Model-based Control of Upper Extremity Human-Robot Rehabilitation Systems

by

Borna Ghannadi

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, 2017

© Borna Ghannadi 2017

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. Mojtaba Ahmadi (Professor)
Supervisor	Dr. John McPhee (Professor)
Internal Member 1	Dr. Gordon Savage (Professor)
Internal Member 2	Dr. Nasser Lashgarian Azad (Associate Professor)
Internal/External Member	Dr. Dana Kulić (Associate Professor)

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 control structures and human-robot interaction models 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:

- Ghannadi, B., Sharif Razavian, R., and McPhee, J. (2017). Optimal impedance control of an upper extremity stroke rehabilitation manipulandum. Frontiers in Robotics and AI (Under Review).

  Contributions of R. Sharif Razavian: technical help with the experiments.

- Ghannadi, B., Sharif Razavian, R., and McPhee, J. (2017). A modified homotopy optimization for parameter identification in dynamic systems with backlash discontinuity. Nonlinear Dynamics (Under Review).

  Contributions of R. Sharif Razavian: technical help with the experiments.

- Ghannadi, B., Sharif Razavian, R., Ezati, M., and McPhee, J. (2017). Dynamic parameter identification of an upper extremity rehabilitation robot with friction using direct collocation method. IEEE/ASME Transactions on Mechatronics (Under Review).

  Contributions of R. Sharif Razavian: technical help with the experiments.
  Contributions of M. Ezati: technical help with the direct collocation method.

- Ghannadi, B., Sharif Razavian, R., and McPhee, J. (2017). Dynamic parameter identification of an upper extremity rehabilitation robot using GPOPS-II. In 8th ECCOMAS Thematic Conference on Multibody Dynamics. Prague, Czech Republic.

  Contributions of R. Sharif Razavian: technical help with the experiments.

- Ghannadi, B., Mehrabi, N., Razavian, R. S., and McPhee, J. (2017). Nonlinear model predictive control of an upper extremity rehabilitation robot using a two-dimensional human-robot interaction model. In 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Vancouver, British Columbia, Canada.

  Contributions of N. Mehrabi: technical help with the GPOPS-II software.
  Contributions of R. Sharif Razavian: technical help with the modes of simulations.

- Ghannadi, B., and McPhee, J. (2015). Optimal impedance control of an upper limb stroke rehabilitation robot. In ASME 2015 Dynamic Systems and Control Conference (p. V001T09A002). Columbus, Ohio, USA: ASME. http://doi.org/10.1115/DSCC2015-9689

- Ghannadi, B., Mehrabi, N., and McPhee, J. (2015). Development of a human-robot dynamic model to support model-based control design of an upper limb rehabilitation robot. In 7th ECCOMAS Thematic Conference on Multibody Dynamics. Barcelona, Spain.

  Contributions of N. Mehrabi: technical help with the $MA2DI_1$ musculoskeletal arm development.

Abstract

Stroke rehabilitation technologies have focused on reducing treatment cost while improving effectiveness. Rehabilitation robots are generally developed for home and clinical usage to: 1) deliver repetitive and stimulating practice to post-stroke patients, 2) minimize therapist interventions, and 3) increase the number of patients per therapist, thereby decreasing the associated cost. The control of rehabilitation robots is often limited to black- or gray-box approaches; thus, safety issues regarding the human-robot interaction are not easily considered. Furthermore, despite numerous studies of control strategies for rehabilitation, there are very few rehabilitation robots in which the tasks are implemented using optimal control theory. Optimal controllers using physics-based models have the potential to overcome these issues.

This thesis presents advanced impedance- and model-based controllers for an end-effector-based upper extremity stroke rehabilitation robot. The final goal is to implement a biomechanically-plausible *real-time* nonlinear model predictive control for the studied rehabilitation system. The real-time term indicates that the controller computations finish within the sampling frequency time. This control structure, along with advanced impedance-based controllers, can be applied to any human-environment interactions. This makes them promising tools for different types of assistive devices, exoskeletons, active prostheses and orthoses, and exercise equipment.

In this thesis, a high-fidelity *biomechatronic* model of the human-robot interaction is developed. The rehabilitation robot is a 2 Degree-of-Freedom (DOF) parallelogram linkage with joint friction and backlash, and nonlinear dynamics. The *mechatronic* model of the robot with relatively accurate identified dynamic parameters is used in the human-robot interaction plant. Different musculoskeletal upper extremity, *biomechanic*, models are used to model human body motions while interacting with the rehabilitation robot model. Human-robot interaction models are recruited for model-in-loop simulations, thereby tuning the developed controllers in a structured resolution. The interaction models are optimized for real-time simulations. Thus, they are also used within the model-based control structures to provide biofeedback during a rehabilitation therapy.

In robotic rehabilitation, because of physical interaction of the patient with a mechanical device, safety is a fundamental element in the design of a controller. Thus, impedance-based assistance is commonly used for robotic rehabilitation. One of our objectives is to achieve a *reliable* and real-time implementable controller. In our definition, a reliable controller is capable of handling variable exercises and admittance interactions. The controller should reduce therapist intervention and improve the quality of the rehabilitation.

Hence, we develop advanced impedance-based assistance controllers for the rehabilitation robot. Overall, two types of impedance-based (i.e., hybrid force-impedance and optimal impedance) controllers are developed and tuned using model-in-loop simulations. Their performances are assessed using simulations and/or experiments. Furthermore, their drawbacks are discussed and possible methods for their improvements are proposed.

In contrast to black/gray-box controllers, a physics-based model can leverage the inherent dynamics of the system and facilitate implementation of special control techniques, which can optimize a specific performance criterion while meeting stringent system constraints. Thus, we present model-based controllers for the upper extremity rehabilitation robot using our developed musculoskeletal models. Two types of model-based controllers (i.e., nonlinear model predictive control using external 3-Dimensional (3D) musculoskeletal model or internal 2-Dimensional (2D) musculoskeletal model) are proposed. Their performances are evaluated in simulations and/or experiments. The biomechanically-plausible nonlinear model predictive control using internal 2D musculoskeletal model predicts muscular activities of the human subject and provides optimal assistance in real-time experiments, thereby conforming to our final goal for this project.

Acknowledgments

I would like to express my sincere gratitudes to my benevolent supervisor, Professor John McPhee, who graciously helped me to carry out this project.

I am also thankful of my colleagues (Dr. Reza Sharif Razavian, Dr. Naser Mehrabi, Mr. Sadegh Tajeddin, Mr. Brock Laschowski, Ms. Mahdokht Ezati) for their help in sharing the information about different aspects of this project.

I must express my heartfelt gratitude to Mahdokht, my fiance, for her continued support and encouragement.

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.

I would also wish to thank Quanser Consulting Inc. for providing the upper extremity rehabilitation robot, and Toronto Rehabilitation Institute for collaborating.

Dedication

I dedicate my dissertation work to my encouraging fiance and my sympathetic family. A special feeling of gratitude to my parents and siblings for their lifelong support.

Table of Contents

List of Tables

List of Figures

Nomenclature

α^{M}	Muscle pennation angle
α_0^{M}	Muscle pennation angle at optimal length
α_{AC}	Scapula anterior/posterior tilt about Z axis
α_{HU}	Humeroulnar joint flexion/extension about Z axis
$\bar{\bullet}$	Augmented or updated form of $\bullet$
$\bar{\eta}_{\mathrm{H}}^{\iota}$	Amplitude of the ι^{th} high frequency sinusoidal signal
$\bar{\eta}_{\mathrm{L}}^{\iota}$	Amplitude of the ι^{th} low frequency sinusoidal signal
$\bar{\zeta}_j$	Amplitude of the j^{th} sinusoidal signal
β	Backlash gap size
β_0	Positive constant gain
β_1	Positive constant gain
β_{AC}	Scapula lateral/medial rotation about X axis
β_{GH}	Glenohumeral negative elevation about X axis
β_{SC}	Clavicle elevation/depression about X axis
$\boldsymbol{\chi}$	Vector of unknown kinematic parameters
$\boldsymbol{\Gamma}$	System dynamics function with no input and interaction force
$\boldsymbol{\mathit{y}}$	Outer loop control law

$\mathfrak{T}$	Transformation matrix
$\mathfrak{z}$	Vectorized translational bristle deflection
$\mathbf{\Phi}$	Kinematic function that relates the joint angles to the Cartesian coordinates
$\mathbf{\Pi}$	2D human-robot dynamics function
$\mathbf{\Psi}$	Inverse of the kinematic function that relates the Cartesian coordinates to the joint angles
$\boldsymbol{\rho}$	End-effector position in the global coordinates
$\boldsymbol{\theta}$	Vector of the MA2D joint angles ($= [\theta_S, \theta_E]^T$)
$\boldsymbol{\varepsilon}$	Relaxation parameter vector
$\boldsymbol{\varpi}$	Skew symmetric matrix of angular velocity
$\bullet^{\delta}$	δ seconds delayed $\bullet$
$\bullet^{①}$	of or related to the setup ①
$\bullet^{②}$	of or related to the setup ②
$\bullet^{③}$	of or related to the setup ③
$\bullet^{E}$	of or related to the elbow flexion-extension joint on the horizontal plane
$\bullet^{i}$	This superscript denotes the i$^{\text{th}}$ iteration
$\bullet^{S}$	of or related to the shoulder flexion-extension joint on the horizontal plane
$\bullet^{T}$	Transpose of a non-scalar variable
$\bullet^{\cap}$	Upper envelope of $\bullet$
$\bullet^{\iota}$	ι^{th} experiment
$\bullet^{\Sigma}$	Defined in the end-effector's n-t coordinates (i.e., the operational space)
$\bullet_{\perp}$	Normal to the path
$\bullet_{\mathbf{q}}$	of or related to the joint space

$\bullet_{\mathrm{II}}$	in the MA3D-II coordinate frame
$\bullet_{\mathrm{I}}$	in the MA3D-I coordinate frame
$\bullet_{\mu}$	Mean of $\bullet$ bounds ($= \frac{\bullet_{max}+\bullet_{min}}{2}$)
$\bullet_{\sigma}$	Standard deviation of $\bullet$ bounds ($= \frac{\bullet_{max}-\bullet_{min}}{2}$)
$\bullet_{A}$	of or related to MA2D-II
$\bullet_{c}$	of or related to the clavicle
$\bullet_{d}$	This subscript denotes the desired data
$\bullet_{E}$	of or related to the elbow coordinate system
$\bullet_{e}$	This subscript denotes the experimental data
$\bullet_{h}$	of or related to the humerus
$\bullet_{L}$	This subscript denotes the load-side values
$\bullet_{M}$	This subscript denotes the motor-side values
$\bullet_{O}$	of or related to the origin coordinate
$\bullet_{o}$	Optimized value of the $\bullet$
$\bullet_{R}$	of or related to the upper extremity rehabilitation robot
$\bullet_{r}$	of or related to the reference model
$\bullet_{S}$	of or related to the shoulder coordinate system
$\bullet_{s}$	of or related to the scapula
$\bullet_{t}$	of or related to the thorax
$\bullet_{\boldsymbol{\rho}}$	of or related to the Cartesian space
$\bullet_{\Sigma}$	of or related to the end-effector's n-t coordinates (i.e., the operational space, see Fig. 5.3)
$\bullet_{hyb}$	of or related to the hybrid force-impedance control

$\bullet_{imp}$ of or related to the impedance model

$\bullet_{max}$ This subscript denotes the upper bound of a variable or parameter

$\bullet_{min}$ This subscript denotes the lower bound of a variable or parameter

$\bullet_{opt}$ of or related to optimal

$\bullet_{tol}$ Allowable tolerance

Δ_X Path length along X axis

Δ_Z Path length along Z axis

ϵ Floating-point relative accuracy in MATLAB®

η Transformed joint coordinate ($= q_2 - q_1$)

γ Positive constant gain

γ_{AC} Scapula retraction/protraction about Y axis

γ_{GH1} Glenohumeral plane of elevation about Y axis

γ_{GH2} Glenohumeral-axial rotation about Y axis

γ_{RU} Radioulnar pronation/supination about Y axis

γ_{SC} Clavicle retraction/protraction about Y axis

$\hat{\bullet}$ This accent denotes the normalized value of the variable

$\hat{F}_d$ Desired force magnitude

Λ Observer gain adjuster function

λ Observer gain adjuster

$\mathbf{R}^O_{\mathscr{j}}\big\rfloor_{(i,j)}$ $(i,j)^{\mathrm{th}}$ element of the rotation matrix converting the joint $\mathscr{j}$ local frame to the origin frame

$\mathbb{Q}$ Element of $\mathbf{Q}$

$\mathbb{R}$ Element of $\mathbf{R}$

$\$_i$	i^{th} singular value
$\mathbf{a}_\bullet$	Axis of the $\bullet$
$\mathbf{B}$	Damping matrix
$\mathbf{C}$	Coriolis-centripetal matrix
$\mathbf{d}$	Dynamic equation noise
$\mathbf{e}_{\text{EMG}}$	Processed and normalized EMG signal vector
$\mathbf{f}_F$	End-effector friction force vector
$\mathbf{f}_T$	Friction torque vector
$\mathbf{F}_{ext}$	Robot to human interaction force which is measured experimentally
$\mathbf{F}_{int}$	Robot to human interaction force which is measured in the HRI model
$\mathbf{G}$	Gradient vector
$\mathbf{H}$	Hessian matrix
$\mathbf{I}^c$	Motors' input current vector
$\mathbf{J}$	Geometric Jacobian
$\mathbf{K}$	Stiffness matrix
$\mathbf{M}$	Mass matrix
$\mathbf{p}$	Unknown dynamic parameter vector
$\mathbf{p}^{②\text{-}①}$	Parameters of the setup ② which are different from the parameters of the setup ①
$\mathbf{p}^{③\text{-}②}$	Parameters of the setup ③ which are different from the parameters of the setup ②
$\mathbf{q}$	Angular displacement vector of robot driving joints
$\mathbf{q}$	Joint angle vector
$\mathbf{q}_0$	Equilibrium angle vector

$\mathbf{R}_\bullet$ Rotation matrix transforming the $\bullet$ coordinates to the global coordinates (i.e., the Cartesian space)

$\mathbf{r}_i$ Center of mass of the i^{th} link in its local coordinate

$\mathbf{r}_\bullet$ Position vector of the $\bullet$

$\mathbf{s}$ Search direction

$\mathbf{U}$ Triangulated Hessian matrix

$\mathbf{u}$ Control input

$\mathbf{v}$ End-effector velocity in Cartesian coordinates

$\mathbf{x}$ System state vector

$\mathbf{x}_1$ Vector of independent coordinates (displacements)

$\mathscr{F}$ Look-up table function

$\mathscr{f}$ Function of muscle activations which is used for updating the tracking gain

$\mathscr{j}$ Corresponding revolute joint number

$\mathscr{K}_i$ i^{th} constant coupling gain

$\mathscr{M}$ Number of muscle group

$\mathscr{m}$ Number of muscle

$\mathscr{N}$ Number of experiments

$n_{\mathscr{j}}$ Number of active joints of the musculoskeletal model

$n_{\mathscr{M}}$ Number of muscle groups

$n_{\mathscr{m}}$ Number of muscles

n_{E} Number of equilibrium angle parameters

n_{I} Number of inertial parameters

n_{S} Number of stiffness parameters

$\mathscr{n}_{\mathrm{F(A1)}}$ Number of unknown friction parameters for the first approach (A1)

$\mathscr{n}_{\mathrm{F(A2)}}$ Number of unknown friction parameters for the second approach (A2)

$\mathscr{n}_{\mathrm{p}}$ Number of unknown dynamic parameters

$\mathscr{p}$ Muscle mechanics parameter

$\mathscr{r}$ Muscle moment arm

$\mathscr{r}_{\mathscr{m}}^{\mathscr{j}}$ $\mathscr{m}^{\mathrm{th}}$ muscle moment arm about joint $\mathscr{j}$ that the muscle acts on

$\mathscr{z}$ Outer loop control law

$\mathfrak{c}$ Path (continuous) constraint

$\mathfrak{H}_{\mathrm{H}}^{\iota}$ Frequency of the ι^{th} high frequency sinusoidal signal

$\mathfrak{H}_{\mathrm{L}}^{\iota}$ Frequency of the ι^{th} low frequency sinusoidal signal

$\mathfrak{J}$ Objective function

$\mathfrak{V}$ Homotopy objective function

$\mathfrak{V}^{k}$ k^{th} homotopy objective function

$\mathfrak{Z}_{j}$ Frequency of the j^{th} sinusoidal signal

c_{12} $\cos(q_1 - q_2)$

s_{12} $\sin(q_1 - q_2)$

$\mu_{\textcircled{S}}$ Stiction with no Coulomb friction

μ_c Coulomb friction

μ_d Dynamic friction coefficient

μ_s Static friction coefficient

μ_v Viscous friction coefficient

ν Sampling-time frequency

Ω Input frequency

ω_0 Angular speed that shapes the function $g(\omega)$

ω_d Dissipative angular speed

$\omega_{1,2}$ Angular speeds of joints 1 and 2 ($\dot{q}_{1,2}$)

σ_0 Bristle stiffness coefficient

σ_1 Bristle damping coefficient

τ Normalized time ($= t/t_f$)

θ Angular displacement

θ_E Elbow angle measured on the transverse plane with respect to the humerus axis

θ_S Shoulder angle measured on the transverse plane with respect to the frontal plane

Υ Summation of the dynamic terms excluding the inertial terms

$\mathbb{A}$ State matrix

$\mathbb{B}$ Input matrix

$\mathbb{F}$ Applied force in the global coordinates (i.e., the Cartesian space)

$\mathbb{K}_D$ Derivative gain

$\mathbb{K}_I$ Integral gain

$\mathbb{K}_P$ Proportional gain

$\mathbb{Q}$ Matrix gain for the state

$\mathbb{R}$ Matrix gain for the effort

$\mathbb{S}$ Non-negative diagonal matrix of the singular values

$\mathbb{U}$ Unitary matrix

$\mathbb{U}_i$ i$^{\text{th}}$ column of the unitary matrix $\mathbb{U}$

$\mathbb{V}$ Unitary matrix in the joint space

ϱ Updating step size

$\widetilde{\bullet}$ ($= \bullet_d - \bullet_e$ or $\mathring{\bullet} - \bullet$) Error of the desired ($\bullet_d$) or estimated ($\mathring{\bullet}$) variable with respect to its experimental ($\bullet_e$) or actual value ($\bullet$)

ξ Average rotational bristle deflection

ζ Transformed joint coordinate ($= q_2 + q_1$)

A Amplitude

a Muscle activation

B Damping coefficient

c_i i^{th} coefficient

c_l Backlash left crossing

c_r Backlash right crossing

d_w Diameter of the added weight

$d_{N(\mu,\sigma)}$ Gaussian random noise with mean μ and variance σ^2

F Force

f Friction torque

F^{iso} Maximum isometric muscle force

F^{M} Muscle force

$F^{\mathrm{CE}}(.)$ Force function of the muscle's contractile element

$F^{\mathrm{PE}}(.)$ Force function of the muscle's passive element

f_1 First joint friction torque

f_2 Second joint friction torque

g Shape function that models the Stribeck effect

h System dynamic function

h_w Height of the added weight

I^c	Motor current input
J	Inertia
J_i	Inertia of the i$^{\text{th}}$ link in robot's parallelogram about link's center of mass
J_w	Inertia of the added weight about its center of mass
K	Stiffness coefficient
$k_\bullet$	Weighting constants
K_T	Motor torque constant
L	Muscle length
L_0^{M}	Muscle optimal fiber length
L_{s}^{T}	Tendon slack length
l_h	Upper arm length
l_i	i$^{\text{th}}$ link length
l_u	Forearm length
M	Mass
m_i	Mass of the i$^{\text{th}}$ link in robot's parallelogram
m_w	Mass of the added weight
N	Weighting factor
n	Number of independent coordinates (displacements)
p_i	i$^{\text{th}}$ parameter
q_i	i$^{\text{th}}$ robot joint angle
r	Mechanical transformer gear ratio
T	Torque
t	Time

t_f	Final time
t_{ch}	Control horizon
t_{ph}	Prediction horizon
u_{step}	Unit step control input
V	Muscle velocity
v	End-effector speed
v_0	Linear velocity that shapes the function $g(v)$
v_d	Dissipative linear velocity
v_t	Linear transition velocity for the stiction
w_k	k^{th} weight
X	Position in X direction in Cartesian coordinates
x_i	Center of mass of the i^{th} link in its local coordinate along X_i axis
x_w	Position of the added weight in the local coordinate of the 4^{th} link along X_4
y_i	i^{th} state variable
Z	Position in Z direction in Cartesian coordinates
z_i	Center of mass of the i^{th} link in its local coordinate along Z_i axis

Acronyms

1D 1-Dimensional 65, 132, 134, 138, 140, 160, 161

2D 2-Dimensional vi, 3–5, 16, 19, 20, 65, 68, 69, 71–75, 83, 84, 86, 88, 89, 95, 96, 103, 131, 138, 140, 153, 158, 160–162

3D 3-Dimensional vi, 3, 4, 16, 18–20, 69, 71, 72, 74, 77, 78, 86, 96, 103, 142, 153, 158, 160–163

ACJ Acromioclavicular Joint 13, 15, 18, 74, 79

AI Artificial Intelligence 25, 148

ARE Algebraic Riccati Equation 134, 136, 138

BCI Brain-computer Interface 18, 25

CE Contractile Element 7, 18, 19, 71, 73, 78, 112

CMC Computed Muscle Control 12, 13

CNS Central Nervous System 6, 142, 148

COM Center of Mass 72, 84

CP Cerebral Palsy 20

DAE Differential-Algebraic Equation 29, 37–39, 41–43

DAQ Data Acquisition 33

DO Dynamic Optimization 10, 11

DOF Degree-of-Freedom v, 3, 4, 23, 31, 34, 36, 39, 49, 57, 65, 71, 74, 88, 89, 147, 149, 152, 160

DPI Dynamic Parameter Identification 3–5, 28–31, 34, 40–44, 48–52, 54–58, 61, 62, 64, 65, 68, 96–98, 100, 103, 158–161, 163

DRUJ Distal Radioulnar Joint 15, 18, 79

EEG Electroencephalogram 22, 25

EID Extended Inverse Dynamics 10, 11

EMG Electromyography 7, 17, 22, 24, 25, 97–100, 102, 148, 149, 158, 163

F/T Force/Torque 32, 33

FES Functional Electrical Stimulation 7, 22, 25

FSO Forward Static Optimization 11–13, 18, 69, 78, 83, 112, 142–145, 147, 149, 158, 161

GHJ Glenohumeral Joint 13, 18, 19, 74, 79, 83

GMRES Generalized Minimal Residual 154, 157

HFIC Hybrid Force-Impedance Control 106, 108, 113, 114, 123, 126, 127, 129, 132, 139, 160, 161

HRI Human-Robot Interaction 22, 26, 27, 30, 68, 85, 99, 104, 111, 132, 141, 142, 148, 152, 153, 157–159, 161

HRJ Humeroradial Joint 15

HSR Humeroscapular Rhythm 15

HTR Humerothorax Rhythm 15, 20

HUJ Humeroulnar Joint 15, 19, 79

IFDO Inverse-forward Dynamic Optimization 11

IFDOC Inverse-forward Dynamic Optimization Control 11

IIDO Inverse-inverse Dynamic Optimization 11, 19

IO Input-Output 7, 48, 50, 51, 53, 65

IPOPT Interior Point Optimizer 57, 61, 143

ISB International Society of Biomechanics 4, 19, 20, 78, 79, 81, 83, 162

LQR Linear Quadratic Regulator 117, 118, 120, 128–130

LTI Linear Time-Invariant 116, 118

LTV Linear Time-Variant 116–118, 122

MoRG Motion Research Group 12, 18

MSO Modified Static Optimization 11–13, 71, 73

MVC Maximum Voluntary Contraction 97

NLP Nonlinear Programming 57, 61, 143, 154

NMPC Nonlinear Model Predictive Control 3, 5, 12, 27, 28, 52, 68, 96, 142, 143, 145–149, 152–155, 157–159, 161–163

ODE Ordinary Differential Equation 7, 29, 35, 56, 112, 135

OIC Optimal Impedance Control 116, 123, 124, 126, 128–130, 139, 140, 161

PE Parallel-elastic Element 7, 18, 19, 71, 78, 81, 155, 158, 162

PI Proportional-Integral 108

PID Proportional-Integral-Derivative 55

PRUJ Proximal Radioulnar Joint 15

RC Rounded Corner 61, 64

RCJ Radiocarpal Joint 15, 18, 79

RMSE Root-Mean Square Error 44, 46, 47, 101, 102, 152, 155

ROM Range of Motion 3, 16, 19, 20, 27, 74

RPM Radau Pseudospectral Method 60

SAS Subacromial Space 13

SC Sharp Corner 61

SCI Spinal Cord Injury 1, 20

SCJ Sternoclavicular Joint 13, 15, 18, 74, 79

SE Series-elastic Element 7, 18, 19, 71, 78

SNOPT Sparse Nonlinear Optimizer 57

SNR Signal to Noise Ratio 61, 64, 102

SO Static Optimization 10–12, 71, 73, 78, 83, 91–93, 95

SOIC Standard Optimal Impedance Control 115, 116, 123, 124, 126–130, 132, 140, 161

SQP Sequential Quadratic Programming 29, 42, 44, 45, 48

STJ Scapulathoracic Joint 13, 18

SVD Singular Value Decomposition 130, 131

TBI Traumatic Brain Injury 1, 20

TRI Toronto Rehabilitation Institute 30, 31, 159

Chapter 1

Introduction

There are a large number of people with movement disabilities who have difficulties doing their daily tasks independently. These movement defects are caused by many issues such as injuries and surgeries, stroke, Traumatic Brain Injury (TBI), Spinal Cord Injury (SCI), and various neurological disorders [170]. To give the hope of independent life to these people, many researchers are working on rehabilitation procedures for the therapy. In Canada, 40,000 people are suffering from SCI, with an annual rate of 1500 new cases [131], and 50,000 people suffer from TBI [310]. Along with various statistical reports on brain injuries, there is a significant motivation to seek rehabilitation of these patients (specifically stroke survivors) because of the following reasons. Each year in Canada, 62,000 people experience a stroke; among these patients, over 6,500 cases access the in-patient care which costs the Canadian health care system about $3.6 billion per year [108]. Although it has been found that therapy is effective in the treatment of movement disorders, because of economic burdens therapy hours per patient have decreased [232]. Studies have shown that comprehensive and optimal stroke care can decrease the associated costs significantly, and this will save Canadian health care $700 million per year [21, 153]. This optimal care can be achieved by implementing new technologies. That is why the design and development of automated rehabilitation devices have gained more importance.

1.1 Motivations and objective

Rehabilitation robots are currently being used in clinical settings under different case studies. However, their effectiveness is arguably indeterminate [29, 164]. Even the latest rehabilitation robots are generally limited by:

1. out-of-date motor learning methods used in rehabilitation robotics [179],
2. few identified motor learning approaches [178],
3. incomplete clinical assessments for evaluating effectiveness of rehabilitation robots in patient's motor function recovery [178], and
4. dependency of positive results of physical therapy on various factors related to the patient and source of impairment [170].

Despite these shortcomings, it is not possible to ignore the potential of rehabilitation robots since physical therapy, which can easily be delivered by these robots, is the key element for motor recovery [149, 230]. Thus, the advancement of the design and control of a rehabilitation robot is very valuable because of the following reasons:

1. insufficiency of traditional physical therapies [128, 137],
2. reduction of financial burden of costly therapy sessions [71, 150],
3. innovative motor learning scenarios [29, 38] supported by the optimization of robot's mechatronic design and control algorithms.
4. in-home rehabilitation and patient care [71, 217],
5. variations in delivering therapy, which can facilitate new achievements in motor learning studies [29, 230],
6. meaningful restoration of functional activities by introducing a new class of interactive and user-friendly robots [151].

Best design practices demand proper modeling of the whole system, which for this case consists of a human body interacting with a rehabilitation robot. However, there is a lack of studies considering human body interaction with the rehabilitation robot. Hence, since this interaction will affect rehabilitation procedures, the goal of this research is to fill this gap. For modeling human body motion and interaction, musculoskeletal models are used. Upper extremity motor defects are common among stroke patients [184], so this project will be focused on the modeling and control of an upper extremity rehabilitation robot interacting with musculoskeletal upper extremity models.

1.2 Challenges

The goal of this research is to develop model-based controllers for an upper extremity rehabilitation robot. Fulfillment of this goal involves two steps of modeling and control of a human-robot rehabilitation system, which leads to the following challenges:

1. Develop suitable 2D and 3D musculoskeletal models to interact with the robot. Musculoskeletal models consist of muscles, bony segments, and joints. There are various musculoskeletal arm models; a proper selection among them requires the careful study of each part. For muscles, their mechanics, path models, and muscle redundancy solution method must be considered. For bony segments and joints, their definitions such as Range of Motion (ROM) and DOF should be clarified.

2. Model the studied upper extremity rehabilitation robot. The robot has friction and backlash. There are different friction and backlash models; selecting appropriate models for among them are challenging parts of this model development.

3. Model the human-robot system as a whole, which involves an efficient integration of the musculoskeletal and robot models based on the human's and robot's workspace envelopes for the planned rehabilitation activities.

4. Identify the dynamic parameters of the developed musculoskeletal and robot models. Because of the highly nonlinear structure of both models, their Dynamic Parameter Identification (DPI) is one of the cumbersome parts of this project. Developing a method to identify unknown dynamic parameters of the robot which has backlash discontinuity is a particularly challenging part.

5. Design optimal and hybrid impedance-based control structures for the robot and tune them using model-in-loop simulations. Impedance and admittance control are widely used in rehabilitation robotics. Development of optimal and hybrid controllers of these types and using a musculoskeletal model to simulate and evaluate the robot's performance will help us to not only tune the proposed controllers but also assess the physiological properties of the system.

6. Design Nonlinear Model Predictive Control (NMPC) framework for the robot using human-robot interaction dynamics. Development of this framework is highly dependent on the accurate and efficient modeling of human-robot interaction system. This also involves selection and tailoring of suitable NMPC methods.

7. Implement the proposed controllers on the robot with real-time processing. Tailoring the proposed controllers for real-time applications is the last and most important challenge of this project.

1.3 Contributions and applications of the research

This research involves modeling and DPI of mechatronic and bio-mechanical systems (i.e., *biomechatronics*). The contributions of this work are listed below (underlined), followed by a variety of applications of the research:

- DPI of a rehabilitation manipulandum with a nonlinear and discontinuous dynamic model: the robot's friction model and DPI can be used in other robotic applications. The proposed methodology for DPI of the robot with backlash can be extended to include other mechanical discontinuities, such as dead-zone and hysteresis.

- Development of a 3D musculoskeletal upper extremity model with muscle wrapping according to International Society of Biomechanics (ISB) recommendation: the 3D musculoskeletal model can be used in upper extremity studies as a reference model. This model is derived from the Texas upper extremity model [96, 98, 99], which is modified by defining it in ISB recommended coordinates [322] and including shoulder rhythm movements [326]. The model's optimized C-Code is generated using the MapleSim^{TM}software package, which is advantageous for many real-time applications.

- Building low-fidelity (2D) musculoskeletal models based on the 3D musculoskeletal model: mapping techniques to construct a 2D musculoskeletal arm model from the 3D model for interaction with a planar rehabilitation robot can be used in a variety of human-interfaced robotic applications with 2 DOF. Our developed 2D model behaves similarly to the actual 3D musculoskeletal arm model in the horizontal plane reaching movements. The implemented shoulder rhythm in the 2D musculoskeletal arm model can enhance any 2D upper extremity musculoskeletal model.

- Dynamic parameter identification of the musculoskeletal upper extremity model: the DPI method for the musculoskeletal upper extremity model can be extended to other musculoskeletal models and applications.

- Building a human-robot interaction model: the integration of two systems can be used to model human-machine interactions, and this has a variety of applications ranging from work- to home-space human-environment interaction.

Several optimal controllers are developed for the human-robot interaction system, which can result in the following advantages:

- Design and model-based tuning of hybrid force-impedance and optimal impedance controllers for rehabilitation: the proposed optimal/hybrid impedance-based controllers can be used in controlling robot manipulators in contact with objects in their workspace. The robotic contact problem is very important in many applications involving motion that is constrained by the environment. This gains more importance, once the interaction environment is human (i.e., rehabilitation robotics). In rehabilitation robotics, impedance-based controllers are widely used. However, design of optimal impedance-based controller and tuning with the aid of model-in-loop simulations are pretty novel and has not been done previously. These novel methodologies will enhance the rehabilitation quality.

- Development of real-time human-robot interaction controllers in NMPC framework: End-effector based rehabilitation robots as well as exoskeleton robots can benefit from the developed NMPC framework to optimize rehabilitation. This framework can be implemented not only in other rehabilitation robot applications but also in prosthesis design. Prosthetic limbs are in contact with different environments while interacting with the human body. These robotic limbs are controlled using myoelectric signals. Implementing the proposed control scenarios may optimize the performance of these robots to consume less energy while performing the desired task.

1.4 Thesis outline

In the next chapter (Background), a literature review on the musculoskeletal upper extremity modeling, upper extremity movement disorders, upper extremity rehabilitation devices and DPI are provided. In Chapter 3 (System Models: Rehabilitation Robot), the model of the studied robot is developed, and two DPI methods are proposed, and their significance is discussed. In Chapter 4 (System Models: Human and Human-Robot Interaction), different musculoskeletal upper extremity models are developed. Then, the robot and human models are integrated, and their integration is discussed. Some preliminary cross-validation studies are done on the musculoskeletal models. Finally, a method for the DPI of the 2D musculoskeletal model is presented. In Chapter 5 (Rehabilitation Robot Control: Robot Control), two low-level controllers based on the impedance-based assistance are developed and tuned. The proposed optimal impedance control is validated by

experimental results and compared to a conventional optimal control. Finally, advancements of these controllers are discussed. In Chapter 6 (Rehabilitation Robot Control: Human-Robot Interaction Control), model-based control structures based on the NMPC idea are presented and validated by model-in-loop simulations. In Chapter 7 (Conclusion), concluding remarks are provided. Followed by the list of references, the appendices are provided at the end of this thesis.

Chapter 2

Background

This thesis combines mechatronic (i.e., the rehabilitation robot) and bio-mechanical (i.e., the musculoskeletal upper extremity model) systems (i.e., *biomechatronics*). In this chapter, first, actuator models of the musculoskeletal model (i.e., muscle models), which provide information about fundamental muscle mechanics, muscle paths, and solving the muscle redundancy problem, are reviewed. Next, musculoskeletal upper extremity models and their components are studied. Then, a brief introduction to upper extremity movement disorders is given. Finally, a comprehensive review of upper extremity rehabilitation devices is provided.

2.1 Muscle models

When a motor task is executed by the body, the Central Nervous System (CNS) stimulates muscles to generate forces. These forces are applied to the skeletal structure using tendons. In other words, muscles–or more precisely musculotendons–are connecting the CNS to the moving body segments. Thus, the study of musculotendon models for rehabilitation engineers, who are working on the design of rehabilitation devices to restore impaired motor unit functionality, is of great importance. There are three different aspects of muscle modeling:

- Muscle mechanics,
- Muscle paths, and

- Solving the muscle redundancy problem.

In upcoming subsections, each of these aspects is reviewed.

2.1.1 Muscle mechanics

Muscle mechanic models can be defined implicitly or explicitly. Implicit models use Electromyography (EMG) activities of the muscles [78] or Functional Electrical Stimulation (FES) response of the muscles [248] to assess their force contributions. This estimation is qualitative and sometimes is combined with muscle force-length and force-velocity properties. Explicit models have mathematical representations. These models can be based on a physics-based or a "reductionist" approach [336].

The physics-based approach is defined based on macroscopic properties of the muscle, and it is an Input-Output (IO) model regarding muscle dynamic characteristics [336]. One of the first models of this type is the simple second-order system model, in which the muscle is modeled as a mass-spring-damper system with a simple Ordinary Differential Equation (ODE), and there is no tendon in the model [315]. The advantage of these models is their mathematical simplicity [251]. However, the parameters of the model are task-specific, so they should be adjusted for each task [315]. The second physics-based approach is the Hill-type model, which resulted from Hill's studies on isolated muscles [115]. There are various types of Hill-type models; the most common is the model with three elements consisting of Contractile Element (CE), Parallel-elastic Element (PE), and Series-elastic Element (SE) [251, 259]. There are other models of the physics-based approach available in the literature such as [37, 112, 174, 311, 312].

The reductionist approach considers the microstructure of the muscles [336]. Hence, the equations are based on microscopic properties of the muscles. One of the models of this type is the Huxley-based model, in which the internal structure of the muscle is considered in the muscle dynamics model [315]. Other models of this type are available in the literature such as [17, 303].

Implicit models cannot be used in math-based modeling since they are not mathematical and they are qualitative. Among explicit models, the selection of a suitable approach depends on the objective of the project. In this study–math-based modeling–we are not dealing with the microscopic physical and chemical musculotendon properties; besides, the description of the system is in terms of inputs and outputs with known muscle structure and function. Therefore, we utilize the physics-based approach considering muscle characteristic curves. Concurrent with previous research, this study utilizes Hill-based lumped

parameter models, unlike other physics-based approaches such as second-order systems [50, 313]. Through various types of Hill-based models, Thelen [283] has adjusted parameters of these models to simulate muscle mechanics of older adults. Rehabilitation is common among older adults, and Thelen's modifications have been used in various studies such as [53, 134, 223, 250, 252, 299]. Hence, in this study, Thelen's modified Hill-based muscle model is used.

2.1.2 Muscle path models

Beside muscle mechanics models, which are used to estimate forces, muscle moment arms about joints and muscle lengths may affect the results of the musculoskeletal simulations [9, 19, 26, 78, 111, 123, 207]. In other words, the results of the simulations depend on the muscle paths [78, 316], which have been modeled in different ways.

Straight-line model

In this model, the muscle path is defined by a straight line connecting muscle attachment sites to each other [25, 56, 73, 107, 244]. Although these models are easy to implement, they may not give accurate results [78, 97, 339].

Centroid-line model

In this model, the points on the cross-sectional areas of a muscle are connected by a curve to generate the muscle path [8, 9, 16, 35, 132, 289, 319]. One of the methods for approximating the muscle centroid-line path is the via-points approach. In this method, muscle path sites are defined by points attached to the skeletal segments. These via-points can be fixed to the skeletal segments [25, 63, 257, 273, 314, 331], or they can be muscle points moving relative to the attached bony segments [273]. Although the via-points approach seems to be accurate for one-dimensional joints such as elbow flexion/extension joint, they may not be significant in multi-dimensional joint movements or for large joint angles [97].

Geodesic models

Muscle paths are often modeled such that the muscle wraps around some surfaces which are representative of anatomical constraints [46, 97, 175, 200, 294]. These methods consider that the acting muscle force is along the centroid line of the muscle, and the muscle is a

frictionless elastic strap which wraps over the anatomical constraints [97, 241, 338]. Natural geodesic variation methods are more applicable for complex wrapping geometries [241, 338]. However, if the wrapping surfaces are modeled with simple shapes, the obstacle-set method is more efficient than other geodesic methods [241].

Finite element model

This model is based on the finite element method, which is used for the detailed representation of the complicated muscles [23, 86, 94, 161]. Thus, the method needs a large amount of data and mathematical calculations for an accurate three-dimensional model generation.

Non-geometric path model

Detailed musculoskeletal models increase dynamic simulation times. The reason can be found in the muscle geometrical path definitions for these models. To overcome this issue, instead of using the geometric muscle path model, fitted functions are used to relate musculotendon length and moment arm to joint angles. In [225] regression-based polynomial equations are fitted for muscle moment arm and musculotendon length in terms of joint angles. In [43, 293] a polynomial equation is fitted for musculotendon length in terms of joint angles; then considering the virtual work principle, the muscle moment arm about a joint is calculated by differentiating the musculotendon length function with respect to the joint angle.

Discussion

A point to note about muscle path models: if the number of muscle origin or insertion points is more than one, these muscles are modeled as two or more muscle elements [121]. Although a non-geometric path model is mathematically efficient, fitting a function for musculotendon length for a large number of joint angle variations needs an appropriate geometric muscle path model. Among the mentioned geometric path models, the obstacle-set method is selected for this study because of relatively realistic and simple representation of the muscle for a relatively small amount of mathematical load.

2.1.3 Solving the muscle redundancy problem

Having a good picture of muscle and joint forces is of importance in many biomechanical fields. Direct measurements of these loads are almost impossible; hence, this motivates

us to use indirect methods such as musculoskeletal modeling to approximate these forces. Nonetheless, since the number of muscles acting on a joint is more than one, the dynamic problem of the musculoskeletal models becomes redundant. In other words, there can be numerous results for a single equilibrium condition. To overcome this issue, it seems a logical interpretation to assume that the CNS adjusts muscle forces in an optimized manner [3, 55]. Different methods for solving the muscle force sharing problem are summarized below:

Static Optimization (SO)

In this method, muscle forces are determined by instantaneous optimization of an objective function. Although this method is mathematically efficient, it may lead to results with abrupt changes because of ignoring muscles contraction and activation dynamics. Other than that, since each time step is optimized independently, it is not possible to have time history dependent objective functions such as metabolic energy. Different types of instantaneous objective functions have been defined in the literature such as [288].

Dynamic Optimization (DO)

This method is the optimal control of the musculoskeletal model [61]. The neural excitations are the control inputs for the forward or inverse dynamics problem to produce desired motion trajectories or joint torques [185], respectively. To solve the two point boundary value problem in the optimal control, there are some studies that have changed the optimal control method to a parameterized optimization problem, and they have reported satisfactory results [10, 60, 89, 208, 245]. DO is suitable for the motions affected by dominant muscle activation dynamics and co-contraction [191]. Since DO uses the time history of the simulation along with muscle contraction and activation dynamics, unphysiological results are not obtained, and it is possible to define time-integral cost functions. However, considering the whole time history increases the computation cost enormously, especially for complicated musculoskeletal models.

Extended Inverse Dynamics (EID)

This approach, proposed by Ackermann [2], is a variation of dynamic optimization whereby:

1. The problem is solved by inverse dynamic equations.

2. The inputs to the optimizer are the muscle forces along the corresponding tendons. The activations and neural excitations are evaluated by inverting contraction and activation dynamics, respectively, once the optimization problem is being solved.

The computational load of this method is less than DO, and in contrast to SO, it considers contraction and activation dynamics together with a time-integral cost function [3].

Inverse-forward Dynamic Optimization (IFDO)

It is a type of DO which includes both the forward and inverse contraction/activation dynamics [198]. This method can also be combined with the forward dynamic model of the musculoskeletal model to generate a correction control input to the optimization problem; the resulted approach is Inverse-forward Dynamic Optimization Control (IFDOC).

Inverse-inverse Dynamic Optimization (IIDO)

If in a DO problem, some design parameters should be identified in addition to the unknown muscle force prediction, single inverse dynamic optimization cannot solve the problem. Rasmussen et al. [227] has proposed a method called IIDO which solves this issue by introducing outer inverse dynamic parameter identification [60].

Modified Static Optimization (MSO)

Because of high computational costs of DO and EID, for real-time simulations, it is often necessary to use SO with instantaneous cost functions. However, SO does not account for contraction and activation dynamics, which may cause unphysiological results. Therefore, MSO was proposed to solve this issue [2]. MSO is similar to SO, with the difference in having nonlinear constraints, which are related to the contraction and activation dynamics. This approach is also analogous to EID, but the optimal problem is solved instantaneously and the results of a time step are only dependent on the results of the previous time step.

Forward Static Optimization (FSO)

This approach, which has been utilized by Sharif Shourijeh [251], is like a DO in which the objective function is optimized at each time step while satisfying kinematic constraint equations of forward dynamics integration. In another point of view, FSO is similar to

MSO, where instead of inverse dynamics, the forward dynamics problem is solved with activations or excitations as inputs. Looking closely, FSO is a special case of NMPC, which optimizes the current time step while considering the future time step [180, 254]. Another variation of this method has been utilized in [43, 293]; these studies use different numerical approaches for implicit solution of the dynamic equation.

Computed Muscle Control (CMC)

CMC, which is presented by Thelen et al. [285], is a kind of SO with implemented control algorithms to track the kinematics from a forward dynamics integration [284]. This method cannot use time-integral cost functions, similar to SO.

Muscle synergy

Muscle synergy theory has a biologically-plausible approach to deal with muscle force sharing problem [286]. Based on this method, during a task muscles are activated in a group called synergies. Although this method is computationally efficient [20], mostly it has been used in inverse dynamics analysis in these studies [253, 262, 269, 337]. Yoshikawa et al. [332] use equilibrium-point-based synergies to translate human movement to a robot. Parallel to our current study, a project in Motion Research Group (MoRG) at the University of Waterloo has been done by Sharif Razavian [248], which considers muscle synergies based on the task space in forward dynamic simulations [250].

Other approaches

Analytical optimization technique has been used by Challis and Kerwin [44] to estimate muscle forces. However, the evaluated forces are not bounded. Hence, this method may result in negative values for muscle forces, which is unphysiological. In another approach, the pseudo-inverse is used to evaluate the inverse of the muscle moment arm matrix, then the muscle forces constraints are imposed with the aid of the null space definition in quadratic programming [281]. This approach has required some simplifications such as considering muscles as string elements. There are also other studies which use specific tools such as stochastic modeling and muscle fatigue criterion to deal with muscle force sharing problem [176, 249].

Table 2.1: Components of the shoulder

No	Joint	Location	Mechanical function
1	Sternoclavicular Joint (SCJ)	Base of clavicle	Connecting clavicle to thorax
2	Acromioclavicular Joint (ACJ)	Base of scapula	Connecting scapula to clavicle
3	Glenohumeral Joint (GHJ)	Humeral head	Connecting humerus to scapula
4	Subacromial Space (SAS)	The space between the acromion and rotator cuff at humeral head	Allows sliding between acromion and rotator cuff
5	Scapulathoracic Joint (STJ)	Floating movement base of scapula	Allows sliding of scapula on thorax

Discussion

In this study, it is desired to have a musculoskeletal model to interact with a rehabilitation robot model. Consequently, fairly quick and instantaneous evaluation of the muscle forces for design and tuning of a controller for the robot is demanded. Therefore, among various solutions for the muscle force sharing problem, the methods based on a static optimization approach (i.e. FSO, MSO, and CMC) seem to be most applicable to our research.

2.2 Musculoskeletal models of the upper extremity

In this section, an anatomical review of the upper extremity is presented, which is adopted from [242]. This review presents the issues that should be considered in an upper extremity model excluding the hand. Then, we will briefly review available musculoskeletal upper extremity models for the shoulder, elbow, and wrist.

2.2.1 Anatomical review and issues

The upper extremity is composed of the shoulder girdle, upper arm, forearm, and hand (Fig. 2.1, it is the modified version of SolidWorks® model adopted from [143]).

The shoulder girdle (clavicle and scapula) is an interface between the upper extremity and thorax. Among different components of the upper extremity, the shoulder is more complicated because:

1. It is composed of 5 joints (see Table 2.1 which is adopted from [65, 242]).

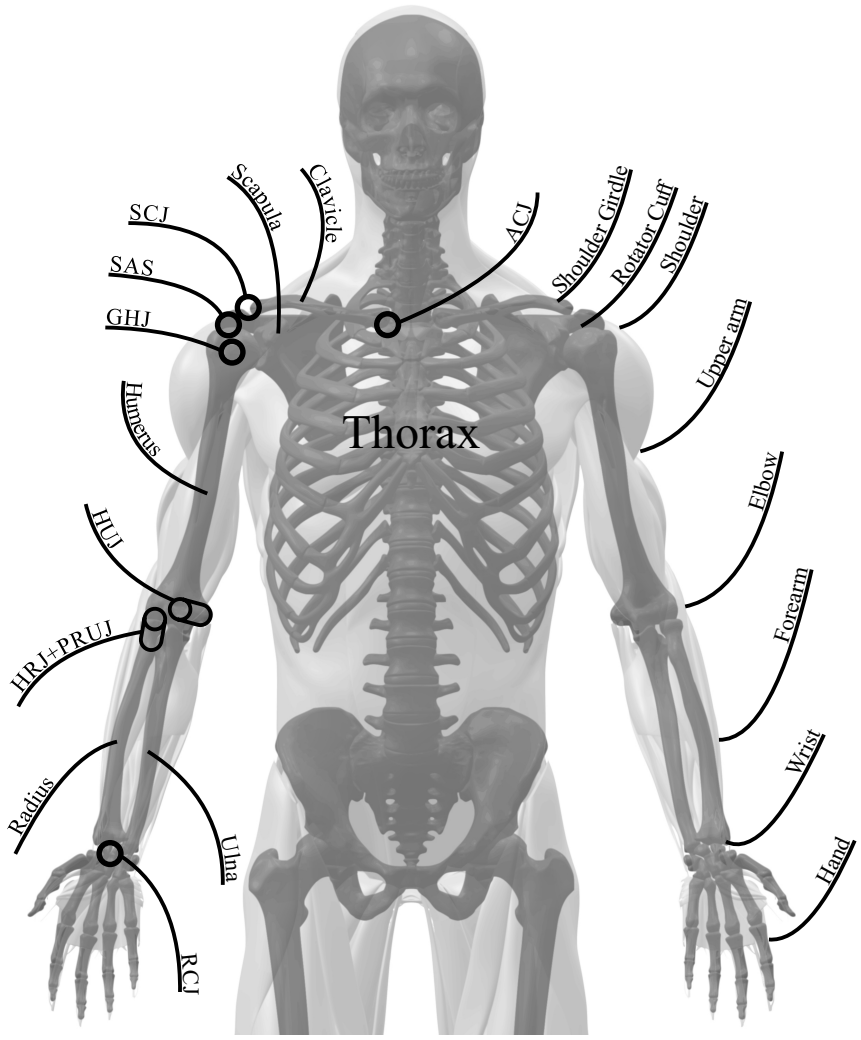


Figure 2.1: Upper extremity anatomy.

2. In addition to muscles and joints, there are two ligaments which affect the shoulder motion.

3. The joint capsules and ligaments of the shoulder joint are weak. Most shoulder defects are related to soft tissue, and 45% of dislocations correspond to the shoulder joint.

4. During shoulder abduction, the upper arm and scapula angles vary with the ra-

tio of 2:1. This is called Humeroscapular Rhythm (HSR). In general, the shoulder joint movement constrains and imposes motion to the SCJ and ACJ, which is called *shoulder rhythm* or Humerothorax Rhythm (HTR). These movements can be recorded by motion capturing techniques. There are some other methods such as regression-based equations which evaluate the dependent motions [326].

The shoulder girdle is located in the upper extremity, and the hip girdle is in the lower extremity. The hip girdle is mainly for supporting gait and total trunk weight. Hence, the functionality of the shoulder girdle is more than the hip girdle, and this boosts the importance of the shoulder study and modeling in biomechanics.

The elbow is the second important component of the upper extremity. It consists of three joints: Humeroulnar Joint (HUJ), Humeroradial Joint (HRJ), and Proximal Radioulnar Joint (PRUJ). The HUJ connects the ulna to the humerus and provides elbow flexion/extension. The HRJ attaches radius to the humerus and allows elbow pronation/supination. The PRUJ joins proximal parts of radius to the ulna, and while allowing relative pronation/supination of the radius with respect to the ulna, it transfers elbow flexion/extension to the radius. The PRUJ can be modeled as a revolute joint with a fixed base on the ulna.

Finally, the wrist is the last important component. It consists of two joints: Radiocarpal Joint (RCJ) and Distal Radioulnar Joint (DRUJ). The RCJ connects hand to the radius; it can be modeled as an ellipsoid joint[1]. The DRUJ attaches distal parts of the ulna to radius and couples pronation/supination in proximal parts. The DRUJ can be modeled as a revolute joint with a fixed base on the radius.

Other than skeletal complexity, there are 52 muscles (consisting of 9 shoulder girdle muscles, 9 shoulder joint muscles, 4 upper arm muscles, 19 forearm muscles, and 11 hand muscles) in the upper extremity, which increase the complication of the musculoskeletal models.

2.2.2 Musculoskeletal shoulder models

There are several musculoskeletal upper extremity models, which are summarized in Table A.1. The common feature of these models is the inclusion of the shoulder. However, as in Table A.1, the shoulder has been modeled with a different number of joints discussed

[1]This joint is composed of an ellipsoid and ellipsoidal cavity.

previously [77]. These models use different numbers of muscles considering muscle mechanics and muscle paths to solve the muscle redundancy problem by one of the mentioned methods. Most of these models are 3D, but some of them are 2D.

2.2.3 Musculoskeletal elbow models

Some elbow models are integrated into the upper extremity models in Table A.1. In these models, the elbow has been modeled as a single revolute joint (flexion/extension joint) or two revolute joints (flexion/extension and pronation/supination joints). In the literature, there are some detailed models of the forearm pronation/supination [142]. However, for simplicity, in current models, the pronation/supination joint is modeled only by a radioulnar joint which connects proximal parts of radius and ulna. In other words, there are no humeroradial and distal radioulnar joints in current musculoskeletal models.

2.2.4 Musculoskeletal wrist models

Among musculoskeletal upper extremity models, only the Texas and Stanford models [96, 122] have a wrist model. In these models, the wrist is modeled as two independent revolute joints even though the wrist consists of a radiocarpal joint that is best modeled by an ellipsoid joint. Furthermore, the ROM for two independent revolute joints is more than the ROM of an ellipsoid joint. Thus, two independent revolute joints may result in unphysiological configurations for the wrist.

2.2.5 Sensitivity analysis of the musculoskeletal upper extremity model

Muscle parameters based on collective data of experiments are being used in muscle simulation models. However, these parameters cannot be representative of everyone's muscle parameters since muscle properties are changing from one subject to another [324]. These parameters may be different for various ages, genders, and neuromuscular conditions [134, 158, 194], and this difference may lead to unreliable evaluations. Thus, subject-specific muscle properties are required to include in musculoskeletal models. It is not easy to measure muscle properties for each subject, and some generic values are usually used for these parameters [63]. To determine the effect of each muscle, a sensitivity analysis of muscle parameters should be conducted. For example, it has been shown that a musculoskeletal system with a muscle-tendon model is more sensitive to the tendon slack length

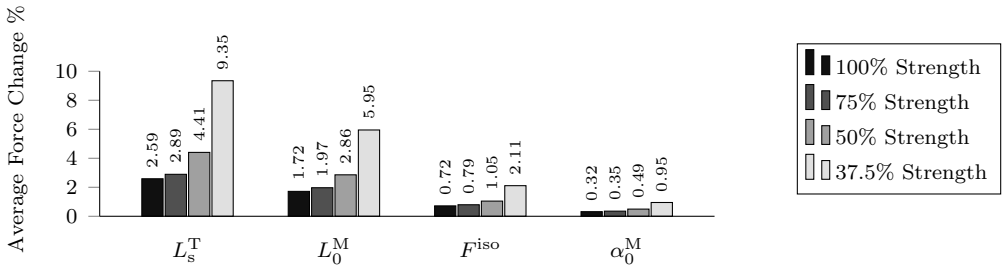


Figure 2.2: Average muscle force change when perturbing each corresponding parameter with 1% perturbation in the musculoskeletal upper extremity model with different strength levels.

parameter than other muscle parameters [40, 85, 91]. There are two types of sensitivity analysis of Hill-type muscle models:

1. A muscle parameter is perturbed, and the muscle force is recalculated [62, 243].
2. A muscle parameter is perturbed, and the musculoskeletal system behavior is evaluated [109, 228].

In [41], the Stanford musculoskeletal upper extremity model with different strength levels (which models healthy to impaired subjects) is used to analyze its muscle force sensitivity to the perturbed muscle parameters including: tendon slack length (L_s^T), muscle optimal fiber length (L_0^M), maximum isometric muscle force (F^{iso}) and muscle pennation angle at optimal length (α_0^M). This evaluation is done in a static pose and results are summarized in Fig. 2.2. Muscle force estimation is more sensitive to the tendon slack length than other parameters. Once the patient is weaker in strength, the muscle force sensitivity increases significantly.

Other than muscle parameter sensitivity study, simulation results may be sensitive to the kinematic parameters of the skeletal body. In a sensitivity analysis which was done on an EMG-driven musculoskeletal elbow model [325], this feature is studied. The muscle force sensitivity was high for the length of the humerus segment.

2.2.6 Discussion

There are some other models which have been developed for specific applications:

- Kinematic musculoskeletal upper extremity model for real-time interaction [209]: which is developed for SANTOS^{TM}software as a part of the Virtual Soldier Research program. In this model, the shoulder is modeled with 3 spherical joints, including: SCJ, ACJ, and GHJ. The elbow is modeled using two revolute joints for RCJ and DRUJ. The model has 21 major muscles and uses the obstacle-set method for muscle wrapping.

- Flexible body musculoskeletal upper extremity model [145]: this is a finite-element musculoskeletal model which is developed from MADYMO human models[2]. It is composed of spherical joints at the shoulder, elbow, and wrist. The model has 31 muscles, and they are modeled with the CE/PE/SE Hill-type definition. A type of FSO is used to solve the muscle force sharing problem. However, since it is a finite-element model, it is computationally expensive.

- 3D musculoskeletal driver model: it is developed in MoRG at the University of Waterloo [181]. Because of the task-specific definition of this model, it has fewer degrees of freedom than the presented models. In this model, the universal shoulder joint, revolute elbow joint and spherical wrist joint are the defined joints. The driver model has 10 muscles defined by straight-line and via-point methods. The muscle mechanics model is based on the CE/PE Hill-type muscle definition, and its redundancy has been solved by the FSO approach.

- Dynamic upper extremity model [240]: this model is the modified dynamic Stanford model, in which masses and inertias for the shoulder girdle elements have been defined. The model is developed in the OpenSim open-source software package, which is a musculoskeletal simulation toolkit from Simbios [64, 246].

- Musculoskeletal arm model with real-time simulation [43]: this model has been developed for the design and assessment of a controller for a Brain-computer Interface (BCI). The goal is to enable user-in-the-loop experiments; thus, real-time simulations are demanded. The model uses 29 muscles (138 elements) with the CE/PE/SE Hill-type definition and non-geometric muscle path method. The shoulder is modeled by three spherical joints, and the STJ and shoulder stability are imposed as constraints. Other than muscles, one ligament and joint passive moments are considered. Shoulder rhythm is controlled by muscle forces, and the muscle force sharing problem is solved by FSO. For simulations OpenSim is used, where the upper extremity model definition is based on the Delft model [198].

[2]MADYMO (from Tass International) is a software package to analyze passenger safety

- Closed-chain human arm dynamics [60]: in which the musculoskeletal upper extremity model of The AnyBody Modeling System™is used. The model has 104 muscles with the CE/PE/SE Hill-type definition. Since the model rotates a crank on the horizontal plane, the number of active joints reduces to the shoulder flexion/extension at GHJ and the elbow flexion/extension at HUJ. Since there are some unknown design parameters in this closed-chain problem, the IIDO is used to solve the muscle force-sharing problem with unknown design parameters.

Although there are many musculoskeletal upper extremity models, comparison of them and selection of a suitable model is difficult, because they are not using the same anthropometric data and joint coordinate systems. For this reason, the ISB has proposed a standard approach for the upper extremity joint coordinate systems [322] to unify the research in this area. Some of the mentioned musculoskeletal upper extremity models are using this recommendation. The Stanford, Delft[3], and Lisbon models [122, 221] have more common features with the ISB definition. These models are using a protocol [297] for the definition of the scapula coordinate, which is different from the ISB recommendation. There are also commercially available musculoskeletal models like SIMM (from Musculographics Inc.), The AnyBody Modeling System™[58], VIMS [45], OpenSim [64, 246], MADYMO, SANTOS™, and LifeMOD™(from LifeModeler®). Nonetheless, since the goal of this project is to develop model-based controllers for an upper extremity rehabilitation robot, using the above software packages is not advantageous for symbolic modeling, and real-time simulation and feedback control.

In the current study, primarily we will utilize 2D musculoskeletal arm models [197, 276] because for real-time simulations it is often necessary to use low-fidelity models. Then, we will use 3D musculoskeletal arm model (i.e., the high-fidelity model) to justify the significance of the 2D model (i.e., the low-fidelity model). Since the Stanford model is a complete model of the upper extremity and also is available in OpenSim, we will utilize and modify this model for our primary 3D musculoskeletal model. The Stanford model is limited by:

1. ROM for some joints: for example, maximum positive shoulder rotation angle in this model is 20 degrees [122], while this value in most references has been reported to be about 90 degrees [229].

2. Imperfect shoulder rhythm definition: in the Stanford model, the shoulder rhythm is defined by simplified regression-based equations which are only dependent on

[3]Since "dynamic upper extremity model" and "musculoskeletal arm model with real-time simulation" are using Stanford and Delft models as their base models, respectively, we have not mentioned them here.

the shoulder elevation angle [122]. However, the most accurate shoulder-rhythm regression-based equations are in terms of three shoulder joint angles [326].

3. Incorrect muscle path generation for some orientations: this is due to the imperfect muscle path definition based on the via-point method.

These shortcomings led us to develop an advanced 3D musculoskeletal arm model based on the ISB recommendation. Defining muscle-wrapping geometries requires a proper database. The Texas upper extremity model has a complete muscle path definition based on the obstacle-set method. This model is defined in different coordinates than the ISB recommendation and it does not account for the motion of the shoulder joint (shoulder rhythm HTR). To develop the advanced musculoskeletal arm, the primary goal is to define the Texas model using the ISB recommended coordinates with the shoulder rhythm determined by the regression-based approach of [326]. The Lisbon model is the Texas model defined in ISB recommended coordinates except for the scapula coordinate. This redefinition has some deficiencies, and we will overcome those imperfections by redefining the ISB recommended coordinates, proper positioning of the clavicle and scapula, and defining non-geometric muscle path which is required for real-time simulations. Finally, to get real-time responses from the model-based controllers, we will develop 2D musculoskeletal arm models for interaction with a planar rehabilitation robot.

2.3 Upper extremity movement disorders

Upper extremity movement defects are caused by different sources such as upper extremity component injuries and surgeries, overuse syndromes [260], stroke, TBI, SCI, motoneuron defects, and neurological diseases such as Cerebral Palsy (CP) and Parkinson's disease [170]. Most of these defects need sessions of physical therapy to improve joint ROM, strengthen muscles [260], restore functional capabilities and resolve impairments [170].

Stroke causes longstanding impairments, and it is common for older adults. Thus, procedures are being used to rehabilitate this long-term disability [31, 206, 290]. Among various movement disorders followed by stroke [14], upper extremity motor defects are very common [184]. Therefore, rehabilitation approaches for upper extremity motor control and function recovery are of importance. Consequently, we focus on upper extremity movement disorders in post-stroke patients. Neurological complications of stroke are various [93], such as:

1. Hemispheric behavioral differences: stroke patients may show different behaviors in doing a task. Those with right hemiplegia have hardship accomplishing consecutive tasks so that these patients may need some assistance in their therapy. On the other hand, patients with left hemiplegia have task perception problem, and they overestimate their abilities. Fluctuations in doing a task are common among them. Besides, because of the wrong perception, safety issues should be considered carefully.

2. Perceptual dysfunction: it is common among left hemiplegia patients, and can be revealed as one of these symptoms: body scheme, spatial relation, and agnosia. Body scheme is the difficulty in realizing the relationship between body parts. The spatial relation is having trouble in perceiving the relationship between body and other objects. Agnosia is the problem in distinguishing incoming information which can be visual, auditory or tactile.

3. Osteoporosis and fracture risk: because of the lack of physical activity, these patients may get osteoporosis. Osteoporosis is a bone disease for which the mass of bone will decrease and cause fractures.

2.4 Upper extremity rehabilitation devices

There are two main types of training for stroke rehabilitation: unilateral and bilateral [321]. Unilateral training is therapy for the single impaired limb. Constraint induced therapy, which is an intensive use of the impaired limb while constraining the unaffected limb, is a kind of unilateral training therapy. Taking into account bimanual daily activities like hand washing, the idea of getting more help from undamaged neural pathways, and case-dependent use of unilateral training, has led to bilateral training theory. Bilateral training is used for symmetric/asymmetric movements of both impaired and unimpaired limbs [272]. Although these two training approaches are different, they are pursuing the same goal. Recent studies [292, 321] have stated that there are no significant outcomes that can make one method of training superior to the other. The procedures of these training methods are developed by motor learning theories. These theories are sometimes in contrast and are not completely determined; some of the available ones are [29]:

- Implicit or explicit learning: implicit learning is unconscious during indirect task execution, while explicit learning is directed.

- Massed or variable practice: massed practice is repetitive single task accomplishment, while variable practice is multiple task training.

- Feedback distortion or assistance: feedback distortion is magnifying movement errors instead of assisting the patient to reduce the errors.
- Real world practice: real world practice can be done by virtual reality methods.

To show the need for rehabilitation robots, we should survey the goals of therapy [230]:

- Increase activity: it is done by the use of Thera-bands, pegboards, and blocks in conventional therapy.
- Provide assistance: conventionally is accomplished by the help of splints, and arm-supports.
- Improve assessment: traditionally is achieved by force gauges, goniometers, and timers.

Considering these goals and their effectiveness, the features of robotic devices like automation and versatility in procedures and assessments [230], increasing physically impaired patient population [170], and the limited number of therapists and decreased therapy hours because of economic issues, justify the employment of rehabilitation robots in therapy sessions [232]. Complete reviews of upper extremity rehabilitation robots has been done in [29, 30, 113, 170, 219]. Based on [29], these robotic systems can be categorized by training approaches. Accordingly, these robots are:

1. Unilateral trainers, which can provide:
    - Gross motor movements: in which massed practice with explicit learning is accomplished. It is an old method of therapy used in various rehabilitation robots, or
    - Fine motor movements: it is mostly related to hand and wrist rehabilitation. This method can be used for increasing ROM or regulation of motor tasks like independent movements of fingers.
2. Bilateral trainers, which perform bimanual therapy.

Together with the above tasks, some robots have additional features such as real world practice, FES, EMG, Electroencephalogram (EEG), gravity compensation, feedback distortion, telerehabilitation, and progress assessment [29].

In this section, following topics in upper extremity rehabilitation robots are studied: application, mechanical design, control scenarios and Human-Robot Interaction (HRI).

2.4.1 Application

Upper extremity rehabilitation robots can support daily activities and are designed for home or clinical use [170]. The target population for most of these robotic systems is post-stroke patients, for whom these robots can:

1. provide active/passive assistance therapy (*active devices*). In passive mode, the robot moves the patient's limb without any muscular activity of the passive patient, while in active mode the patient is active during training.

2. perform passive resistance therapy (*passive devices*).

3. be used as *haptic devices* for real world practice, to transfer tactile sensing to the patient. They do not assist or resist movement.

4. coach the individual by providing real world practice via visual or auditory feedback (*coaching devices*).

2.4.2 Mechanical design

Upper extremity rehabilitation robotic systems can be classified as [170]:

1. *Manipulanda*: these are end-effector based robots that have a simple structure, and less complex control algorithms. However, it is hard to perform special movements of a distinct joint using these robots. Another design issue in these robots is that the end-effector at most can provide 6 DOF. Thus, the number of anatomical movements should not exceed 6 otherwise it will cause redundancy, which is unsafe. "InMotion Arm" and "ReoGo" are commercially available robots of this type.

2. *Exoskeletons*: these robots can provide movements to particular joints, and the number of anatomical movements can exceed 6. Nonetheless, increasing the number of movement parts increases the number of device modules, so the system setup becomes difficult. Moreover, since the shoulder has a variable joint center, the mechanical design and control algorithms become more complicated. "ArmeoPower" and "ArmeoSpring" are commercial exoskeletons [219].

2.4.3 Control scenarios

For exerting different therapy approaches by upper extremity rehabilitation robots, different control algorithms are utilized. The control inputs are dynamic such as force and torque measurement signals, kinematic displacement and velocity measurement signals, and triggers such as switches and EMG signals. Their feedback to the user are tactile, visual, auditory, and FES. The control strategies for these robots are categorized as [170, 219]:

1. High-level control scenarios, which help to stimulate motor plasticity.
2. Low-level control scenarios, which are used to implement high-level scenarios.

High-level control scenarios

There are three high-level control scenarios [170, 173, 219]:

1. Assistive control:
   (a) Passive control: the device tries to constrain the patient's hand to the desired track. This track can be defined in different ways.
      i. Passive trajectory tracking: it is a reference tracking control.
         A. Kinematic-based position control: the tracking can be done on a smooth trajectory [7, 28, 84, 136, 152, 168, 188, 237, 318] which is determined by "minimum-jerk" hypothesis [87].
         B. "record-and-replay" control: the trajectory can be obtained from unimpaired volunteers [148, 268].
         C. "teach-and-replay" control: the reference path can be generated by the therapist guidance [216].
      ii. Passive mirroring: which is based on bilateral training. Thus, the desired trajectory is a path followed by the unimpaired limb [105, 216].
      iii. Passive stretching: in which the limbs are coordinated by measuring the angle-resistance torque relation [234].
   (b) Triggered passive control: the device uses biosignals as control inputs, but this triggering may cause slacking in which the patient does not show any effort and waits for the robot assistance. These controllers are:
      i. gaze-based tracking [165, 201]

ii. EMG-based [42, 49, 54, 70, 74, 81, 157, 166, 222, 270],
iii. FES-based [124, 140], and
iv. BCI-based (which also includes EEG-based controllers) [15, 27, 88, 92, 238].

(c) Partially assistive control:

i. Impedance-based assistance: different variations of impedance and admittance controls are used to control the rehabilitation robot [39, 51, 57, 106, 139, 187, 231, 287, 334].
ii. Attractive force field control: which uses some type of manipulability ellipsoid to apply force in specific directions [144, 327].
iii. Model-based assistance: in which a musculoskeletal upper extremity model is used to implement a model-based assistive control in an exoskeleton[68, 69].
iv. Learning-based control: in which the adaption to the performance index is done from trial to trial. Offline adaptive [13, 215, 218, 317] and Artificial Intelligence (AI) [110] controls are types of this control structure.
v. Counterbalance-based control: the device applies active/passive counterbalance to the patient limb for gravity compensation [129, 186, 188, 239, 271, 274].
vi. Performance-based adaptive control: the robotic system gets the performance of the patient using error-based strategy and adapts some features for assistance [138, 152, 236].

2. Resistive (challenge-based) control:

(a) Resistance induced: resists patient's movements [190, 210].

(b) Error amplification (feedback distortion): which is done by amplifying errors kinematically [213, 214], visually [28, 214, 309] or tactilely [162].

(c) Constraint induced: this is same as constraint induced therapy [135, 256].

3. Corrective control: it is a kind of time-independent assistive control, in which the assistance is done when there are large tracking, coordination, or skill errors.

(a) Tunneling: it is a kind of impedance-based control which is applied on boundaries of a wider trajectory [104, 146, 172].

(b) Coordination (synergy-based) control: which prevents large coordination errors between joints during a rehabilitation task [33, 52, 103].

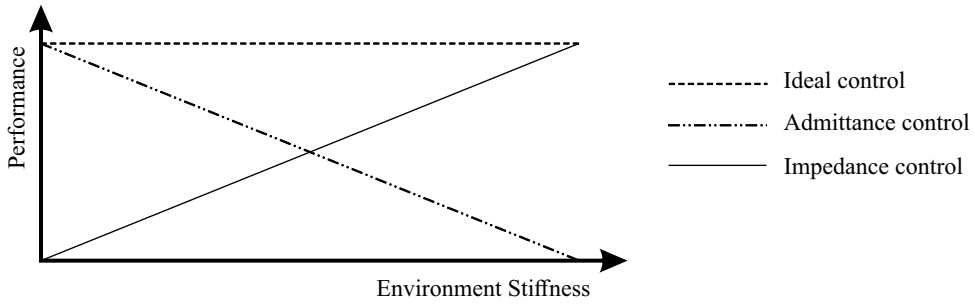


Figure 2.3: Qualitative performance of impedance and admittance controllers in different environments.

(c) Haptic provoke: which is used for providing real world experience based on gaming control schemes [32, 36, 212, 330].

Low-level control scenarios

In robotic rehabilitation, since the human body is interacting with the mechatronic device, safety issues in the design of appropriate control strategies are very important. Conventional position or force control approaches (because of poor dynamic interaction modeling) are not safe enough to be implemented in these devices [117]. Therefore, modified control approaches like impedance and admittance control are used. In impedance control, the position of the impaired limb is measured, and appropriate force is applied (i.e., it is a force control with a position feedback), while in admittance control the applied force by the impaired limb is measured and the corresponding movement is imposed (i.e., it is a position control with a force feedback). Use of these methods is design and task specific. Impedance control has a poor accuracy; however, it becomes more stable by increasing the environment stiffness (see Fig. 2.3 which is adopted from [205]). On the other hand, as in Fig. 2.3, admittance control in stiff environments is not stable, while it has a good accuracy in less stiff environments. Implementing admittance control needs high transmission ratios to be considered in the mechanical design, while impedance control works with direct drives (i.e., it is efficient for a light-weighted back-drivable robot) [205, 219].

2.4.4 Human-robot interaction

The goal of the HRI field is the design, development, and assessment of the "human-centered" products [102, 167]. HRI research in the upper extremity robotic rehabilitation

dates back to 1990s [300]. The interaction term in HRI for rehabilitation robots can be categorized into two levels: physical and social. Since this project does not focus on non-contacting robotic devices, the HRI will be studied at the physical interaction level.

To study HRI in rehabilitation robotics, we should consider these six HRI principles[4] [167]:

1. Ability level: indicates the amount of robot's ability to perform a task, and this factor can have 10 levels varying from the no-assistance to independent control modes. This will be adjusted by the designed low-level controllers.

2. User interface: this can be auditory, tactile, or visual. The user interface is considered tactile.

3. Interaction arrangement: which includes single-robot and single-user, single-robot and multiple-user, and multiple-robot and single-user. Our rehabilitation configuration has a single-robot and single-user arrangement.

4. Learning and adaption: both robot and user should learn and adapt to each other's performances. The NMPC will help to provide the required learning and adaption.

5. Exterior design: the robot should be catchy and appealing for the user. Aesthetics are out of the scope of this study.

6. Therapy time: which is each rehabilitation session's duration. This will be decided later in consultation with a therapist.

along with the following HRI metrics:

1. User acceptance: which indicates how much the user is satisfied with the robot. It is evaluated using a questionnaire, which is available in our ethics application.

2. User participation: this shows how long the user is engaged in the robotic rehabilitation task, and it is measured through robot-based assessments.

3. User accompaniment: this evaluates how often the user is accompanying the robotic task (learning and adaption), and it is measured with robot-based assessments.

4. User safety: which is applied by limiting robot's ROM, kinetic variables, and motor torques. Our proposed low-level controllers deal with this issue.

[4]Underlined terms are the actions taken in this project.

2.5 Dynamic parameter identification

DPI of a mathematical model of the system is of great importance for the model-based control of robots. For example, in robotic rehabilitation, different types of low-level controllers are implemented to deliver desired partial assistance during a therapy session [170, 219]. To implement these controllers, a complete and accurate dynamic model of the robotic system is often required. This requirement may be reduced by integrating robust or adaptive control approaches with the above controllers. For robust control, if the robot's dynamics are too uncertain, the quality of the optimized assistance or resistance may drop during the therapy [255]. In adaptive control, it is required to have a persistently exciting input for the convergence of the adaptive law [255], which will adversely affect the patient's motivation for using the device. Also, both adaptive and robust controllers may need high gains when mechanical discontinuities (such as gear backlash) are introduced to the system dynamics. Moreover, combining these methods with advanced model-based control algorithms, which are used for accurate path planning and validation of the computer simulations, may deteriorate the optimal performance in the presence of the system uncertainties and discontinuities. NMPC of the human-robot system seems to be a promising approach to optimizing robotic rehabilitation performance while considering safety [69, 101].

Similar to other model-based controllers (e.g., impedance controller, which is also used for human-environment interactions), NMPC requires a mathematical representation of the system. However, unlike the impedance controller, which requires an inverse dynamics model, the NMPC model has to be forward dynamic. A simpler model (fewer states) will decrease the computation cost of the NMPC controller. Such a simplified model requires dynamic system parameters tailored to the specific modeling needs. These model parameters can be obtained through a dynamic parameter identification (DPI) process.

A DPI process can be on-line (such as adaptive control and neural network methods) or off-line (physical measurements, computer software techniques and system input-output matching) [323]. System input-output matching, which is more accurate than the other off-line methods [323], can be considered as an optimization problem, in which the vector of unknown system parameters minimizes an objective function. If the parameters of the system are defined linearly in the dynamic system model, the DPI can be done by conventional least square or gradient methods [163]. In robotics, dynamic model parameters appear nonlinearly and cannot be found through conventional DPI methods. Thus, nonlinear DPI methods such as neural network [133, 263], homotopy optimization [305, 306] and nonlinear optimization approaches are being used [202, 267, 282, 329].

The neural network method considers the system as a black box; if the measurements are too noisy, the identified system dynamics will represent the actual dynamics convoluted

with the noise and will not produce repeatable results. Moreover, in this method, the relation between the outputs and inputs of the system should be continuous [12, 48]. Hence, if the system dynamics is altered by backlash, the resulting discontinuity cannot be estimated by the neural network approach.

Nonlinear optimization approaches are suitable for finding the best parameter estimates if the initial guesses are properly selected; otherwise, they may get stuck in local minima. This issue is more prominent in discontinuous and noisy systems. Homotopy optimization, however, usually converges to the global minimum in the DPI [305]. This method is used for identifying the parameters of a model represented by ODEs [305]. Thus, for a system dynamics with discontinuity, which will generate Differential-Algebraic Equation (DAE)s with conditional statements, conventional homotopy optimization cannot be used. Since in this method a forward dynamics problem is solved, filtering the measurement data is not required. However, for this kind of DPI, conventional optimization methods such as Sequential Quadratic Programming (SQP) may fail because of numerical issues and improper choice of initial guesses. With some modifications, which will be discussed in Chapter 3 (System Models: Rehabilitation Robot), homotopy optimization can handle these issues and provide better results than the SQP.

2.6 Concluding remarks

Upper extremity rehabilitation robots are improved by new hardware and control scenarios. New mechanical design and control strategies are currently evaluated after device development and clinical tests. Design and control are iterative approaches and thus, for better outcomes with multiple iterations, the possibility of verification without the construction of these robots becomes important. Since musculoskeletal models have been used successfully for the study of human movement, they are promising tools to interact with rehabilitation devices in simulations. In this study, we propose the use of musculoskeletal models with modified activation dynamics[5] interacting with rehabilitation robots to contribute to the controller design and development of a rehabilitation robot. Some other studies have used models for the evaluation of stroke patient therapy [90, 116, 233, 247]. However, in the current research, the goal is to utilize musculoskeletal models in real-time simulations for model-based design and control of rehabilitation robots.

In contrast to exoskeleton robots, it is hard to perform specific movements of distinct body joints during therapy with an end-effector based robot. However, exoskeleton robots

[5]Since the experiments are done on healthy subjects, activation dynamics has not been altered during this project.

are more complicated to set up than end-effector based robots. In the current study, we will try to overcome this deficiency of end-effector based robots by developing a subject-specific rehabilitation system. This system will move specific joints through desired joint angles since the musculoskeletal model of the subject is available, and subject movements can be predicted and planned.

To check the feasibility of this study, we will use an upper extremity rehabilitation manipulandum interacting with a musculoskeletal arm model. The robot was designed and developed by Quanser Consulting Inc. and the Toronto Rehabilitation Institute (TRI) [169] and modified in our lab to provide force sensing. Since two systems (robot and human) are studied, their models will be developed. Then, dynamic parameters of the musculoskeletal model for the specific subject and robot model will be identified using a proper DPI. During HRI, two types of proposed low-level control algorithms will be created, tuned, and evaluated. Next, these models will be used in new model-based control structures to conclude this study.

Chapter 3

System Models: Rehabilitation Robot

Our upper extremity rehabilitation robot was designed and developed by Quanser Consulting Inc. and the TRI and modified in our lab to provide force sensing. The system is expected to elicit motor recovery of the shoulder and elbow joints of post-stroke patients by repetitive programmed exercises such as reaching movements. This will improve motoneuron control, muscle strength, and range of motion in a systematic approach. The robot can also provide insight into the measurement of the aforementioned parameters.

In this chapter, first, an overview of the robot system will be given, focusing on its mechanical design and computer software. Then, based on the discussed mechanical issues of the robot, dynamic models of the robot using our two proposed DPI methods will be presented. Finally, concluding remarks will be provided.

3.1 Robot overview

3.1.1 Mechanical design

The robot is a 2 DOF parallelogram mechanism that operates in the horizontal plane (see Fig. 3.1.a). Two DC motors with optical encoders are connected to the parallelogram arm through two disc-and-timing-belt mechanisms (see Fig. 3.1.a). The motors' torque constant is $K_T = 0.115$ Nm/Amp, and the mechanical transformers' gear ratio is $r = 16 : 307$. The motors are rated at 115 mN-m of continuous torque. The continuous force at the hand

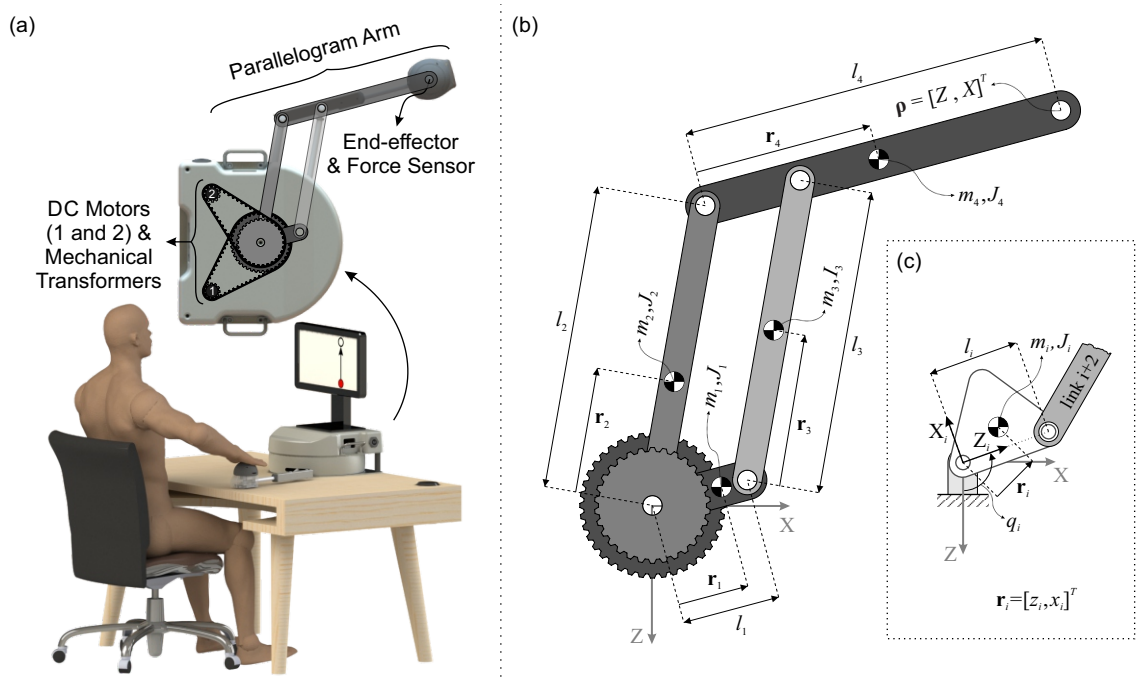


Figure 3.1: a) 2DOF upper extremity rehabilitation robot, b) Kinematics (i$^{\text{th}}$ link length: l_i) and inertial properties (i$^{\text{th}}$ link mass, inertia, and center of mass location: m_i, I_i, and $\mathbf{r}_i$) of the parallelogram arm, c) COM location ($\mathbf{r}_i$) and joint angle (q_i) definitions for the i$^{\text{th}}$ link in its local coordinate system $\{\mathrm{Z}_i, \mathrm{X}_i\}$ with respect to the global coordinate system $\{\mathrm{Z}, \mathrm{X}\}$.

(or end-effector) is limited to 13 N per plane of motion. The motor encoder resolution is 4000 count/revolution, which results in a sensitivity of 0.8 mm/count in detecting changes in the global coordinates. The motors connect to two aluminum links that are connected to a passive aluminum arm, and an aluminum outer arm. Link lengths are provided in Table 3.1. The outer arm is attached to an end-effector where the hand holds can be mounted. The geometry of the links determines the shape of the workspace. The robotic arm will be used in the horizontal plane with the robot firmly clamped to a table.

We have done some modifications to this stroke rehabilitation robot. In our modified version of the robot, the end-effector is equipped with a 6-axis Force/Torque (F/T) sensor (ATI Industrial Automation F/T Sensor: Nano25) that measures the interaction force between the robot and human (see Fig. 3.1.a). The F/T sensor has been calibrated to tolerate maximum 250 N on the horizontal plane and 1000 N normal to the plane. Sensor resolution is 1/24 N in the horizontal plane and 1/48 N normal to the plane. The robot's

Table 3.1: Link lengths of the robot linkage measured in mm.

l_1	l_2	l_3	l_4
100	310	310	375

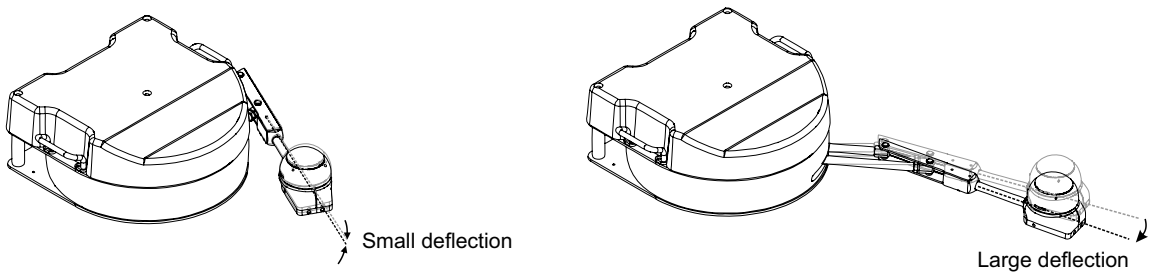


Figure 3.2: Robot out-of-plane arm deflection.

end-effector slides on four ball rollers (with Polyacetal balls), screwed into the end-effector and protruding from the bottom, over the table surface. The end-effector of the robot in the operational condition touches the table surface through the ball rollers, and this will generate friction between the robot and the table surface (see Fig. 3.1.a). The robot mechanism has some issues such as arm deflection in high-range motions (see Fig. 3.2), and this causes variable end-effector friction because of variable normal force caused by deflection. Actuator gear-belt backlash is the other problem in the robot mechanism; the backlash gap is measured by a motion tracking system ("Optotrak Certus") to be about $\beta = 0.005$ rad $= 0.286$ deg. It is worth noting that we are not allowed to modify the robot destructively. Thus, mounting an encoder to the load-side cannot be a solution for the backlash issue.

3.1.2 Computer software

The robot's computer software interface includes Simulink/MATLAB® which incorporates Quanser's real-time control software driver (QUARC). To control the robot, the driver software uses Quanser's Data Acquisition (DAQ) card (Q8). The driver and application software communicate through TCP/IP and shared memory protocol. To read the F/T sensor data, a National Instruments DAQ card (PCI-6229) is used. This card is compatible and operable by the QUARC software. The sampling-time frequency of the experiments is set to 500 Hz, and a fixed-step Euler solver is selected to solve the ordinary differential

equations.

3.2 Dynamic parameter identification using a modified homotopy method

Current DPI methods are more suitable for systems with continuous dynamics. If any discontinuity (e.g., backlash) is present in the system, the DPI may have numerical problems for convergence. In this section, we propose a modified homotopy optimization to identify parameters of a system with mechanical discontinuity (i.e., backlash). The performance of the proposed method was first evaluated through a computer simulation on a system with sandwiched backlash. Results of the DPI showed that the proposed homotopy optimization could identify the discontinuous system parameters with a good accuracy. It was found that ignoring the backlash in the system dynamics imposes large errors in the system DPI. After verifying the proposed method using computer simulations, the DPI was implemented to identify the parameters of a highly noisy and nonlinear rehabilitation robot with actuator backlash. The proposed method provided a better estimate of the system parameters compared to the no-backlash DPI of the experimental robot. Despite the angle measurement noise and its propagation in the velocity and acceleration calculations, the forward dynamics results are quite accurate for all of the tested configurations with the discontinuous backlash model.

3.2.1 Introduction

The objective of this section is to develop a method for the DPI of dynamic systems, which are geared by a sandwiched backlash (with no load-side state measurement) [199]. In this section, we will modify the homotopy optimization approach to identify discontinuous dynamic systems. To validate the proposed method, first, the method will be used to identify a virtual apparatus: the simulation of a 1 DOF mechanism with a sandwiched backlash (see Fig. 3.3). Second, we will identify a discontinuous dynamic system of an experimental apparatus: our 2 DOF upper extremity rehabilitation robot with a closed-chain mechanism (see Fig. 3.1.a,b). There are few publications on closed-chain robot's DPI, since: (1) their workspace is constrained and does not allow a wide range of experimental inputs for identification, and (2) it is hard to perform independent DPI of a single joint [323]. However, since the robot dynamics satisfies the property in which any of its driving joint dynamics can be identified by locking the other driving joint, the DPI is performed

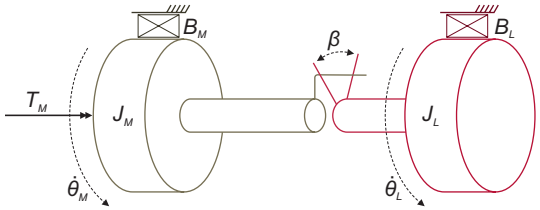


Figure 3.3: Virtual apparatus (1DOF mechanism with a sandwiched backlash), in which T_M and θ_M are the system input and output, respectively.

on a decoupled mechanism of the robot to identify parameters of a single DOF. Full DPI of the robot will be presented in Section 3.3 (Dynamic parameter identification using direct collocation method).

3.2.2 Method

In this subsection, first, our proposed modified homotopy optimization is presented. Then, the procedure to implement the homotopy optimization for the simulation and experimental setups are presented.

Conventional homotopy optimization

For a given ODE system:

$$\begin{cases} \dot{\mathbf{x}}_1 = \mathbf{x}_2 \\ \dot{\mathbf{x}}_2 = \mathbf{h}(\mathbf{x}_1, \mathbf{x}_2, \mathbf{p}, t) \end{cases} \tag{3.1}$$

the homotopy optimization method is implemented by coupling the experimental data with the mathematical model as follows:

$$\begin{cases} \dot{\mathbf{x}}_1 = \mathbf{x}_2 + (\mathbf{x}_{1e} - \mathbf{x}_1)\lambda\mathscr{K}_1 \\ \dot{\mathbf{x}}_2 = \mathbf{h}(\mathbf{x}_1, \mathbf{x}_2, \mathbf{p}, t) + (\mathbf{x}_{1e} - \mathbf{x}_1)\lambda\mathscr{K}_2 \end{cases} \tag{3.2}$$

where subscript e denotes the experimental data, and $\mathbf{x}_1(t) = [y_1(t), y_2(t), ..., y_n(t)]$ and $\mathbf{p}$ are the independent coordinates (displacements) and unknown parameter vector, respectively. λ is the observer gain adjuster, and $\mathscr{K}_1$ and $\mathscr{K}_2$ are the constant coupling gains. In (3.2), the coupling term act as a high-gain observer if $\lambda = 1$, and by decreasing λ, this observer gain will be decreased. For the homotopy optimization, the objective function is:

$$\mathfrak{V}(\mathbf{p}) = \frac{1}{2}\sum_{j=1}^{n}\int_0^{t_f}(y_{je}(t) - y_j(t, \mathbf{p}))^2 dt \tag{3.3}$$

where t_f is the final simulation time, and the states ($y_{1..n}$) are functions of time and parameters. While decreasing λ from 1 (dynamic system with a high-gain observer) to 0 (dynamic system with no-observer), the objective function (3.3) is minimized iteratively by updating $\mathbf{p}$ in a search direction ($\mathbf{s}$):

$$\mathbf{p}^{i+1} = \mathbf{p}^i + \varrho \mathbf{s} \tag{3.4}$$

where ϱ is the updating step size, superscripts i denotes the value at i$^{\text{th}}$ iteration, and $\mathbf{s}$ is evaluated by solving the following equation:

$$\mathbf{H}(\mathbf{p})\,\mathbf{s} = -\mathbf{G}^T(\mathbf{p}) \tag{3.5}$$

In (3.5), $\mathbf{G}$ and $\mathbf{H}$ are the gradient and the approximate Hessian of the objective function and defined as:

$$\mathbf{G}(\mathbf{p}) = \frac{\partial \mathfrak{V}}{\partial \mathbf{p}} = \sum_{j=1}^{n} \int_0^{t_f} \left(y_j(t,\mathbf{p}) - y_{je}(t)\right) \frac{\partial y_j}{\partial \mathbf{p}} dt \tag{3.6}$$

$$\mathbf{H}(\mathbf{p}) = \frac{\partial^2 \mathfrak{V}}{\partial \mathbf{p}^2} \approx \sum_{j=1}^{n} \int_0^{t_f} \frac{\partial y_j}{\partial \mathbf{p}}^T \frac{\partial y_j}{\partial \mathbf{p}} dt \tag{3.7}$$

Modified homotopy optimization

Consider a mechanically discontinuous 1 DOF dynamic system:

$$\begin{aligned} T_M &= J_M \ddot{\theta}_M + \Upsilon_M(\dot{\theta}_M, \theta_M, \mathbf{p}_M) + T_L \\ T_L &= J_L \ddot{\theta}_L + \Upsilon_L(\dot{\theta}_L, \theta_L, \mathbf{p}_L) \end{aligned} \tag{3.8}$$

where subscripts M and L denote motor- and load-side values, respectively. J is the inertia, T is the torque, and Υ is the summation of the dynamic terms excluding the inertial terms. Backlash in this dynamic system can be modeled using the discontinuous model presented in [280] (see Fig. 3.4) with the following equation:

$$\dot{\theta}_L = \begin{cases} \dot{\theta}_M & \dot{\theta}_M > 0 \text{ and } \theta_M \leq \theta_L - c_l \\ \dot{\theta}_M & \dot{\theta}_M < 0 \text{ and } \theta_M \geq \theta_L + c_r \\ 0 & \text{otherwise} \end{cases} \tag{3.9}$$

where $c_r > 0$ is the right "crossing" (positive angle) and $c_l > 0$ is the left "crossing" (negative angle). Since the backlash gap size (β) is assumed to be known, c_r can be written

as $\beta - c_l$. Thus, c_l is the only unknown backlash parameter that should be identified. The state variables of (3.8) are redefined as $\theta_M = y_1$, $\theta_L = y_2$, $\dot{\theta}_M = y_3$, $\dot{\theta}_L = y_4$, and $\ddot{\theta}_L = y_5$. Considering the system dynamics (3.8), the DAE system will be:

$$\begin{aligned}
\dot{y}_1 &= y_3 \\
\dot{y}_2 &= y_4 \\
\dot{y}_3 &= h(\mathbf{y}, \mathbf{p}, t) \\
\dot{y}_4 &= y_5 \\
y_4 &= \begin{cases} y_3 & y_3 > 0 \text{ and } y_1 \leq y_2 - p_1 \\ y_3 & y_3 < 0 \text{ and } y_1 \geq y_2 + \beta - p_1 \\ 0 & \text{otherwise} \end{cases}
\end{aligned} \tag{3.10}$$

where

$$h(\mathbf{y}, \mathbf{p}, t) = \frac{1}{p_2}\Big(T_M(t) - p_3 y_5 - \Upsilon_M(y_3, y_1, \mathbf{p}_M) - \Upsilon_L(y_4, y_2, \mathbf{p}_L)\Big) \tag{3.11}$$

and $\mathbf{p} = [c_l, J_M, J_L, \mathbf{p}_M^T, \mathbf{p}_L^T]^T$ is the parameter vector. In the conventional homotopy optimization, system dynamics is represented only by DAEs; however, discontinuous system dynamics has an algebraic equation alongside the DAEs. In our proposed modified homotopy approach, the algebraic equation will not have a coupling term; instead, an estimated experimental measurement will be introduced to the system to force the observer to consider discontinuous system response. This estimated experimental measurement, which is an estimate of the load-side rotation angle in terms of the motor-side angle, is defined as:

$$y_{2e}(t, \mathbf{p}) = \begin{cases} y_{1e} + p_1 & y_3 > 0 \text{ and } y_1 \leq y_2 - p_1 \\ y_{1e} - \beta + p_1 & y_3 < 0 \text{ and } y_1 \geq y_2 + \beta - p_1 \\ y_2 & \text{otherwise} \end{cases} \tag{3.12}$$

The modified homotopy optimization is solved by coupling the actual and estimated experimental data with the DAE system:

$$\begin{aligned}
\dot{y}_1 &= y_3 + (y_{1e} - y_1)\lambda\mathscr{K}_1 \\
\dot{y}_2 &= y_4 + (y_{2e} - y_2)\lambda\mathscr{K}_2 \\
\dot{y}_3 &= h(\mathbf{y}, \mathbf{p}, t) + (y_{1e} - y_1)\lambda\mathscr{K}_3 \\
\dot{y}_4 &= y_5 + (y_{2e} - y_2)\lambda\mathscr{K}_4 \\
y_4 &= \begin{cases} y_3 & y_3 > 0 \text{ and } y_1 \leq y_2 - p_1 \\ y_3 & y_3 < 0 \text{ and } y_1 \geq y_2 + \beta - p_1 \\ 0 & \text{otherwise} \end{cases}
\end{aligned} \tag{3.13}$$

In (3.13), different coupling gains $\mathscr{K}_{1..4}$ have been assigned for the differential equations; this flexibility will increase the control on observer performance, optimization sensitivity

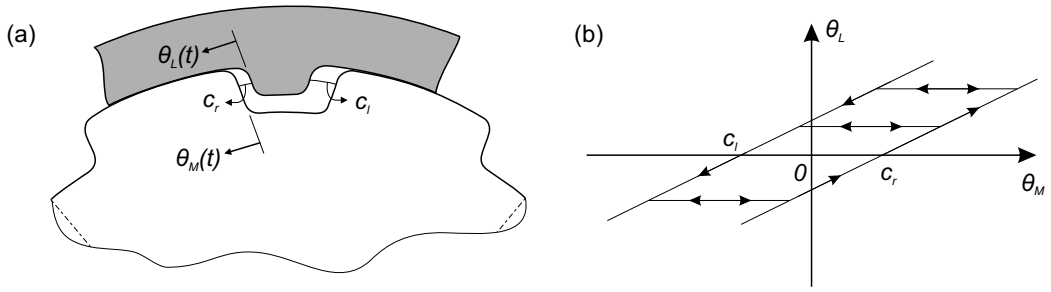


Figure 3.4: Backlash model for simulation and experimental setups. θ_M is the input and θ_L is the output.

and convergence. The objective function for the modified homotopy will be:

$$\mathfrak{V}(\mathbf{p}) = \frac{1}{2}\int_0^{t_f} \left[(y_{1e}(t) - y_1(t,\mathbf{p}))^2 + \Lambda(\lambda)(y_{2e}(t,\mathbf{p}) - y_2(t,\mathbf{p}))^2\right] dt \tag{3.14}$$

where $\Lambda(\lambda) \geq 0$, which is set to 1 throughout this study ($\Lambda(\lambda) = 1$), is the weighting factor for the estimated tracking. This function controls morphing the actual-estimated tracking onto the actual tracking (e.g., if $\Lambda(\lambda) = \lambda$, then by decreasing λ from 1 to 0, the objective function (3.13) will morph onto (3.2)). In contrast to the actual measurements, the estimated experimental measurement is a function of the time and system parameters. Thus, its derivatives will appear in the gradient and approximate Hessian evaluations as follows:

$$\mathbf{G}(\mathbf{p}) = \int_0^{t_f} \left[(y_1(t,\mathbf{p}) - y_{1e}(t))\frac{\partial y_1}{\partial \mathbf{p}} + (y_2(t,\mathbf{p}) - y_{2e}(t,\mathbf{p}))\left(\frac{\partial y_2}{\partial \mathbf{p}} - \frac{\partial y_{2e}}{\partial \mathbf{p}}\right)\right] dt \tag{3.15}$$

$$\mathbf{H}(\mathbf{p}) \approx \int_0^{t_f} \left[\frac{\partial y_1}{\partial \mathbf{p}}^T \frac{\partial y_1}{\partial \mathbf{p}} + \left(\frac{\partial y_2}{\partial \mathbf{p}} - \frac{\partial y_{2e}}{\partial \mathbf{p}}\right)^T \left(\frac{\partial y_2}{\partial \mathbf{p}} - \frac{\partial y_{2e}}{\partial \mathbf{p}}\right)\right] dt \tag{3.16}$$

To reduce the system noise propagation, the search direction equation (3.5) is solved using the following equation:

$$\mathbf{s} = -\mathbf{U}^+(\mathbf{p})\mathbf{G}^T(\mathbf{p}) \tag{3.17}$$

where $\mathbf{U}^+(\mathbf{p})$ is the pseudo-inverse of $\mathbf{U}(\mathbf{p})$, which is the upper triangular matrix of $\mathbf{H}(\mathbf{p})$:

$$U_{ij}(\mathbf{p}) = \begin{cases} H_{ij}(\mathbf{p}) & i \leq j \\ 0 & i > j \end{cases} \tag{3.18}$$

Note that the largest row of $\mathbf{U}(\mathbf{p})$ with non-zero elements (i.e., $i = 1$) should correspond to the backlash crossing. In other words, $p_1 = c_l$. To avoid system instability, the parameter update equation (3.4) is redefined as:

$$\mathbf{p}^{i+1} = \begin{cases} \mathbf{p}^i + \varrho\mathbf{s} & \mathbf{p}_{min} \leq \mathbf{p}^i + \varrho\mathbf{s} \leq \mathbf{p}_{max} \\ \mathbf{p}^i & \text{otherwise} \end{cases} \tag{3.19}$$

where $\mathbf{p}_{min}$ and $\mathbf{p}_{max}$ are the lower and upper bounds for the unknown parameters of the system, respectively.

Simulation setup

The simulation setup is a 1 DOF mechanism with a sandwiched backlash; this backlash is similar to the robot's backlash and modeled in (3.9) (see Fig. 3.4). The dynamic model of this setup is defined as:

$$\begin{aligned} T_M &= J_M\ddot{\theta}_M + B_M\dot{\theta}_M + T_L \\ T_L &= J_L\ddot{\theta}_L + B_L\dot{\theta}_L \end{aligned} \tag{3.20}$$

where B is the damping coefficient.

By redefining the state variables of (3.20) as $\theta_M = y_1$, $\theta_L = y_2$, $\dot{\theta}_M = y_3$, $\dot{\theta}_L = y_4$, and $\ddot{\theta}_L = y_5$, the DAE system will be the same as (3.10) with h defined as:

$$h(\mathbf{y}, \mathbf{p}, t) = \frac{1}{p_2}\left(T_M(t) - p_3y_5 - p_4y_3 - p_5y_4\right) \tag{3.21}$$

and the parameter vector will be $\mathbf{p} = [c_l, J_M, J_L, B_M, B_L]^T$. To implement the modified homotopy optimization, an estimated experimental measurement is considered as in (3.12). Then, the DAE system will be coupled with the actual and estimated experimental data in (3.13), and the homotopy optimization will be solved by minimizing the objective function (3.14), which is solved by updating the parameter vector iteratively using (3.17) and (3.19).

Experimental setup

Neglecting end-effector contact with the table surface and robot driving joint stiffness, the robot dynamics is governed by:

$$\mathbf{T} = \mathbf{M}_R(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}_R(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{f}_T + \mathbf{d} \tag{3.22}$$

where:

$$\mathbf{M}_R(\mathbf{q}) = \begin{bmatrix} M_{R1} & M_{R12}(\mathbf{q}) \\ M_{R12}(\mathbf{q}) & M_{R2} \end{bmatrix} \tag{3.23}$$

and

$$\mathbf{C}_R(\mathbf{q}, \dot{\mathbf{q}}) = \begin{bmatrix} 0 & \frac{\partial M_{R12}(\mathbf{q})}{\partial q_2}\dot{q}_2 \\ \frac{\partial M_{R12}(\mathbf{q})}{\partial q_1}\dot{q}_1 & 0 \end{bmatrix} \tag{3.24}$$

$\mathbf{T}$ is the transformed motors' torque vector, and $\mathbf{q} = [q_1, q_2]^T$ is the vector of robot joint angles when there is a zero-backlash in the system (see Fig. 3.1.c). $\mathbf{f}_T = [f_1, f_2]^T$ is the friction toque, which is caused by the timing belt pre-tensioning mechanism and the DC motor's friction. We have used a continuous velocity-based friction model to estimate the robot friction torque [34]. $\mathbf{d}$ is the unstructured (non-parametric) uncertainty which is originated from noise in measurements and robot end-effector deflection due to gravity.

Although locking the driving joints is not always possible for the closed-chain mechanism DPI, the structure of the dynamic equation (3.22) with the definitions in (3.23) and (3.24) guarantees that locking a driving joint will not affect the dynamics of the other joint. Thus, for simplicity in the DPI, one of the robot's driving joints is locked at different angles in the operational space. Once the robot inertia and friction parameters are identified for a single joint dynamics, the remaining inertia and centripetal terms will be identified when both joints are unlocked. This section only focuses on the first joint DPI; thus, subscript 1 is dropped from the forgoing equations. The complete dynamic equation of the robot with the locked second joint ($\dot{q}_2 = 0$) is:

$$\begin{aligned}
&T_M = J_M\ddot{\theta}_M + f_M + rT_L + d_M \\
&T_L = J_L\ddot{\theta}_L + f_L + d_L \\
&f_M = N_M\hat{f}_M \\
&\hat{f}_M = \hat{\mu}_{dM}\tanh\left(\frac{4\dot{\theta}_M}{\omega_{tM}}\right) + \frac{(\hat{\mu}_{sM}-\hat{\mu}_{dM})\dot{\theta}_M}{\left(\frac{1}{4}(\frac{\dot{\theta}_M}{\omega_{tM}})^2+\frac{3}{4}\right)^2\omega_{tM}} + \hat{\mu}_{vM}\dot{\theta}_M \\
&f_L = N_L\hat{f}_L \\
&\hat{f}_L = \hat{\mu}_{dL}\tanh\left(\frac{4\dot{\theta}_L}{\omega_{tL}}\right) + \frac{(\hat{\mu}_{sL}-\hat{\mu}_{dL})\dot{\theta}_L}{\left(\frac{1}{4}(\frac{\dot{\theta}_L}{\omega_{tL}})^2+\frac{3}{4}\right)^2\omega_{tL}} + \hat{\mu}_{vL}\dot{\theta}_L \\
&\dot{\theta}_L = \begin{cases} r\dot{\theta}_M & \dot{\theta}_M > 0 \text{ and } r\theta_M \le \theta_L - c_l \\ r\dot{\theta}_M & \dot{\theta}_M < 0 \text{ and } r\theta_M \ge \theta_L + c_r \\ 0 & \text{otherwise} \end{cases}
\end{aligned} \tag{3.25}$$

where the subscripts M and L denote the motor- and load-side values, respectively. f is the joint friction and is modeled as a summation of the static, dynamic and viscous

frictions (caused by pre-tensioning mechanism, joint bearings and the DC motors). The constant weighting factor parameter (N), which resembles the normal contact force, varies at different angles of the second joint. The motor-side unit friction ($\hat{f}_M$) is a summation of the static, dynamic and viscous frictions (caused by pre-tensioning mechanism and the DC motor). The load-side unit friction ($\hat{f}_L$) is modeled as a dynamic friction. $\hat{\mu}_v$, $\hat{\mu}_s$ and $\hat{\mu}_d$ are the viscous, static and dynamic friction terms, respectively, and ω_t is the transition velocity for stiction. d, which is the unstructured (non-parametric) uncertainty in the dynamic equation, is assumed to have less significance in the DPI; hence, we are neglecting it in the following equations.

If the state variables of (3.25) are redefined as $r\theta_M = y_1$, $\theta_L = y_2$, $r\dot{\theta}_M = y_3$, $\dot{\theta}_L = y_4$, and $\ddot{\theta}_L = y_5$, the DAE system will be the same as (3.10) with h defined as:

$$h(\mathbf{y}, \mathbf{p}, t) = \frac{1}{p_2}\Bigg(\frac{T_M(t)}{r} - p_3 y_5 - p_{12} p_4 \tanh\left(\frac{4y_3}{p_7}\right) - \frac{(p_6 - p_5)p_{12}y_3}{\left(\frac{1}{4}(\frac{y_3}{p_7})^2 + \frac{3}{4}\right)^2 p_7} - p_{12}p_7 y_3$$
$$- p_{13}p_8 \tanh\left(\frac{4y_4}{p_{11}}\right) - \frac{(p_9 - p_8)p_{13}y_4}{\left(\frac{1}{4}(\frac{y_4}{p_{11}})^2 + \frac{3}{4}\right)^2 p_{11}} - p_{13}p_{10}y_4\Bigg) \tag{3.26}$$

and the parameter vector will be $\mathbf{p} = [c_l, \frac{J_M}{r^2}, J_L, \frac{\hat{\mu}_{dM}}{r^2}, \frac{\hat{\mu}_{sM}}{r^2}, \frac{\hat{\mu}_{vM}}{r^2}, \omega_{tM}, \hat{\mu}_{dL}, \hat{\mu}_{sL}, \hat{\mu}_{vL}, \omega_{tL}, N_M, N_L]^T$. To implement the modified homotopy optimization, the same procedure as before is used. However, since the robot will be fixed at different angles of the second joint, the number of experimental measurements will increase. Consequently, the number of DAE systems coupled with the actual and estimated experimental data in (3.13) will increase based on the number of experiments ($\mathcal{N}$). For the robot, backlash crossing, and motor and load-side weighting factors (N) vary at different angles of the second joint; thus, c_l^k and $\mathbf{p}^k = [N_M^k, N_L^k]^T$ will be the crossing parameter and friction-deflection parameter vector for the k$^{\text{th}}$ experiment, respectively. Subsequently, by renaming y as y^k, $\mathfrak{V}$ as $\mathfrak{V}^k$ and defining the new parameter vector as $\bar{\mathbf{p}} = [c_l^{1..\mathcal{N}}, \frac{J_M}{r^2}, J_L, \frac{\hat{\mu}_{dM}}{r^2}, \frac{\hat{\mu}_{sM}}{r^2}, \frac{\hat{\mu}_{vM}}{r^2}, \omega_{tM}, \hat{\mu}_{dL}, \omega_{tL}\ , (\mathbf{p}^{1..\mathcal{N}})^T]^T$ in (3.12), (3.13) and (3.14), the global homotopy objective function for the combination of $\mathcal{N}$ experiments will be defined as:

$$\mathfrak{V}(\bar{\mathbf{p}}) = \sum_{k=1}^{\mathcal{N}} w_k \mathfrak{V}^k(\bar{\mathbf{p}}) \tag{3.27}$$

where w_k is the k$^{\text{th}}$ objective function weight. Finally, the gradient and approximate Hessian will be updated accordingly.

3.2.3 Simulation and experiment

Simulation setup

The MapleSim$^{\text{TM}}$software package was used to make the simulation setup model with the parameters defined in Table 3.3 in two cases. The motor-side mass was driven by a DC motor with a persistently exciting input current defined as:

$$I^c(t) = I_0 e^{-\Omega_0 t} \sum_{j=1}^{16} \sin(\Omega_j t) \tag{3.28}$$

to generate periodic excitations ($\Omega_{1..16}$ = [0.5, 1, 1.6, 2.5, 3, 4, 5, 6, 8, 10, 12, 16, 20, 25, 30] rad/s) with decaying amplitude (I_0 = 0.04 Amp, Ω_0 = 0.01 Hz). The output motor torque is evaluated from:

$$T_M(t) = K_T I^c(t) \tag{3.29}$$

where K_T is the motor torque constant (in Case A, K_T = 1 Nm/A, and in Case B, K_T = 0.115 Nm/A). While running the setup with the persistently exciting torque for $t_f = 4\pi$ s, the motor-side joint angle was measured with a position sensor to record the simulated measurements (y_{1e}).

For the homotopy optimization, the DAE system of the simulation setup (3.10) with the definition of h in (3.21) is coupled with the simulated (y_{1e}) and estimated (y_{2e} from (3.12) with β = 0.005 rad = 0.286 deg) measurements. Using MapleSim$^{\text{TM}}$an optimized C-code of (3.10), which evaluates the objective function (3.14), gradient (3.15), and approximate Hessian (3.16) values for the entire simulation, with a fixed-step (2 ms) Euler solver is exported. Next, the objective function is minimized by iteratively updating system parameters (3.19) in the search direction (3.17). Lower and upper bounds of parameters are defined in Table 3.3, in which ϵ is the floating-point relative accuracy in MATLAB®. See Table 3.2 for the assigned homotopy optimization values in two cases of simulation setup.

To investigate the effect of backlash on the DPI performance, the modified homotopy results ($\beta > 0$ in Table 3.3) are compared against a conventional homotopy optimization that does not consider the backlash dynamics ($\beta = 0$ in Table 3.3) in identifying this discontinuous system. Moreover, the performance of the modified homotopy optimization is compared to a general optimization problem solver (i.e., SQP) using MATLAB® Global Search algorithm. The SQP solver considers the DAE system (3.8) and the dynamics of (3.21) in two conditions: 1. the objective function only considers actual measurements with the cost in (3.3) (SQP I in Table 3.3), and 2. the objective function accounts for the estimated measurements (3.12) using the cost in (3.14) (SQP II in Table 3.3).

Table 3.2: Simulation setup's assigned parameters

	Motor and Backlash		Homotopy Optimization				
Parameter	K_T (Nm/A)	β (rad)	$\mathscr{K}_1$	$\mathscr{K}_2$	$\mathscr{K}_3$	$\mathscr{K}_4$	ϱ
Case A	1	0.005	10	10	100	10	0.1
Case B	0.115	0.005	10	10	10	10	0.1

Experimental setup

For the DPI of the rehabilitation robot, a special fixture is designed to lock the second joint (q_2) at different angles (see Fig. 3.5). Then the robot's first joint (q_1) is driven by the persistently exciting current defined in (3.28). The output motor torque is evaluated from (3.29) with $K_T = 0.115$ Nm/A. The encoder values of the robot joints are recorded with a sampling rate of 500 Hz. Dashed lines in Fig. 3.5 show the end-effector position while running the experiments at 23 different angles of the second joint (from $-10°$ to $210°$ with $10°$ increment).

For the homotopy optimization, the dynamic model of the system (3.26) coupled with the actual and estimated measurements (3.12) of the k^{th} experiment is developed as DAEs (3.13) in MapleSim$^{\mathrm{TM}}$. Next, all the experiments ($m = 23$) are combined, and the objective function is updated as (3.27). Then, the same procedure as in the simulation setup is followed. Among different experiments, the trajectories at $q_2 = \{0°, 70°, 140°, 210°\}$ (black dashed lines in Fig. 3.5) are selected for reporting the homotopy optimization results.

To assess the improvement of the discontinuous model over the no-backlash model, the DPI of the robot is done by both homotopy methods ($\beta = 0$ and $\beta > 0$). Finally, the results of the forward dynamics simulation of the two models are compared and discussed.

3.2.4 Results and discussion

Simulation

The simulation setup is evaluated with different model parameters in two cases which are shown in Table 3.3. Results of the DPI for these two cases are presented in Table 3.3. With the no-backlash DPI (homotopy with $\beta = 0$), the identified apparent inertia is between the summation and minimum value of the motor- and load-side inertial values; i.e., in Case A: $\min(0.01, 0.02) \leq 0.0278 \leq (0.02 + 0.01)$, and in Case B: $\min(0.01, 0.03) \leq 0.0298 \leq$

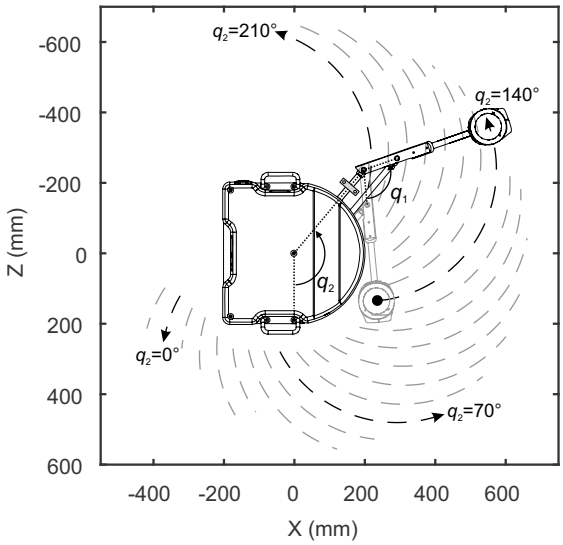


Figure 3.5: Robot experiment setup and paths (dashed lines), where q_2 is fixed at different angles from $-10°$ to $210°$ with $10°$ increment and q_1 is driven to produce paths.

$(0.03 + 0.01)$. Moreover, the identified apparent damping is close to the summation of the load- and motor-side damping coefficients. Considering these rules of thumb, we can expect to get similar behavior in the DPI of the experimental setup. Since of accounts for the estimated measurement in the SQP II objective function, the DPI using SQP II is more reliable than SQP I. The identified inertial values for both SQP methods are quite consistent with the model inertial values. In Case A, SQP II could identify the backlash crossing and damping values within an acceptable degree of accuracy, while this accuracy is not attained in Case B. Thus, SQP II is less successful than the modified homotopy method when a mechanical discontinuity is present in the system. In Section A.2, the performance of the modified homotopy optimization is also compared to other optimization solver results.

System response, i.e., forward dynamic simulations of the system with the identified parameters, are compared against the measurements for the two cases, and the Root-Mean Square Error (RMSE) of the motor-side joint angles and velocities are reported in Fig. 3.6. The homotopy with no backlash results in large angular position and velocity errors, which proves the importance of backlash modeling in the discontinuous system DPI. The angular position and velocity errors for the modified homotopy with backlash are the smallest, and SQP II is more accurate than SQP I. Comparing results of Case B with Case A, one can

Table 3.3: Simulation setup model parameters and the identified values in two cases. Four DPI methods are compared: SQP I, II, and homotopy $\beta = 0$, $\beta > 0$

					Case A Values					Case B Values				
			Bounds		Model	SQP		Homotopy		Model	SQP		Homotopy	
$\mathbf{p}$	Unit	$\mathbf{p}^0$	$\mathbf{p}_{min}$	$\mathbf{p}_{max}$	A	I	II	$\beta = 0$	$\beta > 0$	B	I	II	$\beta = 0$	$\beta > 0$
c_l	rad	0.0	0	β	0.0035	0.0000	0.0033	—	0.0035	0.0025	5.8e−4	0.0015	—	0.0025
J_M	$\mathrm{kg\,m^2}$	0.5	ϵ	1.000	0.0100	0.0084	0.0095	—	0.0100	0.0300	0.0302	0.0330	—	0.0301
J_L	$\mathrm{kg\,m^2}$	0.5	ϵ	1.000	0.0200	0.0184	0.0196	0.0278	0.0199	0.0100	0.0107	0.0117	0.0298	0.0102
B_M	Nms/rad	0.5	ϵ	1.000	0.1000	0.2803	0.1221	—	0.0915	0.2500	0.1864	0.1792	—	0.2468
B_L	Nms/rad	0.5	ϵ	1.000	0.2000	0.0212	0.1785	0.2924	0.2086	0.1500	0.2185	0.2236	0.4367	0.1535

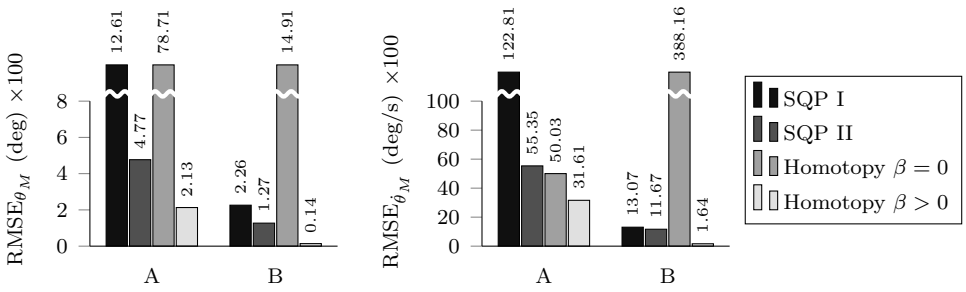


Figure 3.6: RMSE of the motor-side joint angle (chart on left) and velocity (chart on right) from the forward dynamics simulation of the identified models.

conclude that presence of the backlash in a dynamical system may cause large errors once the system input torque increases.

Experiment

The identified robot parameters for four configurations (q_2 is locked at 0°, 70°, 140° and 210°) using continuous (homotopy with $\beta = 0$) and discontinuous (homotopy with $\beta > 0$) models are reported in Table 3.4. Similar to the simulation setup, the identified apparent inertia for $\beta = 0$ is between the summation and minimum value of the motor- and load-side values (i.e., $\min(0.0208, 0.0994) \leq 0.1171 \leq (0.0208 + 0.0994)$. The identified friction parameters for the discontinuous model show that there is no dynamic friction (9.6e−4 and 2.9e−6 are negligible) in the system. On the motor-side, the friction is caused by

Table 3.4: Identified homotopy parameters of the robot at different configurations. $\mathfrak{f}_M = [\frac{\hat{\mu}_{dM}}{r^2}, \frac{\hat{\mu}_{sM}}{r^2}, \frac{\hat{\mu}_{vM}}{r^2}, \omega_{tM}]$ and $\mathfrak{f}_L = [\hat{\mu}_{dL}, \hat{\mu}_{sL}, \hat{\mu}_{vL}, \omega_{tL}]$ are the motor- and load-side friction parameters, respectively. These friction parameters has following vector of units $\mathfrak{U}_\mathfrak{f}$ = [Nm, Nm, Nms/rad, rad/s].

		Homotopy $\beta = 0$				Homotopy $\beta > 0$			
$\mathbf{p}$	Unit	$q_2 = 0°$	$q_2 = 70°$	$q_2 = 140°$	$q_2 = 210°$	$q_2 = 0°$	$q_2 = 70°$	$q_2 = 140°$	$q_2 = 210°$
c_l	rad	—				0.0000	0.0049	0.0042	0.0000
$\frac{J_M}{r^2}$	$\mathrm{kg\,m^2}$	—				0.0208			
J_L	$\mathrm{kg\,m^2}$	0.1171				0.0994			
$\mathfrak{f}_M$	$\mathfrak{U}_\mathfrak{f}$	—				[9.6e−4, 0.5779, 0.1123, 0.0339]			
$\mathfrak{f}_L$	$\mathfrak{U}_\mathfrak{f}$	[0.2341, 0.4065, 0.1591, 0.0164]				[2.9e−6, 0.5551, 6.9e−5, 0.4093]			
N_M	—	—				0.8941	0.6115	0.6915	0.9100
N_L	—	1.5125	1.0812	0.9367	0.9927	1.0154	0.6406	0.5938	0.6039

static and viscous friction, while on the load-side, the friction is caused by static friction (the viscous friction with 6.9e−5 coefficient is negligible) with a relatively high transition velocity ($\approx$ 23.5 deg/s). Unmodeled dynamics of the timing belt and backlash have led to this high transition velocity. The identified friction parameters for the continuous model have dynamic, static and viscous terms that are physically meaningful. However, for the continuous model, if the friction parameters are chosen to be different for different configurations, then the dynamic friction will become higher than the static friction. This phenomenon is because of the robot's backlash.

Forward dynamics simulation results of the robot model using the identified parameters for the continuous (no-backlash) and discontinuous models is presented in Fig. 3.7. In this figure, angular positions of the motor-side rotor are plotted along with the experimental values. RMSE of the system identification for the continuous and discontinuous models are reported in Fig. 3.8. For the no-backlash model, the average angular position RMSE (4.87 deg) is about thrice the value (1.49 deg) with the discontinuous model. Although modeling the backlash has decreased the position error, the average angular position error is five times bigger than the backlash gap size (0.005 rad). For the continuous and discontinuous models, the average angular speed RMSEs are 3.87 deg/s and 2.92 deg/s, respectively. Despite the great difference between angular position errors, these velocity

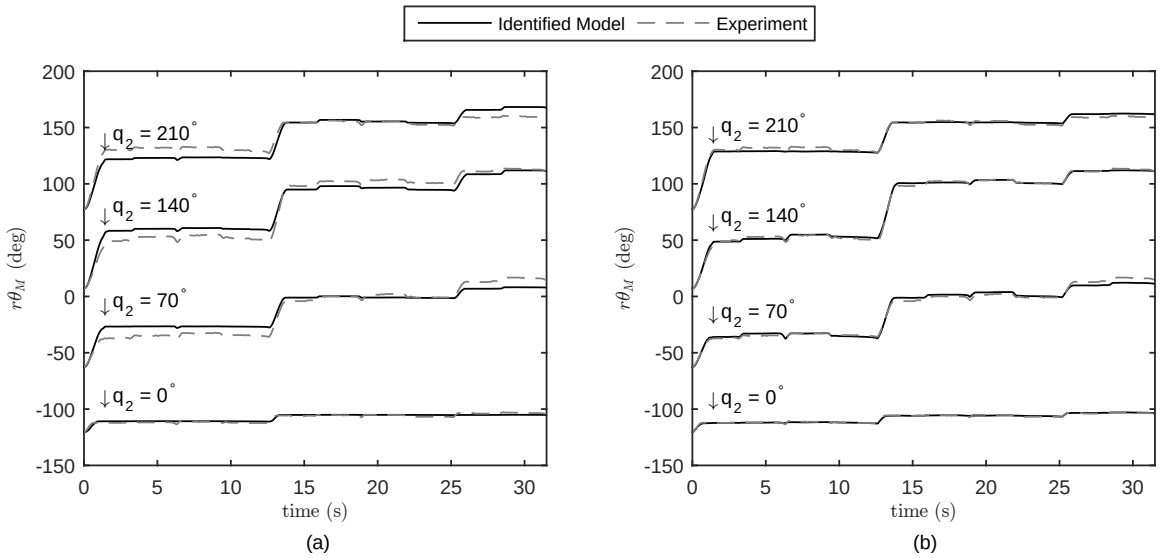


Figure 3.7: Forward dynamics results in the experimental setup. (a) the continuous model DPI (homotopy with $\beta = 0$), and (b) the discontinuous model DPI (homotopy with $\beta > 0$). The plots show the four configurations of q_2.

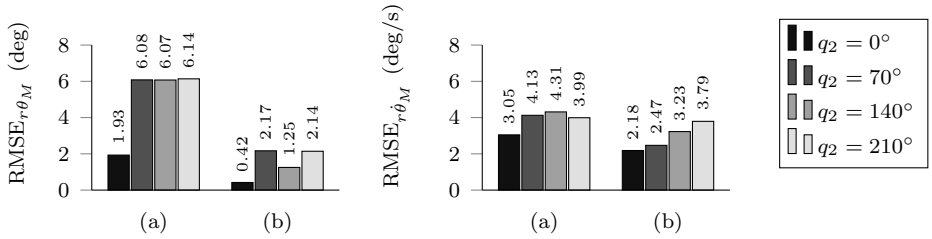


Figure 3.8: RMSE of the transformed motor-side joint angle (chart on left) and velocity (chart on right) from the forward dynamics simulation of the identified models.

errors are close to each other. Thus, despite the decrease in the system response error by including backlash dynamics, unknown dynamics affect the system behavior, and it is not negligible. For example, if we used variable friction parameters for different configurations, the average errors for the continuous and discontinuous models decrease to 1.56 deg and 0.68 deg, respectively. Then the average angular position error (0.68 deg) would be about twice the backlash gap size.

For the discontinuous model, since the angular position errors are small, the perform-

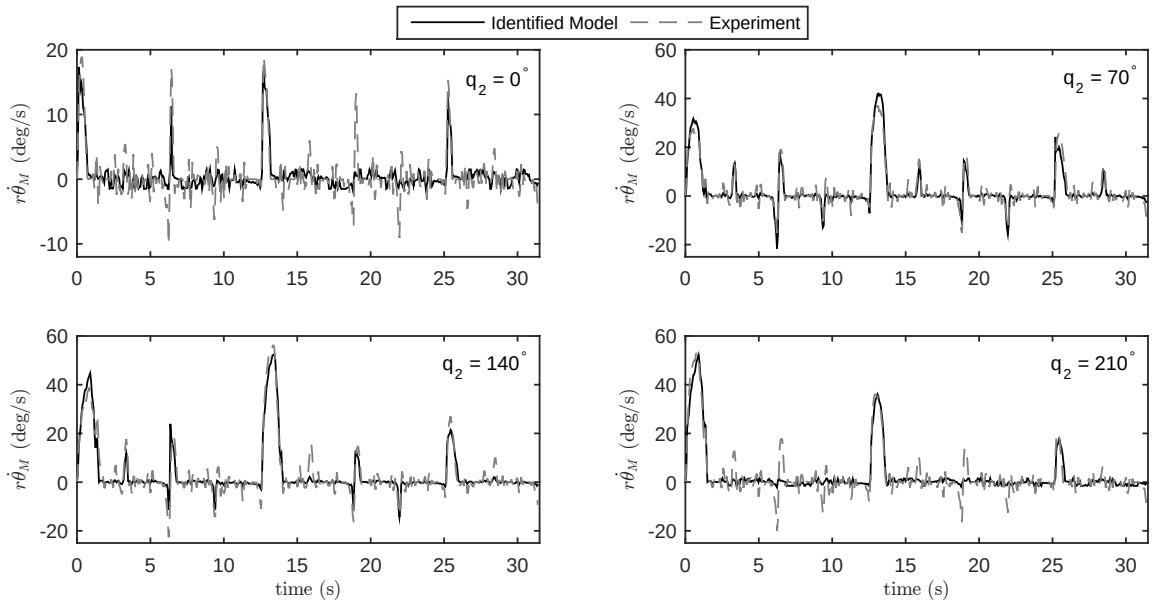


Figure 3.9: Forward dynamics simulation (transformed motor-side angular speed) of the discontinuous ($\beta > 0$) model in four configurations.

ance of the modified homotopy is assessed through the angular speed plots in Fig. 3.9. This figure shows that for different configurations, the simulated model velocity is close to the noise-free parts of the experimental velocity values. In other words, the simulated results are the low-pass filtered values of the experimental measurements for most of the simulation. Noisy IO in the robot DPI affects the SQP methods' global search algorithm such that they move around in a local minimum. Thus, because of the poor performance of the SQP methods, their results are not presented.

Since the objective of this work was to develop a forward dynamic model of the robot to use with model-based controllers, the convergence of the forward dynamic optimization results proves the applicability of the proposed DPI method. Despite the convergence of the forward dynamics results, the driving current resulting from the inverse dynamics simulation does not match the original current input in some instances. The reasons can be listed as: (1) the presence of the stiction friction: in inverse dynamics simulation, the stiction friction generates maximum allowable current that holds the system stationary, while in the experiments this current can be any value less than the maximum allowable current that does not generate motion; (2) the homotopy method uses the forward dynamics model for convergence; therefore, inaccuracy in the inverse dynamics results because of

numerical issues affecting forward dynamics results are more probable.

Other possible reasons for errors in system identification are as follows:

- Neglecting unknown dynamics caused by gravitational deflection (neglecting d_M and d_L in (3.25)).
- Encoder measurement noise.
- Ignoring the unknown motor dynamics and using a simple DC motor model to get the output torque by the motor torque constant gain.
- Because of the timing belt elasticity, the backlash gap is different for different configurations. The belt's dynamics are neglected.

3.2.5 Conclusions

In this section, we proposed a modified homotopy optimization to identify parameters of the system dynamics with mechanical discontinuity (i.e., backlash). To evaluate the performance of the method in the DPI of discontinuous systems, a computer simulation setup was developed, and the proposed method along with other optimization methods was used to identify the parameters of the setup in two cases. Results of the DPI showed that the proposed homotopy optimization could identify the discontinuous system parameters with a good accuracy. The results of the DPI with the modified homotopy was also compared to the conventional homotopy results, in which the system was considered as a continuous model. It was concluded that ignoring backlash in the system dynamics could impose large errors in the system DPI.

After verifying the proposed method using computer simulations, the method was used to identify the parameters of the 2 DOF noisy and nonlinear rehabilitation robot with actuator backlash, when the robot's second joint was locked in different configurations. The proposed method provided better estimates for the system parameters compared to the no-backlash DPI of the robot. Despite the angle measurement noise and its propagation to the velocity and acceleration evaluations of the model, forward dynamics results were quite accurate for all of the configurations with the discontinuous backlash model. We believe that the modified homotopy optimization can be extended to consider other common mechanical discontinuities such as dead-zone and hysteresis. In the next section a complete DPI of the robot, in which the unknown dynamics of the system will also be included, will be presented.

3.3 Dynamic parameter identification using direct collocation method

In the previous section, a modified homotopy approach for the DPI of mechanically discontinuous systems was provided. The proposed dynamic model for the rehabilitation robot was discontinuous (with no load-side state measurement), and the user input (i.e., end-effector movement by the patient) could change the dynamics of the robot while having no information from the load-side angle. Implementing this dynamic model on the robot will cause critical uncertainties and instabilities, since the load-side state can be altered by the user input. Thus, we should develop a continuous model to overcome this uncertainty.

In this section, we present a direct collocation method, which was implemented using the GPOPS-II optimal control software package, for the DPI of the noisy upper extremity rehabilitation robot with sandwiched backlash. Different identification tests were performed, and the results of the proposed method were verified. The robot dynamics was modeled using two different friction models, LuGre and continuous-velocity models, and their results were compared. Finally, the robot experimentally was controlled to follow a desired trajectory using the identified model in a sliding-mode controller structure, and the results verified the accuracy of the robot model.

3.3.1 Introduction

The studied rehabilitation robot is a closed chain mechanism with noisy IO. Consequently, conventional DPI methods may not be able to solve the robot's noisy optimization problem. The objective of this work is to develop a method for the DPI of closed chain robots with noisy inputs and outputs. The significance of this work is introducing a direct collocation method in a reliable and structured framework for the DPI of highly noisy, large scale, mechanically discontinuous and nonlinear systems with friction. In this section, IO matching is stated as an optimal control problem, which minimizes the error between the estimated noise-free torque and the experimental input torque while satisfying the robot dynamics and trajectory tracking constraints. To this end, we present a procedure to perform IO matching on the robot's closed chain mechanism. It starts with the development of a forward dynamics model of the robot with a suitable friction model. Then, a direct collocation method from optimal control theory is used to solve the robot's DPI problem. Finally, the method is verified by performing extra experiments on the robot with the identified parameters. The proposed method is capable of solving the DPI problem of the highly nonlinear noisy systems with an acceptable accuracy.

3.3.2 Method

In this section, the robot dynamics will be presented, followed by the procedure for the IO matching. Then, the proposed DPI method will be discussed.

Robot dynamics

The updated robot dynamic equation considering end-effector friction and robot driving joint stiffness (when the DC motors' dynamics are transformed to the parallelogram arm) is:

$$\mathbf{T} = \mathbf{M}_R(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}_R(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{K}_R(\mathbf{q}-\mathbf{q}_0) + \mathbf{f}_T + \mathbf{J}_R^T\mathbf{f}_F + \mathbf{d} \tag{3.30}$$

$\mathbf{T}$ is evaluated as $K_T\mathbf{I}^c/r$ in terms of the motors' current vector ($\mathbf{I}^c$). $\mathbf{K}_R$ is a 2×2 symmetric stiffness matrix (number of stiffness parameters is $\boldsymbol{\imath}_{\mathrm{S}} = 3$) that models joint stiffness caused by timing belt dynamics, $\mathbf{q}_0$ is the equilibrium position vector (number of equilibrium angle parameters is $\boldsymbol{\imath}_{\mathrm{E}} = 2$), $\mathbf{f}_F$ is the end-effector friction force, and $\mathbf{J}_R$ is the geometric Jacobian. The robot's inertia, $\mathbf{M}_R(\mathbf{q})$, and Coriolis-centripetal matrix, $\mathbf{C}_R(\mathbf{q},\dot{\mathbf{q}})$, are redefined as:

$$\mathbf{M}_R(\mathbf{q}) = \begin{bmatrix} p_1 & p_2\mathrm{c}_{12} + p_3\mathrm{s}_{12} \\ p_2\mathrm{c}_{12} + p_3\mathrm{s}_{12} & p_4 \end{bmatrix} \tag{3.31}$$

and

$$\mathbf{C}_R(\mathbf{q},\dot{\mathbf{q}}) = \begin{bmatrix} 0 & (p_2\mathrm{s}_{12} - p_3\mathrm{c}_{12})\dot{q}_2 \\ (p_3\mathrm{c}_{12} - p_2\mathrm{s}_{12})\dot{q}_1 & 0 \end{bmatrix} \tag{3.32}$$

in which, c_{12} and s_{12} stand for $\cos(q_1-q_2)$ and $\sin(q_1-q_2)$, respectively. $p_{1..4}$ are the minimal set of observable inertial parameters (number of inertial parameters is $\boldsymbol{\imath}_{\mathrm{I}} = 4$); considering Fig. 3.1.b, they are defined as:

$$\begin{cases} p_1 = (x_1^2+z_1^2)m_1 + m_3l_1^2 + (x_4^2+z_4^2)m_4 + J_1 + J_4 \\ p_2 = m_3l_1z_3 + m_4l_2z_4 \\ p_3 = m_3l_1x_3 - m_4l_2x_4 \\ p_4 = (x_2^2+z_2^2)m_2 + m_4l_2^2 + (x_3^2+z_3^2)m_3 + J_2 + J_3 \end{cases} \tag{3.33}$$

Although the normal contact forces change with the robot configuration, the friction forces are modeled as weighted normalized frictions, in which the weighting factors are constant parameters. These frictions are defined as:

$$\begin{cases} \mathbf{f}_T = \mathbf{N}_T \circ \hat{\mathbf{f}}_T \\ \mathbf{f}_F = N_3\hat{\mathbf{f}}_F \end{cases} \tag{3.34}$$

where $\circ$ is the Hadamard (entry-wise) product. Values of the joints' weighting factors ($\mathbf{N}_T = [N_1, N_2]^T$) are always equal to 1. End-effector's weighting factor (N_3) is dependent on the applied normal force to the end-effector; it is equal to 1 when the robot operates with no load on the end-effector.

Normalized friction forces ($\hat{\mathbf{f}}_T = [\hat{f}_1, \hat{f}_2]^T$ and $\hat{\mathbf{f}}_F$) are modeled by following two approaches. In the first approach (A1), a LuGre friction model is implemented. LuGre friction is a general-purpose dynamic friction model for robotic applications that combines the stiction and Stribeck effects with viscous friction [204]. This model introduces additional states (average bristle deflections) and parameters to the problem. Here are the corresponding friction equations from this approach:

1. Normalized friction in the joints ($i = 1, 2$)

$$\begin{cases} \hat{f}_i = \hat{\sigma}_{0i}\xi_i + \hat{\sigma}_{1i}\dot{\xi}_i \mathrm{e}^{-(\frac{\omega_i}{\omega_{di}})^2} + \hat{\mu}_{vi}\omega_i \\ \dot{\xi}_i = q_i - \hat{\sigma}_{0i}\frac{|\omega_i|}{g_i(\omega_i)}\xi_i \\ g_i(\omega_i) = \hat{\mu}_{ci} + \hat{\mu}_{\text{Ⓢ}i}\mathrm{e}^{-(\frac{\omega_i}{\omega_{0i}})^2} \end{cases} \tag{3.35}$$

2. Normalized friction in the end-effector ($i = 3$)

$$\begin{cases} \hat{\mathbf{f}}_F = \hat{\sigma}_{0i}\boldsymbol{\mathfrak{z}} + \hat{\sigma}_{1i}\dot{\boldsymbol{\mathfrak{z}}}\mathrm{e}^{-(\frac{v}{v_d})^2} \\ \dot{\boldsymbol{\mathfrak{z}}} = \mathbf{v} - \frac{\hat{\sigma}_{0i}}{g_i(v)}|\mathbf{v}| \circ \boldsymbol{\mathfrak{z}} \\ g_i(v) = \hat{\mu}_{ci} + \hat{\mu}_{\text{Ⓢ}i}\mathrm{e}^{-(\frac{v}{v_0})^2} \end{cases} \tag{3.36}$$

where ξ and $\boldsymbol{\mathfrak{z}}$ denote the average rotational and vectorized translational bristle deflections, $\hat{\sigma}_0$ and $\hat{\sigma}_1$ are the bristle stiffness and damping coefficients, and ω_d and v_d are the dissipative angular and linear velocities, respectively. Function $g(.)$ models the Stribeck effect, where the lower and upper bounds of $g(.)$ are $\hat{\mu}_c$, which is Coulomb friction, and $\hat{\mu}_c + \hat{\mu}_{\text{Ⓢ}}$, which corresponds to stiction. ω_0 and v_0 are the angular and linear velocities that shape the function g within its bounds. $\hat{\mu}_v$ is the viscous friction coefficient, and $\omega_i = \dot{q}_i$ is the angular speed of joint i. $\mathbf{v} = \dot{\boldsymbol{\rho}} = [\dot{Z}, \dot{X}]^T$ is the end-effector velocity in the global coordinate system (see Fig. 3.1.b), and $v = \|\mathbf{v}\|$ is the Euclidean norm of this velocity. Since there is no lubrication underneath the end-effector, no viscous friction has been considered for the end-effector normalized friction ($\hat{\mathbf{f}}_F$).

In A1, the additional states ($\xi_{1,2}$ and $\boldsymbol{\mathfrak{z}}$) and the large number of normalized friction parameters ($\boldsymbol{\mathfrak{n}}_{\mathrm{F(A1)}} = 2\times 7+6 = 20$ parameters in total) will increase the computation cost both for the DPI and NMPC implementation. Thus, as an alternative approach (A2), a continuous velocity-based friction [34] is used to model the normalized frictions. Following this approach, the corresponding friction equations are as follows:

1. Normalized friction in the joints ($i = 1, 2$)

$$\hat{f}_i = \hat{\mu}_{di} \tanh\left(\frac{4\omega_i}{\omega_{ti}}\right) + \frac{(\hat{\mu}_{si} - \hat{\mu}_{di})\omega_i}{\left(\frac{1}{4}\left(\frac{\omega_i}{\omega_{ti}}\right)^2 + \frac{3}{4}\right)^2 \omega_{ti}} + \hat{\mu}_{vi}\omega_i \tag{3.37}$$

2. Normalized friction in the end-effector ($i = 3$)

$$\hat{\mathbf{f}}_F = \hat{\mu}_{di} \tanh\left(\frac{4v}{v_t}\right)\frac{\mathbf{v}}{v} + \frac{(\hat{\mu}_{si} - \hat{\mu}_{di})\mathbf{v}}{\left(\frac{1}{4}\left(\frac{v}{v_t}\right)^2 + \frac{3}{4}\right)^2 v_t} \tag{3.38}$$

where $\hat{\mu}_d$, $\hat{\mu}_s$ and $\hat{\mu}_v$ are the dynamic, static and viscous frictions, respectively. ω_t and v_t are the angular and linear transition velocities for the stiction. The total number of normalized friction parameters for this approach is $2 \times 4 + 3$ ($\boldsymbol{\pi}_{\mathrm{F(A2)}} = 11$).

The total number of unknown dynamic parameters for A1 and A2 are $\boldsymbol{\pi}_{\mathrm{p}} = 29$ ($= \boldsymbol{\pi}_{\mathrm{I}} + \boldsymbol{\pi}_{\mathrm{S}} + \boldsymbol{\pi}_{\mathrm{E}} + \boldsymbol{\pi}_{\mathrm{F(A1)}}$) and $\boldsymbol{\pi}_{\mathrm{p}} = 20$ ($= \boldsymbol{\pi}_{\mathrm{I}} + \boldsymbol{\pi}_{\mathrm{S}} + \boldsymbol{\pi}_{\mathrm{E}} + \boldsymbol{\pi}_{\mathrm{F(A2)}}$), respectively. If we introduce $\mathbf{p}$ as the vector of unknown parameters, the final format of the robot ODE system is:

$$\begin{cases} \mathrm{A1} : \dot{\mathbf{x}}_{8\times1} = \mathfrak{F}_1(\mathbf{x}_{8\times1}, \mathbf{u}_{2\times1}, \mathbf{p}_{29\times1}) \\ \mathrm{A2} : \dot{\mathbf{x}}_{4\times1} = \mathfrak{F}_2(\mathbf{x}_{4\times1}, \mathbf{u}_{2\times1}, \mathbf{p}_{20\times1}) \end{cases} \tag{3.39}$$

where $\mathbf{x}_{8\times1} = [q_1, q_2, \omega_1, \omega_2, \xi_1, \xi_2, \boldsymbol{\mathfrak{z}}^T]^T$ and $\mathbf{x}_{4\times1} = [q_1, q_2, \omega_1, \omega_2]^T$ are the corresponding state vectors, and $\mathbf{u}_{2\times1}$ is the driving torque vector.

Input-output matching

Since the robot has a closed-chain, conventional IO matching methods cannot be performed, unless one of the joints is locked. However, by locking a joint, it is not possible to identify the coupled dynamics. In this section, first, we introduce the robot's workspace characteristics. Next, using a transformation, the IO matching problem will be solved.

The robot is driven by two joints at its base (q_1 and q_2). Hence, its workspace (see Fig. 3.10) is bounded by some manufacturing limits and the following constraints:

$$\begin{cases} q_{1min} \leq q_1 \leq q_{1max} \\ q_{2min} \leq q_2 \leq q_{2max} \\ q_{21min} \leq q_2 - q_1 \leq q_{21max} \end{cases} \tag{3.40}$$

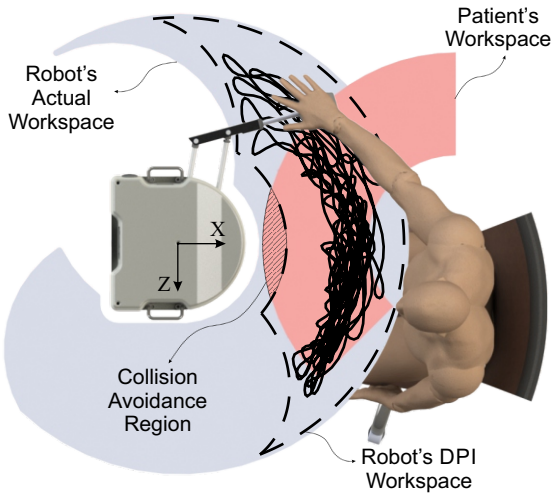


Figure 3.10: Workspaces of the robot and patient. Solid-line shows an example of DPI test on the robot inside the DPI workspace. The DPI workspace overlaps the patient's workspace. To reach the non-overlapping areas of the DPI workspace, the patient's lateral and vertical position can change by moving the chair and adjusting its height.

Moreover, since the robot interacts with a stroke patient, its DPI workspace is more confined than its actual workspace. Fig. 3.10 shows this workspace. To avoid possible collision of the patient's hand with the robot, the hatched area of the patient's workspace is excluded from the robot's DPI workspace, which is bounded by these constraints:

$$\begin{cases} q_{21min} \leq q_2 - q_1 \leq q_{21max} \\ \mathbf{A}_q \mathbf{q} \leq \mathbf{b}_q \end{cases} \tag{3.41}$$

Considering these constraints, to run the DPI experiments using sufficiently rich input, we changed the variables with the following transformation[1]:

$$\begin{cases} \zeta = q_2 + q_1 \\ \eta = q_2 - q_1 \end{cases} \tag{3.42}$$

Then, a sufficiently rich trajectory is generated:

$$\begin{cases} \zeta = \sum_{j=1}^{16} \bar{\zeta}_j \sin(\mathfrak{Z}_j t) + \zeta_0 \\ \eta^\iota = \bar{\eta}_{\mathrm{L}}^\iota \sin(\mathfrak{H}_{\mathrm{L}}^\iota t) + \bar{\eta}_{\mathrm{H}}^\iota \sin(\mathfrak{H}_{\mathrm{H}}^\iota t) + \eta_0 \end{cases} \tag{3.43}$$

[1]If the robot's DPI workspace is approximated by an annular sector ($\circlearrowleft$) in the Cartesian space, then a polar coordinate transformation can be introduced to isolate the corresponding coordinates and run experiments with a sufficiently rich input.

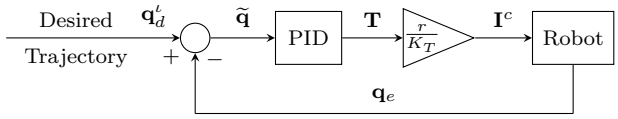


Figure 3.11: PID controller scheme for the DPI experiments. Subscripts d and e denote the desired and experimental values, superscript ι indicates the index of the selected frequency content for the desired trajectory. $\widetilde{\mathbf{q}} = \mathbf{q}_d^\iota - \mathbf{q}_e$ is the tracking error in the joint space.

where $[\mathfrak{Z}_j] = [0.01, 0.05, 0.1, 0.3, 0.5, 0.8, 1, 1.6, 2, 2.5, 3, 4, 5, 6, 8, 10]$ rad/s, $[\mathfrak{H}_{\mathrm{L}}^\iota] = [0.15, 0.25, 1.5, 2.5, 3.5]$ rad/s, $[\mathfrak{H}_{\mathrm{H}}^\iota] = [0.25, 0.85, 5.5, 6.5, 7.5]$ rad/s are the corresponding frequencies. DPI experiments at different frequencies (5 in total) can be done by varying ι from 1 to 5; consequently, $\mathfrak{H}_{\mathrm{L,H}}^\iota$ will change from low to high frequency contents. At low frequencies of $\mathfrak{H}_{\mathrm{L,H}}^\iota$, q_1 and q_2 will have less similarity than in their high frequencies (see Fig. 3.14); in contrast, at high frequencies this behavior will be reversed. $\bar{\zeta}_j$ and $\bar{\eta}_{\mathrm{L,H}}^\iota$ are the corresponding amplitudes that satisfy the following constraint (zero initial velocity constraint):

$$\begin{cases} \sum_{j=1}^{16} \bar{\zeta}_j \mathfrak{Z}_j = 0 \\ \bar{\eta}_{\mathrm{L}}^\iota \mathfrak{H}_{\mathrm{L}}^\iota + \bar{\eta}_{\mathrm{H}}^\iota \mathfrak{H}_{\mathrm{H}}^\iota = 0 \end{cases} \tag{3.44}$$

The control schematic in Fig. 3.11 is used to perform the experiments. The Proportional-Integral-Derivative (PID) controller receives the tracking error (desired trajectory minus end-effector position) in the Cartesian space and provides the input to the motors to follow the end-effector's desired path.

Direct collocation method

The upper extremity rehabilitation robot has noisy inputs and outputs as shown in Fig. 3.12. The currents are fed to the DC motors and they generate driving toques $\mathbf{T} = K_T \mathbf{I}^c(t)/r$; by subtracting noises ($\mathbf{d}$) caused by unknown dynamics $\mathbf{d}_1$ and sandwiched backlash $\mathbf{d}_2$, the model torque input $\mathbf{u} = \mathbf{T} - \mathbf{d}$ is obtained. The robot's outputs are the motor joint angles ($\mathbf{q}_e$), while the model output is the parallelogram joint angles ($\mathbf{q}$), which are different from the motor joint angles due to the backlash. The DPI goal is to find a mathematical model for the parallelogram (with the transformed dynamics of the DC motors) of the rehabilitation robot (see Fig. 3.12). This model has an input $\mathbf{u}$, an output $\mathbf{q}$ and an unknown

parameter vector $\mathbf{p}$, which is calculated from the following optimization problem:

$$\arg\min_{\mathbf{p}} \left(\frac{1}{2} \int_0^{t_f} (\mathbf{q}(t) - \mathbf{q}_e(t))^T (\mathbf{q}(t) - \mathbf{q}_e(t)) dt \right) \tag{3.45}$$

subject to:

$$\text{A1:} \begin{cases} \mathbf{p} \in \mathbb{R}^{29\times 1} \\ \text{Dynamic Equation (3.39.A1)} \end{cases} \tag{3.46}$$

or

$$\text{A2:} \begin{cases} \mathbf{p} \in \mathbb{R}^{20\times 1} \\ \text{Dynamic Equation (3.39.A2)} \end{cases} \tag{3.47}$$

Neglecting $\mathbf{d}$ and assuming $\mathbf{u} = \frac{K_T \mathbf{I}^c(t)}{r}$, this optimization problem can be solved by any nonlinear optimization algorithm; however, the noisy input ($\mathbf{u} = \frac{K_T \mathbf{I}^c(t)}{r} - \mathbf{d}$) impedes the convergence of most algorithms. To address this issue, assuming a normal distribution for $\mathbf{d^T d}$ with zero mean and variance ς^2, we propose an alternative optimal control problem with the following objective functional:

$$\mathfrak{J} = \frac{1}{2} \int_0^{t_f} \left(\mathbf{u}(t) - \frac{K_T \mathbf{I}^c(t)}{r} \right)^T \left(\mathbf{u}(t) - \frac{K_T \mathbf{I}^c(t)}{r} \right) dt \tag{3.48}$$

instead of (3.45). It is subject to (3.46) or (3.47) with the path constraint:

$$\mathfrak{c}(t) = \frac{1}{2} (\mathbf{q}(t) - \mathbf{q}_e(t))^T (\mathbf{q}(t) - \mathbf{q}_e(t)) \leq \beta^2 \tag{3.49}$$

Here, $\beta = 0.005$ rad $= 0.286$ deg is the backlash gap size. This method does input matching with bounded output error. In the proposed problem, the noisy inputs and outputs do not affect the convergence of the DPI process. By treating each DPI experiment (with different frequency content) as an optimal control phase, a multi-phase optimal control problem will result. Since there are experimental values in the problem structure, this multi-phase optimal control problem should be solved using a numerical approach (i.e. indirect and direct approaches). For the proposed problem, a direct approach with state and control parameterization using global orthogonal collocation method (see Fig. 3.13) is appropriate, because [226]:

1. the system dynamics is simulated implicitly,
2. it is not required to use predictor-corrector,
3. ODEs are solved simultaneously,

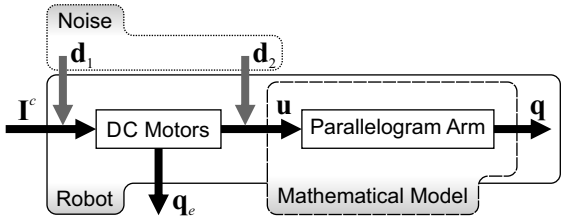


Figure 3.12: Input ($\mathbf{I}^c$) and output ($\mathbf{q}_e$) of the upper extremity rehabilitation robot in the presence of noises caused by unknown dynamics ($\mathbf{d}_1$) and sandwiched backlash ($\mathbf{d}_2$). Desired mathematical model for the parallelogram arm with specified input ($\mathbf{u}$) and output ($\mathbf{q}$).

4. a static parameter estimator can be integrated into this approach,
5. it is more accurate than other numerical methods and has exponential convergence,
6. this method results in a large-scale sparse Nonlinear Programming (NLP), that can be solved efficiently.

The GPOPS-II optimal control software package is used to solve multi-phase optimal control problems with unknown static parameters; this software combines variable-order adaptive orthogonal collocation methods with sparse NLP, i.e. Interior Point Optimizer (IPOPT) and Sparse Nonlinear Optimizer (SNOPT) [211]. The main advantage of this software is the implementation of state-of-the-art nonlinear optimization methods. In this study, GPOPS-II with the IPOPT solver is used to deal with the multi-phase optimal control problem.

3.3.3 Simulation and experiment

Before running the experiments, the desired trajectory is simulated in the Cartesian space, to check if the desired trajectory lies inside the DPI workspace. Fig. 3.14 shows the $\iota = 1$ (low frequency content) desired trajectory input to the PID controller in three spaces, the transformed, joint and Cartesian spaces. As expected, since $\eta = q_2 - q_1$ has low frequency content, q_1 and q_2 are more different at low frequencies compared to high frequencies. This figure also shows the advantage of the defined transformation. At each time instant, the bounds for each DOF (e.g., second column in Fig. 3.14 for three spaces) are evaluated by varying the DOF (e.g., ζ, q_1, and X) inside its DPI workspace (first column in Fig. 3.14) while freezing the latter DOF (e.g., η, q_2, and Z) at its corresponding time instant value.

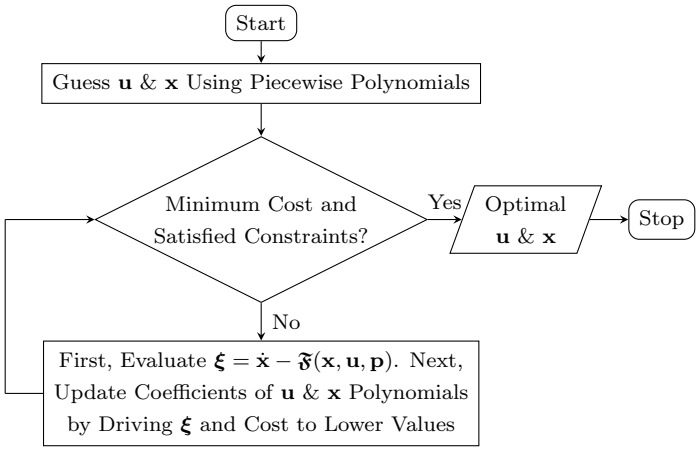


Figure 3.13: Direct collocation method flowchart.

The desired trajectory inside its bounds in the transformed space (ζ, η) is more distinct than the joint space and Cartesian space results.

Other than 5 different frequency contents (changes in the experimental input), 3 different conditions for the robot end-effector (changes in experiment setup) are considered:

① The end-effector lies on a Medium-Density Fiberboard (MDF) board ($N_1 = N_2 = N_3 = 1$).

② To compare different end-effector friction properties, the end-effector lies on an acrylic sheet ($N_1 = N_2 = N_3 = 1$).

③ To measure friction properties, the second condition is tested while a gray cast cylindrical weight (Diameter: $d_w = 76.96$ mm, Height: $h_w = 26.30$ mm, and Mass: $m_w = 951.52$ gr) is added to the end-effector ($N_1 = N_2 = 1$, but N_3 should be identified). The weight is located at $[z_w, x_w] \approx [l_4, 0]$ defined in the 4^{th} link local coordinate. For this setup, to avoid high motor currents, high frequency experiments (where $\iota = 4, 5$) are not performed.

Overall, 13 $(= 5 + 5 + 3)$ tests are performed on the robot. The highest frequency experiments ($\iota = 5$ for setups ① and ②) are kept for validation of the proposed DPI process, and 11 $(= 4 + 4 + 3)$ tests are used for the rehabilitation robot DPI. Each setup (①,②,③) has its own parameter vector $(\mathbf{p}^{①}, \mathbf{p}^{②}, \mathbf{p}^{③})$. $\mathbf{p}^{②}$ has the same parameter values as $\mathbf{p}^{①}$, except for the end-effector normalized frictions and corresponding velocities (i.e., $\hat{\sigma}_{03}$,

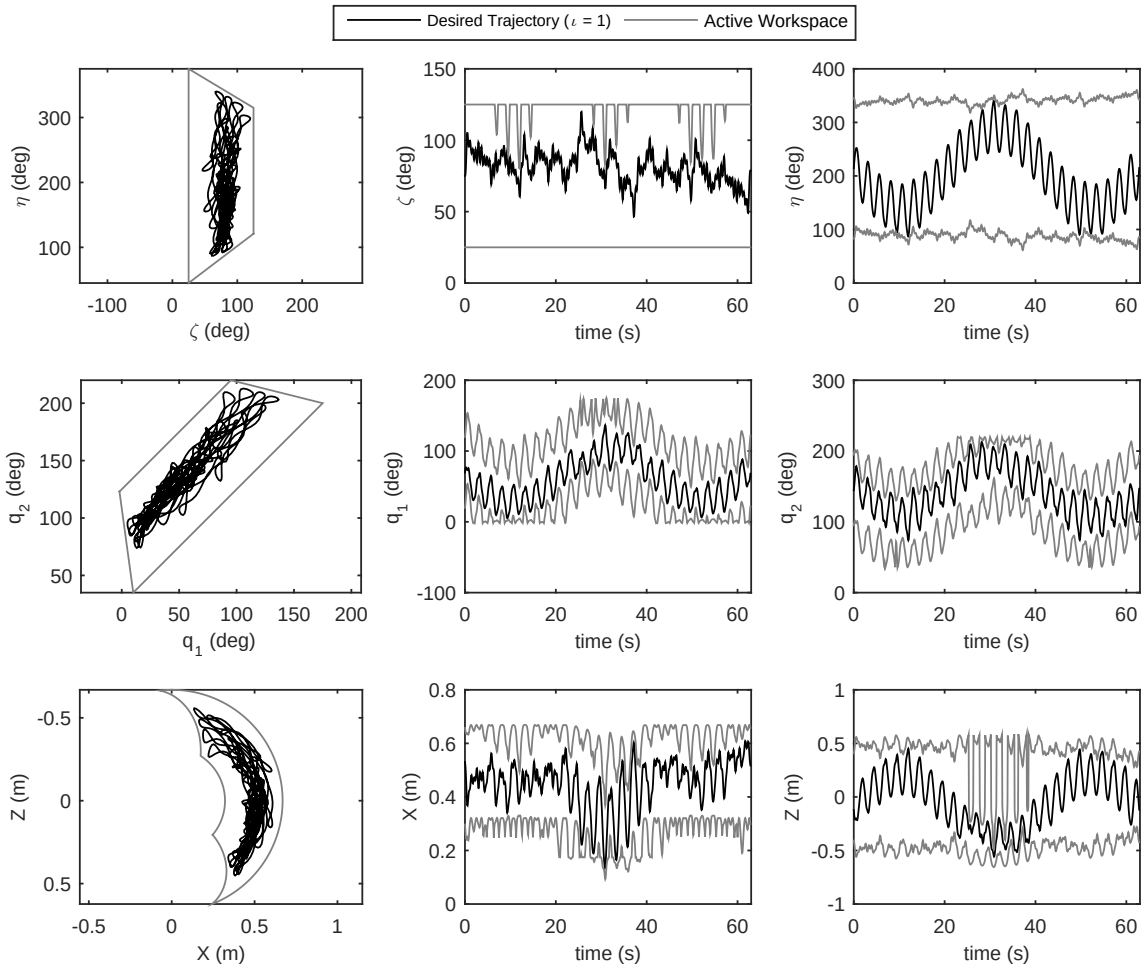


Figure 3.14: Robot's desired trajectory at $\iota = 1$ inside its DPI workspace. First row shows the results in the transformed space (ζ, η), second row is the joint space (q_1, q_2) results, and third row shows the Cartesian space (Z, X) results. In the second and third columns, the active workspace has lower and upper bounds that are evaluated at each time instance based on the corresponding variable's plot.

$\hat{\sigma}_{13}$, $\hat{\mu}_{c3}$, $\hat{\mu}_{\mathbb{S}3}$, v_d, v_0 for A1; and $\hat{\mu}_{d3}$, $\hat{\mu}_{s3}$, and v_t for A2). These parameters constitute the vector denoted by $\mathbf{p}^{②-①}$. Similarly $\mathbf{p}^{③}$ has the same parameter values as $\mathbf{p}^{②}$, except for the inertial parameters ($p_{1..4}$) and end-effector's weighting factor N_3, which constitute the vector $\mathbf{p}^{③-②}$. The extra weight in setup ③ changes the inertial parameter equations (3.33) as follows:

$$\begin{cases} p_1^{③} = p_1^{①} + (x_w^2 + z_w^2)m_w + J_w \\ p_2^{③} = p_2^{①} + m_w l_2 z_w \\ p_3^{③} = p_3^{①} - m_w l_2 x_w \\ p_4^{③} = p_4^{①} + m_w l_2^2 \end{cases} \tag{3.50}$$

where $J_w \approx m_w d_w^2/8$ is the moment of inertia of the added weight to the end-effector. Moreover, this extra weight will increase the end-effector's normal force; hence, considering A2, the following equations are obtained:

$$\begin{aligned} N_3^{③}\hat{\mu}_{d3}^{③} &= N_3^{②}\hat{\mu}_{d3}^{②} + 9.81 m_w \mu_{d3}^{②} \\ N_3^{③}\hat{\mu}_{s3}^{③} &= N_3^{②}\hat{\mu}_{s3}^{②} + 9.81 m_w \mu_{s3}^{②} \end{aligned} \tag{3.51}$$

where $\mu_{d3}^{②}$ and $\mu_{s3}^{②}$ are the dynamic and static friction coefficients between the Polyacetal ball and acrylic sheet. Thus, if the unknown parameters of the system are identified by solving the above equations, the dynamic and static friction coefficients can be estimated. Knowing these coefficients, and solving the following equations:

$$\begin{aligned} \left[\frac{\hat{\mu}_{d3}}{\mu_{d3}}\right]^{②} &= \left[\frac{\hat{\mu}_{d3}}{\mu_{d3}}\right]^{①} \\ \left[\frac{\hat{\mu}_{s3}}{\mu_{s3}}\right]^{②} &= \left[\frac{\hat{\mu}_{s3}}{\mu_{s3}}\right]^{①} \end{aligned} \tag{3.52}$$

one can estimate the dynamic and static friction coefficients ($\mu_{d3}^{①}$ and $\mu_{s3}^{①}$) between the Polyacetal ball and MDF board.

To run the DPI with GPOPS-II, each experiment is considered in a phase. Thus, a problem with 11 phases ($\iota = 1..4$ experiments in setup ①: phases (1) to (4), $\iota = 1..4$ experiments in setup ②: phases (5) to (8), and $\iota = 1..3$ experiments in setup ③: phases (9) to (11)) and a new parameter vector defined as:

$$\bar{\mathbf{p}} = \begin{bmatrix} \mathbf{p}^{①} \\ \mathbf{p}^{②-①} \\ \mathbf{p}^{③-②} \\ N_3^{③} \end{bmatrix} \tag{3.53}$$

is considered. For this GPOPS-II problem, the experimental results are provided as the initial guesses, the number of collocation points is set to 3, applied collocation (a type of Radau

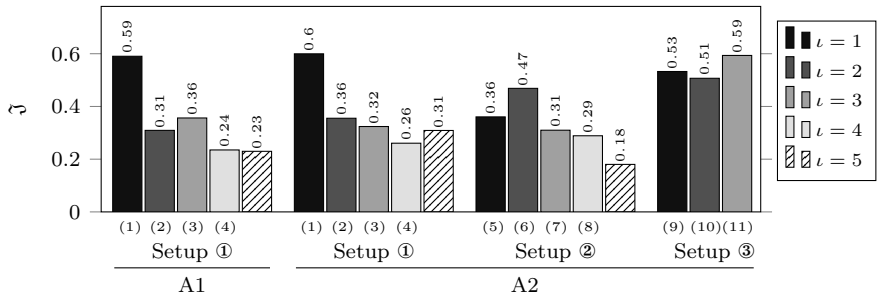


Figure 3.15: Comparing objective functional values of two friction approaches in the robot DPI using setup ① experiments, and objective functional values of the A2 DPI using the experiments of setups ② and ③. Numbers in parentheses indicate the phase number.

Pseudospectral Method (RPM) which uses integration matrix [95]), '`hp-LiuRao-Legendre`' is the mesh refinement method [159], a sparse central derivative is used to supply the first-order derivatives, and the NLP solver is set to IPOPT.

The identified model is used in a sliding-mode control structure to move the robot end-effector in hypotrochoid paths including Sharp Corner (SC) and Rounded Corner (RC) star shapes at two different frequencies $\Omega_d = 1, 4$ rad/s (see Section A.3 for the details of the desired trajectory equation). Equations and parameters of the implemented sliding-mode control, which are derived from [261], are also presented in Section A.4.

3.3.4 Results and discussion

To select a proper friction model for the DPI of 11 phases, two friction approaches are compared in the DPI of the setup ① experiments (phases (1) to (4)). As indicated in Fig. 3.15, the objective functional values of both approaches are similar. However, the CPU time of the A1 DPI is about twice the CPU time of the A2 DPI. The DPI using A2 is as accurate as the A1 DPI, but lower in computational cost. Thus, in the preceding DPI problem, A2 is selected to model the system friction.

Objective functional values of the A2 DPI for 11 phases are shown in Fig. 3.15. The minimum Signal to Noise Ratio (SNR), evaluated on the system torque input, is 15.30, and it occurs in the DPI of the phase (2) (i.e., $\iota = 2$). Lower SNR values happen at low frequencies, where the frictional terms affect the system behavior more than the inertial terms. The objective functional values of the setup ③ are greater than those for the other setups. SNR values for setup ③ are higher than those for the other setups, because:

1. applied torque magnitudes in setup ③ experiments are greater than the other setups' torques (added weight increases robot's effort),

2. added weight increases the effect of inertial terms more than frictional terms.

The proposed DPI identifies the system with at most $\pm 12.7\%$ average noise in the input torques, and by increasing the input frequency or system inertia, this value decreases. See Tables A.5 and A.6 for the list of identified parameters.

The identified inertial parameters of setup ① and ③ are $p^{①}_{1..4} = [0.1145, 0.0931, -0.0027, 0.1669]$ and $p^{③}_{1..4} = [0.2484, 0.2076, -0.0027, 0.2584]$, respectively. Using the identified $p^{①}_{1..4}$ and properties of the added weight in (3.50), inertial parameters of setup ③ can be estimated as $\mathring{p}^{③}_{1..4} = [0.2484, 0.2037, -0.0027, 0.2584]$. Hence, inertial parameters of setup ③ are successfully identified with the following error $\widetilde{p}^{③}_{1..4}/\mathring{p}^{③}_{1..4} = [0.0, 1.9, 0.0, 0.0]\%$.

High frequency experiments ($\iota = 5$) are used to check the system response with the identified parameters. In other words, the optimal control problem in Fig. 3.13 with the identified parameters is solved to minimize the objective functional (3.48) subject to (3.46) or (3.47), and the path constraint (3.49). Objective functional values for these experiments are close to the other experiments (see Fig. 3.15), thereby validating the identified parameters.

To check the effect of unmodeled dynamics on the proposed DPI, system torque input errors ($\widetilde{\mathbf{u}} = \mathbf{u}(t) - \frac{K_T \mathbf{I}^c(t)}{r}$) are compared for different phases (experiments) using covariance matrices, which are displayed in Fig. 3.16. Along the specified k^{th} diagonals, the elements of the corresponding covariance matrices are greater than the other elements. The 5^{th} diagonal elements indicate the correlation between ι^{th} input of setups ① and ②, and ι^{th} input of the setups ② and ③. The 9^{th} diagonal elements indicate the correlation between ι^{th} input of setups ① and ③. The correlation between two setups at ι^{th} input, where the desired trajectory is the same between the experiments of the setups, indicates that there are some unmodeled dynamics in the system. These trajectory dependent unmodeled dynamics may be because of the sandwiched backlash and unmodeled variable normal force in the system friction. Second joint torque inputs for setup ③ are more than the setup ② inputs, which are higher than the setup ③ values. Higher torque inputs will result in larger variances (i.e., the 0^{th} diagonal elements has greater values), and this increases the covariance of the corresponding elements. That is why, the covariance matrix in the second joint has large elements inside the rectangle extending from cell {(5),(5)} to {(11),(11)}; this rectangle covers experiments of setups ② and ③.

The corresponding friction curves are plotted in Fig. 3.17. The identified joint frictions have greater dynamic friction than static friction. This seems to be physically impossible;

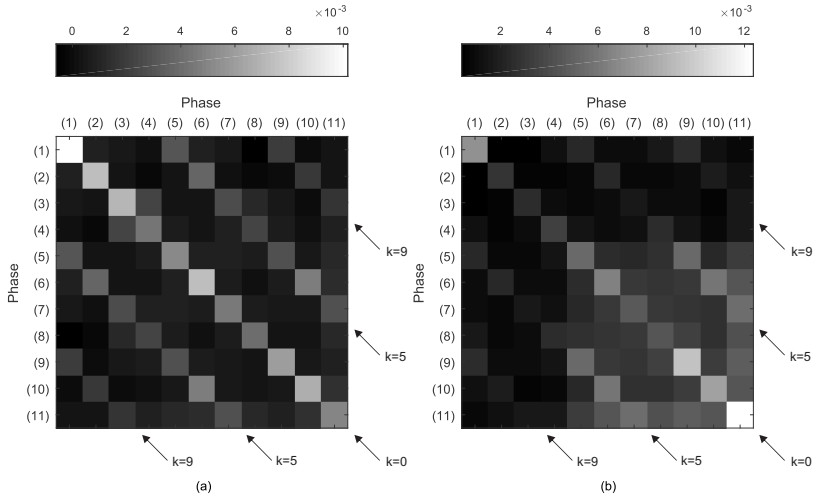


Figure 3.16: Covariance matrix of the system torque input errors for a) joint 1, and b) joint 2.

however, joint backlash can cause this phenomenon. When the motor-side gears are inside the backlash gap, the model captures the static friction. Once the motor-side gear is moving, and the backlash gap is filled (motor-side and load-side gears are in contact), because of load-side addition to the system, the model detects large values for the dynamic friction. The same phenomenon happened in the end-effector friction forces for the same reason and Polyacetal's friction properties. Polyacetal in contact with itself and mild steel shows higher dynamic friction than static friction, and this behavior is consistent in some of the nylon-based materials [59]. Thus, in the end-effector, the Polyacetal balls' contact with the MDF board and the acrylic sheet may cause higher dynamic friction than static friction. One may argue that these ball rollers can only cause rolling friction. However, the end-effector weight and miniature structure of these rollers are preventing the Polyacetal balls from pure rolling.

Solving (3.51) and (3.52) with the identified parameters estimates the dynamic and static friction coefficients between two different contact surfaces. The estimated values are reported in Table 3.5. Because of Polyacetal friction properties, estimated dynamic friction coefficients are more than the static ones. Because of higher friction coefficient values of acrylic compared to the friction coefficients of wood [59], estimated friction coefficients between the Polyacetal and acrylic sheet should be greater than estimated friction coefficients between the Polyacetal and MDF board.

Evaluative path tracking experiments (shown in Fig. 3.18) are done using the identified

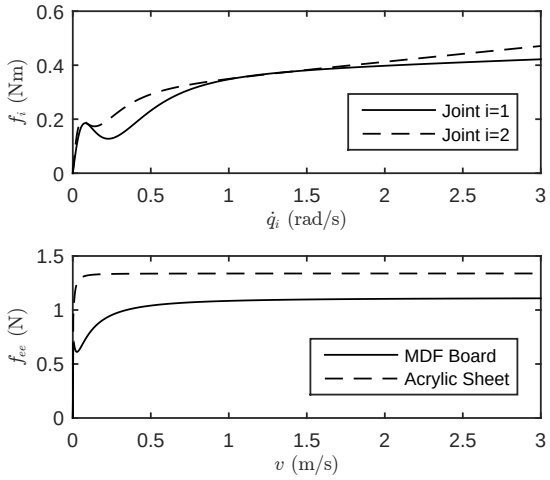


Figure 3.17: Identified friction models of the robot. The top plot shows the friction torques on the robot joints, and the bottom plot shows the end-effector friction forces for setup ① (Polyacetal ball contact with the MDF board) and setup ② (Polyacetal ball contact with the acrylic sheet).

Table 3.5: Estimated dynamic and static friction coefficients between two different contact surfaces (①: Polyacetal and MDF board, ②: Polyacetal and acrylic sheet)

Coefficient	$\mu_{d3}^{①}$	$\mu_{s3}^{①}$	$\mu_{d3}^{②}$	$\mu_{s3}^{②}$
Value	0.0918	0.0505	0.1102	0.0753

model in the sliding mode control structure. SNR values for these tests are greater than 10.17, which indicates that the system should be controlled within $[84.3, 115.7]\%$ of the identified model's torque estimates. The estimate error is 3% greater than the DPI results since the angular position input is the motor-side angle rather than the load-side angle. Position and velocity tracking errors are within $\pm 8\%$ and $\pm 11\%$ of the desired joint angle and velocity inputs, respectively. Comparing the high frequency experiments with the low frequency experiments, when the tracking speed increases, the unknown dynamics effect reduces. In the RC experiments, the robot performs better than the RC experiments, since in RC, the robot tries to change the end-effector's velocity direction gradually, and this increases the effect of frictional terms while decreasing the effect of inertial terms, thereby

increasing the error. It is worth noting that if we remove the identified model from sliding mode structure the controller will change to a proportional-differential controller. Running the robot with this controller moves the end-effector in the horizontal plane with ±89% position tracking error.

3.3.5 Conclusions

In this section, we presented a direct collocation method for the DPI of a noisy IO upper extremity rehabilitation robot. The robot has gear belt backlash (with no load-side state measurement), and its end-effector moves on a surface. The proposed collocation method was implemented using the GPOPS-II optimal control software package. Different identification tests were performed and the results of the proposed method were verified. The robot dynamics was modeled by two different friction models: LuGre and continuous-velocity models, and their accuracies were compared. Using different tests, we tried to identify the surface friction and some inertial properties of the system. Finally, the robot was controlled to follow a desired trajectory using the identified model in a sliding-mode controller structure. The results showed that the robot could move inside its workspace with at most ±8% error in position if the estimated model's torques are adjusted within ±15.7% bound.

3.4 Concluding remarks

In this chapter, based on the discussed mechanical issues of the robot, dynamic models of the robot using our two proposed DPI methods were presented. First, we proposed a modified homotopy optimization to identify parameters of the system dynamics with mechanical discontinuity (i.e., backlash). After verifying the proposed method using computer simulations, the method was used to identify the parameters of the 2 DOF highly noisy and nonlinear rehabilitation robot with actuator backlash, when the robot's second joint was locked in different configurations. The proposed method provided better estimates for the system parameters compared to the no-backlash DPI of the robot. Despite the angle measurement noise and its propagation to the velocity and acceleration evaluations of the model, forward dynamics results were quite accurate for all of the configurations with the discontinuous backlash model.

Although the obtained accuracy in 1-Dimensional (1D) was good, the uncertainty of the states during robot interaction and the computation cost caused by the discontinuity

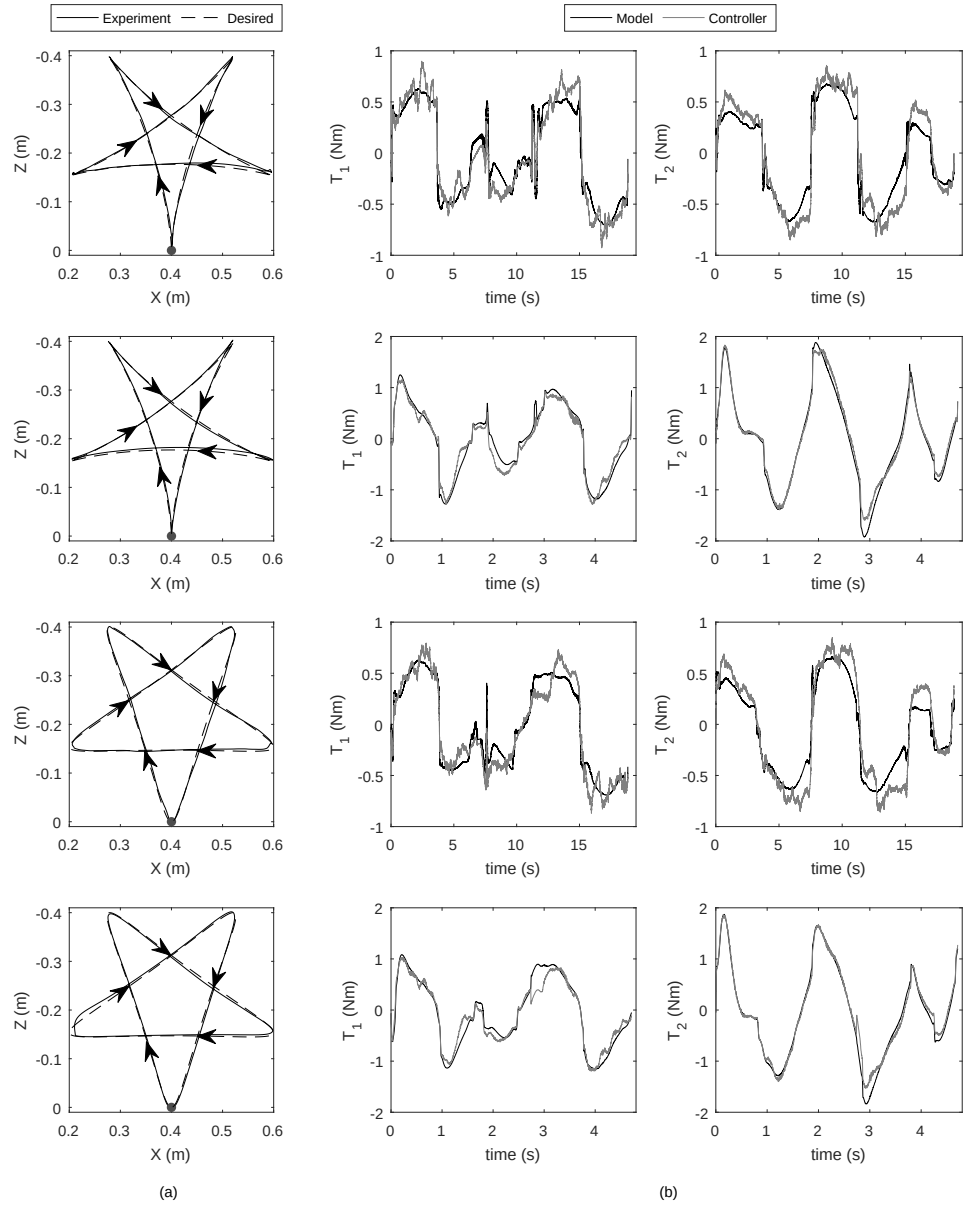


Figure 3.18: Path tracking experiments using the identified model in the sliding mode controller structure. a) The desired path and tracking performance (gray circle indicates the start and finish points). b) Input torques to the robot compared with the model's estimated torque. 1st row: SC $\Omega_d = 1$ rad/s, 2nd row: SC $\Omega_d = 4$ rad/s, 3rd row: RC $\Omega_d = 1$ rad/s, and 4th row: RC $\Omega_d = 4$ rad/s.

led us to develop a continuous model for the 2D robot. Thus, next, we presented a direct collocation method for the DPI of the noisy upper extremity rehabilitation robot. Different identification tests were performed and the results of the proposed method were verified. The robot dynamics were modeled by two different friction models: LuGre and continuous-velocity models, and their accuracies were compared. Using different tests, we tried to identify the surface friction and some inertial properties of the system. Finally, the robot was controlled to follow a desired trajectory using the identified model in a sliding-mode controller structure. The results showed that the robot could move inside its workspace with a good accuracy.

Chapter 4

System Models: Human and Human-Robot Interaction

Since the next subsystem in the human-robot interaction system is human, in this chapter, first, we develop human (musculoskeletal upper extremity) models. Next, since the evaluation and DPI of a musculoskeletal model requires proper instrumentation (i.e., a robotic arm to measure forces and displacements), the integration of the human and robot models are discussed in Section 4.2 (Human-robot interaction models). Finally, using the human models in the HRI, this chapter concludes with cross-validation (simulation) studies for the evaluation of human models and DPI (experimental) studies of human models.

4.1 Musculoskeletal upper extremity models

We have used a musculoskeletal model to represent the human body. To build such a model, one should consider the tradeoff between the model complexity and computation costs. To be used within the NMPC scheme, the musculoskeletal upper extremity model should have a simple structure. Our upper extremity rehabilitation robot is an end-effector based planar robot, which performs reaching movements in the horizontal plane for therapy of shoulder and elbow. Thus, since the robot performs 2D movements, the musculoskeletal arm is modeled as a 2D two-link arm operating in the horizontal plane.

In the first stage of model development, we use a planar 2D musculoskeletal arm model (MA2D-I_1) to interact with this planar robot. In other stages, we will try to enrich the model by developing:

- MA3D-I: using the Stanford 3D musculoskeletal arm model,
- MA2D-I_2: adding shoulder rhythm (see page 16 for the definition) to MA2D-I_1 based on the MA3D-I shoulder joint angles and the regression equation used by [122],
- MA2D-I_3: defining a new 2D model based on MA3D-I,
- MA3D-II: modifying the Lisbon 3D musculoskeletal arm model,
- MA2D-II: defining a new 2D model based on MA3D-II.

For these musculoskeletal models of the upper extremity, it is assumed that there is no tendon compliance in the musculotendon structure. The compliance of tendon is proportional to its slack length. Thus, a tendon is compliant if its normalized slack length is large (≥ 10), and it is very stiff when it is equal to 1 [336]. For most muscles in the upper extremity this value is around 1; hence, the stiff tendon assumption seems to be valid. For accomplishment of each of these stages, combinations of different software packages (OpenSim, MATLAB®, SolidWorks®, MapleSim™, Simscape Multibody™, and MSC® ADAMS) are used. These stages will be covered in this section.

4.1.1 MA2D-I_1

Because of simplicity, the use of upper extremity 2D musculoskeletal models is common for studying reaching movements in the horizontal plane [22, 197, 276, 335]. This model has two hinged links with 6 muscle groups including shoulder mono-articular flexor (Muscle1: $\mathscr{M} = 1$), shoulder mono-articular extensor (Muscle2: $\mathscr{M} = 2$), elbow mono-articular flexor (Muscle3: $\mathscr{M} = 3$), elbow mono-articular extensor (Muscle4: $\mathscr{M} = 4$), shoulder-elbow bi-articular flexor (Muscle5: $\mathscr{M} = 5$), and shoulder-elbow bi-articular extensor (Muscle6: $\mathscr{M} = 6$). In this stage, we are using a two link arm model (see Fig. 4.1) presented in [197, 276]. Model specifications are summarized in Table 4.1.

For consistency in optimization results and reducing optimization fluctuations, damping coefficients are assigned to the shoulder and elbow joints. Following [155], their values are 1.7 Nms/rad for the shoulder joint, and 0.3 Nms/rad for the elbow joint. The cost function for FSO is:

$$\mathfrak{J}_{\text{FSO}} = w_1 \sum_{\mathscr{M}=1}^{n_{\mathscr{M}}} a_{\mathscr{M}}^2 + w_2 \left[k_X \widetilde{X}^2 + k_Z \widetilde{Z}^2 + k_{\dot{X}} \dot{\widetilde{X}}^2 + k_{\dot{Z}} \dot{\widetilde{Z}}^2 \right] \tag{4.1}$$

where $a_{\mathscr{M}}$ is the $\mathscr{M}^{\text{th}}$ muscle group activation, $n_{\mathscr{M}}$ is the number of muscle groups (for this model $n_{\mathscr{M}} = 6$), $\sim$ accent denotes the errors of the positions and velocities in the Cartesian

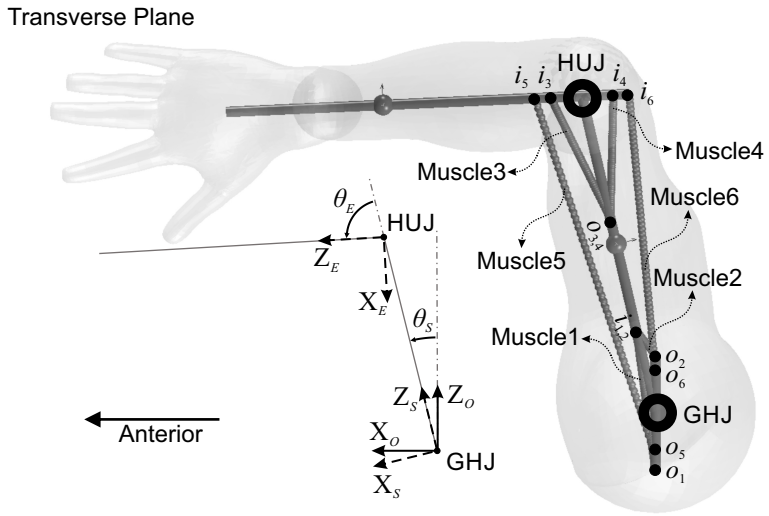


Figure 4.1: MA2D-I_1 configuration. $o_{\mathscr{M}}$ and $i_{\mathscr{M}}$ denote the $\mathscr{M}^{\text{th}}$ muscle group's origin and insertion points, respectively. See Table A.7 for the dynamic parameters. Muscle attachment points are reported in Table A.8.

Table 4.1: Muscle mechanics parameters for MA2D-I_1. For more information regarding muscle parameters refer to Appendix B.

Parameter	Unit	Muscle1	Muscle2	Muscle3	Muscle4	Muscle5	Muscle6
F^{iso}	N	1000	1000	1000	1000	1000	1000
L_0^{M}	cm	125	5.6	14.2	10.1	37	24.9
L_{s}^{T}	cm	0	0	0	0	0	0
α_0^{M}	deg	0	0	0	0	0	0

Table 4.2: MA3D-I specifications

# Muscles (Elements)	12 (22)
Muscle mechanics model	CE/PE/SE Hill-type
Muscle path model	Obstacle-set method $\longrightarrow$ Non-geometric path model
Muscle redundancy problem	SO/MSO
Musculoskeletal model type	3D/ Dynamic
Shoulder	Spherical joint (triple revolute joints)
Shoulder rhythm	None
Shoulder stability	N/A
Elbow	Single revolute joint (flexion/extension)

space, and $k_\bullet$'s are the weighting constants which are tuned manually. All steps including multibody dynamics modeling, visualization, and optimization are accomplished with MapleSim$^{\text{TM}}$, SolidWorks®, and MATLAB®/Simulink software packages, respectively.

4.1.2 MA3D-I

In this subsection, the goal is to develop a high-fidelity 3D musculoskeletal arm model to evaluate simulation results of the 2D model. To validate 2D model simulations, we will use inverse dynamics on its equivalent 3D model followed by an inverse kinematics analysis. A brief overview of the model is presented in Table 4.2. Next, skeletal modeling, muscle model, and implementation for this model will be covered.

Skeletal Model

We utilize the 3D Stanford musculoskeletal arm model, which has 12 DOF up to the wrist. For simplicity we assume no shoulder rhythm (-5 DOF), no elbow pronation/supination (-1 DOF), and a rigid wrist joint (-2 DOF). Thus, our 3D arm model has 4 DOF (see Fig. 4.2.a). It is possible to orient the 3D arm model in the horizontal plane such that the elbow flexion-extension axis becomes perpendicular to the transverse (horizontal) plane. In the 3D model, the shoulder joint is modeled by three revolute joints with intersecting axes. These body-fixed rotation axes are as follows ($\mathscr{j}$ is the joint/axis number, see Fig. 4.2.a):

$\mathscr{j} = 1$**:** Shoulder plane of elevation (in OpenSim: "`elv_angle`") axis (Y axis); rotation about this axis deviates the upper arm from the thorax in the transverse plane.

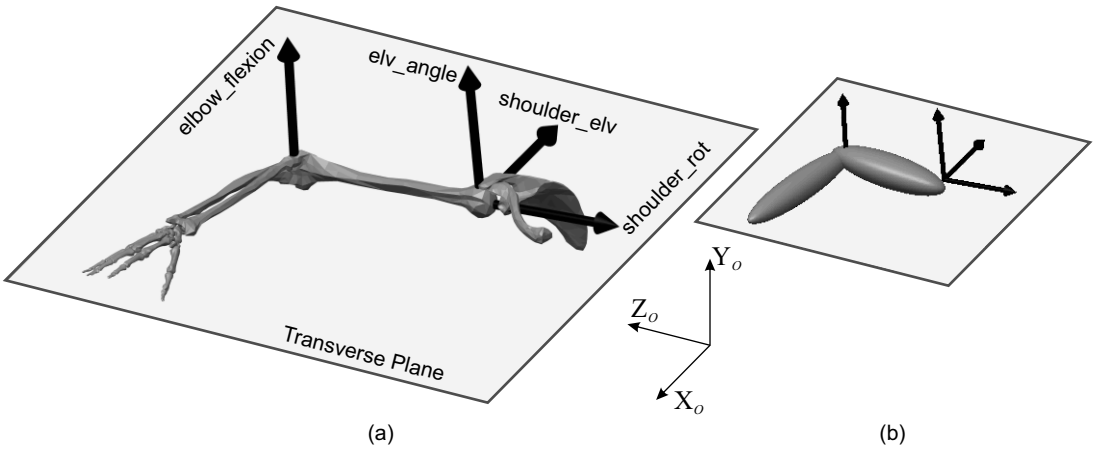


Figure 4.2: MA3D-I configuration. a) The skeletal model, and b) The ellipsoidal model.

$\mathcal{J}^{\cdot} = 2$**:** Shoulder elevation angle (in OpenSim: "`shoulder_elv`") axis (rotated body-fixed Z axis); rotation about this axis results in upper arm abduction/adduction.

$\mathcal{J}^{\cdot} = 3$**:** Shoulder rotation (in OpenSim: "`shoulder_rot`") axis (rotated body-fixed Y axis); rotation about this axis causes internal/external rotation of the upper arm. The shoulder rotation angle is measured with respect to the shoulder plane of elevation; hence, actual shoulder rotation angle is the total amount of rotation about this axis minus shoulder plane of elevation angle.

The elbow joint ($\mathcal{J}^{\cdot} = 4$) is a simple revolute joint defined by "`elbow_flexion`" axis in OpenSim. To generate a 2D-equivalent (MA2D-I_1) of the 3D musculoskeletal arm model (MA3D-I) the following steps are accomplished:

1. Scaling the 3D model based on the ratio of its upper arm length to the 2D model first link (upper arm) length.

2. Finding proper YZY rotation (shoulder rotation), so that the elbow flexion-extension axis becomes perpendicular to the transverse plane (i.e., it aligns with the Y_O axis).

3. Assigning the Center of Mass (COM) and moments of inertia of the segments such that, by following the mentioned YZY rotation, their values become identical with the inertial properties of the 2D model. Based on the anthropometric data [76], ellipsoids (with three unknown radii and two anthropometric constraints related to their ratios) are fitted to the upper and lower arms so that after prescribed rotations, they possess the correct moments of inertia in the horizontal plane (see Fig. 4.2.b).

Muscle model

The Thelen [283] muscle model is used to represent the Hill-type muscle dynamics, which is a physics-based approach to muscle mechanics. However, it is assumed that only the CE of the Hill-type muscle is generating muscle force.

$$F_m^{\mathrm{M}} = F^{a_m,\mathrm{CE}}\left(F_m^{\mathrm{iso}}, L_m, V_m\right)\cos(\alpha_m^{\mathrm{M}}) \tag{4.2}$$

where F_m^{M}, a_m, L_m and V_m are the m^{th} muscle force, activation, length and velocity, respectively, and $F^{\mathrm{CE}}(.)$ is the CE function. See Appendix B for more details.

The musculoskeletal model, which is available in the OpenSim software, uses the obstacle-set method for muscle path (see Fig. 4.3.a). We develop a non-geometric path model by computing musculotendon lengths and moment arms after inverse kinematic analysis of the model in OpenSim. This model will be used to generate new 2D models; thus, the inverse dynamics problem should be solved for this model. Consequently, the muscle force sharing problem can be solved by SO or MSO. The inputs of the SO are muscle activations, while for the MSO, the inputs are muscle forces. We use SO with the following cost function:

$$\mathfrak{J}_{\mathrm{SO}} = w_1 \sum_{m=1}^{n_m} a_m^2 \tag{4.3}$$

subject to:

$$T^j = \sum_{m=1}^{n_m} r_m^j F_m^{\mathrm{M}} \tag{4.4}$$

where m and j denote the muscle and joint numbers, respectively. n_m is the number of muscles[1] (for this model $n_m = 22$), T is the evaluated joint torque, and r and F are the muscle moment arm and force, respectively.

Implementation

Visualization is done in both SolidWorks® and OpenSim. Then for solving inverse kinematics, OpenSim and MATLAB® are utilized. For inverse dynamics analysis, the generated kinematic results are fed into OpenSim, MapleSim™, Simscape Multibody™, and MSC® ADAMS. A manual graph-theoretic approach was also used to validate the inverse dynamic results. Finally, the SO is done in both OpenSim and MATLAB®. Results are presented in Section 4.3 (Cross-validation studies of human models).

[1] In the text, we will use number of muscles instead of number of elements.

Model limitations

This model is limited by:

1. ROM for some joints: for example, maximum positive shoulder rotation angle in this model is 20 degrees [122], while this value in most references has been reported to be about 90 degrees [229].

2. Imperfect shoulder rhythm definition: in the Stanford model, the shoulder rhythm is defined by simplified regression-based equations which are only dependent on the shoulder elevation angle [122]. However, the most accurate shoulder-rhythm regression-based equations are in terms of three shoulder joint angles [326].

3. Incorrect muscle path generation for some orientations: this is due to the imperfect muscle path definition based on the via-point method.

4.1.3 MA2D-I_2

In the shoulder rhythm definition of the Stanford upper extremity musculoskeletal model, approximate regression-based formulations are used. These formulations are only dependent on the shoulder elevation angle, and they evaluate rotations of SCJ and ACJ (two spherical joints) in terms of 6 revolute joint rotations. If we evaluate shoulder elevation angle corresponding to the shoulder rotation in the horizontal plane, it will be possible to evaluate these angles and get the new position of the GHJ. This can be accomplished by mapping from the 2D model (MA2D-I_1) to the developed 3D model (MA3D-I) and back-mapping from MA3D-I to MA2D-I_1. Hence, we get two displacements on the horizontal plane for the shoulder joint. Finally, we apply these displacements to the shoulder joint by two prismatic joints, which are driven by the shoulder rotation angle in the horizontal plane. In this model, the shoulder joint has 3 DOF (1 independent and two dependent), so the new 2D model (MA2D-I_2) will have 4 DOF. Adding this extra DOF did not change the simulation results significantly. Results are presented in Section 4.3 (Cross-validation studies of human models).

4.1.4 MA2D-I_3

In comparing the horizontal plane reaching movements of the MA2D-I with MA3D-I, we noticed that the activation results of the 3D model simulation are different from the 2D

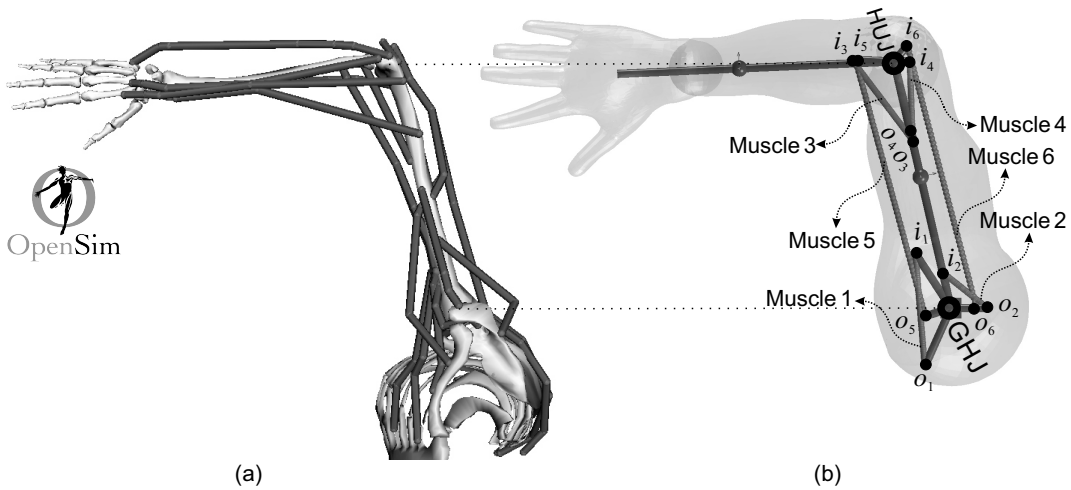


Figure 4.3: a) MA3D-I muscle path, and b) MA2D-I_3 configuration. See Table A.7 for the dynamic parameters. Muscle attachment points are reported in Table A.9.

model. Therefore, we modified the 2D model using a weighted average approach to lump the 22 muscles of MA3D-I into 6 muscle groups in a 2D model (see Fig. 4.3.b). In this subsection, we have modified the weighted average method used in [197]. The procedure of this modified model reduction is as follows:

1. Define muscle groups based on their synergistic muscles:

   Muscle 1: Anterior (Clavicular) and Middle (Acromial) Deltoid; and Clavicular, Sternal and Ribs Pectoralis Major.

   Muscle 2: Posterior (Scapular) Deltoid; Thoracic, Lumbar and Iliac Latissimus Dorsi; and Teres Major.

   Muscle 3: Brachioradialis; Brachialis; Pronator Teres; and Extensor Carpi Radialis Longus.

   Muscle 4: Triceps Brachii Lateral and Medial; Anconeus; and Extensor Carpi Radialis Brevis and Ulnaris.

   Muscle 5: Biceps Long and Short.

   Muscle 6: Triceps Brachii Long.

2. Calculate muscle moment arms about the three revolute shoulder joints and one revolute elbow joint for a specific type of motion derived from inverse kinematics

analysis in OpenSim. Evaluate the shoulder moment arm in the horizontal plane ($\boldsymbol{\mathscr{r}}^{S}_{\mathscr{m},\mathscr{M}}$).

$$\begin{aligned}\boldsymbol{\mathscr{r}}^{S}_{\mathscr{m},\mathscr{M}} &= \mathbf{R}^{O}_{\mathscr{j}=1}\big\rfloor_{(2,2)}\,\boldsymbol{\mathscr{r}}^{\mathscr{j}=1}_{\mathscr{m},\mathscr{M}} + \mathbf{R}^{O}_{\mathscr{j}=2}\big\rfloor_{(2,3)}\,\boldsymbol{\mathscr{r}}^{\mathscr{j}=2}_{\mathscr{m},\mathscr{M}} + \mathbf{R}^{O}_{\mathscr{j}=3}\big\rfloor_{(2,2)}\,\boldsymbol{\mathscr{r}}^{\mathscr{j}=3}_{\mathscr{m},\mathscr{M}} \\ &= \sum_{\mathscr{j}=1}^{3} \mathbf{R}^{O}_{\mathscr{j}}\big\rfloor_{(2,3-|2-\mathscr{j}|)}\,\boldsymbol{\mathscr{r}}^{\mathscr{j}}_{\mathscr{m},\mathscr{M}} \end{aligned} \tag{4.5}$$

where $\boldsymbol{\mathscr{r}}^{\mathscr{j}}_{\mathscr{m},\mathscr{M}}$ is the muscle moment arm about the corresponding $\mathscr{j}^{\text{th}}$ joint. Subscripts $\mathscr{m}$ and $\mathscr{M}$ denote muscle number and its corresponding muscle group number, respectively. $\mathbf{R}^{O}_{\mathscr{j}}\big\rfloor_{(i,j)}$ is the $(i,j)^{\text{th}}$ element of the rotation matrix converting the joint $\mathscr{j}$ local frame to the origin frame. It is worth noting that since the shoulder joint rotation is defined as a YZY rotation (Y: $j=2$; Z: $j=3$) and the goal is to find moment arm about the global Y_O axis ($i=2$), in the above equation following properties are implemented:

- For the first joint ($\mathscr{j}=1$) which corresponds to Y axis ($j=2$), the $(i=2, j=2)^{\text{th}}$ element of the rotation matrix ($\mathbf{R}^{O}_{\mathscr{j}=1}$) is selected.
- For the second joint ($\mathscr{j}=2$) which corresponds to Z axis ($j=3$), the $(i=2, j=3)^{\text{th}}$ element of the rotation matrix ($\mathbf{R}^{O}_{\mathscr{j}=2}$) is selected.
- For the third joint ($\mathscr{j}=3$) which corresponds to Y axis ($j=2$), the $(i=2, j=2)^{\text{th}}$ element of the rotation matrix ($\mathbf{R}^{O}_{\mathscr{j}=3}$) is selected.

3. Evaluate the corresponding maximum shoulder and elbow rotation torques for individual muscles in the horizontal plane.

$$\begin{cases} T^{S}_{\mathscr{m},\mathscr{M}} = \boldsymbol{\mathscr{r}}^{S}_{\mathscr{m},\mathscr{M}} F^{\text{iso}}_{\mathscr{m},\mathscr{M}} \\ T^{E}_{\mathscr{m},\mathscr{M}} = \boldsymbol{\mathscr{r}}^{E}_{\mathscr{m},\mathscr{M}} F^{\text{iso}}_{\mathscr{m},\mathscr{M}} \end{cases} \tag{4.6}$$

where $\boldsymbol{\mathscr{r}}^{E}_{\mathscr{m},\mathscr{M}} = \boldsymbol{\mathscr{r}}^{\mathscr{j}=4}_{\mathscr{m},\mathscr{M}}$ is the elbow moment arm in the horizontal plane.

4. Calculate the corresponding maximum shoulder and elbow torques for the muscle groups in the horizontal plane (4.7). Then, evaluate the muscle group moment arms (4.8).

$$\begin{cases} T^{S}_{\mathscr{M}} = \sum_{\mathscr{m}=1}^{n_{\mathscr{m}}} T^{S}_{\mathscr{m},\mathscr{M}} \\ T^{E}_{\mathscr{M}} = \sum_{\mathscr{m}=1}^{n_{\mathscr{m}}} T^{E}_{\mathscr{m},\mathscr{M}} \end{cases} \tag{4.7}$$

$$\begin{cases} \boldsymbol{\mathscr{r}}^{S}_{\mathscr{M}} = \frac{\sum_{\mathscr{m}=1}^{n_{\mathscr{m}}} \boldsymbol{\mathscr{r}}^{S}_{\mathscr{m},\mathscr{M}} T^{S}_{\mathscr{m},\mathscr{M}}}{T^{S}_{\mathscr{M}}} \\ \boldsymbol{\mathscr{r}}^{E}_{\mathscr{M}} = \frac{\sum_{\mathscr{m}=1}^{n_{\mathscr{m}}} \boldsymbol{\mathscr{r}}^{E}_{\mathscr{m},\mathscr{M}} T^{E}_{\mathscr{m},\mathscr{M}}}{T^{E}_{\mathscr{M}}} \end{cases} \tag{4.8}$$

Table 4.3: Muscle mechanics parameters for MA2D-I_3.

Parameter	Unit	Muscle1	Muscle2	Muscle3	Muscle4	Muscle5	Muscle6
F^{iso}	N	2525	1672	1452	1577	972	799
L_0^{M}	cm	12	21	11.7	10.9	12.9	14.3
L_{s}^{T}	cm	2.9	0	1.8	0.7	18.8	11.9
α_0^{M}	deg	21.6	19.5	1.4	7.8	0	12

5. Approximate the muscle group insertion points by running an optimization on the origin and insertion locations of the 3D model in a way that a straight line path model generates similar moment arms over a period of inverse kinematics motion (the kinematic path is defined in Section 4.3 (Cross-validation studies of human models)).

6. Evaluate the muscle group model parameters ($\mathcal{p}$).

$$\mathcal{p}_{\mathscr{M}} = \begin{cases} \frac{\int_t \sum_{m=1}^{n_m} \mathcal{p}^S_{m,\mathscr{M}} T^S_{m,\mathscr{M}}}{\int_t T^S_{\mathscr{M}}} & \mathscr{M} = 1, 2 \\ \frac{\int_t \sum_{m=1}^{n_m} \mathcal{p}^E_{m,\mathscr{M}} T^E_{m,\mathscr{M}}}{\int_t T^E_{\mathscr{M}}} & \mathscr{M} = 3, 4 \\ \frac{1}{2}\left(\frac{\int_t \sum_{m=1}^{n_m} \mathcal{p}^S_{m,\mathscr{M}} T^S_{m,\mathscr{M}}}{\int_t T^S_{\mathscr{M}}} + \frac{\int_t \sum_{m=1}^{n_m} \mathcal{p}^E_{m,\mathscr{M}} T^E_{m,\mathscr{M}}}{\int_t T^E_{\mathscr{M}}}\right) & \mathscr{M} = 5, 6 \end{cases} \tag{4.9}$$

   where t is the simulation time, and $\mathcal{p}_{\mathscr{M}}$ denotes the parameters listed in Table 4.3.

7. Since we are not including muscle wrapping, negative values of the tendon slack lengths of the muscle groups are set to zero so that the value of the muscle group musculotendon length becomes meaningful.

Finally, muscle attachment points are updated as in Figure 4.3.b and the new muscle model parameters are summarized in Table 4.3.

4.1.5 MA3D-II

MA3D-I is based on the Stanford model. However, the shortcomings of this model led us to develop an advanced 3D musculoskeletal arm model. Defining muscle-wrapping geometries requires a proper database. The Texas upper extremity model has a complete muscle path definition based on the obstacle-set method. This model is defined in different coordinates

Table 4.4: MA3D-II specifications

# Muscles (Elements)	26 (42)
Muscle mechanics model	CE/PE/SE Hill-type
Muscle path model	Obstacle-set method $\longrightarrow$ Non-geometric path model
Muscle redundancy problem	SO/FSO
Musculoskeletal model type	3D/ Dynamic
Shoulder	Spherical joint (triple revolute joints)
Shoulder rhythm	Regression
Shoulder stability	N/A
Elbow	Double revolute joints (flexion/extension and pronation/supination)

than the ISB recommendation, and it does not account for the motion of the shoulder joint (shoulder rhythm). In this subsection, the goal is to define the Texas model using the ISB recommended coordinates with the shoulder rhythm determined by the regression-based approach of Xu et al. [326]. The Lisbon model is the Texas model defined in ISB recommended coordinates except for the scapula coordinate. This redefinition has some deficiencies, and we will overcome those imperfections by redefining the ISB recommended coordinates, proper positioning of the clavicle and scapula, and defining non-geometric muscle path which is required for real-time simulations. A brief overview of the model is presented in Table 4.4. Next, skeletal modeling, muscle model and implementation for this model will be covered.

Skeletal Model

ISB recommendation is based on some specific palpable bony landmarks, but some of these bony landmarks are not available in the Texas model study (see Table 4.5). We use a specific morphing technique to find these points based on the MA3D-I skeletal model. Matias et al. [177] state that, it is possible to find muscle attachment sites with a transformation defined by three bony landmarks. We will use this method to estimate the other unknown bony landmarks on each bony segment of the Texas model. Hence, the following transformations are used to determine the unknown palpable bony landmarks along with skeletal segment visualizations:

1. Thorax: define 3 vectors out of 4 bony landmarks of the thorax for both MA3D-I

Table 4.5: Anatomical features (which includes bony landmarks) of the ISB recommendation, MA3D-I, and MA3D-II. ✓: it is available, ✗: it is not required, ?: it is not available and should be evaluated.

Segment	Anatomical Feature	Description	ISB	MA3D-I	MA3D-II
Thorax	C7	Spinous process of the 7th cervical vertebra	✓	✓	✓
	T8	Spinal process of the 8th thoracic vertebra	✓	✓	✓
	IJ	Deepest point of Incisura Jugularis	✓	✓	✓
	PX	Processus Xiphoideus	✓	✓	✓
Clavicle	SC	SCJ	✓	✓	✓
	AC	ACJ	✓	✓	✓
Scapula	TS	Trigonum Spinae Scapulae	✓	✓	?
	AI	Angulus Inferior	✓	✓	✓
	AA	Angulus Acromialis	✓	✓	?
	PC	Most ventral point of processus coracoideus	✓	✓	✓
	SM	Superior point on ellipsoid	✗	✓	✓
	IM	Inferior point on ellipsoid	✗	✓	✓
Humerus	GH	GHJ	✓	✓	✓
	EL	Most caudal point on lateral epicondyle	✓	✓	?
	EM	Most caudal point on medial epicondyle	✓	✓	?
	$\mathbf{a}_{\mathrm{HL}}$	Long axis of humerus	✗	✓	✓
	HU	HUJ	✗	✓	✓
	$\mathbf{a}_{\mathrm{HU}}$	HUJ axis	✗	✓	✓
Forearm	US	Most caudalmedial point on the ulnar styloid	✓	✓	?
	USC	Center of ulnar styloid process	✗	✓	✓
	RU	DRUJ	✗	✓	✓
	RS	Most caudallateral point on the radial styloid	✓	✓	?
	RSC	Center of radial styloid process	✗	✓	✓
	RC	RCJ	✗	✓	✓

and MA3D-II, and build the thorax transformation matrix as follows:

$$\mathfrak{T}_t = \begin{bmatrix} \mathbf{r}_{\mathrm{T8}} - \mathbf{r}_{\mathrm{C7}} & \mathbf{r}_{\mathrm{IJ}} - \mathbf{r}_{\mathrm{C7}} & \mathbf{r}_{\mathrm{PX}} - \mathbf{r}_{\mathrm{C7}} \end{bmatrix}_{\mathrm{II}} \begin{bmatrix} \mathbf{r}_{\mathrm{T8}} - \mathbf{r}_{\mathrm{C7}} & \mathbf{r}_{\mathrm{IJ}} - \mathbf{r}_{\mathrm{C7}} & \mathbf{r}_{\mathrm{PX}} - \mathbf{r}_{\mathrm{C7}} \end{bmatrix}_{\mathrm{I}}^{-1} \tag{4.10}$$

where $\mathbf{r}$ is the position vector defined in each model's global coordinate system.

2. Clavicle: define the local coordinate of the clavicle for both MA3D-I and MA3D-II, and get the scaling factor by measuring AC to SC length. Then, the following transformation matrix is used to generate the skeletal visualization for the clavicle:

$$\mathfrak{T}_c = \frac{\|\mathbf{r}_{\mathrm{AC}} - \mathbf{r}_{\mathrm{SC}}\|_{\mathrm{II}}}{\|\mathbf{r}_{\mathrm{AC}} - \mathbf{r}_{\mathrm{SC}}\|_{\mathrm{I}}} \mathbf{R}_{c,\mathrm{II}} \mathbf{R}_{c,\mathrm{I}}^{-1} \tag{4.11}$$

where $\mathbf{R}_{c,\mathrm{I}}$ and $\mathbf{R}_{c,\mathrm{II}}$ are the ISB recommended clavicle rotation matrices of the MA3D-I and MA3D-II, respectively, and $\mathfrak{T}_c$ is the clavicle transformation matrix.

3. Scapula: define 3 vectors out of 3 bony landmarks of the scapula for both MA3D-I and MA3D-II, and build the scapula transformation matrix as:

$$\mathfrak{T}_s = \left[\begin{array}{ccc} \mathbf{r}_{\mathrm{AC}} & \mathbf{r}_{\mathrm{AI}} & \mathbf{r}_{\mathrm{SM}} \end{array} \right]_{\mathrm{II}} \left[\begin{array}{ccc} \mathbf{r}_{\mathrm{AC}} & \mathbf{r}_{\mathrm{AI}} & \mathbf{r}_{\mathrm{SM}} \end{array} \right]_{\mathrm{I}}^{-1} \tag{4.12}$$

Then, the unknown bony landmarks of the scapula can be estimated by the following equation:

$$\begin{cases} \mathbf{r}_{\mathrm{TS}}^{\mathrm{II}} = \mathfrak{T}_s \mathbf{r}_{\mathrm{TS}}^{\mathrm{I}} \\ \mathbf{r}_{\mathrm{AA}}^{\mathrm{II}} = \mathfrak{T}_s \mathbf{r}_{\mathrm{AA}}^{\mathrm{I}} \end{cases} \tag{4.13}$$

4. Humerus: define 3 independent axes in MA3D-I and MA3D-II, and get the scaling factor by measuring HU to GH length. Then, the following transformation matrix is used to generate the skeletal visualization for the humerus:

$$\mathfrak{T}_h = \frac{\left\| \mathbf{r}_{\mathrm{HU}} - \mathbf{r}_{\mathrm{GH}} \right\|_{\mathrm{II}}}{\left\| \mathbf{r}_{\mathrm{HU}} - \mathbf{r}_{\mathrm{GH}} \right\|_{\mathrm{I}}} \left[\begin{array}{ccc} \frac{\mathbf{r}_{\mathrm{HU}} - \mathbf{r}_{\mathrm{GH}}}{\left\| \mathbf{r}_{\mathrm{HU}} - \mathbf{r}_{\mathrm{GH}} \right\|} & \mathbf{a}_{\mathrm{HU}} & \mathbf{a}_{\mathrm{HL}} \end{array} \right]_{\mathrm{II}} \left[\begin{array}{ccc} \frac{\mathbf{r}_{\mathrm{HU}} - \mathbf{r}_{\mathrm{GH}}}{\left\| \mathbf{r}_{\mathrm{HU}} - \mathbf{r}_{\mathrm{GH}} \right\|} & \mathbf{a}_{\mathrm{HU}} & \mathbf{a}_{\mathrm{HL}} \end{array} \right]_{\mathrm{I}}^{-1} \tag{4.14}$$

where $\mathfrak{T}_h$ is the humerus transformation matrix. The unknown bony landmarks of the humerus can be estimated by the following equation:

$$\begin{cases} \mathbf{r}_{\mathrm{EL}}^{\mathrm{II}} = \mathbf{r}_{\mathrm{GH}}^{\mathrm{II}} + \mathfrak{T}_h \left(\mathbf{r}_{\mathrm{EL}}^{\mathrm{I}} - \mathbf{r}_{\mathrm{GH}}^{\mathrm{I}} \right) \\ \mathbf{r}_{\mathrm{EM}}^{\mathrm{II}} = \mathbf{r}_{\mathrm{GH}}^{\mathrm{II}} + \mathfrak{T}_h \left(\mathbf{r}_{\mathrm{EM}}^{\mathrm{I}} - \mathbf{r}_{\mathrm{GH}}^{\mathrm{I}} \right) \end{cases} \tag{4.15}$$

5. Ulna: define 3 independent axes in MA3D-I and MA3D-II, and get the scaling factor by measuring US[2] to HU length. Then, the following transformation matrix is used to generate the skeletal visualization for the humerus:

$$\begin{aligned} \mathfrak{T}_u = \frac{\left\| \mathbf{r}_{\mathrm{US}} - \mathbf{r}_{\mathrm{HU}} \right\|_{\mathrm{II}}}{\left\| \mathbf{r}_{\mathrm{US}} - \mathbf{r}_{\mathrm{HU}} \right\|_{\mathrm{I}}} & \left[\begin{array}{ccc} \frac{\mathbf{r}_{\mathrm{US}} - \mathbf{r}_{\mathrm{HU}}}{\left\| \mathbf{r}_{\mathrm{US}} - \mathbf{r}_{\mathrm{HU}} \right\|} & \mathbf{a}_{\mathrm{HU}} & \frac{\mathbf{r}_{\mathrm{US}} - \mathbf{r}_{\mathrm{HU}}}{\left\| \mathbf{r}_{\mathrm{US}} - \mathbf{r}_{\mathrm{HU}} \right\|} \times \mathbf{a}_{\mathrm{HU}} \end{array} \right]_{\mathrm{II}} \\ & \cdot \left[\begin{array}{ccc} \frac{\mathbf{r}_{\mathrm{US}} - \mathbf{r}_{\mathrm{HU}}}{\left\| \mathbf{r}_{\mathrm{US}} - \mathbf{r}_{\mathrm{HU}} \right\|} & \mathbf{a}_{\mathrm{HU}} & \frac{\mathbf{r}_{\mathrm{US}} - \mathbf{r}_{\mathrm{HU}}}{\left\| \mathbf{r}_{\mathrm{US}} - \mathbf{r}_{\mathrm{HU}} \right\|} \times \mathbf{a}_{\mathrm{HU}} \end{array} \right]_{\mathrm{I}}^{-1} \end{aligned} \tag{4.16}$$

where $\mathfrak{T}_u$ is the ulna transformation matrix.

6. Radius: define 3 independent axes in MA3D-I and MA3D-II, and get the scaling factor by measuring RS[3] to RU length. Then, the following transformation matrix

[2]US is estimated based on USC [328].
[3]RS is assumed to be same as RSC.

Table 4.6: Workspace of MA3D-II in terms of its joint angles. Note that $\mu = \frac{\max+\min}{2}$ and $\sigma = \frac{\max-\min}{2}$.

$\mathcal{J}$	Joint Angle	Description	min (deg)	max (deg)	μ (deg)	σ (deg)
1	γ_{GH1}	GH plane of elevation about Y axis	-20	120	50	70
2	β_{GH}	GH negative elevation about X axis	-90	-30	-60	30
3	γ_{GH2}	GH-axial rotation about Y axis	-60	30	-15	45
4	α_{HU}	HU flexion/extension about Z axis	0	140	70	70
5	γ_{RU}	RU pronation/supination about Y axis	0	120	60	60

is used to generate the skeletal visualization for the humerus:

$$\mathfrak{T}_r = \frac{\|\mathbf{r}_{\text{RS}} - \mathbf{r}_{\text{RU}}\|_{\text{II}}}{\|\mathbf{r}_{\text{RS}} - \mathbf{r}_{\text{RU}}\|_{\text{I}}} \begin{bmatrix} \frac{\mathbf{r}_{\text{RS}}-\mathbf{r}_{\text{RU}}}{\|\mathbf{r}_{\text{RS}}-\mathbf{r}_{\text{RU}}\|} & \mathbf{a}_{\text{RU}} & \frac{\mathbf{r}_{\text{RS}}-\mathbf{r}_{\text{RU}}}{\|\mathbf{r}_{\text{RS}}-\mathbf{r}_{\text{RU}}\|} \times \mathbf{a}_{\text{RU}} \end{bmatrix}_{\text{II}} \cdot \begin{bmatrix} \frac{\mathbf{r}_{\text{RS}}-\mathbf{r}_{\text{RU}}}{\|\mathbf{r}_{\text{RS}}-\mathbf{r}_{\text{RU}}\|} & \mathbf{a}_{\text{RU}} & \frac{\mathbf{r}_{\text{RS}}-\mathbf{r}_{\text{RU}}}{\|\mathbf{r}_{\text{RS}}-\mathbf{r}_{\text{RU}}\|} \times \mathbf{a}_{\text{RU}} \end{bmatrix}_{\text{I}}^{-1} \tag{4.17}$$

where $\mathfrak{T}_r$ is the radius transformation matrix.

The ISB recommended coordinates can be defined according to the defined bony landmarks. Driving joints and the entire workspace of MA3D-II in terms of its driving joint angle limits are defined in Table 4.6. We utilize regression formulas to implement the shoulder rhythm. Considering Table 4.7, the equations are in the form of:

$$\begin{aligned}\theta =& (\gamma_{\text{GH1}} - 46.97)c_1 + (\beta_{\text{GH}} + 66.46)c_2 + (\gamma_{\text{GH2}} + 37.64)c_3 + (\gamma_{\text{GH1}} - 46.97)^2 c_4 \\ &+ (\beta_{\text{GH}} + 66.46)^2 c_5 + (\gamma_{\text{GH2}} + 37.64)^2 c_6 + (\gamma_{\text{GH1}} - 46.97)(\beta_{\text{GH}} + 66.46)c_7 \\ &+ (\gamma_{\text{GH1}} - 46.97)(\gamma_{\text{GH2}} + 37.64)c_8 + (\beta_{\text{GH}} + 66.46)(\gamma_{\text{GH2}} + 37.64)c_9 + c_0 \end{aligned} \tag{4.18}$$

where θ indicates the corresponding joint angle. All the angles are measured in degrees.

Muscle model

Muscle dynamics is included using the Hill-type definition presented in [283]. With rigid tendon assumption, the muscle force is evaluated using the following equation:

$$F_m^{\text{M}} = \left[F^{\text{CE}}\left(a_m, F_m^{\text{iso}}, L_m, V_m\right) + F^{\text{PE}}\left(F_m^{\text{iso}}, L_m\right)\right]\cos(\alpha_m^{\text{M}}) \tag{4.19}$$

where $F^{\text{PE}}(.)$ is the PE function. See Appendix B for more details.

Table 4.7: The regression coefficients of the shoulder rhythm.

θ	Description	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8	c_9	c_0
γ_{AC}	Scapula retraction/ protraction about Y axis	0.163	0	0.039	-0.0016	-0.0018	-0.0003	-0.0023	-0.0009	0.0003	38.35
β_{AC}	Scapula lateral/medial rotation about X axis	-0.065	0.322	-0.024	0	- 0.0009	0	0	-0.0014	0	-23.20
α_{AC}	Scapula anterior/ posterior tilt about Z axis	0.060	-0.039	-0.011	0	0	0.0002	0	0.0005	0.0008	-7.11
γ_{SC}	Clavicle retraction/ protraction about Y axis	0.059	0.207	0.013	-0.0017	-0.0025	-0.0005	-0.0020	-0.0020	0	-17.42
β_{SC}	Clavicle elevation/ depression about X axis	-0.025	0.204	-0.031	0	0	0.0002	-0.0007	-0.0003	0.0007	-21.04

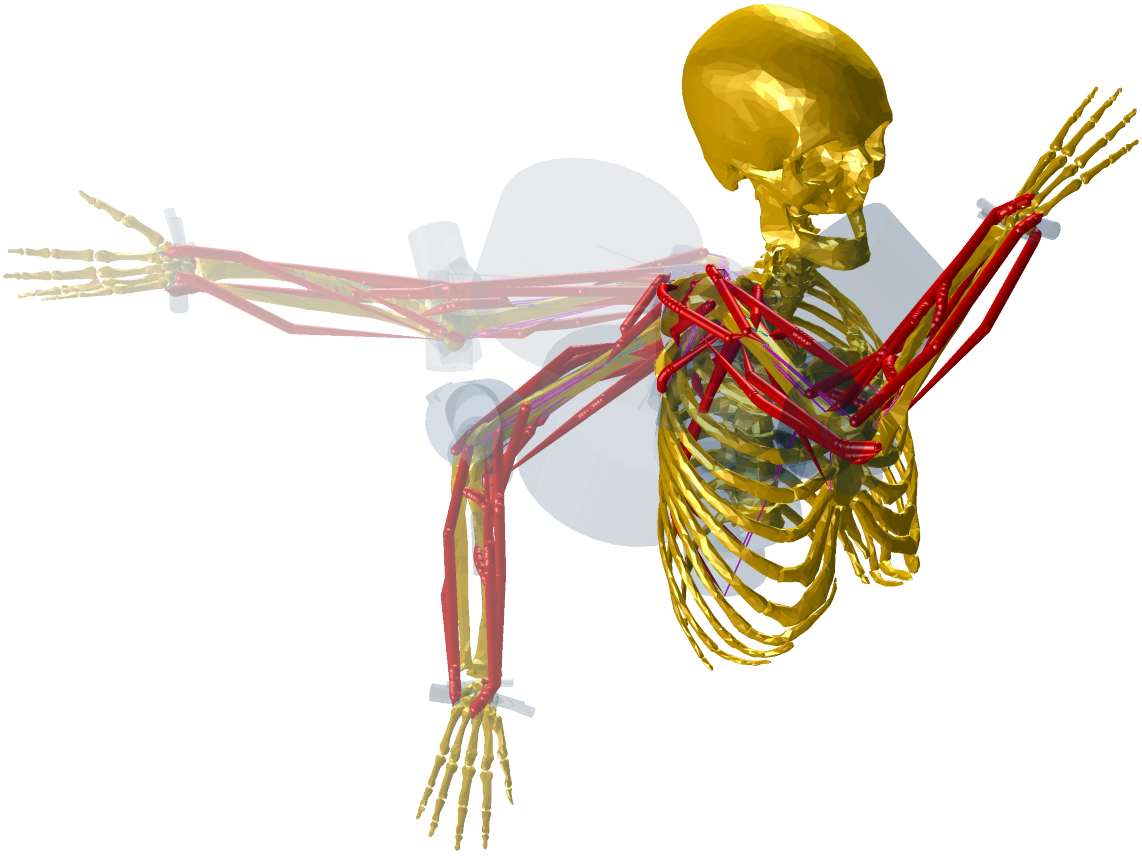


Figure 4.4: MA3D-II skeletal model and muscle wrapping at 3 different postures.

The Texas musculoskeletal model uses the obstacle-set method for muscle path; these muscle path models should be redefined in ISB recommended coordinates (see Fig. 4.4). For real-time simulations, non-geometric muscle path models are required. Thus, for every muscle that acts on specific joints, the corresponding position inputs are given to the active joints to get the muscle length[4]. These inputs are summarized in Table 4.8. Then, the muscle lengths are fitted with fifth-order polynomials of the active joint angles. Based on the virtual work principle, the muscle moment arm for a single muscle can be evaluated using the following equation:

$$\left.\begin{array}{l} T_m^j = F_m^{\mathrm{M}} r_m^j \\ T_m^j \delta\theta_j = -F_m^{\mathrm{M}} \delta L^m \end{array}\right\} \Longrightarrow r_m^j = -\frac{\delta L^m}{\delta \theta_j} = -\frac{\partial L^m}{\partial \theta_j} \tag{4.20}$$

where r_m^j is the m^{th} muscle moment arm about joint j that the muscle acts on.

The muscle force sharing problem can be solved by SO and FSO, with cost functions similar to (4.3) and (4.1).

Implementation

The model is developed in MapleSim™; then, its optimized C-code and MATLAB®/Simulink block are generated for simulations.

4.1.6 MA2D-II

To generate this model, a procedure similar to MA2D-I_3 is performed. However, the following changes are implemented in the MA2D-II development:

1. MA3D-II interacts with the rehabilitation robot. With a specified hand orientation (in two directions) on the end-effector, the best position for the MA3D-II placement is determined when there is intersection between the human and robot workspaces. The hand of MA3D-II in this configuration follows a persistently exciting input and the hand force is recorded. In an optimization, the best position of GHJ, upper and lower arm lengths for the 2D model is determined subject to holding the MA2D-II hand in the same position as the MA3D-II hand during persistently exciting simulations.

[4]Note that since we have used regression based shoulder rhythm, including SC and AC muscles is not required (i.e., 13 muscles of the clavicle and scapula are not included in this model).

Table 4.8: Inputs for muscle length evaluation. Please refer to Table. 4.6 for the unknown parameters.

Joints	# Muscles	Frequency (Ω)	Inputs (which are measured in degrees)	t_f (s)
GH	15	$\frac{\pi}{200}$	$\begin{cases} \gamma_{\mathrm{GH1}} = \gamma_{\mathrm{GH1}\mu} - \gamma_{\mathrm{GH1}\sigma}\cos(\Omega t) \\ \beta_{\mathrm{GH}} = \beta_{\mathrm{GH}\mu} - \beta_{\mathrm{GH}\sigma}\cos(10\Omega t) \\ \gamma_{\mathrm{GH2}} = \gamma_{\mathrm{GH2}\mu} - \gamma_{\mathrm{GH2}\sigma}\cos(100\Omega t) \end{cases}$	200
GH+HU	1	$\frac{\pi}{2000}$	$\begin{cases} \gamma_{\mathrm{GH1}} = \gamma_{\mathrm{GH1}\mu} - \gamma_{\mathrm{GH1}\sigma}\cos(\Omega t) \\ \beta_{\mathrm{GH}} = \beta_{\mathrm{GH}\mu} - \beta_{\mathrm{GH}\sigma}\cos(10\Omega t) \\ \gamma_{\mathrm{GH2}} = \gamma_{\mathrm{GH2}\mu} - \gamma_{\mathrm{GH2}\sigma}\cos(100\Omega t) \\ \alpha_{\mathrm{HU}} = \alpha_{\mathrm{HU}\mu} - \alpha_{\mathrm{HU}\sigma}\cos(1000\Omega t) \end{cases}$	2000
GH+HU+RU	2	$\frac{\pi}{20000}$	$\begin{cases} \gamma_{\mathrm{GH1}} = \gamma_{\mathrm{GH1}\mu} - \gamma_{\mathrm{GH1}\sigma}\cos(\Omega t) \\ \beta_{\mathrm{GH}} = \beta_{\mathrm{GH}\mu} - \beta_{\mathrm{GH}\sigma}\cos(10\Omega t) \\ \gamma_{\mathrm{GH2}} = \gamma_{\mathrm{GH2}\mu} - \gamma_{\mathrm{GH2}\sigma}\cos(100\Omega t) \\ \alpha_{\mathrm{HU}} = \alpha_{\mathrm{HU}\mu} - \alpha_{\mathrm{HU}\sigma}\cos(1000\Omega t) \\ \gamma_{\mathrm{RU}} = \gamma_{\mathrm{RU}\mu} - \gamma_{\mathrm{RU}\sigma}\cos(10000\Omega t) \end{cases}$	20000
HU	3	$\frac{\pi}{20}$	$\alpha_{\mathrm{HU}} = \alpha_{\mathrm{HU}\mu} - \alpha_{\mathrm{HU}\sigma}\cos(\Omega t)$	20
HU+RU	7	$\frac{\pi}{20}$	$\begin{cases} \alpha_{\mathrm{HU}} = \alpha_{\mathrm{HU}\mu} - \alpha_{\mathrm{HU}\sigma}\cos(\Omega t) \\ \gamma_{\mathrm{RU}} = \gamma_{\mathrm{RU}\mu} - \gamma_{\mathrm{RU}\sigma}\cos(10\Omega t) \end{cases}$	20
RU	1	$\frac{\pi}{20}$	$\gamma_{\mathrm{RU}} = \gamma_{\mathrm{RU}\mu} - \gamma_{\mathrm{RU}\sigma}\cos(\Omega t)$	20

2. After finding the optimal position of the shoulder joint and lengths of upper and lower arms, the 2D joint angles (velocities and accelerations) of the shoulder and elbow are evaluated in an inverse kinematic analysis. Then, by an inverse dynamics analysis, dynamic parameters (masses, moments of inertia, COMs, joint damping coefficients) of MA2D-II are determined.

3. Muscle lengths are evaluated in the persistently exciting simulations, and they are determined with a fifth-order polynomial function in terms of the MA2D-II joint angles. Then, muscle moment arms are calculated by (4.20).

4. To define muscle groups based on their synergistic muscles, the signs of muscle moment arms are considered for categorizing the muscles. Consequently, the following muscle groups are resulted:

   Muscle 1: Anterior (Clavicular) Deltoid; Clavicular, Sternal and Ribs Pectoralis Major; Subscapularis; and Coracobrachialis.

Table 4.9: Muscle mechanics parameters for MA2D-II.

Parameter	Unit	Muscle1	Muscle2	Muscle3	Muscle4	Muscle5	Muscle6
F^{iso}	N	450	708	595	1000	434	629
L_0^{M}	cm	17.4	15.7	10.2	5.6	13.7	15.2
L_s^{T}	cm	4.9	7.2	11.5	18.2	23	19
α_0^{M}	deg	0	0	13.8	15.1	10	15

Muscle 2: Posterior (Scapular) and Middle* (Acromial) Deltoid; Thoracic, Lumbar and Iliac Latissimus Dorsi; Supraspinatus; Infraspinatus; and Teres Minor and Major.

Muscle 3: Brachioradialis; Brachialis; Pronator Teres; Flexor Carpi Radialis; and Extensor Carpi Radialis Brevis* and Longus.

Muscle 4: Triceps Brachii Lateral and Medial; Supinator; Flexor Carpi Ulnaris; Extensor Carpi Radialis Ulnaris.

Muscle 5: Biceps Long and Short.

Muscle 6: Triceps Brachii Long.

Underlined muscles (10 in total) are supplementary to the MA3D-I muscles, while the Anconeus muscle is not included in the MA3D-II muscle list. Starred muscles are defined in different muscle groups compared to the MA3D-I, which indicates that Nijhof and Kouwenhoven [197] have not categorized these muscles properly.

Finally, the musculoskeletal model of MA2D-II with visual muscle wrapping (which is developed using muscle length polynomial functions) is developed in MapleSim™(see Fig. 4.5). The new muscle model parameters are summarized in Table 4.9.

4.2 Human-robot interaction models

To develop an HRI model, both systems are modeled separately and then integrated into a single model. To model these systems, the MapleSim™software package is used for the following reasons:

1. Multi-domain capabilities: mechanical system support for modeling a parallelogram linkage, electrical system support to model DC motors, and biological system support for modeling the muscles and musculoskeletal system,

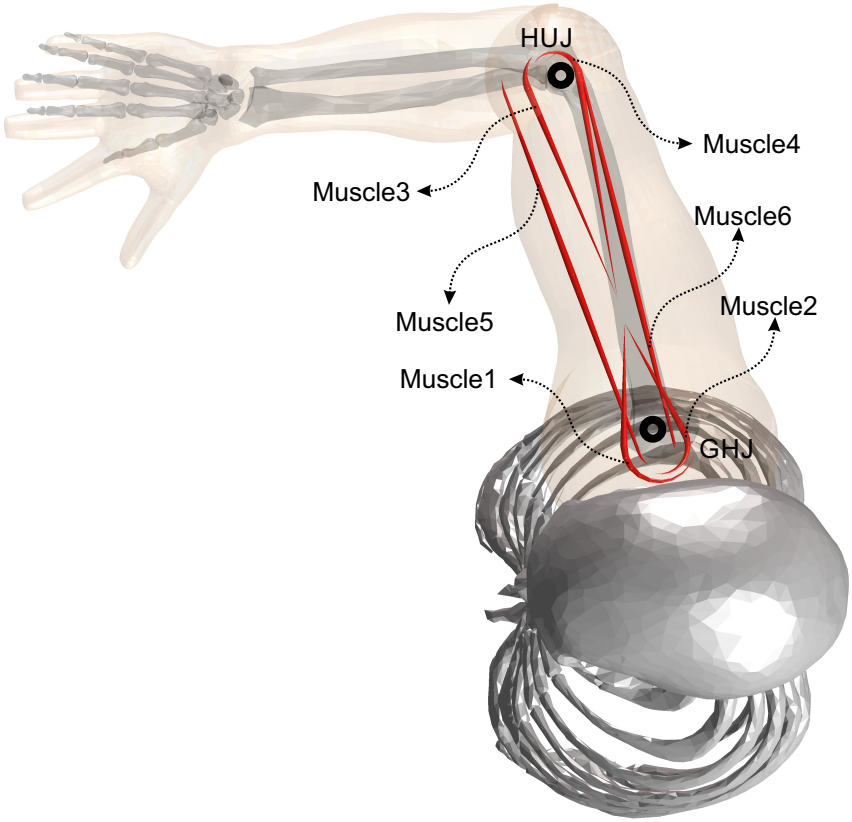


Figure 4.5: MA2D-II musculoskeletal model. See Table A.10 for the dynamic parameters.

2. Symbolic processing,
3. Optimized code generation,
4. CAD support.

The robto's end-effector consists of a passive revolute joint (along the Y axis) and a force sensor. Thus, a revolute joint (along the Y axis) which is equipped with a force sensor is added to the original robot's model; then, the developed models of two systems are connected to each other by a rigid connection. The integrated model is a closed-chain linkage with multiple inputs which includes: 2 motor inputs, and $n_{\mathscr{M}}$ muscle activations for the 2D musculoskeletal models or n_m muscle activations for the 3D musculoskeletal models. This closed-chain linkage has 4 outputs including 2 robot joint angles ($q_{1,2}$) and 2 force sensor outputs ($F_{extZ,X}$).

Here for an efficient model-based control implementation, MA2D-II integration with the robot model is discussed.

4.2.1 Review of the robot kinematics and dynamics

The end-effector kinematics are:

$$\begin{cases} \boldsymbol{\rho}_R = \boldsymbol{\Phi}_R(\mathbf{q}) \\ \dot{\boldsymbol{\rho}}_R = \frac{\partial \boldsymbol{\Phi}_R}{\partial \mathbf{q}} \dot{\mathbf{q}} = \mathbf{J}_R \dot{\mathbf{q}} \\ \ddot{\boldsymbol{\rho}}_R = \dot{\mathbf{J}}_R \dot{\mathbf{q}} + \mathbf{J}_R \ddot{\mathbf{q}} \end{cases} \tag{4.21}$$

where $\boldsymbol{\rho}_R$ is the end-effector position in the global coordinates {Z,X}, and $\mathbf{q} = [q_1, q_2]^T$ is the vector of the robot joint angles. $\mathbf{J}_R$ is the robot geometric Jacobian, and $\boldsymbol{\Phi}_R$ is the kinematic function of the robot. The dynamic model of the system is:

$$\begin{aligned} \mathbf{T}_R - \mathbf{J}_R^T \mathbf{F}_{int} &= \mathbf{M}_R(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}_R(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{K}_R(\mathbf{q} - \mathbf{q}_0) + \mathbf{f}_T + \mathbf{J}_R^T \mathbf{f}_F \\ &= \boldsymbol{\Gamma}_R(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}) \end{aligned} \tag{4.22}$$

where $\mathbf{T}_R$ is the vector of robot motor torques, and $\mathbf{F}_{int}$ and $\mathbf{f}_F$ are robot to human interaction force and friction force under the end-effector in the global coordinates, respectively. $\mathbf{M}_R$ is the robot inertia (mass) matrix, and $\mathbf{C}_R$ is the robot Coriolis-centrifugal matrix. $\mathbf{f}_T$ is the friction torque vector at the joints. $\mathbf{K}_R$ is a 2×2 symmetric joint stiffness matrix, and $\mathbf{q}_0$ is the equilibrium position of the driving joint angles.

4.2.2 Review of the MA2D-II kinematics and dynamics

As it was mentioned in MA2D-II development, this model is mapped from the high-fidelity musculoskeletal model (i.e., MA3D-II). MA3D-II is placed in front of the robot at different heights (sitting positions); then, the mapping is done on a plane that has maximal interference between the robot and MA3D-II workspaces (see Fig. 4.6). The MA2D-II kinematics are:

$$\begin{cases} \boldsymbol{\rho}_A = \boldsymbol{\Phi}_A(\boldsymbol{\theta}) \Rightarrow \boldsymbol{\theta} = \boldsymbol{\Psi}_A(\mathbf{r}_A) \\ \dot{\boldsymbol{\rho}}_A = \frac{\partial \boldsymbol{\Phi}_A}{\partial \boldsymbol{\theta}} \dot{\boldsymbol{\theta}} = \mathbf{J}_A \dot{\boldsymbol{\theta}} \\ \ddot{\boldsymbol{\rho}}_A = \dot{\mathbf{J}}_A \dot{\boldsymbol{\theta}} + \mathbf{J}_A \ddot{\boldsymbol{\theta}} \end{cases} \tag{4.23}$$

where $\boldsymbol{\rho}_A$ is the hand position in the global coordinates, and $\boldsymbol{\theta} = [\theta_S, \theta_E]^T$ is the vector of the shoulder and elbow joint angles. $\mathbf{J}_A$ is the MA2D-II geometric Jacobian, $\boldsymbol{\Phi}_A$ is

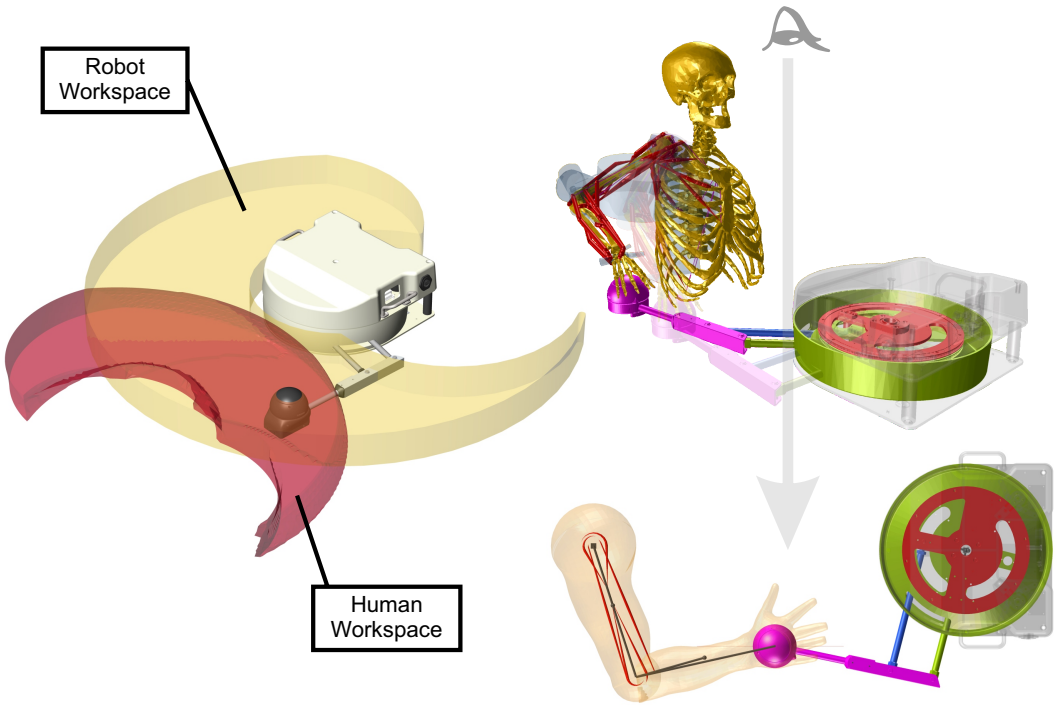


Figure 4.6: (a) Rehabilitation robot workspace and 3D musculoskeletal model workspace evaluated at different chair heights and a single hand orientation with respect to the end-effector. (b) Mapping 3D musculoskeletal model to 2D model while interacting with the rehabilitation robot.

the kinematic function of MA2D-II, and $\mathbf{\Psi}_A$, which has a closed-form expression, is the inverse of vector function $\mathbf{\Phi}_A$. Finally, the dynamic equation of this 2 DOF arm model is as follows:

$$\begin{aligned} \mathbf{T}_A + \mathbf{J}_A^T \mathbf{F}_{int} &= \mathbf{M}_A(\boldsymbol{\theta})\ddot{\boldsymbol{\theta}} + \mathbf{C}_A(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}})\dot{\boldsymbol{\theta}} + \mathbf{B}_A\dot{\boldsymbol{\theta}} \\ &= \mathbf{\Gamma}_A(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}, \ddot{\boldsymbol{\theta}}) \end{aligned} \tag{4.24}$$

where $\mathbf{T}_A$ is the vector of the MA2D-II joint torques, which are functions of muscle lengths ($L_{\mathscr{M}}$), velocities ($V_{\mathscr{M}}$), moment arms ($r_{\mathscr{M}}^{S,E}$) and activations ($a_{\mathscr{M}}$). $\mathbf{M}_A$ and $\mathbf{C}_A$ are the 2D arm inertia (mass) and Coriolis-centrifugal matrices, respectively. $\mathbf{B}_A$ is a diagonal damping matrix.

4.2.3 2D human-robot dynamics

Integration of MA2D-II with the robot is a 2D closed-chain linkage with 8 inputs (2 robot motor torques, $T_{R1,2} \in [-10, 10]$ Nm, and 6 muscle group activations, $a_{1..6} \in [0, 1]$), and four outputs (2 robot joint angles, $q_{1,2}$, and 2 force sensor outputs, $F_{extZ,X}$). For this 2 DOF mechanism, a set of ordinary differential equations can be derived, if the number of generalized coordinates is reduced to 2 (the number of DOFs). In this subsection, we will provide a procedure to obtain a minimum number of equations for the system dynamics. In this closed-chain linkage, translational kinematics of the hand and end-effector should be the same (i.e., $\boldsymbol{\rho}_R = \boldsymbol{\rho}_A = \boldsymbol{\rho}$), and from (4.21) and (4.23) we get:

$$\begin{cases} \boldsymbol{\theta} = \boldsymbol{\Psi}_A(\boldsymbol{\Phi}_R(\mathbf{q})) \\ \dot{\boldsymbol{\theta}} = \mathbf{J}_A^{-1}\mathbf{J}_R\dot{\mathbf{q}} \\ \ddot{\boldsymbol{\theta}} = \mathbf{J}_A^{-1}\left(\dot{\mathbf{J}}_R\dot{\mathbf{q}} + \mathbf{J}_R\ddot{\mathbf{q}} - \dot{\mathbf{J}}_A\dot{\boldsymbol{\theta}}\right) \end{cases} \tag{4.25}$$

Thus, the MA2D-I kinematics can be written in terms of the robot kinematics. Combining (4.22) and (4.24), the 2D human-robot dynamics will be:

$$\mathbf{T}_A + \mathbf{J}_A^T\mathbf{J}_R^{-T}\left(\mathbf{T}_R - \boldsymbol{\Gamma}_R(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})\right) - \boldsymbol{\Gamma}_A(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}, \ddot{\boldsymbol{\theta}}) = \mathbf{0} \tag{4.26}$$

The left-hand-side of (4.26) is a function ($\boldsymbol{\Pi}$) of control inputs ($T_{R1,2}$ and $a_{1..6}$), $\mathbf{q}$, $\dot{\mathbf{q}}$ and $\ddot{\mathbf{q}}$, in which $\boldsymbol{\theta}$, $\dot{\boldsymbol{\theta}}$, and $\ddot{\boldsymbol{\theta}}$ are substituted from (4.25). The rearranged dynamic equation and its corresponding state-space equation is:

$$\ddot{\mathbf{q}} = \left(\frac{\partial \boldsymbol{\Pi}}{\partial \ddot{\mathbf{q}}}\right)^{-1}\left(\boldsymbol{\Pi} - \frac{\partial \boldsymbol{\Pi}}{\partial \ddot{\mathbf{q}}}\ddot{\mathbf{q}}\right) \tag{4.27}$$

$$\dot{\mathbf{x}}_{\mathbf{q}}(t) = \begin{Bmatrix} \dot{\mathbf{q}} \\ \ddot{\mathbf{q}} \end{Bmatrix} = \boldsymbol{\mathfrak{F}}\left(\mathbf{x}_{\mathbf{q}}(t), \mathbf{u}(t)\right) \tag{4.28}$$

where $\mathbf{x}_{\mathbf{q}} = [q_{1,2}, \dot{q}_{1,2}]^T \in \mathbb{R}^4$ and $\mathbf{u} = [T_{R1,2}, a_{1..6}]^T \in \mathbb{R}^6$ are the system state and control input vectors, respectively. The above equations are evaluated symbolically in Maple™.

4.3 Cross-validation studies of human models

To run simulations, we should consider a trajectory for manipulation. Since the upper extremity rehabilitation robot has been designed to perform point to point reaching tasks

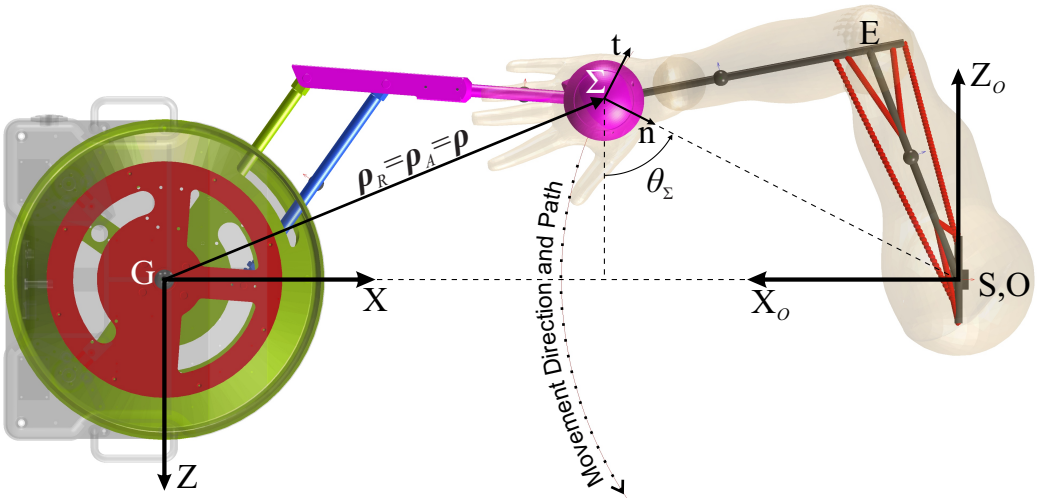


Figure 4.7: Path1 desired hand movement.

in the horizontal plane, these types of movements are studied. There are two hypotheses for the hand movement in the horizontal plane. The first hypothesis states that the movement is done visually in the world space without considering human body joint space, so the movement is a point to point, and it is done approximately in a straight line [1, 87, 189]. The second hypothesis proposes that the movement is done in the joint space considering musculoskeletal properties and it has a slightly curved path [18, 82, 193, 224, 275, 291, 335]. Both of these hypotheses claim that the hand movement is accomplished with a bell-shaped tangential speed profile and continuous jerk profile. In this study we will use both of the hypotheses to generate rehabilitation reaching paths as follows:

Path1: Considering a recent study by Zadravec and Matjačić [335], the trajectory is approximated by a smooth circular path with a large radius of curvature (see Fig. 4.7). To apply a bell-shaped tangential speed profile with a continuous jerk, a cubic spline interpolation approach is used to generate the path.

Path2: The desired trajectory for the simulations is selected based on the coordination of arm movements, in which the best performance in an unconstrained point-to-point reaching task is to generate the smoothest motion, and this objective is determined by minimizing the square of jerk magnitude [87]. This trajectory ($\boldsymbol{\rho}_d = [Z_d(t), X_d(t)]^T$) is a straight line with a bell-shaped tangential speed profile inside the human-robot

Table 4.10: Performances of five different approaches used for solving inverse dynamics problem.

Approach	Linear solver option	Simulation time (s)
Graph-theoretic approach	Fixed step (1 ms)	11
MSC® ADAMS	Fixed step (1 ms)	25
MapleSim™	Fixed step (1 ms)	16
OpenSim	Fixed step (1 ms)	21
Simscape Multibody™	Fixed step (1 ms)	190

workspace and defined as:

$$\begin{cases} Z_d(t) = Z_0 + (15\tau - 6\tau^2 - 10)\Delta_Z \tau^3 \\ X_d(t) = X_0 + (15\tau - 6\tau^2 - 10)\Delta_X \tau^3 \end{cases} \tag{4.29}$$

where Δ_Z and Δ_X are the path lengths along X and Z axes, respectively, and $\tau = t/t_f$.

With this introduction, the dynamic simulations are discussed in the following subsections.

4.3.1 MA3D-I dynamic simulations

Inverse dynamics of the skeletal model

We mapped the Path1 motion kinematics of MA2D-I_1 interacting with the passive robot into MA3D-I. The inverse dynamic results of four multibody dynamics software packages along with a graph-theoretic approach, which is hand-coded in MATLAB®, are presented in Fig. 4.8. The performances of these five different approaches are evaluated in Table 4.10. Considering simulation times and quality of results, MapleSim™performs better than other software platforms because of symbolic solution together with optimized code generation. That is why we utilize this software throughout this project.

SO of the musculoskeletal model with 10 muscles

Here, the SO problem is solved for the MA3D-I with 10 muscles, and the results are compared to the OpenSim model results. Considering the limited range of joint motions of

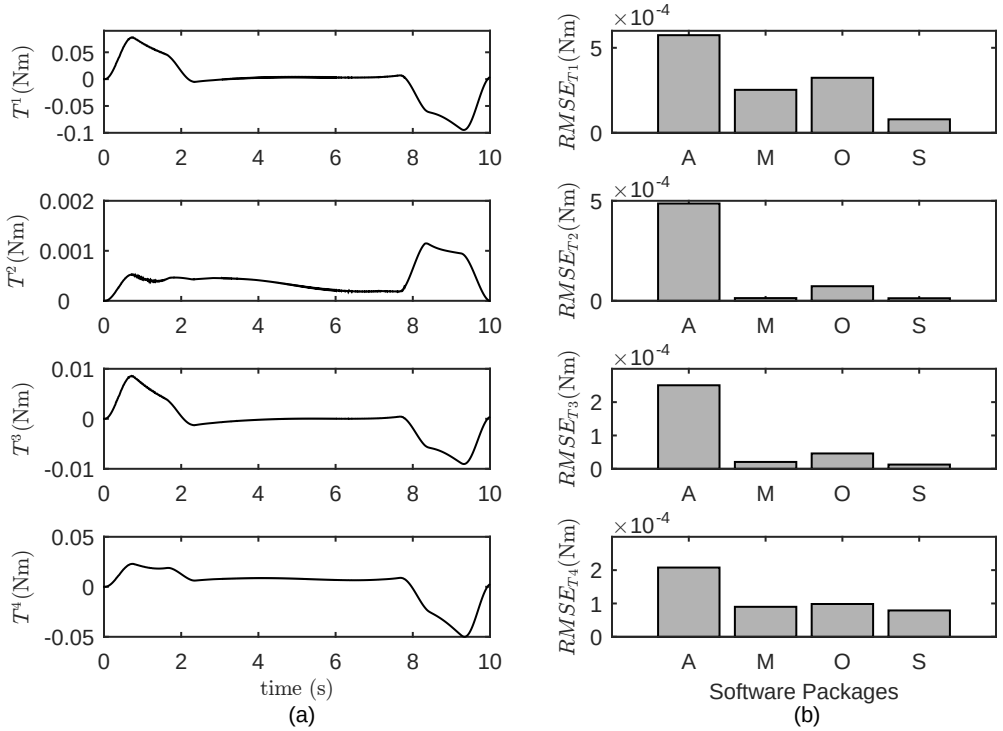


Figure 4.8: MA3D-I inverse dynamics results. a) Joint torques for $\dot{\jmath} = 1..4$, which is evaluated by a hand-coded graph-theoretic approach, and b) Evaluated torques by different software packages (A: MSC® ADAMS, M: MapleSim™, O: OpenSim, S: Simscape Multibody™) are compared to the graph-theoretic approach using RMSE measure.

the OpenSim 3D musculoskeletal model (i.e., shoulder rotation angle because of imperfect muscle wrapping is 20 degrees, while this value in most references have been reported to be about 90 degrees [229]), for the SO, the system is simulated only for about one fifth of the reaching path (i.e., 2 seconds). The activation patterns of the muscles for these two models are completely different; thus, for comparison, we introduce a new term called "activation content" which is the average activation of each muscle during simulation. Then we will discuss the reason for disparity. Activation contents of the muscles for these two models are presented in Fig. 4.9.

Patterns of activation contents for the MA3D-I and OpenSim models are similar (consider the patterns of the dashed and solid lines tracing activation contents). MA3D-I is the complete imitated model of the OpenSim musculoskeletal upper extremity model. Non-

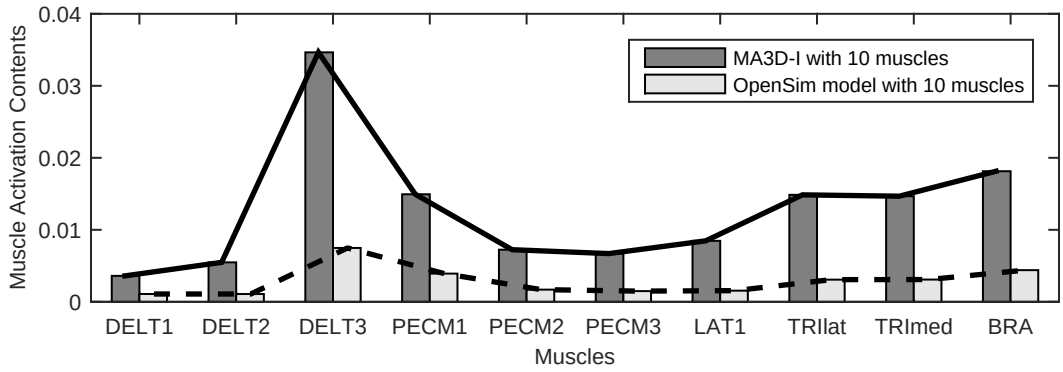


Figure 4.9: Activation contents of muscles for two 3D musculoskeletal upper extremity models with 10 active muscles (DELT1: Deltoid Anterior, DELT2: Deltoid Middle, DELT3: Deltoid Posterior, PECM1: Pectoralis Major Clavicular, PECM2: Pectoralis Major Sternal, PECM3: Pectoralis Major Ribs, LAT1: Latissimus Dorsi Thoracic, TRIlat: Triceps Lateral, TRImed: Triceps Medial, BRA: Brachialis).

etheless, its muscle activations patterns are different from the OpenSim model. To discuss this issue, consider the SO utilized in OpenSim. The OpenSim documentation [114] states that this software uses an objective function similar to (4.3) with a force-length-velocity constraint defined in (4.4). If we feed MA3D-I with the activations resulted from the OpenSim model static optimization, torque constraints will be violated (see Fig. 4.10). As can be seen, the OpenSim model results in noticeable torque error compared to the actual torque value. The reason for this error is the use of a different form of objective function by OpenSim, which can violate low torque constraints, and that is:

$$\text{OpenSim} : \mathfrak{J}_{\text{SO}} = w_1 \sum_{m=1}^{n_m} a_m^2 + w_2 \sum_{j=1}^{n_j} \left(T^j - \sum_{m=1}^{n_m} r_m^j F_m \right)^2 \tag{4.30}$$

where n_j is the number of active joints of the musculoskeletal model. This formulation of the objective function is completely dependent on the defined constant weights (w_i), and improper selection of these weights may result in constraint violation. To summarize, the activation results of the OpenSim model are not correct for the defined inverse dynamics problem.

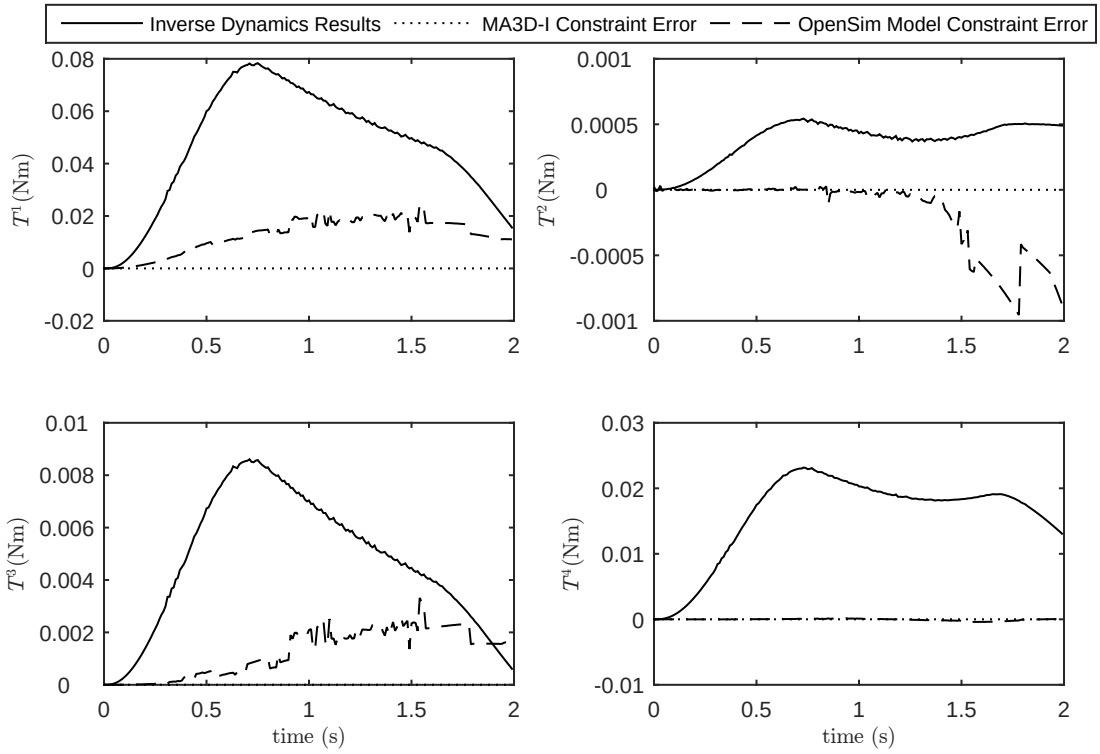


Figure 4.10: Inverse dynamics results along with the torque constraint errors for the MA3D-I and OpenSim models.

4.3.2 MA2D-I_2 dynamic simulations

In this simulation, MA2D-I_2 (the 2D upper extremity model with shoulder rhythm) is used to interact with the passive robot on Path1. The implemented shoulder rhythm is derived from the OpenSim model, which is only dependent on the "`shoulder_elv`" joint ($\mathcal{J} = 2$) motion. The results of this simulation are nearly identical to the results of MA2D-I_1 simulation. The reason for this similarity is the negligible variation of "`shoulder_elv`" joint throughout the entire movement; hence, shoulder joint translation in MA2D-I_2 does not exceed 0.5 mm.

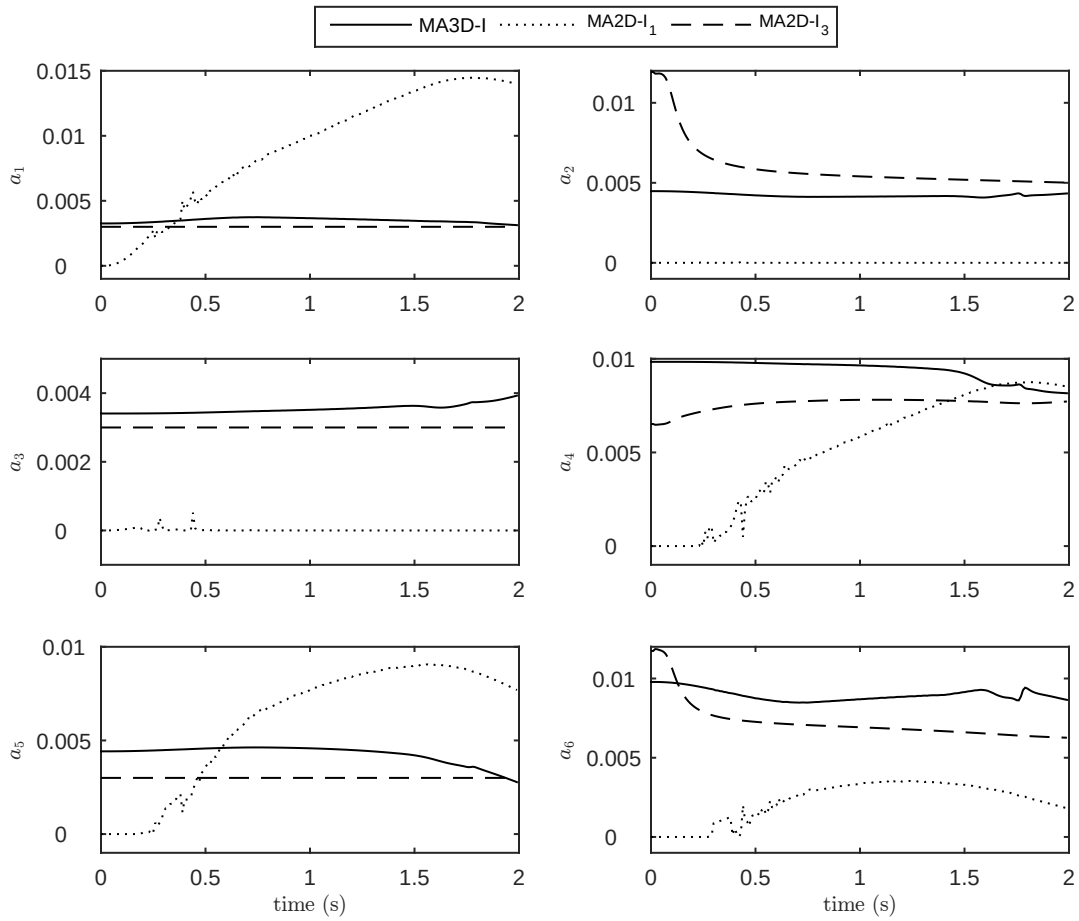


Figure 4.11: Muscle activations of three upper extremity musculoskeletal models interacting with the unactuated robot.

4.3.3 MA2D-I$_3$ dynamic simulations

In this simulation, MA2D-I$_2$ (the 2D upper extremity model with shoulder rhythm) is used to interact with the passive robot on Path1 for 2 seconds, and the SO problem is solved. The results are summarized in Fig. 4.11, in which we see that the muscle activations of MA2D-I$_3$ are a better match to MA3D-I than MA2D-I$_1$. The reasons for the differences between muscle activations of the MA3D-I and MA2D-I$_3$ models can be summarized as follows:

1. An approximate method is used for lumping muscles from MA3D-I into MA2D-I_3.

2. A straight-line approach for muscle paths in MA2D-I_3 will decrease active muscle lengths. This has a significant influence on the behavior of the shoulder muscles, which have complicated wrapping geometries (note the initial activation of Muscle 2).

3. MA3D-I has three shoulder torque constraints, which are (4.4) for $\dot{\mathscr{l}} = 1..3$, while MA2D-I_3 has only one shoulder torque constraint.

Here, we show that the 2D musculoskeletal arm model can effectively represent the 3D motion of the arm; thus, it can be used to evaluate the rehab robot performance. Although development of this 2D musculoskeletal arm model requires a 3D model with muscle wrapping geometries, the lower-fidelity 2D model – in contrast to the higher-fidelity 3D model – can be used in real-time simulations and model-based controllers.

4.4 DPI study of human models

If we recall the sensitivity analysis of the musculoskeletal models (see Subsection 2.2.5), changing musculoskeletal parameters may result in large muscle force estimation errors, especially for impaired patients. To reduce these estimation errors, we need to have a subject-specific musculoskeletal model with parameters identified from experiments.

4.4.1 Method

Since the 2D model (MA2D-II) will be used within our NMPC scheme, DPI with the setup shown in Fig. 4.12 will be performed on this model. The DPI tests are summarized as:

1. Optotrak Certus markers are installed on the three bony landmarks of the subject including AC, EL, and 3$^{\text{rd}}$ metacarpal bone head. Subject's wrist joint is fixed. It is assumed that the subject's frontal plane is always parallel to the YZ plane (i.e., he/she is not leaning). Furthermore, since the subject is not raising his/her hand, the amount of shoulder rhythm in the horizontal plane movement is negligible. Thus, it is possible to measure shoulder and elbow joint angles ($\theta_{Se,Ee}$, see Fig. 4.13).

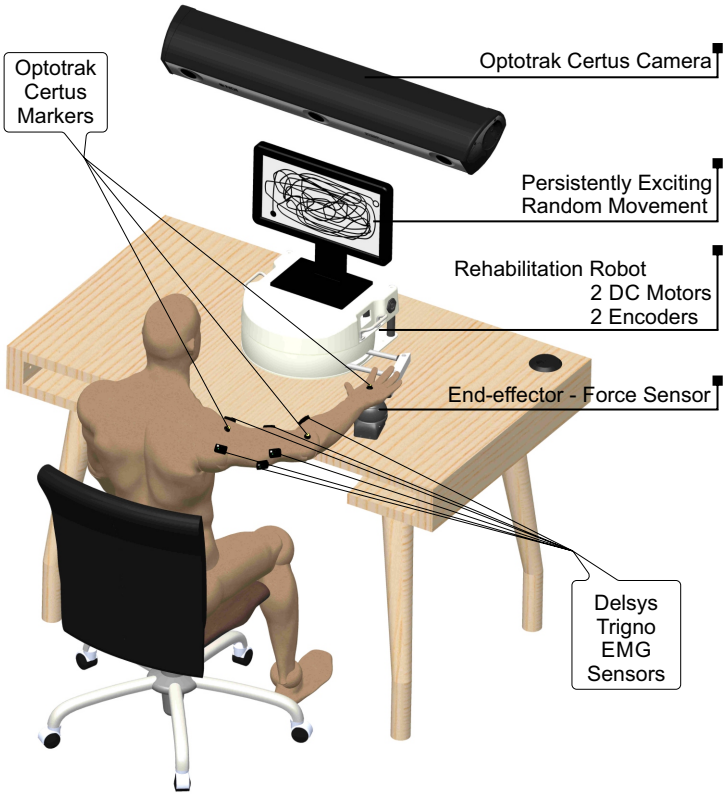


Figure 4.12: DPI setup for the identification of the musculoskeletal model dynamic parameters.

2. Delsys Trigno wireless EMG sensors are attached to 6 muscles including Anterior Deltoid (in Muscle1), Posterior Deltoid (in Muscle2), Brachioradialis (in Muscle3), Triceps Brachii Lateral (in Muscle4), Biceps (in Muscle5), and Triceps Brachii Long (in Muscle6). Each of these muscles is selected as the representative of their corresponding muscle group (see Fig.4.13). Before starting tests, the Maximum Voluntary Contraction (MVC) of each muscle is recorded for the subject.

3. The DPI tests are performed by the subject, while he/she is holding the robot end-effector and moving it in the horizontal plane with randomly periodic movements which can be considered as sufficiently rich movements.

4. During the tests, the robot damps the subject's movement by controlling the end-effector with a differential controller ("D Block" in Fig. 4.14). The robot also meas-

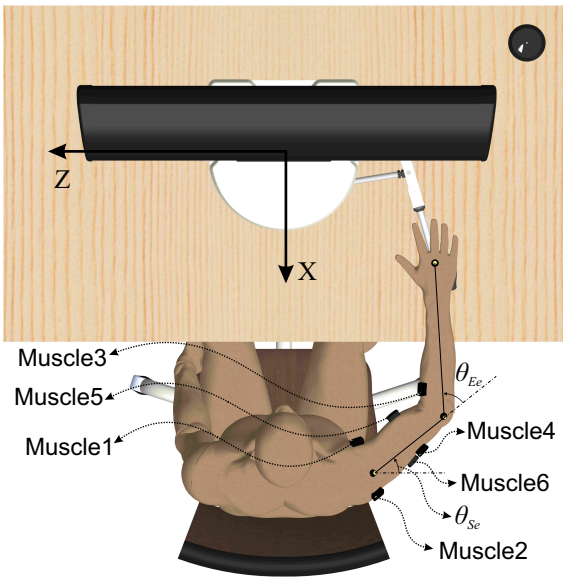


Figure 4.13: Muscles and joint angles configuration in the DPI setup (top view).

ures the interaction force by the end-effector force sensor.

5. The outputs of these DPI tests are the robot joint angles (2 joints), the interaction force (2 forces along X and Z axes), the marker positions (3 positions), and the EMGs of the 6 muscles.

Assume that the dynamic parameters of an impaired subject with unknown parameters in activation dynamics (see Appendix B for the model of activation dynamics) is needed. For the DPI of the MA2D-II model with unknown parameters in activation dynamics, we use the proposed framework that is summarized in Fig. 4.14. These steps are followed in this framework:

1. The DPI experiments are performed.

2. The marker positions and robot joint angles are fed to the "Kinematic Optimization Block". This block, first, evaluates the experimental shoulder and elbow joint angles ($\theta_{Se,Ee}$, see Fig.4.13). Then, the block tries to solve the following kinematic optimization problem:

$$\arg\min_{\boldsymbol{x}} \int_0^{t_f} \left((\theta_{Se} - \theta_{So})^2 + (\theta_{Ee} - \theta_{Eo})^2\right) \tag{4.31}$$

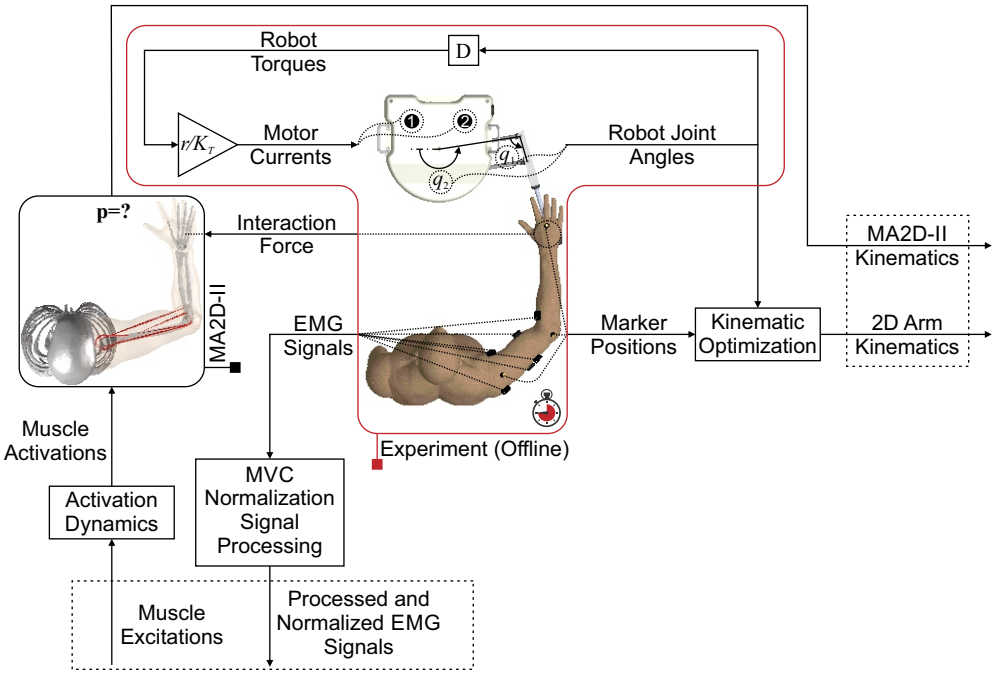


Figure 4.14: MA2D-II DPI workflow with unknown parameters in activation dynamics. Kinematic optimization includes the kinematic constraint (4.32). *Muscle Excitations* coressponds to $\mathbf{e}$, and *Processed and Normalized EMG Signals* denotes $\mathbf{e}_{\text{EMG}}$.

subject to the HRI kinematics (MA2D-II kinematics (4.23) and robot kinematics (4.21)):

$$\boldsymbol{\rho}_A(\boldsymbol{\chi}, \theta_{So}, \theta_{Eo}) - \boldsymbol{\rho}_R = 0 \tag{4.32}$$

where $\boldsymbol{\chi} = [Z_S, X_S, l_h, l_u]^T$ is the vector of unknown kinematic parameters: ($[Z_S, X_S]^T$ is the shoulder joint position in the global coordinates, l_h is the upper arm length, and l_u is the forearm length), and subscript o stands for the optimized kinematic model. $\boldsymbol{\rho}_R$ is the end-effector position which is evaluated by the robot's experimentally measured joint angles. (4.32) constraint is for ensuring that the end-effector and hand are at the same position. Since the shoulder may have unwanted translational movements, its optimal position is estimated in this optimization problem.

3. Assume that the dynamic parameters of the MA2D-II were known. Then, in a noise-free system, if we feed the muscle excitations (which are the control input $\mathbf{e}$ to the MA2D-II model followed by the activation dynamics) with the processed and normalized EMG signals ($\mathbf{e}_{\text{EMG}}$, see the dashed box in Fig. 4.14 on the left), the

MA2D-II kinematics will be close to the optimized kinematics from the previous step (see the dashed box in Fig. 4.14 on the right). However, noisy input ($\mathbf{e} = \mathbf{e}_{\text{EMG}} - \mathbf{d}$) impedes the convergence of most algorithms for solving this DPI problem. To address this issue, assuming a normal distribution for $\mathbf{d^T d}$ with zero mean and variance ς^2, we propose an alternative optimal control problem (similar to (3.48)) with the following objective functional:

$$\mathfrak{J} = \frac{1}{2}\int_0^{t_f} \left(\mathbf{e} - \mathbf{e}_{\text{EMG}}\right)^T \left(\mathbf{e} - \mathbf{e}_{\text{EMG}}\right) dt \tag{4.33}$$

subject to (4.24) with the path constraint:

$$\mathfrak{c}(t) = \frac{1}{2}\left((\theta_S - \theta_{So})^2 + (\theta_E - \theta_{Eo})^2\right) \leq \epsilon^2 \tag{4.34}$$

This optimal control problem will find the unknown dynamic parameters of the human model, while trying to find control inputs such that they minimize (4.33) and hold the dynamic constraint (4.24) with activation dynamics and (4.34).

For the DPI of a healthy subject, since the parameters of the activation dynamics are known (see Appendix B for the model of activation dynamics and assigned parameters for young adults), the DPI of the MA2D-II model with known activation dynamics will have the framework in Fig. 4.15. Noisy input ($\mathbf{a} = \mathbf{a}_{\text{EMG}} - \mathbf{d}$) impedes the convergence of most algorithms for solving this DPI problem. Consequently, assuming a normal distribution for $\mathbf{d^T d}$ with zero mean and variance ς^2, the objective functional for the DPI of a healthy subject will be updated as:

$$\mathfrak{J} = \frac{1}{2}\int_0^{t_f} \left(\mathbf{a} - \mathbf{a}_{\text{EMG}}\right)^T \left(\mathbf{a} - \mathbf{a}_{\text{EMG}}\right) dt \tag{4.35}$$

where $\mathbf{a}_{\text{EMG}}$ are the activations from the processed and normalized EMG signals by passing through the activation dynamics.

4.4.2 Experiments

For the experiments, the above mentioned tests are performed on a healthy female subject (having weight and height of 55 kg and 164 cm, respectively). The subject was asked to do 5 different movements in 5 trials. These randomly periodic movements are as follows:

Trial1: Clockwise rotation of the arm in the horizontal plane.

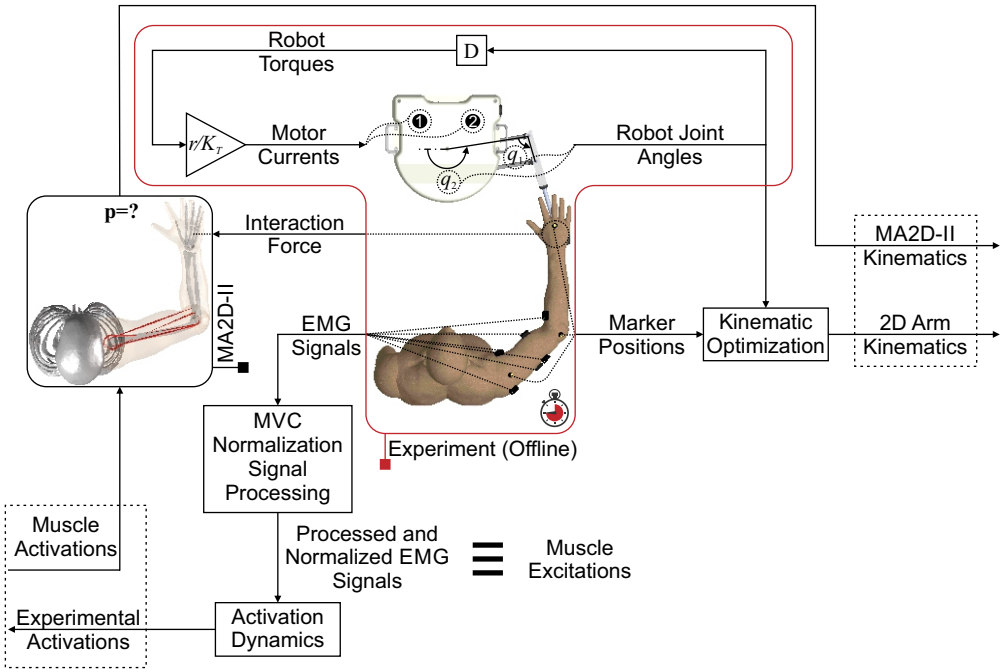


Figure 4.15: MA2D-II DPI workflow with known parameters in activation dynamics. Kinematic optimization includes the kinematic constraint (4.32). *Muscle Activations* coressponds to $\mathbf{a}$, and *Experimental Activations* denotes $\mathbf{a}_{\text{EMG}}$.

Trial2: Counter-clockwise rotation of the arm in the horizontal plane.

Trial3: Cyclic lateral reaching in the horizontal plane.

Trial4: Cyclic anterior reaching in the horizontal plane.

Trial5: Random movement in the horizontal plane.

The duration of each trial is 60 s, and the sampling-time frequency is set to 500 Hz.

4.4.3 Results and discussion

Similar to Section 3.3, the GPOPS-II software is used to identify the dynamic parameters. Each trial is treated as a single phase of a GPOPS-II problem; thus, a 5-phase optimization is solved. Results for the RMSE of the activations, joint angles and velocities are presented

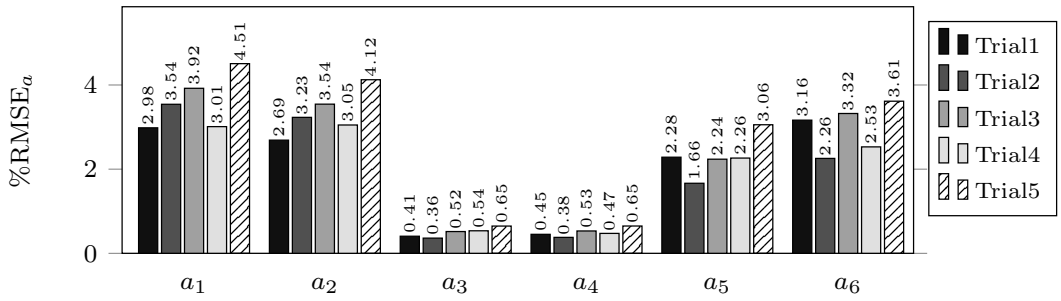


Figure 4.16: Percentage of the RMSE of the muscle activations.

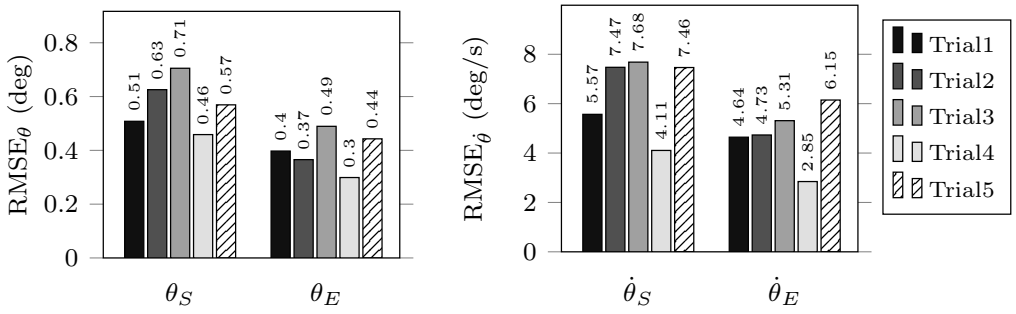


Figure 4.17: RMSE of the joint angles and velocities.

in Figures 4.16 and 4.17. Maximum RMSE for the muscle activations is less than 5%. Considering, gravitational effects and noise in the EMG system, this amount of error is really promising. Furthermore, RMSE for the joint angles and velocities are less than 0.8 deg and 8 deg/s; this is less than the errors reported using inertial position tracking sensors that are used in the human movement studies [79, 80]. Hence, the system has been identified with a good accuracy.

The SNR value for each muscle actiavtion is evaluated and averaged over 5 trials. The minimum value of this average is 11.4 which corresponds to Muscle 1. This value indicates that the experimental activations can be estimated by model activations with $\pm 14.8\%$ error. Furthermore, the SNR value for each joint angle and velocity is evaluated and averaged over 5 trials. The minimum of the average SNR for the joint angles and velocities are 36.4 and 18.2. Thus, the model with estimated muscle activations (which are in $[85.2, 114.8]\%$ of the experimental values) can generate a motion within $[91.8, 108.2]\%$ and $[88.3, 111.7]\%$ of the experimental joint angles and velocities. Identified parameters are reported in Tables A.11 and A.12.

4.4.4 Conclusion

Using the identified parameters for the model, it is possible to control the model with ±14.8% error in the experimental muscle activations as the inputs and achieve ±8.3% and ±11.7% error in tracking the experimental joint angles and velocities, respectively. Thus, our proposed method for the DPI of the musculoskeletal models can estimate the dynamic parameters of the model with an acceptable accuracy. Nonetheless, muscle redundancy causes suboptimal results as the number of trials is reduced. Thus, here we used five different trials to alleviate the muscle redundancy issue.

4.5 Concluding remarks

The 3D musculoskeletal model – the high-fidelity model – is not suitable for real-time simulations, which are required for the design of model-based controllers. Thus, in this chapter, we developed different 2D models because they can be used in real-time simulations and controllers. Results of cross-validation studies show that it is possible to use a 2D musculoskeletal arm model for evaluation and control of the planar robot. However, development of this 2D musculoskeletal arm model requires a 3D musculoskeletal arm model with proper muscle wrappings. Furthermore, sensitivity studies of the upper extremity models show that the estimated muscle forces are highly dependent on subject-specific parameters. Thus, first, the new 3D model (MA3D-II) was developed, and it was mapped into the 2D model (MA2D-II). Second, a method for the DPI of the MA2D-II is provided to include subject-specific parameters in the model. Experiment results showed that it is possible to estimate the dynamic parameters of the model with ±8.3% and ±11.7% error in tracking the experimental joint angles and velocities when the model is controlled with ±14.8% error in the experimental muscle activations as the inputs.

Chapter 5

Rehabilitation Robot Control: Robot Control

Intense, exhilarating and invigorating rehabilitation therapy has shown promising results in stimulating the neural plasticity to treat post-stroke movement impairments [235]. Hence, upper extremity rehabilitation robots are developed to target this kind of therapy [290].

To stimulate neural plasticity, three modes of high-level control scenarios are used in rehabilitation robots: assistive, corrective, and resistive (see Fig. 5.1) [170, 219]. Consequently, low-level control scenarios are needed to implement those high-level control modes. Assistive control mode is the most common in these robots (refer to Fig. 5.1).

In robotic rehabilitation, although the patient is physically interacting with a mechanical device, these robots mostly use black-box (robot control) approaches (Fig. 5.1), which do not consider human-robot interaction. Thus, safety issues may become a concern. For this reason, gray-box (triggered passive), and HRI controllers have been developed to reduce the safety risk (see Fig. 5.1). Though gray-box controllers incorporate feedback from the human body using biosignals [173], they require intensive preparation and are therefore not practical for daily usage with multiple post-stroke patients.

Because of easy and simple implementation for real-time tasks, in this chapter, we will focus on the advancement of the safest black-box controller impedance-based assistance. In some ways, this controller can also be considered as an HRI control, since it considers HRI by including the interaction force in the controller structure. Thus, impedance-based assistance is well-suited for robotic rehabilitation [173], since conventional position/force control scenarios do not consider dynamic interaction of the human-robot system [117]. Furthermore, assist-as-needed therapy, which encourages voluntary participation of the patient,

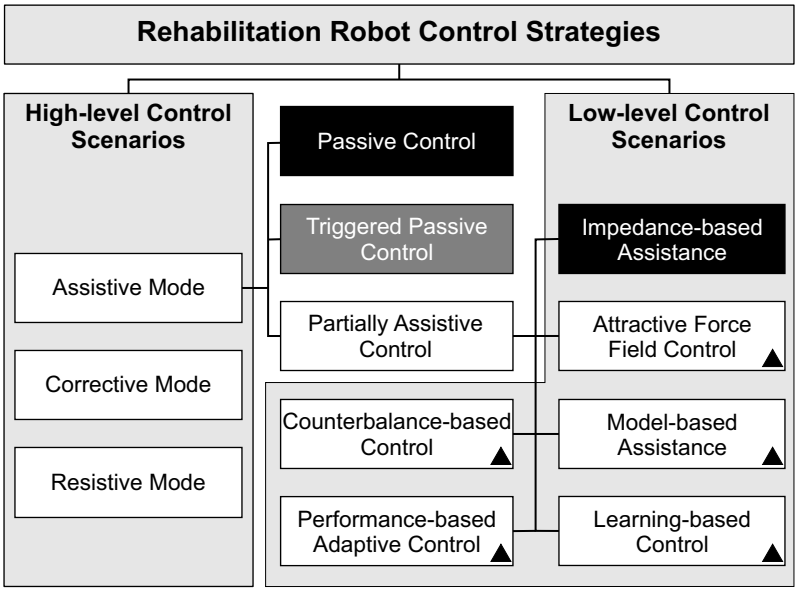


Figure 5.1: Rehabilitation robot control strategies. Gray- and black-box control methods are colored in dark gray and black, respectively. HRI controllers are specified with a black triangle on the corner.

is implementable through the admittance/impedance control scenarios. In impedance-based assistance, the amount of assistance/resistance (i.e., compliance) can be adjusted by controlling the impedance gains. However, in the presence of a variable admittance environment (i.e., different patients) or different trajectories (i.e., robot configurations), the interaction force and configuration will exacerbate inefficiency of the controller with non-optimal gains. For example, a resistive-capacitive impedance control with therapist-adjustable constant stiffness and damping ratios is implemented in our studied upper extremity rehabilitation manipulandum, but these gains cannot be adjusted *optimally* using trial and error by the therapist [126].

The problem here is that the robot interacts with a variable admittance environment; thus, not only the configuration but also the interaction force may affect the controller response with constant gains. The objective of a controller design is to reduce therapist intervention and improve the quality of therapy regarding safety and rehabilitation. Therefore, we need to design a type of impedance-based control that can automatically adjust itself for variable admittance environments or variable robot configurations. Impedance-based controller can be impedance or admittance type. Impedance control is more stable

and safer than an admittance control. Thus, in this chapter, two types of impedance-based controllers are developed. Then, their improvements are discussed.

5.1 Hybrid force-impedance control

Besides other methods of partially assistive control (e.g., attractive force-field control, model-based assistance, learning-based assistance, counter-balance-based assistance, and performance-based adaptive control), adaptive and optimal forms of impedance control have been developed to deal with variable admittance environments. Hussain et al. [127] used an adaptive impedance control for patient-cooperative therapy of a lower-limb exoskeleton, and they verified the controller performance using an experimental setup. In more recent studies, optimal impedance controls for an exoskeleton gait trainer and elbow rehabilitation robot were developed [72, 308]. The proposed methods were implemented in a computer simulation, and the real-time performance of the controllers were not discussed. In an exoskeleton, the impedance control is defined in the joint space, while in a manipulandum, the impedance model is in the operational space. Thus, these controllers which are developed for exoskeletons, are not suitable for a manipulandum. Furthermore, these controllers are developed for some sort of predefined rhythmic motions and they are not implementable for random reaching movements. In a recent related study, Maldonado et al. [171] used a stiffness-based tuning for an adaptive impedance control of an upper extremity manipulandum; nevertheless, the method was verified using computer simulations and its real-time capabilities were not mentioned.

To improve impedance control performance, the compliance can be controlled by an outer-loop force control. In other words, it is desired to implement an Hybrid Force-Impedance Control (HFIC). In literature, there are different types of hybrid impedance controllers [11, 83, 160, 192]. These studies used some type of selection matrix to alternate between the impedance and force controllers. Furthermore, they considered an incomplete impedance model with the Cartesian space implementation. Here, the selection matrix is embedded in the gain matrices and the physics of the problem causes alternation between two controller. For the impedance model, we use a complete type of impedance model (with position, velocity and acceleration errors) with an operational space implementation.

Before presenting the controller, first, we review the impedance control. Then, our proposed control scheme is discussed. The controller is tuned by simulating the human-robot system (MA2DI_1).

5.1.1 Impedance control review

Impedance control is defined in the global or operational coordinates, so for the best practice, the system dynamic equation should be defined in these coordinates. However, the dynamic equation of the robot (4.22) is defined in the joint space, and it has nonlinear terms. To implement impedance control in the operational space, we should follow these steps:

Step1: Define an inner loop control law ($\mathbf{u}$) using nonlinear feedback linearization (inverse dynamics approach) to cancel the nonlinear terms:

$$\mathbf{u} = \mathbf{J}_R^T \mathbf{F}_{ext} + \mathbf{\Gamma}_R(\mathbf{q}, \dot{\mathbf{q}}, \boldsymbol{\mathcal{y}}) \tag{5.1}$$

where $\boldsymbol{\mathcal{y}}$ is the outer loop control law.

Step2: Define the outer loop control law such that it changes manipulator behavior to a linear impedance under interaction force error. In other words, it is desired to have the linear impedance model in the global coordinates (i.e., the Cartesian space) as:

$$-\widetilde{\mathbf{F}}_{ext} = \mathbf{M}_{imp}\ddot{\widetilde{\boldsymbol{\rho}}} + \mathbf{B}_{imp}\dot{\widetilde{\boldsymbol{\rho}}} + \mathbf{K}_{imp}\widetilde{\boldsymbol{\rho}} \tag{5.2}$$

where subscript *imp* stands for the impedance model. The above impedance model can be achieved if the outer loop control law is defined as:

$$\boldsymbol{\mathcal{y}} = \mathbf{J}_R^{-1}\mathbf{M}_{imp}^{-1}\left(\mathbf{M}_{imp}(\ddot{\boldsymbol{\rho}}_d - \dot{\mathbf{J}}_R\dot{\mathbf{q}}) + \mathbf{B}_{imp}\dot{\widetilde{\boldsymbol{\rho}}} + \mathbf{K}_{imp}\widetilde{\boldsymbol{\rho}} + \widetilde{\mathbf{F}}_{ext}\right) \tag{5.3}$$

Step3: Since it is desired to have some compliance along the operational space directions (normal and tangent to the path, end-effector's n-t coordinates in Fig. 5.3.a), the impedance model is updated as:

$$-\widetilde{\mathbf{F}}_{ext}^{\Sigma} = \mathbf{M}_{imp}\ddot{\widetilde{\boldsymbol{\rho}}}^{\Sigma} + \mathbf{B}_{imp}\dot{\widetilde{\boldsymbol{\rho}}}^{\Sigma} + \mathbf{K}_{imp}\widetilde{\boldsymbol{\rho}}^{\Sigma} \tag{5.4}$$

here, superscript Σ denotes that the corresponding vector is defined in the end-effector's n-t coordinates (i.e., the operational space). If $\mathbf{R}_\Sigma$ is the rotation matrix transforming the n-t coordinates to the global coordinates, $\widetilde{\boldsymbol{\rho}}^{\Sigma}$ can be obtained from the following equations:

$$\begin{cases} \widetilde{\boldsymbol{\rho}}^{\Sigma} = \mathbf{R}_\Sigma^T \widetilde{\boldsymbol{\rho}} \\ \mathbf{R}_\Sigma = \begin{bmatrix} \cos(\theta_\Sigma) & -\sin(\theta_\Sigma) \\ \sin(\theta_\Sigma) & \cos(\theta_\Sigma) \end{bmatrix} \end{cases} \tag{5.5}$$

here, θ_Σ is defined in Fig. 5.3.a. Based on the updated impedance model, the outer loop control law is redefined as:

$$\boldsymbol{\mathscr{y}} = \mathbf{J}_R^{-1}\mathbf{R}_\Sigma\mathbf{M}_{imp}^{-1}\left(\mathbf{M}_{imp}(\dot{\mathbf{b}} - \dot{\mathbf{J}}_R^\Sigma\dot{\mathbf{q}}) + \mathbf{B}_{imp}\widetilde{\dot{\boldsymbol{\rho}}}^\Sigma + \mathbf{K}_{imp}\widetilde{\boldsymbol{\rho}}^\Sigma + \widetilde{\mathbf{F}}_{ext}^\Sigma\right) \tag{5.6}$$

where:

$$\begin{cases} \dot{\mathbf{b}} = \mathbf{R}_\Sigma^T\ddot{\boldsymbol{\rho}}_d - \dot{\boldsymbol{\varpi}}_\Sigma\widetilde{\boldsymbol{\rho}}^\Sigma + \boldsymbol{\varpi}_\Sigma\boldsymbol{\varpi}_\Sigma\widetilde{\boldsymbol{\rho}}^\Sigma + \boldsymbol{\varpi}_\Sigma\mathbf{R}_\Sigma^T(\mathbf{J}_R\dot{\mathbf{q}} - 2\dot{\boldsymbol{\rho}}_d) \\ \dot{\mathbf{J}}_R^\Sigma = \mathbf{R}_\Sigma^T\dot{\mathbf{J}}_R - \boldsymbol{\varpi}_\Sigma\mathbf{R}_\Sigma^T\mathbf{J}_R \\ \dot{\mathbf{R}}_\Sigma = \boldsymbol{\varpi}_\Sigma\mathbf{R}_\Sigma \\ \boldsymbol{\varpi}_\Sigma = \begin{bmatrix} 0 & -\dot{\theta}_\Sigma \\ \dot{\theta}_\Sigma & 0 \end{bmatrix} \end{cases} \tag{5.7}$$

5.1.2 Proposed controller

In (5.6), the term $\mathbf{J}_R^{-1}\mathbf{R}_\Sigma\mathbf{M}_{imp}^{-1}\widetilde{\mathbf{F}}_{ext}^\Sigma$ makes the manipulator compliant in its interaction with the environment [258]; thus, it is a compliance term. To have variable impedance control response, one way is to control the compliance term. In other words, we can rewrite (5.6) as:

$$\boldsymbol{\mathscr{y}} = \mathbf{J}_R^{-1}\mathbf{R}_\Sigma\mathbf{M}_{imp}^{-1}\left(\mathbf{M}_{imp}(\dot{\mathbf{b}} - \dot{\mathbf{J}}_R^\Sigma\dot{\mathbf{q}}) + \mathbf{B}_{imp}\widetilde{\dot{\boldsymbol{\rho}}}^\Sigma + \mathbf{K}_{imp}\widetilde{\boldsymbol{\rho}}^\Sigma + \mathbf{F}_{hyb}^\Sigma\right) \tag{5.8}$$

where subscript *hyb* stands for the HFIC. $\mathbf{F}_{hyb}^\Sigma$ is determined by another outer loop control law, which is used to control the compliance in the impedance control. Similar to the work done by [83, 84], this outer loop control can be defined as a Proportional-Integral (PI) force control:

$$\mathbf{F}_{hyb}^\Sigma = \mathbb{K}_P\widetilde{\mathbf{F}}_{ext}^\Sigma + \mathbb{K}_I\int\widetilde{\mathbf{F}}_{ext}^\Sigma dt \tag{5.9}$$

here, $\mathbb{K}_P$ and $\mathbb{K}_I$ are the proportional and integral gains. In $\widetilde{\mathbf{F}}_{ext}^\Sigma = \mathbf{F}_d^\Sigma - \mathbf{F}_{ext}^\Sigma$, desired interaction force ($\mathbf{F}_d^\Sigma$) can be obtained from a predefined external input, or an inner or outer loop planner. For example, Erol and Sarkar [84] used an inner velocity loop to get $\mathbf{F}_d^\Sigma$ [83]. If one combines the inner and outer loops in (5.1), (5.8), and (5.9), the control scheme in Fig. 5.2 will result, and following impedance model will be obtained:

$$-\mathbb{K}_P\widetilde{\mathbf{F}}_{ext}^\Sigma - \mathbb{K}_I\int\widetilde{\mathbf{F}}_{ext}^\Sigma dt = \mathbf{M}_{imp}\widetilde{\ddot{\boldsymbol{\rho}}}^\Sigma + \mathbf{B}_{imp}\widetilde{\dot{\boldsymbol{\rho}}}^\Sigma + \mathbf{K}_{imp}\widetilde{\boldsymbol{\rho}}^\Sigma \tag{5.10}$$

Based on the above equation, if the position tracking error is zero (i.e., the right-hand side of the above equation is zero), then the controller will change to a PI force control. When

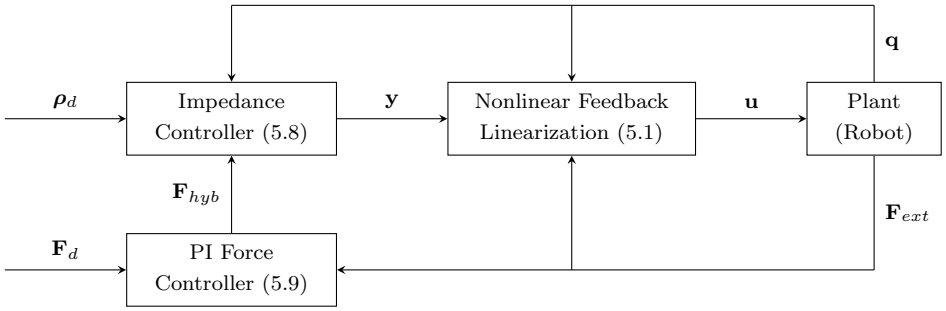


Figure 5.2: Hybrid force-impedance controller scheme.

the force tracking error is zero (i.e., the left-hand side of the above equation is zero), the controller is an acceleration control. If none of the tracking errors are zero, then the gains should be selected with these specifications:

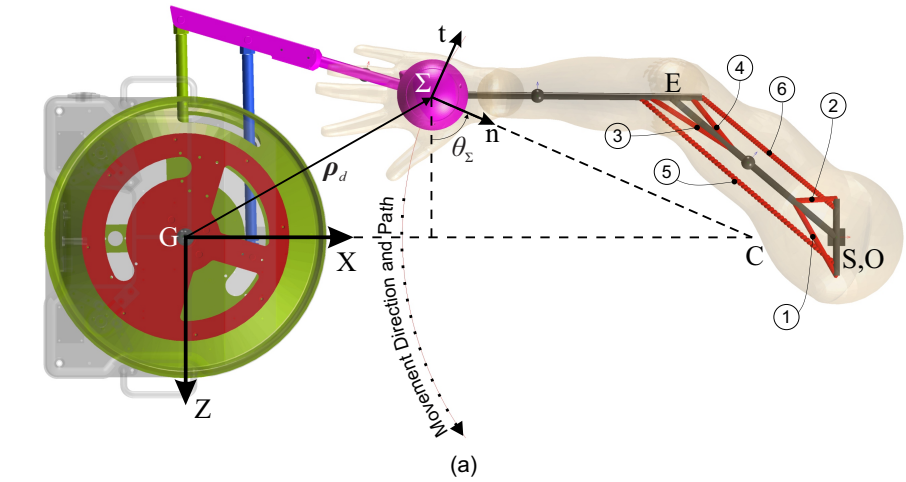
- If the position tracking is required in a specific direction (for example, normal to the desired trajectory), then the corresponding gains (in the right-hand side of (5.10)) of this direction should be dominant than the left-hand side gains of (5.10).
- If the force tracking is required in a specific direction (for example, tangent to the desired trajectory), then the corresponding gains (in the left-hand side of (5.10)) of this direction should be dominant than the right-hand side gains of (5.10).

This controller is advantageous, since it accounts for the change in the interaction force error while regulating the position error with the impedance model. Furthermore, by implementing this controller, the position and force control can be achieved in orthogonal directions.

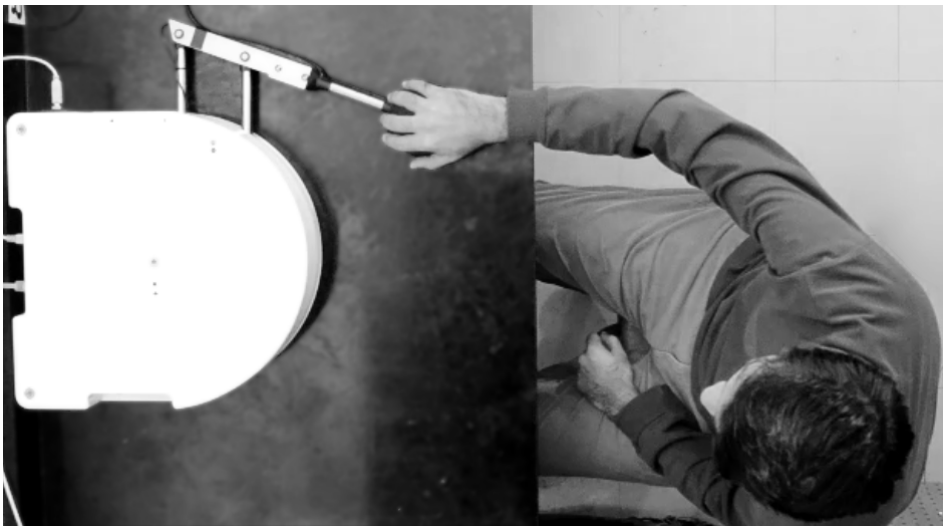
5.1.3 Simulation

In robotic rehabilitation, it is usually desired to follow a predefined path. During a path-following task, at least three therapy cases can occur [6, 67]:

1. *Passive* case: the patient cannot accomplish the task, so the robot actively manipulates the patient's hand.
2. *Active-assisted* case: the patient is unable to finish the task independently in a specified time interval. Thus, the robot assists the patient as needed.



(a)



(b)

Figure 5.3: Human-robot rehabilitation system: (a) MapleSim^{TM}model (circled numbers show the corresponding muscle number), (b) experimental setup.

3. *Active-constrained* case: the patient can accomplish the task independently even faster than the predefined time interval. Hence, the robot tries to resist against patient's rapid movements.

Here, to evaluate the performance of the controller during a rehabilitation procedure, four modes of movement are considered: impaired, healthy, delayed and time-advanced

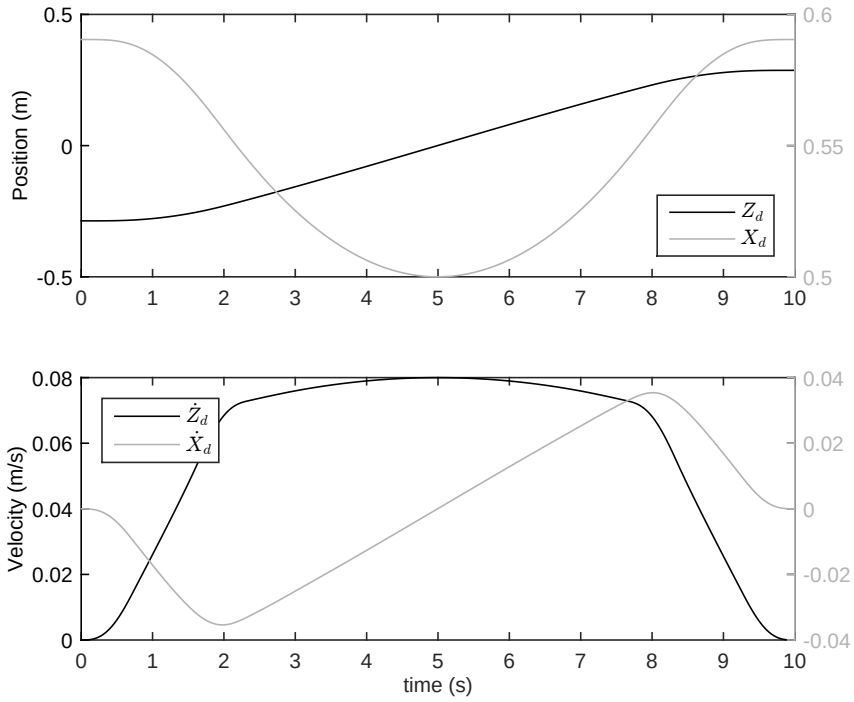


Figure 5.4: Desired trajectory for point-to-point reaching movement (Path1).

hand movement along the specified path (Path1, see Fig. 5.3.a and Fig. 5.4). In the impaired hand movement mode, the upper extremity of the patient is totally dysfunctional (zero muscle activation), so the passive case will occur. In the healthy mode, the patient has normal timing and coordination, so one of the active cases can happen depending on the healthy subject performance. The delayed hand movement mode is used to model a stroke patient who needs assistance during therapy, and it leads to the active-assisted case. The time-advanced mode models a patient with rapid hand movements; thus, the active-constrained case will be enabled. In the next section, the performance of the proposed hybrid controller is compared to other controllers.

Generated MapleSim$^{\text{TM}}$models of the HRI (as in Fig. 5.3.a, which is used for model-in-loop simulation) and human (MA2DI_1, which is used for solving the muscle redundancy problem and controlling the muscle activations in HRI) are exported as optimized MATLAB® S-functions into the Simulink/MATLAB® environment. Sampling-time frequency (ν) of the simulations is set to 500 Hz, and a fixed-step Euler solver is selected to

solve the ODEs. For $MA2DI_1$, the passive elements of the arm muscles were assumed to have less contribution than the active elements in muscle forces. Hence, the CE of the Hill-type muscle model is used to model muscle dynamics, and FSO is implemented to solve the muscle force sharing problem. See Table A.13 for the assigned control parameters.

5.1.4 Results and discussion

Results of these simulations include muscle activations and input motor torques (see Fig. 5.5), and interaction force and position errors (see Fig. 5.6). Active muscle activations in three modes of simulation (healthy, delayed, and time-advanced hand movements) are reported in Fig. 5.5. Since muscle activities less than 0.003 are mostly caused by suboptimal results and round-off calculation errors, muscles with activations less than this amount are not reported. Instead, active muscles with activations more than 0.003 are studied. These muscles are: Muscle 1, mono-articular shoulder flexor; Muscle 4, mono-articular shoulder extensor; and Muscle 5, bi-articular shoulder-elbow flexor.

In Fig. 5.5, the amount of activations for assistive (i.e., delayed hand movement) mode in most of the path is less than the other two modes. This result shows the success of the controller in providing assistance. However, the amount of assistance provided is not that high because the rehabilitated hand is performing well. The healthy hand is not moving precisely along the path; thus, the robot is helping to reduce the position error. Bi-articular muscle activations are more dependent on the interaction force field rather than the traveled path by the hand [24]. Hence, in all three modes, because of the presence of the interaction force, bi-articular Muscle 5 is activated.

In the assistive mode, the robot tries to assist by following the hand movement, so the position error is increased along the path (see Fig. 5.6). For the resistive (i.e., time-advanced hand movement) mode, since the desired force input is kept constant in all modes, the robot assists the hand along the path (the robot can resist if the input desired force is defined in the resisting direction). In the impaired hand movement, since the robot is interacting with a dysfunctional subject, it tries to generate the desired interaction force by increasing the error in the tangential direction. In the normal direction it tries to do position tracking while keeping the normal force bounded. However, since the desired interaction force is constant and relatively high, the robot causes large errors in tangential force tracking and normal position tracking.

In three modes of simulation (healthy, delayed, and time-advanced hand movements), the tangential force has converged to the desired input force (-0.1 N, which is in the assistance direction) and the tangential position error is bounded; on the other hand,

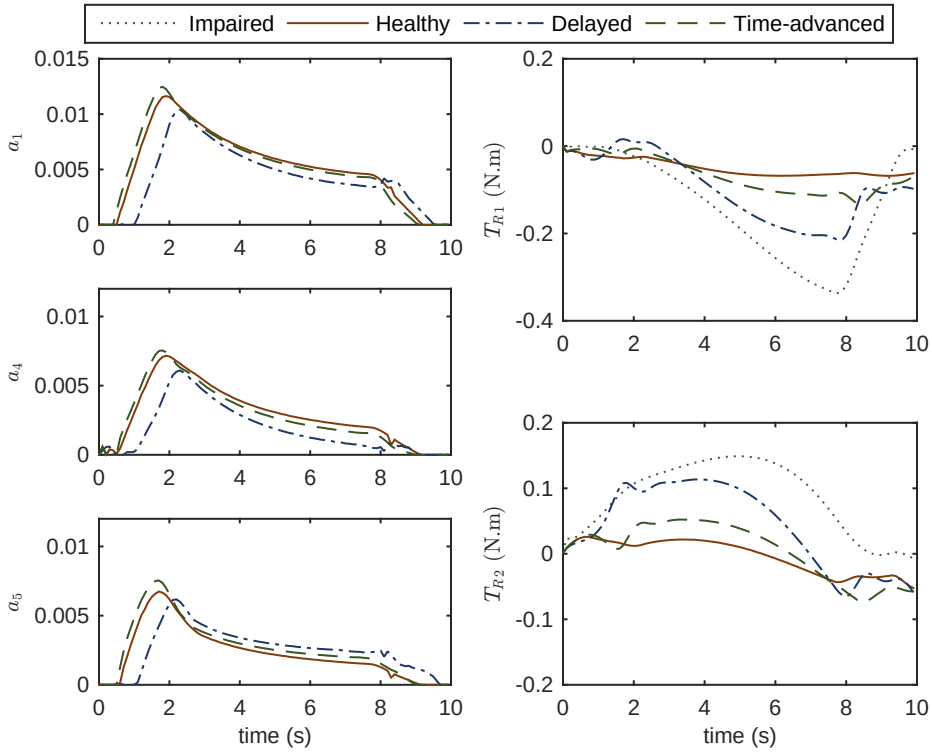


Figure 5.5: Activations of the active muscles (on the left) and input motor torques (on the right) in four modes of simulation while controlling the robot with HFIC. Note that in the impaired mode, muscle activations are zero.

the normal force is bounded, and the normal position error is small (less than 1mm, see Fig. 5.6). These are due to the use of the HFIC, which results in the tangential force tracking with the bounded tangential position error, and the normal position tracking with the bounded normal force in the operational space.

5.1.5 Conclusion

In a subspace (tangent to the path), the proposed low-level controller (HFIC) is trying to ensure force tracking with bounded position error. While in an orthogonal subspace (normal to the path), the position tracking with bounded force error is achieved. The force tracking needs a proper knowledge of the interacting system (human body) to define the desired force that the subject can safely tolerate. Thus, although HFIC is successful

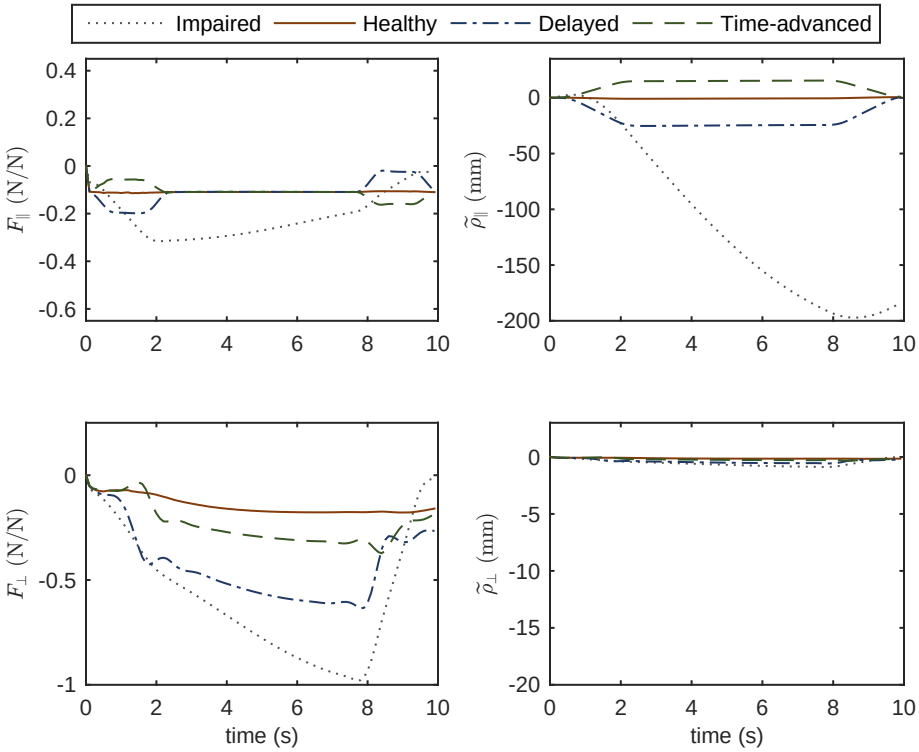


Figure 5.6: Operational space normalized interaction force and position error in four modes of simulations while controlling the robot with HFIC. Where subscripts $\parallel$ and $\perp$ indicate the tangent and normal directions, respectively.

in assisting the healthy and less strong subjects, it is not recommended to be used in interaction with weak subjects. This disadvantage can be resolved by introducing an outer-loop controller for HFIC that will be discussed in Section 5.3.

5.2 Optimal impedance control

To deal with variable admittance environments, despite other methods of partially assistive control (i.e., attractive force-field control, model-based assistance, learning-based assistance, counter-balance-based assistance, and performance-based adaptive control), adaptive and optimal forms of impedance control have been developed. Hussain et al. [127] used an adaptive impedance control for patient-cooperative therapy of a lower-limb exo-

skeleton, and they verified the controller performance using an experimental setup. In more recent studies, optimal impedance controls for an exoskeleton gait trainer and elbow rehabilitation robot were developed [72, 308]. The proposed methods were implemented in a computer simulation, and the real-time performance of the controllers was not discussed. In an exoskeleton, the impedance control is defined in the joint space, while in a manipulandum, the impedance model is in the operational space. Thus, the controllers developed for exoskeletons are not suitable for a manipulandum. Furthermore, these controllers are developed for some sort of predefined rhythmic motions and they are not implementable for random reaching movements. In some studies, to improve impedance control performance, the compliance has been controlled by an outer-loop force control [83, 100, 258]. However, this method only controls the compliance (i.e. interaction force) term, and the impedance gains are not optimal. Thus, it results in a superficial variable impedance model and requires careful selection of the gains. In a recent related study, Maldonado et al. [171] used a stiffness-based tuning for an adaptive impedance control of an upper extremity manipulandum; nevertheless, the method was verified using computer simulations only and its real-time capabilities were not mentioned.

In the previous section, to improve impedance control performance, the compliance is controlled by an outer loop force control. However, this method only controls the compliance (i.e., interaction force) term, and the impedance gains are not optimal. Thus, it results in a superficial variable impedance model and requires careful selection of the gains.

A general solution for an optimal impedance problem can be obtained with optimization techniques (i.e., an optimal control approach). Such techniques can resolve the issue with variable admittance environments and different robot configurations. However, real-time control of the system limits the utilizable non-linear optimization methods. Ding et al. [69] used a musculoskeletal human model (without including muscle dynamics) together with surface elecromyography (sEMG) signals to implement model-based assistance control on a rehabilitation *exoskeleton*, and this controller which is developed for an exoskeleton, is not suitable for a manipulandum.

Since multi-link manipulanda are controlled in the joint space to achieve the desired impedance at the end-effector in the operational space, the optimal impedance gains should be assigned to the different robot configurations. For different configurations, the manipulability ellipsoid in robotics is introduced to determine the easiest manipulation direction [327]. Thus, a method is required to optimally change the impedance gains based on the robot's manipulability ellipsoid. Hogan [118] solved an optimal impedance problem, the Standard Optimal Impedance Control (SOIC), which minimizes an objective function with position and force penalty. This problem was solved using covariance propagation equations. To the knowledge of the author, there is no other optimal impedance control

approach that has resolved different robot configuration problem independently.

There is a lack of study in the design of *real-time* and optimal impedance control for different configurations of rehabilitation manipulanda. Furthermore, previous low-level controls of rehabilitation manipulanda have not included human-robot interactions for the adjustment of the robot controller. The objective of this work is to design a general real-time optimal impedance control for rehabilitation manipulanda. This controller is designed to reduce therapist intervention (with fewer gain adjustments) and improve the quality of therapy in terms of safety (less interaction force based on robot manipulability) and rehabilitation (optimal tracking). In this section, we present an Optimal Impedance Control (OIC) for the upper extremity stroke rehabilitation robot; adjustment and performance-evaluation of the controller is done by simulating the robot interacting with a musculoskeletal upper extremity model. Here, a general method that optimally adjusts impedance gains for variable robot configurations is developed and tuned by simulating the human-robot system. The proof that justifies the existence of a Linear Time-Variant (LTV) impedance model is provided. The controller is implemented on the robot. Then, the performance of the controller in terms of interaction force and tracking accuracy is evaluated and compared to the SOIC [118] through simulations and experiments. In experiments, a complete dynamic model of the robot including joint and end-effector frictions, and joint stiffness are considered.

5.2.1 Standard optimal impedance control review

For an impedance control with an inner loop control law in (5.1), if we use (5.8) as the outer loop control law, a Linear Time-Invariant (LTI) impedance model defined in (5.4) will result. For this controller, considering a diagonal apparent mass matrix, the standard optimum stiffness and damping are as follows [118]:

$$\begin{cases} \mathbf{M}_{imp} = \text{Diag}\left(M_{imp1}, M_{imp2}\right) \\ \mathbf{B}_{imp} = \text{Diag}\left(\frac{F_{tol1}}{y_{tol1}}, \frac{F_{tol2}}{y_{tol2}}\right) \\ \mathbf{K}_{imp} = \text{Diag}\left(\sqrt{\frac{2F_{tol1}M_{imp1}}{y_{tol1}}}, \sqrt{\frac{2F_{tol2}M_{imp2}}{y_{tol2}}}\right) \end{cases} \tag{5.11}$$

where $y_{tol,k}$ and $F_{tol,k}$ are the k$^{\text{th}}$ element of the force and position vector tolerances.

5.2.2 Controller Design

In an optimal control structure, it is desired to carry out a desired task while minimizing a cost functional. For linear system dynamics, the system can be controlled by a Linear Quadratic Regulator (LQR), which is suitable for real-time optimal control. If we consider the dynamic equation of the robot (4.22), the state-space representation for this equation can be expressed as:

$$\dot{\mathbf{x}}_{\mathbf{q}} = \left\{ \begin{array}{c} \dot{q}_1 \\ \dot{q}_2 \\ \mathbf{M}_R^{-1}(\mathbf{u} - \mathbf{\Gamma}_R(\mathbf{q}, \dot{\mathbf{q}}, \mathbf{0})) \end{array} \right\} = \mathfrak{F}(\mathbf{x}_{\mathbf{q}}, \mathbf{u}) \tag{5.12}$$

where:

$$\mathbf{u} = \mathbf{T}_R - \mathbf{J}_R^T \mathbf{F}_{ext} \tag{5.13}$$

and

$$\mathbf{x}_{\mathbf{q}} = \left\{ \begin{array}{c} \mathbf{q} \\ \dot{\mathbf{q}} \end{array} \right\} = \left\{ \begin{array}{c} q_1 \\ q_2 \\ \dot{q}_1 \\ \dot{q}_2 \end{array} \right\} \tag{5.14}$$

The objective is to develop a real-time controller that optimizes impedance gains at different configurations. Since the state-space representation (5.12) is nonlinear, application of nonlinear optimal control approaches will be limited by the computation time. On the other hand, if (5.12) was linear, a linear optimal controller could solve this problem in real-time. The robot performs point to point reaching tasks in the horizontal plane [169]. Thus, we can perform Jacobian linearization on the robot dynamics along the preplanned rehabilitation trajectory to apply a systematic linear control technique (i.e., LQR), which can allow for real-time control. The LTV state-space equation of the robot's error dynamics will be:

$$\begin{aligned} \dot{\mathbf{x}}_{\mathbf{q}d} - \dot{\mathbf{x}}_{\mathbf{q}} &= \left. \frac{\partial \mathfrak{F}}{\partial \mathbf{x}_{\mathbf{q}}} \right|_{\mathbf{x}_{\mathbf{q}} = \mathbf{x}_{\mathbf{q}d}, \mathbf{u} = \mathbf{u}_d} (\mathbf{x}_{\mathbf{q}d} - \mathbf{x}_{\mathbf{q}}) + \left. \frac{\partial \mathfrak{F}}{\partial \mathbf{u}} \right|_{\mathbf{x}_{\mathbf{q}} = \mathbf{x}_{\mathbf{q}d}, \mathbf{u} = \mathbf{u}_d} (\mathbf{u}_d - \mathbf{u}) \\ &= \dot{\widetilde{\mathbf{x}}}_{\mathbf{q}} = \mathbb{A}_{\mathbf{q}} \widetilde{\mathbf{x}}_{\mathbf{q}} + \mathbb{B}_{\mathbf{q}} \widetilde{\mathbf{u}} \end{aligned} \tag{5.15}$$

where $\mathbb{A}$ and $\mathbb{B}$ are the state and input matrices, respectively. The desired control input is defined by the following equation:

$$\mathbf{u}_d = \mathbf{\Gamma}_R(\mathbf{q}_d, \dot{\mathbf{q}}_d, \ddot{\mathbf{q}}_d) \tag{5.16}$$

At each operational point, which is defined every $1000/\nu$ ms of the rehabilitation trajectory, the model is linearized and the interaction force is applied to the robot. It is worth noting that, if very few operational points are defined, the system may be biased into optimizing for static situations. At each operational point, there is an LTV impedance model which is relating the operational space error to interaction force:

$$\mathbf{F}_{ext}^{\Sigma} = \mathbf{M}_{imp}\widetilde{\ddot{\boldsymbol{\rho}}}^{\Sigma} + \mathbf{B}_{imp}\widetilde{\dot{\boldsymbol{\rho}}}^{\Sigma} + \mathbf{K}_{imp}\widetilde{\boldsymbol{\rho}}^{\Sigma} \tag{5.17}$$

The above equation is similar to the LTI impedance model (5.4) except the desired interaction force is zero (i.e., $\widetilde{\mathbf{F}}_{ext}^{\Sigma} = -\mathbf{F}_{ext}^{\Sigma}$), and the impedance gains ($\mathbf{M}_{imp}$, $\mathbf{B}_{imp}$ and $\mathbf{K}_{imp}$) are time-dependent. The LTV state-space equation (5.15) is in terms of errors, so we can use the infinite time[1] LQR to optimally control the robot along the desired trajectory. For the LQR approach, the quadratic cost functional is:

$$\mathfrak{J}_{\mathbf{q}} = \frac{1}{2}\int_0^{\infty} \left(\widetilde{\mathbf{x}}_{\mathbf{q}}^T \mathbb{Q}_{\mathbf{q}} \widetilde{\mathbf{x}}_{\mathbf{q}} + \widetilde{\mathbf{u}}^T \mathbb{R}_{\mathbf{q}} \widetilde{\mathbf{u}}\right) dt \tag{5.18}$$

The above cost functional is for minimizing the joint space error together with the consumed energy. An impedance control approach controls the robot performance in the operational space as in (5.17) [11, 117, 258]. Thus, for an optimal impedance control it will be desired to minimize the operational space error together with the operationally applied force (effort) error while satisfying (5.17). In other words, the following cost functional is more appropriate than (5.18):

$$\mathfrak{J}_{\Sigma} = \frac{1}{2}\int_0^{\infty} \left(\widetilde{\mathbf{x}}_{\Sigma}^T \mathbb{Q}_{\Sigma} \widetilde{\mathbf{x}}_{\Sigma} + \widetilde{\mathbb{F}}_{\Sigma}^T \mathbb{R}_{\Sigma} \widetilde{\mathbb{F}}_{\Sigma}\right) dt \tag{5.19}$$

where $\widetilde{\mathbb{F}}_{\Sigma}$ is the operational space transformation of the applied force error in the Cartesian space ($\widetilde{\mathbb{F}}$):

$$\begin{cases} \mathbb{F} = \mathbf{J}_R^{-T}\mathbf{u} = \mathbf{J}_R^{-T}\mathbf{T}_R - \mathbf{F}_{ext} \\ \widetilde{\mathbb{F}} \approxeq \mathbf{J}_{Rd}^{-T}\widetilde{\mathbf{u}} \end{cases} \tag{5.20}$$

To solve the LQR problem with the updated cost functional in (5.19), we use the mapping from the operational space into the joint space and then solve the LQR problem with the ordinary cost functional in (5.18).

[1]This research is focused on "Errand Completion Tasks" as opposed to "Time Management Tasks" [265], i.e. we assume that the timing in performing the rehabilitation task is not critical.

Mapping the operational into joint space

We define the joint, Cartesian and operational state errors as follows:

$$\widetilde{\mathbf{x}}_{\mathbf{q}} = \left\{ \begin{array}{c} \mathbf{q}_d - \mathbf{q} \\ \dot{\mathbf{q}}_d - \dot{\mathbf{q}} \end{array} \right\} = \left\{ \begin{array}{c} \widetilde{\mathbf{q}} \\ \widetilde{\dot{\mathbf{q}}} \end{array} \right\} \tag{5.21}$$

$$\widetilde{\mathbf{x}}_{\boldsymbol{\rho}} = \left\{ \begin{array}{c} \boldsymbol{\rho}_d - \boldsymbol{\rho} \\ \dot{\boldsymbol{\rho}}_d - \dot{\boldsymbol{\rho}} \end{array} \right\} = \left\{ \begin{array}{c} \widetilde{\boldsymbol{\rho}} \\ \widetilde{\dot{\boldsymbol{\rho}}} \end{array} \right\} \tag{5.22}$$

$$\widetilde{\mathbf{x}}_{\Sigma} = \left\{ \begin{array}{c} \boldsymbol{\rho}_d^{\Sigma} - \boldsymbol{\rho}^{\Sigma} \\ \dot{\boldsymbol{\rho}}_d^{\Sigma} - \dot{\boldsymbol{\rho}}^{\Sigma} \end{array} \right\} = \left\{ \begin{array}{c} \widetilde{\boldsymbol{\rho}}^{\Sigma} \\ \widetilde{\dot{\boldsymbol{\rho}}}^{\Sigma} \end{array} \right\} \tag{5.23}$$

Based on the inverse kinematics of the robot (4.21), the geometric Jacobian definition [258] and first-order Taylor series expansion, the relation between the Cartesian and joint space errors can be defined as:

$$\left\{ \begin{array}{l} \widetilde{\boldsymbol{\rho}} \approxeq \mathbf{J}_{Rd} \widetilde{\mathbf{q}} \\ \widetilde{\dot{\boldsymbol{\rho}}} \approxeq \dot{\mathbf{J}}_{Rd} \widetilde{\mathbf{q}} + \mathbf{J}_{Rd} \widetilde{\dot{\mathbf{q}}} \end{array} \right. \tag{5.24}$$

Thus, the operational state error in terms of the joint state error can be defined by the following equation:

$$\widetilde{\mathbf{x}}_{\boldsymbol{\rho}} \approxeq \left[\begin{array}{cc} \mathbf{J}_{Rd} & \mathbf{0} \\ \dot{\mathbf{J}}_{Rd} & \mathbf{J}_{Rd} \end{array} \right] \widetilde{\mathbf{x}}_{\mathbf{q}} \tag{5.25}$$

Consider Fig. 5.3.a, the operational coordinate (Σ:n-t) is the rotated and translated Cartesian coordinate (G:ZX) by angle θ_{Σ} and desired position vector $\boldsymbol{\rho}_d$, respectively; thus, the relation between the operational and Cartesian space errors can be defined as:

$$\left\{ \begin{array}{l} \widetilde{\boldsymbol{\rho}} = \mathbf{R}_{\Sigma} \widetilde{\boldsymbol{\rho}}^{\Sigma} \\ \widetilde{\dot{\boldsymbol{\rho}}} = \dot{\mathbf{R}}_{\Sigma} \widetilde{\boldsymbol{\rho}}^{\Sigma} + \mathbf{R}_{\Sigma} \widetilde{\dot{\boldsymbol{\rho}}}^{\Sigma} \end{array} \right. \tag{5.26}$$

By defining $\boldsymbol{\varpi}_{\Sigma}$ as the skew symmetric matrix of the angular velocity ($\dot{\theta}_{\Sigma}$), the operational state error can be defined in terms of the Cartesian state error as:

$$\widetilde{\mathbf{x}}_{\boldsymbol{\rho}} = \left[\begin{array}{cc} \mathbf{R}_{\Sigma} & \mathbf{0} \\ \boldsymbol{\varpi}_{\Sigma} \mathbf{R}_{\Sigma} & \mathbf{R}_{\Sigma} \end{array} \right] \widetilde{\mathbf{x}}_{\Sigma} \tag{5.27}$$

Finally, the operational and joint state errors can be related as:

$$\widetilde{\mathbf{x}}_{\Sigma} \approxeq \left[\begin{array}{cc} \mathbf{R}_{\Sigma} & \mathbf{0} \\ \boldsymbol{\varpi}_{\Sigma} \mathbf{R}_{\Sigma} & \mathbf{R}_{\Sigma} \end{array} \right]^{-1} \left[\begin{array}{cc} \mathbf{J}_{Rd} & \mathbf{0} \\ \dot{\mathbf{J}}_{Rd} & \mathbf{J}_{Rd} \end{array} \right] \widetilde{\mathbf{x}}_{\mathbf{q}} = \left[\begin{array}{cc} \mathbf{J}_{Rd}^{\Sigma} & \mathbf{0} \\ \dot{\mathbf{J}}_{Rd}^{\Sigma} & \mathbf{J}_{Rd}^{\Sigma} \end{array} \right] \widetilde{\mathbf{x}}_{\mathbf{q}} = \boldsymbol{\mathfrak{T}}_{\mathbf{q}}^{\Sigma} \widetilde{\mathbf{x}}_{\mathbf{q}} \tag{5.28}$$

Building updated LQR matrices

We define $\mathbb{Q}_\Sigma$ and $\mathbb{R}_\Sigma$ in the operational space cost functional (5.19) as positive definite diagonal matrices. Using the mapping equation (5.28) we can correlate the first terms of the two quadratic cost functionals (5.18,5.19); thus, $\mathbb{Q}_\mathbf{q}$ can be defined as:

$$\mathbb{Q}_\mathbf{q} = \left(\mathfrak{T}_\mathbf{q}^\Sigma\right)^T \mathbb{Q}_\Sigma \mathfrak{T}_\mathbf{q}^\Sigma \tag{5.29}$$

Note that $\mathbb{Q}_\mathbf{q}$ is positive definite, since $\mathbb{Q}_\Sigma$ is positive definite. Since:

$$\widetilde{\mathbb{F}}_\Sigma = \mathbf{R}_\Sigma \widetilde{\mathbb{F}} \tag{5.30}$$

considering (5.20) we can also rearrange the energy term in the joint space cost functional (5.18) as:

$$\begin{aligned} \widetilde{\mathbf{u}}^T \mathbb{R}_\mathbf{q} \widetilde{\mathbf{u}} &= \left(\mathbf{J}_{Rd}^T \widetilde{\mathbb{F}}\right)^T \mathbb{R}_\mathbf{q} \left(\mathbf{J}_{Rd}^T \widetilde{\mathbb{F}}\right) = \left(\mathbf{J}_{Rd}^T \mathbf{R}_\Sigma \widetilde{\mathbb{F}}_\Sigma\right)^T \mathbb{R}_\mathbf{q} \left(\mathbf{J}_{Rd}^T \mathbf{R}_\Sigma \widetilde{\mathbb{F}}_\Sigma\right) \\ &= \widetilde{\mathbb{F}}_\Sigma^T \mathbf{R}_\Sigma^T \mathbf{J}_{Rd} \mathbb{R}_\mathbf{q} \mathbf{J}_{Rd}^T \mathbf{R}_\Sigma \widetilde{\mathbb{F}}_\Sigma \geqslant 0 \end{aligned} \tag{5.31}$$

Now since $\mathbb{R}_\Sigma$ is positive definite, if:

$$\mathbb{R}_\mathbf{q} = \left(\mathbf{R}_\Sigma^T \mathbf{J}_{Rd}\right)^{-1} \mathbb{R}_\Sigma \left(\mathbf{J}_{Rd}^T \mathbf{R}_\Sigma\right)^{-1} = \left(\mathbf{J}_{Rd}^\Sigma\right)^{-1} \mathbb{R}_\Sigma \left(\mathbf{J}_{Rd}^\Sigma\right)^{-T} \tag{5.32}$$

$\mathbb{R}_\mathbf{q}$ is also positive definite unless the robot is at a singularity point. Based on (5.32), minimizing the energy term in the joint space cost functional (5.18) will indirectly minimize the energy term in the operational space cost functional (5.19).

Optimal impedance control

With the updated LQR matrices, the optimal impedance controller scheme takes the structure shown in Fig. 5.7. Using (5.13), the driving torque will be:

$$\mathbf{T}_R = \mathbf{u} + \mathbf{J}_R^T \mathbf{F}_{ext} \tag{5.33}$$

where the control input $\mathbf{u}$ is defined such that it should optimally control the error dynamics (by $-\widetilde{\mathbf{u}}$) while applying the nominal control input ($\mathring{\mathbf{u}}$). Thus, it will have the following form:

$$\begin{cases} \mathbf{u} \triangleq \mathring{\mathbf{u}} - \widetilde{\mathbf{u}} \\ \mathring{\mathbf{u}} = \mathbf{u}_d - \mathbf{J}_{Rd}^T \mathbf{F}_{ext} \\ \widetilde{\mathbf{u}} = -\mathbb{K} \widetilde{\mathbf{x}}_\mathbf{q} \end{cases} \tag{5.34}$$

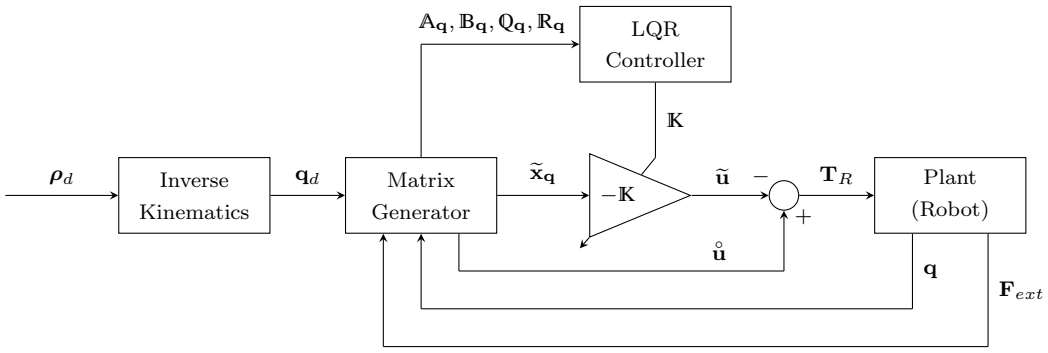


Figure 5.7: Optimal impedance controller scheme. $\boldsymbol{\rho}_d$ and $\mathbf{q}_d$ are the desired positions in the operational and joint spaces, respectively. $\widetilde{\mathbf{x}}_\mathbf{q}$ is the state error vector in the joint space, $\widetilde{\mathbf{u}}$ is the optimal control input for the error dynamics, and $\mathring{\mathbf{u}}$ is the nominal control input (desired torque minus the torque caused by the interaction force). $\mathbb{A}_\mathbf{q}$, $\mathbb{B}_\mathbf{q}$, and $\mathbb{Q}_\mathbf{q}$ and $\mathbb{R}_\mathbf{q}$ are the time-varying state, input, and LQR gain matrices, respectively.

Note that the nominal control input is equal to the desired system dynamics (desired control input) minus the torque caused by the interaction force at any desired location. This subtraction ($\mathbf{u}_d - \mathbf{J}_{Rd}^T \mathbf{F}_{ext}$) at a zero tracking error will lead to a zero desired interaction force in (5.17). Finally, the equations (5.33) and (5.34) are used to satisfy the impedance model (5.17) (as shown in the next subsection), in order to overcome the robot dynamics and interaction force.

Satisfying the impedance model?

By applying the control law (5.33) and (5.34) to the robot dynamics (4.22) and substituting $\mathbf{u}_d$ from (5.16), we get:

$$\mathbb{K}\widetilde{\mathbf{x}}_\mathbf{q} + \boldsymbol{\Gamma}_R(\mathbf{q}_d, \dot{\mathbf{q}}_d, \ddot{\mathbf{q}}_d) - \mathbf{J}_{Rd}^T \mathbf{F}_{ext} = \boldsymbol{\Gamma}_R(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}) \tag{5.35}$$

Using Taylor series expansion and (5.21), (5.35) can be rearranged as:

$$\begin{aligned} \mathbf{J}_{Rd}^T \mathbf{F}_{ext} &= \mathbb{K}\widetilde{\mathbf{x}}_\mathbf{q} + \mathbf{M}_{Rd}\widetilde{\ddot{\mathbf{q}}} + \left.\frac{\partial \boldsymbol{\Gamma}_R}{\partial \dot{\mathbf{q}}}\right|_{\mathbf{q}=\mathbf{q}_d, \dot{\mathbf{q}}=\dot{\mathbf{q}}_d} \widetilde{\dot{\mathbf{q}}} + \left.\frac{\partial \boldsymbol{\Gamma}_R}{\partial \mathbf{q}}\right|_{\mathbf{q}=\mathbf{q}_d, \dot{\mathbf{q}}=\dot{\mathbf{q}}_d} \widetilde{\mathbf{q}} \\ &= \mathbf{M}_{Rd} \begin{bmatrix} \mathbf{0} & \mathbb{I} \end{bmatrix} \widetilde{\dot{\mathbf{x}}}_\mathbf{q} + \begin{bmatrix} \mathbb{K}_P & \mathbb{K}_D \end{bmatrix} \widetilde{\mathbf{x}}_\mathbf{q} \end{aligned} \tag{5.36}$$

where $\mathbb{I}$ is an identity matrix, and:

$$\begin{cases} \mathbb{K} = \begin{bmatrix} \mathbb{K}_1 & \mathbb{K}_2 \end{bmatrix} \\ \mathbb{K}_P = \frac{\partial \mathbf{\Gamma}_R}{\partial \mathbf{q}}\Big|_{\mathbf{q}=\mathbf{q}_d, \dot{\mathbf{q}}=\dot{\mathbf{q}}_d} + \mathbb{K}_1 \\ \mathbb{K}_D = \frac{\partial \mathbf{\Gamma}_R}{\partial \dot{\mathbf{q}}}\Big|_{\mathbf{q}=\mathbf{q}_d, \dot{\mathbf{q}}=\dot{\mathbf{q}}_d} + \mathbb{K}_2 \end{cases} \tag{5.37}$$

The following equations can be derived from (5.28):

$$\widetilde{\mathbf{x}}_{\mathbf{q}} \approxeq \begin{bmatrix} \mathbf{J}_{Rd}^{\Sigma} & \mathbf{0} \\ \dot{\mathbf{J}}_{Rd}^{\Sigma} & \mathbf{J}_{Rd}^{\Sigma} \end{bmatrix}^{-1} \widetilde{\mathbf{x}}_{\Sigma} = \begin{bmatrix} \left(\mathbf{J}_{Rd}^{\Sigma}\right)^{-1} & \mathbf{0} \\ -\left(\mathbf{J}_{Rd}^{\Sigma}\right)^{-1} \dot{\mathbf{J}}_{Rd}^{\Sigma} \left(\mathbf{J}_{Rd}^{\Sigma}\right)^{-1} & \left(\mathbf{J}_{Rd}^{\Sigma}\right)^{-1} \end{bmatrix} \widetilde{\mathbf{x}}_{\Sigma} = \mathfrak{T}_{\Sigma}^{\mathbf{q}} \widetilde{\mathbf{x}}_{\Sigma} \tag{5.38}$$

$$\dot{\widetilde{\mathbf{x}}}_{\mathbf{q}} \approxeq \dot{\mathfrak{T}}_{\Sigma}^{\mathbf{q}} \widetilde{\mathbf{x}}_{\Sigma} + \mathfrak{T}_{\Sigma}^{\mathbf{q}} \dot{\widetilde{\mathbf{x}}}_{\Sigma} \tag{5.39}$$

thus, (5.36) can be written as:

$$\mathbf{J}_{Rd}^{T} \mathbf{F}_{ext} = \mathbf{M}_{Rd} \begin{bmatrix} \mathbf{0} & \mathbb{I} \end{bmatrix} \mathfrak{T}_{\Sigma}^{\mathbf{q}} \dot{\widetilde{\mathbf{x}}}_{\Sigma} + \mathbf{M}_{Rd} \begin{bmatrix} \mathbf{0} & \mathbb{I} \end{bmatrix} \dot{\mathfrak{T}}_{\Sigma}^{\mathbf{q}} \widetilde{\mathbf{x}}_{\Sigma} + \begin{bmatrix} \mathbb{K}_P & \mathbb{K}_D \end{bmatrix} \mathfrak{T}_{\Sigma}^{\mathbf{q}} \widetilde{\mathbf{x}}_{\Sigma} \tag{5.40}$$

(5.40) is corresponding to the LTV impedance model (5.17), if:

$$\begin{cases} \mathbf{M}_{imp} = (\mathbf{J}_{Rd}^{\Sigma})^{-T} \mathbf{M}_{Rd} (\mathbf{J}_{Rd}^{\Sigma})^{-1} \\ \mathbf{B}_{imp} = (\mathbf{J}_{Rd}^{\Sigma})^{-T} \mathbb{K}_D (\mathbf{J}_{Rd}^{\Sigma})^{-1} - 2\mathbf{M}_{imp} \dot{\mathbf{J}}_{Rd}^{\Sigma} (\mathbf{J}_{Rd}^{\Sigma})^{-1} \\ \mathbf{K}_{imp} = (\mathbf{J}_{Rd}^{\Sigma})^{-T} \mathbb{K}_P (\mathbf{J}_{Rd}^{\Sigma})^{-1} - \left(\mathbf{M}_{imp} \ddot{\mathbf{J}}_{Rd}^{\Sigma} + \mathbf{B}_{imp} \dot{\mathbf{J}}_{Rd}^{\Sigma}\right) (\mathbf{J}_{Rd}^{\Sigma})^{-1} \end{cases} \tag{5.41}$$

Choosing LQR gains

Matrices $\mathbb{Q}_{\Sigma}$ and $\mathbb{R}_{\Sigma}$ have diagonal weights:

$$\begin{cases} \mathbb{Q}_{\Sigma} = \mathrm{Diag}\left(\mathbb{Q}_{\Sigma 1}, \mathbb{Q}_{\Sigma 2}, \mathbb{Q}_{\Sigma 3}, \mathbb{Q}_{\Sigma 4}\right) \\ \mathbb{R}_{\Sigma} = \mathrm{Diag}\left(\mathbb{R}_{\Sigma 1}, \mathbb{R}_{\Sigma 2}\right) \end{cases} \tag{5.42}$$

where these weights are chosen such that the objective functional results in the allowable error associated with the state or effort, in other words:

$$\begin{cases} \mathbb{Q}_{\Sigma,i} = y_{tol,i}^{-2} \quad (i = 1..4) \\ \mathbb{R}_{\Sigma,j} = cF_{tol,j}^{-2} \quad (j = 1..2) \end{cases} \tag{5.43}$$

in which $y_{tol,k}$ and $F_{tol,k}$ are the allowable amount of the k^{th} element of the state ($\widetilde{\mathbf{x}}_{\Sigma}$) and effort ($\mathbb{F}_{\Sigma}$) vector errors, respectively. These weights should also be adjusted such that the (5.41) results in positive definite impedance gains. Coefficient $c > 0$ will be controlled by the therapist to adjust the effort/state balance.

5.2.3 Simulation and experiment

To evaluate the performance of the controller during a rehabilitation procedure, the same simulation procedures as in the HFIC evaluation are repeated. The performance of the proposed optimal controller OIC is compared to the SOIC, which is also designed for the robot to perform in four modes of the movement. See Tables A.14 and A.15 for the assigned control parameters.

To evaluate the performance of the controllers experimentally during a rehabilitation procedure, a healthy subject performed four modes of movement similar to the simulations (see Fig. 5.3.b). To this end, the following protocols are considered:

- Impaired-hand movement mode: the subject is asked to relax his/her upper extremity muscles and avoid any contractions as much as possible.
- Healthy hand movement mode: the subject should do his/her best in following the desired trajectory.
- Delayed hand movement mode: the subject is asked to follow a path that is delayed compared to the desired trajectory.
- Time-advanced hand movement mode: the subject should follow a path for which the desired trajectory is the delayed form.

In the above active protocols (i.e., last three protocols), the path point at each simulation time step is defined by a circle. The subject should try to keep the end-effector position inside the circle. To reduce the effect of random/noisy movements, each mode for each controller was performed in 10 trials. Tests of the two controllers were alternated randomly to reduce the effect of learning. Sampling-time frequency of the experiments is set to $\nu = 500$ Hz, and a fixed-step Euler solver is selected to solve the ordinary differential equations.

In contrast to admittance control, impedance control is used for back-drivable systems. Thus, for implementing the proposed optimal controller, we assume that the robot friction is negligible. In simulations, the robot model has no friction and the musculoskeletal model has only approximate parameters for the muscles and inertial properties; thus, we do not expect a close quantitative match between simulation and experimental results. Nevertheless, the model will be effective for the design and tuning of a feedback controller if a good qualitative match between simulation and experimental results is achieved.

5.2.4 Results and discussion

In this section, first, we assess the performance of the controllers (i.e., OIC and SOIC) in simulations based on the activation results from the musculoskeletal model interacting with the robot. In this section similar to the previous section, muscles with activations less than 0.003 are not reported, instead active muscles with activations more than 0.003 are studied. These muscles are: Muscle 1, mono-articular shoulder flexor; Muscle 4, mono-articular shoulder extensor; and Muscle 5, bi-articular shoulder-elbow flexor. As shown in Fig. 5.8, for the delayed hand movement in both controllers, the robot assistance has decreased the amount of muscle activations compared to the other modes for most of the path. This decrease for the OIC is more than the SOIC (see the Root-Mean Square (RMS) values for the delayed mode in Fig. 5.9), which shows the effectiveness of the OIC in the active-assisted therapy of the patients. For the time-advanced mode in both controllers, the amount of maximum muscle activations is higher than the other modes. In the healthy hand movement mode, both controllers result in the same amount of activation.

Second, the dynamic response of the system is used to evaluate the controllers. To this end, normalized interaction forces and position errors in the operational space are compared for four modes (see Fig. 5.10 and Fig. 5.11 for simulation and experimental results). Interaction force results are normalized to the maximum applied force in the horizontal plane to show similar trends to the approximate and highly idealized simulation model. Both in experiments and simulations, tangential interaction force plots show that the amount of assistance or resistance for the OIC is slightly more than the SOIC. Both in simulations and experiments, normal interaction force amount in the impaired hand movement mode for the OIC is not more than the SOIC, while for the other modes, the OIC results in higher values than the SOIC. This is because the position error in the normal direction is reduced by the OIC. However, the normal position error for the impaired hand movement mode in the SOIC is significantly more than the OIC. This shows that the optimal performance of the SOIC, especially in experiments, has failed to deal with impaired patients. The tangential position error is similar for both controllers.

In simulations with the SOIC (Fig. 5.10), after 3 seconds of the simulation, normal position error for the time-advanced hand movement is strictly increasing, and this will result in instability issues. However, this does not happen in experiments, since robot instability limited the selection of higher gains for the SOIC. Thus, in experiments, the robot in the SOIC is set to be more compliant. In simulations for the SOIC, between the position error and the interaction force, there is a linear relationship which is due to the LTI impedance model of the controller. However, for the OIC this relationship is nonlinear, and this is because of the LTV impedance model of the controller. One cannot

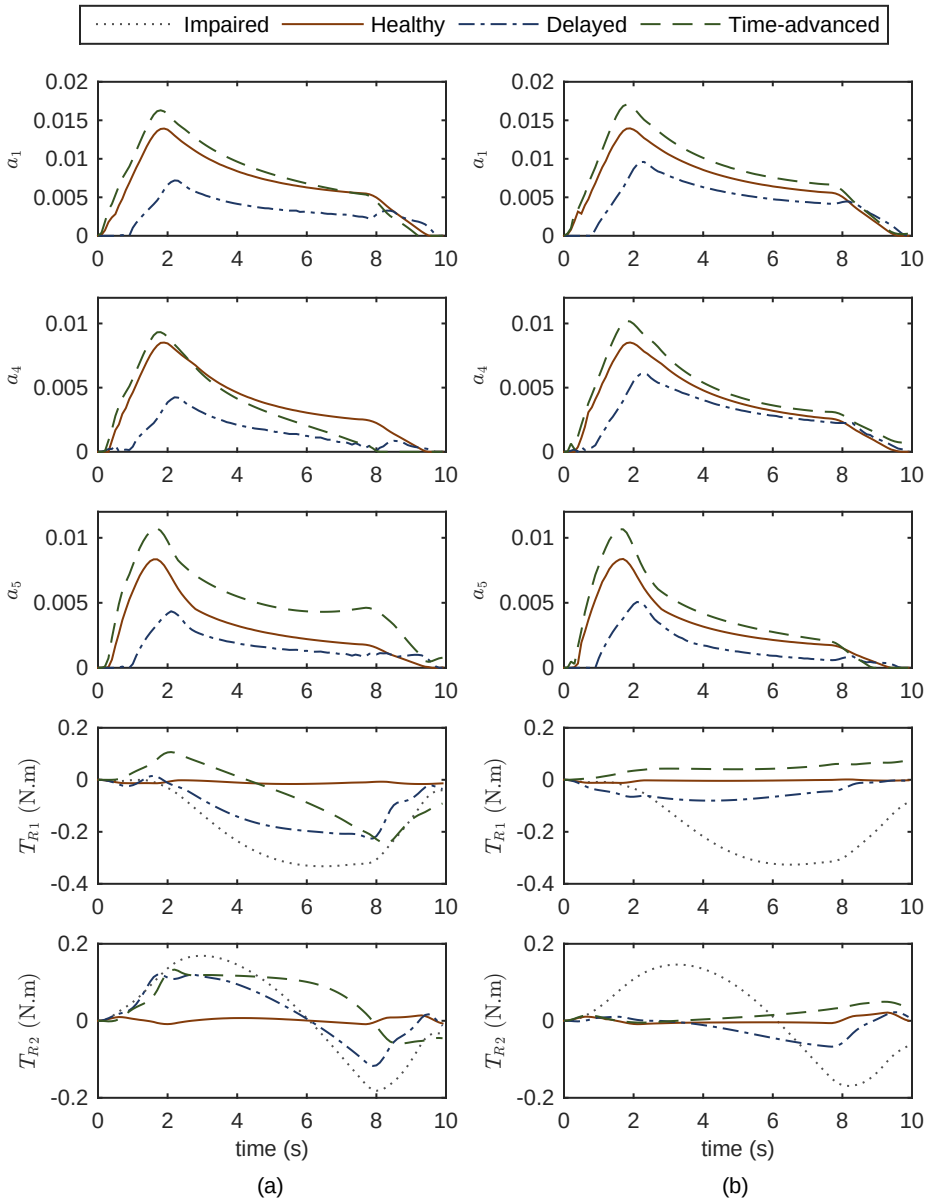


Figure 5.8: Activations of the active muscles and input motor torques in four modes of simulation while controlling the robot (a) OIC, and (b) SOIC. Note that in the impaired mode, muscle activations are zero.

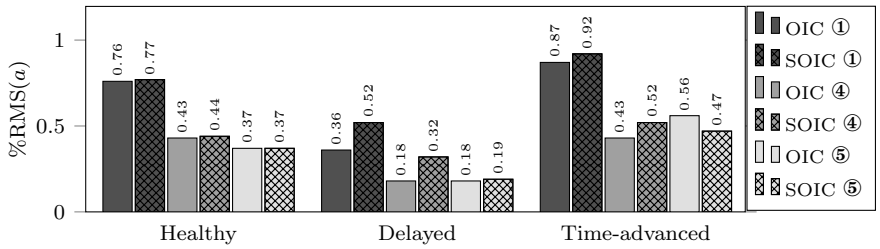


Figure 5.9: RMS of active muscle activations in three modes using OIC (solid fill) and SOIC (crosshatch fill). Circled numbers are corresponding to the active muscle numbers.

see this nonlinear relationship because the robot's frictional forces have changed the system behavior and made it linear.

At each operational point, the controllability ($\mathscr{C}$) and observability ($\mathscr{O}$) matrices are defined as:

$$\mathscr{C} = \begin{bmatrix} \mathbb{B}_{\mathbf{q}} & \mathbb{A}_{\mathbf{q}}\mathbb{B}_{\mathbf{q}} & \mathbb{A}_{\mathbf{q}}^2\mathbb{B}_{\mathbf{q}} & \mathbb{A}_{\mathbf{q}}^3\mathbb{B}_{\mathbf{q}} \end{bmatrix}_{4\times 8}, \qquad \mathscr{O} = \begin{bmatrix} \mathbb{I} \\ \mathbb{A}_{\mathbf{q}} \\ \mathbb{A}_{\mathbf{q}}^2 \\ \mathbb{A}_{\mathbf{q}}^3 \end{bmatrix}_{16\times 4} \tag{5.44}$$

For the OIC, the state-space model is controllable and observable because at each operational point, $\mathscr{C}$ and $\mathscr{O}$ are rank 4; furthermore, the dominant pole position of the LQR controller (which is the closest eigenvalue of $[\mathbb{A}_{\mathbf{q}} - \mathbb{B}_{\mathbf{q}}\mathbb{K}]$ to the imaginary axis) at each operational point has a negative real value, which makes the system critically damped. On the other hand, the optimum values of SOIC are such as to result in an under-damped system with a damping ratio of $\sqrt{2}/2$.

Here, we also compare the HFIC results with the other two controllers. Comparing figures 5.5 and 5.6 with 5.8 and 5.10, respectively, one can make following discussion:

- The resistance and assistance of the HFIC are lower than the other controllers, since the desired tangential force is less than the tangential interaction forces of the other controllers.
- Because of less resistance and assistance, the amount of motor torques for the HFIC is less than others.
- Normal position error for the HFIC is close to the OIC results.

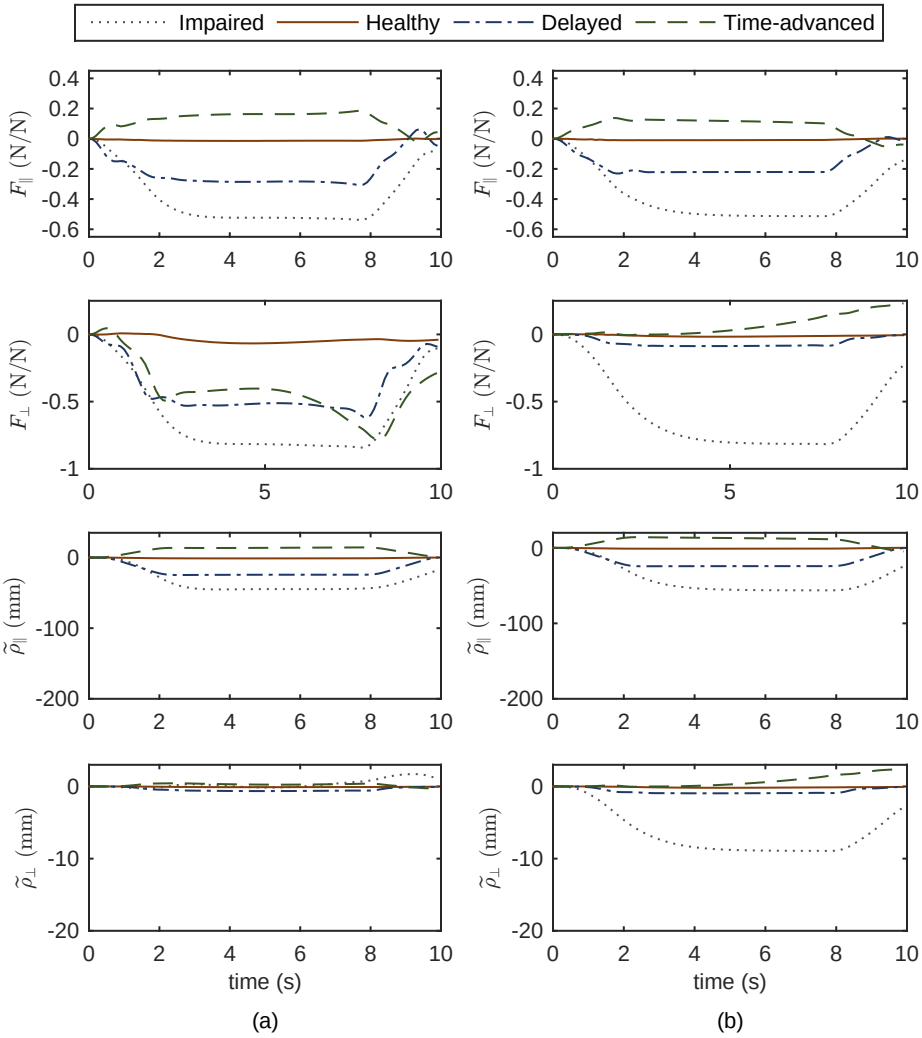


Figure 5.10: Operational space normalized interaction force and position error in four modes of simulations while controlling the robot with (a) OIC, (b) SOIC. Where subscripts $\parallel$ and $\perp$ indicate the tangent and normal directions, respectively.

- The HFIC similar to the SOIC is not successful in dealing with weak subjects (see the impaired hand movement results).

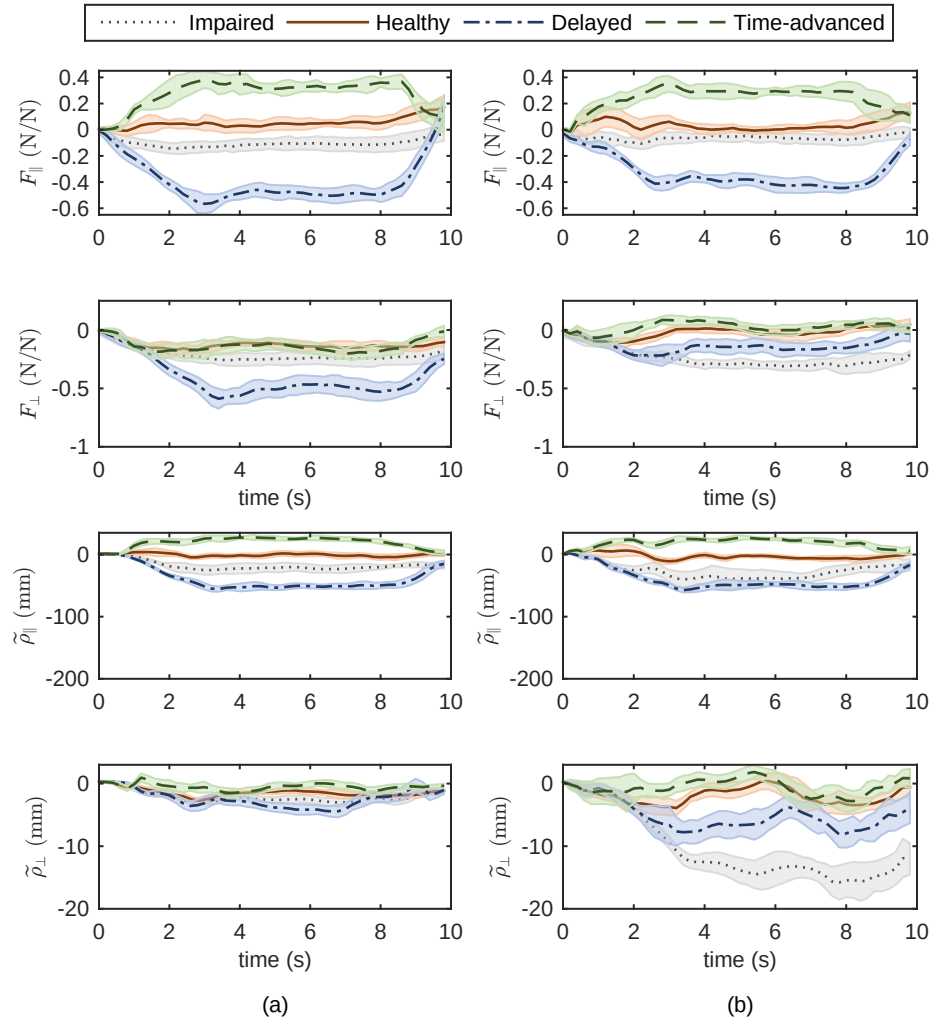


Figure 5.11: Operational space normalized interaction force and position error in four modes of experiments while controlling the robot with (a) OIC, (b) SOIC. The shaded area denotes twice the standard deviation at each instance of experiment.

5.2.5 Conclusion

In this section, we designed and verified a modified LQR controller for optimal impedance control, which indirectly considers the operational space and interaction forces. Despite some similarities to the SOIC, the OIC has proven to be more efficient in passive, active-

assisted, active-constrained therapy since it updates the impedance gains optimally during a reaching task (at different robot configurations). Physiologically, this efficient behavior causes less muscle activations in active-assisted therapy. Dynamically, the controller is more robust to the disturbances caused by unknown dynamics, and the tracking error and interaction force are in a safer region. The controller's computational cost is the same as that of the SOIC, even if the LQR gains are adjusted online. For experiments, since the QUARC does not support online LQR gain adjustments, an offline gain selection is done. Hence, the implemented controller can be considered an optimal passive trajectory tracking controller. In OIC, therapists will be able to modify the controller with a single parameter c in (5.43), which represents the effort/state balance weight; the inclusion of a single calibration parameter contributes to the superiority of the OIC over SOIC.

Here, an integrated human-robot dynamic system is used to fine-tune the controller gains. This method is advantageous for efficient tuning of the robot controllers in experiments. A good qualitative agreement between experiments and simulations verifies the effectiveness of this method.

Our proposed controller and tuning method can be used in any rehabilitation manipulandum system. Possible improvements for this method are as follows. First, for a linear robot model, the OIC assumes an apparent mass for the robot equal to its mass matrix, while the SOIC permits offline changes to the robot's apparent mass. For considering the patient interaction dynamics, the robot's apparent mass should vary online as a function of the input frequencies of the system. However, neither the OIC nor SOIC offer such updates. Moreover, in regards to experiments, the unknown dynamics of the robot presents a challenging issue, independent of the controller. In the next section, we present a method to implement an OIC on the robot which also allows for online changes to the robot's apparent mass. Second, in the impedance model (5.17), the desired interaction force is assumed to be zero, while for implementing any high-level controller that deals with variable admittance environments (different patients or the same patients at different stages of their therapy) this desired interaction force should be updated by an outer-loop control law. In the next section, we also develop the outer-loop controller to enhance the proposed OIC.

5.3 Enhancement of the proposed controllers

In this section, two methods will be introduced for the enhancement of the proposed controllers. The first method (attractive force field method), which is a type of high-level controller, is used to obtain a desired reference force that can be used in HFIC, SOIC, and

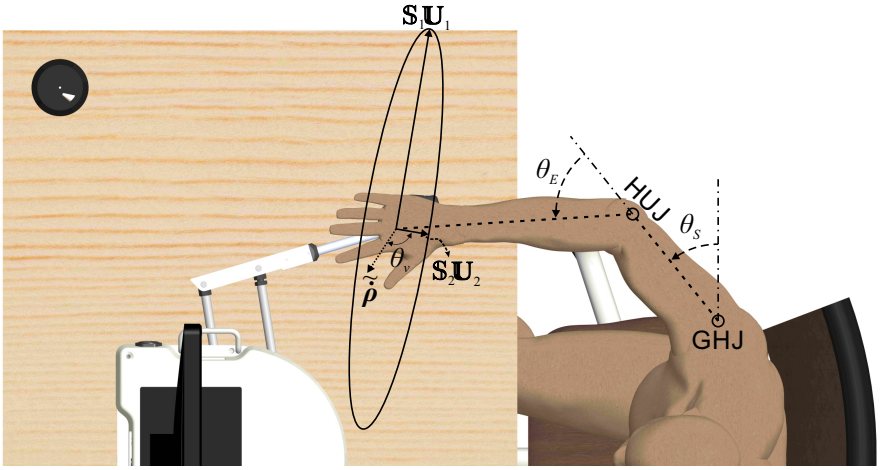


Figure 5.12: Manipulability ellipsoid in a single configuration of the human-robot rehabilitation system. $\mathbb{U}_1$ and $\mathbb{U}_2$ are the unit vectors of the major and minor axes of the ellipse. $\mathbb{S}_1$ and $\mathbb{S}_2$ are the lengths of the major and minor axes of the ellipse.

OIC. The second method, adaptive LQR control, is used to predict the apparent mass of the robot in SOIC or OIC.

5.3.1 Attractive force field

For a robotic arm, "manipulability ellipsoid" is used to determine the easiest manipulation direction [327, 333]. Based on this kinematic measure, the largest manipulating force can be generated along the minor axis of the manipulability ellipsoid. This ellipsoid is evaluated by Singular Value Decomposition (SVD) of the geometric Jacobian matrix. For example, if the geometric Jacobian is $\mathbf{J} \in \mathbb{R}^{2\times 2}$, then its SVD is:

$$\mathbf{J} = \mathbb{U}\mathbb{S}\mathbb{V}^T \tag{5.45}$$

where $\mathbb{U} = [\mathbb{U}_1, \mathbb{U}_2] \in \mathbb{R}^{2\times 2}$ is a unitary matrix, and $\mathbb{U}_1$ and $\mathbb{U}_2$ denote the direction of the major and minor axes of the manipulability ellipsoid. $\mathbb{S} = \text{Diag}(\mathbb{S}_1, \mathbb{S}_2)$ is a non-negative diagonal matrix, and $\mathbb{S}_1$ and $\mathbb{S}_2$ correspond to the lengths of the major and minor axes of the manipulability ellipsoid. $\mathbb{V}$ is the unitary matrix defined in the joint space.

If we represent the human upper extremity as a robotic arm, we can use this kinematic measure to obtain the attractive force field for the upper extremity. This can be achieved by following these steps:

1. Desired force direction: in this chapter, since we are working with black-box controllers, we do not utilize the dynamics of the upper extremity (i.e., musculoskeletal models) in the controller structure. However, in a black-box controller, if the patient's trunk is secured with respect to the robot, it is possible to obtain the kinematics of the 2D upper extremity based on the robot end-effector position (4.23). Once the arm kinematics are obtained, the geometric Jacobian $\mathbf{J}_A$ can be evaluated at each point on the desired trajectory (i.e., Path1). If we evaluate the SVD of the geometric Jacobian by using (5.45), $\mathbb{U}_2$ will represent the desired force direction (see Fig. 5.12). Next, we should obtain the magnitude of the desired force.

2. Desired force magnitude ($\hat{F}_d$): before starting the robotic rehabilitation, the patient is asked to apply maximum static forces to the locked end-effector positioned at different locations of the workspace. The patient is instructed to apply the force along the minor axis of the manipulability ellipsoid. Then, a look-up table can be generated based on the minor axis lengths and the magnitude of patient's applied force; thus, along the desired trajectory, if one knows the length of the minor axis of the manipulability ellipsoid, the magnitude of the desired force can be evaluated from the look-up table:
$$\hat{F}_d = \mathscr{F}(\mathbb{S}_2) \geq 0 \tag{5.46}$$
where $\mathscr{F}$ is the look-up table function.

3. Desired force weighting (N_d): in the previous steps, the direction and magnitude of the desired force are determined. In this step, the need for assistance or resistance and their corresponding amounts are evaluated using a weighting factor. This factor is the cosine of the angle between the end-effector velocity error and the direction of the minor axis of the manipulability ellipse (see Fig. 5.12), that is:
$$N_d = \cos(\theta_v) = \frac{\widetilde{\dot{\boldsymbol{\rho}}} \cdot \mathbb{U}_2}{\|\widetilde{\dot{\boldsymbol{\rho}}}\|} \tag{5.47}$$

Finally, the desired force will have the following equation:

$$\mathbf{F}_d = N_d \hat{F}_d \mathbf{U}_2 + \mathrm{e}^{-\frac{\hat{F}_d^2}{c}} \mathbf{F}_{ext} \tag{5.48}$$

where the second term (i.e., Gaussian function) is considered to address the large position errors in the dysfunctional patient's rehabilitation (check the relatively large position errors for the impaired hand experiments in Figures 5.6, 5.10, and 5.11). For weaker patients (where $\hat{F}_d$ is smaller), the first term gets smaller while the second term increases. Thus,

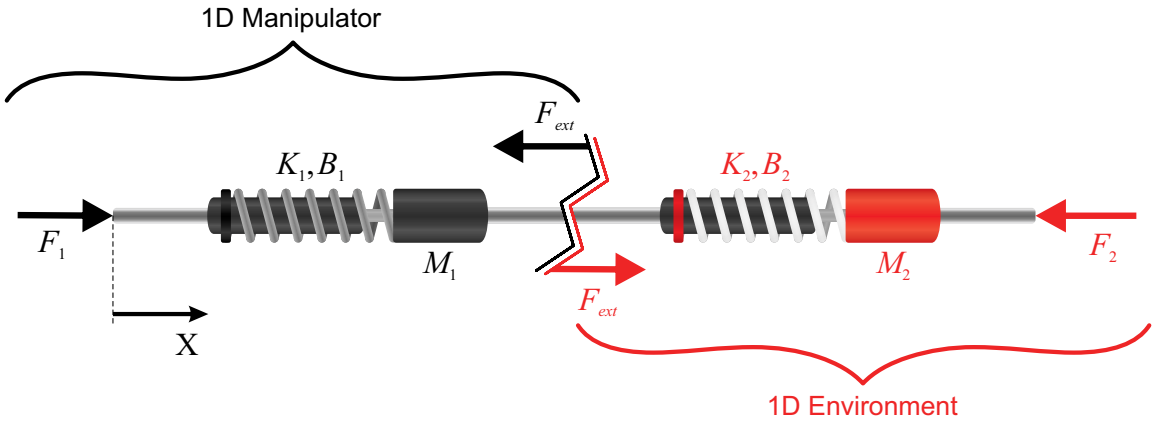


Figure 5.13: 1D plant: 1D manipulator with an actuating force F_1 interacts with a 1D environment with externally applied force F_2. M, B, and K are the mass, damping and stiffness of the each mass-spring-damper system.

the HFIC or SOIC will change to an acceleration control problem once $\hat{F}_d = 0$ [118, 156]. In (5.48), coefficient $c > 0$ is adjusted by the therapist based on the performance of the patient.

If in a robot control structure any kinematic or dynamic information from the interacting subject (human) is used, then the robot is being controlled by an HRI controller. Since the attractive force field control uses kinematic and dynamic information from the subject, it can be considered a type of HRI control; this has also been mentioned in Table 5.1.

Since this method is a type of high-level controller, it needs clinical settings for experiments. Evaluative results for this method are part of future work for this project.

5.3.2 Adaptive optimal impedance control

Since an impedance control only permits offline changes to the robot's apparent mass, the objective of this subsection is to find a rule for the online change of the robot's apparent mass. Without loss of generality, we consider a 1D linear manipulator (i.e., a mass-spring-damper with an actuating force) that interacts with another mass-spring damper as an environment (see Fig. 5.13).

If we exclude the environment in the system dynamics by including the interaction force (F_{ext}), which is measured by a force sensor, the manipulator dynamics will have the

following form:

$$M_1\ddot{X} + B_1\dot{X} + K_1X = F_1 - F_{ext} \tag{5.49}$$

Now, assume that it is desired to optimally track the following reference impedance model:

$$M_{imp}\ddot{\widetilde{X}}_r + B_{imp}\dot{\widetilde{X}}_r + K_{imp}\widetilde{X}_r = -\widetilde{F}_{int} \tag{5.50}$$

where $\widetilde{X}_r = X_d - X_r$, in which X_r is the reference model's position, and X_d is the desired position. Apparent mass M_{imp} is unknown, while stiffness (K_{imp}) and damping (B_{imp}) coefficients can be evaluated from (5.11). $\widetilde{F}_{int} = F_d - F_{int}$ is an unknown interaction force that will be discussed later. This optimal control problem can be solved with an augmented system representation [220], which is used for optimal reference trajectory tracking.

Optimal trajectory tracking

Consider the state-space representation for the manipulator's error dynamics, which is similar to (5.15) in one dimension:

$$\dot{\widetilde{\mathbf{x}}} = \mathbf{A}\widetilde{\mathbf{x}} + \mathbf{B}\widetilde{u} \tag{5.51}$$

where $\mathbf{x} = [X, \dot{X}]^T$ and u are the system state vector and control input, respectively. u_d is equal to the mass-spring-damper dynamics at desired trajectory that is $u_d = M_1\ddot{X}_d + B_1\dot{X}_d + K_1X_d$. If we define the control input as:

$$u = F_1 - F_{int} \tag{5.52}$$

the objective is to design the optimal tracking control in such a way that the manipulator tracks the reference impedance model while it estimates M_{imp} by making F_{int} as close as possible to F_{ext}. The state-space representation for the reference impedance model is as follows:

$$\begin{aligned}\dot{\widetilde{\mathbf{x}}}_r &= \begin{bmatrix} 0 & 1 \\ -\frac{K_{imp}}{M_{imp}} & -\frac{B_{imp}}{M_{imp}} \end{bmatrix} \begin{bmatrix} \widetilde{X}_r \\ \dot{\widetilde{X}}_r \end{bmatrix} + \begin{bmatrix} 0 \\ -\frac{1}{M_{imp}} \end{bmatrix} \widetilde{F}_{int} \\ &= \mathbf{A}_{imp}\widetilde{\mathbf{x}}_r + \mathbf{B}_{imp}\widetilde{F}_{int} \end{aligned} \tag{5.53}$$

If we combine the two state-space equations (5.51) and (5.53), the following augmented system will result:

$$\begin{aligned}\dot{\bar{\mathbf{x}}} &= \begin{bmatrix} \mathbf{A} & \mathbf{0} \\ \mathbf{0} & \mathbf{A}_{imp} \end{bmatrix} \begin{bmatrix} \widetilde{\mathbf{x}} \\ \widetilde{\mathbf{x}}_r \end{bmatrix} + \begin{bmatrix} \mathbf{B} & \mathbf{0} \\ \mathbf{0} & \mathbf{B}_{imp} \end{bmatrix} \begin{bmatrix} \widetilde{u} \\ \widetilde{F}_{int} \end{bmatrix} \\ &= \bar{\mathbf{A}}\bar{\mathbf{x}} + \bar{\mathbf{B}}\bar{\mathbf{u}} \end{aligned} \tag{5.54}$$

Then the objective functional for this optimal tracking problem is:

$$\mathfrak{J} = \frac{1}{2}\int_0^\infty \left((\widetilde{\mathbf{x}} - \widetilde{\mathbf{x}}_r)^T \mathbb{Q}(\widetilde{\mathbf{x}} - \widetilde{\mathbf{x}}_r) + \bar{\mathbf{u}}^T \bar{\mathbb{R}} \bar{\mathbf{u}}\right) dt = \frac{1}{2}\bar{\mathbf{x}}^T \mathbf{P} \bar{\mathbf{x}} \tag{5.55}$$

The optimal policy for this objective functional is [220]:

$$\begin{cases} \bar{\mathbf{u}} = -\bar{\mathbb{K}}\bar{\mathbf{x}} \\ \bar{\mathbb{K}} = \mathbb{R}^{-1}\bar{\mathbf{B}}\mathbf{P} \end{cases} \tag{5.56}$$

where $\mathbf{P}$ satisfies the following Algebraic Riccati Equation (ARE):

$$\bar{\mathbf{A}}^T\mathbf{P} + \mathbf{P}\bar{\mathbf{A}} + \bar{\mathbf{Q}} - \mathbf{P}\bar{\mathbf{B}}\bar{\mathbf{R}}^{-1}\bar{\mathbf{B}}^T\mathbf{P} = \mathbf{0} \tag{5.57}$$

here, $\bar{\mathbf{Q}}$ is:

$$\bar{\mathbf{Q}} = \begin{bmatrix} \mathbb{Q} & -\mathbb{Q} \\ -\mathbb{Q} & \mathbb{Q} \end{bmatrix} \tag{5.58}$$

Note that in (5.55), energy-term gain $\bar{\mathbb{R}}$ is defined as follows:

$$\bar{\mathbb{R}} = \begin{bmatrix} \mathbb{R} & \mathbb{R} \\ \mathbb{R} & (1+c)\mathbb{R} \end{bmatrix} \tag{5.59}$$

where coefficient $c > 0$ is adjusted by the therapist. The energy term based on (5.52) will result in the following equation:

$$\begin{aligned} \bar{\mathbf{u}}^T\bar{\mathbb{R}}\bar{\mathbf{u}} &= \begin{bmatrix} u_d - F_1 + F_d - \widetilde{F}_{int} & \widetilde{F}_{int} \end{bmatrix} \begin{bmatrix} \mathbb{R} & \mathbb{R} \\ \mathbb{R} & (1+c)\mathbb{R} \end{bmatrix} \begin{bmatrix} u_d - F_1 + F_d - \widetilde{F}_{int} \\ \widetilde{F}_{int} \end{bmatrix} \\ &= \mathbb{R}(u_d - F_1 + F_d)^2 + c\mathbb{R}\widetilde{F}_{int}^2 \end{aligned} \tag{5.60}$$

Thus, this energy-term gain includes the quadratic form of the interaction force error inside the objective functional equation. Finally, the 1D manipulator should be controlled by the following input force:

$$F_1 = \begin{bmatrix} 1 & 1 \end{bmatrix} \bar{\mathbb{K}}\bar{\mathbf{x}} + u_d + F_d \tag{5.61}$$

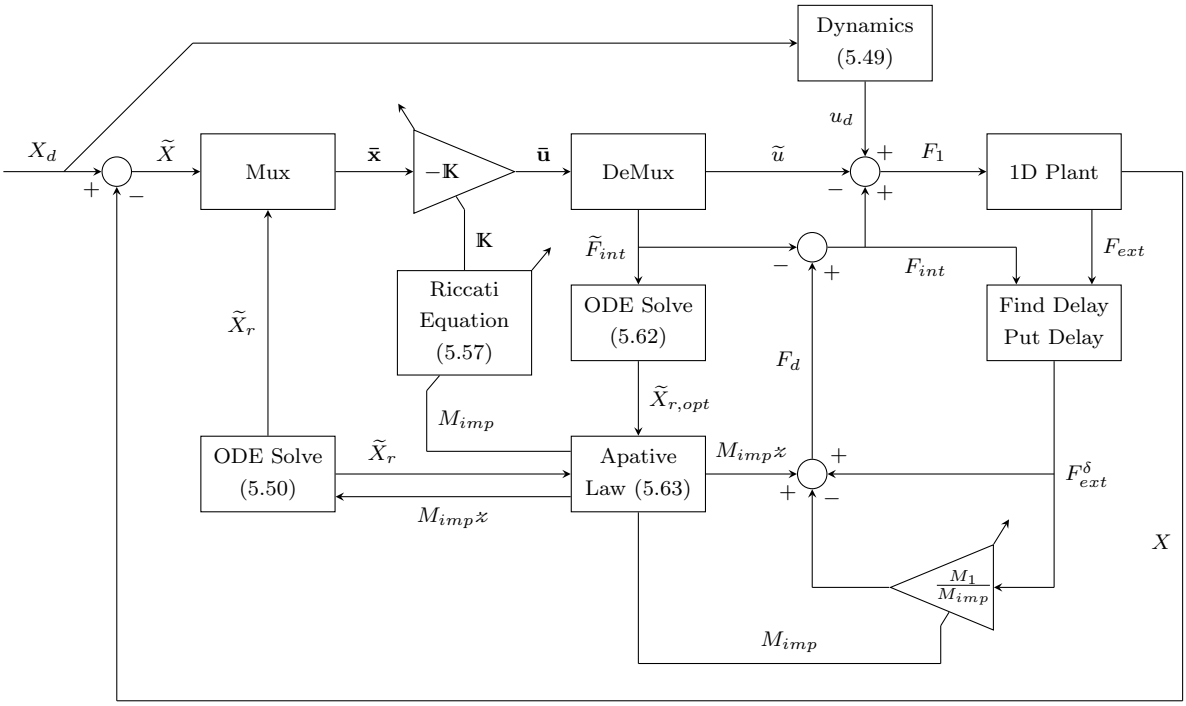


Figure 5.14: Adaptive optimal impedance controller scheme. Mux and DeMux stand for multiplexer and demultiplexer, respectively.

Adaptive control law

In (5.61), there are two unknowns including F_d and $\mathbf{x}_r$. Here, we try to find these unknowns using an adaptive control law. If the resulted optimal interaction force error is applied to an impedance model identical to (5.50), the optimal reference position error ($\widetilde{X}_{r,opt}$) can be evaluated by solving the following ODE:

$$M_{imp}\ddot{\widetilde{X}}_{r,opt} + B_{imp}\dot{\widetilde{X}}_{r,opt} + K_{imp}\widetilde{X}_{r,opt} = \begin{bmatrix} 0 & 1 \end{bmatrix} \mathbb{K}\bar{\mathbf{x}} \tag{5.62}$$

If we use the obtained optimal reference position error, M_{imp} can be updated by the following adaptive law:

$$\begin{cases} \dot{M}_{imp} = -\gamma\ddot{\varkappa}(\dot{\widetilde{X}}_r - \dot{\widetilde{X}}_{r,opt}) \\ \ddot{\varkappa} = \ddot{\widetilde{X}}_{r,opt} - \beta_1(\dot{\widetilde{X}}_r - \dot{\widetilde{X}}_{r,opt}) - \beta_0(\widetilde{X}_r - \widetilde{X}_{r,opt}) \end{cases} \tag{5.63}$$

where γ, β_1, and β_0 are positive coefficients, and $\widetilde{X}_r$ is obtained by solving (5.50) while providing $\widetilde{F}_{int} \triangleq M_{imp}\ddot{\widetilde{X}}$ as the input. Then, the desired input force F_d will be:

$$F_d = M_{imp}\ddot{\widetilde{X}} + \left(1 - \frac{M_1}{M_{imp}}\right) F_{ext}^{\delta} \tag{5.64}$$

where F_{ext}^{δ} is the δ seconds delayed interaction force which is recorded by the force sensor, and δ is updated by matching F_{ext}^{δ} with F_{int}. Finally, the adaptive optimal impedance controller scheme has the structure shown in Fig. 5.14

Simulation

To evaluate the proposed controller, a model similar to Fig. 5.13, in which $B_1 = 0$, $K_1 = 0$, $B_2 = 0$, $K_2 = 0$, and $M_1 = 1$ kg, is developed in MapleSim$^{\text{TM}}$. Three different values are assigned for the second mass (i.e., $M_2 = \{1, 10, 50\}$ kg), and the externally applied force F_2 is set to have the following function:

$$F_2 = A\sin(\Omega t)(u_{step}(t - t_0) + 1) + d_{N(\mu,\sigma)} \tag{5.65}$$

where $A = \{10, 100\}$ N and $\Omega = \{1, 20\}$ rad/s are input amplitudes and frequencies, and u_{step} is the unit step control input. $t_0 = 200$ s, and $d_{N(\mu,\sigma)}$ is the Gaussian random noise with the mean $\mu = 0$ and variance $\sigma^2 = A^2/100$ N. The goal is to control the position of the first mass (X) such that it tracks the desired trajectory $X_d = t/100$ with optimal force and position error for different values of second mass, amplitude and frequency. The simulation time and step size are set to 100 minutes and 1 ms, respectively. The ARE is solved numerically every 0.1 s.

Results and discussion

Results of the simulations are plotted in figures 5.15, 5.16, and 5.17. In each of these figures, first and second rows correspond to $A = 10$ N and $A = 100$ N, respectively. Furthermore, first and second columns correspond to $\Omega = 1$ rad/s and $\Omega = 20$ rad/s, respectively. Fig. 5.15 shows that at higher input force frequencies, the controller performance is independent of the adaption law and the robot's apparent mass does not change. However, at lower input force frequencies, adaption law plays an important role, and the robot's apparent mass has changed. When the second mass increases, the time-derivative of the robot's apparent mass change decreases, since higher second mass values absorbs most of the applied force's energy. In other words, once the second mass decreases, the

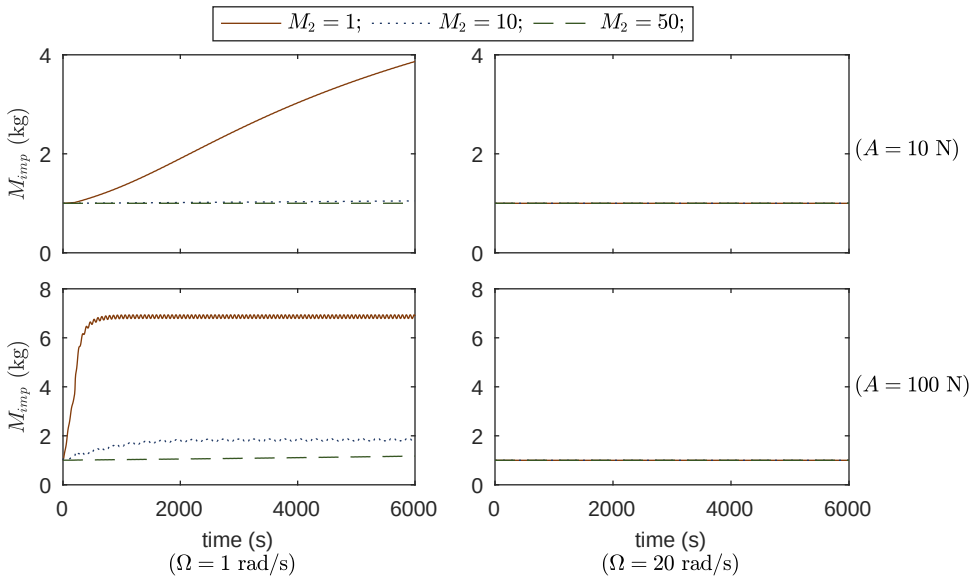


Figure 5.15: Robot's apparent mass during simulation with different input amplitudes, frequencies, and second masses.

adaptive controller tries to find higher apparent masses for the robot to absorb the applied force's energy. Thus, the apparent mass for the lower values of the second mass is higher. Note that at input force with $A = 10$ N, since the amplitude and the frequency were low, the apparent mass for the unit second mass has not converged in 6000 s of the simulation, but the pattern of the Fig. 5.16 indicates that it will eventually converge.

In Fig.5.16, the upper envelope of the velocity error is plotted to capture the controller performance regarding the decrease in velocity error. There is a big jump in all plots (see figures 5.16 and 5.17), which is caused by the unit step function in (5.65). High-frequency spikes in these figures are resulted because of the Gaussian random noise input in (5.65). For most of the path, while the apparent mass is converging to a constant value, the upper envelope of the velocity error has decreased or kept constant in all plots. This shows the success of the controller in tracking the desired trajectory while the apparent mass is being updated. Fig.5.16 shows the upper envelope of the interaction force error to check if the controller has converged to an optimal interaction force. In all of the plots, as the apparent mass is converging to a constant value, the interaction force has decreased or kept constant, and this proves that the controller tries to find lower interaction forces for different conditions of the input force.

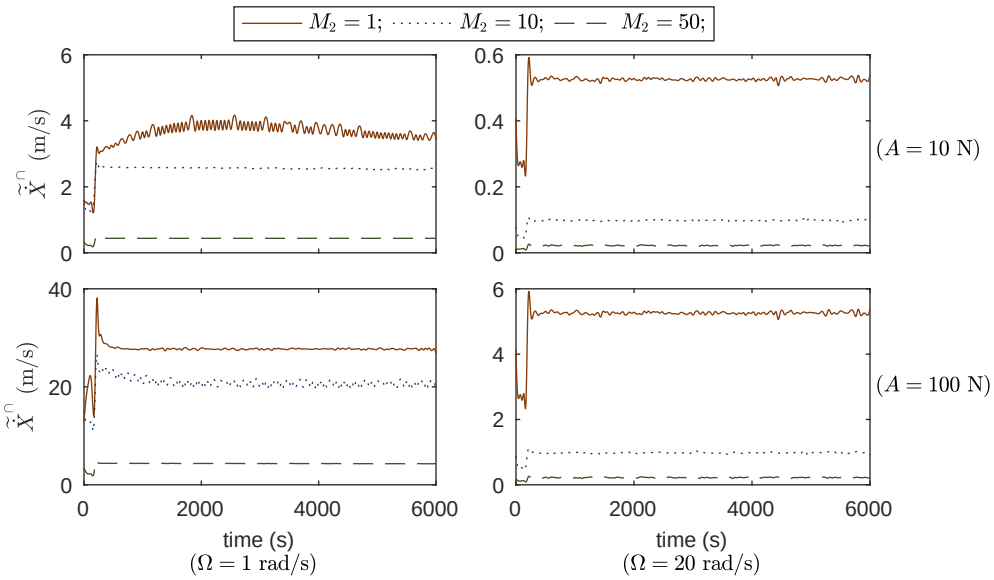


Figure 5.16: Upper envelope of the velocity error during simulation with different input amplitudes, frequencies, and second masses. Superscript $\cap$ indicates the upper envelope of a variable.

Conclusion

Here, we showed that it is possible to implement the impedance control optimally with an adaptive law which updates the robot's apparent mass. Results of the simulations indicate that at higher input force frequencies or higher environmental mass, the adaption law does not change the apparent mass a lot. The method is designed for a 1D interaction models; however, it can be extended to 2D interaction model. For example, if the state-space equation in (5.15) is redefined in the operational space using (5.28), then the same controller structure as in Fig. 5.14 can be used to implement our proposed method. In this case, the only problem is the large dimension of the ARE, which can be solved using policy iteration presented by Vrabie et al. [304] for real-time implementation.

5.4 Concluding remarks

In this chapter, we developed controllers for the rehabilitation robot according to the impedance-based assistance. The goal was to achieve a "reliable" and real-time implementable controller. In our definition, a "reliable" controller is capable of handling variable

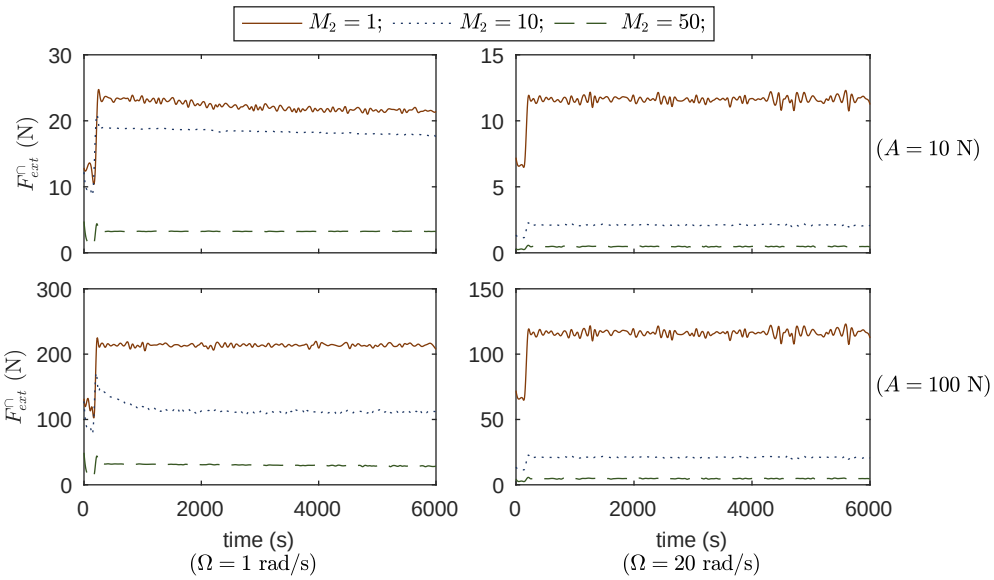


Figure 5.17: Upper envelope of the interaction force during simulation with different input amplitudes, frequencies, and second masses.

exercises and admittance interactions. The controller should reduce therapist intervention and improve the quality of the rehabilitation. Overall, two types of impedance-based controllers were developed, and their advancements were discussed.

In Section 5.1, to improve impedance control performance in variable admittance environments, the compliance is controlled by an outer loop force control (i.e., HFIC). Thus, in a subspace (tangent to the trajectory), the proposed low-level controller (HFIC) was trying to ensure force tracking with bounded position error. In an orthogonal subspace (normal to the trajectory), the position tracking with bounded force error was achieved. The force tracking needs a knowledge of the interacting system (human body) to define the desired force that the subject can safely tolerate. Thus, although HFIC was successful in assisting the healthy and less strong subjects, it was not recommended to be used in interaction with weak subjects. This disadvantage might be resolved by introducing an outer-loop attractive force field controller for HFIC, that was discussed in Section 5.3.1.

HFIC only controls the compliance (i.e., interaction force) term, and the impedance gains are not optimal. Thus, it results in a superficial variable impedance model and requires careful selection of the gains. In Section 5.2, a general method (i.e., OIC) that optimally adjusts impedance gains for variable robot configurations was developed and tuned through simulating the human-robot system. Here, an integrated human-robot dynamic

system was used to fine-tune the controller gains. This method was advantageous for efficient tuning of the robot controllers in experiments. A good qualitative agreement between experimental and simulation results verified the effectiveness of this method compared to the conventional SOIC.

For a linear robot model, the OIC assumes an apparent mass for the robot equal to its mass matrix, while the SOIC permits offline changes to the robot's apparent mass. For optimal results, the robot's apparent mass should vary online as a function of the input frequencies of the system. However, neither the OIC nor SOIC offer such updates. Moreover, in regards to experiments, the unknown dynamics of the robot presents a challenging issue, independent of the controller. In Section 5.3.2, a rule (adaptive optimal impedance control) for the online change of the robot's apparent mass was developed. The method was verified by simulation on a simple 1D system. Results of the simulations showed that at higher input force frequencies or higher environmental mass, the adaption law does not change the apparent mass a lot. The method was designed for a 1D interaction model; however, it can be extended to 2D interaction model.

Chapter 6

Rehabilitation Robot Control: Human-Robot Interaction Control

In the previous chapter, we developed and tuned advanced black-box controllers with model-in-loop simulations. For safety in robotic rehabilitation, other than the black-box controllers, we focus on HRI control structures (see Fig. 5.1). Among HRI controllers, model-based control is more attractive, since:

1. It can provide information regarding the human-robot dynamic interaction.
2. It can be used for dynamic parameter identification of the human body,
3. In contrast to black/gray-box controllers, a physics-based model can leverage the inherent dynamics of the system and facilitate implementation of special control techniques [4, 266], that can optimize a specific performance criterion while meeting stringent system constraints.

Studies on model-based control of rehabilitation robots are limited in the literature. Ding et al. [69] used a musculoskeletal upper extremity model (without including muscle dynamics) to implement a model-based assistive controller for an upper extremity rehabilitation exoskeleton. In this chapter, we present model-based controllers for the upper extremity rehabilitation robot using our developed musculoskeletal models.

6.1 Nonlinear model predictive control with an external 3D musculoskeletal model

In this section, the robot's computer model is integrated with the 3D upper extremity musculoskeletal model (MA3D-II). The goal is to control the robot optimally based on the estimates from the FSO of the external musculoskeletal model (i.e., human). Here, we assume that the dynamic parameters of the internal musculoskeletal model in HRI are identical to the ones that the external musculoskeletal model has.

For the optimal control, since the robot dynamics is nonlinear, an NMPC is utilized to control the HRI system. The proposed controller regulates the weights of the objective functional based on the feedback from FSO to assist or resist the patient's movements.

6.1.1 Controller structure

Consider the robot dynamics in (4.22). The goal is to control the robot by minimizing the instantaneous cost functional:

$$\mathfrak{J} = \int_{t_0}^{t_0+t_{ph}} \left(\widetilde{\mathbf{x}}_{\boldsymbol{\rho}}^T \mathbb{Q} \widetilde{\mathbf{x}}_{\boldsymbol{\rho}}^T + \mathbf{u}^T \mathbb{R} \mathbf{u} \right) dt \tag{6.1}$$

where $\mathbb{R}$ and $\mathbb{Q}$ are positive-definite diagonal weighting matrices, and t_0 and t_{ph} are the current time step and prediction horizon, respectively. $\mathbf{u}$ is the control input, which contains robot motor torque inputs. $\widetilde{\mathbf{x}}_{\boldsymbol{\rho}}$ is the state vector error in the global coordinates which is defined in (5.22). In (5.25), this error vector is in terms of the state vector error in the joint space.

In the human body, it is hypothesized that a control structure similar to the NMPC with receding horizon might be used by the CNS to control the arm motion. FSO is a special case of NMPC, which optimizes the current time step while considering the future time step [180, 254]. Thus, in this section, FSO with a cost functional defined in (4.1) is used to estimate the optimal muscle activations of the subject interacting with the robot, while tracking the desired path.

The robot should cope with the external musculoskeletal model's performance. This is possible if the tracking gain ($\mathbb{Q}$) in the robot's cost functional (6.1) is regulated by the muscular activities of the subject. In other words, it is desired to redefine the cost functional as:

$$\mathfrak{J} = \int_{t_0}^{t_0+t_{ph}} \left(\mathscr{f}(a_{1..n_m}) \widetilde{\mathbf{x}}_{\boldsymbol{\rho}}^T \mathbb{Q} \widetilde{\mathbf{x}}_{\boldsymbol{\rho}}^T + \mathbf{u}^T \mathbb{R} \mathbf{u} \right) dt \tag{6.2}$$

where $\mathscr{f}$ is a function of the subject's muscular activations which are obtained from FSO. If the subject shows higher muscular activities, the robot should decrease its assistance (i.e., $\mathscr{f}$ should decrease), and vice versa. Thus, we update $\mathscr{f}$ using the following equation:

$$\mathscr{f}(a_{1..n_m}) = 1 - c_1 \tanh\left(\frac{\sqrt{\sum_{m=1}^{n_m} a_m^2}}{c_2}\right) \tag{6.3}$$

where $0 \leq c_1 \leq 1$ is a coefficient such that $(1 - c_1)$ specifies the amount of the robot's assistance. $0 < c_2 < 1$ specifies the less-active activation level.

The NMPC optimizes the cost functional in (6.2) subject to the robot dynamics (4.22) and following inequality constraint:

$$\left\{ \begin{array}{c} \mathbf{x_q}_{min} \\ \mathbf{u}_{min} \end{array} \right\} \leq \left\{ \begin{array}{c} \mathbf{x_q}(t) \\ \mathbf{u}(t) \end{array} \right\} \leq \left\{ \begin{array}{c} \mathbf{x_q}_{max} \\ \mathbf{u}_{max} \end{array} \right\} \tag{6.4}$$

Finally, the controller schematic in a model-in-loop simulation is provided in Fig. 6.1. The human-robot rehabilitation system, modeled in MapleSim$^{\text{TM}}$, receives two sets of inputs; the robot motors are driven by the NMPC output for joint torques ($u_{1,2}(t)$), and muscle activations of the musculoskeletal model are produced by the FSO block. The FSO block uses the external musculoskeletal model to solve the muscle force sharing for tracking a desired trajectory. In a previous time ($t = t_{0-}$), the robot's end-effector force and position are sent to the FSO block, which evaluates the current ($t = t_0$) muscle activations and feeds into the human-robot rehabilitation system and the controller (i.e., the NMPC block). In the current time step, the controller also receives the joint angles and end-effector force. In the NMPC block, the cost functional is (6.2) which is subject to the robot dynamics (4.22) and inequality constraint in (6.4). The controller performance is evaluated by comparing angular position inputs/outputs (dotted lines in Fig. 6.1). The FSO performance is assessed by comparing the end-effector position inputs/outputs (dashed lines in Fig. 6.1).

6.1.2 Simulations

In this study, the proposed NMPC problem is solved by the GPOPS-II optimal control package [211]. GPOPS-II uses orthogonal collocation, which is a direct optimization method. In this method, both state and control input variables are approximated by a series of polynomials and fed to an NLP problem. The IPOPT is used to solve this NLP problem.

Two modes of simulations for the assessment of the proposed controller on the developed human-robot interaction model are performed.

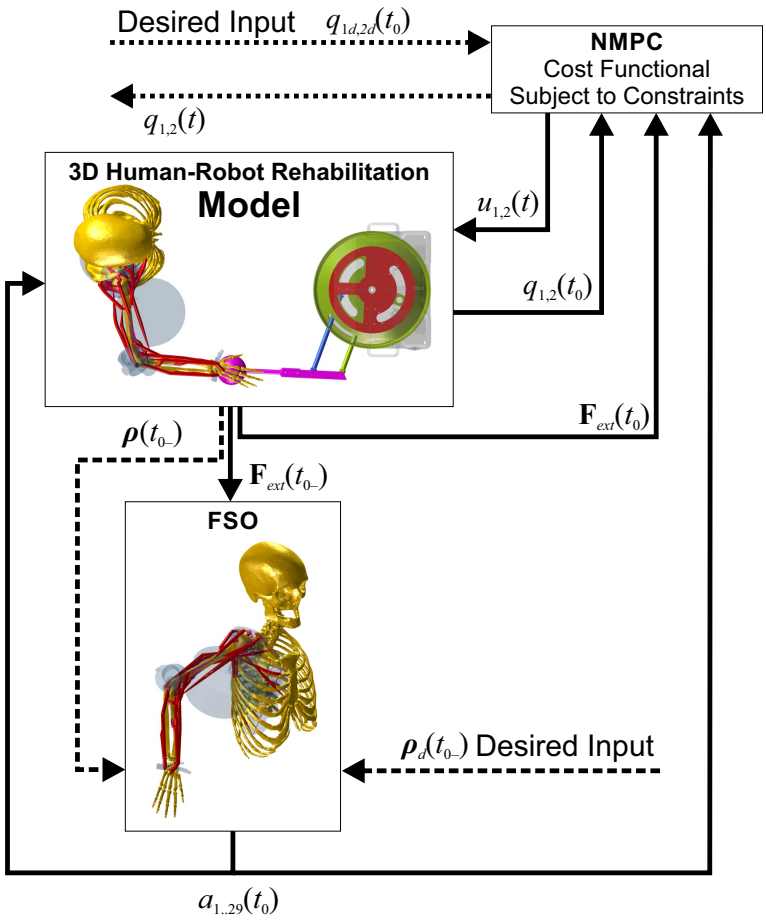


Figure 6.1: Controller structure of the 3D human-robot rehabilitation system for model-in-loop testing.

Mode-I: Healthy human interaction mode. In this mode, the objective functional of the FSO has higher tracking gains to simulate the robot interaction with a healthy human subject. The controller, in this mode, should generate a lower amount of torque for the accomplishment of the reaching task.

Mode-II: Weak patient interaction mode (i.e., assistive mode). In this mode, the objective functional of the FSO has lower tracking gains to simulate the robot interaction with a weak human subject. The controller, in this mode, should produce a higher amount of torque for the accomplishment of the reaching task.

The desired trajectory for the simulations is selected based on the coordination of arm

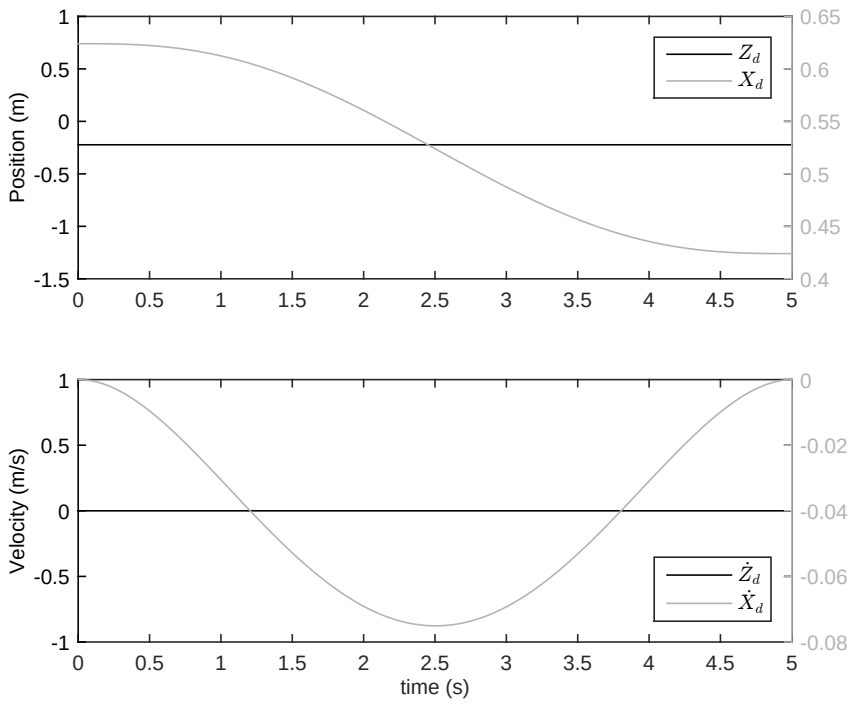


Figure 6.2: Desired trajectory for point-to-point reaching movement (Path2).

movements (Path2). This trajectory ($\boldsymbol{\rho}_d = [Z_d(t), X_d(t)]^T$, see Fig. 6.2) is a straight line with a bell-shaped tangential speed profile inside the human-robot workspace and defined as:

$$\begin{cases} Z_d(t) = Z_0 \\ X_d(t) = X_0 + (15\tau - 6\tau^2 - 10)\Delta_X \tau^3 \end{cases} \tag{6.5}$$

The path length and simulation time are set to $\Delta_X = 20$ cm and $t_f = 5$ s, respectively (see Fig. 6.2). The simulation time step is 10 ms, and the prediction and control horizons for the NMPC are $t_{ph} = t_{ch} = 100$ ms. See Table A.16 for the assigned control parameters.

6.1.3 Results and discussion

Results of simulations are presented in Fig. 6.3. Magnitudes of motor torques for the healthy subject interaction (i.e., Mode-I) are lower than the weak subject interaction (i.e., Mode-II). In Mode-I, the FSO tries to minimize muscular activities while performing the

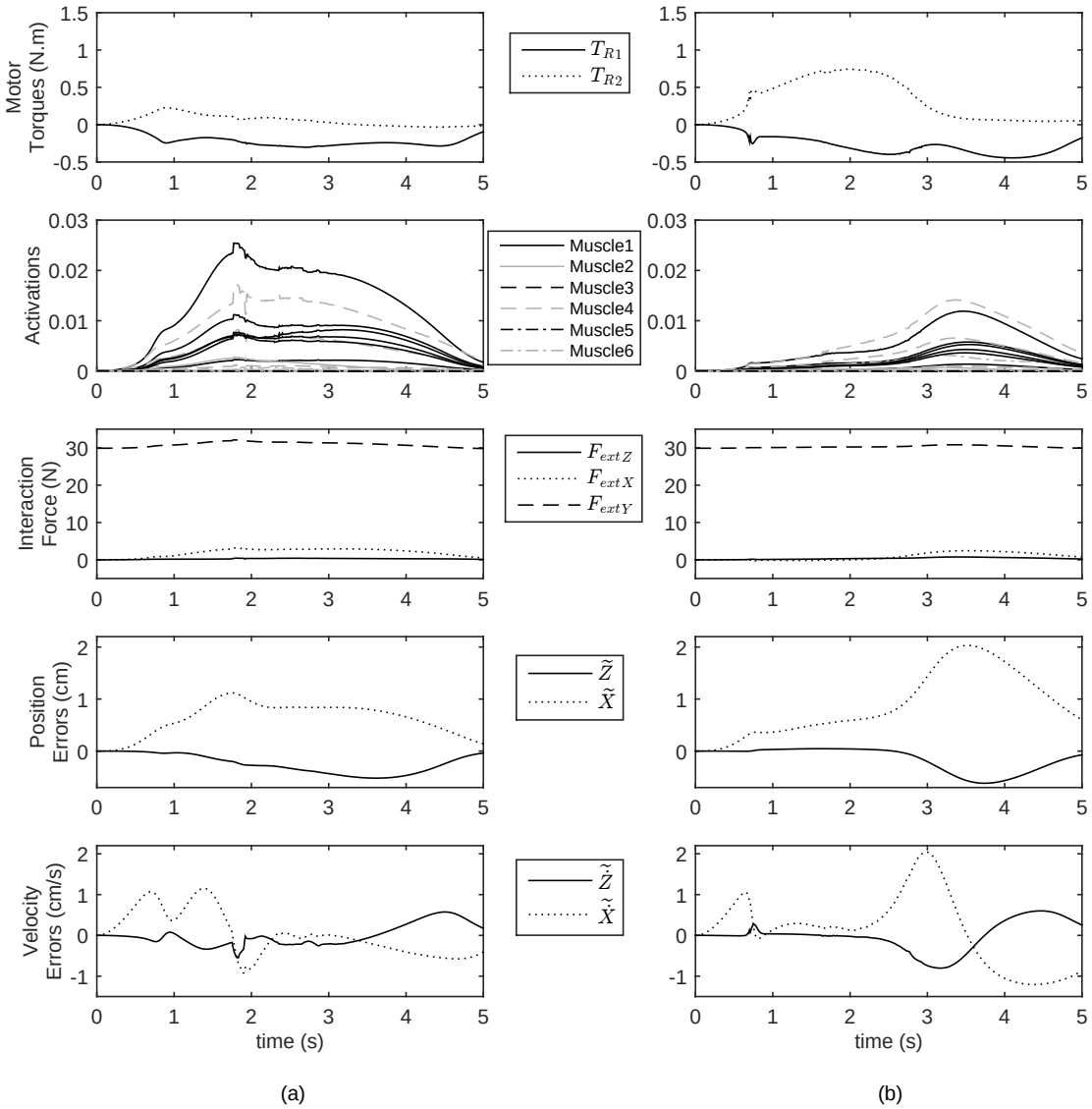


Figure 6.3: 3D Human-robot rehabilitation system simulation results using NMPC in two modes: (a) Mode-I, and (b) Mode-II. Muscles activations are reported in their synergistic groups as on page 85. Position and velocity errors are in the global coordinates (ZX).

reaching movement with a good accuracy, thereby increasing the muscle activations and decreasing $\mathscr{f}$. Thus, since the NMPC tries to minimize the motor torques while performing less accurate tracking, the motor torques decrease compared to Mode-II, in which the

FSO minimizes muscular activities while performing less accurate reaching movements. In Mode-II, the less accurate reaching of the weak subject is improved by the robot's assistance. Hence, the muscle activations are low, while the motor torques are high.

Since the tracking gain of the NMPC is updated by the muscular activities, the amount of the interaction force for both modes are in the same range. In Mode-I, the interaction force along Y-axis (normal to the horizontal plane) varies more than Mode-II, since the muscular activities of Mode-I are higher than the other mode. Overall, for both modes, the range of this normal force is about 30 N which is caused by the weight of the upper extremity.

In Mode-I, the robot allows the subject to finish the task independently with less assistance; thus, the large position and velocity errors are caused by the subject's accuracy in the reaching movement. However, in Mode-II, the robot tries to help the subject as much as possible; hence, more accurate tracking results.

6.1.4 Conclusion

Here, we tried to implement an NMPC with varying tracking gain on the robot. The tracking gain is updated based on the information received on the muscular activities of the interacting subject. If the dynamics of the interacting subject is identified, then the external FSO of the identified musculoskeletal model can estimate the muscular activities of the subject. The proposed controller performed well in interacting with healthy and weak subjects, in a way that it kept the interaction force in the same range while having less assistance for the healthy subject, and more assistance for the weak one.

The proposed control structure cannot be used in real-time simulations and experiments, since MA3D-II, which is used in the FSO, has 29 muscle activation inputs and 3 DOF that increase the computation cost of the FSO. In the next section, we will try to use a less complex upper extremity model (i.e., MA2D-II) within an NMPC structure to explore the possibility of real-time implementation.

6.2 Nonlinear model predictive control with an internal 2D human robot interaction model

The objective of this section is to predict muscle activities of a post-stroke patient during interaction with a rehabilitation robot, and provide quantitative therapy evaluations,

thereby improving the quality of therapy in terms of safety and motor function improvement. To achieve this goal, we used our developed HRI model (i.e., MA2D-II and robot), which is presented in 4.2.3. This model can be used with a controller to achieve this goal. For example, it can be used with gray-box controllers, in which the EMG signals can estimate the muscle activation levels of the patient, or be integrated with AI and any other type of controller that can predict muscle activations.

Recent progress in the development of the NMPC motivates us to use the developed non-linear human-robot interaction model to control the rehabilitation robot. In our research, the HRI model is confined within an NMPC of the stroke rehabilitation robot. The proposed controller uses the musculoskeletal model of the upper extremity to predict the movements and muscle activations [183], thereby providing optimal assistance to the patient.

6.2.1 Controller structure

In the human body, the arm motion is controlled by the CNS using a combination of feed-forward and feedback control commands [183]. The feed-forward commands are estimated using an internal representation of the arm, and the feedback control is a set of corrective commands resulting from sensory organs in the arm. This control structure is analogous to the NMPC with receding horizon. The NMPC uses: 1) a forward dynamics (feed-forward) model of the system to predict optimal movements, and 2) feedback information for error correction. During a prediction horizon (t_{ph}), optimal movements are determined by minimizing a cost functional ($\mathfrak{J}$) subject to the system dynamics (i.e., (4.28)) and some constraints (e.g., motor torque limits).

In our human-robot rehabilitation system, the goal is to optimally control the robot while estimating the human body muscular activities. Thus, the system can be controlled by an NMPC with feed-forward predictions for the optimal movements of the human body and robot, and feedback commands for prediction improvement. To minimize the robot energy consumption, the human's physiological effort, and end-effector tracking error, our proposed cost functional is the same as (6.1), in which $\mathbb{R}$ and $\mathbb{Q}$ weights are chosen such that the objective functional results in the allowable error associated with the state or effort. Thus, the end-effector compliance can be controlled by adjusting the $\mathbb{Q}$. $\mathbf{u}$ is the control input, which contains robot motor torque inputs and predicts muscle activations.

Considering the human-robot dynamics (4.28), interaction force is assumed to be an internal force. Nevertheless, if the human's muscle activation patterns are different from the human-robot model activations, the modeled interaction force ($\mathbf{F}_{int}$) will not be equal

to the measured values by the force sensor ($\mathbf{F}_{ext}$). Consequently, an internal force constraint is introduced to the system dynamics:

$$\boldsymbol{\psi}\left(\mathbf{x}_{\mathbf{q}}(t), \mathbf{u}(t), t\right) = \mathbf{F}_{int} - \mathbf{F}_{ext} = \mathbf{0} \tag{6.6}$$

where considering (4.24), the modeled interaction force can be written as:

$$\mathbf{F}_{int} = \mathbf{J}_R{}^{-T}\left(\mathbf{T}_R - \boldsymbol{\Gamma}_R\right) \tag{6.7}$$

Then, the human-robot inequality constraint is:

$$\left\{ \begin{array}{c} \mathbf{x}_{\mathbf{q}min} \\ \mathbf{u}_{min} \\ \mathbf{0} \end{array} \right\} \leq \left\{ \begin{array}{c} \mathbf{x}_{\mathbf{q}}(t) \\ \mathbf{u}(t) \\ \left|\boldsymbol{\psi}\left(\mathbf{x}_{\mathbf{q}}(t), \mathbf{u}(t), t\right)\right| \end{array} \right\} \leq \left\{ \begin{array}{c} \mathbf{x}_{\mathbf{q}max} \\ \mathbf{u}_{max} \\ \boldsymbol{\varepsilon} \end{array} \right\} \tag{6.8}$$

Since the dynamic properties of the human body may be different from the 2 DOF model's dynamic properties, the internal force constraint (6.6) is relaxed to reduce the effect of unwanted dynamics due to the mentioned difference. Thus, in (6.8), a relaxation parameter ($\boldsymbol{\varepsilon}$) is added to the internal force constraint.

The controller schematic is presented in Fig. 6.4. The human-robot rehabilitation system, modeled in MapleSim$^{\text{TM}}$, receives two sets of inputs: 1) the robot motors are driven by the NMPC output for joint torques, 2) muscle activations of the musculoskeletal model are either produced by the FSO, in which a musculoskeletal system behavior is modeled through static minimization of physiological cost functional, or measured EMG data in real patient experiments. The NMPC controller optimizes the cost functional (6.1) subject to the system dynamics (4.28) and constraints (6.8). Current (at time $t = t_0$) robot joint angles and end-effector force are sent to the controller. The model/controller performance is evaluated by comparing the muscle activation inputs/outputs (dashed lines in Fig. 6.4) and position inputs/outputs (dotted lines in Fig. 6.4).

6.2.2 Simulations

In this study, the proposed NMPC problem is solved by the same method defined in the previous section. Two modes of simulations for the assessment of the proposed controller on the developed human-robot interaction model are performed.

Mode-I: Healthy human interaction mode. In this mode, the controller weights are adjusted for the system shown in Fig. 6.4 by connecting the controller activation outputs

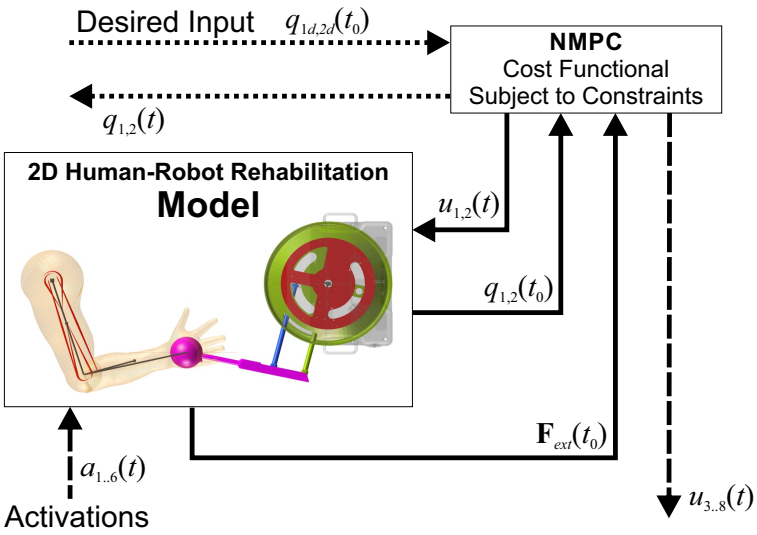


Figure 6.4: Controller structure of the 2D human-robot rehabilitation system for model-in-loop testing.

to the activation inputs of the human-robot rehabilitation system (i.e., dashed lines are connected to each other). The adjustment is done by trial and error while checking the controller performance inside the desired criteria (less interaction force with good tracking). This mode simulates the robot interaction with a healthy human subject.

Mode-II: Impaired patient interaction mode. In this mode, the controller performance is tested by driving the human-robot rehabilitation system with zero muscular activations. In this mode, a totally dysfunctional patient is interacting with the robot. If the activations from the controller are at the same level as the input activations (i.e., zero activations), the controller will be successful in estimating the human's behavior while interacting with the robot.

The desired trajectory, simulation time step, and control and prediction horizons are the same to the previous section. See Table A.17 for the assigned control parameters.

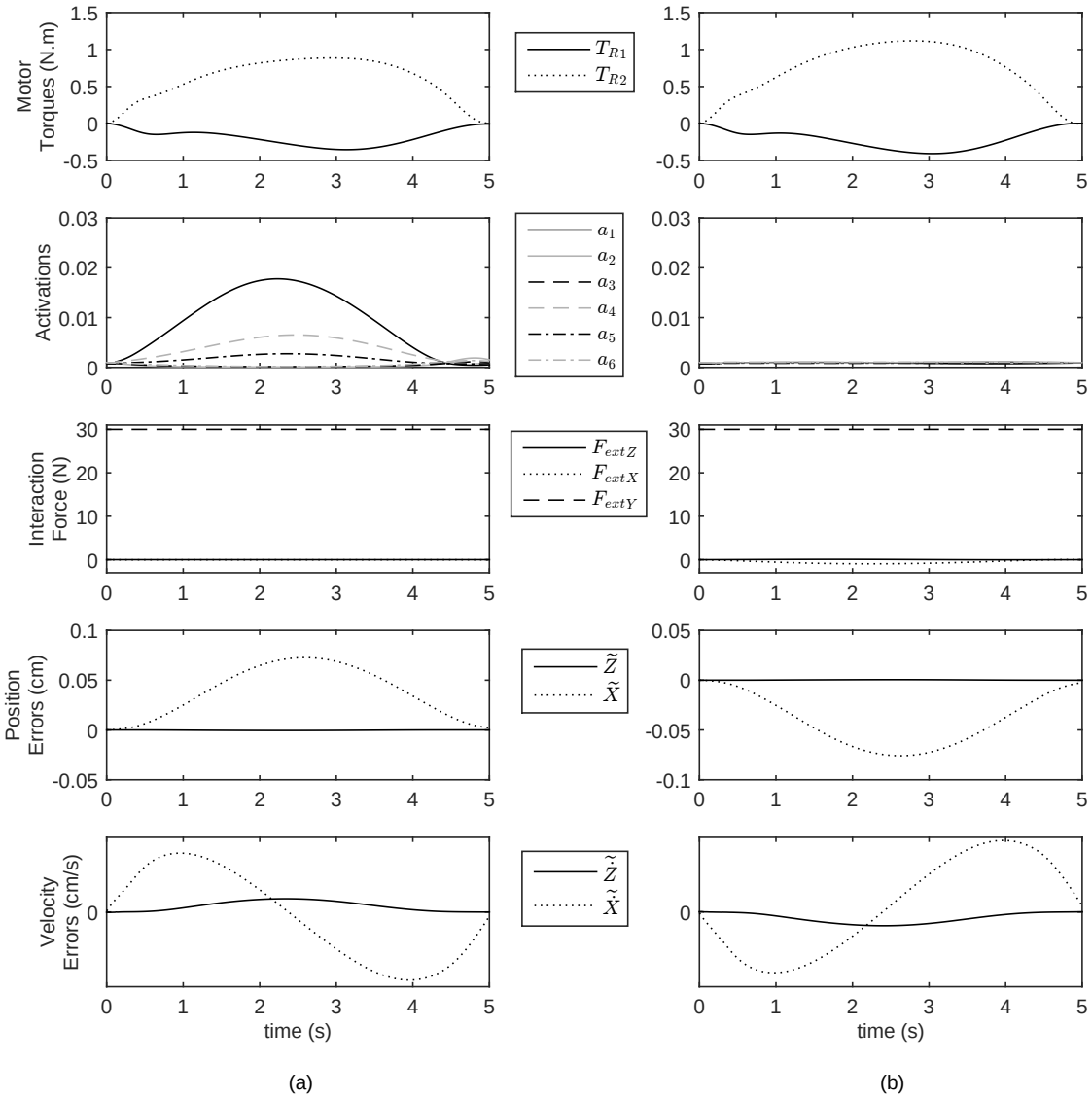


Figure 6.5: 2D Human-robot rehabilitation system simulation results using NMPC in two modes: (a) Activations of the human-robot system are provided by the NMPC output (Mode-I), and (b) Activations of the human-robot system are zero to simulate a dysfunctional patient interaction with the robot (Mode-II). Position and velocity errors are in the global coordinates (ZX).

6.2.3 Results and discussion

Results of simulations are presented in Fig. 6.5. Magnitudes of motor torques for healthy human interaction mode (Mode-I) are lower than the dysfunctional patient interaction mode (Mode-II). In Mode-I, the healthy subject tries to minimize his/her muscular activities while performing the task; hence, this reduces robot motor torques compared to Mode-II. In the meantime, the robot tries to help the person while minimizing the robot energy consumption, the subject's muscular activities and the end-effector tracking error. In an ideal case, where there is no relaxation of the internal force constraint (see equation (6.6)), the interaction force should be zero. However, relaxing the constraint (see equation (6.8)) results in a small amount of interaction force along Z and X directions (see Fig. 6.5.a).

In Mode-II, the dysfunctional patient cannot apply any forces (has zero muscular activities). The NMPC with the previous objective tries to find the best motor torques, predict the dysfunctional subject's muscle activations while tracking the desired trajectory. Since the robot should overcome the patient's dynamics while reducing tracking error, the interaction force and motor torques increase compared to Mode-I (see Fig. 6.5.b).

In Mode-II, the maximum amount of predicted muscle activations is about 0.003, which is very small and may be caused by integration errors, round-off calculations, and relaxation of the internal force constraint. This shows that the NMPC has predicted the dysfunctional patient's muscular activities with a reliable accuracy (the RMSE is 9.78e−4). In Fig. 6.5, the same amount of position and velocity errors for both modes indicates that the tracking term in the cost functional is dominant; thus, the NMPC tries to keep the subject's hand on the trajectory as much as possible. Position and velocity errors in the Z direction is less than the X direction since in the NMPC cost functional, the robot X direction is more compliant than the Z direction.

6.2.4 Conclusion

In this section, the HRI model is integrated with an NMPC controller for a human-robot rehabilitation system. This HRI model has 2 DOF, and its dynamics has a minimal set of equations (4.28). This model is used in our proposed NMPC structure (see Fig. 6.4) to evaluate the controller performance in two modes: healthy subject, and dysfunctional patient interaction with the robot while performing a point-to-point reaching task in a smooth movement path for the hand. The controller can successfully predict the muscular activations for the dysfunctional patient while providing the same therapy (tracking accuracy) to the patient.

Our proposed NMPC is implemented using GPOPS-II because of its variable-order adaptive collocation method for solving an optimal control problem. Although the controller performance is excellent in simulations, it is not possible to use it in our real-time experimental setup because: 1) GPOPS-II is not fast enough to run in real-time simulations, 2) the robot's data acquisition card (Q8) which is operable by Quanser's real-time control software driver (QUARC) does not support the GPOPS-II software.

6.3 Real-time implementation

The final goal of this project is to implement a real-time HRI controller for the rehabilitation robot. In previous sections, we discussed two types of NMPC frameworks (i.e., HRI with an external 3D musculoskeletal model, and HRI with an internal 2D musculoskeletal model). To implement a real-time model-based controller, one should consider the following modifications:

1. Simplifying the model: this is accomplished throughout Section 6.2. First, the 2D (low-fidelity) model is used in the HRI structure. Next, a minimal set of equations are derived for the HRI model.
2. Modifying the NMPC structure: which is done by minimizing the number of constraints.
3. Implementing NMPC using a real-time optimal control method

In this section, we implement the last two items, and present the results of the experiments.

6.3.1 Reducing the number of constraints

If we reconsider the constraint equations (6.8), the following modification can be done to reduce the computation cost:

$$\begin{cases} (\mathbf{x_q} - \mathbf{x_{q}}_{min}) \circ (\mathbf{x_q} - \mathbf{x_{q}}_{max}) \leq \mathbf{0} \\ (\mathbf{u} - \mathbf{u}_{min}) \circ (\mathbf{u} - \mathbf{u}_{max}) \leq \mathbf{0} \\ \boldsymbol{\psi}\left(\mathbf{x_q}(t), \mathbf{u}(t), t\right)^T \boldsymbol{\psi}\left(\mathbf{x_q}(t), \mathbf{u}(t), t\right) = 0 \end{cases} \tag{6.9}$$

where $\circ$ is the Hadamard (element-wise) product. Using the above constraint equations, the number of constraints (14 inequality constraints) is halved (6 inequality and 1 equality

constraint). It is worth noting that, for real-time experiments, the internal force constraint (6.6) is not relaxed since we have assumed that the effect of unwanted dynamics is negligible.

6.3.2 Implementing a real-time nonlinear model predictive control method

For solving a nonlinear optimal control problem, there are two types of numerical approaches: direct and indirect methods. Direct methods use calculus of variations and solve a boundary value problem, while the indirect methods use optimization techniques and solve a finite-dimensional optimization problem [226]. The direct methods can be implemented by either control parameterization or state and control parameterization. Single and multiple shooting methods are used for control parameterization, and local and global collocation methods are implemented using NLP for state and control parameterization. On the other hand, indirect methods use analytical equations of the system to generate the Hamiltonian functional derivatives to solve the optimal control problem. Depending on the NMPC structure, one of the mentioned methods can be advantageous over the other one [226]. In previous sections, our proposed NMPC frameworks are solved using the GPOPS-II optimal control software package. This software uses direct global collocation method to solve a multi-phase optimal control problem. However, real-time implementation of the controller using this software was not successful. Thus, we researched indirect optimal control methods, and leverage a symbolic processing and optimized code generation for real-time computation.

Recently, by the use of Newton's approach for solving a root finding problem in a Generalized Minimal Residual (GMRES) method, a new direct method (so called Newton/GMRES) was introduced to solve optimal control problems efficiently. Studies have shown decent improvements in computation time [182, 203, 277–279, 307]. Here, we use the MPsee toolbox developed by Tajeddin [277] to automatically generate a real-time NMPC for running our experiments. The NMPC is solved by a single shooting method in which the discretized states and Lagrangian multipliers of the co-states are solved implicitly.

6.3.3 Experiments

To evaluate the performance of the controller experimentally during a rehabilitation procedure, a healthy subject performed Mode-II similar to the simulations in Subsection 6.1.2. To this end, the subject is asked to relax his/her upper extremity muscles and avoid any

contractions as much as possible. To reduce the effect of random/noisy movements, the experiment was performed in 20 trials. The desired trajectory is same as simulations (Path2 in Fig. 6.2). The sampling-time for the experiments is set to 2 ms, and the prediction and control horizons for the NMPC are $t_{ph} = t_{ch} = 10$ ms. See Table A.17 for the assigned control parameters.

The processor of the computer running the experiments is an Intel® Core™ i7-7700K CPU at 4.2 GHz (8 CPUs), and computer's memory is 32 GB.

6.3.4 Results and discussion

Average interaction forces along Z, X, and Y axes are 0.23 N, 1.13 N, and 4.02 N respectively (see Fig. 6.6). The amount of interaction force along the X axis shows the robot's assistance along the path. The small amount of interaction force along the Z axis indicates that the patient is not producing noticeable force normal to the path. This is in agreement with Mode-II (weak patient interaction) and the simulation results (see Fig. 6.7). Nevertheless, in comparison to the force along the Y axis in simulations (see Fig. 6.5), the lower amount of the interaction force along the Y axis (which is about 0.4 kg) implies that the patient is resisting the gravitational force. This might lead to:

1. higher activations of the extensor muscles at the shoulder joint for the first 4 seconds of the movement (see extensor activation in Fig. 6.6), and

2. activations of the shoulder flexor muscles for the last one second of simulation where the arm is extended in the forward direction and the extensor muscles are not able to counterbalance the upper extremity weight (see flexor activation in Fig. 6.6).

Since the patient is told to relax on the horizontal plane, the corresponding flexor muscles might co-activate to balance the forces on the horizontal plane (see flexor activation in Fig. 6.6).

Average of the position RMSE along Z and X axes are 0.5 mm and 5.1 mm, respectively (see Fig. 6.6). Average of the velocity RMSE along Z and X axes are 2.4 mm/s and 9.1 mm/s, respectively (see Fig. 6.6). These tracking errors are slightly higher than the simulation results, since the musculoskeletal model inside the NMPC structure is not a perfect representative of the subject. Furthermore, the PE is not included in the musculoskeletal model; thus, some of the muscular activatities are mispredicted due to the exclusion of the PE.

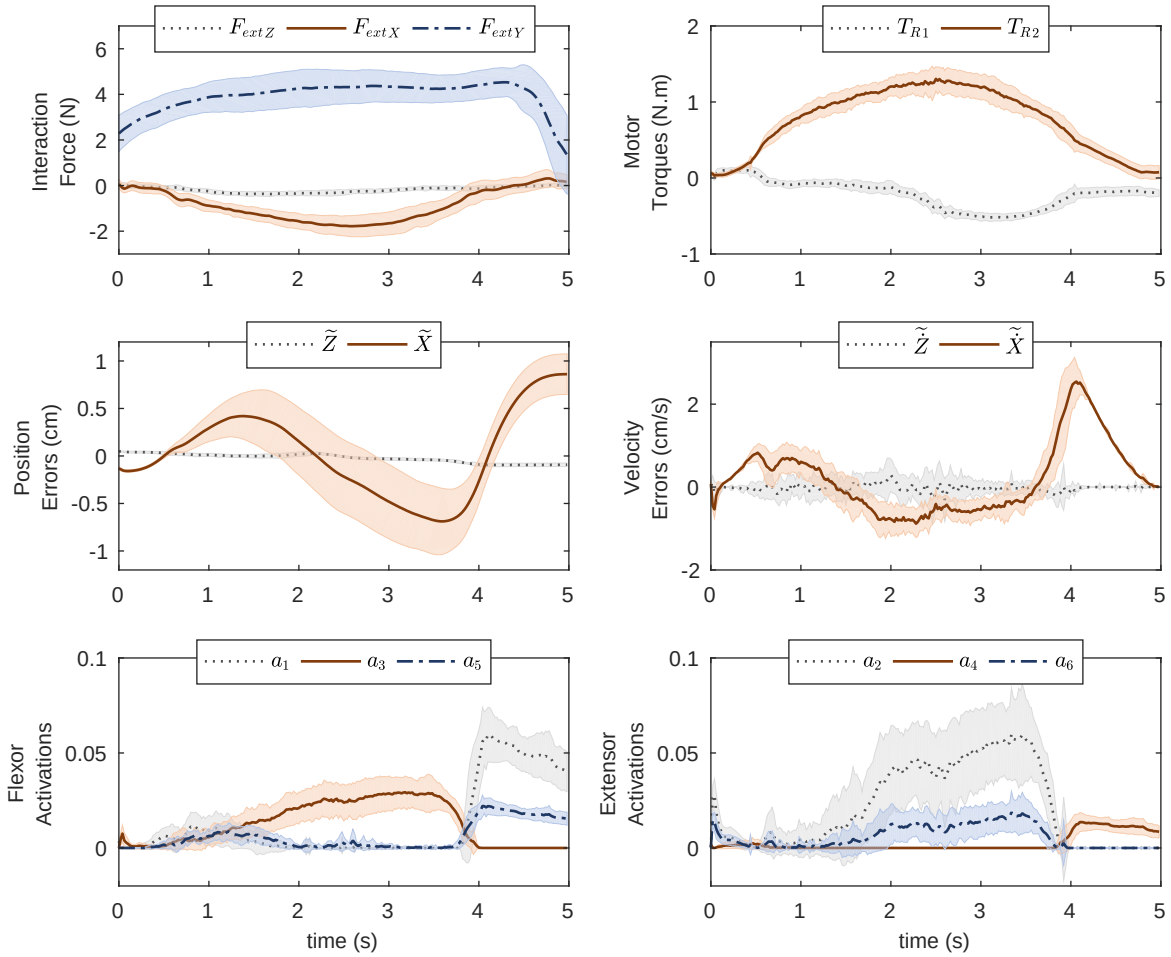


Figure 6.6: Experimental results of the rehabilitation system controlled by the real-time HRI control using internal 2D musculoskeletal model. Position and velocity errors are in the global coordinates (ZX). The shaded area denotes twice the standard deviation at each instance of experiment.

If one compares the trends of the motor torques, they match the simulation results (see Fig. 6.7). Finally, negative values of the muscle activations (at time = 4 s) are indicating constraint violations; this might be due to inaccurate musculoskeletal model parameters.

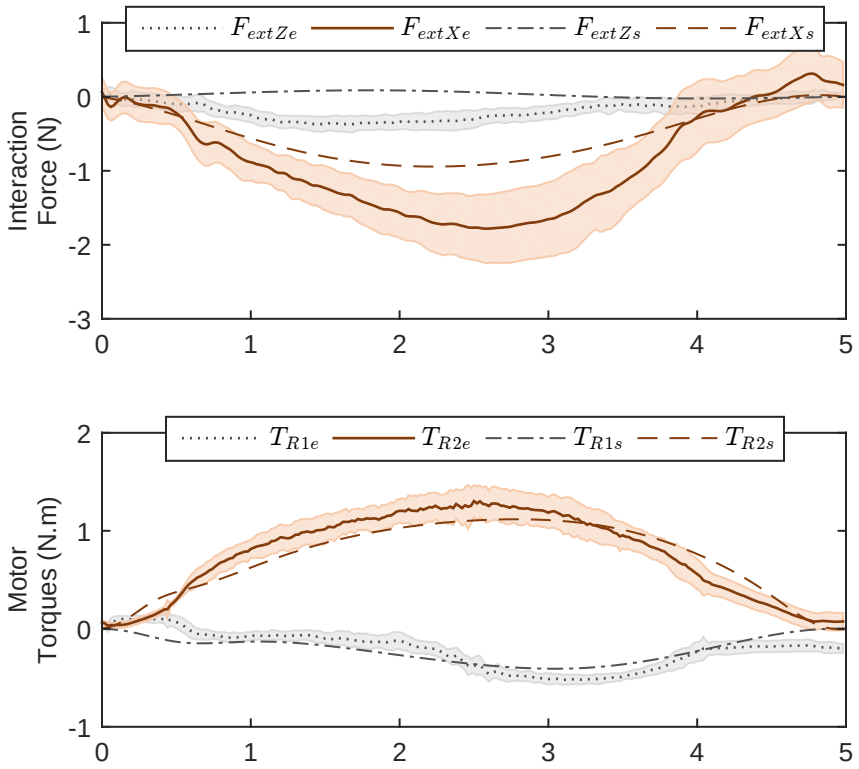


Figure 6.7: Experimental results of the rehabilitation system controlled by the real-time HRI control using internal 2D musculoskeletal model compared to the simulation results from Fig. 6.5. The shaded area denotes twice the standard deviation at each instance of experiment. The curves without shaded areas correspond to the simulation results. Subscripts s and e denote simulation and experiments, respectively.

6.3.5 Conclusion

In this section, we presented and discussed real-time implementation of the HRI control with the internal musculoskeletal model. The NMPC problem was solved using Newton/GMRES method with an efficient code generation and reduced-number of constraint equations. In experiments, the robot was able to accomplish the task with small interaction force and tracking error. However, the relaxed subject's muscular activities were higher than expected. This discrepancy may be due to inaccurate musculoskeletal model

parameters, exclusion of the PE, and active counterbalancing by the subject.

6.4 Concluding remarks

HRI control structures leverage some kinematic or dynamic information from the human subject to control the robot. Here, we introduced two types of model-based HRI control structures, which were implemented using NMPC with the robot model in model-in-loop simulations. In the first controller (NMPC with an external 3D musculoskeletal model), the tracking gain of the objective functional was varied by the results of an external FSO, which (in parallel) was optimizing muscular activations of an external 3D musculoskeletal model (high-fidelity model) for a specific task. This 3D musculoskeletal model should have similar dynamic properties as the internal model in the HRI that was used in a model-in-loop simulation. In other words, successful implementation of this controller depended on good DPI of the subject. Simulation results showed that the controller successfully meets the HRI design criteria.

Since the FSO of the high-fidelity model ran externally, it decreased the computation cost. Nonetheless, this controller was not implementable in real-time studies. In Section 6.2, a low-fidelity model was used within the structure of the NMPC to reduce the computation cost. In other words, instead of the 3D musculoskeletal model, a 2D musculoskeletal model was used to estimate the muscle activations. This model also needed a reliable DPI of the subject. Then, the FSO and the robot controllers were integrated and a single NMPC with the 2D HRI model was created to predict muscle activations along with robot motor torque inputs. The controller could predict the muscular activities. Its performance was excellent in simulations, but its computation cost was the same as the previous. Thus, it was not possible to use it in real-time studies.

Finally in Section 6.3, real-time implementation of the NMPC with the internal 2D musculoskeletal model was provided. In experiments, the robot was able to accomplish the task with small interaction force and tracking error. However, predicted higher than expected values for the relaxed subject's muscular activities. This discrepancy may be due to the inaccurate musculoskeletal model parameters, exclusion of the PE, and active counterbalancing by the subject. As a part of future work, more experiments with the use of EMG sensors will be done for physiological validation of the proposed method.

Chapter 7

Conclusion

Upper extremity rehabilitation robotic systems can be improved by new hardware and control scenarios. New mechanical design and control strategies are currently evaluated after device development and clinical tests. Design and control are iterative approaches, and thus, for better outcomes with multiple iterations, the possibility of verification without the construction of these robots becomes important. Since musculoskeletal models have been used successfully for the study of human movement, they are promising tools to interact with rehabilitation devices in simulations. In this study, we proposed the use of musculoskeletal models with modified activation dynamics interacting with rehabilitation robots to contribute to the controller design and development of a rehabilitation robot. In the current research, the goal was to utilize musculoskeletal models in real-time simulations for model-based design and control of rehabilitation robots.

To check the feasibility of this study, we used an upper extremity rehabilitation manipulandum [169], which was designed and developed by Quanser Consulting Inc. and the TRI, interacting with a musculoskeletal arm model. Since two systems (robot and human) are studied, their models were developed in chapters 3 (System Models: Rehabilitation Robot) and 4 (System Models: Human and Human-Robot Interaction). In these chapters, dynamic parameters of the musculoskeletal model for specific subject and robot model were identified using DPI. Next, in Chapter 5 (Rehabilitation Robot Control: Robot Control), two types of low-level control algorithms based on impedance-based assistance were developed and evaluated. Finally, in Chapter 6 (Rehabilitation Robot Control: Human-Robot Interaction Control), HRI models were used in NMPC control structures for model-based control. In the following, outcomes of each chapter are discussed.

In Chapter 3 (System Models: Rehabilitation Robot), based on the discussed mechan-

ical issues of the robot, dynamic models of the robot using our two proposed DPI methods were presented. First, we proposed a modified homotopy optimization to identify parameters of the system dynamics with mechanical discontinuity (i.e., backlash). After verifying the proposed method using computer simulations, the method was used to identify the parameters of the 2 DOF noisy and nonlinear rehabilitation robot with actuator backlash. The proposed method provided better estimates for the system parameters compared to the no-backlash DPI of the robot. Although the obtained accuracy for 1D motion was good, the uncertainty of the states during robot interaction and the computation cost caused by the discontinuity led us to develop a continuous model for the 2D robot. Thus, next, we presented a direct collocation method for the DPI. Different identification tests were performed, and the results of the proposed method were verified. The robot dynamics were modeled by two different friction models: LuGre and continuous-velocity models, and their accuracies were compared. They had the same accuracy, but the continuous-velocity model had less computation cost.

In Chapter 4 (System Models: Human and Human-Robot Interaction), since the high-fidelity 3D musculoskeletal model was not suitable for real-time simulations, we developed different 2D models suitable for real-time simulations. In this chapter, results of cross-validation studies showed that it is possible to use a 2D musculoskeletal arm model for evaluation and control of the planar robot. However, development of this 2D musculoskeletal arm model required a 3D musculoskeletal arm model (MA3D-II) with muscle wrappings. Furthermore, sensitivity studies of the upper extremity models showed that the estimated muscle forces are highly dependent on subject-specific parameters. Thus, a method for the DPI of the 2D model (MA2D-II) was provided to include subject-specific parameters in the model.

In Chapter 5 (Rehabilitation Robot Control: Robot Control), we developed impedance-based assistance controllers for the rehabilitation robot. The goal was to achieve a *reliable* and real-time implementable controller. In our definition, a reliable controller is capable of handling variable exercises and admittance interactions. The controller should reduce therapist intervention and improve the quality of the rehabilitation. Overall, two types of impedance-based controllers were developed, and their performances were discussed. First, to improve impedance control performance in variable admittance environments, the compliance is controlled by an outer loop force control (i.e., HFIC). Thus, in a tangential subspace, the proposed low-level controller (HFIC) was trying to ensure force tracking with bounded position error. In an orthogonal subspace, the position tracking with bounded force error was achieved. HFIC only controls the compliance (i.e., interaction force) term, and the impedance gains are not optimal. Thus, it results in a superficial variable impedance model and requires careful selection of the gains. As the second low-level controller,

a general method (OIC) that optimally adjusts impedance gains for variable robot configurations was developed and tuned through simulating the human-robot system. For this controller, an integrated human-robot dynamic system was used to fine-tune the controller gains. This method was advantageous for efficient tuning of the robot controllers in experiments. A good qualitative agreement between experimental and simulation results verified the effectiveness of this method compared to the conventional SOIC.

Although HFIC was successful in assisting healthy and less strong subjects, it was not recommended to be used in interaction with weak subjects. This disadvantage might be resolved by introducing an outer-loop attractive force field controller for HFIC, that was called attractive force field method and is presented in this chapter. Furthermore, for a linear robot model, the OIC assumes an apparent mass for the robot equal to its mass matrix, while the SOIC permits offline changes to the robot's apparent mass. For optimal results, the robot's apparent mass should vary online as a function of the input frequencies of the system. However, neither the OIC nor SOIC offered such updates. In this chapter, a rule (adaptive optimal impedance control) for the online change of the robot's apparent mass was developed. Results of the simulations showed that at higher input force frequencies or higher environmental mass, the adaption law does not change the apparent mass. The method was verified by simulation. The method was designed for a 1D interaction model; however, it can be extended to 2D interaction models.

Chapter 6 (Rehabilitation Robot Control: Human-Robot Interaction Control) presented HRI control structures that leverage kinematic or dynamic information from the subject to control the robot. In this chapter, we introduced two types of model-based HRI control structures, which were implemented using NMPC in model-in-loop simulations. In the first controller (NMPC with an external 3D musculoskeletal model), the tracking gain of the objective functional was varied by the results of an external FSO, which (in parallel) was optimizing muscular activations of an external 3D musculoskeletal model (high-fidelity model) for a specific task. Simulation results showed the success of this controller. Since the FSO of the high-fidelity model ran externally, it increased computation cost. Consequently, this controller was not implementable in real-time studies. As the second approach, a low-fidelity model (i.e., a 2D musculoskeletal model) was used within the structure of the NMPC to reduce the computation cost. This model also needed a reliable DPI of the subject. Then, the FSO and the robot controllers were integrated and a single NMPC with the 2D HRI model was considered to predict muscle activations along with optimal robot motor torque inputs. The controller performance was excellent in simulations and was more efficient (two times faster) than the previous one. However, since it was implemented using GPOPS-II, it was not possible to use it in real-time studies. Finally, real-time implementation of the NMPC with the internal 2D musculoskeletal model was

provided. In experiments, the robot was able to accomplish the task with small interaction force and tracking error. However, it was predicting the relaxed subject's muscular activities in higher amounts. This discrepancy may be due to inaccurate musculoskeletal model parameters, exclusion of the PE, and active counterbalancing by the subject. Despite these discrepancies, this controller is efficient for the rehabilitation robot since it is possible to interpret the predicted muscular activities during a rehabilitation therapy. Furthermore, the motor torques and interaction forces on the horizontal plane have the same trends as in the simulations, which provides evidence for the effectiveness of the model-based controller in real-time experiments.

7.1 List of contributions

Contributions of this research include:

1. Dynamic parameter identification (DPI) of a rehabilitation manipulandum with a nonlinear and discontinuous dynamic model. This work can be extended directly to any rehabilitation manipulandum.
2. Development of a 3D musculoskeletal upper extremity model with muscle wrapping according to ISB recommendations.
3. Construction of a low-fidelity (2D) musculoskeletal model based on the 3D musculoskeletal model.
4. Dynamic parameter identification of the musculoskeletal upper extremity model.
5. Construction of a human-robot interaction model.
6. Development of a hybrid force-impedance and optimal impedance controller for rehabilitation.
7. Development of real-time human-robot interaction controllers using a model-based NMPC framework.

7.2 Future work

For future research, the following steps are recommended:

1. Evaluate the performance of the DPI for systems with hysteresis and dead-zone discontinuities.
2. Assess the DPI results for higher-fidelity musculoskeletal models by doing more experiments with motion capture and EMG.
3. Apply the proposed DPI methods on other robots and human subjects.
4. Evaluate the performance of the proposed control strategies for the rehabilitation robot by performing clinical trials.
5. Analyze the stability of the proposed impedance-based controllers.
6. Study the effect of model uncertainty on the human-robot interaction control performance.
7. Perform experiments with EMG sensors for the biomechanical validation of the real-time NMPC framework.
8. Compare the performances of the proposed controllers with the same type of experiments and therapy modes.
9. Implement the proposed human-robot interaction control with learning-based algorithms to adapt to different individuals.
10. Integrate the proposed human-robot interaction control with robust control approaches (e.g. Monte Carlo simulations) to eliminate possible model uncertainty and instability issues.

References

[1] W. Abend, E. Bizzi, and P. Morasso. Human arm trajectory formation. *Brain: a Journal of Neurology*, 105(Pt 2):331–348, 1982.

[2] M. Ackermann. *Dynamics and energetics of walking with prostheses*. Phd thesis, University of Stuttgart, Stuttgart, Germany, 2007.

[3] M. Ackermann and W. Schiehlen. Physiological methods to solve the force-sharing problem in biomechanics. *Computational Methods in Applied Sciences*, 12:1–23, 2008.

[4] F. Alambeigi, A. Zamani, G. Vossoughi, and M. R. Zakerzadeh. Robust shape control of two SMA actuators attached to a flexible beam based on DK iteration. In *2012 12th International Conference on Control, Automation and Systems*, pages 316–321, 2012.

[5] J. Ambrósio, C. Quental, B. Pilarczyk, and J. Folgado. Multibody biomechanical models of the upper limb. *Procedia IUTAM*, 2:4–17, 2011. ISSN 2210-9838. doi: 10.1016/j.piutam.2011.04.002.

[6] F. Amirabdollahian. Rehabilitation robots. In K. Dautenhahn and J. Saunders, editors, *New Frontiers in Human-Robot Interaction*, volume 2 of *Advances in Interaction Studies*, pages 305–326. John Benjamins Publishing Company, Amsterdam, dec 2011. ISBN 978 90 272 0455 4. doi: 10.1075/ais.2.

[7] F. Amirabdollahian, R. Loureiro, E. Gradwell, C. Collin, W. Harwin, and G. Johnson. Multivariate analysis of the Fugl-Meyer outcome measures assessing the effectiveness of GENTLE/S robot-mediated stroke therapy. *Journal of NeuroEngineering and Rehabilitation*, 4(1):4, mar 2007. ISSN 1743-0003. doi: 10.1186/1743-0003-4-4.

[8] K. N. An, F. C. Hui, B. F. Morrey, R. L. Linscheid, and E. Y. Chao. Muscles across the elbow joint: A biomechanical analysis. *Journal of Biomechanics*, 14:659–669, 1981. ISSN 00219290. doi: 10.1016/0021-9290(81)90048-8.

[9] K. N. An, K. Takahashi, T. P. Harrigan, and E. Y. Chao. Determination of muscle orientations and moment arms. *Journal of Biomechanical Engineering*, 106:280–282, 1984. ISSN 01480731. doi: 10.1115/1.3138494.

[10] F. C. Anderson and M. G. Pandy. Dynamic optimization of human walking. *Journal of Biomechanical Engineering*, 123:381, 2001. ISSN 01480731. doi: 10.1115/1.1392310.

[11] R. Anderson and M. Spong. Hybrid impedance control of robotic manipulators. *IEEE Journal on Robotics and Automation*, 4(5):549–556, 1988. ISSN 08824967. doi: 10.1109/56.20440.

[12] M. A. Arain, H. V. Hultmann Ayala, and M. A. Ansari. *Nonlinear system identification using neural network*, pages 122–131. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012. ISBN 978-3-642-28962-0. doi: 10.1007/978-3-642-28962-0_13.

[13] S. Balasubramanian, Ruihua Wei, M. Perez, B. Shepard, E. Koeneman, J. Koeneman, and Jiping He. RUPERT: An exoskeleton robot for assisting rehabilitation of arm functions. In *2008 International Conference on Virtual Rehabilitation (ICVR)*, pages 163–167. IEEE, aug 2008. ISBN 978-1-4244-2700-0. doi: 10.1109/ICVR.2008.4625154.

[14] S. Bansil, N. Prakash, J. Kaye, S. Wrigley, C. Manata, C. Stevens-Haas, and R. Kurlan. Movement disorders after stroke in adults: A review. *Tremor and Other Hyperkinetic Movements*, 2:1–7, jan 2012. ISSN 2160-8288.

[15] M. Barsotti, D. Leonardis, C. Loconsole, M. Solazzi, E. Sotgiu, C. Procopio, C. Chisari, M. Bergamasco, and A. Frisoli. A full upper limb robotic exoskeleton for reaching and grasping rehabilitation triggered by MI-BCI. In *2015 IEEE International Conference on Rehabilitation Robotics (ICORR)*, pages 49–54. IEEE, aug 2015. ISBN 978-1-4799-1808-9. doi: 10.1109/ICORR.2015.7281174.

[16] R. W. Bassett, A. O. Browne, B. F. Morrey, and K. N. An. Glenohumeral muscle force and moment mechanics in a position of shoulder instability. *Journal of Biomechanics*, 23:405–415, 1990. ISSN 00219290. doi: 10.1016/0021-9290(90)90295-E.

[17] N. A. Battista, A. J. Baird, and L. A. Miller. A mathematical model and MATLAB code for muscle-fluid-structure simulations. In *Integrative and Comparative Biology*, volume 55, pages 901–911. Princeton University Press, Princeton, nov 2015. ISBN 1540-7063. doi: 10.1093/icb/icv102.

[18] B. Berret, C. Darlot, F. Jean, T. Pozzo, C. Papaxanthis, and J. P. Gauthier. The inactivation principle: Mathematical solutions minimizing the absolute work and biological implications for the planning of arm movements. *PLoS Computational Biology*, 4(10):e1000194, oct 2008. ISSN 1553734X. doi: 10.1371/journal.pcbi.1000194.

[19] A. Biscarini. Dynamic effects of muscle moment arm variation and heavy external loads on hinge joints. *Journal of Applied Biomechanics*, 19:223–38., 2003.

[20] E. Bizzi, V. C. Cheung, A. D'Avella, P. Saltiel, and M. Tresch. Combining modules for movement. *Brain Research Reviews*, 57(1):125–133, jan 2008. ISSN 01650173. doi: 10.1016/j.brainresrev.2007.08.004.

[21] D. Blacquiere, M. P. Lindsay, N. Foley, C. Taralson, S. Alcock, C. Balg, S. Bhogal, J. Cole, M. Eustace, P. Gallagher, A. Ghanem, A. Hoechsmann, G. Hunter, K. Khan, A. Marrero, B. Moses, K. Rayner, A. Samis, E. Smitko, M. Vibe, G. Gubitz, D. Dowlatshahi, S. Phillips, and F. L. Silver. Canadian stroke best practice recommendations: Telestroke best practice guidelines update 2017. *International Journal of Stroke*, 12(8):886–895, oct 2017. ISSN 1747-4930. doi: 10.1177/1747493017706239.

[22] D. Blana, R. F. Kirsch, and E. K. Chadwick. Combined feedforward and feedback control of a redundant, nonlinear, dynamic musculoskeletal system. *Medical & Biological Engineering & Computing*, 47(5):533–42, may 2009. ISSN 1741-0444. doi: 10.1007/s11517-009-0479-3.

[23] S. S. Blemker and S. L. Delp. Three-dimensional representation of complex muscle architectures and geometries. *Annals of Biomedical Engineering*, 33(5):661–673, may 2005. ISSN 0090-6964. doi: 10.1007/s10439-005-1433-7.

[24] B. M. Bolhuis, C. C. Gielen, and G. J. van Ingen Schenau. Activation patterns of mono- and bi-articular arm muscles as a function of force and movement direction of the wrist in humans. *The Journal of Physiology*, 508(Pt 1):313–24, apr 1998. ISSN 0022-3751.

[25] R. A. Brand, R. D. Crowninshield, C. E. Wittstock, D. R. Pedersen, C. R. Clark, and F. M. van Krieken. A model of lower extremity muscular anatomy. *Journal of Biomechanical Engineering*, 104:304–310, 1982. ISSN 01480731. doi: 10.1115/1.3138363.

[26] R. A. Brand, D. R. Pedersen, D. T. Davy, G. M. Kotzar, K. G. Heiple, and V. M. Goldberg. Comparison of hip force calculations and measurements in the

same patient. *The Journal of Arthroplasty*, 9:45–51, 1994. ISSN 08835403. doi: 10.1016/0883-5403(94)90136-8.

[27] D. Brauchle, M. Vukelić, R. Bauer, and A. Gharabaghi. Brain state-dependent robotic reaching movement with a multi-joint arm exoskeleton: Combining brain-machine interfacing and robotic rehabilitation. *Frontiers in Human Neuroscience*, 9: 564, oct 2015. ISSN 1662-5161. doi: 10.3389/fnhum.2015.00564.

[28] B. Brewer, R. Klatzky, and Y. Matsuoka. Initial therapeutic results of visual feedback manipulation in robotic rehabilitation. In *2006 International Workshop on Virtual Rehabilitation*, pages 160–166. IEEE, 2006. ISBN 1-4244-0280-8. doi: 10.1109/IWVR.2006.1707546.

[29] B. R. Brewer, S. K. McDowell, and L. C. Worthen-Chaudhari. Poststroke upper extremity rehabilitation: A review of robotic systems and clinical results. *Topics in Stroke Rehabilitation*, 14(6):22–44, 2007. ISSN 1074-9357. doi: 10.1310/tsr1406-22.

[30] S. Brochard, J. Robertson, B. Médée, and O. Rémy-Néris. What's new in new technologies for upper extremity rehabilitation? *Current Opinion in Neurology*, 23: 683–687, 2010. ISSN 1350-7540. doi: 10.1097/WCO.0b013e32833f61ce.

[31] J. Broeren, M. Rydmark, and K. S. Sunnerhagen. Virtual reality and haptics as a training device for movement rehabilitation after stroke: A single-case study. *Archives of Physical Medicine and Rehabilitation*, 85:1247–1250, 2004. ISSN 00039993. doi: 10.1016/j.apmr.2003.09.020.

[32] J. Broeren, M. Dixon, K. S. Sunnerhagen, and M. Rydmark. Rehabilitation after stroke using virtual reality, haptics (force feedback) and telemedicine. *Studies in health technology and informatics*, 124:51–6, 2006. ISSN 0926-9630.

[33] E. B. Brokaw, T. Murray, T. Nef, and P. S. Lum. Retraining of interjoint arm coordination after stroke using robot-assisted time-independent functional training. *Journal of rehabilitation research and development*, 48(4):299–316, 2011. ISSN 1938-1352.

[34] P. Brown and J. McPhee. A continuous velocity-based friction model for dynamics and control with physically meaningful parameters. *Journal of Computational and Nonlinear Dynamics*, 11(5):054502, jun 2016. ISSN 1555-1415. doi: 10.1115/1.4033658.

[35] T. S. Buchanan, M. J. Moniz, J. P. Dewald, and W. Zev Rymer. Estimation of muscle forces about the wrist joint during isometric tasks using an EMG coefficient method. *Journal of Biomechanics*, 26(4-5):547–560, 1993.

[36] G. C. Burdea. Virtual rehabilitation–Benefits and challenges. *Methods of information in medicine*, 42(5):519–23, 2003. ISSN 0026-1270. doi: 10.1267/METH03050519.

[37] S. G. Campbell, P. C. Hatfield, and K. S. Campbell. A mathematical model of muscle containing heterogeneous half-sarcomeres exhibits residual force enhancement. *PLoS Computational Biology*, 7(9):e1002156, sep 2011. ISSN 1553734X. doi: 10.1371/journal.pcbi.1002156.

[38] R. Cano-de-la Cuerda, A. Molero-Sánchez, M. Carratalá-Tejada, I. Alguacil-Diego, F. Molina-Rueda, J. Miangolarra-Page, and D. Torricelli. Theories and control models and motor learning: Clinical applications in neurorehabilitation. *Neurología (English Edition)*, 30(1):32–41, jan 2015. ISSN 21735808. doi: 10.1016/j.nrleng.2011.12.012.

[39] C. Carignan, J. Tang, and S. Roderick. Development of an exoskeleton haptic interface for virtual task training. In *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3697–3702. IEEE, oct 2009. ISBN 978-1-4244-3803-7. doi: 10.1109/IROS.2009.5354834.

[40] M. G. Carmichael and D. Dikai Liu. Estimating physical assistance need using a musculoskeletal model. *IEEE Transactions on Biomedical Engineering*, 60(7):1912–1919, jul 2013. ISSN 0018-9294. doi: 10.1109/TBME.2013.2244889.

[41] M. G. Carmichael and D. Dikai Liu. Upper limb strength estimation of physically impaired persons using a musculoskeletal model: A sensitivity analysis. In *2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, volume 2015, pages 2438–2441. IEEE, aug 2015. ISBN 978-1-4244-9271-8. doi: 10.1109/EMBC.2015.7318886.

[42] B. Cesqui, P. Tropea, S. Micera, and H. Krebs. EMG-based pattern recognition approach in post stroke robot-aided rehabilitation: A feasibility study. *Journal of NeuroEngineering and Rehabilitation*, 10(1):75, 2013. ISSN 1743-0003. doi: 10.1186/1743-0003-10-75.

[43] E. K. Chadwick, D. Blana, R. F. Kirsch, and A. J. van den Bogert. Real-time simulation of three-dimensional shoulder girdle and arm dynamics. *IEEE Trans-*

actions on Biomedical Engineering, 61(7):1947–56, jul 2014. ISSN 1558-2531. doi: 10.1109/TBME.2014.2309727.

[44] J. H. Challis and D. G. Kerwin. An analytical examination of muscle force estimations using optimization techniques. *Proceedings of the Institution of Mechanical Engineers. Part H, Journal of Engineering in Medicine*, 207:139–148, 1993. ISSN 0954-4119. doi: 10.1243/PIME_PROC_1993_207_286_02.

[45] E. Y. Chao, R. S. Armiger, H. Yoshida, J. Lim, and N. Haraguchi. Virtual interactive musculoskeletal system (VIMS) in orthopaedic research, education and clinical patient care. *Journal of Orthopaedic Surgery and Research*, 2(1):2, mar 2007. ISSN 1749799X. doi: 10.1186/1749-799X-2-2.

[46] I. W. Charlton and G. R. Johnson. Application of spherical and cylindrical wrapping algorithms in a musculoskeletal model of the upper limb. *Journal of Biomechanics*, 34:1209–1216, 2001. ISSN 00219290. doi: 10.1016/S0021-9290(01)00074-4.

[47] I. W. Charlton and G. R. Johnson. A model for the prediction of the forces at the glenohumeral joint. *Proceedings of the Institution of Mechanical Engineers, Part H: Journal of Engineering in Medicine*, 220(8):801–812, jan 2006. ISSN 0954-4119. doi: 10.1243/09544119JEIM147.

[48] S. Chen, S. A. Billings, and P. M. Grant. Non-linear system identification using neural networks. *International Journal of Control*, 51(6):1191–1214, jan 1990. ISSN 0020-7179. doi: 10.1080/00207179008934126.

[49] C. Choi and J. Kim. A real-time EMG-based assistive computer interface for the upper limb disabled. In *2007 IEEE 10th International Conference on Rehabilitation Robotics (ICORR)*, pages 459–462. IEEE, jun 2007. ISBN 978-1-4244-1319-5. doi: 10.1109/ICORR.2007.4428465.

[50] R. I. Close. Dynamic properties of mammalian skeletal muscles. *Physiological Reviews*, 52:129–197, 1972. ISSN 0031-9333.

[51] R. Colombo, F. Pisano, S. Micera, A. Mazzone, C. Delconte, M. C. Carrozza, P. Dario, and G. Minuco. Robotic techniques for upper limb evaluation and rehabilitation of stroke patients. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 13(3):311–24, sep 2005. ISSN 1534-4320. doi: 10.1109/TNSRE.2005.848352.

[52] V. Crocher, A. Sahbani, J. Robertson, A. Roby-Brami, and G. Morel. Constraining upper limb synergies of hemiparetic patients using a robotic exoskeleton in the perspective of neuro-rehabilitation. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 20(3):247–257, may 2012. ISSN 1534-4320. doi: 10.1109/TNSRE.2012.2190522.

[53] K. M. Crossley, T. W. Dorn, H. Ozturk, J. van den Noort, a. G. Schache, and M. G. Pandy. Altered hip muscle forces during gait in people with patellofemoral osteoarthritis. *Osteoarthritis and cartilage / OARS, Osteoarthritis Research Society*, 20(11):1243–9, nov 2012. ISSN 1522-9653. doi: 10.1016/j.joca.2012.07.011.

[54] J. L. Crow, N. B. Lincoln, F. M. Nouri, and W. D. Weerdt. The effectiveness of EMG biofeedback in the treatment of arm function after stroke. *International Disability Studies*, 11(4):155–160, jan 1989. ISSN 0259-9147. doi: 10.3109/03790798909166667.

[55] R. D. Crowninshield and R. a. Brand. A physiologically based criterion of muscle force prediction in locomotion. *Journal of Biomechanics*, 14(11):793–801, jan 1981. ISSN 0021-9290.

[56] R. D. Crowninshield, R. C. Johnston, J. G. Andrews, and R. A. Brand. A biomechanical investigation of the human hip. *Journal of Biomechanics*, 11:75–85, 1978. ISSN 00219290. doi: 10.1016/0021-9290(78)90045-3.

[57] P. R. Culmer, A. E. Jackson, S. Makower, R. Richardson, J. A. Cozens, M. C. Levesley, and B. B. Bhakta. A control strategy for upper limb robotic rehabilitation with a dual robot system. *IEEE/ASME Transactions on Mechatronics*, 15(4):575–585, aug 2010. ISSN 1083-4435. doi: 10.1109/TMECH.2009.2030796.

[58] 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*, 14(8):1100–1111, nov 2006. ISSN 1569190X. doi: 10.1016/j.simpat.2006.09.001.

[59] J. R. Davis. *Surface Engineering for Corrosion and Wear Resistance*. Number 751. ASM International, 2001. ISBN 0871707004. doi: 10.1361/secw2001p001.

[60] S. Davoudabadi Farahani, M. Svinin, M. S. Andersen, M. de Zee, and J. Rasmussen. Prediction of closed-chain human arm dynamics in a crank-rotation task. *Journal of Biomechanics*, 49(13):2684–2693, sep 2016. ISSN 00219290. doi: 10.1016/j.jbiomech.2016.05.034.

[61] D. T. Davy and M. L. Audu. A dynamic optimization technique for predicting muscle forces in the swing phase of gait. *Journal of Biomechanics*, 20:187–201, 1987. ISSN 00219290. doi: 10.1016/0021-9290(87)90310-1.

[62] S. L. Delp and F. E. Zajac. Force- and moment-generating capacity of lower-extremity muscles before and after tendon lengthening. *Clinical Orthopaedics and Related Research*, (284):247–59, nov 1992. ISSN 0009-921X.

[63] S. L. Delp, J. P. Loan, M. G. Hoy, F. E. Zajac, E. L. Topp, and J. M. Rosen. An interactive graphics-based model of the lower extremity to study orthopaedic surgical procedures. *IEEE Transactions on Biomedical Engineering*, 37:757–767, 1990. ISSN 00189294. doi: 10.1109/10.102791.

[64] S. L. Delp, F. C. Anderson, A. S. Arnold, P. Loan, A. Habib, C. T. John, E. Guendelman, and D. G. Thelen. OpenSim: Open-source software to create and analyze dynamic simulations of movement. *IEEE Transactions on Biomedical Engineering*, 54 (11):1940–50, nov 2007. ISSN 0018-9294. doi: 10.1109/TBME.2007.901024.

[65] C. R. Dickerson. Modeling and simulation of tissue load in the upper extremities. In H. Youlian and R. Bartlett, editors, *Routledge Handbook of Biomechanics and Human Movement Science*, chapter 2, pages 18–34. Routledge, 2008. ISBN 9780415408813.

[66] C. R. Dickerson, D. B. Chaffin, and R. E. Hughes. A mathematical musculoskeletal shoulder model for proactive ergonomic analysis. *Computer Methods in Biomechanics and Biomedical Engineering*, 10(6):389–400, dec 2007. ISSN 1025-5842. doi: 10.1080/10255840701592727.

[67] D. Ding, R. Simpson, Y. Matuoka, and E. LoPresti. Rehabilitation robotics. In R. A. Cooper, H. Ohnabe, and D. A. Hobson, editors, *An Introduction to Rehabilitation Engineering*, pages 211–238. Taylor & Francis, 2007. ISBN 9781420012491.

[68] M. Ding, J. Ueda, and T. Ogasawara. Pinpointed muscle force control using a power-assisting device: System configuration and experiment. In *Proceedings of the 2nd Biennial IEEE/RAS-EMBS International Conference on Biomedical Robotics and Biomechatronics, BioRob 2008*, pages 181–186. IEEE, oct 2008. ISBN 9781424428830. doi: 10.1109/BIOROB.2008.4762829.

[69] M. Ding, K. Hirasawa, Y. Kurita, H. Takemura, J. Takamatsu, H. Mizoguchi, and T. Ogasawara. Pinpointed muscle force control in consideration of human motion and external force. In *2010 IEEE International Conference on Robotics and Biomimetics*

(ROBIO), pages 739–744. IEEE, dec 2010. ISBN 9781424493173. doi: 10.1109/ROBIO.2010.5723418.

[70] L. Dipietro, M. Ferraro, J. J. Palazzolo, H. I. Krebs, B. T. Volpe, and N. Hogan. Customized interactive robotic treatment for stroke: EMG-triggered therapy. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 13(3):325–34, sep 2005. ISSN 1534-4320. doi: 10.1109/TNSRE.2005.850423.

[71] S. Dong, K.-Q. Lu, J. Q. Sun, and K. Rudolph. Smart rehabilitation devices: Part II - Adaptive motion control. *Journal of intelligent material systems and structures*, 17(7):555–561, jan 2006. ISSN 1045-389X. doi: 10.1177/1045389X06059076.

[72] W. M. Dos Santos and A. A. Siqueira. Optimal impedance control for robot-Aided rehabilitation of walking based on estimation of patient behavior. In *Proceedings of the IEEE RAS and EMBS International Conference on Biomedical Robotics and Biomechatronics*, volume 2016-July, pages 1023–1028. IEEE, jun 2016. ISBN 9781509032877. doi: 10.1109/BIOROB.2016.7523765.

[73] W. F. Dostal and J. G. Andrews. A three-dimensional biomechanical model of hip musculature. *Journal of Biomechanics*, 14:803–812, 1981. ISSN 00219290. doi: 10.1016/0021-9290(81)90036-1.

[74] D. M. Duc, T. Kazuhiko, and M. Takanori. EMG-moment model of human arm for rehabilitation robot system. In *2008 10th International Conference on Control, Automation, Robotics and Vision*, pages 190–195. IEEE, dec 2008. ISBN 978-1-4244-2286-9. doi: 10.1109/ICARCV.2008.4795515.

[75] J. Dul. A biomechanical model to quantify shoulder load at the work place. *Clinical Biomechanics*, 3(3):124–8, aug 1988. ISSN 0268-0033. doi: 10.1016/0268-0033(88)90057-5.

[76] R. Dumas, L. Chèze, and J.-P. Verriest. Adjustments to McConville et al. and Young et al. body segment inertial parameters. *Journal of Biomechanics*, 40(3):543–53, jan 2007. ISSN 0021-9290. doi: 10.1016/j.jbiomech.2006.02.013.

[77] S. Duprey, A. Naaim, F. Moissenet, M. Begon, L. Chèze, K.-N. An, W. Murray, and J. Arora. Kinematic models of the upper limb joints for multibody kinematics optimisation: An overview. *Journal of Biomechanics*, 1(0):271–293, dec 2016. ISSN 00219290. doi: 10.1016/j.jbiomech.2016.12.005.

[78] G. Durandau, D. Farina, and M. Sartori. Real-time musculoskeletal modeling driven by electromyograms. *IEEE Transactions on Biomedical Engineering*, pages 1–1, may 2017. ISSN 0018-9294. doi: 10.1109/TBME.2017.2704085.

[79] M. El-Gohary and J. McNames. Shoulder and elbow joint angle tracking with inertial sensors. *IEEE Transactions on Biomedical Engineering*, 59(9):2635–2641, sep 2012. ISSN 0018-9294. doi: 10.1109/TBME.2012.2208750.

[80] M. El-Gohary and J. McNames. Human joint angle estimation with inertial sensors and validation with a robot arm. *IEEE Transactions on Biomedical Engineering*, 62 (7):1759–1767, jul 2015. ISSN 0018-9294. doi: 10.1109/TBME.2015.2403368.

[81] B. M. Elbagoury and L. Vladareanu. A hybrid real-time EMG intelligent rehabilitation robot motions control based on Kalman Filter, support vector machines and particle swarm optimization. In *2016 10th International Conference on Software, Knowledge, Information Management & Applications (SKIMA)*, pages 439–444. IEEE, 2016. ISBN 978-1-5090-3298-3. doi: 10.1109/SKIMA.2016.7916262.

[82] S. Engelbrecht and J. Fernandez. Invariant characteristics of horizontal-plane minimum-torque-change movements with one mechanical degree of freedom. *Biological cybernetics*, 76(5):321–329, 1997. ISSN 0340-1200. doi: 10.1007/s004220050345.

[83] D. Erol and N. Sarkar. Design and implementation of an assistive controller for rehabilitation robotic systems. *International Journal of Advanced Robotic Systems*, 4(3):271–278, 2007.

[84] D. Erol and N. Sarkar. Intelligent control for robotic rehabilitation after stroke. *Journal of Intelligent and Robotic Systems*, 50(4):341–360, nov 2007. ISSN 0921-0296. doi: 10.1007/s10846-007-9169-2.

[85] P. Favre, J. G. Snedeker, and C. Gerber. Numerical modelling of the shoulder for clinical applications. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 367(1895):2095–2118, may 2009. ISSN 1364-503X. doi: 10.1098/rsta.2008.0282.

[86] P. Favre, C. Gerber, and J. G. Snedeker. Automated muscle wrapping using finite element contact detection. *Journal of Biomechanics*, 43(10):1931–1940, jul 2010. ISSN 00219290. doi: 10.1016/j.jbiomech.2010.03.018.

[87] T. Flash and N. Hogan. The coordination of arm movements: An experimentally confirmed mathematical model. *The Journal of neuroscience*, 5(7):1688–1703, 1985. ISSN 0270-6474. doi: 4020415.

[88] S. Fok, R. Schwartz, M. Wronkiewicz, C. Holmes, J. Zhang, T. Somers, D. Bundy, and E. Leuthardt. An EEG-based brain computer interface for rehabilitation and restoration of hand control following stroke using ipsilateral cortical physiology. In *2011 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pages 6277–6280. IEEE, aug 2011. ISBN 978-1-4577-1589-1. doi: 10.1109/IEMBS.2011.6091549.

[89] A. Frank C. and M. G. Pandy. A dynamic optimization solution for vertical jumping in three dimensions. *Computer Methods in Biomechanics and Biomedical Engineering*, 2(3):201–231, jan 1999. ISSN 1025-5842. doi: 10.1080/10255849908907988.

[90] C. T. Freeman, A.-M. Hughes, J. H. Burridge, P. H. Chappell, P. L. Lewin, and E. Rogers. A model of the upper extremity using FES for stroke rehabilitation. *Journal of Biomechanical Engineering*, 131:031011, 2009. ISSN 01480731. doi: 10.1115/1.3005332.

[91] B. J. Fregly, M. L. Boninger, and D. J. Reinkensmeyer. Personalized neuromusculoskeletal modeling to improve treatment of mobility impairments: A perspective from European research sites. *Journal of NeuroEngineering and Rehabilitation*, 9(1):18, 2012. ISSN 1743-0003. doi: 10.1186/1743-0003-9-18.

[92] A. Frisoli, C. Loconsole, D. Leonardis, F. Banno, M. Barsotti, C. Chisari, and M. Bergamasco. A new gaze-BCI-driven control of an upper limb exoskeleton for rehabilitation in real-world tasks. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 42(6):1169–1179, nov 2012. ISSN 1094-6977. doi: 10.1109/TSMCC.2012.2226444.

[93] G. Fulk, S. B. O'sullivan, and T. J. Schmitz. *Physical Rehabilitation*. F.A. Davis Company, 6th edition, 2014. ISBN 978-0-8036-2579-2.

[94] F. Gao, M. Damsgaard, J. Rasmussen, and S. Tørholm Christensen. Computational method for muscle-path representation in musculoskeletal models. *Biological Cybernetics*, 87(3):199–210, sep 2002. ISSN 03401200. doi: 10.1007/s00422-002-0326-1.

[95] D. Garg, M. A. Patterson, C. Francolin, C. L. Darby, G. T. Huntington, W. W. Hager, and A. V. Rao. Direct trajectory optimization and costate estimation offinite-horizon and infinite-horizon optimal control problems using a Radau pseudospectral

method. *Computational Optimization and Applications*, 49(2):335–358, jun 2011. ISSN 0926-6003. doi: 10.1007/s10589-009-9291-0.

[96] B. a. Garner and M. G. Pandy. A kinematic model of the upper limb based on the visible human project (VHP) image dataset. *Computer Methods in Biomechanics and Biomedical Engineering*, 2(2):107–124, jan 1999. ISSN 1476-8259. doi: 10.1080/10255849908907981.

[97] B. A. Garner and M. G. Pandy. The obstacle-set method for representing muscle paths in musculoskeletal models. *Computer Methods in Biomechanics and Biomedical Engineering*, 3:1–30, 2000. ISSN 1025-5842. doi: 10.1080/10255840008915251.

[98] 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*, 4(2):93–126, feb 2001. ISSN 1025-5842. doi: 10.1080/10255840008908000.

[99] B. a. Garner and M. G. Pandy. Estimation of musculotendon properties in the human upper limb. *Annals of Biomedical Engineering*, 31(2):207–220, feb 2003. ISSN 0090-6964. doi: 10.1114/1.1540105.

[100] B. Ghannadi, N. Mehrabi, and J. McPhee. Hybrid force-impedance control of an upper-limb stroke rehabilitation robot interacting with a musculoskeletal arm model. In *Proceedings of the 7th World Congress of Biomechanics*, Boston, USA, 2014.

[101] B. Ghannadi, N. Mehrabi, and J. McPhee. Nonlinear model predictive control of a human-robot rehabilitation system. In *Canadian Society for Biomechanics*, Hamilton, Ontario, 2016.

[102] M. A. Goodrich and A. C. Schultz. Human-robot interaction: A survey. *Foundations and Trends in Human-Computer Interaction*, 1(3):203–275, 2007. ISSN 1551-3955. doi: 10.1561/1100000005.

[103] M. Guidali, M. Schmiedeskamp, V. Klamroth, and R. Riener. Assessment and training of synergies with an arm rehabilitation robot. In *2009 IEEE International Conference on Rehabilitation Robotics (ICORR)*, pages 772–776. IEEE, jun 2009. ISBN 9781424437894. doi: 10.1109/ICORR.2009.5209516.

[104] M. Guidali, A. Duschau-Wicke, S. Broggi, V. Klamroth-Marganska, T. Nef, and R. Riener. A robotic system to train activities of daily living in a virtual environment. *Medical & Biological Engineering & Computing*, 49(10):1213–1223, oct 2011. ISSN 0140-0118. doi: 10.1007/s11517-011-0809-0.

[105] S. Guo, W. Zhang, W. Wei, J. Guo, Y. Ji, and Y. Wang. A kinematic model of an upper limb rehabilitation robot system. In *2013 IEEE International Conference on Mechatronics and Automation*, pages 968–973. IEEE, aug 2013. ISBN 978-1-4673-5560-5. doi: 10.1109/ICMA.2013.6618046.

[106] A. Gupta and M. O'Malley. Design of a haptic arm exoskeleton for training and rehabilitation. *IEEE/ASME Transactions on Mechatronics*, 11(3):280–289, jun 2006. ISSN 1083-4435. doi: 10.1109/TMECH.2006.875558.

[107] D. Hardt. Determining muscle forces in the leg during normal human walking - An application and evaluation of optimization methods. *Journal of Biomechanical Engineering*, 100(May):72–78, 1978.

[108] D. Hebert, M. P. Lindsay, A. McIntyre, A. Kirton, P. G. Rumney, S. Bagg, M. Bayley, D. Dowlatshahi, S. Dukelow, M. Garnhum, E. Glasser, M.-L. Halabi, E. Kang, M. MacKay-Lyons, R. Martino, A. Rochette, S. Rowe, N. Salbach, B. Semenko, B. Stack, L. Swinton, V. Weber, M. Mayer, S. Verrilli, G. DeVeber, J. Andersen, K. Barlow, C. Cassidy, M.-E. Dilenge, D. Fehlings, R. Hung, J. Iruthayarajah, L. Lenz, A. Majnemer, J. Purtzki, M. Rafay, L. K. Sonnenberg, A. Townley, S. Janzen, N. Foley, and R. Teasell. Canadian stroke best practice recommendations: Stroke rehabilitation practice guidelines update 2015. *International Journal of Stroke*, 11 (4):459–484, jun 2016. ISSN 1747-4930. doi: 10.1177/1747493016643553.

[109] R. Heine, K. Manal, and T. S. Buchanan. Using Hill-type muscle models and EMG data in a forward dynamic analysis of joint moment. *Journal of Mechanics in Medicine and Biology*, 03(02):169–186, jun 2003. ISSN 0219-5194. doi: 10.1142/S0219519403000727.

[110] A. Hernández Arieta, R. Kato, W. Yu, and H. Yokoi. The man-machine interaction: The influence of artificial intelligence on rehabilitation robotics. In *50 Years of Artificial Intelligence*, pages 221–231. Springer Berlin Heidelberg, Berlin, Heidelberg, 2007. doi: 10.1007/978-3-540-77296-5_21.

[111] W. Herzog. Sensitivity of muscle force estimations to changes in muscle input parameters using nonlinear optimization approaches. *Journal of Biomechanical Engineering*, 114:267, 1992. ISSN 01480731. doi: 10.1115/1.2891382.

[112] W. Herzog, K. Powers, K. Johnston, and M. Duvall. A new paradigm for muscle contraction. *Frontiers in Physiology*, 6(MAY):1–11, 2015. ISSN 1664042X. doi: 10.3389/fphys.2015.00174.

[113] S. Hesse, H. Schmidt, C. Werner, and A. Bardeleben. Upper and lower extremity robotic devices for rehabilitation and for studying motor control. *Current Opinion in Neurology*, 16:705–710, 2003. ISSN 1350-7540. doi: 10.1097/00019052-200312000-00010.

[114] J. Hicks and C. Dembia. How static optimization works - OpenSim Documentation - Confluence [Available: http://simtk-confluence.stanford.edu:8080/display/OpenSim/How+Static+Optimization+Works], 2017.

[115] A. Hill. The heat of shortening and the dynamic constants of muscle. *Proceedings of the Royal Society of London. Series B, Biological Sciences*, 126(843):136–195, 1938.

[116] B. Hingtgen, J. R. McGuire, M. Wang, and G. F. Harris. An upper extremity kinematic model for evaluation of hemiparetic stroke. *Journal of Biomechanics*, 39: 681–8, 2006. ISSN 0021-9290. doi: 10.1016/j.jbiomech.2005.01.008.

[117] N. Hogan. Impedance control: An approach to manipulation: Part I - Theory. *Journal of Dynamic Systems, Measurement, and Control*, 107(1):1, 1985. ISSN 00220434. doi: 10.1115/1.3140702.

[118] N. Hogan. Impedance control: An approach to manipulation: Part III - Applications. *Journal of Dynamic Systems, Measurement, and Control*, 107(1):17, 1985. ISSN 00220434. doi: 10.1115/1.3140701.

[119] C. Högfors, G. Sigholm, and P. Herberts. Biomechanical model of the human shoulder–I. Elements. *Journal of Biomechanics*, 20:157–166, 1987. ISSN 00219290. doi: 10.1016/0021-9290(87)90307-1.

[120] C. Högfors, B. Peterson, G. Sigholm, and P. Herberts. Biomechanical model of the human shoulder joint - II. The shoulder rhythm. *Journal of Biomechanics*, 24:699–709, 1991. ISSN 00219290. doi: 10.1016/0021-9290(91)90334-J.

[121] C. Högfors, D. Karlsson, and B. Peterson. Structure and internal consistency of a shoulder model. *Journal of Biomechanics*, 28(7):767–77, jul 1995. ISSN 0021-9290.

[122] K. R. S. Holzbaur, W. M. Murray, and S. L. Delp. A model of the upper extremity for simulating musculoskeletal surgery and analyzing neuromuscular control. *Annals of Biomedical Engineering*, 33(6):829–840, jun 2005. ISSN 0090-6964. doi: 10.1007/s10439-005-3320-7.

[123] M. G. Hoy, F. E. Zajac, and M. E. Gordon. A musculoskeletal model of the human lower extremity: The effect of muscle, tendon, and moment arm on the moment-angle relationship of musculotendon actuators at the hip, knee, and ankle. *Journal of Biomechanics*, 23(2):157–69, jan 1990. ISSN 0021-9290.

[124] X. Hu and R. K. Y. Tong. FES in rehabilitation robotics. In *2014 IEEE 19th International Functional Electrical Stimulation Society Annual Conference (IFESS)*, pages 1–3. IEEE, sep 2014. ISBN 978-1-4799-6483-3. doi: 10.1109/IFESS.2014.7036730.

[125] R. Hughes and K. An. Monte Carlo simulation of a planar shoulder model. *Medical and Biological Engineering and Computing*, 35:544–548, 1997.

[126] R. Huq, E. Lu, R. Wang, and A. Mihailidis. Development of a portable robot and graphical user interface for haptic rehabilitation exercise. In *2012 4th IEEE RAS & EMBS International Conference on Biomedical Robotics and Biomechatronics (BioRob)*, pages 1451–1457, Rome, Italy, jun 2012. IEEE. ISBN 978-1-4577-1200-5. doi: 10.1109/BioRob.2012.6290273.

[127] S. Hussain, S. Q. Xie, and P. K. Jamwal. Adaptive impedance control of a robotic orthosis for gait rehabilitation. *IEEE Transactions on Cybernetics*, 43(3):1025–1034, jun 2013. ISSN 2168-2267. doi: 10.1109/TSMCB.2012.2222374.

[128] N. L. Ifejika-Jones and A. M. Barrett. Rehabilitation-emerging technologies, innovative therapies, and future objectives. *Neurotherapeutics*, 8(3):452–462, jul 2011. ISSN 19337213. doi: 10.1007/s13311-011-0057-x.

[129] A. Jackson, P. Culmer, S. Makower, M. Levesley, R. Richardson, A. Cozens, M. M. Williams, and B. Bhakta. Initial patient testing of iPAM - a robotic system for Stroke rehabilitation. In *2007 IEEE 10th International Conference on Rehabilitation Robotics (ICORR)*, pages 250–256. IEEE, jun 2007. ISBN 978-1-4244-1319-5. doi: 10.1109/ICORR.2007.4428435.

[130] D. A. Jacobs and T. Uchida. Thelen 2003 muscle model - OpenSim Documentation - Confluence, 2013.

[131] T. Jeji. Spinal cord injury, [Available: http://onf.org/our-programs/spinal-cord-injury], 2013.

[132] R. H. Jensen and D. T. Davy. An investigation of muscle lines of action about the hip: A centroid line approach vs the straight line approach. *Journal of Biomechanics*, 8:103–110, 1975. ISSN 00219290. doi: 10.1016/0021-9290(75)90090-1.

[133] Z.-H. Jiang, T. Ishida, and M. Sunawada. Neural network aided dynamic parameter identification of robot manipulators. In *2006 IEEE International Conference on Systems, Man and Cybernetics*, pages 3298–3303. IEEE, oct 2006. ISBN 1-4244-0099-6. doi: 10.1109/ICSMC.2006.384627.

[134] C. T. John, F. C. Anderson, J. S. Higginson, and S. L. Delp. Stabilisation of walking by intrinsic muscle properties revealed in a three-dimensional muscle-driven simulation. *Computer Methods in Biomechanics and Biomedical Engineering*, 16(4):451–62, apr 2013. ISSN 1476-8259. doi: 10.1080/10255842.2011.627560.

[135] M. J. Johnson, H. F. M. Van der Loos, C. G. Burgar, P. Shor, and L. J. Leifer. Design and evaluation of Driver's SEAT: A car steering simulation environment for upper limb stroke therapy. *Robotica*, 21(01):13–23, jan 2003. ISSN 0263-5747. doi: 10.1017/S0263574702004599.

[136] M. J. Johnson, K. J. Wisneski, J. Anderson, D. Nathan, and R. O. Smith. Development of ADLER: The activities of daily living exercise robot. In *Proceedings of the First IEEE/RAS-EMBS International Conference on Biomedical Robotics and Biomechatronics, 2006, BioRob 2006*, volume 2006, pages 881–886. IEEE, 2006. ISBN 1424400406. doi: 10.1109/BIOROB.2006.1639202.

[137] H. S. Jorgensen, H. Nakayama, H. O. Raaschou, J. Vive-Larsen, M. Stoier, and T. S. Olsen. Outcome and time course of recovery in stroke. Part II: Time course of recovery. The Copenhagen Stroke Study. *Archives of Physical Medicine and Rehabilitation*, 76:406–412, 1995. ISSN 00039993. doi: 10.1016/S0003-9993(95)80568-0.

[138] L. Kahn, W. Rymer, and D. Reinkensmeyer. Adaptive assistance for guided force training in chronic stroke. In *The 26th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, volume 3, pages 2722–2725. IEEE, 2004. ISBN 0-7803-8439-3. doi: 10.1109/IEMBS.2004.1403780.

[139] L. E. Kahn, M. L. Zygman, W. Z. Rymer, and D. J. Reinkensmeyer. Robot-assisted reaching exercise promotes arm movement recovery in chronic hemiparetic stroke: A randomized controlled pilot study. *Journal of NeuroEngineering and Rehabilitation*, 3:12, jun 2006. ISSN 1743-0003. doi: 10.1186/1743-0003-3-12.

[140] N. M. Kapadia, M. K. Nagai, V. Zivanovic, J. Bernstein, J. Woodhouse, P. Rumney, and M. R. Popovic. Functional electrical stimulation therapy for recovery of reaching and grasping in severe chronic pediatric stroke patients. *Journal of Child Neurology*, 29(4):493–499, apr 2014. ISSN 0883-0738. doi: 10.1177/0883073813484088.

[141] D. Karlsson and B. Peterson. Towards a model for force predictions in the human shoulder. *Journal of Biomechanics*, 25(2):189–99, feb 1992. ISSN 0021-9290.

[142] A. Kecskeméthy and A. Weinberg. An improved elasto-kinematic model of the human forearm for biofidelic medical diagnosis. *Multibody System Dynamics*, 14:1–21, 2005.

[143] R. Kelc. Zygote Body: A new interactive 3-dimensional didactical tool for teaching anatomy. *WebmedCentral ANATOMY*, 3(1):WMC002903, 2012. ISSN 20461690. doi: 10.9754/journal.wmc.2012.002903.

[144] H. Kim, L. M. Miller, I. Fedulow, M. Simkins, G. M. Abrams, N. Byl, and J. Rosen. Kinematic data analysis for post-stroke patients following bilateral versus unilateral rehabilitation with an upper limb wearable robotic system. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 21(2):153–164, mar 2013. ISSN 15344320. doi: 10.1109/TNSRE.2012.2207462.

[145] S. Kim, D.-M. Kim, and S.-W. Chae. Musculoskeletal upper limb modeling with muscle activation for flexible body simulation. *International Journal of Precision Engineering and Manufacturing*, 10(4):123–129, oct 2009. ISSN 1229-8557. doi: 10.1007/s12541-009-0080-6.

[146] V. Klamroth-Marganska, J. Blanco, K. Campen, A. Curt, V. Dietz, T. Ettlin, M. Felder, B. Fellinghauer, M. Guidali, A. Kollmar, A. Luft, T. Nef, C. Schuster-Amft, W. Stahel, and R. Riener. Three-dimensional, task-specific robot therapy of the arm after stroke: A multicentre, parallel-group randomised trial. *The Lancet Neurology*, 13(2):159–166, feb 2014. ISSN 14744422. doi: 10.1016/S1474-4422(13)70305-3.

[147] M. D. Klein Breteler, C. W. Spoor, and F. C. Van der Helm. Measuring muscle and joint geometry parameters of a shoulder for modeling purposes. *Journal of Biomechanics*, 32(11):1191–7, nov 1999. ISSN 0021-9290.

[148] S. Kousidou, N. G. Tsagarakis, C. Smith, and D. G. Caldwell. Task-orientated biofeedback system for the rehabilitation of the upper limb. In *2007 IEEE 10th International Conference on Rehabilitation Robotics (ICORR)*, pages 376–384. IEEE, jun 2007. ISBN 978-1-4244-1319-5. doi: 10.1109/ICORR.2007.4428453.

[149] J. Kowalczewski and A. Prochazka. Technology improves upper extremity rehabilitation. *Progress in Brain Research*, 192:147–159, 2011. ISSN 00796123. doi: 10.1016/B978-0-444-53355-5.00010-5.

[150] H. I. Krebs and N. Hogan. Robotic therapy: The tipping point. *American journal of physical medicine & rehabilitation*, 91(11 Suppl 3):S290–7, nov 2012. ISSN 1537-7385. doi: 10.1097/PHM.0b013e31826bcd80.

[151] H. I. Krebs and B. T. Volpe. Rehabilitation robotics. *Handbook of Clinical Neurology*, 110:283–294, 2013. ISSN 00729752. doi: 10.1016/B978-0-444-52901-5.00023-X.

[152] H. I. Krebs, J. J. Palazzolo, L. Dipietro, M. Ferraro, J. Krol, K. Rannekleiv, B. T. Volpe, and N. Hogan. Rehabilitation robotics: Performance-based progressive robot-assisted therapy. *Autonomous Robots*, 15(1):7–20, 2003. ISSN 09295593. doi: 10.1023/A:1024494031121.

[153] H. Krueger, P. Lindsay, R. Cote, M. K. Kapral, J. Kaczorowski, and M. D. Hill. Cost avoidance associated with optimal stroke care in canada. *Stroke*, 43(8):2198–2206, 2012. ISSN 00392499. doi: 10.1161/STROKEAHA.111.646091.

[154] B. Laursen, K. Søgaard, and G. Sjøgaard. Biomechanical model predicting electromyographic activity in three shoulder muscles from 3D kinematics and external forces during cleaning work. *Clinical Biomechanics*, 18(4):287–295, may 2003. ISSN 02680033. doi: 10.1016/S0268-0033(03)00004-4.

[155] M. K. Lebiedowska. Dynamic properties of human limb segments. In W. Karwowski, editor, *International Encyclopedia of Ergonomics and Human Factors*, page 317. CRC Press, 2nd edition, 2006. ISBN 041530430X.

[156] C. S. G. Lee and B. H. Lee. Resolved motion adaptive control for mechanical manipulators. *Journal of Dynamic Systems, Measurement, and Control*, 106(2):134, 1984. ISSN 00220434. doi: 10.1115/1.3143727.

[157] D. Leonardis, M. Barsotti, C. Loconsole, M. Solazzi, M. Troncossi, C. Mazzotti, V. P. Castelli, C. Procopio, G. Lamola, C. Chisari, M. Bergamasco, and A. Frisoli. An EMG-controlled robotic hand exoskeleton for bilateral rehabilitation. *IEEE Transactions on Haptics*, 8(2):140–151, apr 2015. ISSN 1939-1412. doi: 10.1109/TOH.2015.2417570.

[158] R. L. Lieber. Skeletal muscle architecture. *Journal of Hand Therapy*, 6(2):105–113, apr 1993. ISSN 08941130. doi: 10.1016/S0894-1130(12)80291-2.

[159] F. Liu, W. W. Hager, and A. V. Rao. Adaptive mesh refinement method for optimal control using nonsmoothness detection and mesh size reduction. *Journal of the Franklin Institute*, 352(10):4081–4106, 2015. ISSN 00160032. doi: 10.1016/j.jfranklin.2015.05.028.

[160] G. Liu and A. Goldenberg. Robust hybrid impedance control of robot manipulators. In *Proceedings of the 1991 IEEE Intemational Conference on Robotics and Automation*, number April, pages 1–6, 1991.

[161] J. Liu, J. Shi, L. C. Fitton, R. Phillips, P. O'Higgins, and M. J. Fagan. The application of muscle wrapping to voxel-based finite element models of skeletal structures. *Biomechanics and Modeling in Mechanobiology*, 11(1-2):35–47, jan 2012. ISSN 16177959. doi: 10.1007/s10237-011-0291-5.

[162] Y. Liu, C. Li, L. Ji, S. Bi, X. Zhang, J. Huo, and R. Ji. Development and implementation of an end-effector upper limb rehabilitation robot for hemiplegic patients with line and circle tracking training. *Journal of Healthcare Engineering*, 2017:1–11, jun 2017. ISSN 2040-2295. doi: 10.1155/2017/4931217.

[163] L. Ljung. *System Identification: Theory for User*. PTR Prentice Hall Information and System Sciences Series, 2nd edition, 1987. ISBN 0138816409. doi: 10.1016/0005-1098(89)90019-8.

[164] A. C. Lo, P. D. Guarino, L. G. Richards, J. K. Haselkorn, G. F. Wittenberg, D. G. Federman, R. J. Ringer, T. H. Wagner, H. I. Krebs, B. T. Volpe, C. T. Bever, D. M. Bravata, P. W. Duncan, B. H. Corn, A. D. Maffucci, S. E. Nadeau, S. S. Conroy, J. M. Powell, G. D. Huang, and P. Peduzzi. Robot-assisted therapy for long-term upper-limb impairment after stroke. *The New England Journal of Medicine*, 362:1772–1783, 2010. ISSN 0028-4793. doi: 10.1056/NEJMoa0911341.

[165] C. Loconsole, R. Bartalucci, A. Frisoli, and M. Bergamasco. A new gaze-tracking guidance mode for upper limb robot-aided neurorehabilitation. In *2011 IEEE World Haptics Conference*, pages 185–190. IEEE, jun 2011. ISBN 978-1-4577-0299-0. doi: 10.1109/WHC.2011.5945483.

[166] C. Loconsole, S. Dettori, A. Frisoli, C. A. Avizzano, and M. Bergamasco. An EMG-based approach for on-line predicted torque control in robotic-assisted rehabilitation. In *2014 IEEE Haptics Symposium (HAPTICS)*, pages 181–186. IEEE, feb 2014. ISBN 978-1-4799-3131-6. doi: 10.1109/HAPTICS.2014.6775452.

[167] W.-Y. G. Louie, S. Mohamed, and G. Nejat. Human-robot interaction for rehabilitation robots. In P. Encarnacao and A. M. Cook, editors, *Robotic Assistive Technologies: Principles and Practice*, pages 25–70. Taylor & Francis Group, CRC Press, mar 2017. doi: 10.1201/9781315368788-3.

[168] R. C. V. Loureiro and W. S. Harwin. Reach & grasp therapy: Design and control of a 9-DOF robotic neuro-rehabilitation system. In *2007 IEEE 10th International Conference on Rehabilitation Robotics (ICORR)*, pages 757–763. IEEE, jun 2007. ISBN 1424413206. doi: 10.1109/ICORR.2007.4428510.

[169] E. C. Lu, R. Wang, R. Huq, D. Gardner, P. Karam, K. Zabjek, D. Hebert, J. Boger, and A. Mihailidis. Development of a robotic device for upper limb stroke rehabilitation: A user-centered design approach. *Paladyn, Journal of Behavioral Robotics*, 2 (4):176–184, jan 2011. ISSN 2081-4836. doi: 10.2478/s13230-012-0009-0.

[170] P. Maciejasz, J. Eschweiler, K. Gerlach-Hahn, A. Jansen-Troy, and S. Leonhardt. A survey on robotic devices for upper limb rehabilitation. *Journal of NeuroEngineering and Rehabilitation*, 11(1):3, jan 2014. ISSN 1743-0003. doi: 10.1186/1743-0003-11-3.

[171] B. Maldonado, M. Mendoza, I. Bonilla, and I. Reyna-Gutierrez. Stiffness-based tuning of an adaptive impedance controller for robot-assisted rehabilitation of upper limbs. In *2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 3578–3581. IEEE, aug 2015. ISBN 978-1-4244-9271-8. doi: 10.1109/EMBC.2015.7319166.

[172] Y. Mao, X. Jin, G. Gera Dutta, J. P. Scholz, and S. K. Agrawal. Human movement training with a cable driven arm exoskeleton (CAREX). *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 23(1):84–92, jan 2015. ISSN 1534-4320. doi: 10.1109/TNSRE.2014.2329018.

[173] L. Marchal-Crespo and D. J. Reinkensmeyer. Review of control strategies for robotic movement training after neurologic injury. *Journal of NeuroEngineering and Rehabilitation*, 6(1):20, jan 2009. ISSN 1743-0003. doi: 10.1186/1743-0003-6-20.

[174] L. Marcucci, C. Reggiani, A. N. Natali, and P. G. Pavan. From single muscle fiber to whole muscle mechanics: A finite element model of a muscle bundle with fast and slow fibers. *Biomechanics and Modeling in Mechanobiology*, jun 2017. ISSN 1617-7959. doi: 10.1007/s10237-017-0922-6.

[175] S. P. Marsden and D. C. Swailes. A novel approach to the prediction of musculotendon paths. *Proceedings of the Institution of Mechanical Engineers. Part*

H, Journal of Engineering in Medicine, 222:51–61, 2008. ISSN 0954-4119. doi: 10.1243/09544119JEIM297.

[176] S. Martelli, D. Calvetti, E. Somersalo, and M. Viceconti. Stochastic modelling of muscle recruitment during activity. *Interface Focus*, 5(2):20140094, apr 2015. ISSN 2042-8898. doi: 10.1098/rsfs.2014.0094.

[177] R. Matias, C. Andrade, and A. P. Veloso. A transformation method to estimate muscle attachments based on three bony landmarks. *Journal of Biomechanics*, 42: 331–335, 2009. ISSN 00219290. doi: 10.1016/j.jbiomech.2008.11.027.

[178] S. Mazzoleni, P. Sale, M. Tiboni, M. Franceschini, M. C. Carrozza, and F. Posteraro. Upper limb robot-assisted therapy in chronic and subacute stroke patients: A kinematic analysis. *American Journal of Physical Medicine & Rehabilitation / Association of Academic Physiatrists*, 92:e26–37, 2013. ISSN 1537-7385. doi: 10.1097/PHM.0b013e3182a1e852.

[179] J. McCabe, M. Monkiewicz, J. Holcomb, S. Pundik, and J. J. Daly. Comparison of robotics, functional electrical stimulation, and motor learning methods for treatment of persistent upper extremity dysfunction after stroke: A randomized controlled trial. *Archives of Physical Medicine and Rehabilitation*, 96(6):981–990, jun 2015. ISSN 1532821X. doi: 10.1016/j.apmr.2014.10.022.

[180] N. Mehrabi. *Dynamics and model-based control of electric power steering systems*. Phd thesis, University of Waterloo, Waterloo, Ontario, Canada, 2014.

[181] N. Mehrabi, R. Sharif Razavian, and J. McPhee. A three-dimensional musculoskeletal driver model to study steering tasks. In *Proceedings of the ASME 2013 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference IDETC/CIE 2013*, pages 1–8, 2013.

[182] N. Mehrabi, S. Tajeddin, N. Lashgarian Azad, and J. McPhee. Application of Newton/GMRES method to nonlinear model-predictive control of functional electrical stimulation. In *3rd International Conference on Control, Dynamic Systems, and and Robotics (CDSR)*, Ottawa, Canada, 2016. Avestia.

[183] N. Mehrabi, R. Sharif Razavian, B. Ghannadi, and J. McPhee. Predictive simulation of reaching moving targets using nonlinear model predictive control. *Frontiers in Computational Neuroscience*, 10:143, jan 2017. ISSN 1662-5188. doi: 10.3389/fncom.2016.00143.

[184] J. Mehrholz, A. Hädrich, T. Platz, J. Kugler, and M. Pohl. Electromechanical and robot-assisted arm training for improving generic activities of daily living, arm function, and arm muscle strength after stroke. In J. Mehrholz, editor, *Cochrane Database of Systematic Reviews*, volume 6, page CD006876. John Wiley & Sons, Ltd, Chichester, UK, jun 2012. doi: 10.1002/14651858.CD006876.pub3.

[185] L. L. Menegaldo, A. d. T. Fleury, and H. I. Weber. A 'cheap' optimal control approach to estimate muscle forces in musculoskeletal systems. *Journal of Biomechanics*, 39: 1787–1795, 2006. ISSN 00219290. doi: 10.1016/j.jbiomech.2005.05.029.

[186] M. Mihelj, T. Nef, and R. Riener. A novel paradigm for patient-cooperative control of upper-limb rehabilitation robots. *Advanced Robotics*, 21(8):843–867, jan 2007. ISSN 0169-1864. doi: 10.1163/156855307780851975.

[187] L. M. Miller and J. Rosen. Comparison of multi-sensor admittance control in joint space and task space for a seven degree of freedom upper limb exoskeleton. In *2010 3rd IEEE RAS and EMBS International Conference on Biomedical Robotics and Biomechatronics (BioRob)*, pages 70–75. IEEE, sep 2010. ISBN 9781424477081. doi: 10.1109/BIOROB.2010.5628069.

[188] A. Montagner, A. Frisoli, L. Borelli, C. Procopio, M. Bergamasco, M. C. Carboncini, and B. Rossi. A pilot clinical study on robotic assisted rehabilitation in VR with an arm exoskeleton device. In *2007 International Conference on Virtual Rehabilitation (ICVR)*, pages 57–64. IEEE, sep 2007. ISBN 1424412048. doi: 10.1109/ICVR.2007.4362131.

[189] P. Morasso. Spatial control of arm movements. *Experimental Brain Research*, 42(2), apr 1981. ISSN 0014-4819. doi: 10.1007/BF00236911.

[190] S. L. Morris, K. J. Dodd, and M. E. Morris. Outcomes of progressive resistance strength training following stroke: A systematic review. *Clinical Rehabilitation*, 18 (1):27–39, feb 2004. ISSN 0269-2155. doi: 10.1191/0269215504cr699oa.

[191] M. M. Morrow, J. W. Rankin, R. R. Neptune, and K. R. Kaufman. A comparison of static and dynamic optimization muscle force predictions during wheelchair propulsion. *Journal of Biomechanics*, 47(14):3459–3465, nov 2014. ISSN 1873-2380. doi: 10.1016/j.jbiomech.2014.09.013.

[192] S. Moughamir, A. Deneve, J. Zaytoon, and L. Afilal. Hybrid force/impedance control for the robotized rehabilitation of the upper limbs. In P. Zítek, ed-

itor, *Proceedings of 16th IFAC World Congress*, pages 2169–2169, jul 2005. doi: 10.3182/20050703-6-CZ-1902.02170.

[193] A. J. Nagengast, D. A. Braun, and D. M. Wolpert. Optimal control predicts human performance on objects with internal degrees of freedom. *PLoS Computational Biology*, 5(6):e1000419, jun 2009. ISSN 1553734X. doi: 10.1371/journal.pcbi.1000419.

[194] M. V. Narici, N. Maffulli, and C. N. Maganaris. Ageing of human muscles and tendons. *Disability and Rehabilitation*, 30(20-22):1548–54, jan 2008. ISSN 0963-8288. doi: 10.1080/09638280701831058.

[195] J. Niemi, H. Nieminen, E. P. Takala, and E. Viikari-Juntura. A static shoulder model based on a time-dependent criterion for load sharing between synergistic muscles. *Journal of Biomechanics*, 29(4):451–60, apr 1996. ISSN 0021-9290.

[196] H. Nieminen, J. Niemi, E. P. Takala, and E. Viikari-Juntura. Load-sharing patterns in the shoulder during isometric flexion tasks. *Journal of Biomechanics*, 28(5):555–66, may 1995. ISSN 0021-9290.

[197] E. Nijhof and E. Kouwenhoven. Simulation of multijoint arm movements. In J. M. Winters and P. E. Crago, editors, *Biomechanics and Neural Control of Posture and Movement*, chapter 28, pages 363–372. Springer-Verlag, 2000. ISBN 0-387-94974-7.

[198] A. A. Nikooyan, H. E. J. Veeger, E. K. J. Chadwick, M. Praagman, and F. C. T. van der Helm. Development of a comprehensive musculoskeletal model of the shoulder and elbow. *Medical & biological engineering & computing*, 49(12):1425–35, dec 2011. ISSN 1741-0444. doi: 10.1007/s11517-011-0839-7.

[199] M. Nordin and P.-O. Gutman. Controlling mechanical systems with backlash - A survey. *Automatica*, 38(10):1633–1649, oct 2002. ISSN 00051098. doi: 10.1016/S0005-1098(02)00047-X.

[200] V. Norman Gerum and J. McPhee. Comparison of cylindrical wrapping geometries to via points for modeling muscle paths in the estimation of sit-to-stand muscle forces. In *ASME 2013 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, 2013.

[201] D. Novak and R. Riener. Enhancing patient freedom in rehabilitation robotics using gaze-based intention detection. In *2013 IEEE 13th International Conference on Rehabilitation Robotics (ICORR)*, pages 1–6. IEEE, jun 2013. ISBN 978-1-4673-6024-1. doi: 10.1109/ICORR.2013.6650507.

[202] E. K. Nyarko and R. Scitovski. Solving the parameter identification problem of mathematical models using genetic algorithms. *Applied Mathematics and Computation*, 153(3):651–658, 2004. ISSN 00963003. doi: 10.1016/S0096-3003(03)00661-1.

[203] T. Ohtsuka. A continuation/GMRES method for fast computation of nonlinear receding horizon control. *Automatica*, 40(4):563–574, apr 2004. ISSN 00051098. doi: 10.1016/j.automatica.2003.11.005.

[204] H. Olsson, K. Åström, C. Canudas de Wit, M. Gäfvert, and P. Lischinsky. Friction models and friction compensation. *European Journal of Control*, 4(3):176–195, jan 1998. ISSN 09473580. doi: 10.1016/S0947-3580(98)70113-X.

[205] C. Ott, R. Mukherjee, and Y. Nakamura. Unified Impedance and Admittance Control. *2010 IEEE International Conference on Robotics and Automation (ICRA)*, pages 554–561, may 2010. doi: 10.1109/ROBOT.2010.5509861.

[206] L. Oujamaa, I. Relave, J. Froger, D. Mottet, and J. Y. Pelissier. Rehabilitation of arm function after stroke. Literature review. *Annals of Physical and Rehabilitation Medicine*, 52:269–293, 2009. ISSN 18770657. doi: 10.1016/j.rehab.2008.10.003.

[207] S. Pal, J. E. Langenderfer, J. Q. Stowe, P. J. Laz, A. J. Petrella, and P. J. Rullkoetter. Probabilistic modeling of knee muscle moment arms: Effects of methods, origin-insertion, and kinematic variability. *Annals of Biomedical Engineering*, 35:1632–1642, 2007. ISSN 00906964. doi: 10.1007/s10439-007-9334-6.

[208] M. G. Pandy, F. C. Anderson, and D. G. Hull. A parameter optimization approach for the optimal control of large-scale musculoskeletal systems. *Journal of Biomechanical Engineering*, 114:450–460, 1992. ISSN 01480731. doi: 10.1115/1.2894094.

[209] A. Patrick and K. Abdel-Malek. A musculoskeletal model of the upper limb for real time interaction. *SAE Technical Paper*, (2007-01-2488), jun 2007. doi: 10.4271/2007-01-2488.

[210] C. Patten, J. Dozono, S. Schmidt, M. Jue, and P. Lum. Combined functional task practice and dynamic high intensity resistance training promotes recovery of upper-extremity motor function in post-stroke hemiparesis: A case study. *Journal of Neurologic Physical Therapy (JNPT)*, 30(3):99–115, sep 2006. ISSN 1557-0576.

[211] M. A. Patterson and A. V. Rao. GPOPS-II: A MATLAB software for solving multiple-phase optimal control problems using hp-adaptive Gaussian quadrature collocation methods and sparse nonlinear programming. *ACM Transactions on Mathematical Software*, 41(1):1–37, oct 2014. ISSN 00983500. doi: 10.1145/2558904.

[212] J. Patton, G. Dawe, C. Scharver, F. Mussa-Ivaldi, and R. Kenyon. Robotics and virtual reality: The development of a life-sized 3-D system for the rehabilitation of motor function. In *The 26th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, volume 4, pages 4840–4843. IEEE, 2004. ISBN 0-7803-8439-3. doi: 10.1109/IEMBS.2004.1404339.

[213] J. L. Patton, M. Kovic, and F. A. Mussa-Ivaldi. Custom-designed haptic training for restoring reaching ability to individuals with poststroke hemiparesis. *Journal of rehabilitation research and development*, 43(5):643–56, 2006. ISSN 1938-1352.

[214] J. L. Patton, M. E. Stoykov, M. Kovic, and F. A. Mussa-Ivaldi. Evaluation of robotic training forces that either enhance or reduce error in chronic hemiparetic stroke survivors. *Experimental Brain Research*, 168(3):368–383, jan 2006. ISSN 0014-4819. doi: 10.1007/s00221-005-0097-8.

[215] R. Pérez-Rodríguez, C. Rodríguez, Ú. Costa, C. Cáceres, J. M. Tormos, J. Medina, and E. J. Gómez. Anticipatory assistance-as-needed control algorithm for a multijoint upper limb robotic orthosis in physical neurorehabilitation. *Expert Systems with Applications*, 41(8):3922–3934, jun 2014. ISSN 09574174. doi: 10.1016/j.eswa.2013.11.047.

[216] L. Pignolo, G. Dolce, G. Basta, L. F. Lucca, S. Serra, and W. G. Sannita. Upper limb rehabilitation after stroke: ARAMIS a robo-mechatronic innovative approach and prototype. In *2012 4th IEEE RAS & EMBS International Conference on Biomedical Robotics and Biomechatronics (BioRob)*, pages 1410–1414. IEEE, jun 2012. ISBN 978-1-4577-1200-5. doi: 10.1109/BioRob.2012.6290868.

[217] P. Poli, G. Morone, G. Rosati, and S. Masiero. Robotic technologies and rehabilitation: New tools for stroke patients' therapy. *BioMed Research International*, 2013: 153872, 2013. ISSN 23146133. doi: 10.1155/2013/153872.

[218] T. Proietti, N. Jarrasse, A. Roby-Brami, and G. Morel. Adaptive control of a robotic exoskeleton for neurorehabilitation. In *2015 7th International IEEE/EMBS Conference on Neural Engineering (NER)*, pages 803–806. IEEE, apr 2015. ISBN 978-1-4673-6389-1. doi: 10.1109/NER.2015.7146745.

[219] T. Proietti, V. Crocher, A. Roby-Brami, and N. Jarrasse. Upper-limb robotic exoskeletons for neurorehabilitation: A review on control strategies. *IEEE Reviews in Biomedical Engineering*, 9:4–14, 2016. ISSN 1937-3333. doi: 10.1109/RBME.2016.2552201.

[220] C. Qin, H. Zhang, and Y. Luo. Online optimal tracking control of continuous-time linear systems with unknown dynamics by using adaptive dynamic programming. *International Journal of Control*, 87(5):1000–1009, may 2014. ISSN 0020-7179. doi: 10.1080/00207179.2013.863432.

[221] C. Quental, J. Folgado, J. Ambrósio, and J. Monteiro. A multibody biomechanical model of the upper limb including the shoulder girdle. *Multibody System Dynamics*, 28(1-2):83–108, jan 2012. ISSN 1384-5640. doi: 10.1007/s11044-011-9297-0.

[222] M. H. Rahman, C. Ochoa-Luna, and M. Saad. EMG based control of a robotic exoskeleton for shoulder and elbow motion assist. *Journal of Automation and Control Engineering*, pages 270–276, 2015. ISSN 23013702. doi: 10.12720/joace.3.4.270-276.

[223] M. Raison, C. Detrembleur, P. Fisette, and J.-c. Samin. Assessment of antagonistic muscle forces during forearm flexion/extension. In K. Arczewski, W. Blajer, J. Fraczek, and M. Wojtyra, editors, *Multibody Dynamics: Computational Methods and Applications*, volume 23 of *Computational Methods in Applied Sciences*, pages 215–238. Springer Netherlands, Dordrecht, 2011. ISBN 978-90-481-9970-9. doi: 10.1007/978-90-481-9971-6.

[224] J. Randall Flanagan and D. Ostry. Trajectories of human multi-joint arm movements: Evidence of joint level planning. In V. Hayward and O. Khatib, editors, *Experimental Robotics I SE - 35*, volume 139 of *Lecture Notes in Control and Information Sciences*, pages 594–613. Springer Berlin Heidelberg, 1990. ISBN 978-3-540-52182-2. doi: 10.1007/BFb0042544.

[225] J. W. Rankin and R. R. Neptune. Musculotendon lengths and moment arms for a three-dimensional upper-extremity model. *Journal of Biomechanics*, 45(9):1739–44, jun 2012. ISSN 1873-2380. doi: 10.1016/j.jbiomech.2012.03.010.

[226] A. V. Rao. A survey of numerical methods for optimal control. *Advances in the Astronautical Sciences*, 135(1):497–528, 2009.

[227] J. Rasmussen, M. Damsgaard, and S. T. Christensen. Inverse-inverse dynamics simulation of musculo-skeletal systems. In *Proceedings of the 12th Conference of the European Society of Biomechanics Royal Acedemy of Medicine in Ireland*. Royal Acedemy of Medicine in Ireland, 2000.

[228] C. Redl, M. Gfoehler, and M. G. Pandy. Sensitivity of muscle force estimates to variations in muscle-tendon properties. *Human movement science*, 26(2):306–19, apr 2007. ISSN 0167-9457. doi: 10.1016/j.humov.2007.01.008.

[229] N. B. Reese and W. D. Bandy. *Joint Range of Motion and Muscle Length Testing*. Elsevier, 2nd edition, 2010. ISBN 9781416058847.

[230] D. J. Reinkensmeyer. Robotic assistance for upper extremity training after stroke. In *Studies in Health Technology and Informatics*, volume 145, pages 25–39, 2009. ISBN 9781607500186. doi: 10.3233/978-1-60750-018-6-25.

[231] D. J. Reinkensmeyer, L. E. Kahn, M. Averbuch, A. McKenna-Cole, B. D. Schmit, and W. Z. Rymer. Understanding and treating arm movement impairment after chronic brain injury: Progress with the ARM guide. *Journal of rehabilitation research and development*, 37(6):653–62, 2000. ISSN 0748-7711.

[232] D. J. Reinkensmeyer, P. Lum, and J. M. Winters. Emerging technologies for improving access to movement therapy following neurologic injury. In J. Winters, C. Robinson, R. Simpson, and G. Vanderheiden, editors, *Emerging and Accessible Telecommunications, Information and Healthcare Technologies Engineering Challenges in Enabling Universal Access*. IEEE Press, 2002.

[233] D. J. Reinkensmeyer, E. Burdet, M. Casadio, J. W. Krakauer, G. Kwakkel, C. E. Lang, S. P. Swinnen, N. S. Ward, and N. Schweighofer. Computational neurorehabilitation: Modeling plasticity and learning to predict recovery. *Journal of NeuroEngineering and Rehabilitation*, 13(1):42, dec 2016. ISSN 1743-0003. doi: 10.1186/s12984-016-0148-3.

[234] Y. Ren, S. Hoon Kang, H.-S. Park, Y.-N. Wu, and L.-Q. Zhang. Developing a multijoint upper limb exoskeleton robot for diagnosis, therapy, and outcome evaluation in neurorehabilitation. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 21(3):490–499, may 2013. ISSN 1534-4320. doi: 10.1109/TNSRE.2012.2225073.

[235] C. L. Richards and F. Malouin. Stroke rehabilitation: Clinical picture, assessment, and therapeutic challenge. *Progress in Brain Research*, 218:253–280, 2015. ISSN 00796123. doi: 10.1016/bs.pbr.2015.01.003.

[236] R. Riener, L. Lunenburger, S. Jezernik, M. Anderschitz, G. Colombo, and V. Dietz. Patient-cooperative strategies for robot-aided treadmill training: First experimental results. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 13 (3):380–394, sep 2005. ISSN 1534-4320. doi: 10.1109/TNSRE.2005.848628.

[237] G. Rosati, G. Volpe, and A. Biondi. Trajectory planning of a two-link rehabilitation robot arm. In *Proceedings of the 12th IFToMM World Congress*, Besancon, France, 2007.

[238] T. Sakurada, T. Kawase, K. Takano, T. Komatsu, and K. Kansaku. A BMI-based occupational therapy assist suit: Asynchronous control by SSVEP. *Frontiers in Neuroscience*, 7:172, 2013. ISSN 1662-453X. doi: 10.3389/fnins.2013.00172.

[239] R. Sanchez, J. Jiayin Liu, S. Rao, P. Shah, R. Smith, T. Rahman, S. Cramer, J. Bobrow, and D. Reinkensmeyer. Automating arm movement training following severe stroke: Functional exercises with quantitative feedback in a gravity-reduced environment. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 14(3):378–389, sep 2006. ISSN 1534-4320. doi: 10.1109/TNSRE.2006.881553.

[240] K. R. Saul, X. Hu, C. M. Goehler, M. E. Vidt, M. Daly, A. Velisar, and W. M. Murray. Benchmarking of dynamic simulation predictions in two software platforms using an upper limb musculoskeletal model. *Computer Methods in Biomechanics and Biomedical Engineering*, pages 1–14, jul 2014. ISSN 1476-8259. doi: 10.1080/10255842.2014.916698.

[241] A. Scholz, M. Sherman, I. Stavness, S. Delp, and A. Kecskeméthy. A fast multi-obstacle muscle wrapping method using natural geodesic variations. *Multibody System Dynamics*, 36(2):195–219, feb 2016. ISSN 1573272X. doi: 10.1007/s11044-015-9451-1.

[242] M. Schuenke, E. Schulte, and U. Schumacher. *THIEME Atlas of Anatomy: General Anatomy and Musculoskeletal System*, volume 26. Thieme, New York, first edition, 2010. ISBN 978-1-60406-286-1.

[243] C. Y. Scovil and J. L. Ronsky. Sensitivity of a Hill-based muscle model to perturbations in model parameters. *Journal of Biomechanics*, 39(11):2055–63, jan 2006. ISSN 0021-9290. doi: 10.1016/j.jbiomech.2005.06.005.

[244] A. Seireg and R. J. Arvikar. A mathematical model for evaluation of forces in lower extremeties of the musculo-skeletal system. *Journal of Biomechanics*, 6:313–326, 1973. ISSN 00219290. doi: 10.1016/0021-9290(73)90053-5.

[245] G. Serrancolí, J. Font-Llagunes, and A. Barjau. A weighted cost function to deal with the muscle force sharing problem in injured subjects: A single case study. *Proceedings of the Institution of Mechanical Engineers, Part K: Journal of Multi-body Dynamics*, 228(3):241–251, sep 2014. ISSN 20413068 14644193. doi: 10.1177/1464419314530110.

[246] A. Seth, M. Sherman, J. a. Reinbolt, and S. L. Delp. OpenSim: A musculoskeletal modeling and simulation framework for in silico investigations and exchange. *Procedia IUTAM*, 2:212–232, jan 2011. ISSN 22109838. doi: 10.1016/j.piutam.2011.04.021.

[247] Q. Shao, D. N. Bassett, K. Manal, and T. S. Buchanan. An EMG-driven model to estimate muscle forces and joint moments in stroke patients. *Computer Biology and Medicine*, 39(12):1083–1088, 2010. doi: 10.1016/j.compbiomed.2009.09.002.An.

[248] R. Sharif Razavian. *A human motor control framework based on muscle synergies*. Phd thesis, University of Waterloo, Waterloo, Ontario, Canada, 2017.

[249] R. Sharif Razavian and J. McPhee. Minimization of muscle fatigue as the criterion to solve muscle forces-sharing problem. In *ASME 2015 Dynamic Systems and Control Conference*, number 57243, page V001T15A001. ASME, oct 2015. ISBN 978-0-7918-5724-3. doi: 10.1115/DSCC2015-9678.

[250] R. Sharif Razavian, N. Mehrabi, and J. McPhee. A model-based approach to predict muscle synergies using optimization: Application to feedback control. *Frontiers in Computational Neuroscience*, 9:121, 2015. ISSN 1662-5188. doi: 10.3389/fncom.2015.00121.

[251] M. Sharif Shourijeh. *Optimal control and multibody dynamic modelling of human musculoskeletal systems*. Phd thesis, University of Waterloo, Waterloo, Ontario, Canada, 2013.

[252] M. Sharif Shourijeh and J. McPhee. Optimal control and forward dynamics of human periodic motions using Fourier series for muscle excitation patterns. *Journal of Computational and Nonlinear Dynamics*, 9(2):021005, sep 2013. ISSN 1555-1415. doi: 10.1115/1.4024911.

[253] M. Sharif Shourijeh, T. E. Flaxman, and D. L. Benoit. An approach for improving repeatability and reliability of non-negative matrix factorization for muscle synergy analysis. *Journal of Electromyography and Kinesiology*, 26:36–43, feb 2016. ISSN 18735711. doi: 10.1016/j.jelekin.2015.12.001.

[254] M. Sharif 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*, 12(5):051005, apr 2017. ISSN 1555-1415. doi: 10.1115/1.4036195.

[255] M. Sharifi, S. Behzadipour, and G. Vossoughi. Nonlinear model reference adaptive impedance control for human-robot interactions. *Control Engineering Practice*, 32: 9–27, nov 2014. ISSN 09670661. doi: 10.1016/j.conengprac.2014.07.001.

[256] S. E. Shaw, D. M. Morris, G. Uswatte, S. McKay, J. M. Meythaler, and E. Taub. Constraint-induced movement therapy for recovery of upper-limb function following traumatic brain injury. *Journal of Rehabilitation Research and Development*, 42(6): 769–778, 2005. ISSN 07487711. doi: 10.1682/JRRD.2005.06.0094.

[257] K. B. Shelburne and M. G. Pandy. A musculoskeletal model of the knee for evaluating ligament forces during isometric contractions. *Journal of Biomechanics*, 30:163–176, 1997. ISSN 00219290. doi: 10.1016/S0021-9290(96)00119-4.

[258] B. Siciliano, L. Sciavicco, L. Villani, and G. Oriolo. *Robotics: Modelling, Planning and Control*. Advanced Textbooks in Control and Signal Processing. Springer London, London, 2009. ISBN 978-1-84628-641-4. doi: 10.1007/978-1-84628-642-1.

[259] 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*, 98(2):133–143, feb 2008. ISSN 03401200. doi: 10.1007/s00422-007-0197-6.

[260] T. Skirven, A. Osterman, J. Fedorczyk, and P. Amadio. *Rehabilitation of the Hand and Upper Extremity*. Elsevier, 6th edition, 2011. ISBN 978-0-323-05602-1.

[261] J.-J. E. Slotine and Weiping Li. *Applied Nonlinear Control*. Prentice-Hall International, Englewood Cliffs, N.J, 1991. ISBN 0-13-040890-5.

[262] K. B. Smale, M. Sharif Shourijeh, and D. L. Benoit. Use of muscle synergies and wavelet transforms to identify fatigue during squatting. *Journal of Electromyography and Kinesiology*, 28:158–166, jun 2016. ISSN 18735711. doi: 10.1016/j.jelekin.2016.04.008.

[263] T. Soderstrom and P. G. Stoica. *System Identification*. Prentice-Hall International, 1988.

[264] J. F. Soechting and M. Flanders. Evaluating an integrated musculoskeletal model of the human arm. *Journal of Biomechanical Engineering*, 119(1):93–102, feb 1997. ISSN 0148-0731.

[265] M. M. Sohlberg and C. A. Mateer. *Cognitive rehabilitation: An integrative neuropsychological approach*. Guilford Press, New York, New York, USA, 2001. ISBN 9781572306134.

[266] J. Sovizi, A. Alamdari, M. S. Narayanan, and V. Krovi. Random matrix based input shaping control of uncertain parallel manipulators. In *Volume 7: 2nd Biennial International Conference on Dynamics for Design; 26th International Conference on Design Theory and Methodology*, page V007T05A008. ASME, aug 2014. ISBN 978-0-7918-4640-7. doi: 10.1115/DETC2014-34451.

[267] M. Sreenivasa, K. Ayusawa, and Y. Nakamura. Modeling and identification of a realistic spiking neural network and musculoskeletal model of the human arm, and an application to the stretch reflex. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 24(5):591–602, may 2016. ISSN 1534-4320. doi: 10.1109/TNSRE.2015.2478858.

[268] P. Staubli, T. Nef, V. Klamroth-Marganska, and R. Riener. Effects of intensive arm training with the rehabilitation robot ARMin II in chronic stroke patients: Four single-cases. *Journal of NeuroEngineering and Rehabilitation*, 6(1):46, 2009. ISSN 1743-0003. doi: 10.1186/1743-0003-6-46.

[269] K. M. Steele, A. Rozumalski, and M. H. Schwartz. Muscle synergies and complexity of neuromuscular control during gait in cerebral palsy. *Developmental Medicine and Child Neurology*, 57(12):1176–1182, dec 2015. ISSN 14698749. doi: 10.1111/dmcn.12826.

[270] J. Stein, K. Narendran, J. McBean, K. Krebs, and R. Hughes. Electromyography-controlled exoskeletal upper-limb-powered orthosis for exercise training after stroke. *American Journal of Physical Medicine & Rehabilitation*, 86(4):255–261, apr 2007. ISSN 0894-9115. doi: 10.1097/PHM.0b013e3180383cc5.

[271] A. H. Stienen, E. E. Hekman, F. C. Van der Helm, G. B. Prange, M. J. Jannink, A. M. Aalsma, and H. Van der Kooij. Freebal: Dedicated gravity compensation for the upper extremities. In *2007 IEEE 10th International Conference on Rehabilitation Robotics (ICORR)*, pages 804–808. IEEE, jun 2007. ISBN 978-1-4244-1319-5. doi: 10.1109/ICORR.2007.4428517.

[272] M. E. Stoykov and D. M. Corcos. A review of bilateral training for upper extremity hemiparesis. *Occupational Therapy International*, 16(3-4):190–203, 2009. ISSN 09667903. doi: 10.1002/oti.277.

[273] B. L. Suderman and A. N. Vasavada. Moving muscle points provide accurate curved muscle paths in a model of the cervical spine. *Journal of Biomechanics*, 45:400–404, 2012. ISSN 00219290. doi: 10.1016/j.jbiomech.2011.10.014.

[274] T. M. Sukal, M. D. Ellis, and J. P. Dewald. Source of work area reduction following hemiparetic stroke and preliminary intervention using the ACT 3D system. In *2006 International Conference of the IEEE Engineering in Medicine and Biology Society*, pages 177–180. IEEE, aug 2006. ISBN 1-4244-0032-5. doi: 10.1109/IEMBS.2006.259311.

[275] M. Suzuki, Y. Yamazaki, N. Mizuno, and K. Matsunami. Trajectory formation of the center-of-mass of the arm during reaching movements. *Neuroscience*, 76(2):597–610, jan 1997. ISSN 03064522. doi: 10.1016/S0306-4522(96)00364-8.

[276] K. Tahara, Z.-W. Luo, S. Arimoto, and H. Kino. Sensory-motor control mechanism for reaching movements of a redundant musculo-skeletal arm. *Journal of Robotic Systems*, 22(11):639–651, nov 2005. ISSN 0741-2223. doi: 10.1002/rob.20089.

[277] S. Tajeddin. *Automatic code generation of real-time nonlinear model predictive control for plug-in hybrid electric vehicle intelligent cruise controllers*. Masc thesis, University of Waterloo, Waterloo, Ontario, Canada, aug 2016.

[278] S. Tajeddin and N. L. Azad. Ecological cruise control of a plug-in hybrid electric vehicle: A comparison of different GMRES-based nonlinear model predictive controls. In *2017 American Control Conference (ACC)*, pages 3607–3612. IEEE, may 2017. ISBN 9781509059928. doi: 10.23919/ACC.2017.7963505.

[279] S. Tajeddin, M. Vajedi, and N. L. Azad. A Newton/GMRES approach to predictive ecological adaptive cruise control of a plug-in hybrid electric vehicle in car-following scenarios. *IFAC-PapersOnLine*, 49(21):59–65, jan 2016. ISSN 24058963. doi: 10.1016/j.ifacol.2016.10.511.

[280] G. Tao and P. V. Kokotovic. *Adaptive Control of Systems with Actuator and Sensor Nonlinearities*. Wiley-Interscience, 1996. ISBN 978-0-471-15654-3.

[281] A. Terrier, M. Aeberhard, Y. Michellod, P. Mullhaupt, D. Gillet, A. Farron, and D. P. Pioletti. A musculoskeletal shoulder model based on pseudo-inverse and null-space optimization. *Medical Engineering and Physics*, 32(9):1050–6, nov 2010. ISSN 1873-4030. doi: 10.1016/j.medengphy.2010.07.006.

[282] T. D. Thanh, J. Kotlarski, B. Heimann, and T. Ortmaier. Dynamics identification of kinematically redundant parallel robots using the direct search method. *Mechanism and Machine Theory*, 52:277–295, 2012. ISSN 0094114X. doi: 10.1016/j.mechmachtheory.2012.02.002.

[283] D. G. Thelen. Adjustment of muscle mechanics model parameters to simulate dynamic contractions in older adults. *Journal of Biomechanical Engineering*, 125(1): 70, 2003. ISSN 01480731. doi: 10.1115/1.1531112.

[284] D. G. Thelen and F. C. Anderson. Using computed muscle control to generate forward dynamic simulations of human walking from experimental data. *Journal of Biomechanics*, 39:1107–1115, 2006. ISSN 00219290. doi: 10.1016/j.jbiomech.2005.02.010.

[285] D. G. Thelen, F. C. Anderson, and S. L. Delp. Generating dynamic simulations of movement using computed muscle control. *Journal of Biomechanics*, 36:321–328, 2003. ISSN 00219290. doi: 10.1016/S0021-9290(02)00432-3.

[286] M. C. Tresch. Matrix factorization algorithms for the identification of muscle synergies: Evaluation on simulated and experimental data sets. *Journal of Neurophysiology*, 95(4):2199–2212, dec 2005. ISSN 0022-3077. doi: 10.1152/jn.00222.2005.

[287] B. C. Tsai, W. W. Wang, L. C. Hsu, L. C. Fu, and J. S. Lai. An articulated rehabilitation robot for upper limb physiotherapy and training. In *IEEE/RSJ 2010 International Conference on Intelligent Robots and Systems, IROS 2010 - Conference Proceedings*, pages 1470–1475. IEEE, oct 2010. ISBN 9781424466757. doi: 10.1109/IROS.2010.5649567.

[288] D. Tsirakos, V. Baltzopoulos, and R. Bartlett. Inverse optimization: Functional and physiological considerations related to the force-sharing problem. *Critical reviews in biomedical engineering*, 25(4-5):371–407, jan 1997. ISSN 0278-940X.

[289] Y. H. Tsuang, G. J. Novak, O. D. Schipplein, A. Hafezi, J. H. Trafimow, and G. B. Andersson. Trunk muscle geometry and centroid location when twisting. *Journal of Biomechanics*, 26(4-5):537–46, 1993. ISSN 0021-9290.

[290] A. Turolla, M. Dam, L. Ventura, P. Tonin, M. Agostini, C. Zucconi, P. Kiper, A. Cagnin, and L. Piron. Virtual reality for the rehabilitation of the upper limb motor function after stroke: A prospective controlled trial. *Journal of NeuroEngineering and Rehabilitation*, 10(1):85, jan 2013. ISSN 1743-0003. doi: 10.1186/1743-0003-10-85.

[291] Y. Uno, M. Kawato, and R. Suzuki. Formation and control of optimal trajectory in human multijoint arm movement. *Biological Cybernetics*, 61(2):89–101, 1989. ISSN 03401200. doi: 10.1007/BF00204593.

[292] A. L. E. Q. van Delden, C. L. E. Peper, K. N. Nienhuys, N. I. Zijp, P. J. Beek, and G. Kwakkel. Unilateral versus bilateral upper limb training after stroke: The upper limb training after stroke clinical trial. *Stroke*, 44(9):2613–6, sep 2013. ISSN 1524-4628. doi: 10.1161/STROKEAHA.113.001969.

[293] A. van den Bogert, D. Blana, and D. Heinrich. Implicit methods for efficient musculoskeletal simulation and optimal control. *Procedia IUTAM*, 2:297–316, 2011. doi: 10.1016/j.piutam.2011.04.027.Implicit.

[294] F. C. van der Helm. *The shoulder mechanism: A dynamical approach*. PhD thesis, Delft University of Technology, 1991.

[295] F. C. van der Helm. A finite element musculoskeletal model of the shoulder mechanism. *Journal of Biomechanics*, 27(5):551–69, may 1994. ISSN 0021-9290.

[296] F. C. van der Helm. Analysis of the kinematic and dynamic behavior of the shoulder mechanism. *Journal of Biomechanics*, 27(5):527–50, may 1994. ISSN 0021-9290.

[297] F. C. van der Helm. A standardized protocol for motion recordings of the shoulder. In *Proceedings of the First Conference of the International Shoulder Group*, pages 7–12, 1997.

[298] F. C. van der Helm and R. Veenbaas. Modelling the mechanical effect of muscles with large attachment sites: Application to the shoulder mechanism. *Journal of Biomechanics*, 24(12):1151–63, jan 1991. ISSN 0021-9290.

[299] M. M. van der Krogt, L. Bar-On, T. Kindt, K. Desloovere, and J. Harlaar. Neuromusculoskeletal simulation of instrumented contracture and spasticity assessment in children with cerebral palsy. *Journal of NeuroEngineering and Rehabilitation*, 13(1): 64, jul 2016. ISSN 1743-0003. doi: 10.1186/s12984-016-0170-5.

[300] H. Van der Loos, J. Wagner, N. Smaby, K. Chang, O. Madrigal, L. Leifer, and O. Khatib. ProVAR assistive robot system architecture. In *Proceedings 1999 IEEE International Conference on Robotics and Automation (Cat. No.99CH36288C)*, volume 1, pages 741–746. IEEE, 1999. ISBN 0-7803-5180-0. doi: 10.1109/ROBOT.1999.770063.

[301] H. E. Veeger, F. C. Van der Helm, L. H. Van der Woude, G. M. Pronk, and R. H. Rozendal. Inertia and muscle contraction parameters for musculoskeletal modelling of the shoulder mechanism. *Journal of Biomechanics*, 24(7):615–29, 1991. ISSN 0021-9290.

[302] H. E. Veeger, B. Yu, K. N. An, and R. H. Rozendal. Parameters for modeling the upper extremity. *Journal of Biomechanics*, 30(6):647–52, jun 1997. ISSN 0021-9290.

[303] R. D. Vita, R. Grange, P. Nardinocchi, and L. Teresi. Mathematical model for isometric and isotonic muscle contractions. *Journal of Theoretical Biology*, 425:1–10, 2017. ISSN 00225193. doi: 10.1016/j.jtbi.2017.05.007.

[304] D. Vrabie, O. Pastravanu, M. Abu-Khalaf, and F. Lewis. Adaptive optimal control for continuous-time linear systems based on policy iteration. *Automatica*, 45(2): 477–484, feb 2009. ISSN 00051098. doi: 10.1016/j.automatica.2008.08.017.

[305] C. P. Vyasarayani, T. Uchida, A. Carvalho, and J. McPhee. Parameter identification in dynamic systems using the homotopy optimization approach. *Multibody System Dynamics*, 26(4):411–424, dec 2011. ISSN 1384-5640. doi: 10.1007/s11044-011-9260-0.

[306] C. P. Vyasarayani, T. Uchida, and J. McPhee. Single-shooting homotopy method for parameter identification in dynamical systems. *Physical Review*, 85(3):036201, mar 2012. ISSN 1539-3755. doi: 10.1103/PhysRevE.85.036201.

[307] K. Walker, B. Samadi, M. Huang, J. Gerhard, K. Butts, and I. Kolmanovsky. Design environment for nonlinear model predictive control. *SAE Technical Papers*, apr 2016. doi: 10.4271/2016-01-0627.

[308] R. Wang, J. Zhang, and Z. Qiu. Optimal impedance control for an elbow rehabilitation robot. In *2017 14th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI)*, pages 388–392. IEEE, jun 2017. ISBN 978-1-5090-3056-9. doi: 10.1109/URAI.2017.7992759.

[309] Y. Wei, P. Bajaj, R. Scheldt, and J. Patton. Visual error augmentation for enhancing motor learning and rehabilitative relearning. In *Proceedings of the 2005 IEEE 9th International Conference on Rehabilitation Robotics*, volume 2005, pages 505–510. IEEE, 2005. ISBN 0780390032. doi: 10.1109/ICORR.2005.1501152.

[310] C. Wellington. Investigation of Serum Tau as a biomarker in pediatric TBI, [Available: http://www.ctrc-ccrt.ca/projects/investigation-of-serum-tau-as-a-biomarker-in-pediatric-tbi/], 2013.

[311] D. R. Wilkie. The mechanical properties of muscle. *British Medical Bulletin*, 12(3): 2–7, 1956.

[312] D. R. Wilkie and J. M. Ritchie. The dynamics of muscular contraction. *The Journal of Physiology*, 143(1):104–113, 1924.

[313] J. M. Winters. *Generalized analysis and design of antagonistic muscle models: Effect of nonlinear properties on the control of human movement*. PhD thesis, University of California, Berkeley, USA, 1985.

[314] J. M. Winters and D. G. Kleweno. Effect of initial upper-limb alignment on muscle contributions to isometric strength curves. *Journal of Biomechanics*, 26:143–153, 1993. ISSN 00219290. doi: 10.1016/0021-9290(93)90045-G.

[315] J. M. Winters and L. Stark. Muscle models: What is gained and what is lost by varying model complexity. *Biological Cybernetics*, 55(6):403–420, 1987.

[316] J. M. Winters and L. Stark. Estimated mechanical properties of synergistic muscles involved in movements of a variety of human joints. *Journal of Biomechanics*, 21: 1027–1041, 1988. ISSN 00219290. doi: 10.1016/0021-9290(88)90249-7.

[317] E. Wolbrecht, V. Chan, D. Reinkensmeyer, and J. Bobrow. Optimizing compliant, model-based robotic assistance to promote neurorehabilitation. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 16(3):286–297, jun 2008. ISSN 1534-4320. doi: 10.1109/TNSRE.2008.918389.

[318] E. T. Wolbrecht, V. Chan, V. Le, S. C. Cramer, D. J. Reinkensmeyer, and J. E. Bobrow. Real-time computer modeling of weakness following stroke optimizes robotic assistance for movement therapy. In *Proceedings of the 3rd International IEEE EMBS Conference on Neural Engineering*, pages 152–158. IEEE, may 2007. ISBN 1424407923. doi: 10.1109/CNE.2007.369635.

[319] J. E. Wood, S. G. Meek, and S. C. Jacobsen. Quantitation of human shoulder anatomy for prosthetic arm control–I. Surface modelling. *Journal of Biomechanics*, 22(3):273–92, 1989. ISSN 0021-9290.

[320] J. E. Wood, S. G. Meek, and S. C. Jacobsen. Quantitation of human shoulder anatomy for prosthetic arm control–II. Anatomy matrices. *Journal of Biomechanics*, 22(4):309–25, 1989. ISSN 0021-9290.

[321] C.-Y. Wu, C.-L. Yang, M.-d. Chen, K.-C. Lin, and L.-L. Wu. Unilateral versus bilateral robot-assisted rehabilitation on arm-trunk control and functions post stroke: A randomized controlled trial. *Journal of NeuroEngineering and Rehabilitation*, 10: 35, jan 2013. ISSN 1743-0003. doi: 10.1186/1743-0003-10-35.

[322] G. Wu, F. C. van der Helm, H. (DirkJan) 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*, 38(5):981–992, may 2005. ISSN 00219290. doi: 10.1016/j.jbiomech.2004.05.042.

[323] J. Wu, J. Wang, and Z. You. An overview of dynamic parameter identification of robots. *Robotics and Computer-Integrated Manufacturing*, 26(5):414–419, 2010. ISSN 07365845. doi: 10.1016/j.rcim.2010.03.013.

[324] M. Xiao and J. Higginson. Sensitivity of estimated muscle force in forward simulation of normal walking. *Journal of applied biomechanics*, 26(2):142–9, may 2010. ISSN 1065-8483.

[325] S. Xie and W. Meng. EMG-driven physiological model for upper limb. In *Biomechatronics in Medical Rehabilitation*, pages 103–123. Springer International Publishing, Cham, 2017. doi: 10.1007/978-3-319-52884-7_6.

[326] X. Xu, J.-h. Lin, and R. W. McGorry. A regression-based 3-D shoulder rhythm. *Journal of Biomechanics*, 47(5):1206–10, mar 2014. ISSN 1873-2380. doi: 10.1016/j.jbiomech.2014.01.043.

[327] M. Yamashita. Robotic rehabilitation system for human upper limbs using guide control and manipulability ellipsoid prediction. *Procedia Technology*, 15:559–565, 2014. ISSN 22120173. doi: 10.1016/j.protcy.2014.09.016.

[328] H. Yanagida, S. Ishii, W. H. Short, F. W. Werner, M. M. Weiner, and S. Masaoka. Radiologic evaluation of the ulnar styloid. *Journal of Hand Surgery*, 27(1):49–56, jan 2002. ISSN 03635023. doi: 10.1053/jhsu.2002.30920.

[329] M. Ye. Parameter identification of dynamical systems based on improved particle swarm optimization. In *Intelligent Control and Automation*, pages 351–360. Springer Berlin Heidelberg, 2006. doi: 10.1007/978-3-540-37256-1_42.

[330] S.-C. Yeh, S.-H. Lee, J.-C. Wang, S. Chen, Y.-T. Chen, Y.-Y. Yang, H.-R. Chen, Y.-P. Hung, A. Rizzo, and T.-L. Tsai. Stroke rehabilitation via a haptics-enhanced virtual reality system. In *Advances in Intelligent Systems and Applications - Volume 2*, pages 439–453. Springer, Berlin, Heidelberg, 2013. doi: 10.1007/978-3-642-35473-1_45.

[331] I. Yoon and D. E. Thompson. Planning tendon paths using an interactive graphic workstation. *Journal of Biomechanical Engineering*, 112:387–391, 1990. ISSN 01480731. doi: 10.1115/1.2891201.

[332] F. Yoshikawa, H. Hirai, E. Watanabe, Y. Nagakawa, A. Kuroiwa, E. Grabke, M. Uemura, F. Miyazaki, and H. I. Krebs. Equilibrium-point-based synergies that encode coordinates in task space: A practical method for translating functional synergies from human to musculoskeletal robot arm. In *2016 IEEE-RAS 16th International Conference on Humanoid Robots (Humanoids)*, pages 1135–1140. IEEE, nov 2016. ISBN 978-1-5090-4718-5. doi: 10.1109/HUMANOIDS.2016.7803413.

[333] T. Yoshikawa. Manipulability of robotic mechanisms. *The International Journal of Robotics Research*, 4(2):3–9, jun 1985. ISSN 0278-3649. doi: 10.1177/027836498500400201.

[334] W. Yu, J. Rosen, and X. Li. PID admittance control for an upper limb exoskeleton. *Proceedings of the 2011 American Control Conference*, pages 1124–1129, jun 2011. ISSN 0743-1619. doi: 10.1109/ACC.2011.5991147.

[335] M. Zadravec and Z. Matjačić. Planar arm movement trajectory formation: An optimization based simulation study. *Biocybernetics and Biomedical Engineering*, 33 (2):106–117, jan 2013. ISSN 02085216. doi: 10.1016/j.bbe.2013.03.006.

[336] F. E. Zajac. Muscle and tendon: Properties, models, scaling, and application to biomechanics and motor control. *Critical Reviews in Biomedical Engineering*, 17(4): 359–411, jan 1989. ISSN 0278-940X.

[337] J. Zariffa, J. Steeves, and D. K. Pai. Changes in hand muscle synergies in subjects with spinal cord injury: Characterization and functional implications. *The Journal of Spinal Cord Medicine*, 35(5):310–8, sep 2012. ISSN 1079-0268. doi: 10.1179/2045772312Y.0000000037.

[338] O. Zarifi and I. Stavness. Muscle wrapping on arbitrary meshes with the heat method. *Computer Methods in Biomechanics and Biomedical Engineering*, 20(2):119–129, jan 2017. ISSN 1025-5842. doi: 10.1080/10255842.2016.1205043.

[339] V. M. Zatsiorsky and B. I. Prilutsky. *Biomechanics of Skeletal Muscles*. Human Kinetics, 1 edition, 2012. ISBN 9780736080200.

APPENDICES

Appendix A

General Information

A.1 Musculoskeletal upper extremity models

Several musculoskeletal upper extremity models are summarized in Table A.1. The common feature of these models is the inclusion of the shoulder. However, as in Table A.1, the shoulder has been modeled with a different number of joints discussed previously [77]. These models use different numbers of muscles considering muscle mechanics and muscle paths to solve the muscle redundancy problem by one of the mentioned methods. Most of these models are 3D, but some of them are 2D.

Some elbow models are integrated into the upper extremity models in Table A.1. In these models, the elbow has been modeled as a single revolute joint (flexion/extension joint) or two revolute joints (flexion/extension and pronation/supination joints). In the literature, there are some detailed models of the forearm pronation/supination [142]. However, for simplicity, in current models, the pronation/supination joint is modeled only by a radioulnar joint which connects proximal parts of radius and ulna. In other words, there are no humeroradial and distal radioulnar joints in current musculoskeletal models.

Among musculoskeletal upper extremity models, only the Texas and Stanford models [96, 122] have a wrist model. In these models, the wrist is modeled as two independent revolute joints even though the wrist consists of a radiocarpal joint that is best modeled by an ellipsoid joint. Furthermore, the ROM for two independent revolute joints is more than the ROM of an ellipsoid joint. Thus, two independent revolute joints may result in unphysiological configurations for the wrist.

Table A.1: Partial list of available musculoskeletal upper extremity models.

NO	Model	# Muscles (Elements)	Muscle mechanics model	Muscle path model	Muscle redundancy problem	Musculoskeletal model type	
1	Swiss (Gothenburg) Model [119–121, 141]	21 (34)	—	Straight-line	SO	3D	Static
2	[75]	2 (2)	—	Straight-line	Analytical	2D (frontal plane)	Static
3	Utah Model [319, 320]	19 (30)	—	Straight-line	—	3D	Dynamic
4	Delft (Dutch) Shoulder and Elbow Model [147, 198, 295, 296, 298, 301, 302]	20 (95)	—	Combined	SO	3D	Dynamic
5	Tampere Model [195, 196]	19 (30)	—	Straight-line	SO	3D	Static
6	Mayo Model [125]	4 (6)	—	Straight-line	SO	2D (frontal plane)	Static
7	Minnesota Model [264]	6 (9)	CE/SE Hill-type	Straight-line	—	3D	Dynamic
8	[264] [154]	9 (13)	—	Straight-line	—	3D	Static
9	Texas Model [96, 98, 99]	26 (42)	CE/PE/SE Hill-type	Obstacle-set	SO	3D	Static
10	[197]	12 (6)	CE/PE/SE Hill-type	—	—	2D (horizontal plane)	Dynamic
11	Stanford Model [122]	32 (50)	CE/PE/SE Hill-type	Combined	—	3D	Static
12	Newcastle Model [47]	23 (31)	—	Combined	SO	3D	Dynamic
13	Shoulder Loading Analysis Modulus [66]	23 (38)	—	Combined	SO	3D	Dynamic
14	Lausanne Model [281]	5 (6)	—	Obstacle-set	Analytical	3D	Dynamic
15	Lisbon Model [5, 221]	21 (37)	CE/PE Hill-type	Obstacle-set	DO	3D	Dynamic

Table A.1 (continued)

NO	Model	Shoulder	Shoulder rhythm	Shoulder stability	Elbow	Wrist	# Ligament
1	Swiss (Gothenburg) Model [119–121, 141]	3 Spherical joints + STJ	Extrapolation	—	1 Revolute joint	—	1
2	[75]	2 Revolute joints	HSR	—	—	—	0
3	Utah Model [319, 320]	3 Spherical joints	Measurement	—	2 Revolute joints	—	0
4	Delft (Dutch) Shoulder and Elbow Model [147, 198, 295, 296, 298, 301, 302]	3 Spherical joints + STJ	Measurement	Constrained	1 Revolute joint	—	3
5	Tampere Model [195, 196]	3 Spherical joints + STJ	—	Constrained	1 Revolute joint	—	0
6	Mayo Model [125]	1 Revolute joint	—	—	—	—	0
7	Minnesota Model [264]	1 Spherical joint	—	—	1 Revolute joint	—	0
8	[264] [154]	1 Spherical joint	—	—	1 Revolute joint	—	0
9	Texas Model [96, 98, 99]	3 Spherical joints	—	—	2 Revolute joints	2 Revolute joints	0
10	[197]	1 Revolute joint	—	—	1 Revolute joint	—	0
11	Stanford Model [122]	3 Spherical joints	Regression	—	2 Revolute joints	2 Revolute joints	0
12	Newcastle Model [47]	3 Spherical joints + STJ	Regression	Constrained	2 Revolute joints	—	3
13	Shoulder Loading Analysis Modulus [66]	3 Spherical joints + STJ	Regression	Constrained	—	—	3
14	Lausanne Model [281]	2 Spherical joints	HSR	—	1 Revolute joint	—	0
15	Lisbon Model [5, 221]	3 Spherical joints + STJ	Measurement	Constrained	2 Revolute joints	—	3

A.2 Simulation setup dynamic parameter identification

The DPI of the simulation setup was also done by other conventional optimization methods: Bound Optimization by Quadratic Approximation (BOBYQA) and Pattern Search (PS) in Table A.2, and Particle Swarm Optimization (PSO) and Genetic Algorithm (GA) in Table A.3. The objective functions for these methods are similar to the SQP method.

Table A.2: Simulation setup DPI values using BOBYQA and PS optimization methods.

					Case A Values					Case B Values				
			Bounds		Model	BOBYQA		PS		Model	BOBYQA		PS	
$\mathbf{p}$	Unit	$\mathbf{p}^0$	$\mathbf{p}_{min}$	$\mathbf{p}_{max}$	A	I	II	I	II	B	I	II	I	II
c_l	rad	0.0	0	β	0.0035	2.6e−5	0.0000	0.0030	0.0030	0.0025	1.7e−5	0.0000	0.0000	0.0000
J_M	$\mathrm{kg\,m^2}$	0.5	ϵ	1.000	0.0100	0.0197	0.5000	0.0097	0.0093	0.0300	0.3271	0.4998	0.0625	0.0806
J_L	$\mathrm{kg\,m^2}$	0.5	ϵ	1.000	0.0200	0.3292	0.4998	0.0198	0.0195	0.0100	0.6430	0.5000	0.1344	0.9108
B_M	Nm/rad	0.5	ϵ	1.000	0.1000	0.3036	0.4999	0.1757	0.1757	0.2500	0.3264	0.5000	0.0159	0.1223
B_L	Nm/rad	0.5	ϵ	1.000	0.2000	6.2e−5	0.4999	0.1249	0.1249	0.1500	0.1756	0.5000	0.4669	0.4987

Table A.3: Simulation setup DPI values using PSO and GA optimization methods.

					Case A Values					Case B Values				
			Bounds		Model	PSO		GA		Model	PSO		GA	
$\mathbf{p}$	Unit	$\mathbf{p}^0$	$\mathbf{p}_{min}$	$\mathbf{p}_{max}$	A	I	II	I	II	B	I	II	I	II
c_l	rad	0.0	0	β	0.0035	0.0036	9.4e−4	9.6e−4	0.0022	0.0025	0.0032	0.0011	0.0017	0.0023
J_M	$\mathrm{kg\,m^2}$	0.5	ϵ	1.000	0.0100	0.0097	0.0108	0.0100	0.0106	0.0300	0.0358	0.0417	0.0393	0.0514
J_L	$\mathrm{kg\,m^2}$	0.5	ϵ	1.000	0.0200	0.0199	0.0206	0.0209	0.0208	0.0100	0.0126	0.0102	0.0146	0.1450
B_M	Nm/rad	0.5	ϵ	1.000	0.1000	0.2169	0.0020	0.1656	0.0051	0.2500	0.1443	0.0019	0.1362	0.0635
B_L	Nm/rad	0.5	ϵ	1.000	0.2000	0.0829	0.2996	0.1364	0.2958	0.1500	0.2559	0.3998	0.2635	0.4005

A.3 Desired hypotrochoid paths

The desired trajectory equation in the Cartesian space is set to:

$$\begin{cases} Z_d = \mathscr{Z}_d \left(2\cos(\mho_d t) + c_d \cos(\frac{2}{3}\mho_d t)\right) + \mathscr{Z}_0 \\ X_d = \mathscr{X}_d \left(2\sin(\mho_d t) - c_d \sin(\frac{2}{3}\mho_d t)\right) + \mathscr{X}_0 \end{cases} \tag{A.1}$$

where amplitudes $\mathscr{Z}_d$ and $\mathscr{X}_d$, and initial positions $\mathscr{Z}_0 = 0$ m and $\mathscr{X}_0 = 0.4$ m are defined such that the desired trajectory lies inside the robot's DPI workspace. $\mho_d = \Omega_d \tanh(5t)$ is a hyperbolic frequency function, which provides zero velocity at $t = 0$, then converges to Ω_d as time passes. For the SC star, $[\mathscr{Z}_d = 0.0442$ m$, \mathscr{X}_d = 0.0421$ m$, c_d = 3]$, and for the RC star, $[\mathscr{Z}_d = 0.0368$ m$, \mathscr{X}_d = 0.0350$ m$, c_d = 4]$.

A.4 Sliding-mode control

For a nonlinear multi-input multi-output system defined in (3.30), the sliding-mode (robust) control will be:

$$\mathbf{u} = \mathring{\mathbf{M}}(\mathbf{q})\ddot{\mathbf{q}}_r + \mathring{\mathbf{C}}(\mathbf{q})\dot{\mathbf{q}}_r + \mathring{\mathbf{K}}_P \Delta\mathbf{q} + \mathring{\mathbf{f}}_T + \mathbf{J}^{T\circ}\mathbf{f}_F - \boldsymbol{\mathfrak{K}}\,\mathrm{sat}(\boldsymbol{\phi}^{-1}\boldsymbol{\mathfrak{s}}) \tag{A.2}$$

where $\circ$ accent denotes the estimated value, and $\Delta\mathbf{q}$ is $\mathbf{q} - \mathbf{q}_0$. $\dot{\mathbf{q}}_r$ is the reference velocity which is formed by the following equation:

$$\dot{\mathbf{q}}_r = \dot{\mathbf{q}}_d - \boldsymbol{\Lambda}\widetilde{\mathbf{q}} \tag{A.3}$$

in which $\boldsymbol{\Lambda} = \mathrm{diag}(\Lambda_1, \Lambda_2)$ is a positive definite diagonal matrix, and the $\sim$ accent refers to the error with respect to the desired trajectory (i.e., $\widetilde{\mathbf{q}} = \mathbf{q} - \mathbf{q}_d$). In (A.2), $\boldsymbol{\mathfrak{s}}$ is the sliding surface and it is equal to the reference velocity error (i.e., $\boldsymbol{\mathfrak{s}} = \dot{\mathbf{q}} - \dot{\mathbf{q}}_r$). $\boldsymbol{\mathfrak{K}} = \mathrm{diag}(\mathfrak{K}_1, \mathfrak{K}_2)$ and $\boldsymbol{\phi} = \mathrm{diag}(\phi_1, \phi_2)$ are positive-definite diagonal matrices, and the vector function $\mathrm{sat}(\boldsymbol{\phi}^{-1}\boldsymbol{\mathfrak{s}})$ is defined as:

$$\left[\mathrm{sat}(\boldsymbol{\phi}^{-1}\boldsymbol{\mathfrak{s}})\right]_i = \begin{cases} \frac{\mathfrak{s}_i}{\phi_i} & |\mathfrak{s}_i| \le \phi_i \\ \mathrm{sign}(\frac{\mathfrak{s}_i}{\phi_i}) & |\mathfrak{s}_i| > \phi_i \end{cases} \tag{A.4}$$

to reduce the chattering effect. If the matrix $\boldsymbol{\mathfrak{K}}$ has the following property:

$$\mathfrak{K}_i \ge \left| \left[\widetilde{\mathbf{M}}(\mathbf{q})\ddot{\mathbf{q}}_r + \widetilde{\mathbf{C}}(\mathbf{q})\dot{\mathbf{q}}_r + \widetilde{\mathbf{K}}_P \Delta\mathbf{q} + \widetilde{\mathbf{f}}_T + \mathbf{J}^T\widetilde{\mathbf{f}}_F\right]_i \right| + \phi_i \tag{A.5}$$

then the sliding condition will be satisfied. The $\sim$ accent denotes the modeling error of the corresponding matrices and vectors (e.g., $\widetilde{\mathbf{M}} = \mathring{\mathbf{M}} - \mathbf{M}$); thus, if the identified model of the system is accurate, elements of $\boldsymbol{\mathfrak{K}}$ will be as small as elements of $\boldsymbol{\phi}$. Assigned values for control parameters (which are determined by trial and error) are reported in Table A.4.

Table A.4: Sliding-mode control parameters.

Parameter	Λ_1	Λ_2	ϕ_1	ϕ_2	$\mathfrak{K}_1$	$\mathfrak{K}_2$
Unit	Hz	Hz	rad/s	rad/s	Nm	Nm
Value	5	5	0.2	0.2	0.5	0.5

A.5 Robot dynamic parameters

Identified robot dynamic parameters (using direct collocation method) are reported in Table A.5. The rest of the parameters are in Table A.6. Note that in Table A.6, some of the parameters (l_i: link lengths) are measured directly; others are estimated using a CAD toolbox and CAD model of the robot.

Table A.5: Identified robot dynamic parameters using direct collocation method.

$\bar{\mathbf{p}}$	$\mathbf{p}$	Detail	Unit	Value
$\bar{p}_1$	$p_1^{①,②}$	Inertial Parameters	$\mathrm{kg\,m^2}$	0.1145
$\bar{p}_2$	$p_2^{①,②}$		$\mathrm{kg\,m^2}$	0.0931
$\bar{p}_3$	$p_3^{①,②}$		$\mathrm{kg\,m^2}$	−0.0027
$\bar{p}_4$	$p_4^{①,②}$		$\mathrm{kg\,m^2}$	0.1669
$\bar{p}_5$	$p_5^{①,②,③}$	$\hat{\mu}_{d1}$	N m	0.4034
$\bar{p}_6$	$p_6^{①,②,③}$	$\hat{\mu}_{s1}$	N m	0.1340
$\bar{p}_7$	$p_7^{①,②,③}$	$\hat{\mu}_{v1}$	N m s	0.0155
$\bar{p}_8$	$p_8^{①,②,③}$	ω_{t1}	rad/s	0.2279
$\bar{p}_9$	$p_9^{①,②,③}$	$\hat{\mu}_{d2}$	N m	0.3229
$\bar{p}_{10}$	$p_{10}^{①,②,③}$	$\hat{\mu}_{s2}$	N m	0.1766
$\bar{p}_{11}$	$p_{11}^{①,②,③}$	$\hat{\mu}_{v2}$	N m s	0.0565
$\bar{p}_{12}$	$p_{12}^{①,②,③}$	ω_{t2}	rad/s	0.1419
$\bar{p}_{13}$	$p_{13}^{①}$	$\hat{\mu}_{d3}$	N	1.0395
$\bar{p}_{14}$	$p_{14}^{①}$	$\hat{\mu}_{s3}$	N	0.5657
$\bar{p}_{15}$	$p_{15}^{①}$	v_t	m/s	0.1893
$\bar{p}_{16}$	$p_{16}^{①,②,③}$	$\mathbf{K}_R\rfloor_{(1,1)}$	N m	0.1008
$\bar{p}_{17}$	$p_{17}^{①,②,③}$	$\mathbf{K}_R\rfloor_{(1,2)}$	N m	0.0112
$\bar{p}_{18}$	$p_{18}^{①,②,③}$	$\mathbf{K}_R\rfloor_{(2,2)}$	N m	0.0668
$\bar{p}_{19}$	$p_{19}^{①,②,③}$	$\mathbf{q}_0\rfloor_1$	rad	1.2160
$\bar{p}_{20}$	$p_{20}^{①,②,③}$	$\mathbf{q}_0\rfloor_2$	rad	2.7509
$\bar{p}_{21}$	$p_{13}^{②,③}$	$\hat{\mu}_{d3}$	N	1.2389
$\bar{p}_{22}$	$p_{14}^{②,③}$	$\hat{\mu}_{s3}$	N	0.8799
$\bar{p}_{23}$	$p_{15}^{②,③}$	v_t	m/s	0.0416
$\bar{p}_{24}$	$p_1^{③}$	Inertial Parameters	$\mathrm{kg\,m^2}$	0.2484
$\bar{p}_{25}$	$p_2^{③}$		$\mathrm{kg\,m^2}$	0.2076
$\bar{p}_{26}$	$p_3^{③}$		$\mathrm{kg\,m^2}$	−0.0027
$\bar{p}_{27}$	$p_4^{③}$		$\mathrm{kg\,m^2}$	0.2584
$\bar{p}_{28}$	—	$N_3^{③}$	—	1.8231

Table A.6: Estimated and measured robot dynamic parameters (see Fig. 3.1 for parameter definitions). F_N is the normal force under the end-effector when there is no externally applied load.

Parameter	Unit	Value
x_1	m	0
x_2	m	0
x_3	m	0
x_4	m	0.0082
z_1	m	−0.0051
z_2	m	0.0012
z_3	m	0.1576
z_4	m	0.2744
m_1	kg	2.5783
m_2	kg	3.3986
m_3	kg	0.0620
m_4	kg	1.0832

Parameter	Unit	Value
l_1	m	0.1000
l_2	m	0.3100
l_3	m	0.3100
l_4	m	0.3750
I_1	$\mathrm{kg\,m^2}$	0.0224
I_2	$\mathrm{kg\,m^2}$	0.0606
I_3	$\mathrm{kg\,m^2}$	0.0007
I_4	$\mathrm{kg\,m^2}$	0.0098
F_N	N	11.3404

A.6 Musculoskeletal upper extremity model parameters

Parameters of the developed musculoskeletal models defined in Chapter 4 are presented in following tables.

Table A.7: Dynamic parameters of MA2D-$I_{1,2,3}$.

Parameter	Length	Mass	Inertia	COM
Unit	m	kg	$\mathrm{kg\,m^2}$	m
Upper Arm	0.31	1.93	0.0141	0.165
Forearm	0.34	1.52	0.0188	0.190

Table A.8: MA2D-$I_{1,2}$ muscle origin and insertion points. Values are measured in meters. Note that O, S, and E are the coordinate systems defined in Fig. 4.1

Origin ($\curlywedge$)	o_1 (O)	o_2 (O)	o_3 (S)	o_4 (S)	o_5 (O)	o_6 (O)
$Z_{\curlywedge}$	-0.055	0.055	0.12	0.12	0.04	0.04
Insertion ($\curlywedge$)	i_1 (S)	i_2 (S)	i_3 (E)	i_4 (E)	i_5 (E)	i_6 (E)
$Z_{\curlywedge}$	0.08	0.08	0.03	-0.03	0.045	-0.045

Table A.9: MA2D-I_3 muscle origin and insertion points.

Origin ($\curlywedge$)	o_1 (O)	o_2 (O)	o_3 (S)	o_4 (S)	o_5 (O)	o_6 (O)
$\begin{bmatrix} Z \\ X \end{bmatrix}_{\curlywedge}$	$\begin{bmatrix} -0.0718 \\ 0.0321 \end{bmatrix}$	$\begin{bmatrix} 0.0000 \\ -0.0447 \end{bmatrix}$	$\begin{bmatrix} 0.2098 \\ -0.0012 \end{bmatrix}$	$\begin{bmatrix} 0.2090 \\ -0.0002 \end{bmatrix}$	$\begin{bmatrix} -0.0100 \\ 0.0314 \end{bmatrix}$	$\begin{bmatrix} 0.0000 \\ -0.0315 \end{bmatrix}$
Insertion ($\curlywedge$)	i_1 (S)	i_2 (S)	i_3 (E)	i_4 (E)	i_5 (E)	i_6 (E)
$\begin{bmatrix} Z \\ X \end{bmatrix}_{\curlywedge}$	$\begin{bmatrix} 0.0753 \\ 0.0275 \end{bmatrix}$	$\begin{bmatrix} 0.0447 \\ 0.0000 \end{bmatrix}$	$\begin{bmatrix} 0.0497 \\ -0.0018 \end{bmatrix}$	$\begin{bmatrix} -0.0194 \\ 0.0004 \end{bmatrix}$	$\begin{bmatrix} 0.0472 \\ -0.0043 \end{bmatrix}$	$\begin{bmatrix} -0.0174 \\ -0.0187 \end{bmatrix}$

Table A.10: Dynamic parameters of MA2D-II.

Parameter	Length	Mass	Inertia	COM	Rotational Damping
Unit	m	kg	$\mathrm{kg\,m^2}$	m	Nms/rad
Upper Arm	0.3202	2.16	0.0274	$Z_S = 0.1453, X_S = 0.0094$	1.0322
Forearm	0.3297	2.07	0.0308	$Z_E = 0.1649, X_E = -0.0007$	0.2493

Table A.11: Dynamic parameters of a subject-specific MA2D-II.

Parameter	Length	Mass	Inertia	COM	Rotational Damping
Unit	m	kg	$\mathrm{kg\,m^2}$	m	Nms/rad
Upper Arm	0.2733	1.62	0.0452	$Z_S = 0.1344, X_S = 0$	0
Forearm	0.3259	1.33	0.0465	$Z_E = 0.2830, X_E = -0.0526$	0

Table A.12: Muscle mechanics parameters for a subject-specific MA2D-II.

Parameter	Unit	Muscle1	Muscle2	Muscle3	Muscle4	Muscle5	Muscle6
F^{iso}	N	780	1009	209	210	481	1059
L_0^{M}	cm	11.0	8.8	8.0	8.4	12.1	14.3
$L_{\mathrm{s}}^{\mathrm{T}}$	cm	4.9	7.2	11.5	18.2	23	19
α_0^{M}	deg	0	0	13.8	15.1	10	15

A.7 Rehabilitation robot control parameters

Parameters of the designed controllers in chapters 5 and 6 are defined in following tables. These parameters are determined by trial and error.

Table A.13: HFIC parameters.

Parameter	$\mathbf{M}_{imp}$	$\mathbf{B}_{imp}$	$\mathbf{K}_{imp}$	$\mathbb{K}_P$	$\mathbb{K}_I$	$\mathbf{F}_d^{\Sigma}$
Unit	kg	Ns/m	N/m	—	1/s	N
Value	$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$	$\begin{bmatrix} 63.25 & 0 \\ 0 & 126.49 \end{bmatrix}$	$\begin{bmatrix} 1000 & 0 \\ 0 & 1000 \end{bmatrix}$	$\begin{bmatrix} 1 & 0 \\ 0 & 0.05 \end{bmatrix}$	$\begin{bmatrix} 20 & 0 \\ 0 & 0.2 \end{bmatrix}$	$\begin{bmatrix} 0 \\ -0.1 \end{bmatrix}$

Table A.14: OIC parameters.

Parameter	y_{tol1}	y_{tol2}	y_{tol3}	y_{tol4}	F_{tol1}	F_{tol2}	c
Unit	m	m	m/s	m/s	N	N	—
Value	$\frac{1}{90}$	$\frac{1}{30}$	1	$\frac{100}{32}$	10	$\frac{10}{3}$	1

Table A.15: SOIC parameters.

Parameter	M_{imp1}	M_{imp2}	y_{tol1}	y_{tol2}	F_{tol1}	F_{tol2}
Unit	kg	kg	m	m	N	N
Value	2.2	2.2	$\frac{1}{90}$	$\frac{1}{30}$	10	10

Table A.16: Parameters for the NMPC with an external 3D musculoskeletal model.

Parameter	$\mathbb{Q}$	$\mathbb{R}$	c_1	c_2
Unit	$\mathrm{diag}(1/\mathrm{m}^2, 1/\mathrm{m}^2, \mathrm{s}^2/\mathrm{m}^2, \mathrm{s}^2/\mathrm{m}^2)$	$1/(\mathrm{Nm})^2$	—	—
Value	$\mathrm{diag}(2\mathrm{e}5, 2\mathrm{e}4, 400, 400)$	$\mathrm{diag}(1, 1)$	0.95	0.002

Table A.17: Parameters for the NMPC with an internal 2D musculoskeletal model.

Parameter	$\mathbb{Q}$	$\mathbb{R}$
Unit	$\mathrm{diag}(1/\mathrm{m}^2, 1/\mathrm{m}^2, \mathrm{s}^2/\mathrm{m}^2, \mathrm{s}^2/\mathrm{m}^2)$	$\mathrm{diag}(1/(\mathrm{Nm})^2, 1/(\mathrm{Nm})^2, —, —, —, —, —, —)$
Value	$\mathrm{diag}(1.67\mathrm{e}4, 1.67\mathrm{e}3, 33.3, 33.3)$	$\mathrm{diag}(1\mathrm{e}-4, 1\mathrm{e}-4, 0.5, 0.5, 0.5, 0.5, 0.5, 0.5)$

Appendix B

Muscle Mechanics

A musculotendon can be modeled as a combination of equal and parallel long fibers which have angle α^{M}(pennation angle) with respect to the tendon direction (see Fig. B.1 which is adopted from [336]).

By assuming that the muscles have constant volumes, the distance between two aponeuroses of two ends (in Fig. B.1:w^{M}) is constant [336]. In other words:

$$\mathrm{w}^{\mathrm{M}} = L^{\mathrm{M}} \cos(\alpha^{\mathrm{M}}) = L_0^{\mathrm{M}} \cos(\alpha_0^{\mathrm{M}}) = \text{constant} \tag{B.1}$$

The Hill-type muscle model with three elements consists of (see Fig. B.2):

1. CE: is representative of the active part of the muscle, and is basically a force generator.
2. PE: models the tissue parallel to the muscle fibers and is parallel to the CE element.

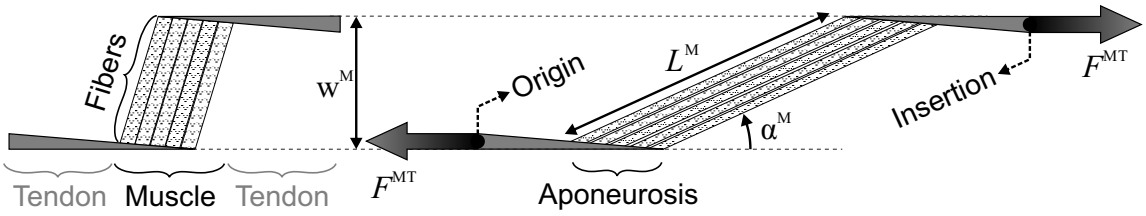


Figure B.1: Simplified musculotendon structure. $F^{\mathrm{MT}} = F^{\mathrm{M}}$ is the musculotendon force.

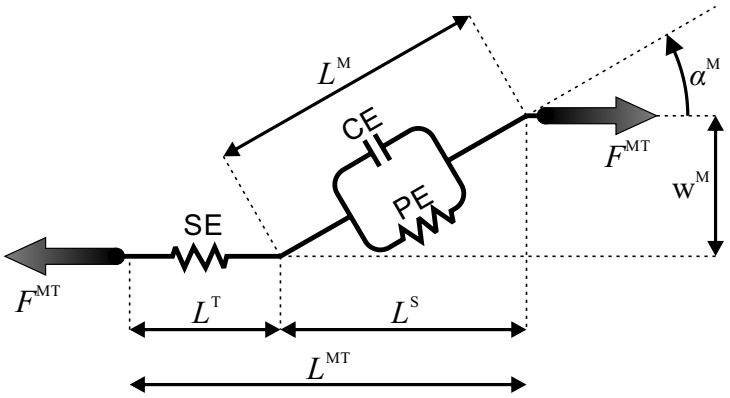


Figure B.2: Three-element Hill-type muscle model.

3. SE: acts as tendon.

Each of these elements has a contribution in generating the total muscle force. To present a general model for all muscles, each of these forces and element lengths are normalized with respect to the maximum isometric muscle force (F^{iso}) and the muscle optimal length (L_0^{M}), at which muscle exerts maximum isometric force, respectively. In upcoming subsections, force contributions of these elements, considering the Thelen muscle model and its updates, will be discussed.

B.1 Thelen muscle model

Most of the material in this section is adopted from [283].

B.1.1 Series-elastic element force contribution

The SE's normalized force in terms of tendon strain has the following equation:

$$f^{\mathrm{T}} = \begin{cases} \frac{f_{toe}^{\mathrm{T}}\left(\exp\left(k_{toe}\hat{\varepsilon}^{\mathrm{T}}\right)-1\right)}{\exp(k_{toe})-1} & \hat{\varepsilon}^{\mathrm{T}} \leq 1 \\ k_{lin}\varepsilon_{toe}^{\mathrm{T}}\left(\hat{\varepsilon}^{\mathrm{T}}-1\right)+f_{toe}^{\mathrm{T}} & \hat{\varepsilon}^{\mathrm{T}} > 1 \end{cases} \tag{B.2}$$

where:

- $\hat{\varepsilon}^{\mathrm{T}} = \frac{\varepsilon^{\mathrm{T}}}{\varepsilon^{\mathrm{T}}_{toe}}$,
- $f^{\mathrm{T}} = \frac{F^{\mathrm{T}}}{F^{iso}}$ is the tendon force normalized to the maximum isometric muscle force,
- $\varepsilon^{\mathrm{T}} = \frac{L^{\mathrm{T}} - L^{\mathrm{T}}_s}{L^{\mathrm{T}}_s} = \frac{l^{\mathrm{T}} L^{\mathrm{M}}_0}{L^{\mathrm{T}}_s} - 1$ is the tendon strain (change in tendon length to its slack length ratio),
- $\varepsilon^{\mathrm{T}}_0$ is the tendon strain due to maximum isometric force, which is 0.04 in old adults (this value is also used for young adult people),
- $k_{toe} = 3$ is an exponential shape factor,
- $k_{lin} = \frac{f^{\mathrm{T}}_{toe} \exp(k_{toe})(k_{toe}-1)+\exp(k_{toe})+f^{\mathrm{T}}_{toe}-1}{\varepsilon^{\mathrm{T}}_{toe}(\exp(k_{toe})-1)} \cong \frac{1.712}{\varepsilon^{\mathrm{T}}_0}$ is a linear scale factor,
- $f^{\mathrm{T}}_{toe} = 0.33$ is the normalized tendon force after which tendon exhibits its linear behavior, and
- $\varepsilon^{\mathrm{T}}_{toe} = \frac{f^{\mathrm{T}}_{toe} k_{toe} \exp(k_{toe})}{k_{lin}(\exp(k_{toe})-1)} \cong 0.609\varepsilon^{\mathrm{T}}_0$ is the tendon strain above which the tendon behaves as a linear spring.

B.1.2 Passive element force contribution

The PE's normalized force in terms of normalized muscle length has the following form:

$$f^{\mathrm{PE}} = \frac{\exp\left(\frac{k^{\mathrm{PE}}(l^{\mathrm{M}}-1)}{\varepsilon^{\mathrm{M}}_0}\right) - 1}{\exp(k^{\mathrm{PE}}) - 1} \tag{B.3}$$

here:

- $f^{\mathrm{PE}} = \frac{F^{\mathrm{PE}}}{F^{iso}}$ is the normalized passive muscle force,
- $k^{\mathrm{PE}} = 5$ is an exponential shape factor,
- $l^{\mathrm{M}} = \frac{L^{\mathrm{M}}}{L^{\mathrm{M}}_0}$ is the normalized muscle length, and
- $\varepsilon^{\mathrm{M}}_0 = l^{\mathrm{M}}_0 - 1$ is the passive muscle strain due to the maximum isometric force. Its value for old adults is 0.5 (for young adults, it is 0.6).

B.1.3 Contractile element force contribution

Force of this element is a function of activation, muscle length and contraction velocity. The equation for the normalized force of this element is as follows:

$$f^{CE} = \frac{F^{CE}}{F^{iso}} = a\ f_l^{\mathrm{CE}}\ f_v^{\mathrm{CE}} \tag{B.4}$$

where a is the activation of the muscle, f_l^{CE} is the active force-length scale factor, and f_v^{CE} is the active muscle force due to the contraction velocity. Below, these last two terms are defined.

Active force-length scale factor

This scale factor is expressed by the following equation:

$$f_l^{\mathrm{CE}} = \exp\left(-\frac{\left(l^{\mathrm{M}}-1\right)^2}{\gamma^{\mathrm{CE}}}\right) \tag{B.5}$$

in which $\gamma^{\mathrm{CE}} = 0.45$ is the shape factor.

Active muscle force due to the contraction velocity

There is a nonlinear relationship between active muscle force and the contraction velocity, which is given by the following equation:

$$f_v^{\mathrm{CE}} = \begin{cases} \frac{A_f\left(v^{\mathrm{M}}+(0.25+0.75a)v_{Max}^{\mathrm{M}}\right)}{A_f\left(v^{\mathrm{M}}+(0.25+0.75a)v_{Max}^{\mathrm{M}}\right)-v^{\mathrm{M}}} & v^{\mathrm{M}} \le 0 \\ \frac{v^{\mathrm{M}}f_{len}^{\mathrm{M}}\left(2+\frac{2}{A_f}\right)+(0.25+0.75a)\left(f_{len}^{\mathrm{M}}-1\right)v_{Max}^{\mathrm{M}}}{v^{\mathrm{M}}\left(2+\frac{2}{A_f}\right)+(0.25+0.75a)\left(f_{len}^{\mathrm{M}}-1\right)v_{Max}^{\mathrm{M}}} & v^{\mathrm{M}} > 0 \end{cases} \tag{B.6}$$

here:

- f_{len}^{M} is the maximum normalized muscle force achievable when the fiber is lengthening. Its value for old adults is 1.8 (this value for young adults is 1.4).
- $A_f = 0.25$ is the force velocity shape factor,

- v^{M} is the contraction velocity of the muscle, which has the unit $\frac{\left[L_0^{\mathrm{M}}\right]}{s}$, and
- v_{Max}^{M} is the maximum contraction velocity, which is $8\,\frac{\left[L_0^{\mathrm{M}}\right]}{s}$ for old adults (for young adults, it is $10\,\frac{\left[L_0^{\mathrm{M}}\right]}{s}$).

B.2 Modified Thelen muscle model

Modification of the Thelen muscle has been done to overcome numerical issues in solving the muscle redundancy problem. Each of the muscle element equations have been updated in the following subsections.

B.2.1 Series-elastic element force contribution

The SE's normalized force in terms of tendon strain has the following updated equation [134]:

$$f^{\mathrm{T}} = 0.001(1+\varepsilon^{\mathrm{T}}) + \begin{cases} 0 & \hat{\varepsilon}^{\mathrm{T}} \leq 0 \\ \frac{f_{toe}^{\mathrm{T}}\left(\exp\left(k_{toe}\hat{\varepsilon}^{\mathrm{T}}\right)-1\right)}{\exp(k_{toe})-1} & 0 < \hat{\varepsilon}^{\mathrm{T}} \leq 1 \\ k_{lin}\varepsilon_{toe}^{\mathrm{T}}\left(\hat{\varepsilon}^{\mathrm{T}}-1\right) + f_{toe}^{\mathrm{T}} & \hat{\varepsilon}^{\mathrm{T}} > 1 \end{cases} \tag{B.7}$$

B.2.2 Passive element force contribution

The PE's normalized force in terms of normalized muscle length has the following updated form [134]:

$$f^{\mathrm{PE}} = \begin{cases} 1 + \frac{k^{\mathrm{PE}}}{\varepsilon_0^{\mathrm{M}}}\left(l^{\mathrm{M}} - (1+\varepsilon_0^{\mathrm{M}})\right) & l^{\mathrm{M}} > 1+\varepsilon_0^{\mathrm{M}} \\ \frac{\exp\left(\frac{k^{\mathrm{PE}}(l^{\mathrm{M}}-1)}{\varepsilon_0^{\mathrm{M}}}\right)-1}{\exp(k^{\mathrm{PE}})-1} & l^{\mathrm{M}} \leq 1+\varepsilon_0^{\mathrm{M}} \end{cases} \tag{B.8}$$

B.2.3 Contractile element force contribution

For the modified Thelen muscle model, the contractile element velocity in terms of contractile element force has the following equation [130, 134]:

$$\frac{v^{\mathrm{M}}}{v^{\mathrm{M}}_{Max}} = \begin{cases} b_1^{\mathrm{CE}}\left(1+\frac{1}{A_f}\right) f^{CE} f_l + c_1^{\mathrm{CE}} & F^{CE} < -F_a A_f \\ \frac{F^{CE}(0.25+0.75a)}{\varepsilon^{\mathrm{CE}}}\left(\frac{\varepsilon^{\mathrm{CE}}-F_a}{F_a+\frac{\varepsilon^{\mathrm{CE}}}{A_f}+\xi^{\mathrm{CE}}}+\frac{F_a}{F_a+\xi^{\mathrm{CE}}}\right)-\frac{F_a(0.25+0.75a)}{F_a+\xi^{\mathrm{CE}}} & -F_a A_f \le F^{CE} < 0 \\ \frac{\left(F^{CE}-F_a\right)(0.25+0.75a)}{F_a+\frac{F^{CE}}{A_f}+\xi^{\mathrm{CE}}} & 0 \le F^{CE} < F_a \\ \frac{\left(F^{CE}-F_a\right)(0.25+0.75a)}{\frac{\left(2+\frac{2}{A_f}\right)\left(F_a f^{\mathrm{M}}_{len}-F^{CE}\right)}{f^{\mathrm{M}}_{len}-1}+\xi^{\mathrm{CE}}} & F_a \le F^{CE} < 0.95 F_a f^{\mathrm{M}}_{len} \\ \left[b_2^{\mathrm{CE}}+\frac{\left(F^{CE}-0.95F_a f^{\mathrm{M}}_{len}\right)\left(c_2^{\mathrm{CE}}-b_2^{\mathrm{CE}}\right)}{\varepsilon^{\mathrm{CE}} F_a f^{\mathrm{M}}_{len}}\right](0.25+0.75a) & 0.95F_a f^{\mathrm{M}}_{len} \le F^{CE} \end{cases} \tag{B.9}$$

where:

- $F_a = a\, F^{iso} f_l$ is the active force in the muscle fiber,
- $b_1^{\mathrm{CE}} = -100$
- $c_1^{\mathrm{CE}} = 25$,
- $\varepsilon^{\mathrm{CE}} = 10^{-6}$,
- $\xi^{\mathrm{CE}} = 0.05$,
- $b_2^{\mathrm{CE}} = \frac{0.95F_a f^{M}_{len}-F_a}{\frac{0.05\left(2+\frac{2}{A_f}\right)F_a f^{M}_{len}}{f^{M}_{len}-1}+\xi^{\mathrm{CE}}}$, and
- $c_2^{\mathrm{CE}} = \frac{\left(0.95-\varepsilon^{\mathrm{CE}}\right)F_a f^{M}_{len}-F_a}{\frac{\left(0.05-\varepsilon^{\mathrm{CE}}\right)\left(2+\frac{2}{A_f}\right)F_a f^{M}_{len}}{f^{M}_{len}-1}+\xi^{\mathrm{CE}}}$.

Equation (B.9) is the inverse of the force-velocity function, where the input is the CE's force and the output is the muscle velocity. However, in most musculoskeletal simulations

the force-velocity relationship is needed to evaluate the CE's force with known muscle velocity. Thus, the following equation can be used:

$$f_v^{\mathrm{CE}} = \begin{cases} \frac{v^{\mathrm{M}}+1}{1+\frac{1}{A_f}} & v^{\mathrm{M}} < -1 \\ \frac{v^{\mathrm{M}}+1}{1-\frac{v^{\mathrm{M}}}{A_f}} & -1 \le v^{\mathrm{M}} < 0 \\ \frac{\left(2+\frac{2}{A_f}\right)v^{\mathrm{M}} f_{len}^{\mathrm{M}} + f_{len}^{\mathrm{M}} - 1}{\left(2+\frac{2}{A_f}\right)v^{\mathrm{M}} + f_{len}^{\mathrm{M}} - 1} & 0 \le v^{\mathrm{M}} < \frac{10(f_{len}^{\mathrm{M}}-1)(0.95 f_{len}^{\mathrm{M}}-1)}{\left(1+\frac{1}{A_f}\right) f_{len}^{\mathrm{M}}} \\ \frac{f_{len}^{\mathrm{M}}}{20(f_{len}^{\mathrm{M}}-1)}\left(\frac{\left(1+\frac{1}{A_f}\right)v^{\mathrm{M}} f_{len}^{\mathrm{M}}}{10(f_{len}^{\mathrm{M}}-1)} + 18.05 f_{len}^{\mathrm{M}} - 18\right) & \frac{10(f_{len}^{\mathrm{M}}-1)(0.95 f_{len}^{\mathrm{M}}-1)}{\left(1+\frac{1}{A_f}\right) f_{len}^{\mathrm{M}}} < v^{\mathrm{M}} \end{cases} \tag{B.10}$$

B.3 High level equations of a musculotendon

B.3.1 Muscle length and velocity

With known musculotendon length, muscle length can be evaluated using equations below:

$$\begin{cases} L^{\mathrm{S}} = L^{\mathrm{MT}} - L^{\mathrm{T}} \\ L^{\mathrm{M}} = \sqrt{(L^{\mathrm{S}})^2 + (\mathrm{w}^{\mathrm{M}})^2} \end{cases} \tag{B.11}$$

where all variables are defined in Fig. B.2, and w^{M} is given in (B.1). Musculotendon length (which is shown by L in the main text) is evaluated from musculoskeletal model kinematics. Muscle velocity is the time derivative of muscle length:

$$\begin{cases} V^{\mathrm{M}} = \frac{dL^{\mathrm{M}}}{dt} \\ v^{\mathrm{M}} = \frac{V^{\mathrm{M}}}{L_0^{\mathrm{M}}} \end{cases} \tag{B.12}$$

B.3.2 Musculotendon force

The force equilibrium equation for the 3-element muscle is given by:

$$F^{\mathrm{S}} = \left(F^{\mathrm{CE}} + F^{\mathrm{PE}}\right)\cos(\alpha^{\mathrm{M}}) = F^{\mathrm{T}} \tag{B.13}$$

where:

$$\cos(\alpha^{\mathrm{M}}) = \frac{L^{\mathrm{S}}}{L^{\mathrm{M}}} \tag{B.14}$$

Note that by using (B.1):

$$\alpha^{\mathrm{M}} = \begin{cases} 0 & L^{\mathrm{M}} = 0 \text{ or } \frac{\mathrm{w}^{\mathrm{M}}}{L^{\mathrm{M}}} \leq 0 \\ \arcsin\left(\frac{\mathrm{w}^{\mathrm{M}}}{L^{\mathrm{M}}}\right) & 0 < \frac{\mathrm{w}^{\mathrm{M}}}{L^{\mathrm{M}}} < 1 \\ \pi/2 & 1 \leq \frac{\mathrm{w}^{\mathrm{M}}}{L^{\mathrm{M}}} \end{cases} \tag{B.15}$$

Musculotendon force can be evaluated by:

$$F^{\mathrm{MT}} = F^{\mathrm{M}} = F^{\mathrm{S}} = \left(F^{\mathrm{CE}} + F^{\mathrm{PE}}\right)\cos(\alpha^{\mathrm{M}}) \tag{B.16}$$

and if there is no PE, then:

$$F^{\mathrm{MT}} = F^{\mathrm{M}} = F^{\mathrm{S}} = F^{\mathrm{CE}}\cos(\alpha^{\mathrm{M}}) \tag{B.17}$$

B.4 Activation dynamics

The time-derivative of the muscle activation is related to the muscle excitation by the following equation:

$$\dot{a} = \begin{cases} \frac{(e-a)}{t_{act}(0.5+1.5a)} & e > a \\ \frac{(e-a)(0.5+1.5a)}{t_{deact}} & e \leq a \end{cases} \tag{B.18}$$

where $t_{act} = 15$ ms, and t_{deact} is 50 or 60 ms for young or old adults, respectively. e is the muscle excitation.