Razavian, and J. McPhee, "Upper extremity rehabilitation robots: A survey," in *Handbook of Biomechatronics*. San Diego, CA, USA: Elsevier, 2018, ch. 9, pp. 319–353.
[55] M. Cardona and C. E. García Cena, "Biomechanical analysis of the lower limb: A full-body musculoskeletal model for muscle-driven simulation," *IEEE Access*, vol. 7, pp. 92 709–92 723, 2019.
[56] W. McNally and J. McPhee, "Dynamic optimization of the golf swing using a six degree-of-freedom biomechanical model," in *Proceedings*, vol. 2. Brisbane, Queensland, Australia: MDPI, 2018, p. 243.
[57] M. Ezati, B. Ghannadi, and J. McPhee, "A review of simulation methods for human movement dynamics with emphasis on gait," *Multibody System Dynamics*, vol. 47, no. 3, pp. 265–292, 2019.
[58] R. S. Razavian, B. Ghannadi, and J. McPhee, "On the relationship between muscle synergies and redundant degrees of freedom in musculoskeletal systems," *Frontiers in Computational Neuroscience*, vol. 13, p. 23, 2019.
[59] M. Damsgaard, J. Rasmussen, S. T. Christensen, E. Surma, and M. de Zee, "Analysis of musculoskeletal systems in the AnyBody Modeling System," *Simulation Modelling Practice and Theory*, vol. 14, no. 8, pp. 1100–1111, 2006.
[60] M. Nordin and V. H. Frankel, *Basic biomechanics of the musculoskeletal system*. Philadelphia, PA, USA: Lippincott Williams & Wilkins, 2001.
[61] C. Jansen and J. McPhee, "Predictive dynamic simulation of Olympic track cycling standing start using direct collocation optimal control," *Multibody System Dynamics*, vol. 49, no. 1, pp. 53–70, 2020.
[62] M. A. Destarac, C. E. García Cena, R. J. Saltarén Pazmiño, M. J. Reyes Urbina, J. López López, and R. E. Gómez, "Modeling and simulation of upper brachial plexus injury," *IEEE Systems Journal*, vol. 10, no. 3, pp. 912–921, 2016.
[63] D. Cazzola, T. P. Holsgrove, E. Preatoni, H. S. Gill, and G. Trewartha, "Cervical spine injuries: A whole-body musculoskeletal model for the analysis of spinal loading," *PLoS ONE*, vol. 12, no. 1, p. e0169329, 2017.
[64] A. Seth, M. Dong, R. Matias, and S. L. Delp, "Muscle contributions to upper-extremity movement and work from a musculoskeletal model of the human shoulder," *Frontiers in Neurorobotics*, vol. 13, p. 90, 2019.
[65] C. F. Hayford, E. Montefiori, E. Pratt, and C. Mazzà, "Predicting longitudinal changes in joint contact forces in a juvenile population: scaled generic versus subject-specific musculoskeletal models," *Computer Methods in Biomechanics and Biomedical Engineering*, pp. 1014–1025, 2020.
[66] L. Zhang, G. Liu, Y. Yan, B. Han, H. Li, J. Ma, and X. Wang, "A subject-specific musculoskeletal model to predict the tibiofemoral contact forces during daily living activities," *Computer Methods in Biomechanics and Biomedical Engineering*, 2022.
[67] S. Kumar, "Isolated planar trunk strengths measurement in normals: Part III - Results and database," *International Journal of Industrial Ergonomics*, vol. 17, no. 2, pp. 103–111, 1996.
[68] R. Dumas, L. Chèze, and J. P. Verriest, "Adjustments to McConville et al. and Young et al. body segment inertial parameters," *Journal of Biomechanics*, vol. 40, no. 3, pp. 543–553, 2007.

[69] J. T. McConville, T. Churchill, I. Kaleps, C. E. Clauser, and J. Cuzzi, "Anthropometric relationships of body and body segment moments of inertia," Air Force Aerospace Medical Research Laboratory, Aerospace Medical Division, Dayton, OH, USA, Tech. Rep., 1980.
[70] T. Harbo, J. Brincks, and H. Andersen, "Maximal isokinetic and isometric muscle strength of major muscle groups related to age, body mass, height, and sex in 178 healthy subjects," *European Journal of Applied Physiology*, vol. 112, no. 1, pp. 267–275, 2012.
[71] M. A. Gallagher, F. Cuomo, L. Polonsky, K. Berliner, and J. D. Zuckerman, "Effects of age, testing speed, and arm dominance on isokinetic strength of the elbow," *Journal of Shoulder and Elbow Surgery*, vol. 6, no. 4, pp. 340–346, 1997.
[72] H. W. Staudte and N. Duhr, "Age- and sex-dependent force-related function of the cervical spine," *European Spine Journal*, vol. 3, no. 3, pp. 155–161, 1994.
[73] Yassierli, M. A. Nussbaum, H. Iridiastadi, and L. A. Wojcik, "The influence of age on isometric endurance and fatigue is muscle dependent: A study of shoulder abduction and torso extension," *Ergonomics*, vol. 50, no. 1, pp. 26–45, 2007.
[74] A. Jordan, J. Mehlsen, P. M. Bülow, K. Ostergaard, and B. Danneskiold-Samsee, "Maximal isometric strength of the cervical musculature in 100 healthy volunteers," *Spine*, vol. 24, no. 13, pp. 1343–1348, 1999.
[75] T. Okada, T. Hakkaku, K. Iwai, and K. Nakazato, "Weight category-dependent trunk muscle strength and its relation with LBP in elite judokas," *Sports Medicine International Open*, vol. 05, no. 01, pp. E14–E21, 2021.
[76] J. M. Roberson, P. Witt, and M. T. Gross, "A comparison of trunk extensor strength and squat lifting ability," *Journal of Orthopaedic and Sports Physical Therapy*, vol. 25, no. 2, pp. 137–144, 1997.
[77] J. C. Otis, R. F. Warren, S. I. Backus, T. J. Santner, and J. D. Mabrey, "Torque production in the shoulder of the normal young adult male," *The American Journal of Sports Medicine*, vol. 18, no. 2, pp. 119–123, 1990.
[78] J. M. Vanswearingen, "Measuring wrist muscle strength," *Journal of Orthopaedic and Sports Physical Therapy*, vol. 4, no. 4, pp. 217–228, 1983.
[79] O. Zuzgina and M. M. Wdowski, "Asymmetry of dominant and non-dominant shoulders in university level men and women volleyball players," *Human Movement*, vol. 20, no. 4, pp. 19–27, 2019.
[80] K. Söderman, E. Bergström, R. Lorentzon, and H. Alfredson, "Bone mass and muscle strength in young female soccer players," *Calcified Tissue International*, vol. 67, no. 4, pp. 297–303, 2000.
[81] Z. Shao, Q. Wu, B. Chen, and H. Wu, "Force and deformation transmission characteristics of a compliant tendon–sheath actuation system based on Hill-type muscle model," *Proceedings of the Institution of Mechanical Engineers, Part H: Journal of Engineering in Medicine*, vol. 233, no. 7, pp. 695–705, 2019.
[82] J. M. Winters, "Hill-based muscle models: A systems engineering perspective," in *Multiple Muscle Systems*. New York, NY, USA: Springer, 1990, ch. 5, pp. 69–93.
[83] Q. Zhang, X. Wang, M. Tian, X. Shen, and Q. Wu, "Modeling of novel compound tendon-sheath artificial muscle inspired by hill muscle model," *IEEE Transactions on Industrial Electronics*, vol. 65, no. 8, pp. 6372–6381, 2018.
[84] K. A. Inkol, C. Brown, W. McNally, C. Jansen, and J. McPhee, "Muscle torque generators in multibody dynamic simulations of optimal sports performance," *Multibody System Dynamics*, vol. 50, no. 4, pp. 435–452, 2020.
[85] S. E. Forrester, M. R. Yeadon, M. A. King, and M. T. Pain, "Comparing different approaches for determining joint torque parameters from isovelocity dynamometer measurements," *Journal of Biomechanics*, vol. 44, no. 5, pp. 955–961, 2011.
[86] M. R. Yeadon, M. A. King, and C. Wilson, "Modelling the maximum voluntary joint torque/angular velocity relationship in human movement," *Journal of Biomechanics*, vol. 39, no. 3, pp. 476–482, 2006.
[87] C. Brown, W. McNally, and J. McPhee, "Optimal control of joint torques using direct collocation to maximize ball carry distance in a golf swing," *Multibody System Dynamics*, vol. 50, no. 3, pp. 323–333, 2020.
[88] S. Bell and J. McPhee, "Muscle torque generator model for a two degree-of-freedom shoulder joint," in *Proceedings of the North American Congress on Biomechanics*, Ottawa, Canada, 2022, p. under press.
[89] R. D. Crowninshield and R. A. Brand, "A physiologically based criterion of muscle force prediction in locomotion," *Journal of Biomechanics*, vol. 14, no. 11, pp. 793–801, 1981.
[90] B. Laschowski, N. Mehrabi, and J. McPhee, "Optimization-based motor control of a paralympic wheelchair athlete," *Sports Engineering*, vol. 21, no. 3, pp. 207–215, 2018.
[91] T. Desplenter and A. L. Trejos, "Evaluating muscle activation models for elbow motion estimation," *Sensors*, vol. 18, no. 4, p. 1004, 2018.
[92] K. Manal and T. S. Buchanan, "A one-parameter neural activation to muscle activation model: Estimating isometric joint moments from electromyograms," *Journal of Biomechanics*, vol. 36, no. 8, pp. 1197–1202, 2003.
[93] T. Heidlauf and O. Rohrle, "A multiscale chemo-electro-mechanical skeletal muscle model to analyze muscle contraction and force generation for different muscle fiber arrangements," *Frontiers in Physiology*, vol. 5, p. 498, 2014.
[94] S. Heitmann, N. Ferns, and M. Breakspear, "Muscle co-contraction modulates damping and joint stability in a three-link biomechanical limb," *Frontiers in Neurorobotics*, vol. 5, p. 5, 2012.

[95] M. Hammer, M. Günther, D. F. Haeufle, and S. Schmitt, "Tailoring anatomical muscle paths: A sheath-like solution for muscle routing in musculoskeletal computer models," *Mathematical Biosciences*, vol. 311, pp. 68–81, 2019.
[96] N. Mehrabi and J. McPhee, "Model-based control of biomechatronic systems," in *Handbook of Biomechatronics*. San Diego, CA, USA: Elsevier, 2018, pp. 95–126.
[97] F. C. Anderson and M. G. Pandy, "Static and dynamic optimization solutions for gait are practically equivalent," *Journal of Biomechanics*, vol. 34, no. 2, pp. 153–161, 2001.
[98] F. Bailly, A. Ceglia, B. Michaud, D. M. Rouleau, and M. Begon, "Real-time and dynamically consistent estimation of muscle forces using a moving horizon EMG-marker tracking algorithm—Application to upper limb biomechanics," *Frontiers in Bioengineering and Biotechnology*, vol. 9, p. 642742, 2021.
[99] R. H. Miller, S. C. Brandon, and K. J. Deluzio, "Predicting sagittal plane biomechanics that minimize the axial knee joint contact force during walking," *Journal of Biomechanical Engineering*, vol. 135, no. 1, p. 011007, 2013.
[100] M. S. Shourijeh, N. Mehrabi, and J. McPhee, "Forward static optimization in dynamic simulation of human musculoskeletal systems: A proof-of-concept study," *Journal of Computational and Nonlinear Dynamics*, vol. 12, no. 5, p. 051005, 2017.
[101] M. S. Shourijeh and J. McPhee, "Forward dynamic optimization of human gait simulations: A global parameterization approach," *Journal of Computational and Nonlinear Dynamics*, vol. 9, no. 3, p. 031018, 2014.
[102] M. Hirashima and T. Oya, "How does the brain solve muscle redundancy? Filling the gap between optimization and muscle synergy hypotheses," *Neuroscience Research*, vol. 104, pp. 80–87, 2016.
[103] A. Erdemir, S. McLean, W. Herzog, and A. J. van den Bogert, "Model-based estimation of muscle forces exerted during movements," *Clinical Biomechanics*, vol. 22, no. 2, pp. 131–154, 2007.
[104] S. Heintz and E. M. Gutierrez-Farewik, "Static optimization of muscle forces during gait in comparison to EMG-to-force processing approach," *Gait and Posture*, vol. 26, no. 2, pp. 279–288, 2007.
[105] D. A. Winter, *Biomechanics and motor control of human movement*, 4th ed. Hoboken, NJ, USA: John Wiley & Sons, 2009.
[106] N. Arjmand, O. Ekrami, A. Shirazi-Adl, A. Plamondon, and M. Parnianpour, "Relative performances of artificial neural network and regression mapping tools in evaluation of spinal loads and muscle forces during static lifting," *Journal of Biomechanics*, vol. 46, no. 8, pp. 1454–1462, 2013.
[107] G. Cecchini, G. M. Lozito, M. Schmid, S. Conforto, F. R. Fulginei, and D. Bibbo, "Neural networks for muscle forces prediction in cycling," *Algorithms*, vol. 7, no. 4, pp. 621–634, 2014.
[108] T. T. Dao, "From deep learning to transfer learning for the prediction of skeletal muscle forces," *Medical and Biological Engineering and Computing*, vol. 57, no. 5, pp. 1049–1058, 2019.
[109] M. Vilimek, "An artificial neural network approach and sensitivity analysis in predicting skeletal muscle forces," *Acta of Bioengineering and Biomechanics*, vol. 16, no. 3, pp. 119–127, 2014.
[110] F. Michaud, M. S. Shourijeh, B. J. Fregly, and J. Cuadrado, "Do muscle synergies improve optimization prediction of muscle activations during gait?" *Frontiers in Computational Neuroscience*, vol. 14, p. 54, 2020.
[111] T. Siebert, C. Rode, W. Herzog, O. Till, and R. Blickhan, "Nonlinearities make a difference: Comparison of two common hill-type models with real muscle," *Biological Cybernetics*, vol. 98, no. 2, pp. 133–143, 2008.
[112] F. Moissenet, C. Bélaise, E. Piche, B. Michaud, and M. Begon, "An optimization method tracking EMG, ground reactions forces, and marker trajectories for musculo-tendon forces estimation in equinus gait," *Frontiers in Neurorobotics*, vol. 13, p. 48, 2019.
[113] R. G. Ellis, J. W. Rankin, and J. R. Hutchinson, "Limb kinematics, kinetics and muscle dynamics during the sit-to-stand transition in greyhounds," *Frontiers in Bioengineering and Biotechnology*, vol. 6, p. 162, 2018.
[114] M. Millard, A. L. Emonds, M. Harant, and K. D. Mombaur, "A reduced muscle model and planar musculoskeletal model fit for the simulation of whole-body movements," *Journal of Biomechanics*, vol. 89, pp. 11–20, 2019.
[115] G. Serrancolí, A. L. Kinney, and B. J. Fregly, "Influence of musculoskeletal model parameter values on prediction of accurate knee contact forces during walking," *Medical Engineering and Physics*, vol. 85, pp. 35–47, 2020.
[116] G. Valente, L. Pitto, D. Testi, A. Seth, S. L. Delp, R. Stagni, M. Viceconti, and F. Taddei, "Are subject-specific musculoskeletal models robust to the uncertainties in parameter identification?" *PLoS ONE*, vol. 9, no. 11, p. e112625, 2014.
[117] V. Norman-Gerum and J. McPhee, "Constrained dynamic optimization of sit-to-stand motion driven by Bézier curves," *Journal of Biomechanical Engineering*, vol. 140, no. 12, p. 121011, 2018.
[118] L. Rane, Z. Ding, A. H. McGregor, and A. M. Bull, "Deep learning for musculoskeletal force prediction," *Annals of Biomedical Engineering*, vol. 47, no. 3, pp. 778–789, 2019.
[119] A. Sohane and R. Agarwal, "Knee muscle force estimating model using machine learning approach," *The Computer Journal*, p. bxaa160, 2020.
[120] J. Gonzalez-Vargas, M. Sartori, S. Dosen, D. Torricelli, J. L. Pons, and D. Farina, "A predictive model of muscle excitations based on muscle modularity for a large repertoire of human locomotion conditions," *Frontiers in Computational Neuroscience*, vol. 9, p. 114, 2015.

[121] B. W. Heller, P. H. Veltink, N. J. Rijkhoff, W. L. Rutten, and B. J. Andrews, "Reconstructing muscle activation during normal walking: A comparison of symbolic and connectionist machine learning techniques," *Biological Cybernetics*, vol. 69, no. 4, pp. 327–335, 1993.

[122] S. Jonic and D. Popovic, "Machine learning for prediction of muscle activations for a rule-based controller," *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology*, vol. 4, pp. 1781–1784, 1997.

[123] S. D. Prentice, A. E. Patla, and D. A. Stacey, "Artificial neural network model for the generation of muscle activation patterns for human locomotion," *Journal of Electromyography and Kinesiology*, vol. 11, no. 1, pp. 19–30, 2001.

[124] R. Tibold and A. J. Fuglevand, "Prediction of muscle activity during loaded movements of the upper limb," *Journal of NeuroEngineering and Rehabilitation*, vol. 12, no. 1, pp. 1–12, 2015.

[125] M. Sekiya, S. Sakaino, and T. Toshiaki, "Linear logistic regression for estimation of lower limb muscle activations," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 27, no. 3, pp. 523–532, 2019.

[126] T. McFarland and S. L. Fischer, "Considerations for Industrial Use: A Systematic Review of the Impact of Active and Passive Upper Limb Exoskeletons on Physical Exposures," *IISE Transactions on Occupational Ergonomics and Human Factors*, vol. 7, no. 3-4, pp. 322–347, 2019.

[127] A. Winter, N. Mohajer, and D. Nahavandi, "Semi-active assistive exoskeleton system for elbow joint," in *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics*. Melbourne, Australia: IEEE, 2022, pp. 2347–2353.

[128] T. Rahman, W. Sample, R. Seliktar, M. T. Scavina, A. L. Clark, K. Moran, and M. A. Alexander, "Design and testing of a functional arm orthosis in patients with neuromuscular diseases," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 15, no. 2, pp. 244–251, 2007.

[129] L. Miakovic, M. Dezman, and T. Petric, "Modular quasi-passive mechanism for energy storage applications: towards lightweight high-performance exoskeleton," in *Proceedings of the 20th International Conference on Advanced Robotics*. Ljubljana, Slovenia: IEEE, 2022, pp. 588–593.

[130] A. U. Pehlivan, O. Celik, and M. K. O'Malley, "Mechanical design of a distal arm exoskeleton for stroke and spinal cord injury rehabilitation," in *Proceedings of the IEEE International Conference on Rehabilitation Robotics*. Zurich, Switzerland: IEEE, 2011, pp. 1–5.

[131] C. Di Natali, A. Sadeghi, A. Mondini, E. Bottenberg, B. Hartigan, A. De Eyto, L. W. O'Sullivan, E. Rocon, K. S. Stadler, B. Mazzolai, D. G. Caldwell, and J. Ortiz, "Pneumatic quasi-passive actuation for soft assistive lower limbs exoskeleton," *Frontiers in Neurorobotics*, vol. 14, p. 31, 2020.

[132] A. Otten, C. Voort, A. H. Stienen, R. Aarts, E. Van Asseldonk, and H. Van Der Kooij, "LIMPACT:A hydraulically powered self-aligning upper limb exoskeleton," *IEEE/ASME Transactions on Mechatronics*, vol. 20, no. 5, pp. 2285–2298, 2015.

[133] R. Altenburger, D. Scherly, and K. S. Stadler, "Design of a passive, iso-elastic upper limb exoskeleton for gravity compensation," *Robomech Journal*, vol. 3, no. 1, pp. 1–7, 2016.

[134] Y. Bougrinat, S. Achiche, and M. Raison, "Design and development of a lightweight ankle exoskeleton for human walking augmentation," *Mechatronics*, vol. 64, p. 102297, 2019.

[135] T. Shank, M. Eppes, J. Hossain, M. Gunn, and T. Rahman, "Outcome measures with COPM of children using a Wilmington robotic exoskeleton," *The Open Journal of Occupational Therapy*, vol. 5, no. 1, p. 3, 2017.

[136] M. Jamsek, T. Petric, and J. Babix, "Gaussian mixture models for control of quasi-passive spinal exoskeletons," *Sensors*, vol. 20, no. 9, p. 2705, 2020.

[137] T. du Plessis, K. Djouani, and C. Oosthuizen, "A review of active hand exoskeletons for rehabilitation and assistance," *Robotics*, vol. 10, no. 1, p. 5351–5356, 2021.

[138] M. R. Sajadi, A. Nasr, S. A. A. Moosavian, and H. Zohoor, "Mechanical design, fabrication, kinematics and dynamics modeling, multiple impedance control of a wrist rehabilitation robot," in *Proceedings of the International Conference on Robotics and Mechatronics*. Tehran, Iran: IEEE, 2015, pp. 290–295.

[139] C. Nguiadem, M. Raison, and S. Achiche, "Motion planning of upper-limb exoskeleton robots: A review," *Applied Sciences*, vol. 10, no. 21, pp. 1–21, 2020.

[140] G. Gaudet, M. Raison, and S. Achiche, "Current trends and challenges in pediatric access to sensorless and sensor-based upper limb exoskeletons," *Sensors*, vol. 21, no. 10, 2021.

[141] N. Kashiri, A. Abate, S. J. Abram, A. Albu-Schaffer, P. J. Clary, M. Daley, S. Faraji, R. Furnemont, M. Garabini, H. Geyer, A. M. Grabowski, J. Hurst, J. Malzahn, G. Mathijssen, D. Remy, W. Roozing, M. Shahbazi, S. N. Simha, J. B. Song, N. Smit-Anseeuw, S. Stramigioli, B. Vanderborght, Y. Yesilevskiy, and N. G. Tsagarakis, "An overview on principles for energy efficient robot locomotion," *Frontiers in Robotics and AI*, vol. 5, p. 129, 2018.

[142] N. K. Al-Hayali, S. M. Nacy, J. S. Chiad, and O. Hussein, "Analysis and evaluation of a quasi-passive lower limb exoskeleton for gait rehabilitation," *Al-Khwarizmi Engineering Journal*, vol. 17, no. 4, pp. 36–47, 2021.

[143] N. Kh Al-Hayali, J. S. Chiad, S. M. Nacy, and O. Hussein, "A review of passive and quasi-passive lower limb exoskeletons for gait rehabilitation," *Journal of Mechanical Engineering Research and Developments*, vol. 44, no. 9, pp. 436–447, 2021.

[144] R. P. Matthew, E. J. Mica, W. Meinhold, J. A. Loeza, M. Tomizuka, and R. Bajcsy, "Introduction and initial exploration of an active/passive exoskeleton framework for portable assistance," in *Proceedings of the IEEE International Conference on Intelligent Robots and Systems*, vol. 2015-Decem. Hamburg, Germany: IEEE, 2015, pp. 5351–5356.

[145] J. Naito, A. Nakayama, G. Obinata, and K. Hase, "Development of a wearable robot for assisting carpentry workers," *International Journal of Advanced Robotic Systems*, vol. 4, no. 4, p. 48, 2007.

[146] B. M. Otten, R. Weidner, and A. Argubi-Wollesen, "Evaluation of a novel active exoskeleton for tasks at or above head level," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 2408–2415, 2018.

[147] R. L. Smith, J. Lobo-Prat, H. Van Der Kooij, and A. H. Stienen, "Design of a perfect balance system for active upper-extremity exoskeletons," in *Proceedings of the IEEE International Conference on Rehabilitation Robotics*. Seattle, WA, USA: IEEE, 2013, p. 1–6.

[148] L. Ren, M. Cong, W. Zhang, and Y. Tan, "Harvesting the negative work of an active exoskeleton robot to extend its operating duration," *Energy Conversion and Management*, vol. 245, p. 114640, 2021.

[149] Z. Hassan and W. Sadik, "Design quasi passive exoskeleton for below knee prosthesis," *Journal of Engineering and Applied Sciences*, vol. 13, no. 23, pp. 8994–9001, 2018.

[150] B. G. Lambrecht and H. Kazerooni, "Design of a semi-active knee prosthesis," in *Proceedings of the IEEE International Conference on Robotics and Automation*. Kobe, Japan: IEEE, 2009, pp. 639–645.

[151] M. V. Pillai, L. Van Engelhoven, and H. Kazerooni, "Evaluation of a lower leg support exoskeleton on floor and below hip height panel work," *Human Factors*, vol. 62, no. 3, pp. 489–500, 2020.

[152] Z. Wang, X. Wu, Y. Zhang, C. Chen, S. Liu, Y. Liu, A. Peng, and Y. Ma, "A semi-active exoskeleton based on EMGs reduces muscle fatigue when squatting," *Frontiers in Neurorobotics*, vol. 15, p. 30, 2021.

[153] S. Bai, M. R. U. Islam, K. Hansen, J. Nørgaard, C. Y. Chen, and G. Yang, "A semi-active upper-body exoskeleton for motion assistance," *Biosystems and Biorobotics*, vol. 27, pp. 301–305, 2022.

[154] A. Zahedi, Y. Wang, U. Martinez-Hernandez, and D. Zhang, "A wearable elbow exoskeleton for tremor suppression equipped with rotational semi-active actuator," *Mechanical Systems and Signal Processing*, vol. 157, p. 107674, 2021.

[155] K. A. Inkol and J. McPhee, "Assessing control of fixed-support balance recovery in wearable lower-limb exoskeletons using multibody dynamic modelling," in *Proceedings of the IEEE RAS and EMBS International Conference on Biomedical Robotics and Biomechatronics*. New York, NY, USA: IEEE, 2020, pp. 54–60.

[156] D. Galinski, J. Sapin, and B. Dehez, "Optimal design of an alignment-free two-DOF rehabilitation robot for the shoulder complex," in *Proceedings of the IEEE International Conference on Rehabilitation Robotics*. Seattle, Washington, USA: IEEE, 2013, pp. 1–7.

[157] M. Sarac, M. Solazzi, E. Sotgiu, M. Bergamasco, and A. Frisoli, "Design and kinematic optimization of a novel underactuated robotic hand exoskeleton," *Meccanica*, vol. 52, no. 3, pp. 749–761, 2017.

[158] P. Manns, M. Sreenivasa, M. Millard, and K. D. Mombaur, "Motion optimization and parameter identification for a human and lower back exoskeleton model," *IEEE Robotics and Automation Letters*, vol. 2, no. 3, pp. 1564–1570, 2017.

[159] Y. Shi and Q. Peng, "Improved benchmarking method using kinematics analysis in design of an upper limb exoskeleton rehabilitation device," in *Proceedings of the ASME International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, vol. 4. Quebec City, Quebec, Canada: ASME, 2018, p. V004T05A001.

[160] Y. Hayashi, R. Dubey, and K. Kiguchi, "Torque optimization for a 7DOF upper-limb power-assist exoskeleton robot," in *Proceedings of the IEEE Workshop on Robotic Intelligence In Informationally Structured Space*. Paris, France: IEEE, 2011, pp. 49–54.

[161] G. D. Marinou and K. D. Mombaur, "Optimizing active spinal exoskeletons to minimize low back loads," *Biosystems and Biorobotics*, vol. 27, pp. 455–460, 2022.

[162] Y. Aoustin and A. M. Formalskii, "Walking of biped with passive exoskeleton: evaluation of energy consumption," *Multibody System Dynamics*, vol. 43, no. 1, pp. 71–96, 2018.

[163] C. M. Harbauer, M. Fleischer, C. E. Bandmann, and K. Bengler, "Optimizing force transfer in a soft exoskeleton using biomechanical modeling," in *Proceedings of the 21st Congress of the International Ergonomics Association*, vol. 223. Vancouver, BC, Canada: Springer, 2022, pp. 274–281.

[164] L. Zhou, Y. Li, and S. Bai, "A human-centered design optimization approach for robotic exoskeletons through biomechanical simulation," *Robotics and Autonomous Systems*, vol. 91, pp. 337–347, 2017.

[165] D. Park and K. J. Cho, "Development and evaluation of a soft wearable weight support device for reducing muscle fatigue on shoulder," *PLoS ONE*, vol. 12, no. 3, p. e0173730, 2017.

[166] N. Mehrabi, R. S. Razavian, B. Ghannadi, and J. McPhee, "Predictive simulation of reaching moving targets using nonlinear model predictive control," *Frontiers in Computational Neuroscience*, vol. 10, p. 143, 2017.

[167] R. Mallat, M. Khalil, G. Venture, V. Bonnet, and S. Mohammed, "Human-Exoskeleton Joint Misalignment: A Systematic Review," in *International Conference on Advances in Biomedical Engineering, ICABME*, vol. 2019-Octob. Tripoli, Lebanon: IEEE, 2019, pp. 1–4.

[168] E. Panero, G. G. Muscolo, L. Gastaldi, and S. Pastorelli, "Multibody analysis of a 3D human model with trunk exoskeleton for industrial applications," *Computational Methods in Applied Sciences*, vol. 53, pp. 43–51, 2020.
[169] C. Cortes, A. De Los Reyes-Guzman, D. Scorza, A. Bertelsen, E. Carrasco, A. Gil-Agudo, O. Ruiz-Salguero, and J. Florez, "Inverse kinematics for upper limb compound movement estimation in exoskeleton-assisted rehabilitation," *BioMed Research International*, vol. 2016, pp. 1–14, 2016.
[170] A. Nasr and J. McPhee, "Biarticular MuscleNET: A machine learning model of biarticular muscles," in *Proceedings of the North American Congress on Biomechanics*, Ottawa, Canada, 2022.
[171] D. Copaci, D. Serrano, L. Moreno, and D. Blanco, "A high-level control algorithm based on sEMG signalling for an elbow joint SMA exoskeleton," *Sensors*, vol. 18, no. 8, p. 2522, 2018.
[172] A. J. Young and D. P. Ferris, "State of the art and future directions for lower limb robotic exoskeletons," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 25, no. 2, pp. 171–182, 2017.
[173] K. Zhu, T. Xue, T. Zhang, and M. Zhang, "SEMG-based joint moment estimation for hip exoskeleton general assistive strategy," in *Proceedings of the Chinese Automation Congress*. Hangzhou, China: IEEE, 2019, pp. 3826–3830.
[174] A. Rudenko, L. Palmieri, M. Herman, K. M. Kitani, D. M. Gavrila, and K. O. Arras, "Human motion trajectory prediction: A survey," *International Journal of Robotics Research*, vol. 39, no. 8, pp. 895–935, 2020.
[175] K. Abdel-Malek and J. Arora, *Human motion simulation: Predictive dynamics*. Elsevier, 2013.
[176] C. Yang, X. Xi, S. Chen, S. M. Miran, X. Hua, and Z. Luo, "SEMG-based multifeatures and predictive model for knee-joint-angle estimation," *AIP Advances*, vol. 9, no. 9, p. 095042, 2019.
[177] A. Fleming, N. Stafford, S. Huang, X. Hu, D. P. Ferris, and H. H. Huang, "Myoelectric control of robotic lower limb prostheses: A review of electromyography interfaces, control paradigms, challenges and future directions," *Journal of Neural Engineering*, vol. 18, p. 041004, 2021.
[178] T. Kawase, S. Takeshi, K. Yasuharu, and K. Kenji, "A hybrid BMI-based exoskeleton for paresis: EMG control for assisting arm movements," *Journal of Neural Engineering*, vol. 14, no. 1, p. 016015, 2017.
[179] M. Asghari Oskoei and H. Hu, "Myoelectric control systems-A survey," *Biomedical Signal Processing and Control*, vol. 2, no. 4, pp. 275–294, 2007.
[180] D. Farina, N. Jiang, H. Rehbaum, A. Holobar, B. Graimann, H. Dietl, and O. C. Aszmann, "The extraction of neural information from the surface EMG for the control of upper-limb prostheses: Emerging avenues and challenges," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 22, no. 4, pp. 797–809, 2014.
[181] M. Atzori, M. Cognolato, and H. Müller, "Deep learning with convolutional neural networks applied to electromyography data: A resource for the classification of movements for prosthetic hands," *Frontiers in Neurorobotics*, vol. 10, p. 9, 2016.
[182] U. Côté-Allard, C. L. Fall, A. Drouin, A. Campeau-Lecours, C. Gosselin, K. Glette, F. Laviolette, and B. Gosselin, "Deep learning for electromyographic hand gesture signal classification using transfer learning," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 27, no. 4, pp. 760–771, 2019.
[183] L. Inzelberg, M. David-Pur, E. Gur, and Y. Hanein, "Multi-channel electromyography-based mapping of spontaneous smiles," *Journal of Neural Engineering*, vol. 17, no. 2, p. 026025, 2020.
[184] F. Xiao, Y. Chen, and Y. Zhu, "GADF/GASF-HOG: Feature extraction methods for hand movement classification from surface electromyography," *Journal of Neural Engineering*, vol. 17, no. 4, p. 046016, 2020.
[185] A. J. Young, L. H. Smith, E. J. Rouse, and L. J. Hargrove, "Classification of simultaneous movements using surface EMG pattern recognition," *IEEE Transactions on Biomedical Engineering*, vol. 60, no. 5, pp. 1250–1258, 2013.
[186] X. Zhai, B. Jelfs, R. H. Chan, and C. Tin, "Self-recalibrating surface EMG pattern recognition for neuroprosthesis control based on convolutional neural network," *Frontiers in Neuroscience*, vol. 11, p. 379, 2017.
[187] L. Zhizeng, W. Fei, and M. Wenjie, "Pattern classification of surface electromyography based on AR model and high-order neural network," in *Proceedings of the 2nd IEEE/ASME International Conference on Mechatronic and Embedded Systems and Applications*. Beijing, China: IEEE, 2007, pp. 1–6.
[188] M. Ziaur ur Rehman, A. Waris, S. O. Gilani, M. Jochumsen, I. K. Niazi, M. Jamil, D. Farina, and E. N. Kamavuako, "Multiday EMG-based classification of hand motions with deep learning techniques," *Sensors*, vol. 18, no. 8, p. 2497, 2018.
[189] Y. Chen, C. Dai, and W. Chen, "Cross-comparison of EMG-to-force methods for multi-DoF finger force prediction using one-DoF training," *IEEE Access*, vol. 8, pp. 13 958–13 968, 2020.
[190] R. Hu, X. X. X. Chen, C. Huang, S. Cao, X. Zhang, and X. X. X. Chen, "Elbow-flexion force estimation during arm posture dynamically changing between pronation and supination," *Journal of Neural Engineering*, vol. 16, no. 6, p. 066005, 2019.
[191] C. Chen, K. Huang, D. Li, Z. Zhao, and J. Hong, "Multi-segmentation parallel CNN model for estimating assembly torque using surface electromyography signals," *Sensors*, vol. 20, no. 15, pp. 1–22, 2020.
[192] C. Li, G. Li, G. Jiang, D. Chen, and H. Liu, "Surface EMG data aggregation processing for intelligent prosthetic action recognition," *Neural Computing and Applications*, vol. 32, no. 22, pp. 16 795–16 806, 2020.

[193] J. Liu, S. H. Kang, D. Xu, Y. Ren, S. J. Lee, and L. Q. Zhang, "EMG-Based continuous and simultaneous estimation of arm kinematics in able-bodied individuals and stroke survivors," *Frontiers in Neuroscience*, vol. 11, p. 480, 2017.

[194] Y. Zhang, X. Zhang, Z. Lu, Z. Jiang, and T. Zhang, "A novel wrist joint torque prediction method based on EMG and LSTM," in *Proceedings of the 10th IEEE International Conference on Cyber Technology in Automation, Control and Intelligent Systems*. Xi'an, China: IEEE, 2020, pp. 242–245.

[195] N. Jiang, J. L. Vest-Nielsen, S. Muceli, and D. Farina, "EMG-based simultaneous and proportional estimation of wrist/hand kinematics in uni-lateral trans-radial amputees," *Journal of NeuroEngineering and Rehabilitation*, vol. 9, no. 1, p. 42, 2012.

[196] C. Castellini and P. Van Der Smagt, "Surface EMG in advanced hand prosthetics," *Biological Cybernetics*, vol. 100, no. 1, pp. 35–47, 2009.

[197] Y. Du, Y. Wong, W. Jin, W. Wei, Y. Hu, M. Kankanhalli, and W. Geng, "Semi-supervised learning for surface EMG-based gesture recognition," in *Proceedings of the IJCAI International Joint Conference on Artificial Intelligence*, vol. 0. Melbourne, Australia: AAAI Press, 2017, pp. 1624–1630.

[198] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 8, pp. 1798–1828, 2013.

[199] L. J. Hargrove, B. A. Lock, and A. M. Simon, "Pattern recognition control outperforms conventional myoelectric control in upper limb patients with targeted muscle reinnervation," in *Proceedings of the 35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*. Osaka, Japan: IEEE, 2013, pp. 1599–1602.

[200] S. Muceli and D. Farina, "Simultaneous and proportional estimation of hand kinematics from EMG during mirrored movements at multiple degrees-of-freedom," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 20, no. 3, pp. 371–378, 2012.

[201] A. Phinyomark, P. Phukpattaranont, and C. Limsakul, "Feature reduction and selection for EMG signal classification," *Expert Systems with Applications*, vol. 39, no. 8, pp. 7420–7431, 2012.

[202] E. Scheme and K. Englehart, "Training strategies for mitigating the effect of proportional control on classification in pattern recognition-based myoelectric control," *Journal of Prosthetics and Orthotics*, vol. 25, no. 2, pp. 76–83, 2013.

[203] A. M. Simon, L. J. Hargrove, B. A. Lock, and T. A. Kuiken, "Target achievement control test: Evaluating real-time myoelectric pattern-recognition control of multifunctional upper-limb prostheses," *Journal of Rehabilitation Research and Development*, vol. 48, no. 6, pp. 619–628, 2011.

[204] A. Phinyomark and E. Scheme, "EMG pattern recognition in the era of big data and deep learning," *Big Data and Cognitive Computing*, vol. 2, no. 3, pp. 1–27, 2018.

[205] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Communications of the ACM*, vol. 60, no. 6, pp. 84–90, 2017.

[206] O. Abdel-Hamid, A. R. Mohamed, H. Jiang, L. Deng, G. Penn, and D. Yu, "Convolutional neural networks for speech recognition," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 22, no. 10, pp. 1533–1545, 2014.

[207] Q. Wang, H. Wang, F. Hu, C. Hua, and D. Wang, "Using convolutional neural networks to decode EEG-based functional brain network with different severity of acrophobia," *Journal of Neural Engineering*, vol. 18, no. 1, p. 016007, 2020.

[208] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2323, 1998.

[209] A. Ameri, M. A. Akhaee, E. Scheme, and K. Englehart, "Real-time, simultaneous myoelectric control using a convolutional neural network," *PLoS ONE*, vol. 13, no. 9, p. e0203835, 2018.

[210] W. Geng, Y. Du, W. Jin, W. Wei, Y. Hu, and J. Li, "Gesture recognition by instantaneous surface EMG images," *Scientific Reports*, vol. 6, p. 36571, 2016.

[211] T. Bao, A. Zaidi, S. Xie, and Z. Zhang, "Surface-EMG based wrist kinematics estimation using convolutional neural network," in *Proceedings of the 16th International Conference on Wearable and Implantable Body Sensor Networks*. Chicago, IL, USA: IEEE, 2019, pp. 1–4.

[212] H. Dantas, D. J. Warren, S. M. Wendelken, T. S. Davis, G. A. Clark, and V. J. Mathews, "Deep learning movement intent decoders trained with dataset aggregation for prosthetic limb control," *IEEE Transactions on Biomedical Engineering*, vol. 66, no. 11, pp. 3192–3203, 2019.

[213] A. Ameri, M. A. Akhaee, E. Scheme, and K. Englehart, "Regression convolutional neural network for improved simultaneous EMG control," *Journal of Neural Engineering*, vol. 16, no. 3, p. 36015, 2019.

[214] W. Wangshow, B. Chen, P. Xia, J. Hu, and Y. Peng, "Sensor fusion for myoelectric control based on deep learning with recurrent convolutional neural networks," *Artificial Organs*, vol. 42, no. 9, pp. E272–E282, 2018.

[215] P. Xia, J. Hu, and Y. Peng, "EMG-based estimation of limb movement using deep learning with recurrent convolutional neural networks," *Artificial Organs*, vol. 42, no. 5, pp. E67–E77, 2018.

[216] Y. Liu, D. Xin, J. Hua, and M. Liu, "SEMG motion intention recognition based on wavelet time-frequency spectrum and ConvLSTM," *Journal of Physics: Conference Series*, vol. 1631, no. 1, 2020.

[217] M. Akmal, S. Zubair, M. Jochumsen, E. N. Kamavuako, and I. K. Niazi, "A tensor-based method for completion of missing electromyography data," *IEEE Access*, vol. 7, pp. 104 710–104 720, 2019.
[218] A. Naber, E. Mastinu, and M. Ortiz-Catalan, "Stationary wavelet processing and data imputing in myoelectric pattern recognition on a low-cost embedded system," *IEEE Transactions on Medical Robotics and Bionics*, vol. 1, no. 4, pp. 256–266, 2019.
[219] S. Conforto, P. A. Mathieu, M. Schmid, D. Bibbo, J. R. Florestal, and T. D'Alessio, "How much can we trust the electromechanical delay estimated by using electromyography?" in *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology*. New York, NY, USA: IEEE, 2006, pp. 1256–1259.
[220] H. Sözen, E. Cè, A. V. Bisconti, S. Rampichini, S. Longo, G. Coratella, S. Shokohyar, C. Doria, M. Borrelli, E. Limonta, and F. Esposito, "Differences in electromechanical delay components induced by sex, age and physical activity level: New insights from a combined electromyographic, mechanomyographic and force approach," *Sport Sciences for Health*, vol. 15, no. 3, pp. 623–633, 2019.
[221] G. Durandau, D. Farina, G. Asín-Prieto, I. Dimbwadyo-Terrer, S. Lerma-Lara, J. L. Pons, J. C. Moreno, and M. Sartori, "Voluntary control of wearable robotic exoskeletons by patients with paresis via neuromechanical modeling," *Journal of NeuroEngineering and Rehabilitation*, vol. 16, no. 1, pp. 1–18, 2019.
[222] G. Durandau, D. Farina, and M. Sartori, "Robust real-time musculoskeletal modeling driven by electromyograms," *IEEE Transactions on Biomedical Engineering*, vol. 65, no. 3, pp. 556–564, 2018.
[223] B. Kalita, J. Narayan, and S. K. Dwivedy, "Development of active lower limb robotic-based orthosis and exoskeleton devices: A systematic review," *International Journal of Social Robotics*, vol. 13, no. 4, pp. 775–793, 2021.
[224] P. Malcolm, S. Galle, and D. De Clercq, "Fast exoskeleton optimization," *Science*, vol. 356, no. 6344, pp. 1230–1231, 2017.
[225] J. Zhang, P. Fiers, K. A. Witte, R. W. Jackson, K. L. Poggensee, C. G. Atkeson, and S. H. Collins, "Human-in-the-loop optimization of exoskeleton assistance during walking," *Science*, vol. 356, no. 6344, pp. 1280–1283, 2017.
[226] A. Hashemi and J. McPhee, "Assistive sliding mode control of a rehabilitation robot with automatic weight adjustment," in *Proceedings of the 43rd Annual International Conference of the IEEE Engineering in Medicine & Biology Society*. Mexico: IEEE, 2021, pp. 4891–4896.
[227] M. Sartori, G. Durandau, S. Došen, and D. Farina, "Robust simultaneous myoelectric control of multiple degrees of freedom in wrist-hand prostheses by real-time neuromusculoskeletal modeling," *Journal of Neural Engineering*, vol. 15, no. 6, p. 066026, 2018.
[228] A. Waris, I. Mendez, K. Englehart, W. Jensen, and E. N. Kamavuako, "On the robustness of real-time myoelectric control investigations: A multiday Fitts' law approach," *Journal of Neural Engineering*, vol. 16, no. 2, p. 026003, 2019.
[229] C. Chen, G. Chai, W. Guo, X. Sheng, D. Farina, and X. Zhu, "Prediction of finger kinematics from discharge timings of motor units: Implications for intuitive control of myoelectric prostheses," *Journal of Neural Engineering*, vol. 16, no. 2, p. 026005, 2019.
[230] H. H. Huang, J. Si, A. Brandt, and M. Li, "Taking both sides: Seeking symbiosis between intelligent prostheses and human motor control during locomotion," *Current Opinion in Biomedical Engineering*, vol. 20, p. 100314, 2021.
[231] T. Akbas and J. S. Sulzer, "Musculoskeletal simulation framework for impairment-based exoskeletal assistance post-stroke," in *Proceedings of the EEE International Conference on Rehabilitation Robotics*, vol. 2019-June. Toronto, ON, Canada: IEEE, 2019, pp. 1185–1190.
[232] Z. Li, K. Zhao, L. Zhang, X. Wu, T. Zhang, Q. Li, X. Li, and C. Y. Su, "Human-in-the-loop control of a wearable lower limb exoskeleton for stable dynamic walking," *IEEE/ASME Transactions on Mechatronics*, vol. 26, no. 5, pp. 2700–2711, 2021.
[233] Y. Wang, H. Wang, and Y. Tian, "Adaptive interaction torque-based AAN control for lower limb rehabilitation exoskeleton," *ISA Transactions*, p. In press, 2021.
[234] Y. Wen, J. Si, A. Brandt, X. Gao, and H. H. Huang, "Online reinforcement learning control for the personalization of a robotic knee prosthesis," *IEEE Transactions on Cybernetics*, vol. 50, no. 6, pp. 2346–2356, 2020.
[235] N. Naghavi, A. Akbarzadeh, S. M. Tahamipour Zarandi, and I. Kardan, "Assist-as-needed control of a hip exoskeleton based on a novel strength index," *Robotics and Autonomous Systems*, vol. 134, p. 103667, 2020.
[236] J. Liu, Y. Ren, D. Xu, S. H. Kang, and L. Q. Zhang, "EMG-based real-time linear-nonlinear cascade regression decoding of shoulder, elbow, and wrist movements in able-bodied persons and stroke survivors," *IEEE Transactions on Biomedical Engineering*, vol. 67, no. 5, pp. 1272–1281, 2020.
[237] F. Xiao, "Proportional myoelectric and compensating control of a cable-conduit mechanism-driven upper limb exoskeleton," *ISA Transactions*, vol. 89, pp. 245–255, 2019.
[238] M. A. Gull, S. Bai, and T. Bak, "A review on design of upper limb exoskeletons," *Robotics*, vol. 9, no. 1, p. 16, 2020.
[239] B. Noronha and D. Accoto, "Exoskeletal devices for hand assistance and rehabilitation: A comprehensive analysis of state-of-the-art technologies," *IEEE Transactions on Medical Robotics and Bionics*, vol. 3, no. 2, pp. 525–538, 2021.

[240] E. Trigili, S. Crea, M. Moisè, A. Baldoni, M. Cempini, G. Ercolini, D. Marconi, F. Posteraro, M. C. Carrozza, and N. Vitiello, "Design and experimental characterization of a shoulder-elbow exoskeleton with compliant joints for post-stroke rehabilitation," *IEEE/ASME Transactions on Mechatronics*, vol. 24, no. 4, pp. 1485–1496, 2020.
[241] L. Shore, V. Power, A. de Eyto, and L. W. O'Sullivan, "Technology acceptance and user-centred design of assistive exoskeletons for older adults: A commentary," *Robotics*, vol. 7, no. 1, p. 3, 2018.
[242] W. Van Dijk and H. Van Der Kooij, "XPED2: A passive exoskeleton with artificial tendons," *IEEE Robotics and Automation Magazine*, vol. 21, no. 4, pp. 56–61, 2014.
[243] K. A. Witte and S. H. Collins, "Design of lower-limb exoskeletons and emulator systems," in *Wearable Robotics: Systems and Applications*. Amsterdam: Elsevier, 2019, ch. 13, pp. 251–274.
[244] H. J. Asl, K. Katagiri, T. Narikiyo, M. Yamashita, and M. Kawanishi, "Satisfying task completion and assist-as-needed performance in robotic exoskeletons," *IEEE Transactions on Medical Robotics and Bionics*, vol. 3, no. 3, pp. 791–800, 2021.
[245] X. Cui, W. Chen, X. Jin, and S. K. Agrawal, "Design of a 7-DOF cable-driven arm exoskeleton (CAREX-7) and a controller for dexterous motion training or assistance," *IEEE/ASME Transactions on Mechatronics*, vol. 22, no. 1, pp. 161–172, 2017.
[246] Y. Yihun, V. Adhikari, A. Majidirad, and J. Desai, "Task-Based knee rehabilitation with assist-as-needed control strategy and recovery tracking system," *Journal of Engineering and Science in Medical Diagnostics and Therapy*, vol. 3, no. 2, pp. 19–1051, 2020.
[247] Y. Long, Z. Du, L. Cong, W. Wang, Z. Zhang, and W. Dong, "Active disturbance rejection control based human gait tracking for lower extremity rehabilitation exoskeleton," *ISA Transactions*, vol. 67, pp. 389–397, 2017.
[248] D. Sanz-Merodio, M. Cestari, J. C. Arevalo, X. A. Carrillo, and E. Garcia, "Generation and control of adaptive gaits in lower-limb exoskeletons for motion assistance," *Advanced Robotics*, vol. 28, no. 5, pp. 329–338, 2014.
[249] J. Rosen and J. C. Perry, "Upper limb powered exoskeleton," *International Journal of Humanoid Robotics*, vol. 4, no. 3, pp. 529–548, 2007.
[250] Z. Tang, K. Zhang, S. Sun, Z. Gao, L. Zhang, and Z. Yang, "An upper-limb power-assist exoskeleton using proportional myoelectric control," *Sensors*, vol. 14, no. 4, pp. 6677–6694, 2014.
[251] L. Teng, M. A. Gull, and S. Bai, "PD-Based fuzzy sliding mode control of a wheelchair exoskeleton robot," *IEEE/ASME Transactions on Mechatronics*, vol. 25, no. 5, pp. 2546–2555, 2020.
[252] J. Han, S. Yang, L. Xia, and Y. H. Chen, "Deterministic adaptive robust control with a novel optimal gain design approach for a fuzzy 2-DOF lower limb exoskeleton robot system," *IEEE Transactions on Fuzzy Systems*, vol. 29, no. 8, pp. 2373–2387, 2021.
[253] K. Kiguchi, M. H. Rahman, M. Sasaki, and K. Teramoto, "Development of a 3DOF mobile exoskeleton robot for human upper-limb motion assist," *Robotics and Autonomous Systems*, vol. 56, no. 8, pp. 678–691, 2008.
[254] A. Mauri, J. Lettori, G. Fusi, D. Fausti, M. Mor, F. Braghin, G. Legnani, and L. Roveda, "Mechanical and control design of an industrial exoskeleton for advanced human empowering in heavy parts manipulation tasks," *Robotics*, vol. 8, no. 3, p. 65, 2019.
[255] X. Bao, V. Molazadeh, A. Dodson, and N. Sharma, "Model predictive control-based knee actuator allocation during a standing-up motion with a powered exoskeleton and functional electrical stimulation," *Advances in Motor Neuroprostheses*, pp. 89–100, 2020.
[256] J. Hu, X. Xu, and K. Zhu, "Arm exoskeleton based on model predictive control with input/output feedback linearization," *Journal of Medical Imaging and Health Informatics*, vol. 3, no. 3, pp. 432–439, 2013.
[257] C. A. Rodriguez, P. Ponce, and A. Molina, "ANFIS and MPC controllers for a reconfigurable lower limb exoskeleton," *Soft Computing*, vol. 21, no. 3, pp. 571–584, 2017.
[258] T. Teramae, T. Noda, and J. Morimoto, "EMG-based model predictive control for physical human-robot interaction: application for assist-as-needed control," *IEEE Robotics and Automation Letters*, vol. 3, no. 1, pp. 210–217, 2018.
[259] V. Molazadeh, Q. Zhang, X. Bao, and N. Sharma, "Neural-network based iterative learning control of a hybrid exoskeleton with an MPC allocation strategy," in *Proceedings of the ASME Dynamic Systems and Control Conference*, vol. 1. Park City, Utah, USA: ASME, 2019, p. V001T05A011.
[260] G. Menga and M. Ghirardi, "Lower limb exoskeleton for rehabilitation with improved postural equilibrium," *Robotics*, vol. 7, no. 2, p. 28, 2018.
[261] W. Yu and J. Rosen, "A novel linear PID controller for an upper limb exoskeleton," in *Proceedings of the IEEE Conference on Decision and Control*. Atlanta, GA, USA: IEEE, 2010, pp. 3548–3553.
[262] L. Huang, J. R. Steger, and H. Kazerooni, "Hybrid control of the Berkeley lower extremity exoskeleton (BLEEX)," *Dynamic Systems and Control, Parts A and B*, vol. 74 DSC, no. 2 PART B, pp. 1429–1436, 2005.
[263] I. Jammeli, A. Chemori, H. Moon, S. Elloumi, and S. Mohammed, "An assistive explicit model predictive control framework for a knee rehabilitation exoskeleton," *IEEE/ASME Transactions on Mechatronics*, pp. 1–12, 2021.
[264] S. Li, Y. Shi, L. Hu, and Z. Sun, "A generalized model predictive control method for series elastic actuator driven exoskeleton robots," *Computers and Electrical Engineering*, vol. 94, p. 107328, 2021.

[265] Y. Cao and J. Huang, "Neural-network-based nonlinear model predictive tracking control of a pneumatic muscle actuator-driven exoskeleton," *IEEE/CAA Journal of Automatica Sinica*, vol. 7, no. 6, pp. 1478–1488, 2020.

[266] W. Chen, B. Zhang, J. Yao, and X. Zhao, "Control of SEA modular joint for rehabilitation exoskeleton based on modified model predictive control," in *Proceedings of the 10th Annual International Conference on CYBER Technology in Automation, Control, and Intelligent Systems*. Jiaxing, China: IEEE, 2021, pp. 527–532.

[267] S. M. Tahamipour Zarandi, S. K. Hosseini Sani, A. Akbarzadeh, and I. Kardan, "An assistive strategy for compliantly actuated exoskeletons using non-linear model predictive control method," in *Proceedings of the 26th Iranian Conference on Electrical Engineering*. Mashhad, Iran: IEEE, 2018, pp. 982–987.

[268] S. M. Tahamipour Zarandi, S. K. Hosseini Sani, M. R. Akbarzadeh Tootoonchi, A. Akbarzadeh Tootoonchi, and M. G. Farajzadeh-D, "Design and implementation of a real-time nonlinear model predictive controller for a lower limb exoskeleton with input saturation," *Iranian Journal of Science and Technology, Transactions of Electrical Engineering*, vol. 45, no. 1, pp. 309–320, 2021.

[269] U. Keller, H. J. Van Hedel, V. Klamroth-Marganska, and R. Riener, "ChARMin: The first actuated exoskeleton robot for pediatric arm rehabilitation," *IEEE/ASME Transactions on Mechatronics*, vol. 21, no. 5, pp. 2201–2213, 2016.

[270] S. Dehghani and F. Bahrami, "How does the CNS control arm reaching movements? Introducing a hierarchical nonlinear predictive control organization based on the idea of muscle synergies," *PLoS ONE*, vol. 15, no. 2, p. e0228726, 2020.

[271] K. D. Mombaur and D. Clever, "Inverse optimal control as a tool to understand human movement," in *Geometric and numerical foundations of movements*, J.-P. Laumond, , N. Mansard, , and J.-B. Lasserre, Eds. Springer, 2017, vol. 117, pp. 163–186.

[272] R. Shadmehr and H. H. Holcomb, "Inhibitory control of competing motor memories," *Experimental Brain Research*, vol. 126, no. 2, pp. 235–251, 1999.

[273] R. Nasiri, M. Shushtari, and A. Arami, "An adaptive assistance controller to optimize the exoskeleton contribution in rehabilitation," *Robotics*, vol. 10, no. 3, p. 95, 2021.

[274] M. Shushtari, R. Nasiri, and A. Arami, "Online reference trajectory adaptation: A personalized control strategy for lower limb exoskeletons," *IEEE Robotics and Automation Letters*, vol. 7, no. 1, pp. 128–134, 2022.

[275] T. M. Morrison, P. Pathmanathan, M. Adwan, and E. Margerrison, "Advancing regulatory science with computational modeling for medical devices at the FDA's office of science and engineering laboratories," *Frontiers in Medicine*, vol. 5, no. SEP, p. 241, 2018.

[276] R. E. Hughes, M. E. Johnson, S. W. O'Driscoll, and K. N. An, "Age-related changes in normal isometric shoulder strength," *American Journal of Sports Medicine*, vol. 27, no. 5, pp. 651–657, 1999.

[277] P. Gentil, B. Fischer, A. S. Martorelli, R. M. Lima, and M. Bottaro, "Effects of equal-volume resistance training performed one or two times a week in upper body muscle size and strength of untrained young men," *Journal of Sports Medicine and Physical Fitness*, vol. 55, no. 3, pp. 144–149, 2015.

[278] G. Wu, F. C. Van Der Helm, H. E. Veeger, M. Makhsous, P. Van Roy, C. Anglin, J. Nagels, A. R. Karduna, K. McQuade, X. Wang, F. W. Werner, and B. Buchholz, "ISB recommendation on definitions of joint coordinate systems of various joints for the reporting of human joint motion - Part II: Shoulder, elbow, wrist and hand," *Journal of Biomechanics*, vol. 38, no. 5, pp. 981–992, 2005.

[279] J. G. Reid and R. K. Jensen, "Human body segment inertia parameters: A survey and status report," *Exercise and Sport Sciences Reviews*, vol. 18, no. 1, pp. 225–241, 1990.

[280] J. L. Durkin and J. J. Dowling, "Analysis of body segment parameter differences between four human populations and the estimation errors of four popular mathematical models," *Journal of Biomechanical Engineering*, vol. 125, no. 4, pp. 515–522, 2003.

[281] M. J. Pavol, T. M. Owings, and M. D. Grabiner, "Body segment inertial parameter estimation for the general population of older adults," *Journal of Biomechanics*, vol. 35, no. 5, pp. 707–712, 2002.

[282] K. Moromizato, R. Kimura, H. Fukase, K. Yamaguchi, and H. Ishida, "Whole-body patterns of the range of joint motion in young adults: Masculine type and feminine type," *Journal of Physiological Anthropology*, vol. 35, no. 1, pp. 1–12, 2016.

[283] P. Parijat, T. E. Lockhart, and J. Liu, "EMG and kinematic responses to unexpected slips after slip training in virtual reality," *IEEE Transactions on Biomedical Engineering*, vol. 62, no. 2, pp. 593–599, 2015.

[284] T. S. Keller and A. L. Roy, "Posture-dependent isometric trunk extension and flexion strength in normal male and female subjects," *Journal of Spinal Disorders and Techniques*, vol. 15, no. 4, pp. 312–318, 2002.

[285] J. N. Côté, D. Raymond, P. A. Mathieu, A. G. Feldman, and M. F. Levin, "Differences in multi-joint kinematic patterns of repetitive hammering in healthy, fatigued and shoulder-injured individuals," *Clinical Biomechanics*, vol. 20, no. 6, pp. 581–590, 2005.

[286] Z. Bańkosz and S. Winiarski, "Correlations between angular velocities in selected joints and velocity of table tennis racket during topspin Forehand and Backhand," *Journal of Sports Science and Medicine*, vol. 17, no. 2, pp. 330–338, 2018.

[287] V. F. Ferrario, C. Sforza, G. Serrao, G. P. Grassi, and E. Mossi, "Active range of motion of the head and cervical spine: A three-dimensional investigation in healthy young adults," *Journal of Orthopaedic Research*, vol. 20, no. 1, pp. 122–129, 2002.
[288] M. Simoneau, M. Denninger, and T. C. Hain, "Role of loading on head stability and effective neck stiffness and viscosity," *Journal of Biomechanics*, vol. 41, no. 10, pp. 2097–2103, 2008.
[289] Z. Dvir and T. Prushansky, "Reproducibility and instrument validity of a new ultrasonography-based system for measuring cervical spine kinematics," *Clinical Biomechanics*, vol. 15, no. 9, pp. 658–664, 2000.
[290] N. Doriot and X. Wang, "Effects of age and gender on maximum voluntary range of motion of the upper body joints," *Ergonomics*, vol. 49, no. 3, pp. 269–281, 2006.
[291] A. E. Engin and I. Kaleps, "Active muscle torques about long-bone axes of major human joints," *Aviation Space and Environmental Medicine*, vol. 51, no. 6, pp. 551–555, 1980.
[292] B. A. Garner and M. G. Pandy, "Musculoskeletal model of the upper limb based on the visible human male dataset," *Computer Methods in Biomechanics and Biomedical Engineering*, vol. 4, no. 2, pp. 93–126, 2001.
[293] J. M. Soucie, C. Wang, A. Forsyth, S. Funk, M. Denny, K. E. Roach, and D. Boone, "Range of motion measurements: Reference values and a database for comparison studies," *Haemophilia*, vol. 17, no. 3, pp. 500–507, 2011.
[294] E. L. Hutchins, "The musculoskeletal geometry of the human elbow and wrist: An analysis using torque-angle relationships," Ph.D. dissertation, University of Texas at Austin, Austin, TX, USA, 1993.
[295] R. Van Den Tillaar and G. Ettema, "A three-dimensional analysis of overarm throwing in experienced handball players," *Journal of Applied Biomechanics*, vol. 23, no. 1, pp. 12–19, 2007.
[296] W. Y. Fu, G. Cheng, Y. F. Ma, and A. P. Yang, "Experimental research of range of motion about wrist joint," in *Proceedings of the International Conference on Digital Human Modeling and Applications in Health, Safety, Ergonomics and Risk Management*, vol. 9745. Toronto, ON, Canada: Springer, 2016, pp. 3–12.
[297] S. L. Delp, A. E. Grierson, and T. S. Buchanan, "Maximum isometric moments generated by the wrist muscles in flexion-extension and radial-ulnar deviation," *Journal of Biomechanics*, vol. 29, no. 10, pp. 1371–1375, 1996.
[298] M. G. Lewis, M. R. Yeadon, and M. A. King, "The effect of accounting for biarticularity in hip flexor and hip extensor joint torque representations," *Human Movement Science*, vol. 57, pp. 388–399, 2018.
[299] D. E. Anderson, M. L. Madigan, and M. A. Nussbaum, "Maximum voluntary joint torque as a function of joint angle and angular velocity: Model development and application to the lower limb," *Journal of Biomechanics*, vol. 40, no. 14, pp. 3105–3113, 2007.
[300] P. D. Hoang, R. B. Gorman, G. Todd, S. C. Gandevia, and R. D. Herbert, "A new method for measuring passive length-tension properties of human gastrocnemius muscle in vivo," *Journal of Biomechanics*, vol. 38, no. 6, pp. 1333–1341, 2005.
[301] Y. S. Yoon and J. M. Mansour, "The passive elastic moment at the hip," *Journal of Biomechanics*, vol. 15, no. 12, pp. 905–910, 1982.
[302] G. T. Yamaguchi, *Dynamic modeling of musculoskeletal motion: A vectorized approach for biomechanical analysis in three dimensions*, 1st ed. Boston, MA, USA: Springer, 2006.
[303] A. N. Vasavada, S. Li, and S. L. Delp, "Three-dimensional isometric strength of neck muscles in humans," *Spine*, vol. 26, no. 17, pp. 1904–1909, 2001.
[304] R. T. Silva, G. C. Gracitelli, M. F. Saccol, C. Frota De Souza Laurino, A. C. Silva, and J. L. Braga-Silva, "Shoulder strength profile in elite junior tennis players: Horizontal adduction and abduction isokinetic evaluation," *British Journal of Sports Medicine*, vol. 40, no. 6, pp. 513–517, 2006.
[305] J. P. Weir, L. L. Wagner, T. J. Housh, and G. O. Johnson, "Horizontal abduction and adduction strength at the shoulder of high school wrestlers across age," *Journal of Orthopaedic and Sports Physical Therapy*, vol. 15, no. 4, pp. 183–186, 1992.
[306] L. A. Frey-Law, A. Laake, K. G. Avin, J. Heitsman, T. Marler, and K. Abdel-Malek, "Knee and Elbow 3D Strength Surfaces: Peak Torque-Angle-Velocity Relationships," *Journal of Applied Biomechanics*, vol. 28, no. 6, pp. 726–737, 2012.
[307] J. F. Kramer, D. Nusca, L. Bisbee, J. MacDermid, D. Kemp, and S. Boley, "Forearm pronation and supination: Reliability of absolute torques and nondorninant/dorninant ratios," *Journal of Hand Therapy*, vol. 7, no. 1, pp. 15–20, 1994.
[308] D. Haering, C. Pontonnier, N. Bideau, G. Nicolas, and G. Dumont, "Using torque-angle and torque-velocity models to characterize elbow mechanical function: Modeling and applied aspects," *Journal of Biomechanical Engineering*, vol. 141, no. 8, p. 084501, 2019.
[309] M. A. King, C. Wilson, and M. R. Yeadon, "Evaluation of a torque-driven model of jumping for height," *Journal of Applied Biomechanics*, vol. 22, no. 4, pp. 264–274, 2006.
[310] R. M. Alexander, "Optimum take-off techniques for high and long jumps." *Philosophical transactions of the Royal Society of London. Series B, Biological sciences*, vol. 329, no. 1252, pp. 3–10, 1990.

[311] G. A. Dudley, R. T. Harris, M. R. Duvoisin, B. M. Hather, and P. Buchanan, "Effect of voluntary vs. artificial activation on the relationship of muscle torque to speed," *Journal of Applied Physiology*, vol. 69, no. 6, pp. 2215–2221, 1990.
[312] A. J. van Soest and M. F. Bobbert, "The contribution of muscle properties in the control of explosive movements," *Biological Cybernetics*, vol. 69, no. 3, pp. 195–204, 1993.
[313] E. J. Sprigings, "Simulation of the force enhancement phenomenon in muscle," *Computers in Biology and Medicine*, vol. 16, no. 6, pp. 423–430, 1986.
[314] B. Katz, "The relation between force and speed in muscular contraction," *Journal of Physiology*, vol. 96, no. 1, pp. 45–64, 1939.
[315] S. J. MacKenzie and E. J. Sprigings, "A three-dimensional forward dynamics model of the golf swing," *Sports Engineering*, vol. 11, no. 4, pp. 165–175, 2009.
[316] F. Romero and F. J. Alonso, "A comparison among different Hill-type contraction dynamics formulations for muscle force estimation," *Mechanical Sciences*, vol. 7, no. 1, pp. 19–29, 2016.
[317] T. S. Buchanan, D. G. Lloyd, K. Manal, and T. F. Besier, "Neuromusculoskeletal modeling: Estimation of muscle forces and joint moments and movements from measurements of neural command," *Journal of Applied Biomechanics*, vol. 20, no. 4, pp. 367–395, 2004.
[318] A. J. Meyer, "Prediction of optimal rehabilitation outcomes post-stroke," Ph.D. dissertation, University of Florida, Gainesville, FL, USA, 2016.
[319] R. Rockenfeller, M. Günther, S. Schmitt, and T. Götz, "Comparative sensitivity analysis of muscle activation dynamics," *Computational and Mathematical Methods in Medicine*, vol. 2015, p. 1–16, 2015.
[320] D. G. Thelen, "Adjustment of muscle mechanics model parameters to simulate dynamic contractions in older adults," *Journal of Biomechanical Engineering*, vol. 125, no. 1, pp. 70–77, 2003.
[321] D. G. Thelen, A. Lenz, and A. Hernandez, "Measurement and simulation of joint motion induced via biarticular muscles during human walking," *Procedia IUTAM*, vol. 2, pp. 290–296, 2011.
[322] C. Brown and J. McPhee, "Predictive forward dynamic simulation of manual wheelchair propulsion on a rolling dynamometer," *Journal of Biomechanical Engineering*, vol. 142, no. 7, p. 071008, 2020.
[323] J. M. Banerjee and J. McPhee, "Symbolic sensitivity analysis of multibody systems," *Computational Methods in Applied Sciences*, vol. 28, pp. 123–146, 2013.
[324] H. Shah, S. Tripathi, L. F. Lee, and V. Krovi, "Role of automated symbolic generation of equations of motion in mechanism and robotics education," in *Proceedings of the ASME Design Engineering Technical Conference*, vol. 2. Montreal, Quebec, Canada: ASME, 2010, pp. 995–1002.
[325] D. G. E. Robertson, G. E. Caldwell, J. Hamill, G. Kamen, and S. N. Whittlesey, *Research Methods in Biomechanics*. Human Kinetics, Inc., 2014, vol. 73603966.
[326] B. Ghannadi, N. Mehrabi, R. S. Razavian, and J. McPhee, "Nonlinear model predictive control of an upper extremity rehabilitation robot using a two-dimensional human-robot interaction model," in *Proceedings of the IEEE International Conference on Intelligent Robots and Systems*, vol. 2017-Septe. Vancouver, BC, Canada: IEEE, 2017, pp. 502–507.
[327] M. B. Carpenter, "The co-ordination and regulation of movements," *Journal of Neuropathology and Experimental Neurology*, vol. 27, no. 2, p. 348, 1968.
[328] R. M. Alexander, "The gaits of bipedal and quadrupedal animals," *The International Journal of Robotics Research*, vol. 3, no. 2, pp. 49–59, 1984.
[329] T. Flash and N. Hogan, "The coordination of arm movements: An experimentally confirmed mathematical model," *Journal of Neuroscience*, vol. 5, no. 7, pp. 1688–1703, 1985.
[330] R. R. Neptune and M. L. Hull, "Evaluation of performance criteria for simulation of submaximal steady-state cycling using a forward dynamic model," *Journal of Biomechanical Engineering*, vol. 120, no. 3, pp. 334–341, 1998.
[331] M. Ackermann and A. J. van den Bogert, "Optimality principles for model-based prediction of human gait," *Journal of Biomechanics*, vol. 43, no. 6, pp. 1055–1060, 2010.
[332] R. S. Razavian and J. McPhee, "Minimization of muscle fatigue as the criterion to solve muscle forces-sharing problem," in *Proceedings of the ASME Dynamic Systems and Control Conference*, vol. 1. Columbus, Ohio, USA: ASME, 2015, p. V001T15A001.
[333] D. J. Reinkensmeyer, J. H. Wynne, and S. J. Harkema, "A robotic tool for studying locomotor adaptation and rehabilitation," in *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology*, vol. 3. Houston, TX, USA: IEEE, 2002, pp. 2353–2354.
[334] J. M. Winters and D. G. Kleweno, "Effect of initial upper-limb alignment on muscle contributions to isometric strength curves," *Journal of Biomechanics*, vol. 26, no. 2, pp. 143–153, 1993.
[335] A. A. Amis, D. Dowson, and V. Wright, "Analysis of elbow forces due to high-speed forearm movements," *Journal of Biomechanics*, vol. 13, no. 10, pp. 825–831, 1980.
[336] T. S. Buchanan, S. L. Delp, and J. A. Solbeck, "Muscular resistance to varus and valgus loads at the elbow," *Journal of Biomechanical Engineering*, vol. 120, no. 5, pp. 634–639, 1998.

[337] R. V. Gonzalez, "A computational musculoskeletal model of the human elbow and forearm in the analysis of ballistic movements," Ph.D. dissertation, The University of Texas at Austin, Austin, TX, USA, 1994.
[338] G. J. Alderink and D. J. K. Kuch, "Isokinetic shoulder strength of high school and college-aged pitchers," *Journal of Orthopaedic and Sports Physical Therapy*, vol. 7, no. 4, pp. 163–172, 1986.
[339] C. Brown, "Predictive forward dynamic simulation of manual wheelchair propulsion," Ph.D. dissertation, University of Waterloo, Waterloo, ON, Canada, 2018.
[340] Z. Dvir, "An isokinetic study of submaximal effort in elbow flexion," *Perceptual and Motor Skills*, vol. 84, no. 3 PART II, pp. 1431–1438, 1997.
[341] K. E. Timm, "The isokinetic torque curve of shoulder instability in high school baseball pitchers," *Journal of Orthopaedic and Sports Physical Therapy*, vol. 26, no. 3, pp. 150–154, 1997.
[342] J. G. Hincapie and R. F. Kirsch, "EMG-based control for a C5/C6 spinal cord injury upper extremity neuroprosthesis," in *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology*. Lyon, France: IEEE, 2007, pp. 2432–2435.
[343] E. Pina-Martinez, R. Roberts, S. Leal-Merlo, and E. Rodriguez-Leal, "Vision system-based design and assessment of a novel shoulder joint mechanism for an enhanced workspace upper limb exoskeleton," *Applied Bionics and Biomechanics*, vol. 2018, p. 1–14, 2018.
[344] M. G. Lewis, M. R. Yeadon, and M. A. King, "Are torque-driven simulation models of human movement limited by an assumption of monoarticularity?" *Applied Sciences*, vol. 11, no. 9, p. 3852, 2021.
[345] Y. Zhu, G. Zhang, C. Zhang, G. Liu, and J. Zhao, "Biomechanical modeling and load-carrying simulation of lower limb exoskeleton," *Bio-Medical Materials and Engineering*, vol. 26, no. s1, pp. S729–S738, 2015.
[346] T. M. Morrison, P. Pathmanathan, M. Adwan, and E. Margerrison, "Advancing regulatory science with computational modeling for medical devices at the FDA's office of science and engineering laboratories," *Medicine*, vol. 5, no. SEP, pp. 1–11, 2018.
[347] A. Nasr, K. A. Inkol, S. Bell, and J. McPhee, "InverseMuscleNET: Alternative machine learning solution to static optimization and inverse muscle modelling," *Frontiers in Computational Neuroscience*, vol. 15, p. 759489, 2021.
[348] R. L. Whittaker, N. J. La Delfa, and C. R. Dickerson, "Algorithmically detectable directional changes in upper extremity motion indicate substantial myoelectric shoulder muscle fatigue during a repetitive manual task," *Ergonomics*, vol. 62, no. 3, pp. 431–443, 2019.
[349] R. L. Whittaker, W. Park, and C. R. Dickerson, "Application of a symbolic motion structure representation algorithm to identify upper extremity kinematic changes during a repetitive task," *Journal of Biomechanics*, vol. 72, pp. 235–240, 2018.
[350] H. Hislop, D. Avers, and M. Brown, *Daniels and Worthingham's muscle testing: techniques of manual examination and performance testing*. St. Louis, MO, USA: Elsevier, 2013.
[351] S. Galle, P. Malcolm, W. Derave, and D. De Clercq, "Adaptation to walking with an exoskeleton that assists ankle extension," *Gait and Posture*, vol. 38, no. 3, pp. 495–499, 2013.
[352] J. J. Burkitt, V. Staite, A. Yeung, D. Elliott, and J. L. Lyons, "Effector mass and trajectory optimization in the online regulation of goal-directed movement," *Experimental Brain Research*, vol. 233, no. 4, pp. 1097–1107, 2015.
[353] J. D. Drake and J. P. Callaghan, "Elimination of electrocardiogram contamination from electromyogram signals: An evaluation of currently used removal techniques," *Journal of Electromyography and Kinesiology*, vol. 16, no. 2, pp. 175–187, 2006.
[354] H. J. Hermens, B. Freriks, C. Disselhorst-Klug, and G. Rau, "Development of recommendations for sEMG sensors and sensor placement procedures," *Journal of Electromyography and Kinesiology*, vol. 10, no. 5, pp. 361–374, 2000.
[355] M. B. I. Reaz, S. Hussain, and F. Mohd-Yasin, "Techniques of EMG signal analysis: Detection, processing, classification and applications," *Biological procedures online*, vol. 8, no. 1, pp. 11–35, 2006.
[356] A. Nasr, J. He, N. Jiang, and J. McPhee, "Muscle modelling using machine learning and optimal filtering of sEMG signals," in *Proceedings of the 45th Meeting of the American Society of Biomechanics*, Virtual, 2021, p. 83.
[357] R. D. Gurchiek, N. Cheney, and R. S. McGinnis, "Estimating biomechanical time-series with wearable sensors: A systematic review of machine learning techniques," *Sensors*, vol. 19, no. 23, p. 5227, 2019.
[358] M. H. Bollen and I. Y. H. Gu, *Signal Processing of Power Quality Disturbances*. Hoboken, NJ, USA: John Wiley & Sons, 2005.
[359] J. H. Lee, J. Shin, and M. J. Realff, "Machine learning: Overview of the recent progresses and implications for the process systems engineering field," *Computers and Chemical Engineering*, vol. 114, pp. 111–121, 2018.
[360] I. Williams and T. G. Constandinou, "Computationally efficient modeling of proprioceptive signals in the upper limb for prostheses: A simulation study," *Frontiers in Neuroscience*, vol. 8, p. 181, 2014.
[361] S. Arora and P. Doshi, "A survey of inverse reinforcement learning: Challenges, methods and progress," *Artificial Intelligence*, vol. 297, 2021.
[362] X. Ying, "An overview of overfitting and its solutions," *Journal of Physics: Conference Series*, vol. 1168, no. 2, p. 022022, 2019.
[363] R. Angold, J. Lubin, M. Solano, C. Paretich, and T. Mastaler, "Exoskeleton and method of providing an assistive torque to an arm of a wearer," p. 16, 2016.

[364] R. Gopura, K. Kiguchi, and D. S. Bandara, "A brief review on upper extremity robotic exoskeleton systems," in *Proceedings of the 6th International Conference on Industrial and Information Systems*. Kandy, Sri Lanka: IEEE, 2011, pp. 346–351.
[365] A. J. Veale and S. Q. Xie, "Towards compliant and wearable robotic orthoses: A review of current and emerging actuator technologies," *Medical Engineering and Physics*, vol. 38, no. 4, pp. 317–325, 2016.
[366] I. Hunter, J. Ballantyne, and J. M. Hollerbach, "A comparative analysis of actuator technologies for robotics," *Robotics Review*, vol. 2, pp. 299–342, 1991.
[367] P. L. García, S. Crispel, E. Saerens, T. Verstraten, and D. Lefeber, "Compact gearboxes for modern robotics: A review," *Frontiers in Robotics and AI*, vol. 7, p. 103, 2020.
[368] S. Seok, A. Wang, M. Y. Chuah, D. J. Hyun, J. Lee, D. M. Otten, J. H. Lang, and S. Kim, "Design principles for energy-efficient legged locomotion and implementation on the MIT Cheetah robot," *IEEE/ASME Transactions on Mechatronics*, vol. 20, no. 3, pp. 1117–1129, 2015.
[369] P. M. Wensing, A. Wang, S. Seok, D. M. Otten, J. H. Lang, and S. Kim, "Proprioceptive actuator design in the MIT cheetah: Impact mitigation and high-bandwidth physical interaction for dynamic legged robots," *IEEE Transactions on Robotics*, vol. 33, no. 3, pp. 509–522, 2017.
[370] A. Hughes and B. Drury, *Electric Motors and Drives: Fundamentals, Types and Applications*. Elsevier, 2019.
[371] K. A. Inkol and J. McPhee, "Towards compliant human-exoskeleton interactions within multibody dynamics simulations of assisted human motor control," in *Proceedings of the ECCOMAS Thematic Conference on Multibody Dynamics*, Budapest, Hungary, 2021.
[372] N. Jarrasse and G. Morel, "Connecting a human limb to an exoskeleton," *IEEE Transactions on Robotics*, vol. 28, no. 3, pp. 697–709, 2012.
[373] A. H. Stienen, E. E. Hekman, F. C. van der Helm, and H. van der Kooij, "Self-aligning exoskeleton axes through decoupling of joint rotations and translations," *IEEE Transactions on Robotics*, vol. 25, no. 3, pp. 628–633, 2009.
[374] S. Alabdulkarim and M. A. Nussbaum, "Influences of different exoskeleton designs and tool mass on physical demands and performance in a simulated overhead drilling task," *Applied Ergonomics*, vol. 74, pp. 55–66, 2019.
[375] L. Grazi, E. Trigili, G. Proface, F. Giovacchini, S. Crea, and N. Vitiello, "Design and experimental evaluation of a semi-passive upper-limb exoskeleton for workers with motorized tuning of assistance," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 28, no. 10, pp. 2276–2285, 2020.
[376] S. Kim and M. A. Nussbaum, "A follow-up study of the effects of an arm support exoskeleton on physical demands and task performance during simulated overhead work," *IISE Transactions on Occupational Ergonomics and Human Factors*, vol. 7, no. 3-4, pp. 163–174, 2019.
[377] S. Kim, M. A. Nussbaum, M. I. Mokhlespour Esfahani, M. M. Alemi, B. Jia, and E. Rashedi, "Assessing the influence of a passive, upper extremity exoskeletal vest for tasks requiring arm elevation: Part II – "Unexpected" effects on shoulder motion, balance, and spine loading," *Applied Ergonomics*, vol. 70, pp. 323–330, 2018.
[378] J. Theurel, K. Desbrosses, T. Roux, and A. Savescu, "Physiological consequences of using an upper limb exoskeleton during manual handling tasks," *Applied Ergonomics*, vol. 67, pp. 211–217, 2018.
[379] Z. H. Liu, Y. X. Qiu, Z. H. Zhu, Y. W. Zhou, and Z. Tang, "Effects of memory foam on optimizing shoulder fatigue of wearable exoskeleton," *Journal of Donghua University (English Edition)*, vol. 33, no. 4, pp. 536–539, 2016.
[380] S. Liu, D. Hemming, R. B. Luo, J. Reynolds, J. C. Delong, B. J. Sandler, G. R. Jacobsen, and S. Horgan, "Solving the surgeon ergonomic crisis with surgical exosuit," *Surgical Endoscopy*, vol. 32, no. 1, pp. 236–244, 2018.
[381] J. H. Kim and D. Roberts, "A joint-space numerical model of metabolic energy expenditure for human multibody dynamic system," *International Journal for Numerical Methods in Biomedical Engineering*, vol. 31, no. 9, p. e02721, 2015.
[382] A. Nasr, J. He, N. Jiang, and J. McPhee, "Activation torque estimation of muscles by forward neural networks (Forward-MuscleNET) for sEMG-based control of assistive robots," in *Proceedings of the 7th International Conference of Control, Dynamic Systems, and Robotics*. Virtual Conference: Avestia Publishing, 2020, p. 146.
[383] M. S. Shourijeh, R. S. Razavian, and J. McPhee, "Estimation of maximum finger tapping frequency using musculoskeletal dynamic simulations," *Journal of Computational and Nonlinear Dynamics*, vol. 12, no. 5, p. 051009, 2017.
[384] A. Fougner, O. Stavdahl, P. J. Kyberd, Y. G. Losier, and P. A. Parker, "Control of upper limb prostheses: Terminology and proportional myoelectric controla review," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 20, no. 5, pp. 663–677, 2012.
[385] E. Nsugbe, O. W. Samuel, M. G. Asogbon, and G. Li, "Phantom motion intent decoding for transhumeral prosthesis control with fused neuromuscular and brain wave signals," *IET Cyber-Systems and Robotics*, vol. 3, no. 1, pp. 77–88, 2021.
[386] L. M. Vaca Benitez, M. Tabie, N. Will, S. Schmidt, M. Jordan, and E. A. Kirchner, "Exoskeleton technology in rehabilitation: Towards an EMG-based orthosis system for upper limb neuromotor rehabilitation," *Journal of Robotics*, vol. 2013, p. 610589, 2013.

[387] A. Nasr, S. Ferguson, and J. McPhee, "Model-based design and optimization of passive shoulder exoskeletons," in *Proceedings of the ASME International Design Engineering Technical Conferences & Computers and Information in Engineering Conference*. Online, Virtual: ASME, 2021, pp. 2021–69 437.

[388] O. A. Kannape and H. M. Herr, "Volitional control of ankle plantar flexion in a powered transtibial prosthesis during stair-ambulation," in *Proceedings of the 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*. Chicago, IL, USA: IEEE, 2014, pp. 1662–1665.

[389] G. Gowrishankar, "Mechanisms of motor learning: by humans, for robots," Ph.D. dissertation, Imperial College London, London, United Kingdom, 2010.

[390] D. J. Reinkensmeyer, J. L. Emken, and S. C. Cramer, "Robotics, motor learning, and neurologic recovery," *Annual Review of Biomedical Engineering*, vol. 6, pp. 497–525, 2004.

[391] R. Shadmehr and F. A. Mussa-Ivaldi, "Adaptive representation of dynamics during learning of a motor task," *Journal of Neuroscience*, vol. 14, no. 5 II, pp. 3208–3224, 1994.

[392] B. Katz, J. D. Carlo, and S. Kim, "Mini cheetah: A platform for pushing the limits of dynamic quadruped control," *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 2019-May, pp. 6295–6301, 2019.

[393] Y. Jiang, Z. P. Jiang, and N. Qian, "Optimal control mechanisms in human arm reaching movements," in *Proceedings of the 30th Chinese Control Conference*. Yantai, China: IEEE, 2011, pp. 1377–1382.

[394] N. Lan and P. E. Crago, "Optimal control of antagonistic muscle stiffness during voluntary movements," *Biological Cybernetics*, vol. 71, no. 2, pp. 123–135, 1994.

[395] M. Sharifi, S. H. Pourtakdoust, and M. Parnianpour, "Optimal control of human-like musculoskeletal arm: Prediction of trajectory and muscle forces," *Optimal Control Applications and Methods*, vol. 38, no. 2, pp. 167–183, 2017.

[396] J. M. Davis and S. P. Bailey, "Possible mechanisms of central nervous system fatigue during exercise," *Medicine and Science in Sports and Exercise*, vol. 29, no. 1, pp. 45–57, 1997.

[397] P. Jalali, M. R. S. Noorani, R. Hassannejad, and M. M. Ettefagh, "Modeling the central nervous system functionality in controlling the calf muscle activity during running with sport shoes," *Proceedings of the Institution of Mechanical Engineers, Part H: Journal of Engineering in Medicine*, vol. 233, no. 2, pp. 254–266, 2019.

[398] W. Li, *Optimal control for biological movement systems*. San Diego, CA, USA: University of California, 2006, vol. 85.

[399] Y. Ueyama, "Effects of cost structure in optimal control on biological arm movement: A simulation study," in *Proceedings of the International Conference on Neural Information Processing*. Ueyama, Yuki: Springer, 2013, pp. 241–248.

[400] S. Huang, J. P. Wensman, and D. P. Ferris, "Locomotor adaptation by transtibial amputees walking with an experimental powered prosthesis under continuous myoelectric control," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 24, no. 5, pp. 573–581, 2016.

[401] ——, "An experimental powered lower limb prosthesis using proportional myoelectric control," *Journal of Medical Devices, Transactions of the ASME*, vol. 8, no. 2, p. 024501, 2014.

[402] B. Brahmi, M. Driscoll, I. K. El Bojairami, M. Saad, and A. Brahmi, "Novel adaptive impedance control for exoskeleton robot for rehabilitation using a nonlinear time-delay disturbance observer," *ISA Transactions*, vol. 108, pp. 381–392, 2021.

[403] B. He, H. Huang, G. C. Thomas, and L. Sentis, "A complex stiffness human impedance model with customizable exoskeleton control," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 28, no. 11, pp. 2468–2477, 2020.

[404] G. Menga and M. Ghirardi, "Control of the sit-to-stand transfer of a biped robotic device for postural rehabilitation," *Robotics*, vol. 8, no. 4, p. 91, 2019.

[405] K. I. Sherwani, N. Kumar, A. Chemori, M. Khan, and S. Mohammed, "RISE-based adaptive control for EICoSI exoskeleton to assist knee joint mobility," *Robotics and Autonomous Systems*, vol. 124, p. 103354, 2020.

[406] Z. Lu, X. Chen, X. Zhang, K. Y. Tong, and P. Zhou, "Real-time control of an exoskeleton hand robot with myoelectric pattern recognition," *International Journal of Neural Systems*, vol. 27, no. 5, p. 1750009, 2017.

[407] B. Laschowski, W. Mcnally, A. Wong, and J. Mcphee, "Environment classification for robotic leg prostheses and exoskeletons using deep convolutional neural networks," *Frontiers in Neurorobotics*, p. In press, 2021.

[408] D. Cafolla, "A 3D visual tracking method for rehabilitation path planning," in *New Trends in Medical and Service Robotics*, vol. 65. Cassino, Italy: Springer, 2019, pp. 264–272.

[409] B. D. M. Chaparro-Rico, D. Cafolla, E. Castillo-Castaneda, and M. Ceccarelli, "Design of arm exercises for rehabilitation assistance," *Journal of Engineering Research (Kuwait)*, vol. 8, no. 3, pp. 203–218, 2020.

[410] M. Ackermann and A. J. van den Bogert, "Optimality principles for model-based prediction of human gait," *Journal of Biomechanics*, vol. 43, no. 6, pp. 1055–1060, 2010.

[411] J. C. Gillette, S. Saadat, and T. Butler, "Electromyography-based fatigue assessment of an upper body exoskeleton during automotive assembly," *Wearable Technologies*, vol. 3, 2022.

[412] G. Goncu-Berk and B. G. Tuna, "The effect of sleeve pattern and fit on e-textile electromyography (EMG) electrode performance in smart clothing design," *Sensors*, vol. 21, no. 16, 2021.

[413] R. A. Soraa and E. Fosch-Villaronga, "Exoskeletons for all: The interplay between exoskeletons, inclusion, gender, and intersectionality," *Paladyn*, vol. 11, no. 1, pp. 217–227, 2020.

[414] S. Kim, A. Moore, D. Srinivasan, A. Akanmu, A. Barr, C. Harris-Adamson, D. M. Rempel, and M. A. Nussbaum, "Potential of Exoskeleton Technologies to Enhance Safety, Health, and Performance in Construction: Industry Perspectives and Future Research Directions," *IISE Transactions on Occupational Ergonomics and Human Factors*, vol. 7, no. 3-4, pp. 185–191, 2019.

[415] M. A. Ergin and V. Patoglu, "ASSISTON-SE: A self-aligning shoulder-elbow exoskeleton," in *Proceedings of the IEEE International Conference on Robotics and Automation*. Saint Paul, MN, USA: IEEE, 2012, pp. 2479–2485.

[416] J. Hunt, H. Lee, and P. Artemiadis, "A novel shoulder exoskeleton robot using parallel actuation and a passive slip interface," *Journal of Mechanisms and Robotics*, vol. 9, no. 1, 2017.

[417] M. R. Islam, C. Spiewak, M. H. Rahman, and R. Fareh, "A brief review on robotic exoskeletons for upper extremity rehabilitation to find the gap between research porotype and commercial type," *Advances in Robotics & Automation*, vol. 06, no. 03, p. 2, 2017.

Glossary

AAN assist-as-needed. vi, vii, xvii, xix, xx, xxiii, 4–6, 13–16, 67, 79, 108, 111–130, 137, 144, 148

ADEL anterior deltoid. xix, xx, 44, 51, 57, 105, 109, 127, 132, 134

ADLs activities of daily living. vi, 2, 150, 152

ANN artificial neural network. vi, xxii, 8, 9, 49, 81, 83, 84, 92, 95, 98, 102, 107, 148

BLDC brushless direct current. xiv, xix, xx, xxii, xxiii, 11, 65, 66, 75, 78, 108, 109, 116, 117, 127, 131, 132, 145, 148, 149

BRD brachioradialis. xix, xx, 109, 127, 132, 134, 142

BSIP body segment inertial parameters. 19

CAN-bus controller area network. xxiii, 108, 131, 133, 134

cHRI cognitive human-robot interaction. 14

CNN convolutional neural network. vi, xviii, 12, 81, 85, 86, 88, 92, 95, 96, 98–100, 102, 107, 148

CNS central nervous system. vii, xiv, xix, xx, 5, 6, 8, 10, 11, 13–16, 28, 32, 33, 58, 60, 69, 74, 110, 111, 117–121, 124–128, 130

COM center of mass. xxii, 7, 17–20, 41, 61, 66

CTM computed-torque method. vii, xvii, xix, xx, 5, 14–16, 67, 108, 112–114, 116, 118–130, 137, 144, 145, 148

CV cross-validation. xvi, 48, 51, 53

DAE differential-algebraic equation. 10, 68, 90, 108, 148

DoF degree of freedom. vi, xiv, xvi, xvii, xix, xxii, 2, 4, 7, 10, 12, 17, 21, 22, 24, 25, 27, 28, 36, 37, 39, 42, 58, 59, 68, 69, 105, 106, 108, 109, 114, 119, 129, 131, 147, 148, 150–152

e-textile electronic textile. 149

EEG electroencephalography. 6

EFE elbow flexion/extension. xiv, xvi, xvii, 21, 39, 44, 58, 59, 89, 90, 102, 142

EJC elbow joint center. 68

EMG electromyography. xv, 4, 6, 8, 9, 11, 13, 28, 35, 42, 105, 147, 149

FC fully connected. xviii, 85, 87, 88, 92, 95

FDO forward dynamic optimization. 34

FOC field-oriented control. 132, 134, 138

FPS forearm pronation/supination. xiv, xvi, xvii, 21, 25, 27, 39, 58, 59

FSO forward static optimization. 34

HITL human-in-the-loop. vi, 4, 13, 130, 131, 145, 148, 149, 151

HREI human-robot-environment interaction. xiv, 3, 14, 15

HRI human-robot interaction. 2–4, 129, 145, 151, 152

IM internal model. 15, 16, 33, 60, 110, 111, 119, 120, 125, 126, 129

IMU inertial measurement unit. xx, xxi, 6, 72, 111, 127–129, 132, 134, 135, 137, 145, 146, 149

INFR infraspinatus. 44, 51, 57, 105

ISB international society of biomechanics. xiv, xxii, 18, 19, 45

LATS latissimus dorsi. 44, 51, 105

LJC lumbar joint center. xvii, 59, 62, 68

LSTM long short-term memory. xviii, 12, 88, 89, 95

LTRA lower trapezius. 44, 51, 105

MDEL middle deltoid. xix, xx, 44, 51, 57, 105, 109, 127, 132, 134

MMEE muscles metabolic energy expenditure. 75, 140

MPC model predictive control. vii, xx, 5, 14–16, 108, 115, 116, 121, 123–130, 148

MSD musculoskeletal disease. 1, 75

MSE mean squared normalized error. xvi, xxi, xxii, 48, 54, 81, 85, 92, 94, 102, 103, 140

MSK musculoskeletal. vi, vii, xiv, xviii, 3, 4, 6, 7, 15–18, 21, 24, 29, 33, 39, 42, 58, 60, 100, 106, 108, 137, 139, 147, 151, 152

MTG muscle torque generator. vi, xiv, xv, xviii, 4, 7, 11, 17, 22–24, 28–30, 32–35, 37, 40, 42, 58, 75, 80, 90, 100, 103, 106, 108, 147, 148

MTRA middle trapezius. xix, xx, 44, 51, 105, 109, 127, 132, 134

NARX nonlinear autoregressive with external input neural. xvi, 46–49, 55, 83

NFE neck flexion/extension. xiv, 21, 58

NMPC nonlinear model predictive control. vii, xiv, xx, 5, 11, 16, 33, 34, 60, 110, 118, 120, 124–126, 128, 130, 148

NMSK neuromusculoskeletal. xviii, 100, 102, 105, 106, 140, 148

NRLAR neck right/left axial rotation. xiv, 21, 25, 27, 58

NRLLF neck right/left lateral flexion. xiv, 21, 25, 27, 58

ODE ordinary differential equation. 10, 24, 29, 30, 59

OOS out-of-sample. 48

PD proportional-derivative. 117, 138

PDEL posterior deltoid. xix, xx, 44, 51, 105, 109, 127, 132, 134

PECC pectoralis major. 44, 51, 57, 105

pHRI physical human-robot interaction. 14

PID proportional-integral-derivative. 14–16, 138

PWR power-to-weight ratio. 2, 3, 10, 151

RCNN recurrent convolutional neural network. vi, xviii, 12, 81, 85, 86, 88, 89, 95, 97–99, 102, 107, 148

RDNN recurrent deep neural network. 9

ReLU rectified linear unit. 86, 87

RMSE root mean squire of error. xxiii, 143

RNN recurrent neural network. vi, vii, xvi–xviii, xxii, 4, 5, 9, 12, 43, 45–51, 54–57, 81, 83–85, 90, 95–100, 102, 103, 105–107, 137, 148

ROM range of motion. vi, xvi, xvii, xxii, 2, 4, 17, 22, 25–27, 35–39, 41, 57, 62, 63, 150

SAA shoulder adduction/abduction. xiv, xv, xvii, 7, 13, 21, 29, 37, 58, 59, 62, 68, 76, 79, 80, 117

SDK software development toolkit. 133

sEMG surface electromyography. vi, vii, xiv–xxi, 4–7, 11–13, 16, 35, 43–57, 61, 67, 83–107, 109, 111, 116, 118, 123, 126–130, 132–135, 137, 139–142, 144–146, 148, 149

SERR serratus anterior. 44, 51, 57, 105

SFE shoulder flexion/extension. xiv, xv, xvii, 7, 13, 21, 28, 29, 32, 37, 38, 43–45, 57–59, 68, 76, 79, 80, 89–91, 102, 117

SHAA shoulder horizontal adduction/abduction. xv, 28, 37, 38

SJC shoulder joint center. xvii, 58, 59, 62, 68

SMLR shoulder medial/lateral rotation. xiv, xv, xvii, 13, 21, 25, 27, 38, 44, 58, 59, 68

SPI serial peripheral interface. 133

STD Standard deviation. xxi, 141, 142, 144

SUPR supraspinatus. 44, 51, 105

SVAA shoulder vertical adduction/abduction. xv, 25, 27, 28, 37, 38

TCP transmission control protocol. xxiii, 133, 134

TCU trigno control utility. 133

TFE torso flexion/extension. xiv, xvii, 21, 32, 58, 59, 76

TRLAR torso right/left axial rotation. xiv, xvii, 21, 58, 59, 76

TRLLF torso right/left lateral flexion. xiv, xvii, 21, 58, 59, 76

Tukey HSD Tukey honestly significant difference. xxi, xxiii, 143–145

TWR torque-weight ratio. 9, 78, 79

UTRA upper trapezius. xix, xx, 44, 51, 57, 105, 109, 126, 132, 134

WFE wrist flexion/extension. xiv, xvi, xvii, 21, 25, 39, 58, 59

WRUD wrist radial/ulnar deviation. xiv, xvi, xvii, 21, 25, 27, 39, 58, 59