ADAPTIVE ROBUST IMPEDANCE CONTROL AND NEURAL NETWORK BASED CONTROL FOR TELEREHABILITATION WITH UPPER-LIMB ROBOTIC EXOSKELETONS

by

Georgeta Bauer

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy

at

Dalhousie University
Halifax, Nova Scotia
May 2021

Table of Contents

List of Tables . . . . . **vi**

List of Figures . . . . . **viii**

Abstract . . . . . **xviii**

List of Abbreviations and Symbols Used . . . . . **xviii**

Acknowledgements . . . . . **xxv**

Chapter 1 Introduction **1**

1.1 Background on Rehabilitation 3
1.1.1 Human Arm Movement and Upper-Limb Muscle Function Impairments 3
1.1.2 Upper-Limb Conventional Rehabilitation and Rehabilitation with Robotic Devices 5

1.2 Research Motivation and Objective 12

1.3 Contributions 13

1.4 Thesis Organization 16

Chapter 2 Literature Review **17**

2.1 Upper-limb Robotic Exoskeleton Control Methods 18
2.1.1 Model-based Control Methods 19
2.1.2 Model-free Control Methods 21

2.2 Modality for Safe Motion and Adjustable Compliance 23

2.3 Telerehabilitation 24
2.3.1 Telepresence 25
2.3.2 Communication Time Delays 26

Chapter 3 Background Theories **29**

3.1 Fundamental Robotics Concepts 29
3.1.1 Kinematics 29
3.1.2 Dynamics 31

3.2 System Stability Analysis 32
3.2.1 Lyapunov's Direct Method 33

3.2.2 Barbălat's Lemma 33

Chapter 4 Problem Formulation 35

4.1 Problem Scope 35

4.2 System Overview 35

4.3 System Models 36
4.3.1 Robotic Exoskeleton 37
4.3.2 Patient Dynamic Model 37
4.3.3 Physical Human-Machine Interaction (pHMI) 38
4.3.4 Therapist Dynamic Model 42

4.4 Robotic Setup and Models for Validation 42
4.4.1 1-DOF Quanser QUBE for Experimental Validation 43
4.4.2 2-DOF Upper-Limb Robotic Exoskeleton 47

4.5 Telerehabilitation 50
4.5.1 Unilateral Teleoperation 50
4.5.2 Bilateral Teleoperation 51

4.6 Performance Metrics of the Control Methods 51
4.6.1 Tracking Performance 51
4.6.2 Telepresence Fidelity 52

4.7 Control Objectives 52

Chapter 5 Adaptive Robust Impedance Integral (ARII) Controller-Design, Simulations, and Experiments 54

5.1 Adaptive Control 54

5.2 Impedance Control 55

5.3 ARII Controller Design 58
5.3.1 ARII Controller Theoretical Development 59
5.3.2 Stability Proof 62

5.4 ARII Method Validation with the Single-System 1-DOF Quanser QUBE Robot Testbed 65
5.4.1 Simulations Validation with the Quanser QUBE Robot 65
5.4.2 Experimental Validation with the Quanser QUBE Robot . . . 69

5.5 ARII Method Validation with the Single-System 2-DOF Robotic Exoskeleton Simulations 71

5.6 Telerehabilitation Human Torque Synthesis Experiments with the Quanser QUBEs Testbed . . . 75
5.6.1 Direct $\boldsymbol{\tau}_{refl}$ Addition . . . 76
5.6.2 Scaled $\boldsymbol{\tau}_{refl}$ Addition . . . 79
5.6.3 Human Torque Regulator . . . 81

5.7 Telerehabilitation Human Torque Combination Simulations with 2-DOF Robotic Exoskeleton Model . . . 85
5.7.1 Direct $\boldsymbol{\tau}_{refl}$ Addition . . . 86
5.7.2 Scaled $\boldsymbol{\tau}_{refl}$ Addition . . . 89
5.7.3 Human Torque Regulator . . . 91

5.8 Discussions . . . 94

5.9 Contributions . . . 94

Chapter 6 Adaptive Robust Integral Radial Basis Function Neural Networks Impedance (RBFNN-I) Controller - Design, Simulations, Experiments, and Comparison . . . 96

6.1 Radial Basis Function Neural Networks Control . . . 96

6.2 Adaptive Robust Integral Radial Basis Function Neural Networks with Impedance (RBFNN-I) Controller Design . . . 97
6.2.1 RBFNN-I Controller Theoretical Development . . . 99
6.2.2 Stability Proof . . . 102

6.3 RBFNN-I Method Validation with the Single-System 1-DOF Quanser QUBE Robot Testbed . . . 105
6.3.1 Simulations Validation with the Quanser QUBE Robot . . . 105
6.3.2 Experimental Validation with the Quanser QUBE Robot . . . 108
6.3.3 RBFNN-I Method Validation with 2-DOF Robotic Exoskeleton Simulations . . . 111

6.4 Telerehabilitation Human Torque Synthesis Experiments with Quanser QUBEs . . . 115
6.4.1 Direct $\boldsymbol{\tau}_{refl}$ Addition . . . 116
6.4.2 Scaled $\boldsymbol{\tau}_{refl}$ Addition . . . 118
6.4.3 Human Torque Regulator . . . 120

6.5 Telerehabilitation Human Torque Combination Simulations with 2-DOF Robotic Exoskeleton Model . . . 123
6.5.1 Direct $\boldsymbol{\tau}_{refl}$ Addition . . . 123
6.5.2 Scaled $\boldsymbol{\tau}_{refl}$ Addition . . . 126
6.5.3 Human Torque Regulator . . . 128

6.6 Discussion . . . 133
6.7 Contributions . . . 133

Chapter 7 ARII and RBFNN-I Telerehabilitation Case Studies and Comparison . . . 134

7.1 Quanser QUBE Experimental Case Studies and Comparison . . . 134
7.2 2-DOF Exoskeleton Simulation Case Studies and Comparison . . . 140
7.3 RBFNN-I Experimental Range of Operation Exploration . . . 144
7.4 Discussion . . . 146

Chapter 8 Conclusions and Future Work . . . 147

8.1 Conclusions . . . 147
8.2 Future Work . . . 148

Bibliography . . . 150

Appendix A ARII and RBFNN-I Case Studies Plots . . . 165

A.1 Quanser QUBE Experimental Case Studies . . . 165
A.1.1 Quanser QUBE Experimental Case Studies with ARII Control Methods . . . 165
A.1.2 Quanser QUBE Experimental Tests Case Studies with RBFNN-I Control Methods . . . 170

A.2 2-DOF Exoskeleton Simulation Case Studies . . . 175
A.2.1 2-DOF Exoskeleton Simulation Case Studies with ARII Control Methods . . . 175
A.2.2 2-DOF Exoskeleton Simulation Case Studies with RBFNN-I Control Methods . . . 185

Appendix B Author's Publications . . . 195

List of Tables

4.1 Simulation Parameters 49

5.1 Simulations and Experiments Regions for Single-System Quanser QUBE 66

5.2 ARII QUBE Simulation Parameters 67

5.3 Performance Metrics for ARII Single-System Simulation with Quanser QUBE 68

5.4 Performance Metrics for ARII Single-System Experiment with Quanser QUBE 70

5.5 Performance Metrics for ARII Single-System Simulation versus Experiment with Quanser QUBE 71

5.6 Regions Definition for Single-System 2-DOF Exoskeleton Simulations 72

5.7 ARII Single-System Simulation with 2-DOF Exoskeleton: Simulation Parameters 73

5.8 Performance Metrics for ARII Single-System Simulation with the 2-DOF Exoskeleton 75

5.9 Regions Definition for Quanser QUBE Telerehabilitation Experiments 76

5.10 Performance Metrics for ARII Telerehabilitation Experiments with Quanser QUBEs 85

5.11 Regions Definition for Single-System 2-DOF Exoskeleton Simulations 85

5.12 Performance Metrics for ARII Telerehabilitation Simulations with 2-DOF Exoskeleton 94

6.1 RBFNN-I QUBE Simulation Parameters 106

6.2 Performance Metrics for ARII and RBFNN-I Single-System Simulation with Quanser QUBE 107

6.3 Performance Metrics for ARII and RBFNN-I Single-System Experiment with Quanser QUBE 109

6.4 Performance Metrics for RBFNN-I Single-System Simulation versus Experiment with Quanser QUBE 111

6.5 RBFNN-I Single-System Simulation with 2-DOF Exoskeleton: Simulation Parameters 112

6.6 Performance Metrics for ARII and RBFNN-I 2-DOF Exoskeleton Single-System Simulation 113

6.7 Performance Metrics for RBFNN-I Telerehabilitation Experiments with Quanser QUBEs 121

6.8 Performance Metrics for ARII and RBFNN-I Quanser QUBEs Telerehabilitation Experiments, with α=1, α=3, and HTR . . . 122

6.9 Performance Metrics for RBFNN-I Telerehabilitation Simulations with 2-DOF Exoskeleton 130

6.10 2-DOF Exoskeleton Telerehabilitation Simulations, with α=1, α=3, and HTR 132

7.1 Quanser QUBE Telerehabilitation Experiments Case Study: Parameters 135

7.2 Quanser QUBE Telerehabilitation Experiments Case Study: Performance Metrics using ARII and RBFNN-I 139

7.3 2-DOF Exoskeleton Telerehabilitation Simulation Case Study: Parameters 141

7.4 2-DOF Exoskeleton Telerehabilitation Simulations Case Study: Performance Metrics for Joint 1 using ARII and RBFNN-I . . . 142

7.5 2-DOF Exoskeleton Telerehabilitation Simulations Case Study: Performance Metrics for Joint 2 using ARII and RBFNN-I . . . 143

7.6 Quanser QUBE Telerehabilitation Experiments with Different Motion Frequencies using RBFNN-I: Tracking Error Metrics . . 144

List of Figures

1.1 Percentage of physiotherapists whose July 2020 work hours were lower than before the pandemic [21] 3

1.2 Top view of osteokinematic elbow and wrist extension/flexion movements 4

1.3 Upper-limb rehabilitation robotic devices 10

1.4 Commercial upper-limb rehabilitation robotic devices 11

2.1 Market forecast for robotic exoskeletons in all fields [96] 18

2.2 Block diagram of a closed-loop control system 19

2.3 Schematic diagram of an adaptive controller for exoskeleton devices 20

2.4 Schematic diagram of an SMC controller for exoskeletons . . . 21

2.5 PID controller 22

2.6 RBFNN controller, based on [73] 23

2.7 Telerehabilitation system 25

3.1 Top view diagram of a 2-DOF horizontally-planar robot 30

4.1 Overview of a telerehabilitation system showing the patient on the master side and the therapist on the slave side 36

4.2 Kelvin-Voigt viscoelastic model 37

4.3 Three-legged race: two participants having their legs bound together and holding on to each other to move as one 39

4.4 Exoskeleton diagram 39

4.5 Human arm diagram 40

4.6 The Quanser QUBE setup 43

4.7 Quanser QUBE model identification (a) experimental input: voltage, and (b) experimental output: angular position 44

4.8 Experimental versus simulated output data used to identify the model of the Quanser QUBE 45

4.9 Model validation showing (a) control input and (b) position output of the simulated versus experimental closed-loop system using the RBFNN-I control method 46

4.10 OptoForce force sensor 47

4.11 Experimental and simulated force sensor data 47

4.12 2-DOF elbow-wrist robotic exoskeleton conceptual design used in simulations 48

4.13 OptoForce force sensor (yellow) in the hand sub-assembly . . . 49

5.1 Designed impedance response to a 1 Nm step input 56

5.2 Joint angular position and tracking errors 57

5.3 ARII control system diagram 58

5.4 ARII single-system simulation with Quanser QUBE: (a) position trajectory, (b) tracking error, and (c) torque 67

5.5 Estimated parameters using the ARII control method with the Quanser QUBE single-system simulator 68

5.6 ARII single-system experiment with Quanser QUBE: (a) position trajectory, (b) tracking error, and (c) torque 69

5.7 Estimated parameters using the ARII control method with the Quanser QUBE single-system apparatus 70

5.8 The profile of $\tilde{\xi}$ for ARII Quanser QUBE simulation and experiment 70

5.9 ARII single-system simulation with 2-DOF exoskeleton: (a) position trajectory, (b) tracking error, and (c) patient torque for joint 1 74

5.10 ARII single-system simulation with 2-DOF exoskeleton: (a) position trajectory, (b) tracking error, and (c) patient torque for joint 2 74

5.11 Estimated parameters using the ARII control method with the 2-DOF exoskeleton simulator 75

5.12 ARII telerehabilitation experiment with Quanser QUBEs with $\alpha = 1$: (a) position trajectory, (b) tracking error, and (c) torques 77

5.13 Estimated parameters for the ARII telerehabilitation experiment with the Quanser QUBEs with $\alpha = 1$ 78

5.14 Torques for the ARII telerehabilitation experiment with Quanser QUBEs with $\alpha = 1$ during the bilateral phase 78

5.15 ARII telerehabilitation experiment with Quanser QUBEs with $\alpha = 3$: (a) position trajectory, (b) tracking error, and (c) torques 80

5.16 Estimated parameters for the ARII telerehabilitation experiment with the Quanser QUBEs with $\alpha = 3$ 80

5.17 Torques for the ARII telerehabilitation experiment with Quanser QUBEs with $\alpha = 3$ during the bilateral phase 81

5.18 ARII telerehabilitation experiment with Quanser QUBEs with HTR: (a) position trajectory, (b) tracking error, and (c) torques 83

5.19 Estimated parameters for the ARII telerehabilitation experiment with the Quanser QUBEs with HTR 83

5.20 Torques for the ARII telerehabilitation experiment with Quanser QUBEs with HTR during the bilateral phase 84

5.21 Joint 1 ARII 2-DOF telerehabilitation simulation with $\alpha = 1$: (a) position trajectory, (b) tracking error, and (c) torques . . . 86

5.22 Joint 1 torques for the ARII 2-DOF telerehabilitation simulation with $\alpha = 1$ during the bilateral phase 87

5.23 Joint 2 ARII 2-DOF telerehabilitation simulation with $\alpha = 1$: (a) position trajectory, (b) tracking error, and (c) torques . . . 88

5.24 Joint 2 torques for the ARII 2-DOF telerehabilitation simulation with $\alpha = 1$ during the bilateral phase 88

5.25 Joint 1 ARII 2-DOF telerehabilitation simulation with $\alpha = 3$: (a) position trajectory, (b) tracking error, and (c) torques . . . 89

5.26 Joint 1 torques for the ARII 2-DOF telerehabilitation simulation with $\alpha = 3$ during the bilateral phase 90

5.27 Joint 2 ARII 2-DOF telerehabilitation simulation with $\alpha = 3$: (a) position trajectory, (b) tracking error, and (c) torques . . . 90

5.28 Joint 2 torques for the ARII 2-DOF telerehabilitation simulation with $\alpha = 3$ during the bilateral phase 91

5.29 Joint 1 ARII 2-DOF telerehabilitation simulation with HTR: (a) position trajectory, (b) tracking error, and (c) torques . . . 92

5.30 Joint 1 torques for the ARII 2-DOF telerehabilitation simulation with HTR during the bilateral phase 92

5.31 Joint 2 ARII 2-DOF telerehabilitation simulation with HTR: (a) position trajectory, (b) tracking error, and (c) torques . . . 93

5.32 Joint 2 torques for the ARII 2-DOF telerehabilitation simulation with HTR during the bilateral phase 93

6.1 Radial basis function neural network structure 97

6.2 RBFNN-I control system diagram 98

6.3 RBFNN-I single-system simulation with Quanser QUBE: (a) position trajectory, (b) tracking error, and (c) patient torque . 107

6.4 Estimated parameters using the RBFNN-I control method with the Quanser QUBE single-system simulator 107

6.5 The profile of $\tilde{\xi}$ for ARII versus RBFNN-I Quanser QUBE simulation 108

6.6 RBFNN-I single-system experiment with Quanser QUBE: (a) position trajectory, (b) tracking error, and (c) patient torque . 109

6.7 Estimated parameters using the RBFNN-I control method with the Quanser QUBE single-system apparatus 109

6.8 The profile of $\tilde{\xi}$ for the ARII and the RBFNN-I Quanser QUBE experiments 110

6.9 The profile of $\tilde{\xi}$ for the RBFNN-I Quanser QUBE simulation and experiment 110

6.10 RBFNN-I single-system 2-DOF Simulation: (a) position trajectory, (b) tracking error, and (c) patient torque for joint 1 . . . 113

6.11 RBFNN-I single-system 2-DOF simulation: (a) position trajectory, (b) tracking error, and (c) patient torque for joint 2 . . . 114

6.12 Estimated parameters for the RBFNN-I single-system 2-DOF exoskeleton aimulation 114

6.13 The profile of $\tilde{\xi}$ for the ARII and the RBFNN-I 2-DOF exoskeleton simulation joint 1 115

6.14 The profile of $\tilde{\xi}$ for the ARII and the RBFNN-I 2-DOF exoskeleton simulation Joint 2 115

6.15 RBFNN-I telerehabilitation experiment with Quanser QUBEs with $\alpha = 1$: (a) position trajectory, (b) tracking error, and (c) torques 116

6.16 Estimated parameters for the RBFNN-I telerehabilitation experiment with Quanser QUBEs with $\alpha = 1$ 117

6.17 Torques for the RBFNN-I telerehabilitation experiment with Quanser QUBEs with $\alpha = 1$ during the bilateral phase 117

6.18 RBFNN-I telerehabilitation experiment with Quanser QUBEs with $\alpha = 3$: (a) position trajectory, (b) tracking error, and (c) torques 118

6.19 Estimated parameters for the RBFNN-I telerehabilitation experiment with Quanser QUBEs with $\alpha = 3$ 119

6.20 Torques for the RBFNN-I telerehabilitation experiment with Quanser QUBEs with $\alpha = 3$ during the bilateral phase 119

6.21 RBFNN-I telerehabilitation experiment with Quanser QUBEs with HTR: (a) position trajectory, (b) tracking error, and (c) torques 120

6.22 Estimated parameters for the RBFNN-I telerehabilitation experiment with Quanser QUBEs with HTR 121

6.23 Torques for the RBFNN-I telerehabilitation experiment with Quanser QUBEs with HTR during the bilateral phase 121

6.24 Joint 1 RBFNN-I 2-DOF telerehabilitation simulation with $\alpha = 1$: (a) position trajectory, (b) tracking error, and (c) torques . . . 124

6.25 Joint 1 torques for the RBFNN-I 2-DOF telerehabilitation simulation with $\alpha = 1$ during the bilateral phase 124

6.26 Joint 2 RBFNN-I 2-DOF telerehabilitation simulation with $\alpha = 1$: (a) position trajectory, (b) tracking error, and (c) torques . . . 125

6.27 Joint 2 torques for the RBFNN-I 2-DOF telerehabilitation simulation with $\alpha = 1$ during the bilateral phase 125

6.28 Joint 1 RBFNN-I 2-DOF telerehabilitation simulation with $\alpha = 3$: (a) position trajectory, (b) tracking error, and (c) torques . . . 126

6.29 Joint 1 torques for the RBFNN-I 2-DOF telerehabilitation simulation with $\alpha = 3$ during the bilateral phase 127

6.30 Joint 2 RBFNN-I 2-DOF telerehabilitation simulation with $\alpha = 3$: (a) position trajectory, (b) tracking error, and (c) torques . . . 127

6.31 Joint 2 torques for the RBFNN-I 2-DOF telerehabilitation simulation with $\alpha = 3$ during the bilateral phase 128

6.32 Joint 1 RBFNN-I 2-DOF telerehabilitation simulation with HTR: (a) position trajectory, (b) tracking error, and (c) torques . . . 129

6.33 Joint 1 torques for the RBFNN-I 2-DOF telerehabilitation simulation with HTR during the bilateral phase 130

6.34 Joint 2 RBFNN-I 2-DOF telerehabilitation simulation with HTR: (a) position trajectory, (b) tracking error, and (c) torques . . . 131

6.35 Joint 2 torques for the RBFNN-I 2-DOF telerehabilitation simulation with HTR during the bilateral phase 131

7.1 Baseline and case 1 position tracking 136

7.2 Designed impedance responses to a 1 Nm step input 136

7.3 Baseline and case 2 position tracking 137

7.4 Baseline and case 3 position tracking 137

7.5 Baseline and case 4 position tracking 138

7.6 Baseline and case 5 signal noise 138

7.7 Range of operation experimental test with Quanser QUBES, RBFNN-I controller, and w=$5\pi/8$ 145

7.8 Range of operation experimental test with Quanser QUBES, RBFNN-I controller, and ψ=0.001 146

A.1 Case 1 ARII telerehabilitation experiment with Quanser QUBEs using HTR: (a) position trajectory, (b) tracking error 165

A.2 Torques for case 1 ARII telerehabilitation experiment with Quanser QUBEs with HTR during the bilateral phase 165

A.3 Case 2 ARII telerehabilitation experiment with Quanser QUBEs using HTR: (a) position trajectory, (b) tracking error 166

A.4 Torques for case 2 ARII telerehabilitation experiment with Quanser QUBEs with HTR during the bilateral phase 166

A.5 Case 3 ARII telerehabilitation experiment with Quanser QUBEs using HTR: (a) position trajectory, (b) tracking error 167

A.6 Torques for case 3 ARII telerehabilitation experiment with Quanser QUBEs with HTR during the bilateral phase 167

A.7 Case 4 ARII telerehabilitation experiment with Quanser QUBEs using HTR: (a) position trajectory, (b) tracking error 168

A.8 Torques for case 4 ARII telerehabilitation experiment with Quanser QUBEs with HTR during the bilateral phase 168

A.9 Case 5 ARII telerehabilitation experiment with Quanser QUBEs using HTR: (a) position trajectory, (b) tracking error 169

A.10 Torques for case 5 ARII telerehabilitation experiment with Quanser QUBEs with HTR during the bilateral phase 169

A.11 Case 1 RBFNN-I telerehabilitation experiment with Quanser QUBEs using HTR: (a) position trajectory, (b) tracking error 170

A.12 Torques for case 1 RBFNN-I telerehabilitation experiment with Quanser QUBEs with HTR during the bilateral phase 170

A.13 Case 2 RBFNN-I telerehabilitation experiment with Quanser QUBEs using HTR: (a) position trajectory, (b) tracking error 171

A.14 Torques for case 2 RBFNN-I telerehabilitation experiment with Quanser QUBEs with HTR during the bilateral phase 171

A.15 Case 3 RBFNN-I telerehabilitation experiment with Quanser QUBEs using HTR: (a) position trajectory, (b) tracking error 172

A.16 Torques for case 3 RBFNN-I telerehabilitation experiment with Quanser QUBEs with HTR during the bilateral phase 172

A.17 Case 4 RBFNN-I telerehabilitation experiment with Quanser QUBEs using HTR: (a) position trajectory, (b) tracking error 173

A.18 Torques for case 4 RBFNN-I telerehabilitation experiment with Quanser QUBEs with HTR during the bilateral phase 173

A.19 Case 5 RBFNN-I telerehabilitation experiment with Quanser QUBEs using HTR: (a) position trajectory, (b) tracking error 174

A.20 Torques for case 5 RBFNN-I telerehabilitation experiment with Quanser QUBEs with HTR during the bilateral phase 174

A.21 Joint 1, Case 1 ARII telerehabilitation experiment with 2-DOF Exoskeletons using HTR: (a) position trajectory, (b) tracking error 175

A.22 Joint 1 torques for case 1 ARII telerehabilitation 2-DOF simulation with HTR during the bilateral phase 175

A.23 Joint 2, Case 1 ARII telerehabilitation experiment with 2-DOF Exoskeletons using HTR: (a) position trajectory, (b) tracking error 176

A.24 Joint 2 torques for case 1 ARII telerehabilitation 2-DOF simulation with HTR during the bilateral phase 176

A.25 Joint 1, Case 2 ARII telerehabilitation experiment with 2-DOF Exoskeletons using HTR: (a) position trajectory, (b) tracking error 177

A.26 Joint 1 torques for case 2 ARII telerehabilitation 2-DOF simulation with HTR during the bilateral phase 177

A.27 Joint 2, Case 2 ARII telerehabilitation experiment with 2-DOF Exoskeletons using HTR: (a) position trajectory, (b) tracking error 178

A.28 Joint 2 torques for case 2 ARII telerehabilitation 2-DOF simulation with HTR during the bilateral phase 178

A.29 Joint 1, Case 3 ARII telerehabilitation experiment with 2-DOF Exoskeletons using HTR: (a) position trajectory, (b) tracking error 179

A.30 Joint 1 torques for case 3 ARII telerehabilitation 2-DOF simulation with HTR during the bilateral phase 179

A.31 Joint 2, Case 3 ARII telerehabilitation experiment with 2-DOF Exoskeletons using HTR: (a) position trajectory, (b) tracking error 180

A.32 Joint 2 torques for case 3 ARII telerehabilitation 2-DOF simulation with HTR during the bilateral phase 180

A.33 Joint 1, Case 4 ARII telerehabilitation experiment with 2-DOF Exoskeletons using HTR: (a) position trajectory, (b) tracking error 181

A.34 Joint 1 torques for case 4 ARII telerehabilitation 2-DOF simulation with HTR during the bilateral phase 181

A.35 Joint 2, Case 4 ARII telerehabilitation experiment with 2-DOF Exoskeletons using HTR: (a) position trajectory, (b) tracking error 182

A.36 Joint 2 torques for case 4 ARII telerehabilitation 2-DOF simulation with HTR during the bilateral phase 182

A.37 Joint 1, Case 5 ARII telerehabilitation experiment with 2-DOF Exoskeletons using HTR: (a) position trajectory, (b) tracking error 183

A.38 Joint 1 torques for case 5 ARII telerehabilitation 2-DOF simulation with HTR during the bilateral phase 183

A.39 Joint 2, Case 5 ARII telerehabilitation experiment with 2-DOF Exoskeletons using HTR: (a) position trajectory, (b) tracking error 184

A.40 Joint 2 torques for case 5 ARII telerehabilitation 2-DOF simulation with HTR during the bilateral phase 184

A.41 Joint 1, Case 1 RBFNN-I telerehabilitation experiment with 2-DOF Exoskeletons using HTR: (a) position trajectory, (b) tracking error 185

A.42 Joint 1 torques for case 1 RBFNN-I telerehabilitation 2-DOF simulation with HTR during the bilateral phase 185

A.43 Joint 2, Case 1 RBFNN-I telerehabilitation experiment with 2-DOF Exoskeletons using HTR: (a) position trajectory, (b) tracking error 186

A.44 Joint 2 torques for case 1 RBFNN-I telerehabilitation 2-DOF simulation with HTR during the bilateral phase 186

A.45 Joint 1, Case 2 RBFNN-I telerehabilitation experiment with 2-DOF Exoskeletons using HTR: (a) position trajectory, (b) tracking error 187

A.46 Joint 1 torques for case 2 RBFNN-I telerehabilitation 2-DOF simulation with HTR during the bilateral phase 187

A.47 Joint 2, Case 2 RBFNN-I telerehabilitation experiment with 2-DOF Exoskeletons using HTR: (a) position trajectory, (b) tracking error 188

A.48 Joint 2 torques for case 2 RBFNN-I telerehabilitation 2-DOF simulation with HTR during the bilateral phase 188

A.49 Joint 1, Case 3 RBFNN-I telerehabilitation experiment with 2-DOF Exoskeletons using HTR: (a) position trajectory, (b) tracking error 189

A.50 Joint 1 torques for case 3 RBFNN-I telerehabilitation 2-DOF simulation with HTR during the bilateral phase 189

A.51 Joint 2, Case 3 RBFNN-I telerehabilitation experiment with 2-DOF Exoskeletons using HTR: (a) position trajectory, (b) tracking error 190

A.52 Joint 2 torques for case 3 RBFNN-I telerehabilitation 2-DOF simulation with HTR during the bilateral phase 190

A.53 Joint 1, Case 4 RBFNN-I telerehabilitation experiment with 2-DOF Exoskeletons using HTR: (a) position trajectory, (b) tracking error 191

A.54 Joint 1 torques for case 4 RBFNN-I telerehabilitation 2-DOF simulation with HTR during the bilateral phase 191

A.55 Joint 2, Case 4 RBFNN-I telerehabilitation experiment with 2-DOF Exoskeletons using HTR: (a) position trajectory, (b) tracking error 192

A.56 Joint 2 torques for case 4 RBFNN-I telerehabilitation 2-DOF simulation with HTR during the bilateral phase 192

A.57 Joint 1, Case 5 RBFNN-I telerehabilitation experiment with 2-DOF Exoskeletons using HTR: (a) position trajectory, (b) tracking error 193

A.58 Joint 1 torques for case 5 RBFNN-I telerehabilitation 2-DOF simulation with HTR during the bilateral phase 193

A.59 Joint 2, Case 5 RBFNN-I telerehabilitation experiment with 2-DOF Exoskeletons using HTR: (a) position trajectory, (b) tracking error 194

A.60 Joint 2 torques for case 5 RBFNN-I telerehabilitation 2-DOF simulation with HTR during the bilateral phase 194

Abstract

With the unprecedented increase in global senior population and the accompanying debilitating neurological diseases, there is a growing and unmet demand for physical rehabilitation. Additionally, the drastic reduction of in-person care and the increased physical and cerebrovascular injuries during the current pandemic is not only interrupting the essential continuum of physiotherapy, but also causing the divergence between the demand and the supply of the service to grow even more.

Telerehabilitation with robotic exoskeletons is an emerging, and compelling complementary rehabilitation modality which could help address the widening gap between the demand and the supply of physiotherapeutic services. Some of the prevailing challenges for the robot controllers are to overcome the effects of dynamic modeling uncertainties and ensure good tracking performance, stability, safe and compliant motion, and a high degree of telepresence between the two remotely-separated human-robot systems in the presence of nonlinearities, human torques, and communication constraints such as time delays.

To address these challenges, two control methods are further developed, implemented, and validated for telerehabilitation with upper-limb robotic exoskeletons: Adaptive Robust Integral Impedance model (ARII) control and Adaptive Robust Integral Radial Basis Function Neural Networks-based Impedance model (RBFNN-I) control. Both implementations have been extended to provide compliant behaviour using an adjustable impedance model controller and revealed desirable performance for the conditions used in this research. A salient contribution of this research is the creation and implementation of a novel human torque regulator (HTR) which was shown to provide higher fidelity telepresence for the therapist compared to existing methods to enhance the safety and perception of the closed-loop physical interaction. Experiments were performed using single-joint robots while simulations were carried out using two-degrees-of-freedom (2-DOF) exoskeletons models to validate the proposed controllers and advance the state-of-the-art control for telerehabilitation with upper-limb robotic exoskeletons.

List of Abbreviations and Symbols Used

Chapter 1

ADL	activities of daily living
ARII	adaptive robust integral impedance
COM	center of mass
COVID-19	coronavirus disease 2019
DOF	degree(s) of freedom
HTR	human torque regulator
PD	proportional derivative
PI	proportional integral
PID	proportional integral derivative
PPE	personal protective equipment
RBFNN	radial basis function neural network
RBFNN-I	adaptive robust integral radial basis function neural networks impedance
SMC	sliding mode control

Chapter 2

y	measured sensor value
y_{ref}	reference model output
u	controller torque
u_{sw}	switching controller torque
u_{nom}	nominal controller torque
K_P	proportional gain
$e(t)$	difference between the actual and desired joint angular position
$\dot{e}(t)$	first time derivative of $e(t)$
K_D	derivative gain
K_I	integral gain

K_P	proportional gain
t	time

Chapter 3

l_i	length of robot link i
l_{ci}	location of COM of robot link i
m_i	mass of robot link i
q_{ei}	joint angular position of exoskeleton joint i
$\boldsymbol{q}_e$	vector of exoskeleton joint angular positions
$\dot{\boldsymbol{q}}_e$	vector of exoskeleton joint angular velocities
$\ddot{\boldsymbol{q}}_e$	vector of exoskeleton joint angular accelerations
$\boldsymbol{q}_p$	vector of patient joint angular positions
$\dot{\boldsymbol{q}}_p$	vector of patient joint angular velocities
$\ddot{\boldsymbol{q}}_p$	vector of patient joint angular accelerations
x	x coordinate of the end-effector position
y	y coordinate of the end-effector position
$\mathcal{L}$	the Lagrangian
$\Re$	real domain
KE	kinetic energy
PE	potential energy
$\boldsymbol{u}$	vector of applied joint torques
v_{ci}	linear velocity of link i's COM
I_i	inertia of link i about link i's COM
M	robotic exoskeleton inertia matrix
C	Coriolis effects and centrifugal torques matrix
n	degrees of freedom of the robotic device
V	Lyapunov function
$\dot{V}$	first time derivative of the Lyapunov function
$\ddot{V}$	second time derivative of the Lyapunov function
$\boldsymbol{x}$	system state vector

Chapter 4

q_d	desired joint angular position
$\dot{q}_d$	desired joint angular velocity
$\ddot{q}_d$	desired joint angular acceleration
q_{desL}	q_d reflected to the slave side
$\dot{q}_{desL}$	$\dot{q}_d$ reflected to the slave side
τ_{net}	net human torque between the patient and the therapist
q_m	actual joint angular position of master exoskeleton
$\dot{q}_m$	actual joint angular velocity of master exoskeleton
u_m	master robotic exoskeleton controller torque output
q_{sd}	q_m reflected to the slave side
$\dot{q}_{sd}$	$\dot{q}_m$ reflected to the slave side
q_s	actual joint angular position of slave exoskeleton
$\dot{q}_s$	actual joint angular velocity slave exoskeleton
u_s	slave robotic exoskeleton controller torque
τ_t	therapist torque
τ_p	patient torque
τ_{refl}	reflected therapist torque
M_m	master side robotic exoskeleton inertia matrix
C_m	master side Coriolis effects and centrifugal torques matrix
M_s	slave side robotic exoskeleton inertia matrix
C_s	slave side Coriolis effects and centrifugal torques matrix
$\boldsymbol{\tau}_d$	external disturbance torques
M_p	patient inertia matrix
C_p	patient Coriolis effects and centrifugal torques matrix
$\boldsymbol{q}_o$	patient's resting reference joint angular positions vector
pHMI	physical human-machine (exoskeleton) interaction
k_i	stiffness of the viscoelastic robot-patient connection for joint i
d_i	viscous damping of the viscoelastic robot-patient connection for joint i
l_{si}	location along link i of force sensor i from joint i

Δs_i	arm position difference between the exoskeleton and the patient's arm at the location of force sensor i
$\Delta \dot{s}_i$	rate difference between the exoskeleton and the patient's arm at the location of force sensor i
F_i	patient-applied force at joint i
$\overrightarrow{\boldsymbol{F}}_i$	unit vector for F_i
$\boldsymbol{F}_i$	interaction force $F_i \overrightarrow{\boldsymbol{F}}_i$
v_x	end effector linear velocity in the x direction
v_y	end effector linear velocity in the y direction
J_i	Jacobian matrix for joint i
c_1	identified system model coefficient
c_2	identified system model coefficient
c_3	identified Coulomb friction model coefficient
c_{3cw}	identified clockwise Coulomb friction model coefficient
c_{3ccw}	identified counter-clockwise Coulomb friction model coefficient
$\tilde{\xi}_i$	joint angular position impedance error for joint i
$\tilde{\xi}_{i,RMSE}$	root mean squared error of $\tilde{\xi}_i$
N	number of data points considered
$\tilde{\xi}_{i,max}$	maximum value of $\tilde{\xi}_i$ for joint i
TP_{RMSE}	root mean square error of the difference between the patient and the therapist torques
TP_{max}	maximum value of the telepresence
τ_{p_i}	patient torque for joint i
τ_{refl_i}	reflected therapist's torque for joint i

Chapter 5

M_a	the adaptive control law's best approximations of the actual robotic exoskeleton M
C_a	the adaptive control law's best approximations of the actual robotic C
E_M	modeling errors between model M_a and actual system M
E_C	modeling errors between model C_a and actual system C

J_d	positive, diagonal inertia matrix
B_d	positive, diagonal damping matrix
K_d	positive, diagonal stiffness matrix
$\tilde{\boldsymbol{q}}_{\tau net}$	designed desired impedance joint angular position tracking error vector
$\dot{\tilde{\boldsymbol{q}}}_{\tau net}$	designed desired impedance joint angular velocity tracking error vector
$\ddot{\tilde{\boldsymbol{q}}}_{\tau net}$	designed desired impedance joint angular accel. tracking error vector
ω	natural frequency
ζ	damping ratio
$\boldsymbol{q}_{d,imp}$	impedance desired joint angular position vector
$\dot{\boldsymbol{q}}_{d,imp}$	impedance desired joint angular velocity
$\ddot{\boldsymbol{q}}_{d,imp}$	impedance desired joint angular acceleration vector
$\boldsymbol{q}_d$	original desired joint angular position trajectory vector
$\boldsymbol{q}_{d,imp}$	new desired joint angular position trajectory vector considering the impedance response from $\boldsymbol{\tau}_{net}$ vector
$\boldsymbol{q}$	actual joint angular position trajectory vector
$\tilde{\boldsymbol{q}}$	actual joint angular position tracking error from the original desired joint angular position trajectory vector
$\tilde{\boldsymbol{\xi}}$	joint angular position impedance error vector
$\dot{\tilde{\boldsymbol{\xi}}}$	joint angular velocity impedance error vector
$\hat{M}_a$	estimate of the inertia matrix M_a
$\hat{C}_a$	estimate of the Coriolis and centripetal torques C_a
K_p	diagonal, positive, definite proportional gain matrix
K_i	diagonal, positive, definite integral gain matrix
$\boldsymbol{\tau}_r$	robust control term vector
$\boldsymbol{\tau}_{p,meas}$	measured human disturbance torque vector
$\boldsymbol{v}$	virtual reference velocity vector
$\dot{\boldsymbol{v}}$	virtual reference acceleration vector
$\boldsymbol{r}$	sliding mode error vector
$\boldsymbol{p}$	vector of m model parameters
$\hat{\boldsymbol{p}}$	vector of m adaptive control law estimated parameters
Y	dynamic regressor matrix

Γ	symmetric positive definite matrix
$\mathcal{L}_2$	Euclidean norm
$\mathcal{L}_\infty^n$	∞ norm of a function
ϵ	smoothing term
$\between$	between
α	positive symmetric diagonal matrix containing the $\boldsymbol{\tau}_{refl}$-scaling factors
ψ	positive definite diagonal matrix, controls how quickly telepresence is achieved

Chapter 6

$\boldsymbol{c}_j$	center vector for neuron j
$\boldsymbol{x}$	input variable vector
ϕ	activation function
σ	width of the Gaussian function
w_{ji}	synaptic weights connecting the hidden layer to output neurons
$\boldsymbol{f}$	network output layer, sum of weighted outputs from the hidden layer
g	input variable matrix dimension
m	number of neurons in the hidden layer
d	number of output variables
M_{NN}	RBFNN-I's best approximations of the actual robotic exoskeleton M
C_{NN}	RBFNN-I's's best approximations of the actual robotic C
$\hat{M}_{NN}$	estimate of the inertia matrix M_{NN}
$\hat{C}_{NN}$	estimate of the Coriolis and centripetal torques C_{NN}
W_M	ideal RBFNN weight matrix
W_C	ideal RBFNN weight matrix
$\hat{W}_M$	estimate of W_M
$\hat{W}_C$	estimate of W_C
Φ_M	output vector of the hidden layers
Φ_C	output vector of the hidden layers

Acknowledgements

I would like to express my deep gratitude to my supervisor, Dr. Ya-Jun Pan, for her support and patience throughout this journey. I would also like to thank the members of the Supervisory Committee, Dr. Seto and Dr. Adamson, for their valuable feedback over the years. I would like to express my gratitude to Dr. Dubay for accepting to be the external examiner for my thesis defence.

Immense gratitude goes out to my husband, my son, my mom, my in-laws, my friends and relatives around the world, and my dog, for their unfaltering love and support.

I would like to thank my colleagues in the Advanced Control and Mechatronics Lab for their friendship, the staff and faculty of the Mechanical, Electrical, and Biomedical Engineering Departments at Dalhousie University, Physiotherapists Heather, Linda, and Michael, and Rehabilitation Engineer Kim from the Nova Scotia Rehabilitation Center.

I am grateful for the financial support from the Killam Trust Fund, the Dr. Eliza Ritchie Scholarship, the Nova Scotia Graduate Scholarship, and the Natural Sciences and Engineering Research Council of Canada (NSERC).

'Nothing splendid has ever been achieved except by those who dared believe that something inside them was superior to circumstance.'

$\sim$ *Bruce Barton*

Chapter 1

Introduction

The unprecedented growth in the worldwide senior population − which is expected to more than double by 2050 [1, 2] − and the surging survival rate with disabilities [3, 4, 5], is creating an escalating and unmet demand for physical rehabilitation [6]. Stroke in particular is the largest cause of disability in the USA [7, 8] and one of the leading causes of long-term disability worldwide [9, 10, 11, 12]. Before the pandemic, the demand for physical rehabilitation in the USA was estimated to be approximately 17% greater than what could be provided [13], and is expected to grow to 36% by 2030 [14]. Currently, about 1 billion people worldwide live with some sort of disability and the predictions indicate that 2.4 billion people will require physical rehabilitation in 2030 [2].

The COVID-19 pandemic has had a horrendous effect on the delivery of physiotherapy, thus far. Newly-discovered correlations between the mildly-symptomatic COVID-19 patients and incidence of major stroke [15, 16], as well increased physical injuries due to increased sedentariness, 'bad home-office setups', excessive exercising, or unsafe exercising [17] further escalate the demand for the service. Additionally, due to the surge in critically-ill COVID-19 patients requiring hospitalization, there is a scarcity of available hospital beds for patients suffering from other illnesses, including stroke. For example, at one point during the pandemic, hospitals in New York City discharged patients and stopped accepting new ones in their acute inpatient rehabilitation units [18]. Furthermore, there are many places around the world where insufficient safety precautions such as inadequate Personal Protective Equipment (PPE), reused PPE, and the lack of PPE for the physiotherapist put them at unacceptable risk while performing their jobs [19, 20]. Due to the concern of spreading or contracting the disease, especially if aerosol-generating procedures are required, many who need physiotherapy cancel or avoid in-person sessions potentially resulting in deteriorating health [21]. As a result, even though the need for physiotherapy continues to

escalate, the delivery of physiotherapy is reduced due to the effects of the pandemic. This reduction in physiotherapy can be seen in Fig. 1.1 which is from the "Impact of COVID-19 on the Physical Therapy Profession" report published in August 2020 by The American Physical Therapy Association. This figure shows the significant percentage of physiotherapists working in different environments whose July 2020 work hours were lower than before the pandemic. Furthermore, due to closure restrictions in different parts of the world, many clinics are left with providing support and instructions to their patients via a telephone helpline or video-consultations [22]. However, as one of the commentators on the article remarked: "What's the value of physical therapy without physical contact?" [22]. Many physiotherapists were also reassigned to COVID-19-related duties on the front line, or were layed off due to clinics closing − further reducing access to physiotherapy to many sufferers. It is abundantly clear that the COVID-19 pandemic is further exacerbating the demand [23] and preventing the delivery of physical rehabilitation in a safe, timely, uninterrupted, sufficient, and effective manner and can have a severe impact on the health and lives of millions of people.

Professor Christopher Murray, Director of the Institute for Health Metrics and Evaluation at the University of Washington, USA, has stated that: "as disability becomes an increasingly large share of the global disease burden and a larger component of health expenditure, there is an urgent and compelling need to identify new, more effective interventions" [10]. Clearly, alternatives are urgently needed to address this escalating issue. The physical rehabilitation of upper limbs, in particular, is paramount [24] to improve the quality of life and lessen personal and societal burdens incurred by neuromuscular dysfunctions.

This thesis presents research carried out on upper-limb robotic exoskeleton telerehabilitation control methods which builds on the state-of-the-art to hopefully enable the delivery of physiotherapy to more people in the future.

In this chapter, human arm movement and common muscle impairments are first presented. Next, upper-limb conventional rehabilitation and rehabilitation robotic devices are introduced. The thesis motivation, objective, and contributions are then presented followed by the organization of this thesis.

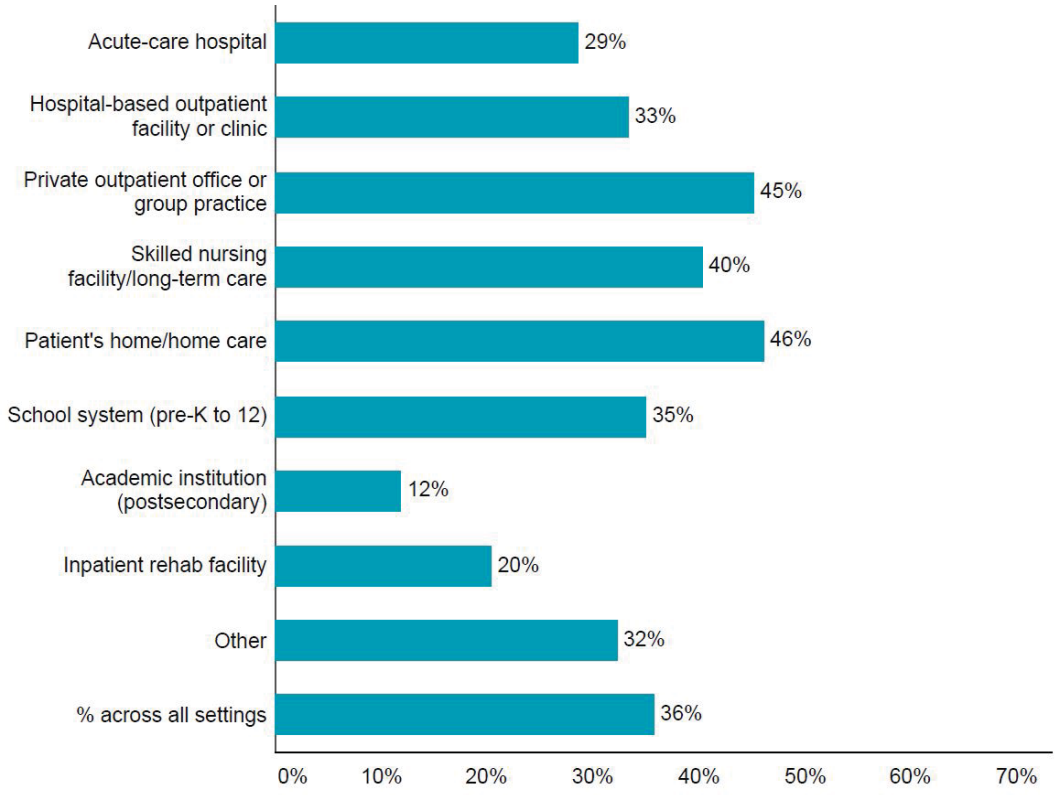


Figure 1.1: Percentage of physiotherapists whose July 2020 work hours were lower than before the pandemic [21]

1.1 Background on Rehabilitation

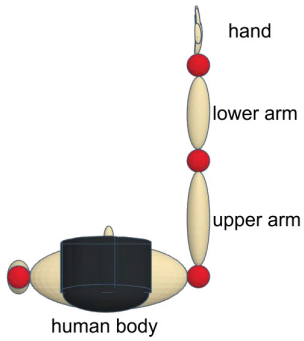
1.1.1 Human Arm Movement and Upper-Limb Muscle Function Impairments

Human Arm Movement

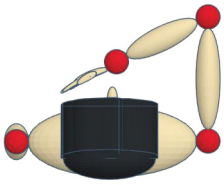
The research presented in this thesis focused on two of the distal osteokinematic movements: the extension and flexion of the elbow joint shown in Figs. 1.2(a) and 1.2 (b), and the extension and flexion of the wrist joint shown in Figs. 1.2(c) and 1.2(d). The general angular range of a non-affected elbow is 150° while that of a normally-functional wrist is 160° [25].

Upper-limb (UL) Muscle Function Impairment

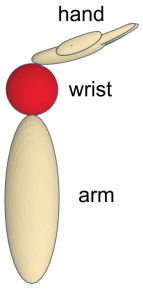
Mobility-affecting damage to the central nervous system can be caused by a variety of neurological diseases such as Multiple Sclerosis, Parkinson's disease, and Cerebral



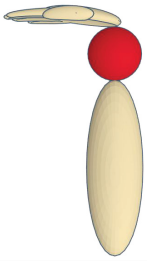
(a) Elbow extension



(b) Elbow flexion



(c) Wrist extension



(d) Wrist flexion

Figure 1.2: Top view of osteokinematic elbow and wrist extension/flexion movements

palsy, and neuromuscular disorders such as Muscular Dystrophy [26], Guillain-Barré, and Amyotrophic Lateral Sclerosis. Additionally, traumatic brain and spinal cord injuries, orthopedic dysfunctions, and accidents can also cause upper-limb muscle dysfunction and their diminished functionality. One of the common physical impairments due to cerebrovascular diseases such as stroke is hemiparesis (weakness on one side of the body). The most prominent and persistent disability in stroke patients, however, is upper-limb impairment, which is reported in 70% of patients upon admission to the hospital [27]. The disabilities can include reduced range of motion, less accurate and reduced smoothness of movements [28], impaired somatosensation (decreased ability to sense the movement of body parts) [29], and excessive muscle contractions (spasticity) [30], which is also called tone and can feel as resistance to passive movement of a limb. Some of the secondary impairments include muscular atrophy, the shortening and stiffening of soft tissue due to disuse, and a decreasing passive range of motion [29]. As a result, sufferers can experience reduced independence, persistent

frustration, anxiety, depression, and withdrawal [31] due to diminished ability to perform activities of daily living (ADL) such as eating, grooming, dressing, housework, exercising, dancing, cooking, driving, interacting with other people, using computers, and operating machinery.

1.1.2 Upper-Limb Conventional Rehabilitation and Rehabilitation with Robotic Devices

A. Conventional Rehabilitation

Physical therapists are crucial medical professionals that specialize in the assessment of functional impairment and the delivery of physical rehabilitation to restore function. On one hand, mobilization, stretching, and actively moving the affected joint as mandated by a physiotherapist or guided by a physiotherapist can greatly improve and fully restore function and range of motion. On the other hand, failed therapy or the lack of physical therapy can leave many sufferers with life-long compromised mobility, reduced quality of life, further complications [32], and can cause those with a nonfunctional arm wishing that it would be amputated [31].

A potential solution to help mitigate the surging and unmet demand for physical rehabilitation therapy is the use of rehabilitation robotic devices. As Buerger *et al.* has stated in "Rehabilitation Robotics: Adapting Robot Behavior to Suit Patient Needs and Abilities" [33], these devices "must closely cooperate with various human subjects; they also must be designed so as to facilitate motor recovery over time. The latter challenge is made particularly difficult by the fact that the mechanisms of motor recovery are not yet fully understood. Therapy robots offer potential help in unraveling this mystery."

Marchal-Crespo *et al.* [34] outlined a number of motivations and rationals for robotic-assisted rehabilitation with active assist exercises. Though not extensively verified in scientific studies at the time their paper was written in 2009, it is interesting to consider the potential benefits and pathways in which rehabilitation robots may aid in mobility recovery. Rehabilitation robots may:

- assist in preventing stiffening of soft tissue and decrease spasticity by stretching the muscles and connective tissue;

- assist to create novel somatosensory stimulation which could help stimulate neuroplasticity by moving the limb in ways not possible voluntarily;
- help the patient learn a movement by demonstration;
- assist the patient's locomotor system in reestablishing the normal pattern of motor output by creating a normative pattern of sensory input;
- make possible higher intensity training by enabling higher number of repetitions;
- make possible higher intensity training by allowing the performance of tasks in a safe manner;
- enable progress in task performance by providing less assistance over time, similar to how a toddler needs progressively less assistance in standing and walking on their own as they learn to balance;
- help improve patients' motivation and their connection of 'intention-to-action' by aiding them in achieving desired movements.

Rehabilitation robots can deliver task-oriented, repetitive, customized, high-intensity, and variable-intensity training enabling improved mobility recovery through functional neuroplasticity [7, 35, 36, 37, 38, 39]. These devices can provide quantified, accurate, and objective measurements related to the patient's range of motion, strength, and functional ability. This rich and customized feedback has been found to enhance therapy by providing extra motivation and encouragement to the patient to continue with the physiotherapy program and empower the therapist with insightful diagnostics allowing for more effective analysis and improved assessment of the patient's progress [40]. As adjuvents, these devices have been proven to result in a 'remarkable difference ... between pre− and post-treatment' [41]. Furthermore, they could be used in telerehabilitation when patients cannot receive the therapy in-person, and can be programmed to adapt to different users and patients' changing abilities. The use of these devices can help in improving a patient's quality of life, performance of ADL, and their contribution to society. Additionally, these rehabilitation robotic devices can help relieve therapists from repetitive and physically laborious work, and can help increase the throughput of service delivery. As a result, the addition of rehabilitation

robotic devices is becoming increasingly accepted by patients and physiotherapists alike, and is becoming progressively popular in clinics and hospitals around the world [42].

Daniel Daarte, a physiotherapist, has said that: "being (a) therapist means knowing how to put the heart in your hands, science in your movements and hope in your eyes". Having been in physical therapy for a multitude of injuries, the present author can personally attest to being the beneficiary of physical therapists' invaluable emotional support effective even through video-consultations during the current pandemic. Without a physiotherapist's physical intervention, however, my injury persists, and further motivates the current research to enable rehabilitation robotic devices to assist physical therapist in the delivery of "movements guided by science" to more people who are in dire need. To do so, this thesis strives to improve the control methods to make upper-limb robotic telerehabilitation a viable coadjutant in reducing the exorbitant unmet demand for physiotherapy.

B. Upper-Limb Rehabilitation Robotic Devices: Mechanical Configuration

There are four main mechanical configurations of upper-limb rehabilitation robotic devices. They are shown in Fig. 1.3 and described in the following sections.

Sling suspension systems employ the use of cables for gravity compensation so that the patient can exercise their upper limb without having to bear its weight or the weight of any equipment. Fig. 1.3(a) depicts the structural concept of a sling suspension system, while Fig. 1.4(a) shows an example of a commercial system called the Armeo Boom from Hocoma. Sling suspension systems are akin to controlling a puppet by strings. A notable advantage of using sling suspension systems is that less fatigue is experienced by both the patient and the physiotherapist since the majority of the weight of the patient's arm is supported by the cable mechanism. Furthermore, since the cable-based actuators can be located away from the joint, the components that are fastened to the patient's arm can be smaller and streamlined, allowing the patient's arm to be closer to the body. One of the drawbacks of sling suspension systems is that the exercises do not necessarily mimic ADL-like movements. Also, the cable systems do not isolate individual joints, so it is possible that incorrect compensatory movement of joints could be learned by the patient while attempting to perform an exercise.

End-Effector type robots are robotic manipulators that typically guide the patient's hand through an end effector such as a joystick or a handle. Fig. 1.3(b) shows the basic concept of an end-effector type rehabilitation robot, while Fig. 1.4(b) shows the Haptic Master from Motek which is a commercial end-effector type rehabilitation robot. In these scenarios, typically the end effector is the sole point of contact where assistive forces are applied to the patient [43] since it is not attached to other sections of the patient's arm to control specific joints [40]. This setup allows for only multi-joint gross movements of the afflicted arm [44].

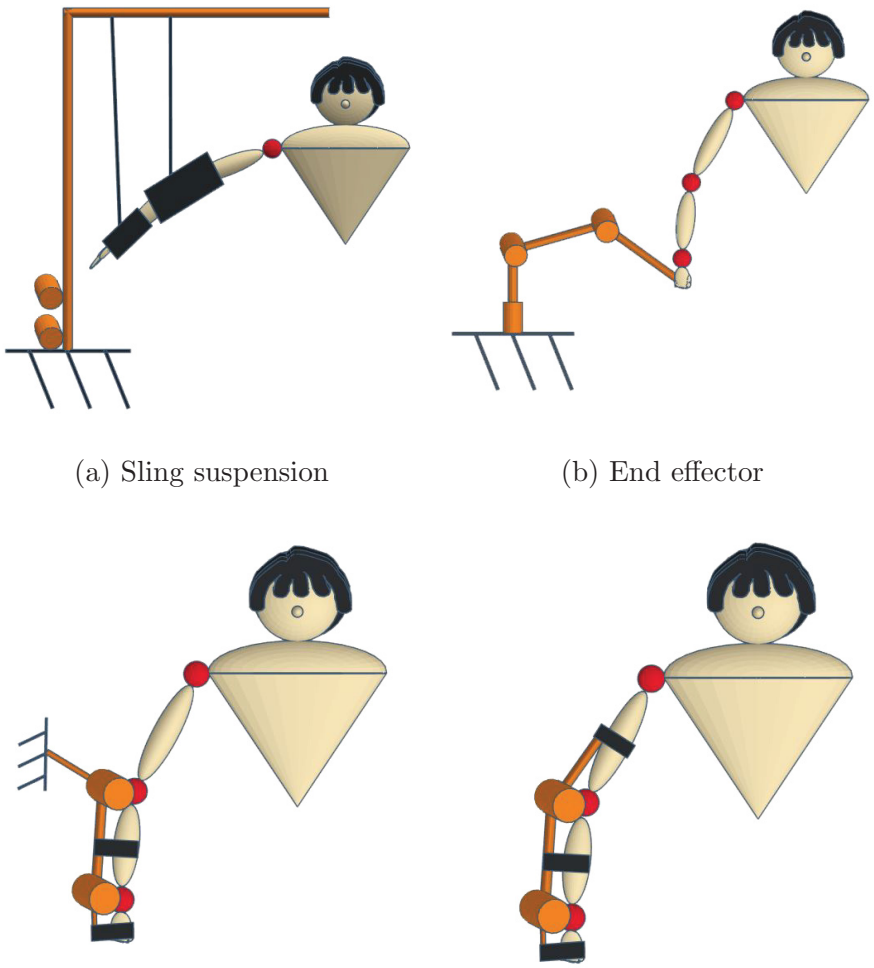
One of the main advantages of using end-effector type devices is that typically they are simpler to design and build than exoskeletons. Additionally, end-effector type devices are easier to interface with the patient considering the single-point of contact. Therefore, they are the more popular type of rehabilitation robotic devices [45]. A disadvantage of end-effector type devices is that it only allows for limited movements due to the 'uncontrolled load transfer between the limb's joints' [46]. Another disadvantage of end-effector type devices is that it is not possible to assess the function of the patient's individual joints since it is only the motion and forces at the contact point that can be measured. Furthermore, using an end-effector type robotic device allows for a combination of joint movements to take place at the same time, making it hard to distinguish which section of the arm is being rehabilitated [46]. The biggest disadvantage of end-effector type rehabilitation robots is that the movement of the patient's joints cannot be independently controlled nor assisted, and thus the patient can actually recruit the use of other unaffected and stronger muscles in the body or move their arm in unnatural ways to complete a task, deeming the exercise counterproductive or even resulting in injury [47]. For example, it is not uncommon for patients to use their torso to perform a manipulation task when using end-effector type rehabilitation robotic devices to compensate for the dysfunction in the affected arm muscles [48, 49]. Some devices incorporate specialized seats with seat belts and straps to constrain different body parts to help isolate the training to specific segments − increasing the complexity and the cost of the system substantially. These devices have an increasing presence in rehabilitation centers and have had good results in improving muscle function in affected patients [50].

Exoskeleton type devices are anthropomorphic robotic manipulators fastened in parallel to the human arm, with the capability of providing targeted assistive motion to each joint of the human's arm. Fig. 1.3(c) shows the concept of a grounded upper-limb (elbow-wrist) robotic exoskeleton, while Fig. 1.4(c) shows a commercially-available grounded wrist exoskeleton called the Motus Hand from Motus Nova which is 'clinically proven to provide equivalent outcomes to traditional modes of rehabilitation' [51], is FDA approved in the USA, and in 2019 was already in use in twenty-five hospitals [52]. Fig. 1.3(d) depicts the concept of a portable elbow-wrist robotic exoskeleton, while Fig. 1.4(d) shows MyoPro from Myomo, an elbow portable exoskeleton which was developed at MIT with Harvard Medical School, and is commercially-available.

The main advantage of exoskeleton devices is that they can provide isolated therapy at the joint level, which is highly beneficial, and they can prevent undesirable, unnatural, and potentially unsafe compensatory maneuvers that can occur when rehabilitating with end-effector rehabilitation robotic devices [41, 53, 47, 54]. Additionally, valuable and insightful position and force data can be provided independently for each joint of the human's arm. Furthermore, exoskeletons can provide a larger range of motion for each joint compared to end-effector type devices. The main disadvantages of exoskeleton systems are that they are generally more complex in architecture, can be more expensive, and require substantial additional safety considerations since they are attached to the human limb. As a result, the clinical trials are arduous obstacles in bringing these types of devices to market [55].

Currently there are numerous robotic rehabilitative devices on the market and many more at the research stage. The main advantage of rehabilitation with robotic devices is that they help narrow the growing gap between rehabilitation service demand and supply. These devices can also be used to augment the therapy regiment by providing precise and objective measurements of the patient's performance and changes in range of motion, strength, and the quality of motion. Ideally the controllers would adapt to different patients and also to patients' changing mobility abilities.

End-effector type devices are the most popular commercial rehabilitation robotic devices, with suspension-type devices being next in popularity; however, exoskeleton-type devices are gaining growing interest due to their distinct therapeutic advantages



(a) Sling suspension (b) End effector

(c) Grounded exoskeleton (d) Portable exoskeleton

Figure 1.3: Upper-limb rehabilitation robotic devices

due to their kinematic compatibility to the human limb.

(a) Hocoma Armeo Boom [56]

(b) Motek Haptic Master [57]

(c) Motus Nova Motus Hand [52]

(d) Myomo Elbow MyoPro [58]

Figure 1.4: Commercial upper-limb rehabilitation robotic devices

1.2 Research Motivation and Objective

Given the escalating unmet demand for physical rehabilitation due to the growing global aging population and the added strain on the medical system due to the COVID-19 pandemic, robotic telerehabilitation is an emerging, timely, and crucial technology which could aid in improving access to care. The majority of the research in robotic telerehabilitation for upper limbs has been performed with end-effector type assistive robots (for example [59, 60, 61, 62, 63]), and very little research has been found in the literature that studies bilateral telerehabilitation control with upper-limb robotic exoskeletons [64, 65]. Considering that the use of robotic exoskeletons has significant and distinct benefits [53, 47, 54], telerehabilitation with upper-limb robotic exoskeletons should be further investigated.

One important aspect of bilateral telerehabilitation training with upper-limb robotic exoskeletons which should be given significantly more consideration is telepresence – which refers to how well the therapist senses the patient's level of effort while trying to assist. Telepresence is influenced by the synthesis of the patient's and therapist's joint torques. For example, Sharifi *et al.* [60] and Lanini *et al.* [65] added therapist's assistive torques which were scaled to reduce therapist fatigue, but produced diminished telepresence. As a result the motivation of this research is to devise a method that can provide improved telepresence in upper-limb robotic exoskeleton telerehabilitation.

Since upper-limb robotic exoskeletons are subject to sensor noise, human torque input, communication time delays and inherently contain nonlinear dynamics due to their complex architecture, further research is required that explores both model-based and model-free control methodologies which could address all these control challenges while providing stability and good tracking. Furthermore, modalities should be investigated which allow for adjustable compliance which provide safe motion and enable customization for different patients, and patient's changing mobility abilities.

Objective: The objective of this research is to build on the state-of-the-art control methodologies for bilateral telerehabilitation with upper-limb robotic exoskeletons to aid in the delivery of physiotherapy to those who direly need it.

1.3 Contributions

The main contributions of this research are as follows:

1. **A novel Human Torque Regulator (HTR) with high-fidelity telepresence for bilateral telerehabilitation:** In bilateral telerehabilitation, the therapist applies assistive joint torques on the slave robot to assist the patient in following a trajectory with the robot on the master side. The existing methods found in the literature of combining the torques from the patient and the therapist to be applied to the master robot are 1) using a direct-addition strategy [60] and 2) using a torque-scaling strategy [60, 65]. Both of these strategies require the therapist to only exert additives torque. Although these methods have the advantage of potentially reducing fatigue for the therapist, they provide diminished therapist's haptic awareness of the torques exerted by the patient. Both of these strategies were implemented and studied in this research in Chapters 5 and 6 and, as a result, a novel Human Torque Regulator (HTR) was created. The proposed HTR produces the desired high-fidelity telepresence for the therapist as shown in Chapters 5 through 7, and written in G. Bauer and Y.J. Pan, "Telerehabilitation with Exoskeletons using Adaptive Robust Integral RBF-Neural-Network Impedance Control under Variable Time Delays," Accepted to *2021 IEEE International Symposium on Industrial Electronics (ISIE)*, Japan, Kyoto, Conference Date: June 2021 [66]. Therein, the therapist has a clearer perception of the amount of effort that the patient is applying through the training resulting in better evaluation and therefore greater efficacy of the therapy.

2. **Integration of impedance model control:** In applications which require humans to be in direct physical contact with moving robots − as is the case in rehabilitation with upper-limb robotic exoskeletons − it is adamant to accommodate for the variability in the patient's mobility abilities, generate compliant motion, enhance safety, and provide system stability. One method of creating compliance found in the literature is to implemented a series of elastic elements [67, 68], while another method is to implementing an impedance model controller. The latter method regulates the patient-exoskeleton interaction torques

based on its inertial, stiffness, and damping parameters, and thereby adjusts the reference trajectory. This strategy was implemented in [60] with end-effector type robots and in [69] with a single-system upper-limb robotic exoskeleton with promising results; however, its use has not been documented in the literature for telerehabilitation with upper-limb robotic exoskeletons. Given its merits, the implementation of impedance model control was integrated in this research and is described in detail in Chapter 5, while its effects are shown in simulation-based and experimental tests in Chapters 5, 6, and 7. By incorporating a designed impedance model controller into the upper-limb robotic exoskeleton control methods proposed in this thesis, the allowance is made for customization based on the patient's changing mobility abilities, adjustable compliance, safe motion, and enhanced stability. The implementation of impedance control was documented by the author in [66, 70].

3. **Adaptive Robust Integration Impedance Control (ARII):** To address the unknown dynamic modeling parameters, nonlinearities, sensor noise, and human torque input while providing good tracking, safe, stable, and adjustable compliant motion which builds on the state-of-the-art, adaptive control, an Adaptive Robust Integration Impedance Controller (ARII) was developed, implemented, and validated. Using adaptive impedance control is similar to [62] which was implemented for end-effector type robots, and [71] where it was used for a finger exoskeleton, but in the research presented in this thesis robust and integral control elements were also implemented. Unlike [72] which uses adaptive impedance control for a simulated single-joint upper-limb exoskeleton and regards the robot and the human arm as one system, this research addresses the coupling dynamics between the robotic exoskeleton and the human arm − as described in Chapter 4. Chapter 5 reveals the design of the proposed ARII method and shows detailed Lyapunov-based stability and convergence analysis. Simulated and experimental results for a single-system Quanser QUBE testbed are provided in Chapter 5. Experimental telerehabilitation results using a Quanser QUBE testbed and simulated 2-degrees-of-freedom (2-DOF) exoskeleton testbed telerehabilitation results are shown in Chapters 5 and 7, which

reveal the desired tracking performance and stability. The results of an earlier version of this method have been published in: G. Bauer, Y.J. Pan, and H.H. Shen, "Adaptive Impedance Control in Bilateral Telerehabilitation with Robotic Exoskeletons," 2020 IEEE International Conference on Systems, Man, and Cybernetics, Oct 2020 (pp. 719-725) [70].

4. **Adaptive Robust Integral Radial Basis Function Neural Network Impedance Control (RBFNN-I):** To address the unknown dynamic modeling parameters, nonlinearities, sensor noise, and human torque input while providing good tracking, safe, stable, and adjustable compliant motion using a model-free strategy, a second control method for upper-limb robotic exoskeletons, RBFNN-I, was developed, implemented, and validated. Like [73], this controller incorporates robust, integral, and adaptive control to adjust the weights of an RBF neural networks. Unlike [73] where an adaptive robust RBF neural network controller was used to control a simulated 2-DOF robotic manipulator, RBFNN-I is applied in simulation and experimental tests to single-joint robots, and in simulation to 2-DOF upper-limb robotic exoskeletons in bilateral telerehabilitation. Additionally, the proposed RBFNN-I controller incorporates an impedance model controller to provide adjustable compliance, stability, and safety to the human in the loop. Chapter 6 presents the design of the method and detailed Lyapunov-based stability and convergence analysis. Simulation and experimental results for a single-system Quanser QUBE testbed, and simulation results for a 2-DOF exoskeleton testbed are provided in Chapter 6. The results of an earlier version of the proposed RBFNN-I control method have been summarized in [66].

5. **Extensive Validation and Performance Comparison of ARII and Adaptive Robust Integral RBFNN-I:** A comparison was performed between the two proposed control methods, ARII and RBFNN-I, for a single-system, and unilateral and bilateral telerehabilitation tests, as shown in Chapter 6 for both simulations and experiments. Furthermore, Chapter 7 presents a comparison between the two methods through experimental case studies using single-joint Quanser QUBE robots with human intervention and simulation case studies

using a 2-DOF exoskeleton testbed. Additionally experimental tests where performed with the RBFNN-I method to explore the range of operation for key parameters.

Additionally, a conceptual design of a novel grounded horizontally-planar 2-DOF upper-limb robotic exoskeleton was developed as shown in Chapter 4 and utilized for simulation-based validation of the proposed control methods in Chapters 5 through 7. Moreover, a first-in-its-field extensive literature review of control methods for upper-limb telerehabilitation with robotic exoskeletons was performed. The results were published in: G. Bauer and Y.-J. Pan, "Review of Control Methods for upper-limb Telerehabilitation with Robotic Exoskeletons," IEEE Access, Nov 2020 [74].

1.4 Thesis Organization

This thesis is focused on the development and implementation of control methods for telerehabilitation with upper-limb robotic exoskeletons. The organization of this thesis is constructed as follows: Chapter 1 presented the validity of robotic-assisted physical rehabilitation, research motivation, thesis objectives, contributions, and novelties. Chapter 2 presents the literature review of the state-of-the-art control methods and key aspects of telerehabilitation for upper-limb robotic exoskeletons. Chapter 3 provides fundamental robotics and system stability concepts. The problem formulation and the system models are presented in Chapter 4. Chapter 5 reveals the first proposed control method, ARII, including its further development, implementation, and validation through simulation and experimental results. Chapter 5 also explains and implements the current human torque syntheses methods present in the literature and introduces the proposed HTR. Chapter 6 reveals the second proposed control method, RBFNN-I, including its development, implementation, and validation through simulation and experimental results. Chapter 6 also provides a comparison between the two methods for single-system and telerehabilitation simulation-based and experimental tests. Chapter 7 describes telerehabilitation simulation and experimental case studies. Chapter 7 also includes experimental tests performed with the RBFNN-I control method to explore the range of operation of key parameters. Chapter 8 draws conclusions and makes recommendations.

Chapter 2

Literature Review

Over the years, there has been extensive research carried out in the field of robotic-assisted rehabilitation [5, 7, 26, 34, 39, 45, 75, 76, 77, 78, 79] as they have been "proven successful in speeding recovery for recent stroke victims" [33], produce "remarkable" improvement [41], and reduce impairment and pain in chronic sufferers [33]. Given the current, urgent interest in exploring alternatives for physiotherapy delivery [4, 23, 74] and especially remotely-administered physiotherapy [29, 45, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89], robot-assisted telerehabilitation is increasingly being investigated [70, 87, 90, 91]. The majority of the research in robotic telerehabilitation for upper limbs has been performed with end-effector type assistive robots [38, 59, 60, 61, 62, 63, 92, 93], but the use of robotic exoskeletons is gaining interest due to significant and distinct benefits [41, 47, 53, 54, 94]. The robotic exoskeleton market across all fields (*e.g.*: rehabilitation, ADL assistance, transportation, healthcare, augmentation, space, defence) is predicted to exponentially grow (Fig. 2.1) from $\$68M$ in 2015 to $\$1.9B$ by 2025, which attests to the validity of robotic exoskeleton technology. Furthermore, the forecast compound annual growth is between 40% [42] and 50% [95] – with lower-limb rehabilitation robotic exoskeletons currently leading the way. As an emerging technology, there are not many studies in the literature on bilateral telerehabilitation control with upper-limb robotic exoskeletons [64, 65]; therefore, this chapter provides an overview of the state-of-the-art control methods pertaining to telerehabilitation with upper-limb robotic exoskeletons. The sections of this chapter present model-based and model-free control methods, modality for safe motion and adjustable compliance, and telerehabilitation-specific topics including: telepresence and communication time delays. Additionally, each section concludes with remarks on the relevance of the presented work to this thesis.

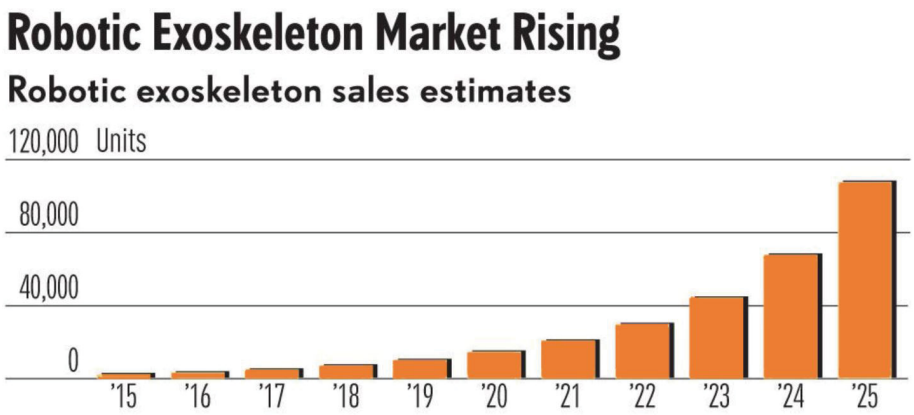


Figure 2.1: Market forecast for robotic exoskeletons in all fields [96]

2.1 Upper-limb Robotic Exoskeleton Control Methods

One of the main reasons for physical rehabilitation is to provoke functional neuroplasticity [35, 7, 36, 37, 38, 39] for mobility recovery. During the acute stage (the first few weeks right after a stroke has taken place), the patient's affected arm could be unable to move voluntarily on its own and typically there is tightness in the arm causing the hand to rest near the chest wall [97]. The objective of exoskeleton robot-assisted physiotherapy is to assist the patient in reducing impairment by delivering, for example, task-oriented, repetitive, customized, high-intensity, and variable-intensity training. Fig. 2.2 shows a block diagram of an upper-limb rehabilitation robotic exoskeleton control system. The desired reference motion on the left side of the diagram is specified by the therapist through routines stored in the computer. This reference motion information is compared to the actual motion measured by sensors on the robotic exoskeleton attached to the patient. The error between the desired and the actual variables is used by the controller to calculate the robotic exoskeleton's motors output. The objective of upper-limb robotic exoskeleton control methods is to enable safe, stable, and compliant motion while attaining converging tracking error in the presence of external disturbances, such as sensor noise, human torque input, and nonlinearities such as friction. Furthermore, one of the biggest control challenges for these systems is handling nonlinear dynamics [69]. The following sections present control methods which attempt to address these challenges.

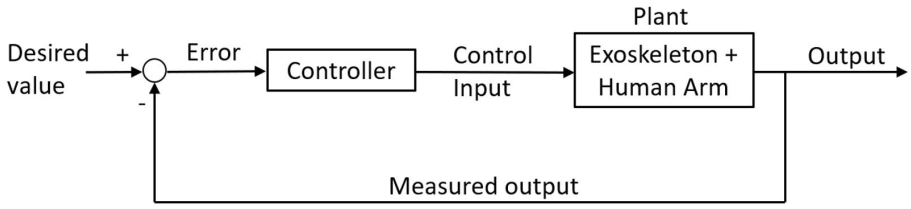


Figure 2.2: Block diagram of a closed-loop control system

2.1.1 Model-based Control Methods

Since robotic exoskeletons are inherently complex nonlinear systems, it is virtually impossible to create perfect mathematical models of them. As a result, any model-based control method implemented has to compensate for the unavoidable discrepancies between the model and the physical plant. Although there are other advanced nonlinear methodologies present in the literature, adaptive control and sliding mode control approaches are introduced in the following sections.

Adaptive Control

Adaptive controllers can perform well with systems that have a fully-known structure, but uncertain dynamic model parameters [98]. Implementing adaptive control requires the rearrangement of the non-linear dynamic equations so that they become linear in the unknown dynamic parameters via a regressor matrix which is multiplied by the vector of unknown parameters. Since this vector contains all the uncertain parameters, the initial values are estimated and converge with time [99]. Fig. 2.3 shows how the unknown system parameters are adjusted in real time based on the difference between the measured sensor values y from the system and those from a reference model y_{ref}. The linear parameterization process can be time-intensive and grows in complexity and time-demand as the number of degrees of freedom increase [100]. An example of an upper-limb rehabilitation robotic exoskeleton system that uses adaptive control is ARMin V [101]. Adaptive controllers do not require *a priori* information about the limits of the uncertain or changing parameters as robust methods do. Adaptive methods, however, are not well-suited for systems that exhibit fast-changing parameters or systems that are exposed to external unmeasured disturbances [98]. Researchers have proposed to addressed the latter problem by combining

adaptive controllers with other types of controllers including robust control methods to handle bounded external disturbances. Adaptive robust control was used in [69] and [102].

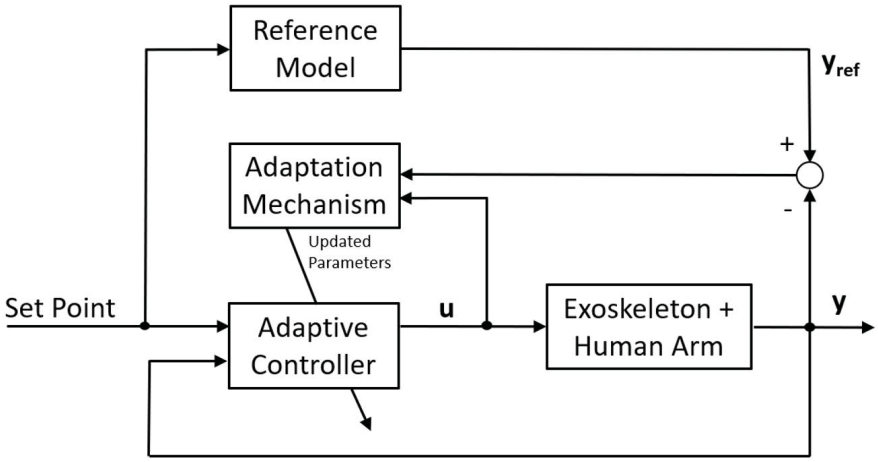


Figure 2.3: Schematic diagram of an adaptive controller for exoskeleton devices

Sliding Mode Control (SMC)

Sliding Mode Control is a nonlinear robust method used to address bounded external disturbances [103] and parametric uncertainties. A switching controller is designed which forces the system state trajectories to converge onto a sliding surface in the state space in a finite amount of time. By limiting the system to be in the neighborhood of the switching function, the system becomes stable, linear, time-invariant, and insensitive to parametric uncertainties and external disturbances [104] under a nominal controller. These nominal and switching controllers are shown in Fig. 2.4, with their computed torques $\boldsymbol{u_{sw}}$ and $\boldsymbol{u_{nom}}$, respectively. The two torques are then added to produce the commanded torque sent to the robot [105]. One of the disadvantages of SMC is that the high-frequency switching action could cause chattering in the commanded output resulting in wear or damage of the mechanical system, energy loss in the electrical system, heating, and vibrations [106]. To help mitigate this problem, mathematical smoothing methods have been implemented [107]. SMC controllers are robust in the presence of disturbances and parameter variations, and can converge faster than adaptive controllers. Additionally, they work well on nonlinear systems that are hard to model since they do not need an exact model [98].

Some upper-limb robotic exoskeleton systems that use SMC are described in Yun *et al.* [108], Brahmi *et al.* [109], and Lo *et al.* [105].

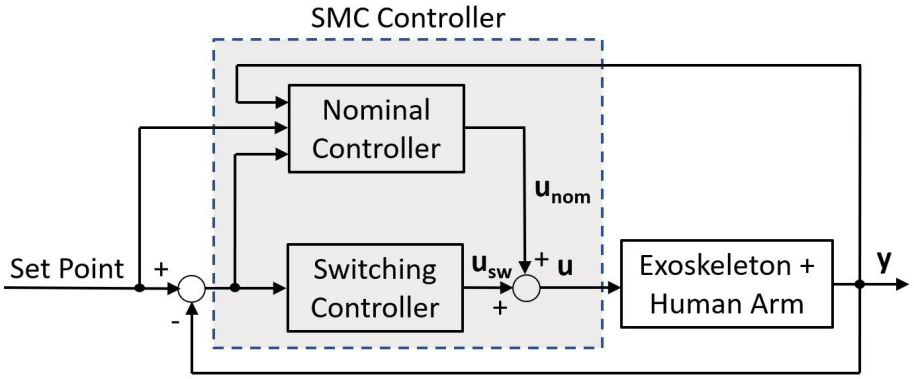


Figure 2.4: Schematic diagram of an SMC controller for exoskeletons

2.1.2 Model-free Control Methods

Proportional Integral Derivative (PID) Control

Some upper-limb robotic exoskeletons are controlled using PID control. This control approach does not require system modeling [54] and is relatively straightforward to implement. The PID control law is shown in Eq.(2.1) where $u(t)$ denotes the control input (motor torque in this case), error $e(t)$ represents the difference between the actual and desired position of joints, and $\dot{e}(t)$ is its derivative with respect to time. The proportional gain K_P operates on the position error. The integral gain K_I operates on the accumulated position error and the derivative gain K_D operates on the velocity error. Fig. 2.5 shows each component of the PID control law in a control block diagram.

$$u(t) = \underbrace{K_P e(t)}_{\text{Proportional}} + \underbrace{K_I \int_0^t e(\tau)d\tau}_{\text{Integral}} + \underbrace{K_D \dot{e}(t)}_{\text{Derivative}}. \tag{2.1}$$

While model-free control leads to simplicity in its implementation, a disadvantage of using this method is that repetitive experimentation is typically required for gain-tuning [110] − a process regarded by some as an art form. Additionally, tuning the gains can be difficult in the presence of time-varying parameters, nonlinear artifacts, and human-machine interactions [110]. Controllers can be created which use

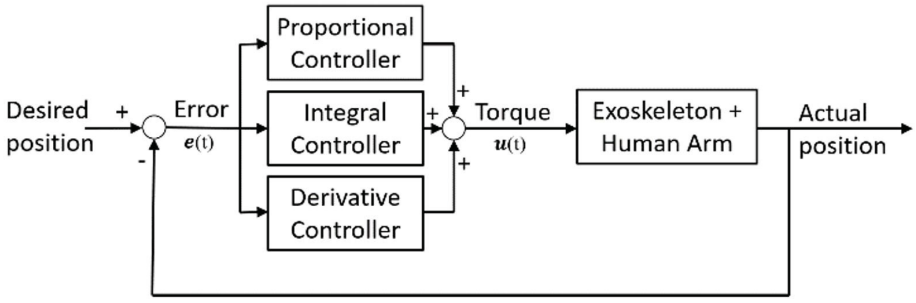


Figure 2.5: PID controller

a subset of the PID controller, such as P, PD, or PI. For example, the upper-limb robotic exoskeleton ARMin III [47] uses a PD controller while EXO-UL7 [111] uses a PID controller. Furthermore, many single-system upper-limb rehabilitation robotic exoskeleton researchers combine elements of the PID controller with advanced control methods such as Robust Control and Adaptive Control, and intelligent control methods such as Neural Networks and Fuzzy Logic. For example, [112] combines PID Control with Robust Control and Fuzzy Logic based Control. In the telerehabilitation studies presented in the literature, Lanini *et al.* [65] and Buongiorno *et al.* [64] used PD controllers on the slave robots.

Radial Basis Function Neural Networks (RBFNN) Control

Roboticists have also implemented Artificial Intelligence (AI) control methods, such as radial basis function neural networks (RBFNN), to approximate the complex nonlinearities and uncertainties in the system [113, 73, 114, 115]. In contrast to adaptive and robust control methods, RBFNN-based controllers have the advantage of not requiring a lot of information about the robot dynamic model *a priori* [116]. Furthermore, RBFNN-based controllers have been shown to successfully learn the neural network values even with large uncertainty of the system [93], which other model-free control methods like PD cannot do. Since upper-limb robotic exoskeleton telerehabilitation applications have inherent dynamic model parameter uncertainties as well as nonlinearities, control methods which can provide options for addressing these issues such as RBFNN should be explored. Other model-free control methods are presented in the literature, such as Reinforcement Learning [117] and Fuzzy Logic [118].

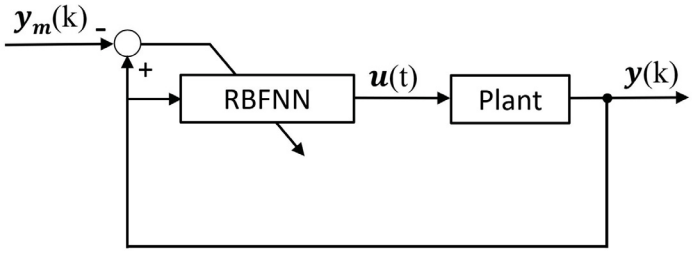


Figure 2.6: RBFNN controller, based on [73]

2.2 Modality for Safe Motion and Adjustable Compliance

In applications where human operators are in direct contact with robotic devices, impedance control is often implemented to enable compliance and safety [62] [119]. Impedance control enables adjustable compliance, meaning that it makes adjustments to the robot's commanded desired trajectory based on the human interaction forces instead of rigidly following a given trajectory [120]. Thus, it is a desirable control modality for rehabilitation where the patient's varying abilities need to be considered in real time. Impedance controllers have been implemented in research with upper-limb rehabilitation robotic exoskeletons for the L-Exos [48], SUEFUL-7 [121], and the ETS-MARSE [109]. In telerehabilitation applications, it has been used by [59, 60] with end-effector type robots. A series of elastic elements was used instead in [67, 68].

The research presented in this thesis implements elements from all of the control methods described in this chapter as follows:

1) a model-based adaptive controller was implemented to overcome the unknown dynamic modeling parameters, a robust control term with a smoothing function was incorporated to help deal with external disturbances, and an integral control term was added to handle any sustained offsets [70]. Moreover, the adaptive controller was extended to include an impedance model to enable adjustable compliance and safe motion, similar to [61, 62, 60] where it was utilized with end-effector type robots and [69] where it was utilized with an upper-limb robotic exoskeleton.

2) a model-free RBFNN-based control method was implement which learned the weights of the neural networks with the addition of an adaptive law. A robust control term was added to handle external disturbances, along with an integral term

to address any accumulated error offsets. Building on the strength of RBF neural networks methods presented by Liu [73], where the plant was a theoretical 2-DOF robotic manipulator, the RBFNN-I control method proposed in this thesis was applied to upper-limb robotic exoskeletons applications for telerehabilitation. Furthermore, the RBFNN was extended to also incorporate an impedance model to allow for cutomizable compliance and safe motion to be applicable for human-machine interface applications [66].

Although the majority of the telerehabilitation with upper-limb robotic exoskeletons control studies in the literature utilized multi-degree of freedom robotic exoskeletons which are equipped with force sensors along the robotic limb, they often did not control the robots in joint space mode, and instead used the exoskeletons in task-space − effectively reducing them to end-effector type robots. Further research is needed to apply exoskeletons in joint space and take advantage of the force sensors along the robotic limb, and explore joint-specific therapies. In this thesis, experimental tests were performed with single-joint robots, and simulations with 2-DOF robotic exoskeletons controlled in joint space.

2.3 Telerehabilitation

When access to physiotherapy services is not possible, telerehabilitation could be the answer. There is an increased interest in telerehabilitation as a promising modality which could help administer therapy to patients who are living in remote areas, are housebound due to their disabilities [80, 81, 90], or are house bound due to the present pandemic-related additional challenges [89, 122, 123] − present author included. Although telerehabilitation can mean phone-consultations, video-consultations, verbally-guided physical therapy, speech therapy, virtual reality-assisted, and robotic-assisted training [123], in this thesis telerehabilitation is strictly referring to robot-assisted telerehabilitation training, as depicted in Fig. 2.7. In this scenario, the patient is connected to a robot at a local site, and the therapist is connected to another robot at the remote side. The patient's arm that requires therapy is moved by the master robot according to a desired trajectory. The measured motion information is sent across the communication channel to the slave side, where the slave robot will move the therapist's arm through matching motion. This strategy allows

the therapist to observe and gauge the patient's mobility capabilities. When there are no therapist torques sent back to the master side, the telerehabilitation is called unilateral. If the therapist desires to intervene − and the system has the capability − the assistive torques applied by the therapist to the slave robot are transmitted and combined with those of the patient on the master side. This type of telerehabilitation is called bilateral.

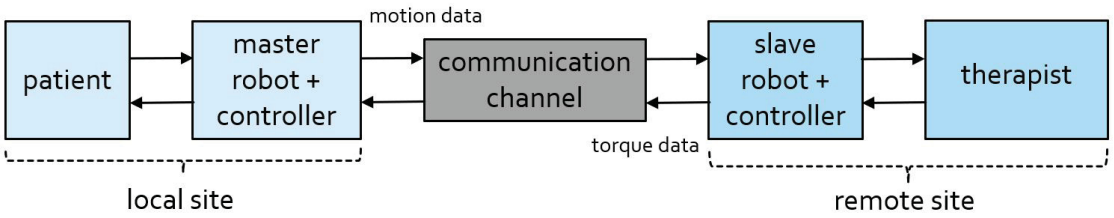


Figure 2.7: Telerehabilitation system

2.3.1 Telepresence

In bilateral telerobotics, "the master robot not only measures motions but also displays forces to the user" [124] − simultaneously affecting the environment (or human) at the remote site while also perceiving the reflected force from that interaction [125]. Sheridan [126] describes the phenomenon of telepresence as what 'enables the human to feel present at the remote location even though not really there'. The goal is for the interaction to be so immersive and natural that the 'human operator is fooled into forgetting about the medium itself' [124]. As Buerger *et al.* [33] explained in 'Rehabilitation robotics: adapting robot behavior to suit patient needs and abilities' and Carignan and Krebs [90] restated in 'Telerehabilitation robotics: Bright lights, big future?', 'the performance of a rehabilitation robot is defined not in terms of its capability to follow a trajectory, but instead by its capability to provide a desired 'feel'...'.

Providing higher telepresence fidelity means increased perceived quality of kinesthetic forces and results in an increased sense of immersion and engagement [127]. In telerehabilitation, high-fidelity telepresence allows the therapist to perceive the torques that the patient is experiencing with maximum clarity which helps the therapist in gauging the patient's current mobility abilities, enhances safety, and improves the efficacy of the therapy. Although studies have been found in the literature

[64, 65] which presented control methods for telerehabilitation training with upper-limb robotic exoskeletons, it is evident that the field is only emerging as none of the cases addressed the topic of telepresence. A tele-assessment clinical trials study [128] which utilized an elbow exoskeleton and an actuated haptic device with a manikin arm as a therapist interface, demonstrated the benefits of telepresence while assessing the condition of four patients. Their results proved very similar to in-person assessment. Another tele-assessment clinical trials study which evaluated the telepresence aspect of a telerehabilitation setup [28] using two exoskeletons reported very positive outcomes as well. This study involved fifteen professional physical therapists who were asked to assess the condition of four subjects. The therapists rated the strategy as a very useful medium (mode: 4 points out of 5 on the Likert Scale) for evaluating a patient's progress over time. The paper concluded that it was a promising approach to complement robot-assisted movement training.

In telerehabilitation with upper-limb end-effector type robots, there are examples found in the literature where the combination of the human torques from the patient and the therapist were added directly together [60], or amplified versions of the therapist's torques were added to the patient's to minimize the therapist's effort [60]. Although useful in reducing therapist fatigue, these human torque syntheses strategies do not give the therapist a very good sense for the amount of effort that the patient is experiencing. These two strategies were implemented in this research, along with the creation and implementation of a new method which provides the therapist with high-fidelity telepresence while assisting the patience was created [66].

2.3.2 Communication Time Delays

Orchestrating the safe and effective reflection of the motion and torques between the two sites is a major undertaking which requires an overall control system that can address all the previously-mentioned issues, and do so in the presence of communication time delays as well. Time delays can cause the stability of the system to be compromised when forces are fed back from slave side to the master side resulting in undesirable and, even more critically, unsafe consequences for the humans in the loop − if not properly addressed [114]. Over the last sixty years, there have been

increasingly more telerobotics applications in various fields such as space [129], underwater control [130], hazardous environments [131], forest fire detection [132], military [133], mobile robots [134], tele-driving [135], telemedicine such as telesurgery [136] and telerehabilitation [64]. As a result, researchers have created and implemented many methods to address the effects of delays across communication channel when operating robots remotely [136, 137, 138, 139, 140, 141, 142, 143].

Some of the most common and successful methods are passivity-based, where the energy entering the system is greater than the energy leaving the system [136, 137]. Stability is ensured by limiting the system energy, introducing boundedness of system variables [125], or injecting damping agents [144] to remove the excess energy. A major advantage of using passivity-based methods is that the dynamic models of both the master and the slave systems are not required to be known [145]. As a result, passivity-based methods are ideal for systems with large uncertainties or multi-degrees of freedom systems which contain nonlinearities and complexities that are hard to model − like upper-limb rehabilitation robotic exoskeletons. However, implementing certain passivity strategies such as boundedness and overdamping are not always desirable since, although they can deal with delays which may cause instability, they cannot ensure a desired synchronization or telepresence performance. In such cases, non-passivity based control methods could be the answer. Non-passivity based controllers are model-based though [137] and, as such, have the added challenge of requiring perfect knowledge of the plant model. Since it is impossible to attain perfect physical system plant models, many researchers have implemented adaptive [146] and robust [147] control elements to enhance their performance.

Of all the telerehabilitation with upper-limb robotic exoskeleton control studies surveyed in the literature, only one incorporated measures to overcome communication time delays by developing and implementing a time-domain passivity-based controller [64], which has been applied in general bilateral teleoperation systems and haptic interfaces [145, 148, 149]. Another study [65] did not incorporate time delays in their experiments, but discussed how they might be addressed theoretically. Zhang *et al.* [67] measured a time lag having a maximum of 15 ms in the communication channel of their experiments but did not deliberately compensated for them. The patient did not feel any obvious discomforts, and the motion of the exoskeleton was

observed to be smooth.

As time delays and other communication channel artifacts are inevitable in telerehabilitation applications, more research should be carried out which explores counteractive modalities. The research presented in this thesis enhanced stability across the communication channels by implementing the passivity-based strategies of boundedness of the relayed signals and implementing an overdamped impedance model system.

Chapter 3

Background Theories

This chapter introduces fundamental robotics concepts and terminologies used throughout the thesis, as well as system stability.

3.1 Fundamental Robotics Concepts

Robots are electro-mechanical devices designed to perform specific tasks that mimic or surpass human capabilities related to speed, force, torque, endurance, and accuracy, and they can be designed to operate in hostile or remote sites. Robots consist of actuators, sensors, and a physical structure. One of the most common configurations is a serial robot. A diagram of a 2-DOF serial planar robot is shown in Fig. 3.1. The robot typically extends from a grounded base and consists of a series of structural links that are chained together by motors (similar to a human's arm) with the purpose of performing a task at its end effector (analogous to a human's hand). A control system uses feedback information collected by the sensors to appropriately actuate the motors to move the links and the end effector to perform a specific task. To control the robot accurately, sufficient information must be known about the kinematic and dynamic properties of the system, and an appropriate control methodology must be utilized.

3.1.1 Kinematics

Kinematic parameters refer to a robot's most fundamental characteristics: its geometrical parameters which are used in the design, analysis, simulation and control of an application. The kinematic parameters of a system can be used to calculate the robot's motion (displacement, velocity, and acceleration) including the position and the orientation of each joint and the end effector (pose). Kinematics do not include information about the causes of the motion, such as forces, torques, weights,

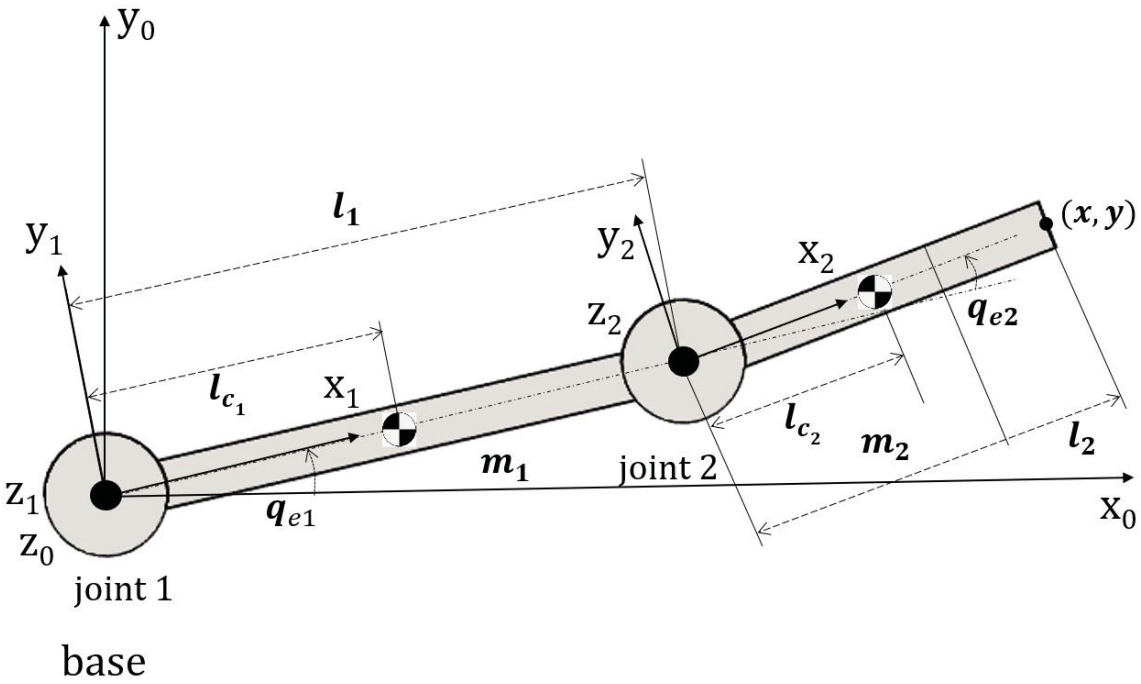


Figure 3.1: Top view diagram of a 2-DOF horizontally-planar robot

or weight–distributions of the components [150]. The dimensions of the robot's components and the range of motion of its joints define the reachable points in space by the end effector which is known as the workspace.

Forward Kinematics

The forward kinematics equations use the joint and link parameters of a robot to determine the end effector's position or pose (position and orientation) relative to the base of the robot [99]. The forward kinematic equations are formulated by properly selecting coordinate frames and fixing them to each robot joint and then constructing the relationship between these frames. Homogeneous transformations that incorporate the translation and rotation between each two successive frames can be used to build up the complete relation of the kinematic chain from the base of the robot to the end effector. The Denavit-Hartenberg notation is a popular method used in robotics to compose the kinematic model of a robotic manipulator which describes the kinematic relationship between the frames. By using a common normal between two consecutive joint axes, the Denavit-Hartenberg convention simplifies the calculations to a minimal representation [99].

Referring to Fig. 3.1, coordinate frames 0 and 1 are co-located at joint 1 where frame 0 is the base inertial reference frame and frame 1 is attached to link 1 of length l_1. Similarly, frame 2 is attached to link 2 having length l_2. Link 1 has a mass m_1 and a center of mass (COM) located at l_{c1} measured from joint 1, while link 2 has a mass m_2 and a COM located at l_{c2} measured from joint 2. The angular position of joints 1 and 2 are angles q_{e1} and q_{e2}, respectively. Furthermore, it can be determined that the forward kinematic equations are

$$x = l_1 \cos(q_{e1}) + l_2 \cos(q_{e1} + q_{e2}), \tag{3.1}$$

$$y = l_1 \sin(q_{e1}) + l_2 \sin(q_{e1} + q_{e2}). \tag{3.2}$$

As can be seen from Eqs. (3.1) and (3.2), the forward kinematics equations for a serial robot are nonlinear.

3.1.2 Dynamics

A robotic manipulator's dynamic equations describe the motion of the robot in response to forces and torques. These dynamic equations include mass and inertia properties as well as kinematic parameters which can be used to calculate the associated forces and torques to control the motion of a serial robot.

To formulate the equation of motion describing the dynamic response of a robot to input torques, Lagrangian mechanics are utilized in this research. The *Lagrangian* function $\mathcal{L} \in \Re$ is the difference between the system's kinetic energy $KE \in \Re$ and potential energy $PE \in \Re$

$$\mathcal{L}(\boldsymbol{q}_e, \dot{\boldsymbol{q}}_e) = KE(\boldsymbol{q}_e, \dot{\boldsymbol{q}}_e) - PE(\boldsymbol{q}_e), \tag{3.3}$$

where $\boldsymbol{q}_e \in \Re^n$ is the vector of joint angular positions and $\dot{\boldsymbol{q}}_e \in \Re^n$ is the vector of joint angular velocities. The equation of motion for an $n-$link serial robot can be derived as follows

$$\frac{d}{dt}\left(\frac{\partial \mathcal{L}(\boldsymbol{q}_e, \dot{\boldsymbol{q}}_e)}{\partial \dot{\boldsymbol{q}}_e}\right) - \frac{\partial \mathcal{L}(\boldsymbol{q}_e, \dot{\boldsymbol{q}}_e)}{\partial \boldsymbol{q}_e} = \boldsymbol{u}, \tag{3.4}$$

where $\boldsymbol{u} \in \Re^n$ is the vector of applied joint torques. For the present case, the potential energy PE of the system is zero since the robot is horizontally-planar (in the human's transverse plane) with gravity acting perpendicular to the plane of operation. The

kinetic energy KE is the sum of the linear and rotational kinetic energy of the two links which can be expressed as

$$KE = \frac{1}{2}\left[(m_1 v_{c_1}^2 + I_1 \dot{q}_{e1}^2) + (m_2 v_{c_2}^2 + I_2 \dot{q}_{e2}^2)\right], \tag{3.5}$$

where v_{c_i} is the linear velocity of link i's COM, $\dot{q}_{ei}$ is the angular velocity of joint i, and I_i is the inertia of link i about link i's COM. Using the derivatives of the forward kinematics equations (3.1) and (3.2) to obtain v_{c_i}, the resulting equation of motion from (3.4) can be written as [99]

$$\underbrace{\begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix}}_{M} \begin{bmatrix} \ddot{q}_{e1} \\ \ddot{q}_{e2} \end{bmatrix} + \underbrace{\begin{bmatrix} -h\dot{q}_{e2} & -h\dot{q}_{e1} - h\dot{q}_{e2} \\ h\dot{q}_{e1} & 0 \end{bmatrix}}_{C} \begin{bmatrix} \dot{q}_{e1} \\ \dot{q}_{e2} \end{bmatrix} = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}, \tag{3.6}$$

where

$$M_{11} = m_1 {l_{c_1}}^2 + I_1 + m_2[{l_1}^2 + {l_{c_2}}^2 + 2 l_1 l_{c_2} \cos(q_{e2})] + I_2, \tag{3.7}$$

$$M_{22} = m_2 l_{c_2}^2 + I_2, \tag{3.8}$$

$$M_{12} = M_{21} = m_2 l_1 l_{c_2} \cos(q_{e2}) + m_2 {l_{c_2}}^2 + I_2, \tag{3.9}$$

$$h = m_2 l_1 l_{c_2} \sin(q_{e2}). \tag{3.10}$$

Therefore, an n-link horizontally-planar rigid robot's equation of motion can be generally represented by the Euler-Lagrange equation [99] as follows

$$M(\boldsymbol{q}_e)\ddot{\boldsymbol{q}}_e + C(\boldsymbol{q}_e, \dot{\boldsymbol{q}}_e)\dot{\boldsymbol{q}}_e = \boldsymbol{u}, \tag{3.11}$$

where $M(\boldsymbol{q}_e) \in \Re^{n \times n}$ is the inertia matrix calculated from the mass and geometry of the robot, $\ddot{\boldsymbol{q}}_e \in \Re^n$ is the joint angular acceleration vector, and $C(\boldsymbol{q}_e, \dot{\boldsymbol{q}}_e) \in \Re^{n \times n}$ is the matrix containing the Coriolis effects and centrifugal torques.

3.2 System Stability Analysis

It is important to know if a control system is stable in order to justify its usefulness and to alleviate hazardous behaviour. A common and powerful tool for studying a nonlinear and non-autonomous (time-varying) system's stability is using Lyapunov's Stability Theory [151, 152, 153].

3.2.1 Lyapunov's Direct Method

In order to gain knowledge about the stability of a nonlinear and non-autonomous (time-variant) dynamic system, Lyapunov's Direct Method can be utilized. It is based on the "fundamental physical observation: if the total energy of a mechanical (or electrical) system is continuously dissipated, then the system, ..., must eventually settle down to an equilibrium point" [153]. This method, therefore, employs the formulation of a scalar function, the Lyapunov function V, formulated in the phase space and resembling an energy function. This function determines the stability properties of a nonlinear and non-autonomous system by analysing how the function changes over time. There is no general approach to define the Lyapunov function for a system, thus trial-and-error, intuition, and experience is used to find an appropriate candidate. The properties for V are:

1. It is positive definite: $V(\boldsymbol{x}) > 0 \; \forall \; \boldsymbol{x} \neq 0$;
2. $V(0) = 0$;
3. $V(\boldsymbol{x}) \to \infty$ when $||\boldsymbol{x}|| \to \infty$;
4. It is radial unbounded with respect to variable $\boldsymbol{x}$.

For the stability of a nonlinear and non-autonomous systems, the derivative of the candidate Lyapunov function is required to be negative semi-definite, meaning that its energy decreases:

$$\dot{V}(\boldsymbol{x}) \leq 0. \tag{3.12}$$

One of the most significant advantages of using the Lyapunov function for nonlinear systems is that, in doing so, it is possible to characterize the stability behaviour of the system without having to solve the more complicated nonlinear differential equation of the system.

3.2.2 Barbălat's Lemma

A function being negative semidefinite, or having a finite limit, or having its derivative converge to 0, however, is not sufficient to guarantee asymptotic stability. To ensure convergence, Barbălat's Lemma [153] requires that $\dot{V}$ be uniformly continuous

(smooth). A convenient way to check this condition is by invoking the second derivative of the Lyapunov function $\ddot{V}$ and verifying that it exists and is bounded. Thus, if the scalar Lyapunov candidate function $V(\boldsymbol{x})$ satisfies the following conditions:

- $V(\boldsymbol{x})$ is lower bounded
- $\dot{V}(\boldsymbol{x})$ is negative semi-definite and
- $\ddot{V}(\boldsymbol{x})$ exists and is bounded (proving that $\dot{V}(\boldsymbol{x})$ is uniformly continuous in time)

then

$$\dot{V}(\boldsymbol{x}) \to 0 \quad as \quad t \to \infty \tag{3.13}$$

A new variation on Barbălat's Lemma [154] stated that it is sufficient for a function to be square-integrable and have a bounded derivative to converge asymptotically to zero.

Lyapunov stability analysis was used in this thesis to prove the stability of the proposed ARII and RBFNN-I controllers.

Chapter 4

Problem Formulation

4.1 Problem Scope

There are many control challenges in successfully enabling a therapist to use an upper-limb robotic exoskeleton at one location to physically assist a patient using an upper-limb robotic exoskeleton at another location. One important control challenge is estimating the unknown nonlinear dynamic model parameters of the system in order to compensate for them. Furthermore, varying dynamic parameters, disturbances to the system such as friction, noise, and inevitable realities such as quantization, sampling of the sensor data, and time delays should be addressed as well. The physical human-robot connection also has an effect in the transference of torques. Additionally, the comfort and safety of the humans in the control loop is essential. Moreover, it is highly beneficial to have a high-fidelity telepresence sensation for the therapist. High-fidelity telepresence can elevate the quality of the therapy delivered to the patient by increasing its effectiveness and safety for the patient as well as by providing the therapist with more meaningful and insightful awareness of the patient's physical efforts and abilities.

4.2 System Overview

The overall system diagram for bilateral telerehabilitation with upper-limb robotic exoskeletons is shown in Fig. 4.1. In this system, the patient's arm is secured to the master robot exoskeleton, while the therapist's arm is attached to the slave robot exoskeleton. A desired motion trajectory specified in terms of joint angular position, velocity, and acceleration is sent to both the master side ($\boldsymbol{q}_d$, $\dot{\boldsymbol{q}}_d$, and $\ddot{\boldsymbol{q}}_d$) and the slave side ($\boldsymbol{q}_{desL}$ and $\dot{\boldsymbol{q}}_{desL}$). The net human torque $\boldsymbol{\tau}_{net}$ between the patient and the therapist is calculated and sent to the master controller along with the desired motion trajectory and the actual joint angular positions and velocities of the master

system ($\boldsymbol{q}_m$ and $\dot{\boldsymbol{q}}_m$). The master robot controller then produces the necessary torque output command $\boldsymbol{u}_m$ to the robotic exoskeleton to move the exoskeleton as desired. The resulting joint angular position and velocity of the master system ($\boldsymbol{q}_m$ and $\dot{\boldsymbol{q}}_m$) are conveyed across the communication channel to the slave side ($\boldsymbol{q}_{sd}$ and $\dot{\boldsymbol{q}}_{sd}$). The slave controller uses this information in conjunction with the actual position and velocity of the slave robotic exoskeleton to generate a torque signal $\boldsymbol{u}_s$ that moves the slave robot ($\boldsymbol{q}_s$ and $\dot{\boldsymbol{q}}_s$) to track the motion of the master robot. As a result, the therapist can experience the master robot's motion on the slave side. In bilateral telerehabilitation, the therapist can choose to apply torques $\boldsymbol{\tau}_t$ which are transmitted to the master side as $\boldsymbol{\tau}_{refl}$ so that the patient can receive assistance from the therapist to reach the desired target position.

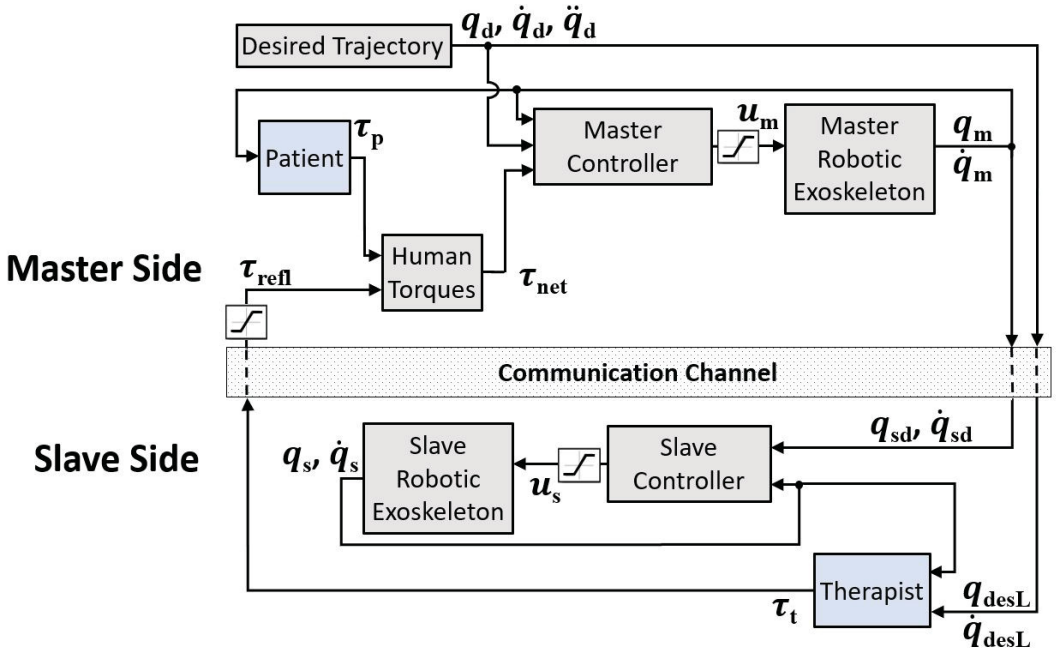


Figure 4.1: Overview of a telerehabilitation system showing the patient on the master side and the therapist on the slave side

4.3 System Models

To develop and implement the control methods covered in this research it was necessary to first develop models of the key components of the system.

4.3.1 Robotic Exoskeleton

Chapter 3 presented the development of the fundamental forward kinematic and dynamic equations of a horizontally-planar robot. By additionally considering the torques applied on each robot joint by the patient ($\boldsymbol{\tau}_p \in \Re^n$) on the master side and the therapist ($\boldsymbol{\tau}_t \in \Re^n$) on the slave side, as well as the influences from joint friction and other external disturbances ($\boldsymbol{\tau}_d \in \Re^n$), the dynamic model for the master and slave robotic exoskeletons can be expressed as

$$M_m(\boldsymbol{q}_e)\ddot{\boldsymbol{q}}_e + C_m(\boldsymbol{q}_e, \dot{\boldsymbol{q}}_e)\dot{\boldsymbol{q}}_e + \boldsymbol{\tau}_d = \boldsymbol{u}_m - \boldsymbol{\tau}_p, \tag{4.1}$$

and

$$M_s(\boldsymbol{q}_e)\ddot{\boldsymbol{q}}_e + C_s(\boldsymbol{q}_e, \dot{\boldsymbol{q}}_e)\dot{\boldsymbol{q}}_e + \boldsymbol{\tau}_d = \boldsymbol{u}_s - \boldsymbol{\tau}_t, \tag{4.2}$$

where subscripts m and s correspond to master and slave, respectively. Furthermore the equations of motion have the following fundamental properties

Property 1 *Positive Definite Symmetric Inertia Matrix:* $M(\boldsymbol{q}) = M(\boldsymbol{q})^T > 0$.

Property 2 *Skew-symmetry of* $\dot{M}(\boldsymbol{q}) - 2C(\boldsymbol{q}, \dot{\boldsymbol{q}}) = 0$.

4.3.2 Patient Dynamic Model

The Kelvin-Voigt viscoelastic model is commonly used in human-robot systems to describe the mechanical behaviour of muscle tissue and activation [155]. It is comprised of a Newtonian damper (D) and Hookean elastic spring (K) connected in parallel as shown in Fig. 4.2.

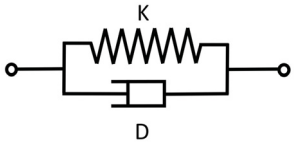


Figure 4.2: Kelvin-Voigt viscoelastic model

This model was incorporated in the current work for the patient model as follows

$$M_p(\boldsymbol{q}_p)\ddot{\boldsymbol{q}}_p + C_p(\boldsymbol{q}_p, \dot{\boldsymbol{q}}_p)\dot{\boldsymbol{q}}_p + K_p(\boldsymbol{q}_p - \boldsymbol{q}_o) + D_p\dot{\boldsymbol{q}}_p = \boldsymbol{\tau}_p, \tag{4.3}$$

where $M_p(\boldsymbol{q}_p) \in \Re^{n\times n}$ is the inertia matrix of the patient's arm and $C_p(\boldsymbol{q}_p, \dot{\boldsymbol{q}}_p) \in \Re^{n\times n}$ is the Coriolis effects and centrifugal torques of the patient's arm. K_p is a diagonal positive definite matrix containing the stiffness of the patient's elbow and wrist joints, while D_p is a diagonal positive definite matrix containing the damping of the patient's elbow and wrist joints. $\boldsymbol{q_o}$ is the patient's resting reference joint angular position vector, $\boldsymbol{q}_p$ and $\dot{\boldsymbol{q}}_p$ are the patient's actual joint angular position and velocity vectors, respectively, and $\boldsymbol{\tau}_p$ has the opposite sign to (4.1) because, considering Newton's third law, it is the torque that the exoskeleton applies on the patient which is equal but opposite to the torque $\boldsymbol{\tau}_p$ shown in (4.1).

4.3.3 Physical Human-Machine Interaction (pHMI)

Being physically connected to − and moving in concert with − a robotic exoskeleton is akin to the backyard game called the three-legged race (Fig. 4.3). In this game, two people have one of their legs tied together and hold their upper bodies together so that they can move as one coordinated unit. Similarly, upper-limb robotic exoskeletons involve the physical coupling of two parallel, independent dynamic systems (the robot and the human) and, as a result, it is important to address the transmission of forces between them. The connections along the limbs should be comfortable but snug enough to minimize 'play' and transfer the desired forces appropriately. In [156], it is stated that the mechanical properties of this connection can depend on many factors including the placement of the actuators on the robot in relation to the patient's joints. An exaggerated analogy to a poorly-matched structurally-parallel dynamic system would be when the two partners in the three-legged race are of significantly different height and/or have significantly-different upper or lower leg link lengths − such as when pairing an adult with a child. Additionally, the contact surfaces and the firmness of the patient's arm − which can vary with age, sex, weight, and overall physical condition − can affect the physical connection and transference of the forces applied. Figs. 4.4 and 4.5 show how the exoskeleton and the human arm have different joint angular positions q_{e_i} and q_{p_i}, respectively, due to the dynamics of the physical human-exoskeleton interaction (pHMI). The connection between the patient arm and the robot exoskeleton occurs at the two force sensors where, as shown in Fig. 4.4, the patient applies forces F_1 and F_2 on the exoskeleton and, as shown in Fig. 4.5, the

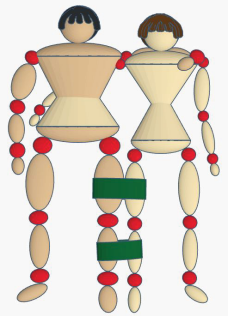


Figure 4.3: Three-legged race: two participants having their legs bound together and holding on to each other to move as one

exoskeleton applies equal but opposite forces on the patient. The dynamics of the connection between the patient arm and the robot exoskeleton arm can be modeled as a viscoelastic spring-damper mechanism [157]. Therefore, in this thesis the following exoskeleton pHMI modeling approach is developed assuming that the resulting force sensor readings F_1 and F_2 on link 1 and link 2 are

$$F_1 = k_1 \Delta s_1 + d_1 \Delta \dot{s}_1, \tag{4.4}$$

$$F_2 = k_2 \Delta s_2 + d_2 \Delta \dot{s}_2. \tag{4.5}$$

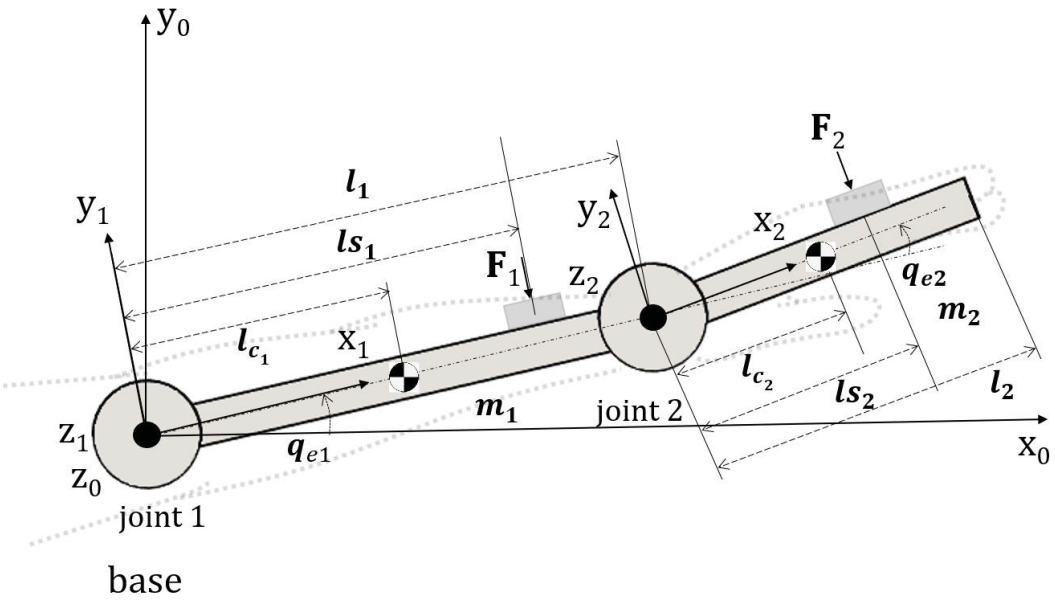


Figure 4.4: Exoskeleton diagram

where i=1 or 2 for the joint or link number, k_i is the stiffness, and d_i is the viscous damping of the viscoelastic robot-patient connection for joint i which act on Δs_i and

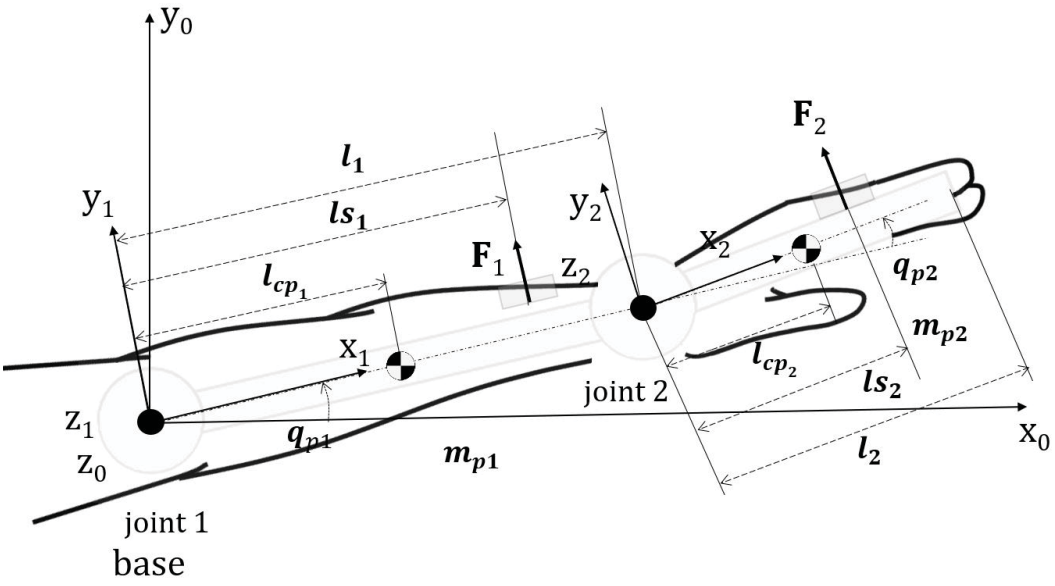


Figure 4.5: Human arm diagram

$\Delta\dot{s}_i$ corresponding to the arm position and rate differences between the exoskeleton and the patient's arm at the location of force sensor i. These position and rate differences can be calculated as

$$\Delta s_i = l_{s_i}(q_{e_i} - q_{p_i}), \tag{4.6}$$

$$\Delta\dot{s}_i = l_{s_i}(\dot{q}_{e_i} - \dot{q}_{p_i}), \tag{4.7}$$

where q_{e_i} and $\dot{q}_{e_i}$ are the exoskeleton angular position and velocity of joint i, while q_{p_i} and $\dot{q}_{p_i}$ are the patient angular position and velocity of joint i. l_{s_i} is the location along link i of force sensor i from joint i. The position vector l_{s_i} along link i from joint i to sensor i can be defined in terms of the exoskeleton joint angles using forward kinematics (3.1) and (3.2) as follows

$$\boldsymbol{l}_{s_1} = \begin{bmatrix} l_{s_1}\cos q_{e_1} \\ l_{s_1}\sin q_{e_1} \end{bmatrix}, \tag{4.8}$$

$$\boldsymbol{l}_{s_2} = \begin{bmatrix} l_{s_2}\cos(q_{e_1} + q_{e_2}) \\ l_{s_2}\sin(q_{e_1} + q_{e_2}) \end{bmatrix}, \tag{4.9}$$

where subscripts 1 and 2 refer to the link number.

Having used the viscoelastic model to determine the size of the force sensor readings F_1 and F_2, their directions − which are perpendicular to $\boldsymbol{l}_{s_1}$ and $\boldsymbol{l}_{s_2}$ − can be

calculated from (4.8) and (4.9) so that the corresponding unit vectors $\overrightarrow{\boldsymbol{F}}_1$ and $\overrightarrow{\boldsymbol{F}}_2$ in the direction of these forces are

$$\overrightarrow{\boldsymbol{F}}_1 = \frac{\begin{bmatrix} l_{s_1} \sin q_{e_1} \\ -l_{s_1} \cos q_{e_1} \end{bmatrix}}{\left\| \begin{bmatrix} l_{s_1} \sin q_{e_1} \\ -l_{s_1} \cos q_{e_1} \end{bmatrix} \right\|}, \tag{4.10}$$

$$\overrightarrow{\boldsymbol{F}}_2 = \frac{\begin{bmatrix} l_{s_2} \sin(q_{e_1} + q_{e_2}) \\ -l_{s_2} \cos(q_{e_1} + q_{e_2}) \end{bmatrix}}{\left\| \begin{bmatrix} l_{s_2} \sin(q_{e_1} + q_{e_2}) \\ -l_{s_2} \cos(q_{e_1} + q_{e_2}) \end{bmatrix} \right\|}. \tag{4.11}$$

The resulting interaction force vectors become $\boldsymbol{F}_1 = F_1 \overrightarrow{\boldsymbol{F}}_1$ and $\boldsymbol{F}_2 = F_2 \overrightarrow{\boldsymbol{F}}_2$. Using $\boldsymbol{F}_1$ and $\boldsymbol{F}_2$, the corresponding patient torque $\boldsymbol{\tau}_p$ can then be determined using the Jacobian matrices J_1 and J_2. To derive the Jacobian matrices, the following steps are taken using the chain rule. Given the (x, y) position of a point on a 2-DOF robot where $x(q_1(t), q_2(t))$ and $y(q_1(t), q_2(t))$

$$v_x = \frac{dx}{dt} = \frac{\partial x}{\partial q_1}\frac{dq_1}{dt} + \frac{\partial x}{\partial q_2}\frac{dq_2}{dt} = \frac{\partial x}{\partial q_1}\dot{q}_1 + \frac{\partial x}{\partial q_2}\dot{q}_2, \tag{4.12}$$

$$v_y = \frac{dy}{dt} = \frac{\partial y}{\partial q_1}\frac{dq_1}{dt} + \frac{\partial y}{\partial q_2}\frac{dq_2}{dt} = \frac{\partial y}{\partial q_1}\dot{q}_1 + \frac{\partial y}{\partial q_2}\dot{q}_2. \tag{4.13}$$

Combining (4.12) and (4.13) as a vector-matrix expression produces

$$\boldsymbol{v} = \begin{bmatrix} v_x \\ v_y \end{bmatrix} = \underbrace{\begin{bmatrix} \dfrac{\partial x}{\partial q_1} & \dfrac{\partial x}{\partial q_2} \\ \\ \dfrac{\partial y}{\partial q_1} & \dfrac{\partial y}{\partial q_2} \end{bmatrix}}_{J} \underbrace{\begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix}}_{\dot{\boldsymbol{q}}} = J\dot{\boldsymbol{q}}, \tag{4.14}$$

where J is the Jacobian matrix. In addition to relating velocity at a point on the robot to joint angular velocities, the Jacobian can be used to relate the force $\boldsymbol{F}$ at a point on the robot to joint torques $\boldsymbol{\tau}$ [158] as

$$\boldsymbol{\tau} = \boldsymbol{J}^T \boldsymbol{F}. \tag{4.15}$$

For the current case of a 2-DOF robot, to attain the Jacobian matrices, the partial differentials of the forward kinematic equations (3.1) and (3.2) for x and y are taken at the force sensor location l_{s_1} and l_{s_2} resulting in

$$J_1 = \begin{bmatrix} -l_{s_1} \sin q_1 & 0 \\ l_{s_1} \cos q_1 & 0 \end{bmatrix}, \tag{4.16}$$

and

$$J_2 = \begin{bmatrix} -l_1 \sin q_1 - l_{s_2} \sin(q_1 + q_2) & -l_{s_2} \sin(q_1 + q_2) \\ l_1 \cos q_1 + l_{s_2} \cos(q_1 + q_2) & l_{s_2} \cos(q_1 + q_2) \end{bmatrix}, \tag{4.17}$$

so that

$$\boldsymbol{\tau}_p = J_1^T \boldsymbol{F}_1 + J_2^T \boldsymbol{F}_2. \tag{4.18}$$

4.3.4 Therapist Dynamic Model

On the slave side, the therapist is modeled as an integral controller so that the therapist acts on the cumulative position error to assist the patient as needed [159]. A diagonal positive gain matrix K_t was incorporated to represent the therapist's urgency to assist the patient as follows

$$\boldsymbol{\tau}_t = K_t \int (\boldsymbol{q}_{desL} - \boldsymbol{q}_s), \tag{4.19}$$

where $\boldsymbol{\tau}_t$ is the calculated therapist's assistive torque, $\mathbf{q}_{desL}$ is the reflected desired joint angular position vector, and $\mathbf{q}_s$ is the actual joint angular position of the slave robot. The higher the gain K_t, the 'quicker' the therapist tries to help the patient.

4.4 Robotic Setup and Models for Validation

The proposed control methods developed in this research were first validated in single-system scenarios on a 1-DOF Quanser QUBE simulation testbed and a 1-DOF Quanser QUBE experimental testbed. Then, the control methods were validated for a higher-degrees-of-freedom system using a 2-DOF simulated exoskeleton testbed. Furthermore, the methods were validated and explored using telerehabilitation scenarios featuring two experimental Quanser QUBES with a 1-DOF Driving Force driving wheel for real-time human intervention emulating the therapist's torque. Finally, to validate the control methods for telerehabilitation using multi-degrees-of-freedom

systems, the methods were implemented on 2-DOF simulated upper-limb robotic exoskeleton systems.

4.4.1 1-DOF Quanser QUBE for Experimental Validation

To prevent damage to the equipment during experimental testing, a simulated model of the Quanser QUBE 1-DOF robot was first created. This model was developed in MATLAB/Simulink and, after validating and tuning the controllers on the Quanser QUBE simulator, the control methods were experimentally validated on the actual Quanser QUBE experimental testbed.

Apparatus, Model and Experimental Setup

The Quanser QUBE shown in Fig. 4.6 is a fully-integrated 1-DOF Robotic platform used in labs around the world for control and mechatronics experiments. The experimental testbed consists of a direct-drive brushed DC motor, an optical encoder, an inertial disk attached to the motor shaft, an internal data acquisition system, and an amplifier [160]. Real-time sensor data and control commands were relayed between MATLAB/Simulink and the Quanser QUBE through the Quanser QUARC modules. The devices were physically connected to the PC through USB.

Figure 4.6: The Quanser QUBE setup

In order to create a model of the Quanser QUBE, an experimental system identification and validation process was carried out using MATLAB/Simulink. A series of different closed-loop step responses were created to generate experimental input-output data for model identification. Fig. 4.7 a) shows the input voltage and Fig. 4.7 b) shows the corresponding output angular position as a function of time.

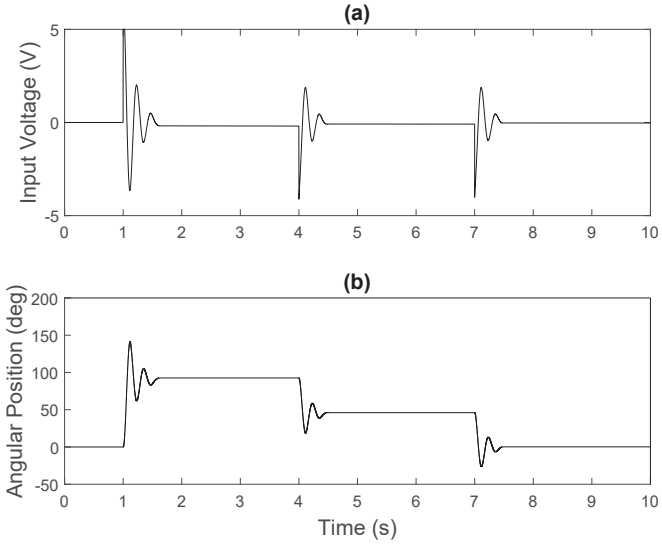


Figure 4.7: Quanser QUBE model identification (a) experimental input: voltage, and (b) experimental output: angular position

The system identification approach updated the model parameters by applying the Nelder-Mead Simplex nonlinear optimization algorithm [161] to a cost function which minimized the 2-norm of the difference between the experimental and the simulated data. The initial model was assumed to be a second-order transfer function which relates the output angular position q to the input motor voltage u as follows

$$\frac{q}{u} = \frac{c_1}{c_2 s^2 + s}, \tag{4.20}$$

where c_1 and c_2 are constants characterizing the mass, inertia, and viscous friction in the system. Written as a second-order differential equation, (4.20) becomes

$$u = \frac{c_2}{c_1}\ddot{q} + \frac{1}{c_1}\dot{q}. \tag{4.21}$$

It was found, however, that a more complex viscous-Coulomb friction model provided better agreement between the identified model and the experimental apparatus. This model incorporates an asymmetric Coulomb friction coefficient c_3 which has different values depending on the direction of rotation. The corresponding differential equation is written as

$$u = \frac{c_2}{c_1}\ddot{q} + \frac{1}{c_1}\dot{q} + c_3 \; sgn(\dot{q}), \tag{4.22}$$

where c_3 has both clockwise (c_{3cw}) and counter-clockwise (c_{3ccw}) values that were identified.

Fig. 4.8 shows a comparison between the experimental results and the simulated identified model results using the same experimental input data shown in Fig. 4.7(a). As can be seen in Fig. 4.8, the identified model closely agrees with the experimental results.

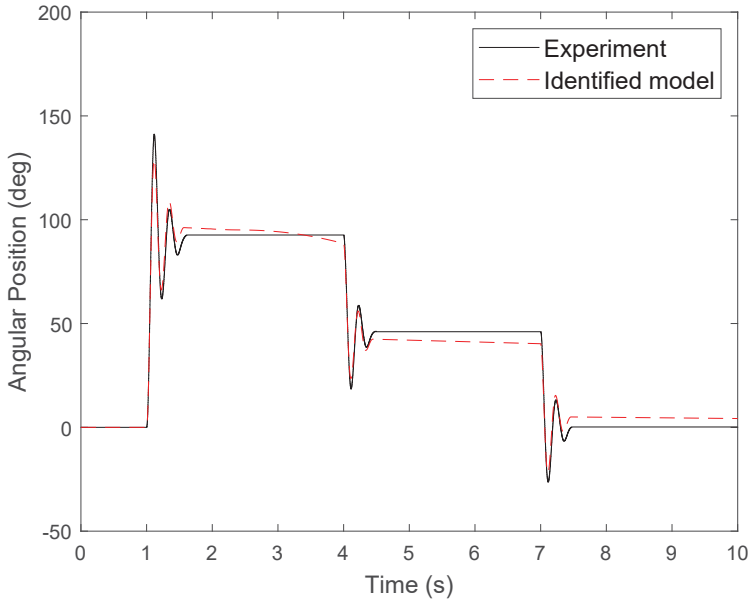


Figure 4.8: Experimental versus simulated output data used to identify the model of the Quanser QUBE

The identified model was subsequently validated on different closed-loop experimental data as shown in Fig. 4.9 using the proposed RBFNN-I method in Chapter 6. Figs. 4.9(a) and (b) clearly show a good level of agreement with an RMSE value between the simulated model and the experimental results of 0.63°, which attests to the accuracy of the identified model. This model identification process was executed in order to identify the specific parameters for both the master and the slave Quanser QUBE 1-DOF robots utilized throughout this thesis.

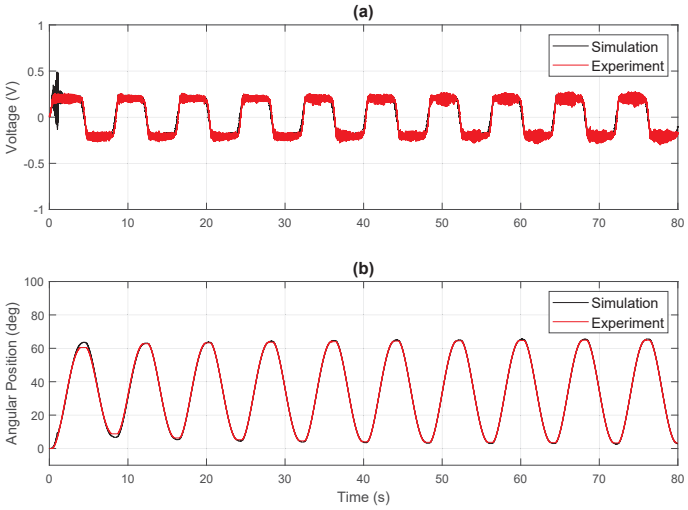


Figure 4.9: Model validation showing (a) control input and (b) position output of the simulated versus experimental closed-loop system using the RBFNN-I control method

Optoforce Force Sensor

For the scope of this work, it was desired to include noise on the patient and therapist torque signals in order to validate the robustness of the controllers. Thus, experiments were carried out with the OptoForce sensor [162] shown in Figs. 4.10 to characterize the sensor noise. Fig. 4.11(a) shows the measured force output signal as a function of time when the sensor was stationary. The standard deviation from the experiments was calculated to be 0.0076 N, with a mean of 0.0129 N, and a quantization interval of 0.012 N/count. For the scope of the work in this thesis, it was assumed that this noise could be modeled as a normally-distributed Gaussian noise signal with similar mean, standard deviation, and quantization values. Fig. 4.11(b) shows the resulting simulated sensor noise used in all the Quanser QUBE simulations and experiments in this thesis.

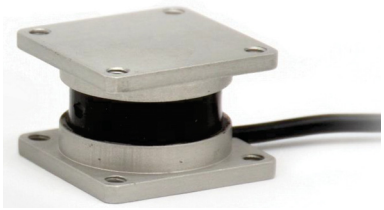


Figure 4.10: OptoForce force sensor

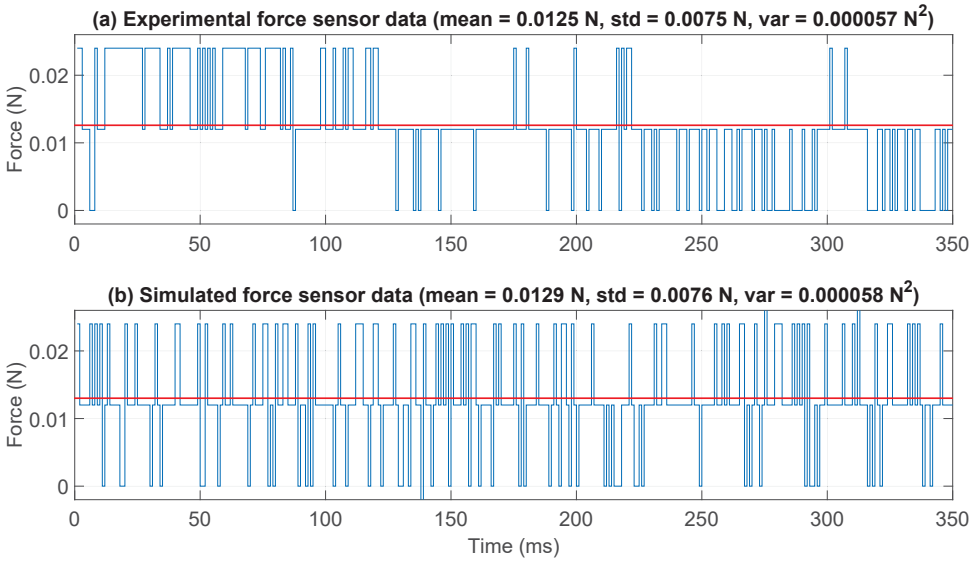


Figure 4.11: Experimental and simulated force sensor data

4.4.2 2-DOF Upper-Limb Robotic Exoskeleton

The control methodologies developed in this thesis were ultimately validated on a 2-DOF upper-limb horizontally-planar rehabilitation robotic exoskeleton simulation testbed. The exoskeleton model used in the simulator was designed in SolidWorks and is shown in Fig. 4.12. Referring to the nomenclature used in Fig. 4.4, the robot link lengths (l_1 and l_2) were adjusted to match those of the author's arm dimensions. The COM locations for each robot link (l_{c_1}, l_{c_2}), masses (m_1, m_2), and moments of inertia about the COM (I_1, I_2) of each robot link were attained from SolidWorks and are summarized in Table 4.1.

The human's forearm was mathematically modeled as a right conical frustum while the hand was modeled as an orthogonal slab similar to [163]. The dimensions of these

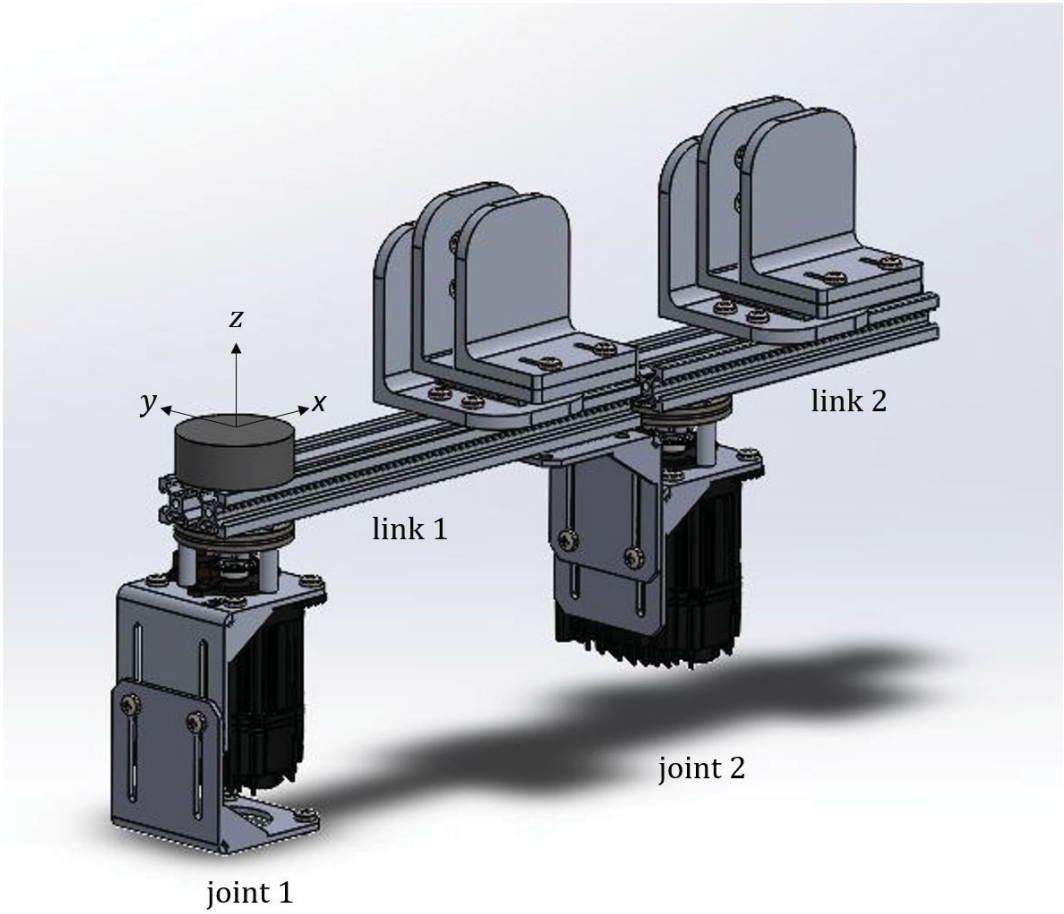


Figure 4.12: 2-DOF elbow-wrist robotic exoskeleton conceptual design used in simulations

human body-part models were based on the author's physical dimensions. The mass of the human's forearm and hand were attained using body mass percentages as prescribed in [164]. Fig. 4.5 illustrates the parameters pertaining to the human arm which are also summarized in Table 4.1.

The exoskeleton elbow (joint 1) has a value of 0° when the elbow is flexed towards the operator's chest wall and 90° when the arm is extended away from the operator. The exoskeleton wrist (joint 2) has a value of $-45°$ when the hand is flexed towards the operator's chest wall, and 45° when it is extended away from the body.

The same exoskeleton design and human operator model is used on both the master and the slave sides, thus the masses, lengths, and inertias used in the simulator are the same on both sides.

Table 4.1: Simulation Parameters

Parameter	Value	Parameter	Value
Exoskeleton:		Patient:	
l_1	0.270 m	l_{p1}	0.270 m
l_2	0.144 m	l_{p2}	0.150 m
l_{c_1}	0.206 m	l_{c_p1}	0.092 m
l_{c_2}	0.064 m	l_{c_p2}	0.075 m
m_1	1.883 kg	m_{p1}	0.903 kg
m_2	0.690 kg	m_{p2}	0.367 kg
I_1	0.011 kgm^2	I_{p1}	0.005 kgm^2
I_2	0.001 kgm^2	I_2	0.001 kgm^2

Optoforce Force Sensor

The Optoforce noise data was also used in all the simulations using the 2-DOF Robotic Exoskeletons. Fig. 4.13 shows the conceptual design of the hand sub-assembly with the Optoforce force sensor mounted between a bracket fasten to the exoskeleton and the hand sub-assembly which is mounted on a linear bearing so that the force between the exoskeleton and the human occurs in only one dimension. The force sensors in the 2-DOF exoskeleton design are located at l_{s_1} along link 1 and l_{s_2} along link 2 to measure the elbow and wrist human torque input applied to the exoskeleton and are used in all the 2-DOF exoskeleton simulations in this thesis.

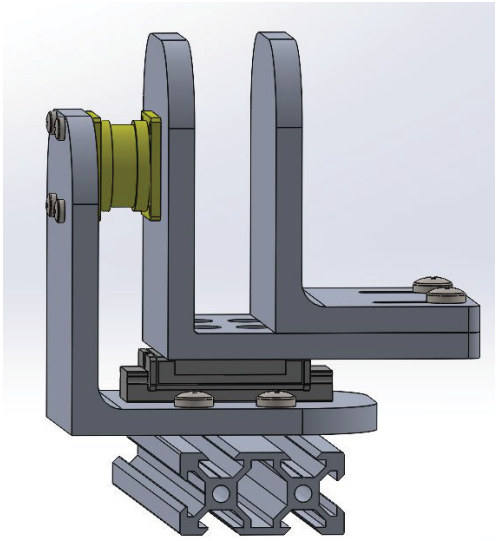


Figure 4.13: OptoForce force sensor (yellow) in the hand sub-assembly

It is important to note that when a patient's arm is in contact with a robot, safety and comfort is crucial. Hard stops should be implemented to limit the range of motion, adjustable torque overload couplings should be placed on the motor shafts, and an emergency stop button should be located within reach. Furthermore, software limits should be placed on the robot's motion as well as motor control torques and reflected torques. This research implements the model of the 2-DOF Robotic Exoskeleton in simulation, and thus has the safety limits implemented in software. Furthermore, to minimize feelings of entrapment and allow for quick removal of the human's arm out of the device in case of an emergency, the current robotic exoskeleton was cleverly designed to not only be widely adjustable for different shapes and sizes of the patient's forearm and hand, but to also allow the patient's arm to be extracted vertically without compromising the normal operation of therapeutic exercise.

4.5 Telerehabilitation

Equations (4.1) and (4.2) showed the nonlinear robotic exoskeleton dynamics equations for the n-DOF horizontal planar exoskeleton models on the master and the slave side (Fig. 4.1). The telerehabilitation simulations and experiments in this work are focused on the initial rehabilitation stage where the patient cannot yet initiate motion, and any patient-generated torques are due to the involuntary toning (tightness) in the patient's elbow and wrist, which is common in hemiplegia stroke patients at the acute stage of their injury [97]. The toning causes both joints to rest near the patient's chest wall. There are two modes of teleoperation described in the following sections: unilateral (open-loop) and bilateral (with force feedback).

4.5.1 Unilateral Teleoperation

Unilateral teleoperation takes place when the motion of the master side robot is transmitted to the system on the slave side, but there are no torques transmitted from the slave side to the master side. This unilateral teleoperation is referred to as passive-mode telerehabilitation. During this stage, the therapist on the slave side observes and assesses the patient's range of motion and level of toning in the joints while the master system aims to track the prescribed motion trajectories. Since there is no force feedback in this scenario, any delays present in the communication

channels merely present the master side motion to the therapist later in time with no detrimental consequences to the stability of the system.

4.5.2 Bilateral Teleoperation

Bilateral telerehabilitation enables the therapist on the slave side to provide assistive torques to the patient on the master side. In bilateral teleoperation, delays in the communication channel can have an adverse effect on the system if measures are not taken to ensure stability.

4.6 Performance Metrics of the Control Methods

To quantitatively compare the performance of the proposed control methods, both tracking performance metrics and telepresence fidelity metrics were applied. Telepresence refers to how well the operator *feels* like they are in the remote environment. In the context of virtual reality video games, Basdogan *et al.* [165] points out that telepresence improves the sense of 'togetherness' between players and makes the experience more meaningful than just visual quantitative information being provided to the user. Thus, in telerehabilitation, for perfect telepresence, the therapist would have to apply the same joint torque as the patient to experience the same amount of effort as the patient.

4.6.1 Tracking Performance

Metrics for joint tracking were implemented to gauge the performance of the proposed control methods. Good tracking performance is attained when $\tilde{\xi}_i$, the difference between the desired impedance angular position and the actual robot angular position of the i^{th} joint, is minimal and converging towards zero. The average $\tilde{\xi}_i$ can be quantified using the root mean squared error (RMSE) as follows

$$\tilde{\xi}_{i,RMSE} \triangleq \sqrt{\frac{1}{N}\sum_{n=1}^{N}(\tilde{\xi}_i(n))^2}, \tag{4.23}$$

where N is the number of data points considered. Additionally, to discover the bounds on $\tilde{\xi}_i$, its maximum value is calculated for the i^{th} joint as follows

$$\tilde{\xi}_{i,max} \triangleq max|\tilde{\xi}_i(n)|. \tag{4.24}$$

4.6.2 Telepresence Fidelity

The quantitative comparison of the achieved telepresence in the bilateral telerehabilitation system is performed by looking at the root mean square of the difference between the patient and the therapist torques as follows

$$TP_{RMSE} \triangleq \sqrt{\frac{1}{N}\sum_{n=1}^{N}(\tau_{p_i}(n) + \tau_{refl_i}(n))^2}, \tag{4.25}$$

where N is the number of data points considered, τ_{p_i} is the patient torque, and τ_{refl_i} is the reflected therapist's torque for joint i. Furthermore, the maximum value of the telepresence is calculated as follows

$$TP_{max} \triangleq max\left|\tau_{p_i}(n) + \tau_{refl_i}(n)\right|, \tag{4.26}$$

with zero being the ideal value. The telepresence metrics are calculated from the time that the master robot begins to move due to the therapist's assistive torques.

4.7 Control Objectives

This work endeavors to improve upon the state-of-the-art of upper-limb robotic exoskeleton control for bilateral telerehabilitation with a focused scope of providing higher-fidelity telepresence for the therapist while ensuring good tracking performance.

First, the state-of-the-art upper-limb robotic exoskeleton control method, called adaptive control, is extended to include robust, integral, and impedance control components. The author coined the term Adaptive Robust Integral Impedance (ARII) to describe this controller. Then a second, RBF neural networks based controller is further developed to include adaptive, robust, integral, and impedance elements. The author refers to this proposed controller as Adaptive Robust Integral Radial Basis Function Neural Networks Impedance (RBFNN-I). Simulation and experimental tests are performed and results are compared and discussed for single robotic systems and telerehabilitation scenarios using the Quanser QUBE testbed and the simulated 2-DOF upper-limb robotic exoskeleton testbed.

The overall goal of the proposed controllers is to properly reflect the motion and torques across the communication system between the human-robot systems with

high-fidelity telepresence and converging tracking errors. At the same time, the motions and torques must be safe and stable for the two human operators that are connected to the robotic exoskeletons.

The following lists the control system objectives:

1. Provide higher-fidelity telepresence for the therapist than current robotic telerehabilitation systems found in the literature;
2. Provide good, converging tracking performance;
3. Provide a stable and safe controller with adjustable compliance for telerehabilitation with upper-limb robotic exoskeletons.

The scope of the problem has the following constraints:

1. The communication time-varying delays are bounded;
2. The simulated patient and therapist dynamics are continuous and have limited bandwidth;
3. The forces, position, velocity, and acceleration ranges of the robotic exoskeletons are bounded;
4. All bounded limits are set in software to protect the humans and the hardware;
5. Experimental force sensor data was collected and the corresponding bias and noise signature was simulated for all tests in this work.

Chapter 5

Adaptive Robust Impedance Integral (ARII) Controller-Design, Simulations, and Experiments

5.1 Adaptive Control

The joint-space dynamics of an n-link rigid horizontally-planar robotic exoskeleton with human torque disturbance $\boldsymbol{\tau}_p$ are given by (4.1) which is reproduced here for convenience (with the subscript e and m notation dropped for brevity)

$$M(\boldsymbol{q})\ddot{\boldsymbol{q}} + C(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{\tau}_d = \boldsymbol{u} - \boldsymbol{\tau}_p. \tag{5.1}$$

Due to realities such as measurement inaccuracies, material deformation, as well as thermal, manufacturing, and system assembly variation, the actual kinematic and dynamic parameters of a robotic exoskeleton cannot be precisely known in practice. Adaptive control can be used to adjust and adapt these unknown model parameters provided that one has *a priori* knowledge of the underlying detailed dynamics model that describes how these unknown model parameters are related to the joint angular position, velocity, and acceleration variables. As a result, the following equation is used to model the dynamics

$$M_a(\boldsymbol{q})\ddot{\boldsymbol{q}} + C_a(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{\tau}_d = \boldsymbol{u} - \boldsymbol{\tau}_p, \tag{5.2}$$

where $M_a(\boldsymbol{q})$ and $C_a(\boldsymbol{q}, \dot{\boldsymbol{q}})$ are the best approximations of the actual master robotic exoskeleton ($M(\boldsymbol{q})$ and $C(\boldsymbol{q}, \dot{\boldsymbol{q}})$ in (5.1)) that an adaptive control law can achieve with this model structure. Thus

$$M = M_a + E_M, \tag{5.3}$$

$$C = C_a + E_C, \tag{5.4}$$

where E_M and E_C are the inevitable modeling errors that occur when using a model to represent an actual system.

5.2 Impedance Control

It is desired to make the motion compliant and accommodating to the patient's changing abilities; therefore, impedance control is utilized. A dynamic impedance model [62] can be described as

$$(J_d s^2 + B_d s + K_d)\tilde{\boldsymbol{q}}_{\tau net} = \boldsymbol{\tau}_{net}, \tag{5.5}$$

where J_d, B_d, and $K_d \in \Re^{n\times n}$ are diagonal, positive definite inertia, damping, and stiffness matrices, respectively. These impedance matrices are designed to give a desired stable dynamic response and can be adjusted based on the patient's varying capabilities throughout the course of the therapy journey. In this equation, $\tilde{\boldsymbol{q}}_{\tau net}$ is the designed desired impedance joint angular position tracking error in response to the net external torques $\boldsymbol{\tau}_{net}$ applied on the master robot by the humans in the system, and s is the Laplace differential operator. Letting $J_d = diag[j_1, j_2, ..., j_n]$, $B_d = diag[b_1, b_2, ..., b_n]$, and $K_d = diag[k_1, k_2, ..., k_n]$, the corresponding single-input single-output i^{th} impedance model transfer function is given by [166]

$$\frac{\tilde{q}_{\tau net,i}}{\tau_{net,i}} = \frac{1}{j_i s^2 + b_i + k_i} = \frac{\frac{1}{j_i}}{s^2 + \frac{b_i}{j_i}s + \frac{k_i}{j_i}}. \tag{5.6}$$

Comparing the denominator of the transfer function with the characteristic equation of a second-order system $s^2 + 2\zeta\omega_n s + \omega_n^2$ [166] yields $2\zeta\omega_n = \frac{b_i}{j_i}$ and $\frac{k_i}{j_i} = \omega_n^2$, resulting in

$$b_i = 2\zeta j_i \sqrt{\frac{k_i}{j_i}} = \sqrt{4\zeta^2 j_i k_i}. \tag{5.7}$$

To ensure that the impedance model does not exhibit any oscillations in the presence of $\boldsymbol{\tau}_{net}$, the impedance model should be critically damped ($\zeta = 1$) or over damped ($\zeta > 1$). Fig. 5.1 shows an example of a desired impedance response to a 1 Nm step input command with the i_i, b_i, and k_i values set to 0.013 kgm^2, 0.667 Nms, and 2.67 Nm, respectively. These values yield a damping ratio of $\zeta = 1.77$, which is greater than 1 and thus over damped as desired. Note, if it is desired to tune the response further, a parameter optimization process could be implemented.

The net torque $\boldsymbol{\tau}_{net}$ applied by the humans onto the master robotic exoskeleton is used by the impedance model to dictate the impedance desired joint angular position,

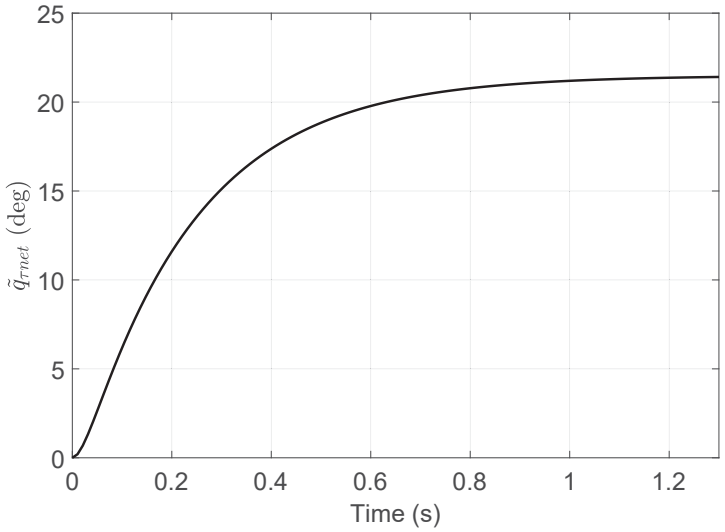


Figure 5.1: Designed impedance response to a 1 Nm step input

velocity, and acceleration trajectory: $\boldsymbol{q}_{d,imp}$, $\dot{\boldsymbol{q}}_{d,imp}$, and $\ddot{\boldsymbol{q}}_{d,imp}$. Fig. 5.2 plots the joint angular position and corresponding tracking errors to illustrate the relationships between the desired, actual, and impedance joint positions where:

- $\boldsymbol{q}_d(t)$ is the original desired joint angular position trajectory;
- $\boldsymbol{q}_{d,imp}(t)$ is the new desired joint angular position trajectory considering the impedance response from an applied net human torque disturbance $\boldsymbol{\tau}_{net}$;
- $\boldsymbol{q}(t)$ is the actual joint angular position trajectory;
- $\tilde{\boldsymbol{q}}_{\tau net}(t)$ is the designed desired impedance joint angular position tracking error due to the applied net human torque disturbance $\boldsymbol{\tau}_{net}$;
- $\tilde{\boldsymbol{q}}(t)$ is the actual joint angular position tracking error from the original desired joint angular position trajectory;
- $\tilde{\boldsymbol{\xi}}(t)$ is the joint angular position impedance error, which corresponds to the difference between the actual joint angular position tracking error and the designed desired impedance joint angular position tracking error (and is desired to converge towards zero).

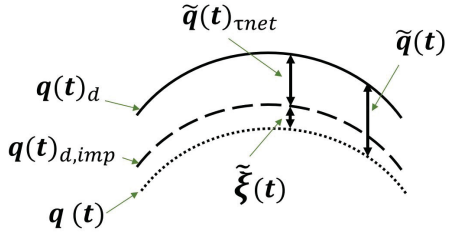


Figure 5.2: Joint angular position and tracking errors

Written out in equation form, we have the following relationships for the angular position and angular velocity tracking errors

$$\tilde{\boldsymbol{q}}(t) \triangleq \boldsymbol{q}(t) - \boldsymbol{q}_d(t), \tag{5.8}$$

$$\dot{\tilde{\boldsymbol{q}}}(t) \triangleq \dot{\boldsymbol{q}}(t) - \dot{\boldsymbol{q}}_d(t). \tag{5.9}$$

The following equations show the relationships between the desired, actual and the designed motion tracking error due to the external human torques

$$\tilde{\boldsymbol{q}}_{\tau net}(t) \triangleq \boldsymbol{q}_{d,imp}(t) - \boldsymbol{q}_d(t), \tag{5.10}$$

$$\dot{\tilde{\boldsymbol{q}}}_{\tau net}(t) \triangleq \dot{\boldsymbol{q}}_{\boldsymbol{d,imp}}(t) - \dot{\boldsymbol{q}}_d(t), \tag{5.11}$$

$$\ddot{\tilde{\boldsymbol{q}}}_{\tau net}(t) \triangleq \ddot{\boldsymbol{q}}_{\boldsymbol{d,imp}}(t) - \ddot{\boldsymbol{q}}_d(t). \tag{5.12}$$

An impedance model controller (IC) tries to ensure that the tracking errors $\tilde{\boldsymbol{q}}(t)$ and $\dot{\tilde{\boldsymbol{q}}}(t)$ match the desired dynamic response $\tilde{\boldsymbol{q}}_{\tau net}(t)$ and $\dot{\tilde{\boldsymbol{q}}}_{\tau net}(t)$ when external torques from the humans $\boldsymbol{\tau}_{net}(t)$ are applied to the system. The impedance angular position error $\tilde{\boldsymbol{\xi}}(t)$ can then be defined as

$$\tilde{\boldsymbol{\xi}}(t) \triangleq \tilde{\boldsymbol{q}}(t) - \tilde{\boldsymbol{q}}_{\tau net}(t), \tag{5.13}$$

and the impedance angular velocity error $\dot{\tilde{\boldsymbol{\xi}}}$ can be defined as

$$\dot{\tilde{\boldsymbol{\xi}}}(t) \triangleq \dot{\tilde{\boldsymbol{q}}}(t) - \dot{\tilde{\boldsymbol{q}}}_{\tau net}(t). \tag{5.14}$$

The objective of the controller is to properly adjust the motor torques $\boldsymbol{u}$ in (5.2) so that

$$\lim_{t\to\infty} \tilde{\boldsymbol{\xi}}(t) = 0, \quad \lim_{t\to\infty} \dot{\tilde{\boldsymbol{\xi}}}(t) = 0. \tag{5.15}$$

If there are no external torques from the human operator, then $\tilde{\boldsymbol{q}}_{\tau net}(t)$ and $\dot{\tilde{\boldsymbol{q}}}_{\tau net}(t)$ are zero and the impedance errors $\tilde{\boldsymbol{\xi}}(t)$ and $\dot{\tilde{\boldsymbol{\xi}}}(t)$ in (5.13) and (5.14) effectively become

the tracking errors $\tilde{\boldsymbol{q}}(t)$ and $\dot{\tilde{\boldsymbol{q}}}(t)$ in (5.8) and (5.9). Using IC results in less accurate tracking but a more compliant and safe-to-use system when humans are physically interacting with the robots.

In telerehabilitation applications, IC enables the therapist, based on their professional judgment, to customize how far the patient's joints are extended based on the patient's current abilities, and control the level of assistance provided to the patient by the robot while tracking desired trajectories.

5.3 ARII Controller Design

It is desired to design the ARII controller to activate the robotic exoskeleton's motors according to the prescribed impedance model under real-time human torque input in the presence of unknown dynamic modeling parameters and external disturbances. Fig. 5.3 shows the corresponding control system diagram when the ARII controller uses the desired and actual joint motion in combination with measured human torques applied to the robotic exoskeleton to generate the control signal $\boldsymbol{u}$.

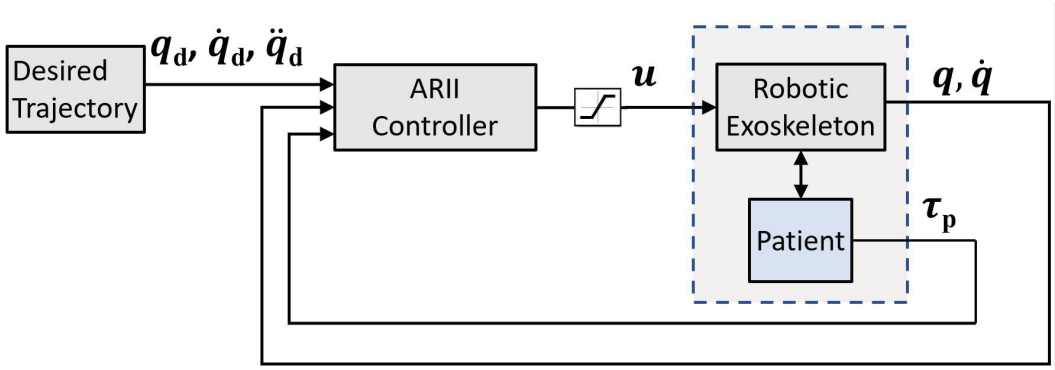


Figure 5.3: ARII control system diagram

Let the proposed ARII control law be

$$\begin{aligned}\boldsymbol{u} \triangleq\ & \hat{M}_a(\boldsymbol{q})\dot{\boldsymbol{v}} + \hat{C}_a(\boldsymbol{q},\dot{\boldsymbol{q}})\boldsymbol{v} + \hat{M}_a(\boldsymbol{q})\ddot{\tilde{\boldsymbol{q}}}_{\tau net} + \hat{C}_a(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\tilde{\boldsymbol{q}}}_{\tau net} \\ & - K_p\boldsymbol{r} - K_i\int \boldsymbol{r}dt - \boldsymbol{\tau}_r + \boldsymbol{\tau}_{p,meas},\end{aligned} \tag{5.16}$$

where $\hat{M}_a(\boldsymbol{q}) \in \Re^{n\times n}$ and $\hat{C}_a(\boldsymbol{q},\dot{\boldsymbol{q}}) \in \Re^{n\times n}$ are estimates of the inertia matrix $M_a(\boldsymbol{q})$ and the Coriolis and centripetal torques matrix $C_a(\boldsymbol{q},\dot{\boldsymbol{q}})$, respectively. $K_p \in \Re^{n\times n}$ and $K_i \in \Re^{n\times n}$ are diagonal, positive definite proportional and integral gain matrices,

$\boldsymbol{\tau}_r \in \Re^n$ is the robust control term, and $\boldsymbol{\tau}_{p,meas} \in \Re^n$ is the measured human disturbance torque. This adaptive control law is based on [167] for robotic manipulators with the addition of an integral control term and a robust control term to address the uncertainty in the measured torque. Furthermore, the adaptive control law is extended to incorporate the impedance control by defining virtual reference velocity and acceleration vectors $\boldsymbol{v} \in \Re^n$ and $\dot{\boldsymbol{v}} \in \Re^n$ in (5.16) as

$$\boldsymbol{v} \triangleq \dot{\boldsymbol{q}}_d - \Lambda\tilde{\boldsymbol{\xi}}, \tag{5.17}$$

$$\dot{\boldsymbol{v}} \triangleq \ddot{\boldsymbol{q}}_d - \Lambda\dot{\tilde{\boldsymbol{\xi}}}, \tag{5.18}$$

where $\ddot{\boldsymbol{q}}_d \in \Re^n$ is the original desired joint angular acceleration trajectory and $\Lambda \in \Re^{n\times n}$ is a diagonal positive definite gain matrix. The sliding mode error $\boldsymbol{r}$ is then given by

$$\boldsymbol{r} \triangleq \dot{\boldsymbol{q}} - \dot{\tilde{\boldsymbol{q}}}_{\boldsymbol{\tau net}} - \boldsymbol{v} = \dot{\tilde{\boldsymbol{\xi}}} + \Lambda\tilde{\boldsymbol{\xi}}. \tag{5.19}$$

Note that from (5.19) as $\boldsymbol{r} \to 0$, $\tilde{\boldsymbol{\xi}} \to 0$ and $\dot{\tilde{\boldsymbol{\xi}}} \to 0$. Similarly

$$\dot{\boldsymbol{r}} \triangleq \ddot{\boldsymbol{q}} - \ddot{\tilde{\boldsymbol{q}}}_{\boldsymbol{\tau net}} - \dot{\boldsymbol{v}} = \ddot{\tilde{\boldsymbol{\xi}}} + \Lambda\dot{\tilde{\boldsymbol{\xi}}}. \tag{5.20}$$

5.3.1 ARII Controller Theoretical Development

This section develops the relationships needed to ultimately prove that, using the proposed ARII control law, as $\boldsymbol{t} \to \infty$, $\tilde{\boldsymbol{\xi}}(t) \to 0$ as desired. Substituting the proposed control law given by (5.16) into the robotic exoskeleton equation of motion given by (5.2), and dropping the $(\boldsymbol{q})$ and $(\boldsymbol{q}, \dot{\boldsymbol{q}})$ notation for brevity gives the following equality

$$\begin{aligned} M\ddot{\boldsymbol{q}} + C\dot{\boldsymbol{q}} + \boldsymbol{\tau}_d \; &= \hat{M}_a\dot{\boldsymbol{v}} + \hat{C}_a\boldsymbol{v} + \hat{M}_a\ddot{\tilde{\boldsymbol{q}}}_{\tau net} + \hat{C}_a\dot{\tilde{\boldsymbol{q}}}_{\tau net} \\ &\quad - K_p\boldsymbol{r} - K_i\int \boldsymbol{r}dt - \boldsymbol{\tau}_r + \boldsymbol{\tau}_{p,meas} - \boldsymbol{\tau}_p. \end{aligned} \tag{5.21}$$

Observing (5.19) and (5.20), the following equalities are true

$$\dot{\boldsymbol{q}} = \boldsymbol{r} + \boldsymbol{v} + \dot{\tilde{\boldsymbol{q}}}_{\tau net}, \tag{5.22}$$

$$\ddot{\boldsymbol{q}} = \dot{\boldsymbol{r}} + \dot{\boldsymbol{v}} + \ddot{\tilde{\boldsymbol{q}}}_{\tau net}. \tag{5.23}$$

Substituting (5.22) and (5.23) into (5.21),

$$\begin{aligned} M(\dot{\boldsymbol{r}} + \dot{\boldsymbol{v}} + \ddot{\tilde{\boldsymbol{q}}}_{\tau net}) + C(\boldsymbol{r} + \boldsymbol{v} + \dot{\tilde{\boldsymbol{q}}}_{\tau net}) \; &= \hat{M}_a \dot{\boldsymbol{v}} + \hat{C}_a \boldsymbol{v} + \hat{M}_a \ddot{\tilde{\boldsymbol{q}}}_{\tau net} + \hat{C}_a \dot{\tilde{\boldsymbol{q}}}_{\tau net} \\ &\quad - K_p \boldsymbol{r} - K_i \int \boldsymbol{r} dt - \boldsymbol{\tau}_r - \boldsymbol{\tau}_d \\ &\quad + \boldsymbol{\tau}_{p,meas} - \boldsymbol{\tau}_p. \end{aligned} \tag{5.24}$$

Further grouping all of the M and C terms gives the following equation

$$\begin{aligned} M\dot{\boldsymbol{r}} + C\boldsymbol{r} + K_i \int \boldsymbol{r} dt \; &= \; (\hat{M}_a - M)(\dot{\boldsymbol{v}} + \ddot{\tilde{\boldsymbol{q}}}_{\boldsymbol{\tau net}}) + (\hat{C}_a - C)(\boldsymbol{v} + \dot{\tilde{\boldsymbol{q}}}_{\boldsymbol{\tau net}}) \\ &\quad - K_p \boldsymbol{r} - \boldsymbol{\tau}_r - \boldsymbol{\tau}_d + \boldsymbol{\tau}_{p,meas} - \boldsymbol{\tau}_p. \end{aligned} \tag{5.25}$$

Substituting (5.3) and (5.4) into (5.25) gives

$$\begin{aligned} M\dot{\boldsymbol{r}} + C\boldsymbol{r} + K_i \int \boldsymbol{r} dt \; &= \; (\hat{M}_a - M_a - E_M)(\dot{\boldsymbol{v}} + \ddot{\tilde{\boldsymbol{q}}}_{\boldsymbol{\tau net}}) \\ &\quad + (\hat{C}_a - C_a - E_C)(\boldsymbol{v} + \dot{\tilde{\boldsymbol{q}}}_{\boldsymbol{\tau net}}) \\ &\quad - K_p \boldsymbol{r} - \boldsymbol{\tau}_r - \boldsymbol{\tau}_d + \boldsymbol{\tau}_{p,meas} - \boldsymbol{\tau}_p. \end{aligned} \tag{5.26}$$

Letting $\tilde{M} = \hat{M}_a - M_a$ and $\tilde{C} = \hat{C}_a - C_a$ and substituting $\tilde{M}$ and $\tilde{C}$ into (5.26) yields

$$\begin{aligned} M\dot{\boldsymbol{r}} + C\boldsymbol{r} + K_i \int \boldsymbol{r} dt \; &= \; \tilde{M}(\dot{\boldsymbol{v}} + \ddot{\tilde{\boldsymbol{q}}}_{\boldsymbol{\tau net}}) + \tilde{C}(\boldsymbol{v} + \dot{\tilde{\boldsymbol{q}}}_{\boldsymbol{\tau net}}) \\ &\quad - E_M(\dot{\boldsymbol{v}} + \ddot{\tilde{\boldsymbol{q}}}_{\boldsymbol{\tau net}}) - E_C(\boldsymbol{v} + \dot{\tilde{\boldsymbol{q}}}_{\boldsymbol{\tau net}}) \\ &\quad - K_p \boldsymbol{r} - \boldsymbol{\tau}_r - \boldsymbol{\tau}_d + \boldsymbol{\tau}_{p,meas} - \boldsymbol{\tau}_p \\ &= \; \tilde{M}(\dot{\boldsymbol{v}} + \ddot{\tilde{\boldsymbol{q}}}_{\boldsymbol{\tau net}}) + \tilde{C}(\boldsymbol{v} + \dot{\tilde{\boldsymbol{q}}}_{\boldsymbol{\tau net}}) \\ &\quad - K_p \boldsymbol{r} + E + E_d + E_p - \boldsymbol{\tau}_r, \end{aligned} \tag{5.27}$$

where

$$\begin{aligned} E \; &\triangleq \; - E_M(\dot{\boldsymbol{v}} + \ddot{\tilde{\boldsymbol{q}}}_{\boldsymbol{\tau net}}) - E_C(\boldsymbol{v} + \dot{\tilde{\boldsymbol{q}}}_{\boldsymbol{\tau net}}), \\ E_d \; &\triangleq \; - \boldsymbol{\tau}_d, \\ E_p \; &\triangleq \; \boldsymbol{\tau}_{p,meas} - \boldsymbol{\tau}_p. \end{aligned}$$

Provided that we have detailed knowledge of the underlying dynamic model of the robotic exoskeleton, then we can express $\tilde{M}(\dot{\boldsymbol{v}} + \ddot{\tilde{\boldsymbol{q}}}_{\boldsymbol{\tau net}}) + \tilde{C}(\boldsymbol{v} + \dot{\tilde{\boldsymbol{q}}}_{\boldsymbol{\tau net}})$ so that it is linear in the unknown dynamic model parameters as

$$\tilde{M}(\dot{\boldsymbol{v}} + \ddot{\tilde{\boldsymbol{q}}}_{\boldsymbol{\tau net}}) + \tilde{C}(\boldsymbol{v} + \dot{\tilde{\boldsymbol{q}}}_{\boldsymbol{\tau net}}) = Y(\boldsymbol{q}, \dot{\boldsymbol{q}}, \boldsymbol{v}, \dot{\boldsymbol{v}}, \dot{\tilde{\boldsymbol{q}}}_{\tau net}, \ddot{\tilde{\boldsymbol{q}}}_{\tau net})\tilde{\boldsymbol{p}}, \tag{5.28}$$

where

$$\tilde{\boldsymbol{p}} = \hat{\boldsymbol{p}} - \boldsymbol{p_a}, \tag{5.29}$$

such that $\boldsymbol{p} \in \Re^m$ is the vector of m model parameters, $\hat{\boldsymbol{p}} \in \Re^m$ is the vector of m adaptive control law estimated parameters, and $Y \in \Re^{n \times m}$ is the dynamic regressor matrix which describes how the known and calculated motion vectors $\boldsymbol{q}, \dot{\boldsymbol{q}}, \boldsymbol{v}, \dot{\boldsymbol{v}}, \dot{\tilde{\boldsymbol{q}}}_{\tau net}$, and $\ddot{\tilde{\boldsymbol{q}}}_{\tau net}$ are related to the unknown model parameters. Let these unknown dynamic model parameters $\boldsymbol{p}_a = [a_1, a_2, a_3]^T$ be as in [61]

$$\begin{aligned} a_1 &= I_1 + I_2 + m_1 l_{c1}^2 + m_2 l_1^2 + m_2 l_{c2}^2, \\ a_2 &= m_2 l_1 l_{c2}, \\ a_3 &= I_2 + m_2 l_{c2}^2, \end{aligned}$$

where l_i is the length of link i, l_{ci} is the distance along link i to its center of mass, m_i is the mass of link i and, I_i is the moment of inertia of link i about its center of mass. Then we can use our detailed knowledge of the equation of motion of the robotic exoskeleton given by (3.6) to write the $\hat{M}$ and $\hat{C}$ matrices in terms of the unknown dynamic parameters a_1, a_2, and a_3 as follows

$$\hat{M} = \begin{bmatrix} a_1 + 2a_2 cos(q_2) & a_3 + a_2 cos(q_2) \\ a_3 + a_2 cos(q_2) & a_3 \end{bmatrix}, \tag{5.30}$$

and

$$\hat{C} = \begin{bmatrix} -a_2 sin(q_2)\dot{q}_2 & -a_2 sin(q_2)(\dot{q}_1 + \dot{q}_2) \\ a_2 sin(q_2)\dot{q}_1 & 0 \end{bmatrix}. \tag{5.31}$$

Substituting these expressions for $\hat{M}$ and $\hat{C}$ into (5.28) along with the unknown dynamic parameters $\boldsymbol{p}_a = [a_1, a_2, a_3]^T$ yields the following expression for the dynamic regressor matrix $Y(\boldsymbol{q}, \dot{\boldsymbol{q}}, \boldsymbol{v}, \dot{\boldsymbol{v}}, \dot{\tilde{\boldsymbol{q}}}_{\tau net}, \ddot{\tilde{\boldsymbol{q}}}_{\tau net})$ in (5.28),

$$Y = \begin{bmatrix} Y_{11} & Y_{12} & Y_{13} \\ Y_{21} & Y_{22} & Y_{23} \end{bmatrix}, \tag{5.32}$$

where

$$
\begin{aligned}
Y_{11} &= \dot{v}_1 + \ddot{\tilde{q}}_{\tau net,1}, \\
Y_{12} &= (2(\dot{v}_1 + \ddot{\tilde{q}}_{\tau net,1}) + \dot{v}_2 + \ddot{\tilde{q}}_{\tau net,2})\cos(q_2) \\
&\quad + (\dot{q}_1\dot{\tilde{q}}_{\tau net,2} - \dot{q}_2\dot{\tilde{q}}_{\tau net,1} - \dot{q}_2\dot{\tilde{q}}_{\tau net,2} \\
&\quad - \dot{q}_1 v_2 - \dot{q}_2 v_1 - \dot{q}_2 v_2)\sin(q_2), \\
Y_{13} &= \dot{v}_2 + \ddot{\tilde{q}}_{\tau net,2}, \\
Y_{21} &= 0, \\
Y_{22} &= (\dot{v}_1 + \ddot{\tilde{q}}_{\tau net,1})\cos(q_2) + \dot{q}_1(v_1 + \dot{\tilde{q}}_{\tau net,1})\sin(q_2), \\
Y_{23} &= \dot{v}_1 + \ddot{\tilde{q}}_{\tau net,1} + \dot{v}_2 + \ddot{\tilde{q}}_{\tau net,2}.
\end{aligned}
$$

Substituting (5.28) in (5.27) gives

$$
M\dot{\boldsymbol{r}} + C\boldsymbol{r} + K_i \int \boldsymbol{r}dt = Y\tilde{\boldsymbol{p}} - K_p\boldsymbol{r} + E + E_d + E_p - \boldsymbol{\tau}_r. \tag{5.33}
$$

For reasons which will become clear in the next section, let the robust term $\boldsymbol{\tau}_r$ be defined as

$$
\boldsymbol{\tau_r} \triangleq (K + K_d + K_{\tau p})sgn(\boldsymbol{r}), \tag{5.34}
$$

where $K{=}diag(k_{ii})$, $k_{ii} \geq |E_i|$, $i = 1 \ldots n$, $K_d{=}diag(k_{d,ii})$, $k_{d,ii} \geq |E_d|$, and $K_{\tau p} = diag(k_{\tau p,ii})$, $k_{\tau p,ii} \geq |E_p|$. In other words, the gains K, K_d, and $K_{\tau p}$ include the upper bounds of E_i, E_d, and E_p respectively. Then

$$
M\dot{\boldsymbol{r}} + C\boldsymbol{r} + K_i \int \boldsymbol{r}dt = Y\tilde{\boldsymbol{p}} - K_p\boldsymbol{r} + E + E_d + E_p - (K + K_d + K_{\tau p})sgn(\boldsymbol{r}). \tag{5.35}
$$

To avoid unwanted chatter associated with the $sgn(r)$ terms, the following smoothing function was utilized instead: $\frac{r}{(|r|+\epsilon)}$. The τ_r terms were optimized through manual systematic selection for all the tests in this research. Alternatively, optimization algorithms could have been implemented.

5.3.2 Stability Proof

To prove the stability of the proposed ARII control law, consider the Lyapunov candidate function

$$
V = \frac{1}{2}\boldsymbol{r}^T M\boldsymbol{r} + \frac{1}{2}(\int \boldsymbol{r}dt)^T K_i(\int \boldsymbol{r}dt) + \frac{1}{2}\tilde{\boldsymbol{p}}^T\Gamma\tilde{\boldsymbol{p}}, \tag{5.36}
$$

where $\Gamma \in \Re^{m\times m}$ is a symmetric positive definite matrix. Taking the time derivative of V yields

$$\begin{aligned}\dot{V} =& \frac{1}{2}\dot{\boldsymbol{r}}^T M\boldsymbol{r} + \frac{1}{2}\boldsymbol{r}^T\dot{M}\boldsymbol{r} + \frac{1}{2}\boldsymbol{r}^T M\dot{\boldsymbol{r}} + \frac{1}{2}\boldsymbol{r}^T K_i(\int \boldsymbol{r}dt) \\ &+ \frac{1}{2}(\int \boldsymbol{r}dt)^T\dot{K}_i(\int \boldsymbol{r}dt) + \frac{1}{2}(\int \boldsymbol{r}dt)^T K_i\boldsymbol{r} \\ &+ \frac{1}{2}\dot{\tilde{\boldsymbol{p}}}^T\Gamma\tilde{\boldsymbol{p}} + \frac{1}{2}\tilde{\boldsymbol{p}}^T\dot{\Gamma}\tilde{\boldsymbol{p}} + \frac{1}{2}\tilde{\boldsymbol{p}}^T\dot{\tilde{\boldsymbol{p}}}.\end{aligned} \tag{5.37}$$

Recognizing that $\dot{K}_i$ and $\dot{\Gamma}$ are zero and further reducing the terms results in

$$\dot{V} = \dot{\boldsymbol{r}}^T M\boldsymbol{r} + \frac{1}{2}\boldsymbol{r}^T\dot{M}\boldsymbol{r} + \boldsymbol{r}^T K_i(\int \boldsymbol{r}dt) + \tilde{\boldsymbol{p}}^T\Gamma\dot{\tilde{\boldsymbol{p}}}. \tag{5.38}$$

Note that the skew-symmetric property of manipulator dynamics (Property 2 in §4.3.1) gives

$$\frac{1}{2}\boldsymbol{r}^T(\dot{M} - 2C)\boldsymbol{r} = 0. \tag{5.39}$$

Substituting (5.39) into (5.38) results in

$$\begin{aligned}\dot{V} &= \boldsymbol{r}^T M\dot{\boldsymbol{r}} + \boldsymbol{r}^T C\boldsymbol{r} + \boldsymbol{r}^T K_i(\int \boldsymbol{r}dt) + \tilde{\boldsymbol{p}}^T\Gamma\dot{\tilde{\boldsymbol{p}}} \\ &= \boldsymbol{r}^T(M\dot{\boldsymbol{r}} + C\boldsymbol{r} + K_i\int \boldsymbol{r}dt) + \tilde{\boldsymbol{p}}^T\Gamma\dot{\tilde{\boldsymbol{p}}}.\end{aligned} \tag{5.40}$$

Substituting (5.35) into (5.40) yields

$$\begin{aligned}\dot{V} &= \boldsymbol{r}^T(Y\tilde{\boldsymbol{p}} - K_p\boldsymbol{r} + E + E_d + E_p - (K + K_d + K_{\tau p})sgn(\boldsymbol{r})) + \tilde{\boldsymbol{p}}^T\Gamma\dot{\tilde{\boldsymbol{p}}} \\ &= -\boldsymbol{r}^T K_p\boldsymbol{r} - \boldsymbol{r}^T(K + K_d + K_{\tau p})sgn(\boldsymbol{r}) \\ &\quad + \boldsymbol{r}^T E + \boldsymbol{r}^T E_d + \boldsymbol{r}^T E_p + \boldsymbol{r}^T Y\tilde{\boldsymbol{p}} + \tilde{\boldsymbol{p}}^T\Gamma\dot{\tilde{\boldsymbol{p}}}.\end{aligned} \tag{5.41}$$

Note that

$$\boldsymbol{r}^T Y\tilde{\boldsymbol{p}} + \tilde{\boldsymbol{p}}^T\Gamma\dot{\tilde{\boldsymbol{p}}} = \tilde{\boldsymbol{p}}^T Y^T\boldsymbol{r} + \tilde{\boldsymbol{p}}^T\Gamma\dot{\tilde{\boldsymbol{p}}} = \tilde{\boldsymbol{p}}^T(Y^T\boldsymbol{r} + \Gamma\dot{\tilde{\boldsymbol{p}}}), \tag{5.42}$$

since $\boldsymbol{r}^T Y\tilde{\boldsymbol{p}}$ is a scalar. Taking the derivative of (5.29) yields

$$\dot{\tilde{\boldsymbol{p}}} = \dot{\hat{\boldsymbol{p}}} - \dot{\boldsymbol{p}}_{\boldsymbol{a}}, \tag{5.43}$$

and noting that $\dot{\boldsymbol{p}}_{\boldsymbol{a}}$=0 since $\boldsymbol{p}_{\boldsymbol{a}}$ is constant, then the **adaptive law** was designed as

$$\dot{\hat{\boldsymbol{p}}} = -\Gamma^{-1}Y^T\boldsymbol{r}, \tag{5.44}$$

where Γ can be used to control the speed of the adaptation of the parameters. Therefore (5.41) becomes

$$\begin{aligned}\dot{V} =& -\boldsymbol{r}^T K_p \boldsymbol{r} + (-\boldsymbol{r}^T K sgn(\boldsymbol{r}) + \boldsymbol{r}^T E) + (-\boldsymbol{r}^T K_d sgn(\boldsymbol{r}) + \boldsymbol{r}^T E_d) \\ &+ (-\boldsymbol{r}^T K_{\tau p}) sgn(\boldsymbol{r}) + \boldsymbol{r}^T E_p) \\ =& -\boldsymbol{r}^T K_p \boldsymbol{r} + (-\sum_{k=1}^{n} K_k |r_k| + \sum_{k=1}^{n} E_k r_k) \\ &+ (-\sum_{k=1}^{n} K_{d,k} |r_k| + \sum_{k=1}^{n} E_{d,k} r_k) \\ &+ (-\sum_{k=1}^{n} K_{\tau p,k} |r_k| + \sum_{k=1}^{n} E_{p,k} r_k). \end{aligned} \tag{5.45}$$

Since $-\boldsymbol{r}^T K_p \boldsymbol{r} \leq 0$ and

$$\begin{aligned}\sum_{k=1}^{n} K_k |r_k| &\geq \sum_{k=1}^{n} E_k r_k, \\ \sum_{k=1}^{n} K_{d,k} |r_k| &\geq \sum_{k=1}^{n} E_{d,k} r_k, \\ \sum_{k=1}^{n} K_{\tau p,k} |r_k| &\geq \sum_{k=1}^{n} E_{p,k} r_k.\end{aligned}$$

then

$$\dot{V} \leq 0. \tag{5.46}$$

This condition implies stability which means that the system will stay in the vicinity of equilibrium. This condition on its own, however, does not imply asymptotic stability, where the system returns to equilibrium. Since V is in a quadratic form (meaning its terms are of the form $\frac{1}{2}\boldsymbol{r}^T M \boldsymbol{r}$), then V is positive definite since M, K_i and Γ are positive definite. Also V is non-increasing since $\dot{V} \leq 0$. Therefore $\boldsymbol{r}$(t) is bounded ($\boldsymbol{r} \in \mathcal{L}_\infty^n$) and $\boldsymbol{r}$(t) is a square integrable function ($\boldsymbol{r} \in \mathcal{L}_2^n$) satisfying

$$\sqrt{\int_t ||r(t)||^2 dt} \ < \ \infty. \tag{5.47}$$

Since $\boldsymbol{r} \in \mathcal{L}_2 \cap \mathcal{L}_\infty$ and $\boldsymbol{r} = \dot{\tilde{\boldsymbol{\xi}}} + \Lambda \tilde{\boldsymbol{\xi}}$ from (5.19), then $\tilde{\boldsymbol{\xi}}$ and $\dot{\tilde{\boldsymbol{\xi}}} \in \mathcal{L}_2^n \cap \mathcal{L}_\infty^n$. According to Barbălat's Lemma [73]

$$\begin{aligned}\text{if} \quad \tilde{\boldsymbol{\xi}} \in \mathcal{L}_2^n \cap \mathcal{L}_\infty^n \quad \text{and} \quad \dot{\tilde{\boldsymbol{\xi}}} \in \mathcal{L}^n \\ \text{then} \quad \lim_{t\to\infty} \tilde{\boldsymbol{\xi}}(t) = 0,\end{aligned} \tag{5.48}$$

as desired, which completes the proof.

5.4 ARII Method Validation with the Single-System 1-DOF Quanser QUBE Robot Testbed

The proposed ARII control method was implemented in MATLAB/Simulink and was first validated using the Quanser QUBE simulation testbed using the identified model of the plant from the previous chapter. The control method was then validated experimentally using the Quanser QUBE experimental testbed.

5.4.1 Simulations Validation with the Quanser QUBE Robot

To validate the proposed ARII control method, a series of motion trajectories were commanded. Initially, harmonic motion, which mimics passively moving the patient's arm as done in physiotherapy exercises, was implemented as shown in Fig. 5.4(a) with the following desired trajectory (red line)

$$q_d = A\sin(wt - \pi/2) + 45\pi/180, \tag{5.49}$$

where the amplitude A and frequency w were set to $\pi/4$ rad and $\pi/4$ rad/s, respectively. The initial patient resting position q_o was set to 0°, which is to demonstrate a stroke patient with high arm muscle tone and having the arm folded towards the chest wall [97]. This oscillatory motion, shown in region A in Fig. 5.4, continues for 48 seconds during which time the adaptive control law is allowed to adjust the model parameters. Referring to Fig. 5.4(a), after 48 seconds the robot was commanded to hold a position of 0° for 4 seconds (region B) and then progressively increase the position by 22.5° (region C) and hold stationary at 90° until the end of the test at 150 seconds (region D). For convenience, Table 5.1 summarizes the regions used throughout the single-system tests with the Quanser QUBE testbeds. Furthermore, Table 5.2 summarizes the ARII Quanser QUBE simulation parameters. The impedance parameters were selected so that the maximum torque applied to the patient by the robot was effectively limited to 1 Nm as shown in Fig. 5.1. The patient stiffness parameter was scaled from [168] so that the patient model could be used with the Quanser QUBE. To select the controller parameters shown in Table 5.2 a systematic and extensive manual tuning process was carried out to produce desirable and converging tracking

errors. Saturation blocks on the control signal were applied to match the motor limits of the Quanser QUBE, and to enhance system stability by implementing the time domain passivity based strategy of boundedness.

Table 5.1: Simulations and Experiments Regions for Single-System Quanser QUBE

Region	**Time Range**	**Operation**
A	0-48 s	oscillate between 0° and 90°, adaptation period
B	48-60 s	hold still at 0°
C	60-94 s	progressively increase the position by 22.5° up to 90°
D	94-150 s	hold still at 90°

Fig. 5.4 shows the results of utilizing the ARII controller with the Quanser QUBE simulator testbed. Fig. 5.4(a) plots the desired joint angular position, the impedance desired joint angular position, and the actual joint angular position as a function of time. The effects of using impedance control are apparent since, rather than following the robot desired angular position trajectory that has a maximum value of 90° (red), in response to the applied patient's torque shown in Fig. 5.4(c) the robot (magenta) follows the impedance designed desired trajectory (green) which has a maximum value of 68.2°. Fig. 5.4(b) shows the corresponding impedance joint angular position tracking error $\tilde{\xi}$ plotted as a function of time which decreases with time as the adaptive control law adjusts the dynamic model parameters. Fig. 5.5 plots the adaptation of the dynamic model parameter terms for the Quanser QUBE ARII simulation as a function of time which are convergent and held constant after the adaptation period of 48 seconds. The performance metrics for this benchmark simulation validation test are listed in Table 5.3. The maximum $\tilde{\xi}$ was recorded as 2.4° while the $\tilde{\xi}_{RMSE}$ had a value of 0.7°.

Table 5.2: ARII QUBE Simulation Parameters

Impedance:	J	0.013 Kgm^2
	B	0.667 Nms
	K	2.667 Nm
Patient:	K_p	0.848 Nm/rad
	D_p	0.2 Nms/rad
Controller:	Γ	diag([2,2])
	Λ	4.0
	K_d	1.0
	K_i	0.1
	K_r	0.01
	ϵ	0.08

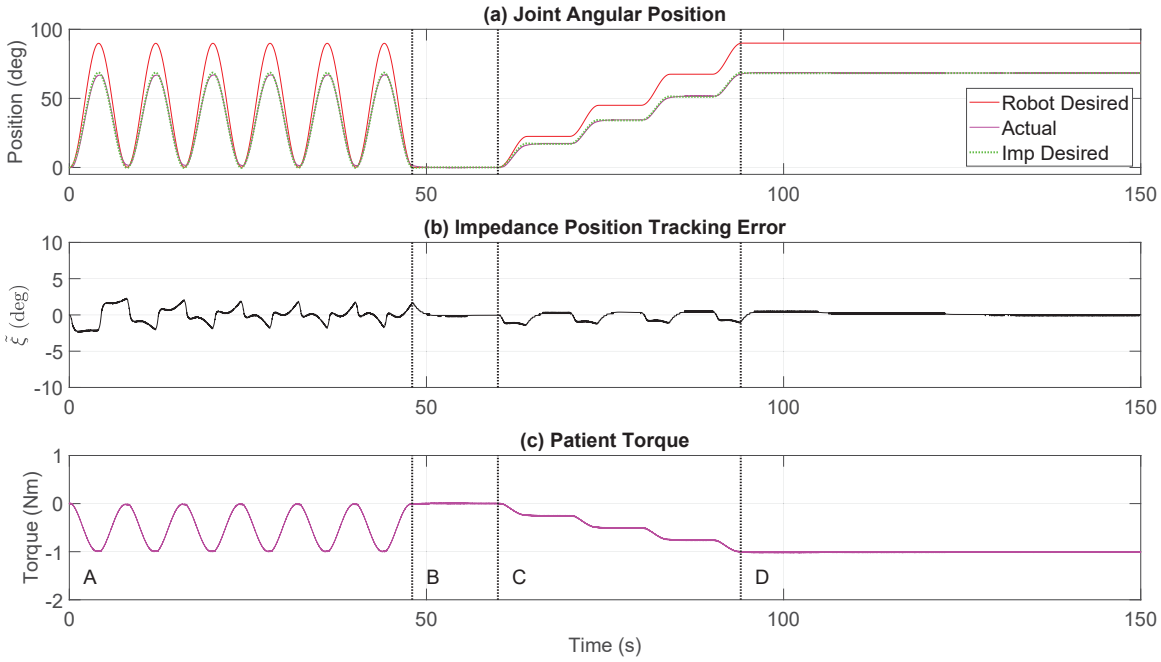


Figure 5.4: ARII single-system simulation with Quanser QUBE: (a) position trajectory, (b) tracking error, and (c) torque

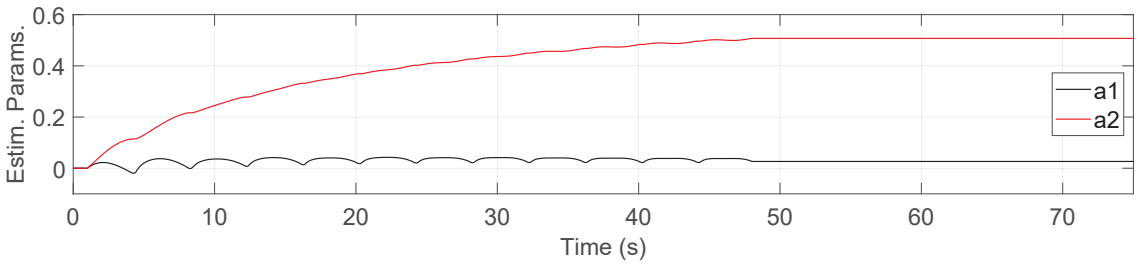


Figure 5.5: Estimated parameters using the ARII control method with the Quanser QUBE single-system simulator

Table 5.3: Performance Metrics for ARII Single-System Simulation with Quanser QUBE

Testbed	**Metrics**	**ARII**
QUBE simulation	$\tilde{\xi}_{max}$	2.4°
	$\tilde{\xi}_{RMSE}$	0.7°

5.4.2 Experimental Validation with the Quanser QUBE Robot

Having successfully implemented the proposed ARII control approach on the Quanser QUBE simulator testbed, the same desired trajectory and controller parameters were used for experimental tests with the Quanser QUBE apparatus. Fig. 5.6 shows the resulting experimental results using the ARII controller with the Quanser QUBE hardware. Similar to Fig. 5.4, Fig. 5.6(a) plots the desired joint angular position, the impedance desired joint angular position, and the actual joint angular position, as functions of time. Fig. 5.6(b) shows the impedance joint angular position tracking error $\tilde{\xi}$, while Fig. 5.6(c) shows the patient torques applied during the test. Fig. 5.7 show the adaptation terms for the Quanser QUBE ARII experimental test which, after the 48 seconds adaptation period, were set to their convergent values for the remainder of the test. The performance metrics for this experiment are listed in Table 5.4. The maximum $\tilde{\xi}$ was recorded as 2.8°, and the $\tilde{\xi}_{RMSE}$ had a value of 1.0°.

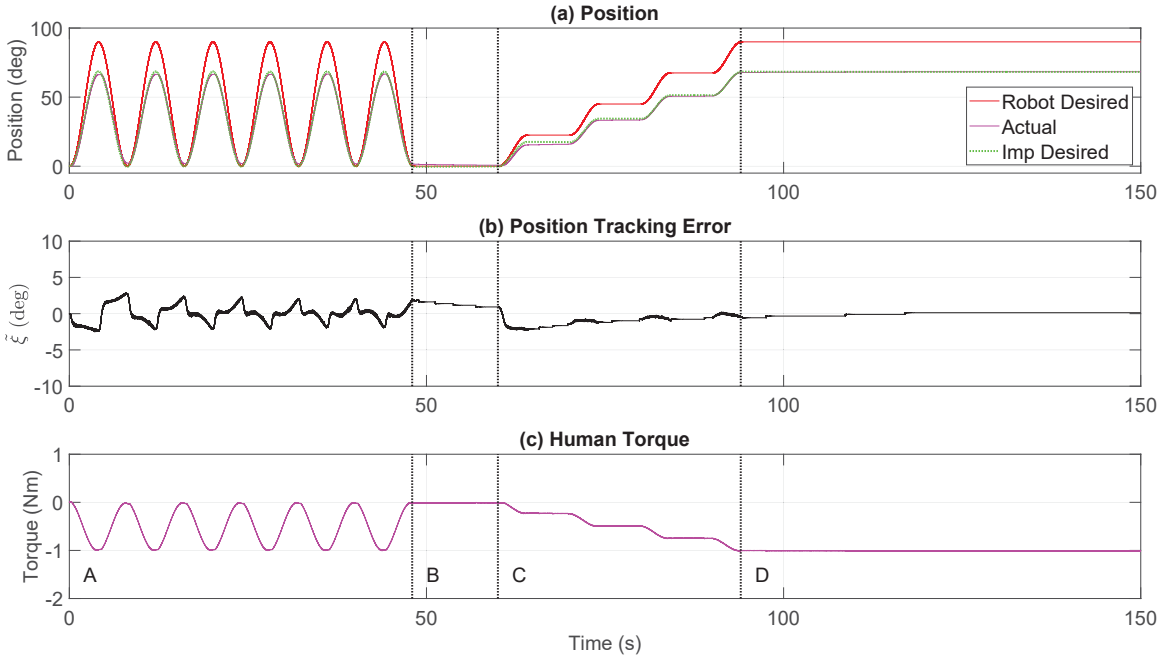


Figure 5.6: ARII single-system experiment with Quanser QUBE: (a) position trajectory, (b) tracking error, and (c) torque

The experimental impedance tracking error $\tilde{\xi}$ results were superimposed on the

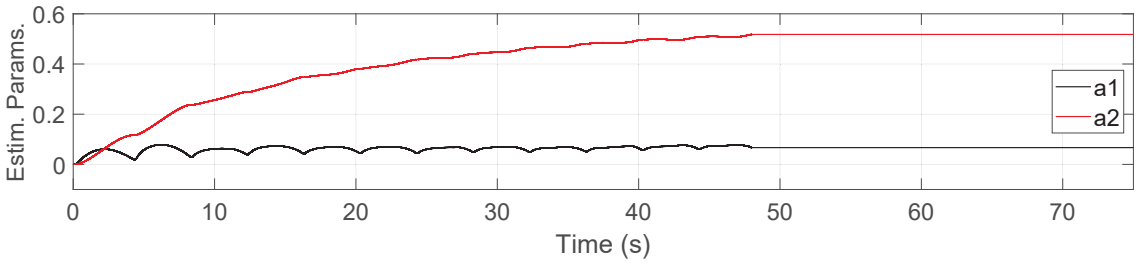


Figure 5.7: Estimated parameters using the ARII control method with the Quanser QUBE single-system apparatus

Table 5.4: Performance Metrics for ARII Single-System Experiment with Quanser QUBE

Testbed	**Metrics**	**ARII**
QUBE experimental setup	$\tilde{\xi}_{max}$	2.8°
	$\tilde{\xi}_{RMSE}$	1.0°

simulation $\tilde{\xi}$ results and shown in Fig. 5.8. Furthermore the performance metrics for the simulated and the experimental tests are listed side-by-side in Table 5.5, for comparison purposes. As can be seen in Fig. 5.8 and Table 5.5, the experimental results are remarkably similar to the simulator results in terms of bound (with only 0.2° difference in the initial phase) and as well as in the oscillatory motion region and, therefore, effectively validate the simulation results.

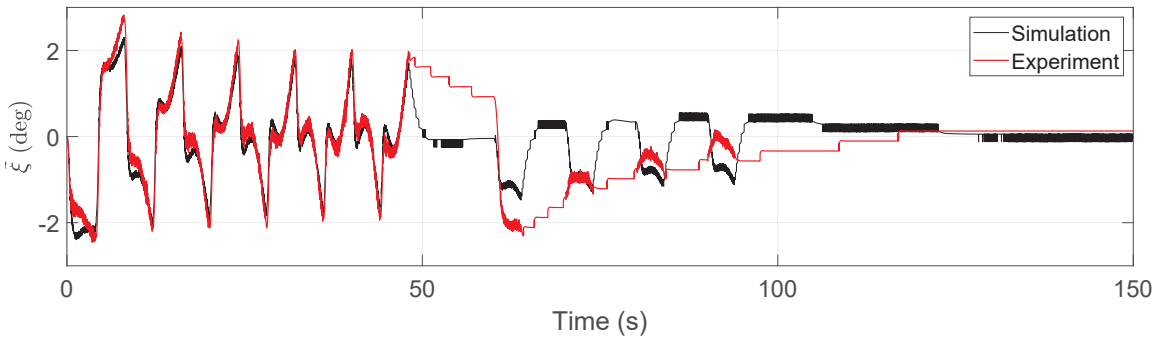


Figure 5.8: The profile of $\tilde{\xi}$ for ARII Quanser QUBE simulation and experiment

Table 5.5: Performance Metrics for ARII Single-System Simulation versus Experiment with Quanser QUBE

Metrics	**Simulation**	**Experiment**
$\tilde{\xi}_{max}$	2.4°	2.8°
$\tilde{\xi}_{RMSE}$	0.7°	1.0°

5.5 ARII Method Validation with the Single-System 2-DOF Robotic Exoskeleton Simulations

Having validated the simulation results experimentally on a 1-DOF robotic system, the simulator was extended to a 2-DOF robotic exoskeleton involving both elbow and wrist joints to validate the proposed ARII control methodology on a higher-degree-of-freedom system. The desired trajectory for the exoskeleton elbow joint was the same as that used in the 1-DOF simulation and experiment

$$q_{d1} = A\sin(wt - \pi/2) + 45\pi/180, \tag{5.50}$$

where the amplitude A and frequency w were set to $\pi/4$ rad and $\pi/4$ rad/s, respectively, so that the trajectory oscillates between between 0° and 90°. The desired trajectory for the exoskeleton's wrist joint was characterized by

$$q_{d2} = A\sin(wt - \pi/2), \tag{5.51}$$

where the amplitude A and frequency w were set to $\pi/4$ rad and $\pi/4$ rad/s, respectively, so that the trajectory oscillates between between $-45°$ and 45° at the same frequency as joint 1. The initial patient resting position for joint 1 (elbow) was set to 0°, while the initial patient resting position for joint 2 (wrist) was set to $-45°$ which is demonstrative of a stroke patient with high arm muscle tone and having the arm and wrist folded towards the chest wall [97]. For joint 1, this oscillatory motion, shown in region A in Fig. 5.9, continues for 48 seconds during which time the adaptive control law is allowed to adjust the model parameters. Fig. 5.9(a) shows how after 48 seconds the robot joint 1 was then commanded to hold a position of 0° for 4 seconds (region B) and then progressively increase the position by 22.5° (region C) and hold stationary at 90° until the end of the test at 150 seconds (region D). For joint 2,

Table 5.6: Regions Definition for Single-System 2-DOF Exoskeleton Simulations

Region	**Time Range**	**Operation**
Joint 1		
A	0-48 s	oscillate between 0° and 90°, adaptation period
B	48-60 s	hold still at 0°
C	60-94 s	progressively increase the position by 22.5° up to 90°
D	94-150 s	hold still at 90°
Joint 2		
A	0-48 s	oscillate between -45° and 45°, adaptation period
B	48-120 s	hold still at -45°
C	120-124 s	go to 45°
D	124-150 s	hold still at 45°

Fig. 5.10(a) shows how after 48 seconds of oscillatory motion the robot joint 2 was commanded to hold a position of $-45°$ (region B) and then at 120 seconds (region C) increase the position to 45° and hold at 45° from 124 seconds until the end of the test at 150 seconds (region D). For convenience, Table 5.6 summarizes the regions used throughout the single-system tests with the 2-DOF exoskeleton testbed.

Furthermore, Table 5.7 summarizes the ARII 2-DOF Exoskeleton simulation parameters. Note that the notation diag($[a, b]$) refers to a diagonal matrix with diagonal elements a and b corresponding to joint 1 and joint 2, respectively. The impedance parameters were set as shown in Table 5.7 to adjust the patient's motion so that the maximum torque applied to the patient by the robotic exoskeleton was effectively limited to 1 Nm for the elbow joint (similar to the Quanser QUBE simulations and experiments) and 0.6 Nm for the wrist joint without therapist assistance. The impedance, therefore, ensures the patient's safety while working with the robot. The patient stiffness parameter was scaled from [168] so that the patient model could be used with the simulated 2-DOF robotic exoskeleton. To select the controller parameters shown in Table 5.7 a systematic and extensive manual tuning process was carried out similar to the Quanser QUBE simulations and experiments to produce desirable and converging tracking errors. Saturation blocks on the control signal were applied to match the motor limits of the conceptualized 2-DOF exoskeleton system, and to enhance system stability by implementing the time domain passivity based strategy

of boundedness.

Table 5.7: ARII Single-System Simulation with 2-DOF Exoskeleton: Simulation Parameters

Impedance:	J	diag([0.013, 0.013]) Kgm^2
	B	diag([0.667, 0.667]) Nms
	K	diag([2.667, 2.667]) Nm
Patient:	K_p	diag([.848,.424])
	D_p	diag([.2,.05])
pHMI:	K_{pHMI}	diag([800, 266.667])N/rad
	D_{pHMI}	diag([10, 2.5])Ns/rad
Controller:	Γ	diag([.02,1,1])
	Λ	diag([1,4])
	K_d	diag([8,2])
	K_i	diag([.1,.1])
	K_r	diag([.01,.01])
	ϵ	$[0.08,0.08]^T$

Figs. 5.9 to 5.11 show the resulting performance of the ARII controller. Figs. 5.9(a) and 5.10(a) show the desired joint angular position, the impedance desired joint angular position, and the actual joint angular position of joint 1 and joint 2, respectively, plotted as a function of time. Note how, similar to the Quanser QUBE results, the patient's motion is adjusted by the impedance controller in response to the patient's elbow input torque shown in Fig. 5.9(c) to limit motion to between 0° and 68.5° when the exoskeleton is commanded to move between 0° to 90° for joint 1. Similarly, the motion is attenuated to a range between $-45°$ and 33.1° instead of from $-45°$ to 45° for joint 2 given the patient's wrist input torque shown in Fig. 5.10(c). Plots (b) on both Figs. 5.9 and 5.10 show the resulting impedance joint angular position tracking error $\tilde{\xi}$ for joints 1 and 2. Fig. 5.11 shows the adaptation of the dynamic parameter terms, and how, with the selected adaptive law Γ parameters, the adaptive terms quickly move towards and oscillate close to their steady state values within the first six seconds. The corresponding performance metrics for this test are listed in Table 5.8. For joint 1, the maximum $\tilde{\xi}$ was recorded as 0.5°, and the $\tilde{\xi}_{RMSE}$ had a value of 0.1°. For joint 2, the maximum $\tilde{\xi}$ was recorded as 0.1°, while the $\tilde{\xi}_{RMSE}$ had a value of 0.0°.

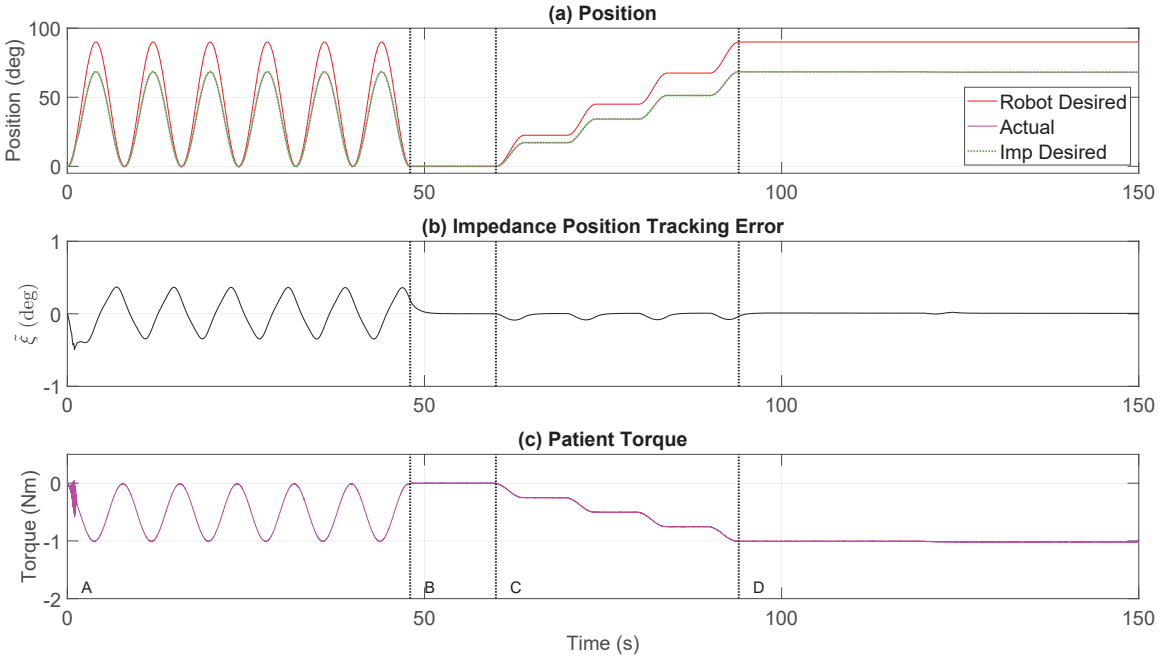


Figure 5.9: ARII single-system simulation with 2-DOF exoskeleton: (a) position trajectory, (b) tracking error, and (c) patient torque for joint 1

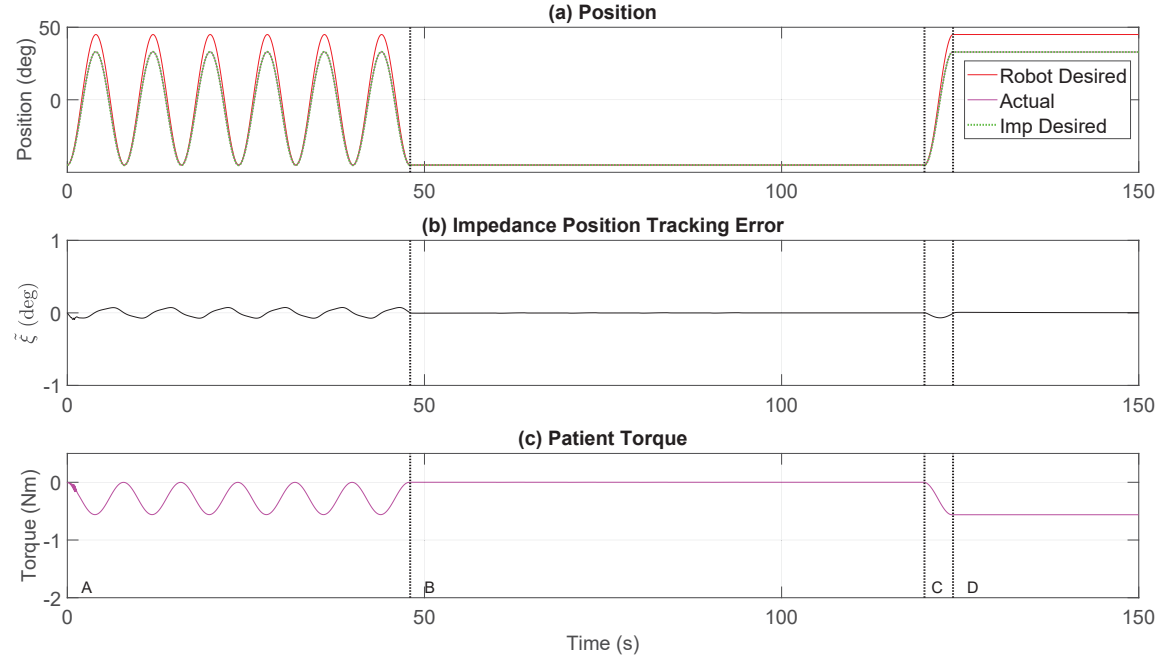


Figure 5.10: ARII single-system simulation with 2-DOF exoskeleton: (a) position trajectory, (b) tracking error, and (c) patient torque for joint 2

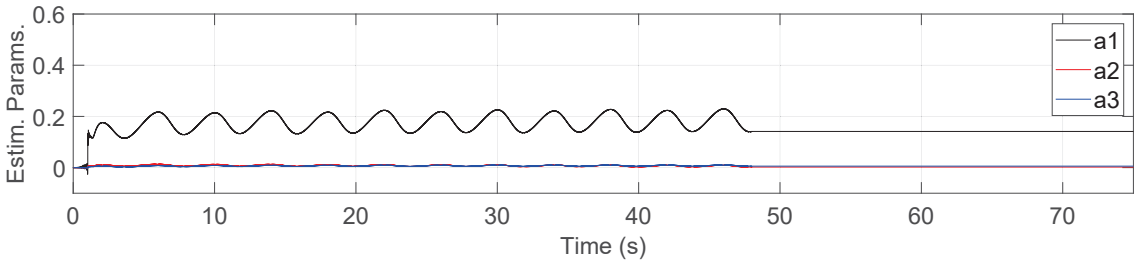


Figure 5.11: Estimated parameters using the ARII control method with the 2-DOF exoskeleton simulator

Table 5.8: Performance Metrics for ARII Single-System Simulation with the 2-DOF Exoskeleton

Testbed	**Metrics**	**ARII**
2-DOF Exo Sim (Joint 1)	$\tilde{\xi}_{max}$	0.5°
	$\tilde{\xi}_{RMSE}$	0.1°
(Joint 2)	$\tilde{\xi}_{max}$	0.1°
	$\tilde{\xi}_{RMSE}$	0.0°

5.6 Telerehabilitation Human Torque Synthesis Experiments with the Quanser QUBEs Testbed

Having validated the proposed ARII control approach on single systems in both simulation and experiment, experimental tests were carried out with two Quanser QUBE experimental testbeds to validate the performance of the ARII method in telerehabilitation scenarios and to examine the tracking performance and telepresence fidelity. The Logitech driving wheel hardware was used to emulate human real-time assistive intervention from the therapist on the slave side. As shown in Table 5.9 and Fig. 5.12, the first 100 seconds of the tests (corresponding to regions A-D in the figure and explained in the previous sections) are unilateral, where the robot follows the same oscillation patterns and incremental step trajectories as carried out in the previous single-system tests. Then, after the 100 seconds mark, the human therapist is allowed to assist the patient-master robotic exoskeleton system to move towards the desired steady-state trajectory position of 90°.

The point at which the therapist begins applying assistive torques defines the start of region E as shown in Fig. 5.12. Bilateral telerehabilitation (referred to as region

Table 5.9: Regions Definition for Quanser QUBE Telerehabilitation Experiments

Region	**Time Range**	**Operation**
A	0-48 s	oscillate between 0° and 90°, adaptation period
B	48-60 s	hold still at 0°
C	60-94 s	increase position in increments of 22.5°, up to 90°
D	94-150 s	hold still at 90°
E	100-150 s	bilateral telerehabilitation is possible
F	start ◊ 100-150 s	therapist's torque assists patient

◊ = *between*

F) officially starts when the robot on the master side begins to move in response to the therapist's assistive torque $\boldsymbol{\tau}_{refl}$. This bilateral telerehabilitation portion of the test is shown as a blue region in all the bilateral position and torque plots in this research. Note that, for the scenario studied in Fig. 5.12, regions E and F overlap. This overlap occurs because for this scenario, as soon as the therapist begins to try to apply assisting torques, the robot on the master side begins to move in response to these torques. During bilateral teleoperation, the net torque $\boldsymbol{\tau}_{net}$ (shown in Fig. 4.1) on the master side is composed of torques from both the patient $\boldsymbol{\tau}_p$ and the reflected torque $\boldsymbol{\tau}_{refl}$ from the therapist on the slave side. The remainder of this chapter carries out telerehabilitation experiments to examine the effects of different strategies of combining the human torques to calculate $\boldsymbol{\tau}_{net}$ on the resulting trajectories and telepresence.

5.6.1 Direct $\boldsymbol{\tau}_{refl}$ Addition

A common method of combining the reflected assistive therapist torque to the system on the master side is to simply add it to the patient's torque [65, 60, 59] as follows

$$\boldsymbol{\tau}_{net} = \boldsymbol{\tau}_p + \alpha \boldsymbol{\tau}_{refl}, \tag{5.52}$$

where $\alpha \in \Re^{n \times n}$ is the identity matrix. Fig. 5.12(a) plots the corresponding experimental position signals as functions of time, while 5.12(b) shows the experimental impedance angular position tracking error ξ when τ_{net} is calculated using (5.52). Fig. 5.12(c) plots the patient torque τ_p in fuchsia and the therapist torque τ_{refl} in black. Fig. 5.12(c) also plots the negative value of the therapist torque $(-\tau_{refl})$ as a

dashed black line for comparison, and the net human torque τ_{refl} is shown in green. It can be seen in 5.12(b) that the impedance angular position tracking error improves during region A when the model parameters are adapting. Fig. 5.13 shows how the parameters adapt and converge over time during region A, after which the parameters are held constant for the remainder of the test.

To better visualize the level of telepresence in this experiment, Fig. 5.14 focuses on regions E and F and plots the patient torque τ_p and therapist torque τ_{refl} in Fig. 5.14(a) as well as the sum of the two human torques (τ_p+τ_{refl}) in Fig. 5.14(b). Note that, with the sign convention used in this research for the definitions of τ_p and τ_{refl}, the highest possible telepresence fidelity is achieved when the therapist's torque τ_{refl} is equal to the negative of the patient's torque τ_p so that, when added together, the ideal telepresence would be zero. For reference, the ideal telepresence is shown as a green line in Fig. 5.14(b) and the area between this line and the achieved telepresence ($\tau_p + \tau_{refl}$) is shown in yellow. This plotting convention is shown in Fig. 5.14(b) as well as in all future bilateral telerehabilitation plots to better display the difference between the achieved and ideal telepresence.

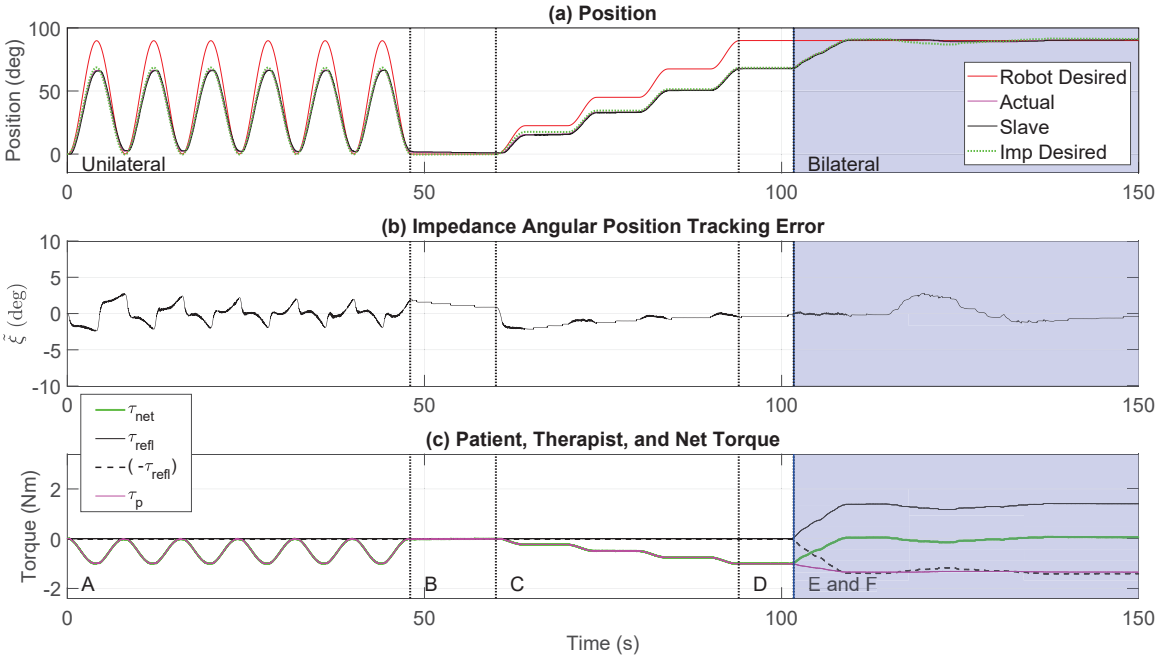


Figure 5.12: ARII telerehabilitation experiment with Quanser QUBEs with $\alpha = 1$: (a) position trajectory, (b) tracking error, and (c) torques

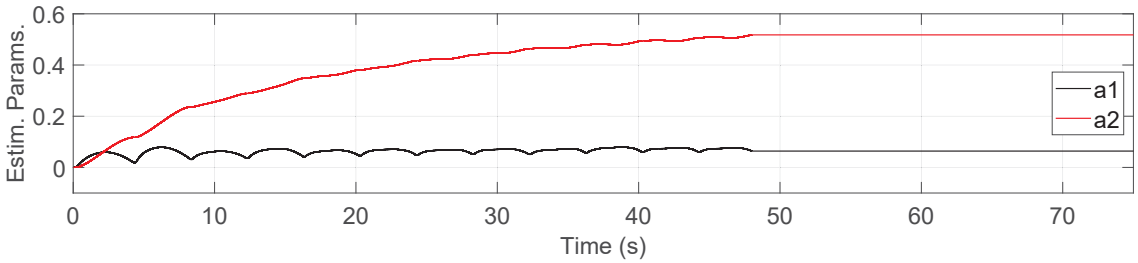


Figure 5.13: Estimated parameters for the ARII telerehabilitation experiment with the Quanser QUBEs with $\alpha = 1$

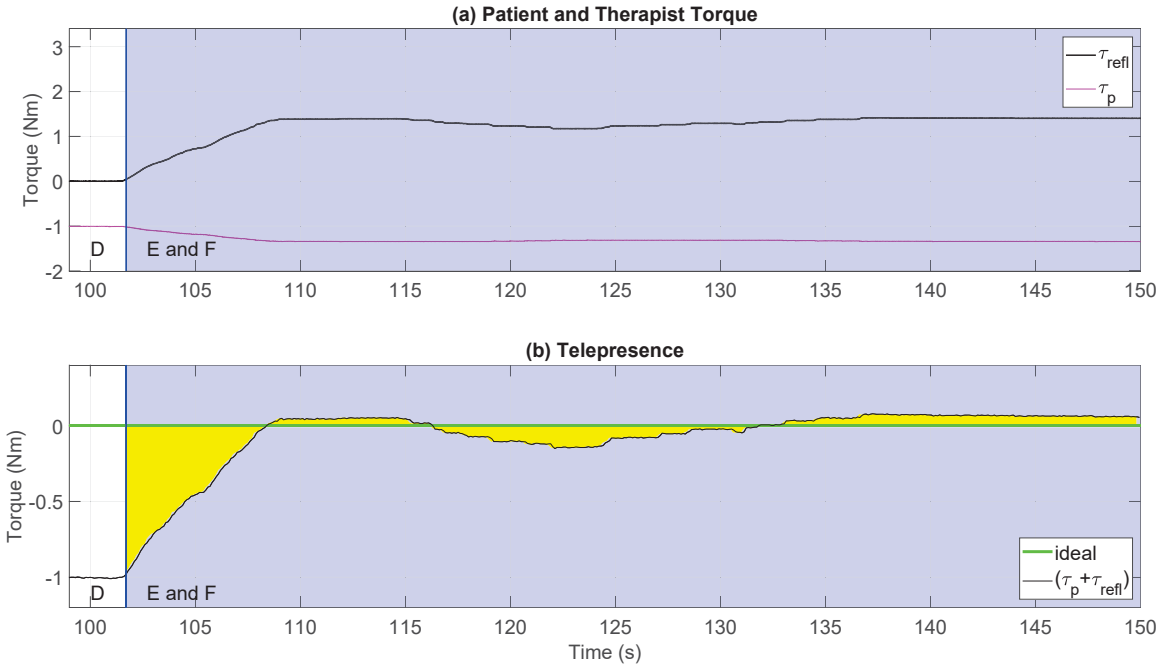


Figure 5.14: Torques for the ARII telerehabilitation experiment with Quanser QUBEs with $\alpha = 1$ during the bilateral phase

It is important to note that using the current direct addition human torque synthesis strategy characterized by (5.52), the initial telepresence fidelity is low since the master robot starts to move as soon as the therapist starts to provide small assistive torques. This initially poor telepresence can be seen in Fig. 5.14(b) as the yellow area at the beginning of region E and F. The therapist's torque then gradually ramps up over the course of about six seconds, and ultimately becomes close to the patient's torque as desired so that $(\tau_p + \tau_{refl})$ is close to the ideal value of zero. The position tracking and telepresence performance metrics for this $\alpha = 1$ experiment are listed in Table 5.10.

5.6.2 Scaled $\boldsymbol{\tau}_{refl}$ Addition

Another strategy presented in the literature [65] [60] for combining the human torques is to scale the therapist's torques as follows

$$\boldsymbol{\tau}_{net} = \boldsymbol{\tau}_p + \alpha\boldsymbol{\tau}_{refl}, \tag{5.53}$$

where $\alpha \in \Re^{n\times n}$ is a positive symmetric diagonal matrix containing the scaling factors applied to the $\boldsymbol{\tau}_{refl}$ vector. When the elements of α are greater than one, the therapist applies small assistive torques which get amplified by the α factor on the master side.

This strategy was implemented on the Quanser QUBE testbeds with the ARII control methodology and the experimental results are shown in Figs. 5.15 to 5.17 when the scaling factors in α are equal to three, as in [60]. Similar to the direct $\boldsymbol{\tau}_{refl}$-addition synthesis method described in the previous section, Fig. 5.15(a) plots the corresponding experimental angular position signals as functions of time, while Fig. 5.15(b) shows the experimental impedance angular position tracking error ξ when τ_{net} is calculated using (5.53). Fig. 5.15(c) plots the patient torque τ_p in fuchsia and the therapist torque τ_{refl} in black. Fig. 5.15(c) also plots the negative value of the therapist torque $(-\tau_{refl})$ in dashed black lines for comparison and the net human torque τ_{refl} in green. Fig. 5.16 shows the dynamic parameters adapting as a function of time and then being held constant from 48 seconds to the end of the test. Similar to Fig. 5.14, Fig. 5.17 focuses on the torques in the bilateral telerehabilitation-related regions E and F. As can be seen in Fig. 5.17(b), the achieved telepresence hovers around -1 Nm when using the scaled therapist's torque, which is worse than the case of direct addition of the therapist's torque. The relatively large yellow area in Fig. 5.17(b) highlights the poor telepresence achieved when using the $\alpha = 3$ scaled τ_{refl} strategy governed by (5.53), rendering the telepresence virtually non-existent with this strategy. The corresponding position tracking and telepresence performance metrics for this $\alpha = 3$ experiment are listed in Table 5.10.

Using this scaled-$\boldsymbol{\tau_{refl}}$ strategy is a double-edge sword: On one side the scaling is beneficial because the therapist is able to apply reduced torques and, therefore, experience less fatigue throughout the day as the therapist assists other patients [60].

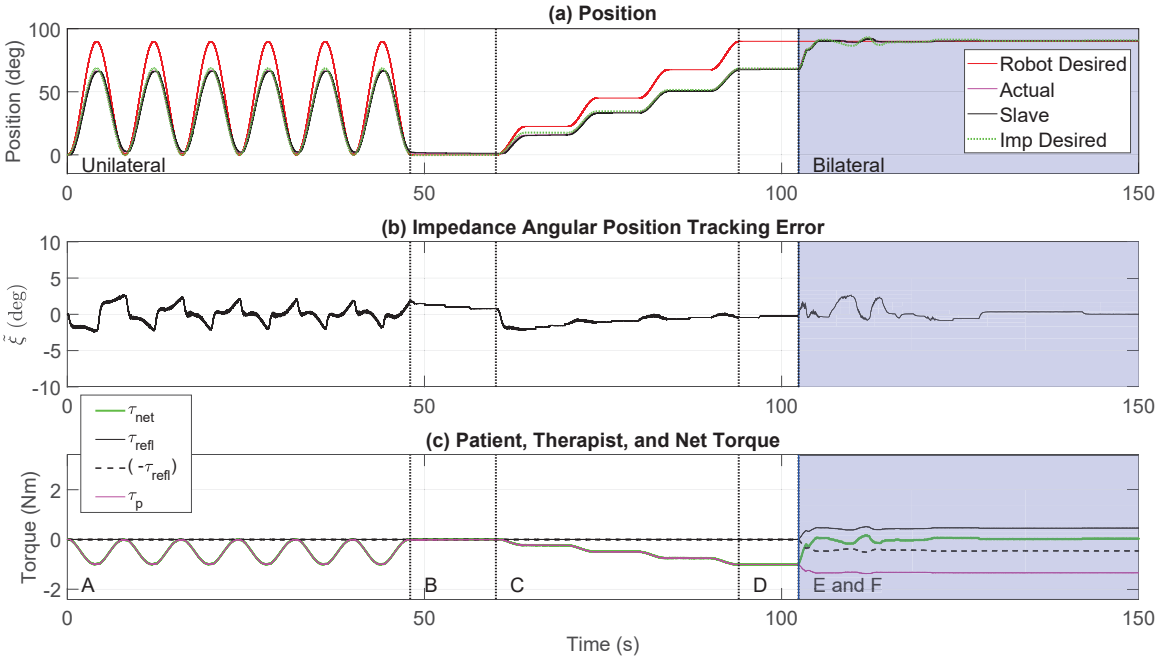


Figure 5.15: ARII telerehabilitation experiment with Quanser QUBEs with $\alpha = 3$: (a) position trajectory, (b) tracking error, and (c) torques

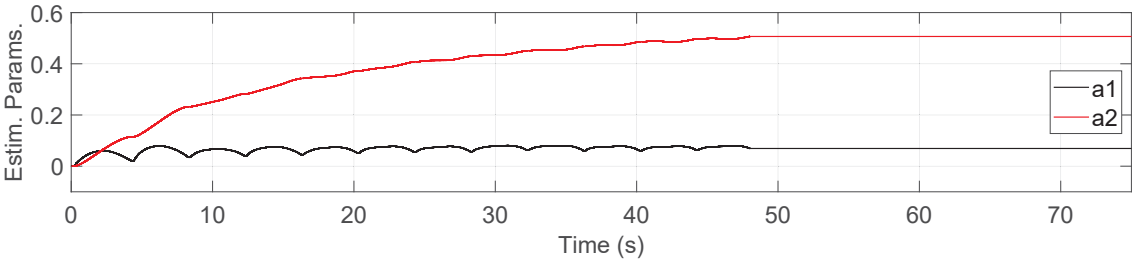


Figure 5.16: Estimated parameters for the ARII telerehabilitation experiment with the Quanser QUBEs with $\alpha = 3$

On the other side, however, by having the therapist apply scaled-down torques which are amplified by α on the master side, there is a disparity between the magnitude of the therapist's and the patient's torques, which destroys any chance for the therapist to appreciate the amount of assistive torque that the patient is actually experiencing. As a result, poor telepresence fidelity occurs in this case. For example, while performing these experimental tests when $\alpha = 3$, the present author did not experience a proportional and comprehensive perceptive awareness of the effort experienced by the patient modeled, which could result in a potential safety hazard and underestimation of the patient's level of inability. Furthermore, a sensation of disjointed kinesthetic

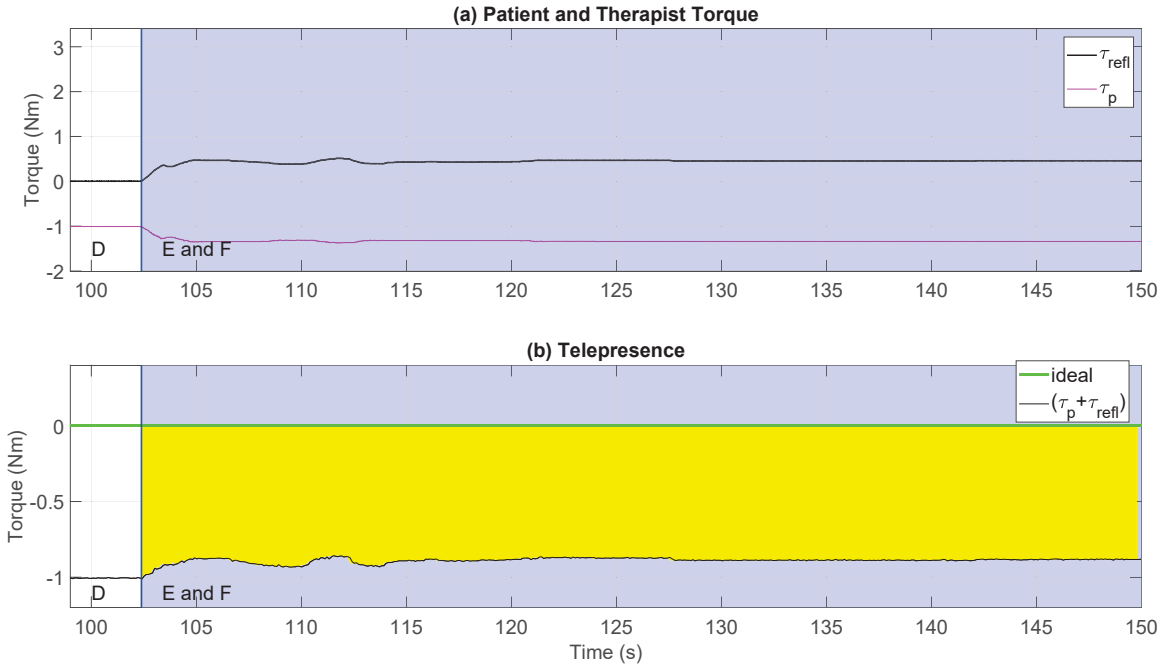


Figure 5.17: Torques for the ARII telerehabilitation experiment with Quanser QUBEs with $\alpha = 3$ during the bilateral phase

feedback was felt by the present author, similar to the feeling of 'slipping on the ice', or using the mouse on a gamer's computer − where the the cursor speed is typically set to a much higher value than for typical word-processing or coding-computer usage. Additionally, there is a learning curve that requires practice to get used to the amplification effects since very small efforts on the therapist's side have large effects on the master side − similar to using a large wrench on a small bolt where one needs to tone down their torque input to prevent any damage.

5.6.3 Human Torque Regulator

While performing extensive experimental tests implementing the two previously-mentioned human torque synthesis strategies on the Quanser QUBE testbeds with the Driving Force wheel for human-input, the present author aspired to improve the telepresence perception for the therapist in bilateral telerehabilitation scenarios beyond what is currently found in the literature. It was desired that the therapist should apply an equal amount of torque on the slave robot as the patient is applying on the master side before the robot's position on the master side is affected by the therapist's torques. Only when the therapist's torque matches the patient's torque should

the therapist's torque start assisting on the master side, with matched intensity as the patient. This strategy would enhance the therapist's telepresence, allowing the therapist to gauge the patient's effort on the master side. Furthermore, applying the same torque as the patient, instead of a torque that could be amplified several times, increases the safety of the telerehabilitation program. It is safer for the patient if the therapist is exerting the same torque as the patient so that the therapist can better sense what the patient is experiencing. To achieve this desired telepresence, the present author proposed and developed the following novel human torque regulator (HTR) approach to calculate $\boldsymbol{\tau}_{net}$

$$\boldsymbol{\tau}_{net} = \begin{cases} \boldsymbol{\tau}_{net} + \psi(\boldsymbol{\tau}_{refl} + \boldsymbol{\tau}_p), & if\ \boldsymbol{\tau}_{refl} \geq \boldsymbol{\tau}_p, \\ \boldsymbol{\tau}_p, & if\ \boldsymbol{\tau}_{refl} < \boldsymbol{\tau}_p. \end{cases} \tag{5.54}$$

As shown in this proposed approach, when $\boldsymbol{\tau}_{refl}$ is equal to or surpasses $\boldsymbol{\tau}_p$ in magnitude, the value of $\boldsymbol{\tau}_{net}$ is adjusted depending on the value of $\boldsymbol{\tau}_{refl}$ and $\boldsymbol{\tau}_p$. When $\boldsymbol{\tau}_{refl}$ is smaller than $\boldsymbol{\tau}_p$ in magnitude, $\boldsymbol{\tau}_{net}$ is equated to $\boldsymbol{\tau}_p$. Furthermore, $\psi \in \Re^{n\times n}$ in (5.54) is a positive definite diagonal matrix of proportional control gains which acts to reduce the error between $\boldsymbol{\tau}_p$ and $\boldsymbol{\tau}_{refl}$. The effect of this proposed formulation is that the net torque consists of a switching region and a linear region of operation with the goal of allowing for customizable speed of operation for the therapist-induced assistive efforts as well as enabling high-fidelity telepresence for the therapist.

Similar to the direct and scaled $\boldsymbol{\tau}_{refl}$-addition synthesis methods, Fig. 5.18(a) plots the corresponding experimental angular position signals as functions of time, while Fig. 5.18(b) shows the experiment impedance angular position tracking error ξ when τ_{net} is calculated using the new HTR shown in (5.54). Figs. 5.18(c) plots the patient torque τ_p in fuchsia and the therapist torque τ_{refl} in black. Fig. 5.18(c) also plots the negative value of the therapist torque $(-\tau_{refl})$ in dashed black lines for comparison and the net human torque τ_{refl} in green.

Fig. 5.19 shows the convergent adaptive parameters and their terminal, constant values from 48 seconds to the end of the test. Fig. 5.20 focuses on the regions D, E, and F shown in Fig. 5.18(c). It can be seen in Fig. 5.20(a) that, unlike the previous $\alpha = 1$ and $\alpha = 3$ strategies, the start of region E is different from the start of region

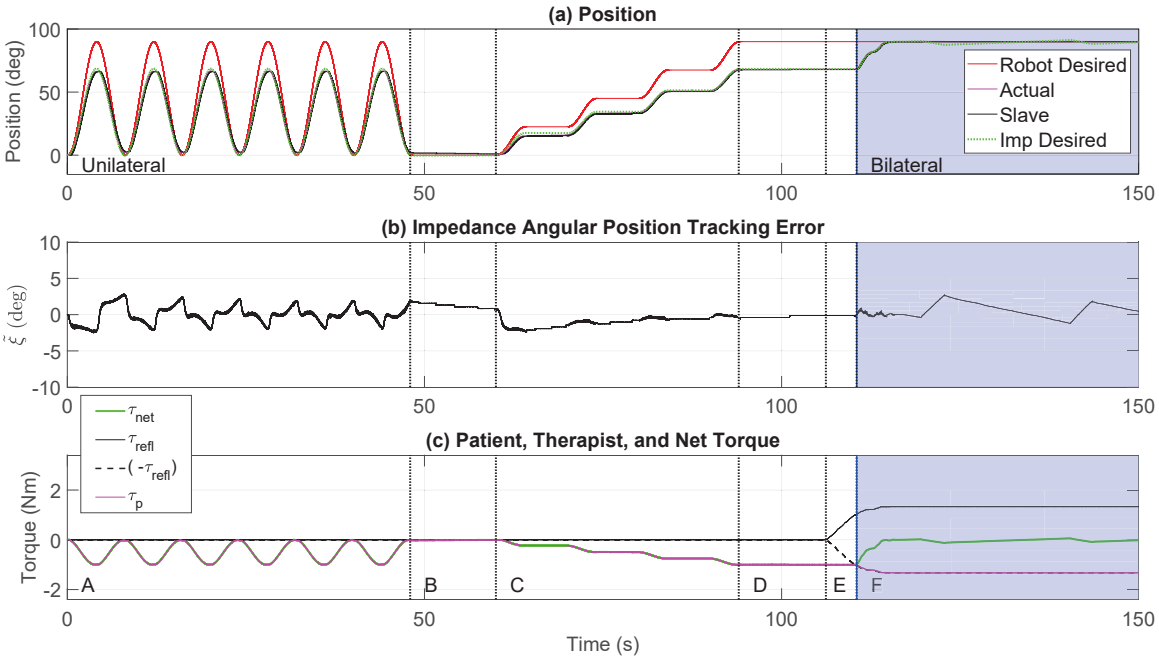


Figure 5.18: ARII telerehabilitation experiment with Quanser QUBEs with HTR: (a) position trajectory, (b) tracking error, and (c) torques

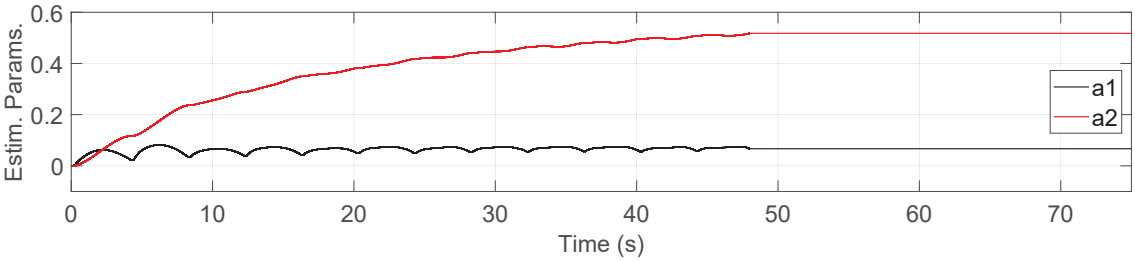


Figure 5.19: Estimated parameters for the ARII telerehabilitation experiment with the Quanser QUBEs with HTR

F. The start of region E corresponds to when the therapist begins to apply torque to try to assist the patient. In the previous $\alpha = 1$ and $\alpha = 3$ strategies, the moment the therapist began applying even small amounts of assistive torque, the patient began to move in response to the therapist's torque − initiating bilateral telerehabilitation defined by region F. Using the proposed HTR approach, however, when the therapist begins to apply torques to try to assist the patient, the patient does not begin to move right away. The therapist has to continue to increase the assistive torque they are applying until the therapist's torque matches that of the patient. Only when the therapist is applying the same torque felt by the patient do the patient's arm and

master robotic exoskeleton begin to respond to the therapist's assistance. Region E in Fig. 5.20(a) corresponds to this build up of the therapist's torque to match the patient's torque, while region F begins only when the therapist and patient torques match and the exoskeletons begin to move to commence the bilateral telerehabilitation stage. Fig. 5.20(b) illustrates the excellent telepresence achieved in the bilateral telerehabilitation region F using the proposed HTR strategy. The proposed HTR approach, therefore, not only allows the therapist to accurately gauge what the patient is feeling by the end of region E (before any bilateral telerehabilitation occurs) but also provides the therapist with excellent telepresence during bilateral telerehabilitation (region F) by ensuring that both sides work together in synergy − with the therapist having a heightened awareness of the magnitude of the patient's torque. The position tracking and telepresence performance metrics for this experiment are listed in Table 5.10. It can be seen in this table that the improved telepresence achieved with the new HTR technique is captured by the zero (displayed to one decimal place) TP_{max} and TP_{RMSE} metrics compared to the non-zero metrics attained using the $\alpha = 1$ and $\alpha = 3$ strategies.

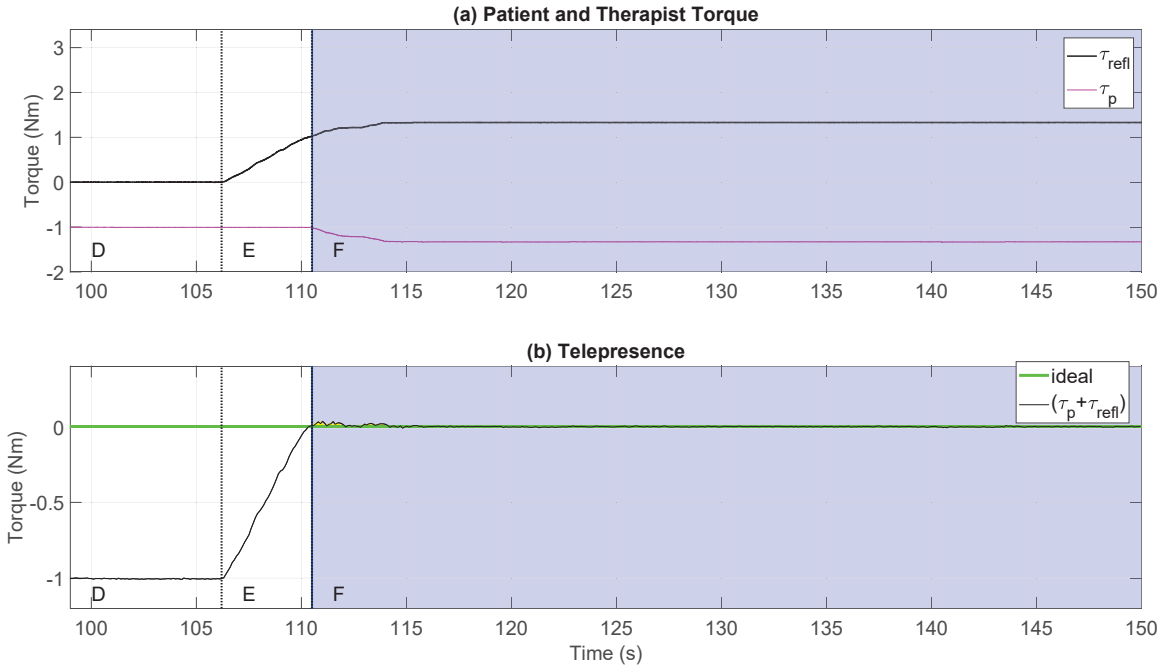


Figure 5.20: Torques for the ARII telerehabilitation experiment with Quanser QUBEs with HTR during the bilateral phase

Table 5.10: Performance Metrics for ARII Telerehabilitation Experiments with Quanser QUBEs

Metrics	$\boldsymbol{\alpha = 1}$	$\boldsymbol{\alpha = 3}$	$\boldsymbol{HTR}$
$\tilde{\xi}_{max}$	2.8°	2.7°	2.8°
$\tilde{\xi}_{RMSE}$	1.1°	1.0°	1.1°
TP_{max}	**1.0**Nm	**1.0**Nm	**0.0**Nm
TP_{RMSE}	**0.2**Nm	**0.9**Nm	**0.0**Nm

5.7 Telerehabilitation Human Torque Combination Simulations with 2-DOF Robotic Exoskeleton Model

Having validated the proposed ARII control approach on single DOF bilateral telerehabilitation experiments, the ARII method is now validated in telerehabilitation scenarios on multi-DOF systems by using the designed 2-DOF robotic exoskeleton simulation testbed and applying the three different net human torque synthesis strategies presented in the previous sections. Table 5.11 summarizes the regions used throughout the 2-DOF exoskeleton telerehabilitation simulations.

Table 5.11: Regions Definition for Single-System 2-DOF Exoskeleton Simulations

Region	**Time Range**	**Operation**
Joint 1		
A	0-48 s	oscillate between 0° and 90°, adaptation period
B	48-60 s	hold still at 0°
C	60-94 s	increase position in 22.5° increments, up to 90°
D	94-150 s	hold still at 90°
E	100-150 s	bilateral telerehabilitation is possible
F	*start* ⟩ 100-150 s	therapist's torque assists patient
Joint 2		
A	0-48 s	oscillate between -45° and 45°, adaptation period
B	48-120 s	hold still at -45°
C	120-124 s	go to 45°
D	124-150 s	hold still at 45°
E	126-150 s	bilateral telerehabilitation is possible
F	*start* ⟩ 126-150 s	therapist's torque assists patient

5.7.1 Direct τ_{refl} Addition

Figs. 5.21 to 5.24 show the results of the simulation when the reflected assistive therapist torque is directly added to the patient's torque using (5.52). Similar to the experiments with the Quanser QUBE experimental testbeds, Figs. 5.21(a) and 5.23(a) plot the experimental angular position signals as functions of time for joint 1 (elbow) and joint 2 (wrist), respectively. Figs. 5.21(b) and 5.23(b) show the impedance angular position tracking error ξ for each joint. Figs. 5.21(c) and 5.22(a) plot the patient torque τ_p in fuchsia and the therapist torque τ_{refl} in black for joint 1, while 5.23(c) and 5.24(a) plot the respective torque signals for joint 2. Figs. 5.21(c) and 5.23(c) also plot the negative value of the therapist torque $(-\tau_{refl})$ in dashed black lines for size comparison, and the net human torque τ_{refl} is shown in green for the entire simulation on both plots.

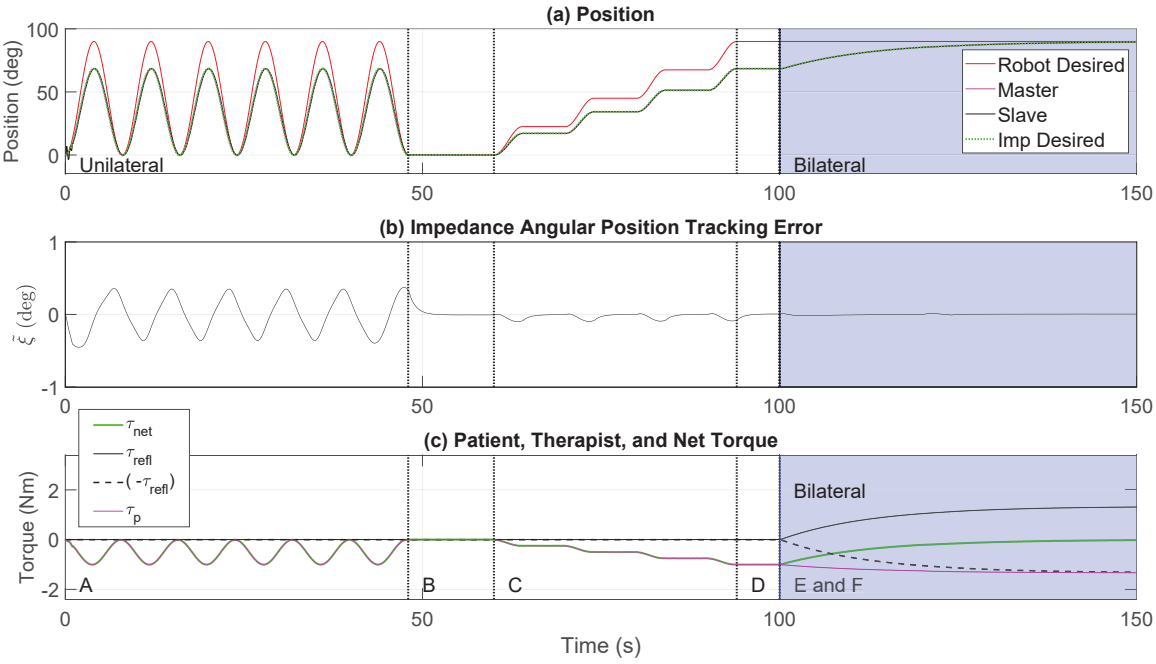


Figure 5.21: Joint 1 ARII 2-DOF telerehabilitation simulation with $\alpha = 1$: (a) position trajectory, (b) tracking error, and (c) torques

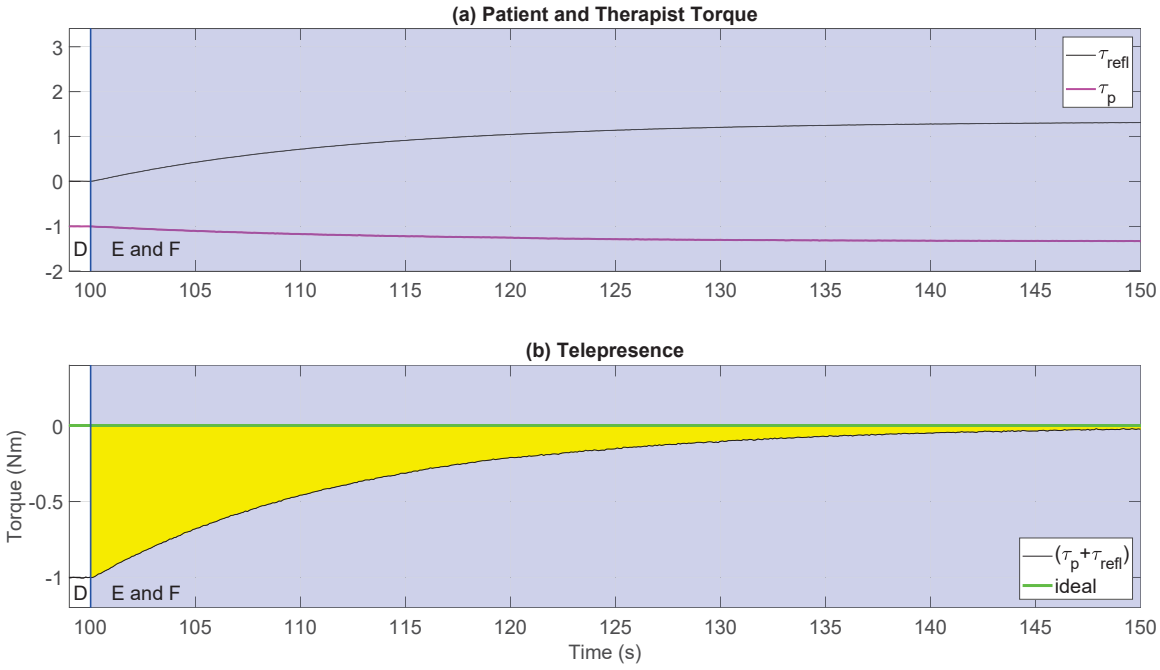


Figure 5.22: Joint 1 torques for the ARII 2-DOF telerehabilitation simulation with $\alpha = 1$ during the bilateral phase

As can be seen in Figs. 5.22(b) and 5.24(b) in regions E and F, the telepresence slowly converges towards the ideal value of zero under the direct addition of the therapist's torque to the patient's torque. Similar to the previous section, the yellow area highlights the relatively large difference between the achieved telepresence and the ideal telepresence. The position tracking and telepresence performance metrics for this $\alpha = 1$ simulation test are summarized in Table 5.12.

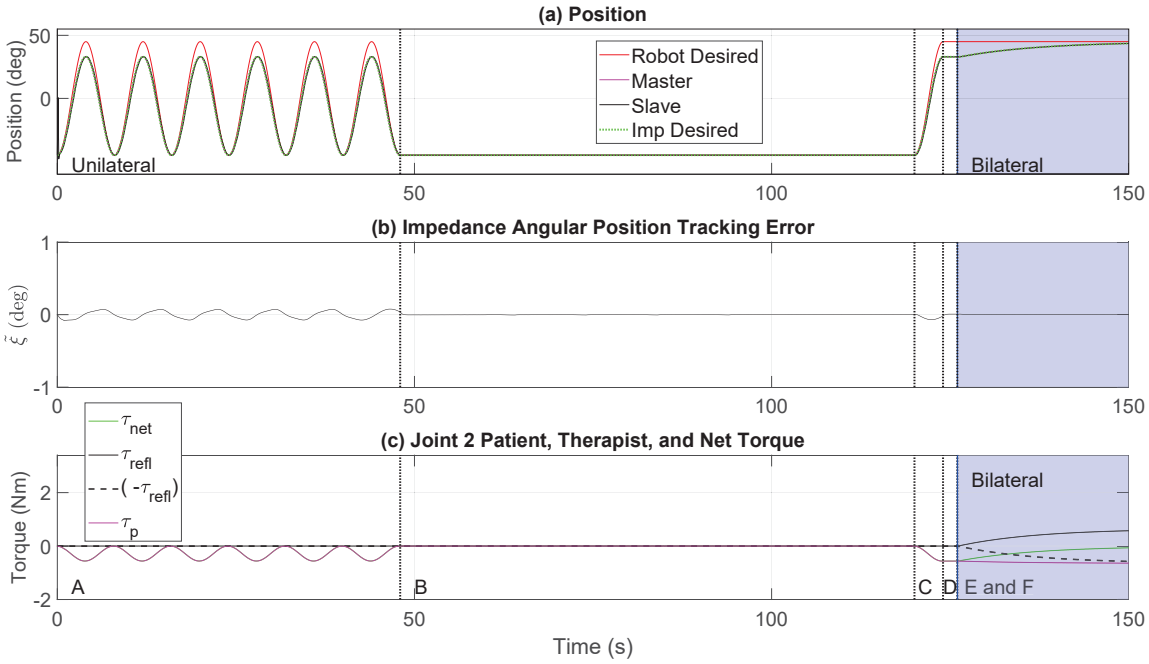


Figure 5.23: Joint 2 ARII 2-DOF telerehabilitation simulation with $\alpha = 1$: (a) position trajectory, (b) tracking error, and (c) torques

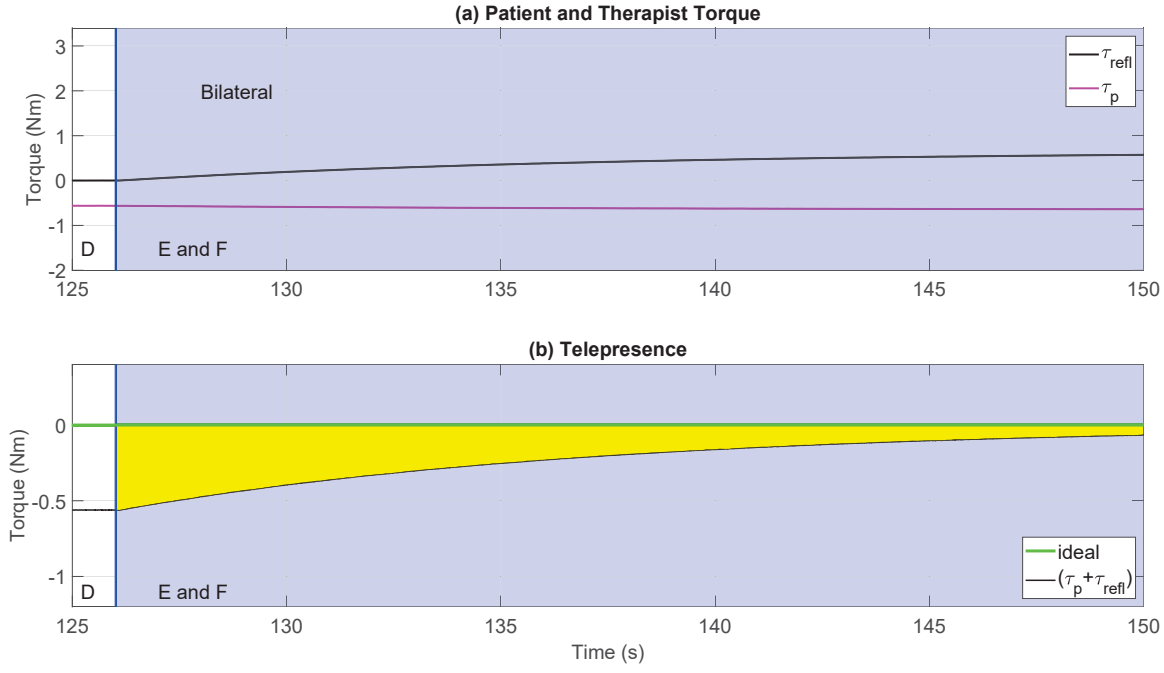


Figure 5.24: Joint 2 torques for the ARII 2-DOF telerehabilitation simulation with $\alpha = 1$ during the bilateral phase

5.7.2 Scaled $\boldsymbol{\tau}_{refl}$ Addition

Figs. 5.25 to 5.28 show the 2-DOF exoskeleton simulation results from the tests using the $\alpha = 3$ scaled-$\boldsymbol{\tau}_{refl}$ addition strategy given by (5.53). These figures follow a similar format to Figs. 5.21 to 5.24. As can be seen in Figs. 5.26(b) and 5.28(b) in regions E and F, the telepresence values converge to just under -1 Nm and -0.5 Nm, for joints 1 and 2, respectively. The low-fidelity telepresence is evident by the large yellow areas in the telepresence plot between the ideal telepresence green line and the telepresence achieved for this test. Comparing 5.26(b) with 5.22(b), it is evident that when $\alpha = 3$ the results exhibit even worse telepresence behaviour for the therapist than when $\alpha = 1$. The position tracking and telepresence performance metrics for this $\alpha = 3$ simulation are listed in Table 5.12 where the worse performance of $\alpha = 3$ compared to $\alpha = 1$ can clearly be seen in the TP_{max} and TP_{RMSE} telepresence metrics.

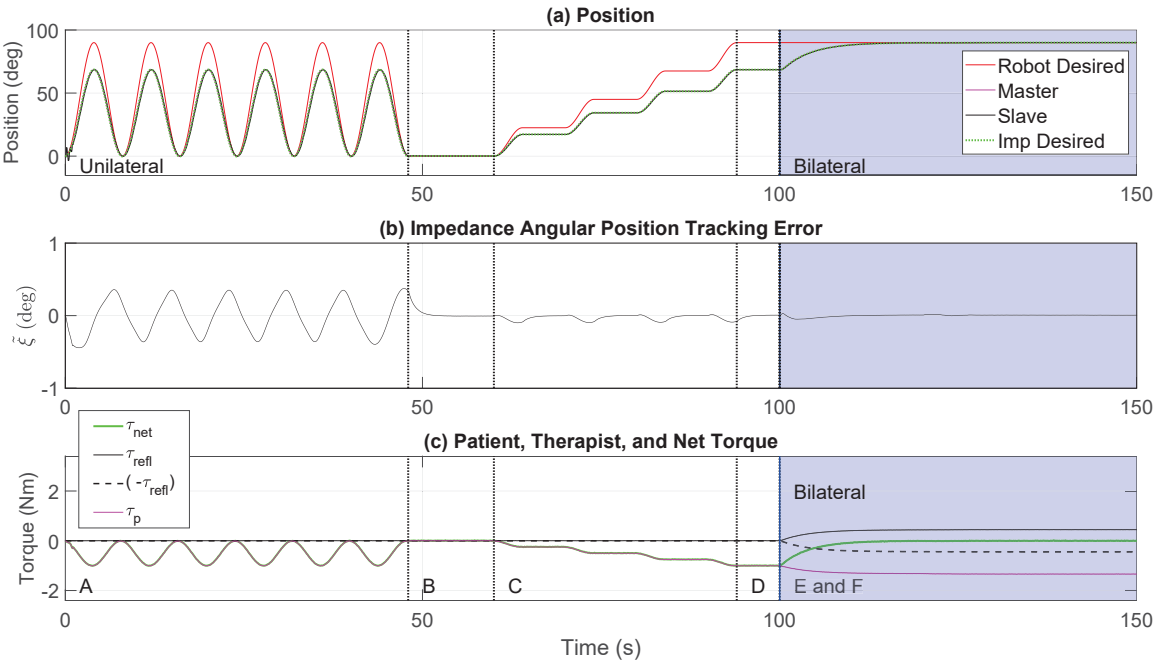


Figure 5.25: Joint 1 ARII 2-DOF telerehabilitation simulation with $\alpha = 3$: (a) position trajectory, (b) tracking error, and (c) torques

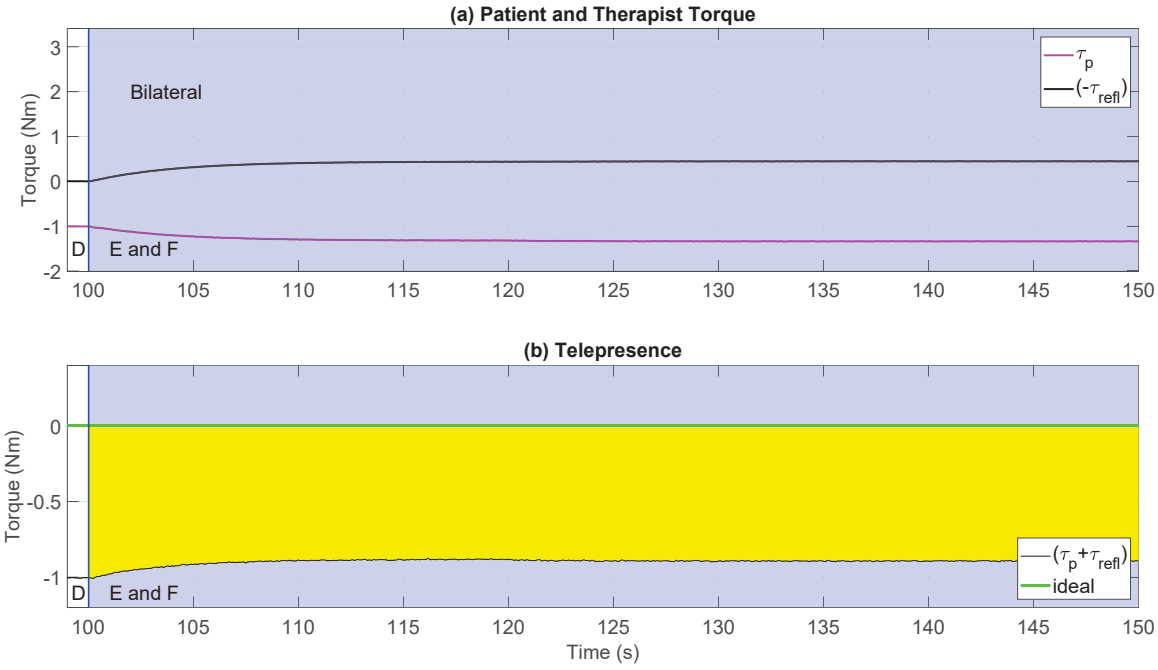


Figure 5.26: Joint 1 torques for the ARII 2-DOF telerehabilitation simulation with $\alpha = 3$ during the bilateral phase

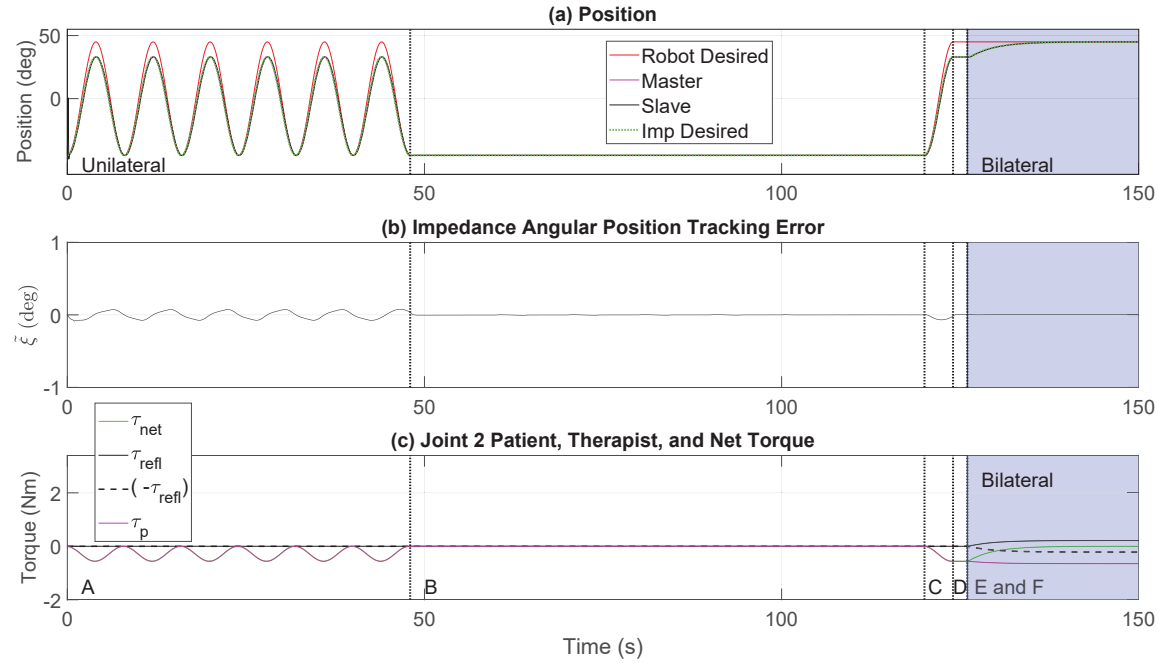


Figure 5.27: Joint 2 ARII 2-DOF telerehabilitation simulation with $\alpha = 3$: (a) position trajectory, (b) tracking error, and (c) torques

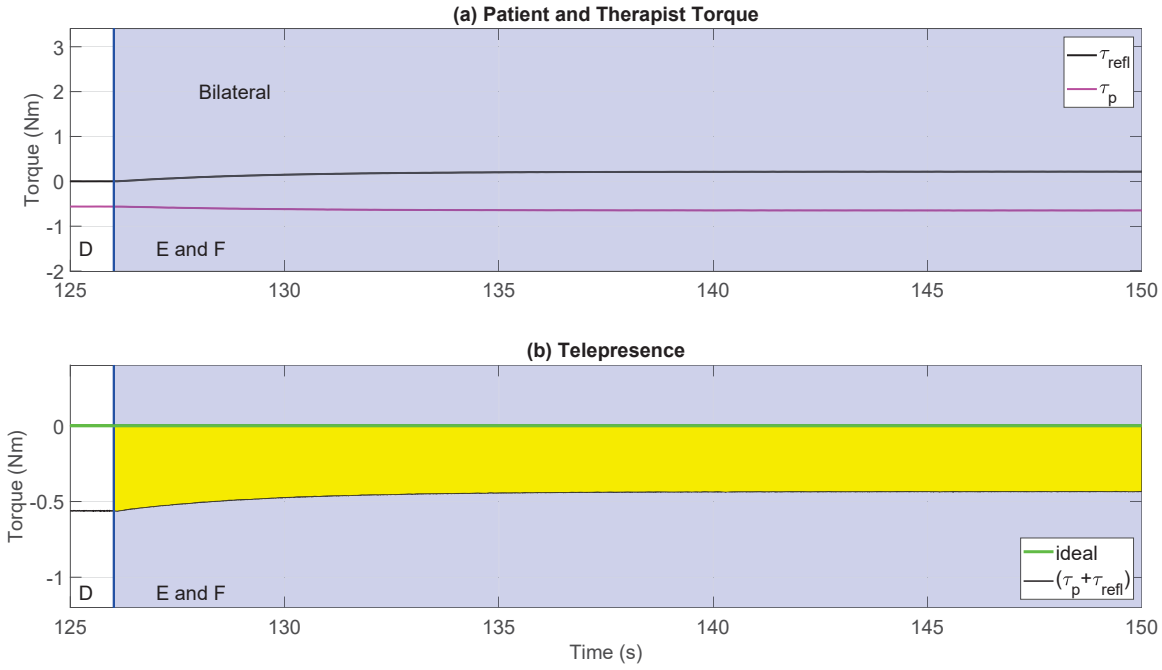


Figure 5.28: Joint 2 torques for the ARII 2-DOF telerehabilitation simulation with $\alpha = 3$ during the bilateral phase

5.7.3 Human Torque Regulator

Figs. 5.29 to 5.32 show the simulation results for multi-degrees of freedom implementation of the new HTR strategy introduced in this research in (5.54). These figures again follow a similar format to Figs. 5.21 to 5.24 and Figs. 5.25 to 5.28 in the previous sections. As can be seen in Figs. 5.30(b) and 5.32(b), once the therapist's torque builds up to the patient's toque in region E, it becomes equal to the patient's torque and bilateral telerehabilitation begins (corresponding to region F). During this bilateral telerehabilitation region, even with 2-DOF, the new HTR strategy provides excellent telepresence. This high-fidelity telepresence is evident by the lack of yellow areas in the telepresence plots of Figs. 5.30 and 5.32 for joints 1 and 2, respectively. The position tracking and telepresence performance metrics for this experiment are listed in Table 5.12. Similar to the telerehabilitation experiment with the Quanser QUBEs, it can be seen in this table that the proposed HTR approach provides superior telepresence compared with the $\alpha = 1$ and $\alpha = 3$ approaches since the telepresence metrics TP_{max} and TP_{RMSE} are zero Nm for both joint 1 and joint 2 as desired.

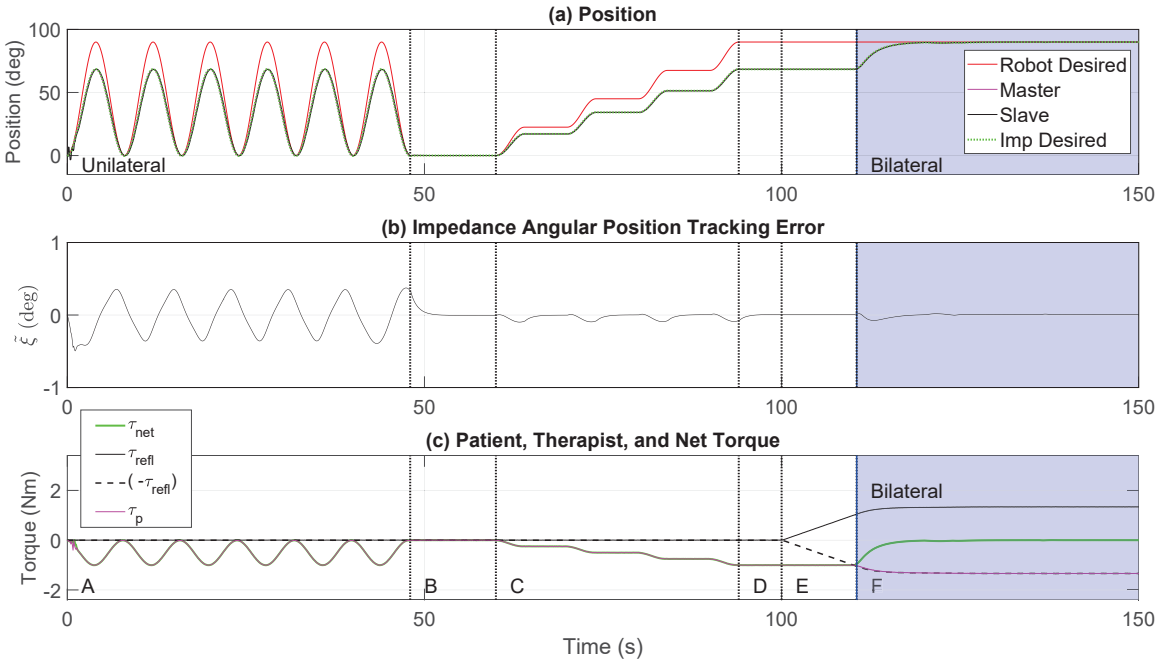


Figure 5.29: Joint 1 ARII 2-DOF telerehabilitation simulation with HTR: (a) position trajectory, (b) tracking error, and (c) torques

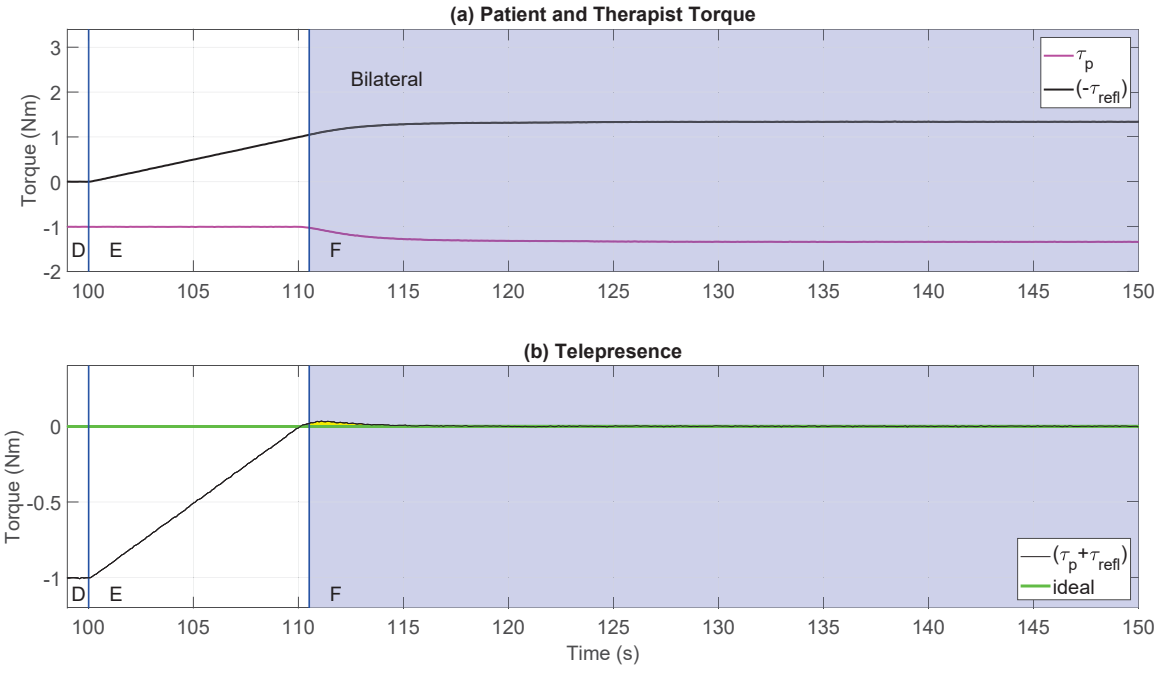


Figure 5.30: Joint 1 torques for the ARII 2-DOF telerehabilitation simulation with HTR during the bilateral phase

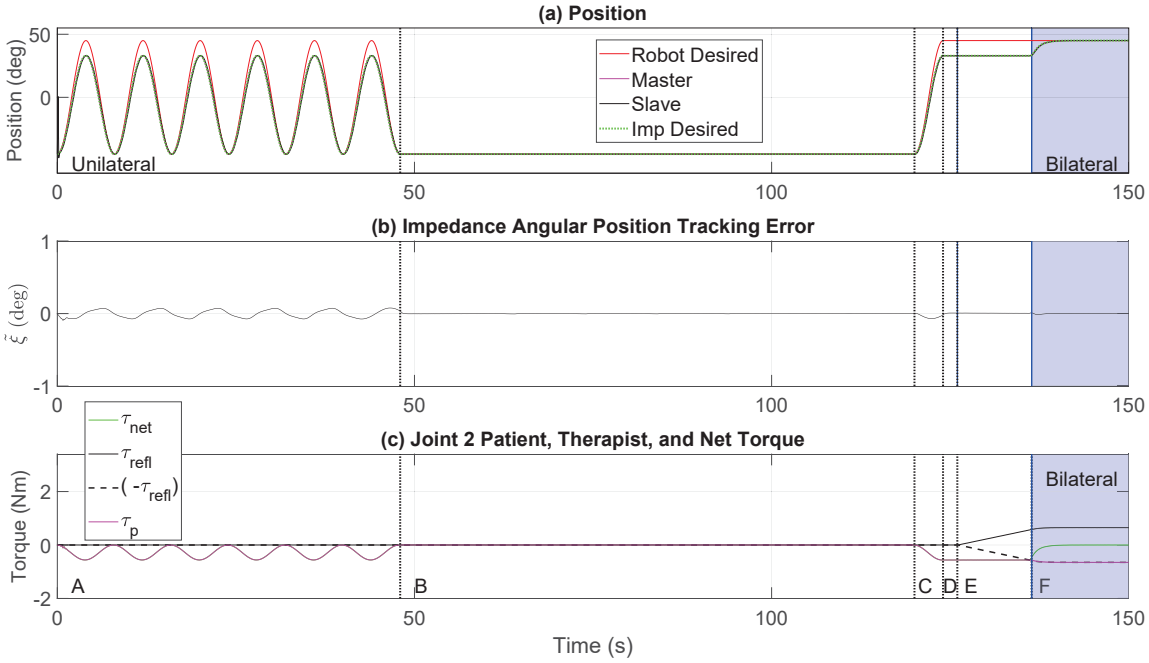


Figure 5.31: Joint 2 ARII 2-DOF telerehabilitation simulation with HTR: (a) position trajectory, (b) tracking error, and (c) torques

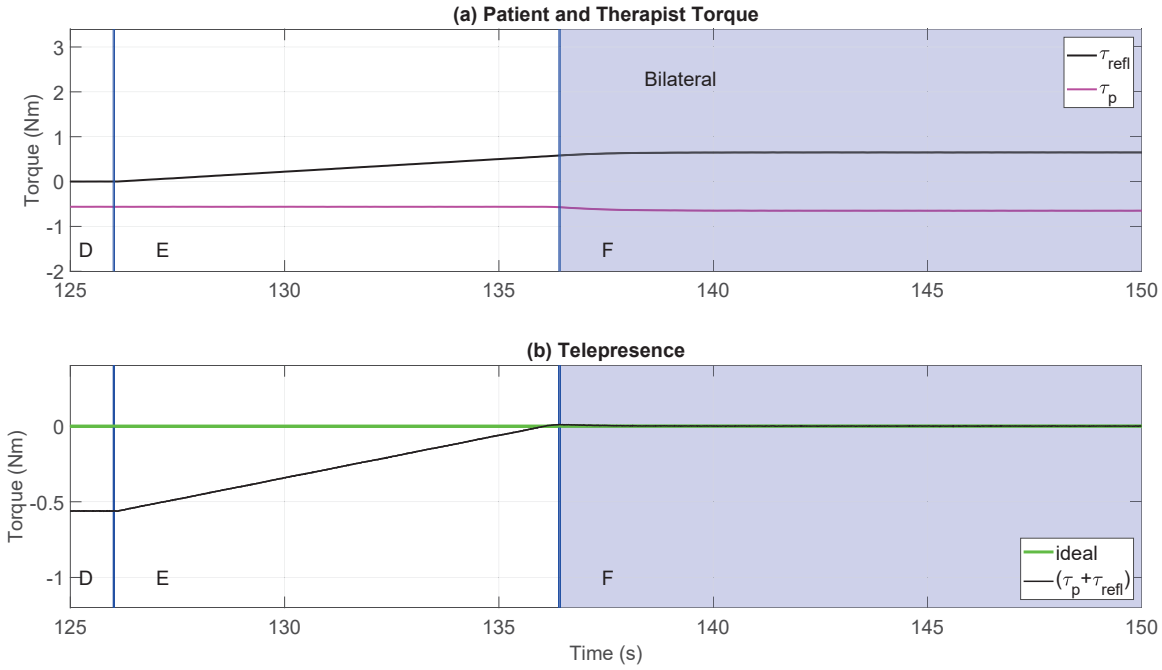


Figure 5.32: Joint 2 torques for the ARII 2-DOF telerehabilitation simulation with HTR during the bilateral phase

Table 5.12: Performance Metrics for ARII Telerehabilitation Simulations with 2-DOF Exoskeleton

Actuator	Metrics	$\boldsymbol{\alpha = 1}$	$\boldsymbol{\alpha = 3}$	$\boldsymbol{HTR}$
Joint 1	$\tilde{\xi}_{max}$	0.5°	0.4°	0.5°
	$\tilde{\xi}_{RMSE}$	0.1°	0.1°	0.1°
	TP_{max}	**1.0**Nm	**1.0**Nm	**0.0**Nm
	TP_{RMSE}	**0.4**Nm	**0.9**Nm	**0.0**Nm
Joint 2	$\tilde{\xi}_{max}$	0.1°	0.1°	0.1°
	$\tilde{\xi}_{RMSE}$	0.0°	0.0°	0.0°
	TP_{max}	**0.6**Nm	**0.6**Nm	**0.0**Nm
	TP_{RMSE}	**0.3**Nm	**0.5**Nm	**0.0**Nm

5.8 Discussions

As can be seen from the experimental and simulation tests presented in this chapter, the ARII method is stable and robust in both single-system, unilateral, and bilateral telerehabilitation for both 1-DOF and 2-DOF systems. The method has been shown to be robust in the presence of sensor noise, friction and human interactions, and adapted well to the unknown dynamic parameters.

5.9 Contributions

The ARII control method presented in this chapter has several contributions:

1. The adaptive control method was augmented to include an impedance model as well as an integral control term and a robust control term. The present author coined the term Adaptive Robust Integral Impedance (ARII) to refer to this proposed control approach.

2. The theoretical proof of the proposed ARII method for n-DOF system was developed.

3. The proposed ARII method was validated in simulation on the 1-DOF Quanser QUBE testbeds and 2-DOF robotic exoskeleton models.

4. The ARII method was experimentally validated using the 1-DOF Quanser

QUBE testbeds and the Logitech driving wheel to provide assistive human-input.

5. The results of an earlier version of the proposed adaptive impedance control method were published in: G. Bauer, Y.J. Pan, and H.H. Shen, "Adaptive Impedance Control in Bilateral Telerehabilitation with Robotic Exoskeletons," In Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics, October 2020, Toronto, Canada, pp.719-725 [70].

Chapter 6

Adaptive Robust Integral Radial Basis Function Neural Networks Impedance (RBFNN-I) Controller - Design, Simulations, Experiments, and Comparison

This chapter presents the development and implementation of an RBFNN-I control methodology and compares the results with the proposed ARII control method.

6.1 Radial Basis Function Neural Networks Control

In the previous chapter, the proposed ARII controller was successfully implemented and tested on three different testbeds. It is important to note, however, that although the ARII control method has the ability to adapt the values of the unknown dynamic model parameters, the method requires *a priori* detailed knowledge of the underlying dynamic equations that govern the behaviour of the plant being controlled. It was desired to design and test a different control method that requires less *a priori* information about the plant dynamics. Neural network-based controllers have the advantage of not requiring a lot of prior information about the structure and parameters of the plant to be controlled [169]. Radial Basis function Neural Network (RBFNN)-based controllers have the ability to solve highly-nonlinear problems and have been shown to be able to successfully adapt the neural network weights even with large uncertainty of the system [73]. An RBFNN has three layers, as shown in Fig. 6.1, consisting of an input layer, a nonlinear hidden layer, and an output layer. Each neuron in the input layer corresponds to an input variable. In the hidden layer, each neuron is activated by a radial basis function such as a Gaussian function, centered around a vector $\boldsymbol{c}_j$ with the same dimension g as the input variables $\boldsymbol{x}$. The nonlinear activation function used in this work is the Gaussian kernel $\phi_j(t)$ as follows

$$\phi_j(t) = \exp\left(- \frac{\|\boldsymbol{x}(t) - \boldsymbol{c}_j(t)\|^2}{\sigma^2} \right), \quad j = 1, ..., m, \tag{6.1}$$

where σ is the width of the Gaussian function $\boldsymbol{c}_j$ and m is the number of neurons in the hidden layer. This activation function provides a nonlinear mapping that enables the solution of highly nonlinear control problems which traditional control approaches cannot practically solve [73]. The network output layer is $\boldsymbol{f}$, which is the sum of the weighted outputs from the hidden layer as follows

$$f_i(t) = \sum_{j=1}^{m} w_{ji}\phi_j(t), \qquad i = 1, ..., d, \tag{6.2}$$

where w_{ji} are the synaptic weights connecting the hidden layer to output neurons, f_i are the elements of the vector $\boldsymbol{f}$, and d is the number of output variables.

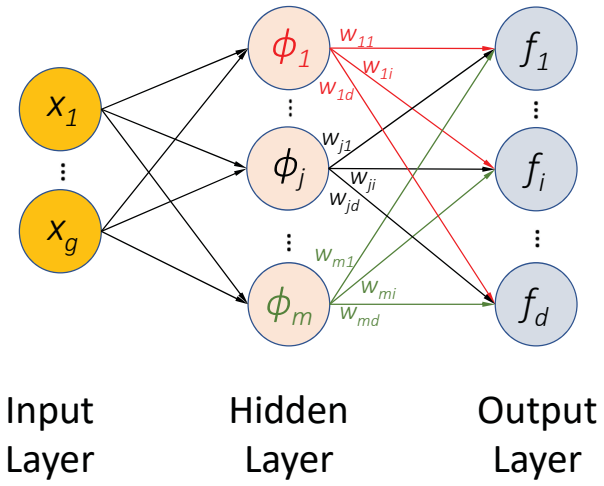


Figure 6.1: Radial basis function neural network structure

6.2 Adaptive Robust Integral Radial Basis Function Neural Networks with Impedance (RBFNN-I) Controller Design

Recall that the joint-space dynamics of an n-link rigid horizontally-planar robotic exoskeleton with human torque disturbance $\boldsymbol{\tau}_p$ are given by (4.1) which is reproduced here for convenience (with the subscript e and m notation dropped for brevity)

$$M(\boldsymbol{q})\ddot{\boldsymbol{q}} + C(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{\tau}_d = \boldsymbol{u} - \boldsymbol{\tau}_p. \tag{6.3}$$

In practice, the actual dynamics parameters of the robotic exoskeleton, $M(\boldsymbol{q})$ and $C(\boldsymbol{q}, \dot{\boldsymbol{q}})$ cannot be precisely known due to, for example, measurement inaccuracies,

material deformation, as well as thermal, manufacturing, and system assembly variation. While the proposed ARII control method in the previous chapter used $M_a(\boldsymbol{q})$ and $C_a(\boldsymbol{q}, \dot{\boldsymbol{q}})$ to model $M(\boldsymbol{q})$ and $C(\boldsymbol{q}, \dot{\boldsymbol{q}})$, the RBFNN-I method uses $M_{NN}(\boldsymbol{q})$ and $C_{NN}(\boldsymbol{q}, \dot{\boldsymbol{q}})$ as follows

$$M_{NN}(\boldsymbol{q})\ddot{\boldsymbol{q}} + C_{NN}(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{\tau}_d = \boldsymbol{u} - \boldsymbol{\tau}_p, \tag{6.4}$$

Note that

$$M = M_{NN} + E_M, \tag{6.5}$$

$$C = C_{NN} + E_C, \tag{6.6}$$

where E_M and E_C are the inherent modeling errors that arise when a model is used to represent a real system.

The RBFNN-based control method presented in [73] for robotic manipulators was further developed by adding impedance control to explore its use with robotic exoskeletons. The resulting proposed method, Adaptive Robust Integral Radial Basis Function Neural Network with Impedance (RBFNN-I), is desired to activate the robotic exoskeleton's motors according to a prescribed impedance model under real-time human torque input in the presence of unknown dynamic modeling parameters and external disturbances. Fig. 6.2 shows the corresponding control system diagram when the RBFNN-I controller uses the desired and actual joint motion in combination with measured human torques applied to the robotic exoskeleton to generate the control signal $\boldsymbol{u}$.

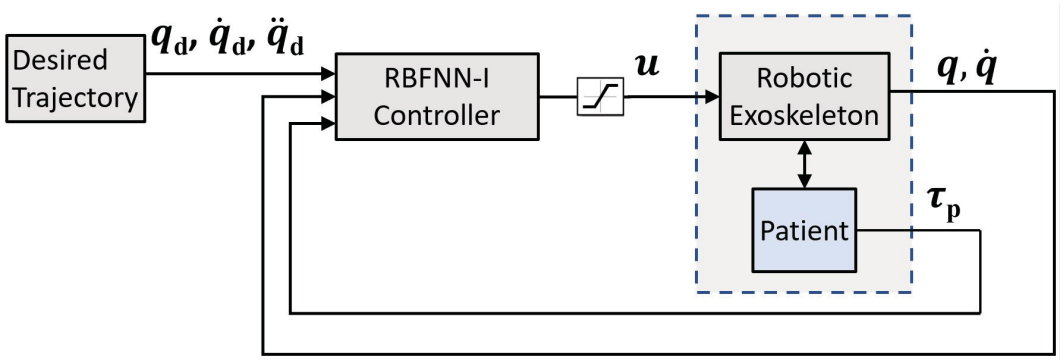


Figure 6.2: RBFNN-I control system diagram

Let the proposed RBFNN-I control law be

$$\boldsymbol{u} \triangleq \underbrace{\hat{M}_{NN}(\boldsymbol{q})\dot{\boldsymbol{v}} + \hat{C}_{NN}(\boldsymbol{q},\dot{\boldsymbol{q}})\boldsymbol{v} + \hat{M}_{NN}(\boldsymbol{q})\ddot{\tilde{\boldsymbol{q}}}_{\tau net} + \hat{C}_{NN}(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\tilde{\boldsymbol{q}}}_{\tau net}}_{\text{model-estimate term accounting for impedance}}$$
$$- \underbrace{K_p\boldsymbol{r}}_{\text{PD}} - \underbrace{K_i\int \boldsymbol{r}dt}_{\text{integral}} - \underbrace{\boldsymbol{\tau}_r}_{\text{robust}} + \underbrace{\boldsymbol{\tau}_{p,meas}}_{\text{patient}}, \tag{6.7}$$

where $\hat{M}_{NN}(\boldsymbol{q}) \in \Re^{n\times n}$ and $\hat{C}_{NN}(\boldsymbol{q},\dot{\boldsymbol{q}}) \in \Re^{n\times n}$ are the RBFNN-I learned estimates of $M_{NN}(\boldsymbol{q})$ and $C_{NN}(\boldsymbol{q},\dot{\boldsymbol{q}})$ in (6.4). In the proposed control law given by (6.7) $K_p \in \Re^{n\times n}$ and $K_i \in \Re^{n\times n}$ are diagonal, positive definite proportional and integral gain matrices, $\boldsymbol{\tau}_r \in \Re^n$ is the robust control term, and $\boldsymbol{\tau}_{p,meas} \in \Re^n$ is the measured human disturbance torque. Recall from §5.3 that, to incorporate impedance control, the virtual reference velocity and acceleration vectors $\boldsymbol{v} \in \Re^n$ and $\dot{\boldsymbol{v}} \in \Re^n$ in (6.7) were defined as

$$\boldsymbol{v} \triangleq \dot{\boldsymbol{q}}_d - \Lambda\tilde{\boldsymbol{\xi}}, \tag{6.8}$$

$$\dot{\boldsymbol{v}} \triangleq \ddot{\boldsymbol{q}}_d - \Lambda\dot{\tilde{\boldsymbol{\xi}}}, \tag{6.9}$$

where $\ddot{\boldsymbol{q}}_d \in \Re^n$ is the original desired joint angular acceleration trajectory and $\Lambda \in \Re^{n\times n}$ is a diagonal positive definite gain matrix. The sliding mode error $\boldsymbol{r}$ is then given by

$$\boldsymbol{r} \triangleq \dot{\boldsymbol{q}} - \dot{\tilde{\boldsymbol{q}}}_{\boldsymbol{\tau net}} - \boldsymbol{v} = \dot{\tilde{\boldsymbol{\xi}}} + \Lambda\tilde{\boldsymbol{\xi}}. \tag{6.10}$$

Note that from (6.10) as $\boldsymbol{r} \to 0$, $\tilde{\boldsymbol{\xi}} \to 0$ and $\dot{\tilde{\boldsymbol{\xi}}} \to 0$. Similarly

$$\dot{\boldsymbol{r}} \triangleq \ddot{\boldsymbol{q}} - \ddot{\tilde{\boldsymbol{q}}}_{\boldsymbol{\tau net}} - \dot{\boldsymbol{v}} = \ddot{\tilde{\boldsymbol{\xi}}} + \Lambda\dot{\tilde{\boldsymbol{\xi}}}. \tag{6.11}$$

6.2.1 RBFNN-I Controller Theoretical Development

This section develops the relationships needed to ultimately prove that, using the proposed RBFNN-I control law, as $\boldsymbol{t} \to \infty$, $\tilde{\boldsymbol{\xi}}(t) \to 0$ as desired. Substituting the proposed control law given by (6.7) into the robotic exoskeleton equation of motion given by (6.4), yields

$$M\ddot{\boldsymbol{q}} + C\dot{\boldsymbol{q}} + \boldsymbol{\tau}_d = \hat{M}_{NN}\dot{\boldsymbol{v}} + \hat{C}_{NN}\boldsymbol{v} + \hat{M}_{NN}\ddot{\tilde{\boldsymbol{q}}}_{\tau net} + \hat{C}_{NN}\dot{\tilde{\boldsymbol{q}}}_{\tau net}$$
$$- K_p\boldsymbol{r} - K_i\int \boldsymbol{r}dt - \boldsymbol{\tau}_r + \boldsymbol{\tau}_{p,meas} - \boldsymbol{\tau}_p. \tag{6.12}$$

Observing (6.10) and (6.11), the following equalities are true

$$\dot{\boldsymbol{q}} = \boldsymbol{r} + \boldsymbol{v} + \dot{\tilde{\boldsymbol{q}}}_{\tau net}, \tag{6.13}$$

$$\ddot{\boldsymbol{q}} = \dot{\boldsymbol{r}} + \dot{\boldsymbol{v}} + \ddot{\tilde{\boldsymbol{q}}}_{\tau net}. \tag{6.14}$$

Substituting (6.13) and (6.14) into (6.12),

$$\begin{aligned} M(\dot{\boldsymbol{r}} + \dot{\boldsymbol{v}} + \ddot{\tilde{\boldsymbol{q}}}_{\tau net}) + C(\boldsymbol{r} + \boldsymbol{v} + \dot{\tilde{\boldsymbol{q}}}_{\tau net}) =& \hat{M}_{NN}\dot{\boldsymbol{v}} + \hat{C}_{NN}\boldsymbol{v} + \hat{M}_{NN}\ddot{\tilde{\boldsymbol{q}}}_{\tau net} + \hat{C}_{NN}\dot{\tilde{\boldsymbol{q}}}_{\tau net} \\ & - K_p\boldsymbol{r} - K_i\int \boldsymbol{r}dt - \boldsymbol{\tau}_r - \boldsymbol{\tau}_d \\ & + \boldsymbol{\tau}_{p,meas} - \boldsymbol{\tau}_p. \end{aligned} \tag{6.15}$$

Further grouping all of the M and C terms on the right side of the equation gives the following equation

$$\begin{aligned} M\dot{\boldsymbol{r}} + C\boldsymbol{r} + K_i\int \boldsymbol{r}dt \ =& \ (\hat{M}_{NN} - M)(\dot{\boldsymbol{v}} + \ddot{\tilde{\boldsymbol{q}}}_{\boldsymbol{\tau net}}) + (\hat{C}_{NN} - C)(\boldsymbol{v} + \dot{\tilde{\boldsymbol{q}}}_{\boldsymbol{\tau net}}) \\ & - K_p\boldsymbol{r} - \boldsymbol{\tau}_r - \boldsymbol{\tau}_d + \boldsymbol{\tau}_{p,meas} - \boldsymbol{\tau}_p. \end{aligned} \tag{6.16}$$

Substituting (6.5) and (6.6) into (6.16) gives

$$\begin{aligned} M\dot{\boldsymbol{r}} + C\boldsymbol{r} + K_i\int \boldsymbol{r}dt \ =& \ (\hat{M}_{NN} - M_{NN} - E_M)(\dot{\boldsymbol{v}} + \ddot{\tilde{\boldsymbol{q}}}_{\boldsymbol{\tau net}}) \\ & + (\hat{C}_{NN} - C_{NN} - E_C)(\boldsymbol{v} + \dot{\tilde{\boldsymbol{q}}}_{\boldsymbol{\tau net}}) \\ & - K_p\boldsymbol{r} - \boldsymbol{\tau}_r - \boldsymbol{\tau}_d + \boldsymbol{\tau}_{p,meas} - \boldsymbol{\tau}_p. \end{aligned} \tag{6.17}$$

Letting $\tilde{M} = \hat{M}_{NN} - M_{NN}$ and $\tilde{C} = \hat{C}_{NN} - C_{NN}$ and substituting $\tilde{M}$ and $\tilde{C}$ into (6.17) yields

$$\begin{aligned} M\dot{\boldsymbol{r}} + C\boldsymbol{r} + K_i\int \boldsymbol{r}dt \ =& \ \tilde{M}(\dot{\boldsymbol{v}} + \ddot{\tilde{\boldsymbol{q}}}_{\boldsymbol{\tau net}}) + \tilde{C}(\boldsymbol{v} + \dot{\tilde{\boldsymbol{q}}}_{\boldsymbol{\tau net}}) \\ & - E_M(\dot{\boldsymbol{v}} + \ddot{\tilde{\boldsymbol{q}}}_{\boldsymbol{\tau net}}) - E_C(\boldsymbol{v} + \dot{\tilde{\boldsymbol{q}}}_{\boldsymbol{\tau net}}) \\ & - K_p\boldsymbol{r} - \boldsymbol{\tau}_r - \boldsymbol{\tau}_d + \boldsymbol{\tau}_{p,meas} - \boldsymbol{\tau}_p \\ =& \ \tilde{M}(\dot{\boldsymbol{v}} + \ddot{\tilde{\boldsymbol{q}}}_{\boldsymbol{\tau net}}) + \tilde{C}(\boldsymbol{v} + \dot{\tilde{\boldsymbol{q}}}_{\boldsymbol{\tau net}}) \\ & - K_p\boldsymbol{r} + E + E_d + E_p - \boldsymbol{\tau}_r, \end{aligned} \tag{6.18}$$

where

$$
\begin{aligned}
E &\triangleq -E_M(\dot{\boldsymbol{v}} + \ddot{\tilde{\boldsymbol{q}}}_{\boldsymbol{\tau net}}) - E_C(\boldsymbol{v} + \dot{\tilde{\boldsymbol{q}}}_{\boldsymbol{\tau net}}),\\
E_d &\triangleq -\boldsymbol{\tau}_d,\\
E_p &\triangleq \boldsymbol{\tau}_{p,meas} - \boldsymbol{\tau}_p.
\end{aligned}
$$

Next, RBF neural networks are used to calculate the values of $\tilde{M}$ and $\tilde{C}$ as follows [73]

$$
\begin{aligned}
\tilde{M} =& \Big[\big\{W_M\big\}^T\big\{\Phi_M(\boldsymbol{q})\big\}\Big] - \Big[\big\{\hat{W}_M\big\}^T\big\{\Phi_M(\boldsymbol{q})\big\}\Big]\\
=& \Big[\big\{\tilde{W}_M\big\}^T\big\{\Phi_M(\boldsymbol{q})\big\}\Big], \qquad (6.19)
\end{aligned}
$$

and

$$
\begin{aligned}
\tilde{C} =& \Big[\big\{W_C\big\}^T\big\{\Phi_C(\boldsymbol{z})\big\}\Big] - \Big[\big\{\hat{W}_C\big\}^T\big\{\Phi_C(\boldsymbol{z})\big\}\Big]\\
=& \Big[\big\{\tilde{W}_C\big\}^T\big\{\Phi_C(\boldsymbol{z})\big\}\Big], \qquad (6.20)
\end{aligned}
$$

where W_M, and W_C are the ideal RBFNN weight matrices, composed of the w_{ji} terms in (6.2), and $\hat{W}_M$ and $\hat{W}_C$ are estimates of W_M and W_C. Φ_M and Φ_C are the output vectors of the hidden layers in the RBFNN (6.1) of dimension m, and $\boldsymbol{z}$=$[\boldsymbol{q}^T, \dot{\boldsymbol{q}}^T]^T$. Incorporating the RBFNN parameters into (6.18), the control law becomes

$$
\begin{aligned}
M\dot{\boldsymbol{r}} + C\boldsymbol{r} + K_i\int \boldsymbol{r}dt \;=\; & \Big[\big\{\tilde{W}_M\big\}^T\big\{\Phi_M(\boldsymbol{q})\big\}\Big](\dot{\boldsymbol{v}} + \ddot{\tilde{\boldsymbol{q}}}_{\boldsymbol{\tau net}})\\
& + \Big[\big\{\tilde{W}_C\big\}^T\big\{\Phi_C(\boldsymbol{z})\big\}\Big](\boldsymbol{v} + \dot{\tilde{\boldsymbol{q}}}_{\boldsymbol{\tau net}})\\
& - K_p\boldsymbol{r} + E + E_d + E_p - \boldsymbol{\tau}_r. \qquad (6.21)
\end{aligned}
$$

Let the robust term $\boldsymbol{\tau}_r$ be defined as

$$
\boldsymbol{\tau_r} \triangleq (K + K_d + K_{\tau p})sgn(r), \qquad (6.22)
$$

where K=$diag(k_{ii})$, $k_{ii} \geq |E_i|$, $i = 1 \ldots n$, K_d=$diag(k_{d,ii})$, $k_{d,ii} \geq |E_d|$, and $K_{\tau p} = diag(k_{\tau p,ii})$, $k_{\tau p,ii} \geq |E_p|$. In other words, the gains K, K_d, and $K_{\tau p}$ include the upper bounds of E_i, E_d, and E_p respectively. Thus

$$
\begin{aligned}
M\dot{\boldsymbol{r}} + C\boldsymbol{r} + K_i\int \boldsymbol{r}dt \;=\; & \Big[\big\{\tilde{W}_M\big\}^T\big\{\Phi_M(\boldsymbol{q})\big\}\Big](\dot{\boldsymbol{v}} + \ddot{\tilde{\boldsymbol{q}}}_{\boldsymbol{\tau net}})\\
& + \Big[\big\{\tilde{W}_C\big\}^T\big\{\Phi_C(\boldsymbol{z})\big\}\Big](\boldsymbol{v} + \dot{\tilde{\boldsymbol{q}}}_{\boldsymbol{\tau net}})\\
& - K_p\boldsymbol{r} + E + E_d + E_p\\
& - (K + K_d + K_{\tau p})sgn(r). \qquad (6.23)
\end{aligned}
$$

To avoid unwanted chatter associated with the $sgn(r)$ term, the following smoothing function was utilized instead: $\frac{r}{(|r|+\epsilon)}$.

6.2.2 Stability Proof

This RBFNN-I stability proof follows a similar Lyapunov function approach as that used for ARII in §5.3.2 and extends the proof described in [73] to incorporate impedance control. Let an integration-type Lyapunov function be defined as

$$\begin{aligned} V \;&= \frac{1}{2}\boldsymbol{r}^T M \boldsymbol{r} \;+\; \frac{1}{2}(\int \boldsymbol{r}dt)^T K_i (\int \boldsymbol{r}dt) \\ &+\frac{1}{2}\sum_{k=1}^{n}\{\tilde{W}_{Mk}\}^T \Gamma_{Mk}^{-1}\{\tilde{W}_{Mk}\} + \frac{1}{2}\sum_{k=1}^{n}\{\tilde{W}_{Ck}\}^T \Gamma_{Ck}^{-1}\{\tilde{W}_{Ck}\}, \end{aligned} \tag{6.24}$$

where K_i, Γ_{Mk}, and Γ_{Ck} are symmetric positive definite constant matrices which can control the speed of adaptation of the weight parameters. Taking the time derivative of V results in

$$\begin{aligned} \dot{V} \;&= \frac{1}{2}\dot{\boldsymbol{r}}^T M \boldsymbol{r} \;+\; \frac{1}{2}\boldsymbol{r}^T \dot{M} \boldsymbol{r} \;+\; \frac{1}{2}\boldsymbol{r}^T M \dot{\boldsymbol{r}} \\ &+\frac{1}{2}\boldsymbol{r}^T K_i (\int \boldsymbol{r}dt) \;+\; \frac{1}{2}(\int \boldsymbol{r}dt)^T K_i \boldsymbol{r} \\ &+\frac{1}{2}\sum_{k=1}^{n}\{\dot{\tilde{W}}_{Mk}\}^T \Gamma_{Mk}^{-1}\{\tilde{W}_{Mk}\} + \frac{1}{2}\sum_{k=1}^{n}\{\tilde{W}_{Mk}\}^T \Gamma_{Mk}^{-1}\{\dot{\tilde{W}}_{Mk}\} \\ &+\frac{1}{2}\sum_{k=1}^{n}\{\dot{\tilde{W}}_{Ck}\}^T \Gamma_{Ck}^{-1}\{\tilde{W}_{Ck}\} + \frac{1}{2}\sum_{k=1}^{n}\{\tilde{W}_{Ck}\}^T \Gamma_{Ck}^{-1}\{\dot{\tilde{W}}_{Ck}\}. \end{aligned} \tag{6.25}$$

Furthermore, since M, $\tilde{W}_{Mk}$, and $\tilde{W}_{Ck}$ are symmetric, (6.25) can be reduced so that

$$\begin{aligned} \dot{V} \;&= \boldsymbol{r}^T M \dot{\boldsymbol{r}} \;+\; \frac{1}{2}\boldsymbol{r}^T \dot{M} \boldsymbol{r} \;+\; \boldsymbol{r}^T K_i (\int \boldsymbol{r}dt) \\ &+\sum_{k=1}^{n}\{\tilde{W}_{Mk}\}^T \Gamma_{Mk}^{-1}\{\dot{\tilde{W}}_{Mk}\} + \sum_{k=1}^{n}\{\tilde{W}_{Ck}\}^T \Gamma_{Ck}^{-1}\{\dot{\tilde{W}}_{Ck}\}. \end{aligned} \tag{6.26}$$

Note that the skew-symmetric property of manipulator dynamics (Property 2 in §4.3.1) gives

$$\frac{1}{2}\boldsymbol{r}^T(\dot{M} - 2C)\boldsymbol{r} = 0. \tag{6.27}$$

Substituting (6.27) in (6.26) results in

$$\begin{aligned}\dot{V} &= \boldsymbol{r}^T M\dot{\boldsymbol{r}} \;+\; \boldsymbol{r}^T C\boldsymbol{r} \;+\; \boldsymbol{r}^T K_i(\int \boldsymbol{r}dt) \\ &\quad + \sum_{k=1}^{n}\{\tilde{W}_{Mk}\}^T\Gamma_{Mk}^{-1}\{\dot{\tilde{W}}_{Mk}\} \;+\; \sum_{k=1}^{n}\{\tilde{W}_{Ck}\}^T\Gamma_{Ck}^{-1}\{\dot{\tilde{W}}_{Ck}\} \\ &= \boldsymbol{r}^T(M\dot{\boldsymbol{r}} + C\boldsymbol{r} + K_i\int \boldsymbol{r}dt) \\ &\quad + \sum_{k=1}^{n}\{\tilde{W}_{Mk}\}^T\Gamma_{Mk}^{-1}\{\dot{\tilde{W}}_{Mk}\} \;+\; \sum_{k=1}^{n}\{\tilde{W}_{Ck}\}^T\Gamma_{Ck}^{-1}\{\dot{\tilde{W}}_{Ck}\}. \qquad (6.28)\end{aligned}$$

Substituting (6.23) into (6.28) yields

$$\begin{aligned}\dot{V} &= \boldsymbol{r}^T\Big(\Big[\{\tilde{W}_M\}^T\{\Phi_M(\boldsymbol{q})\}\Big](\dot{\boldsymbol{v}} + \ddot{\tilde{\boldsymbol{q}}}_{\boldsymbol{\tau net}}) + \Big[\{\tilde{W}_C\}^T\{\Phi_C(\boldsymbol{z})\}\Big](\boldsymbol{v} + \dot{\tilde{\boldsymbol{q}}}_{\boldsymbol{\tau net}}) \\ &\quad - K_p\boldsymbol{r} + E + E_d + E_p - (K + K_d + K_{\tau p})sgn(r)\Big) \\ &\quad + \sum_{k=1}^{n}\{\tilde{W}_{Mk}\}^T\Gamma_{Mk}^{-1}\{\dot{\tilde{W}}_{Mk}\} \;+\; \sum_{k=1}^{n}\{\tilde{W}_{Ck}\}^T\Gamma_{Ck}^{-1}\{\dot{\tilde{W}}_{Ck}\}. \qquad (6.29)\end{aligned}$$

Note that

$$\boldsymbol{r}^T\Big[\{\tilde{W}_M\}^T\{\Phi_M(\boldsymbol{q})\}\Big](\dot{\boldsymbol{v}} + \ddot{\tilde{\boldsymbol{q}}}_{\boldsymbol{\tau net}}) \;=\; \sum_{k=1}^{n}\tilde{W}_{Mk}^T\{\phi_{Mk}(\boldsymbol{q})\}(\dot{\boldsymbol{v}} + \ddot{\tilde{\boldsymbol{q}}}_{\boldsymbol{\tau net}})\boldsymbol{r}_k, \qquad (6.30)$$

and

$$\boldsymbol{r}^T\Big[\{\tilde{W}_C\}^T\{\Phi_C(\boldsymbol{z})\}\Big](\boldsymbol{v} + \dot{\tilde{\boldsymbol{q}}}_{\boldsymbol{\tau net}}) \;=\; \sum_{k=1}^{n}\tilde{W}_{Ck}\{\phi_{Ck}(\boldsymbol{z})\}(\boldsymbol{v} + \dot{\tilde{\boldsymbol{q}}}_{\boldsymbol{\tau net}})\boldsymbol{r}_k. \qquad (6.31)$$

Substitute (6.30) and (6.31) into (6.29) and make the following equalities

$$\sum_{k=1}^{n}\tilde{W}_{Mk}^T\{\phi_{Mk}(\boldsymbol{q})\}(\dot{\boldsymbol{v}} + \ddot{\tilde{\boldsymbol{q}}}_{\boldsymbol{\tau net}})\boldsymbol{r}_k \triangleq -\sum_{k=1}^{n}\tilde{W}_{Mk}^T\Gamma_{Mk}^{-1}\dot{\tilde{W}}_{Mk}, \qquad (6.32)$$

and

$$\sum_{k=1}^{n}\tilde{W}_{Ck}^T\{\phi_{Ck}(\boldsymbol{z})\}(\boldsymbol{v} + \dot{\tilde{\boldsymbol{q}}}_{\boldsymbol{\tau net}})\boldsymbol{r}_k \triangleq -\sum_{k=1}^{n}\tilde{W}_{Ck}^T\Gamma_{Ck}^{-1}\dot{\tilde{W}}_{Ck}. \qquad (6.33)$$

Then

$$\begin{aligned}\dot{\tilde{W}}_{Mk} &= -\Gamma_{Mk}(\tilde{W}_{Mk}^T)^{-1}\tilde{W}_{Mk}^T\{\phi_{Mk}(\boldsymbol{q})\}(\dot{\boldsymbol{v}} + \ddot{\tilde{\boldsymbol{q}}}_{\boldsymbol{\tau net}})\boldsymbol{r}_k \\ &= -\Gamma_{Mk}\{\phi_{Mk}(\boldsymbol{q})\}(\dot{\boldsymbol{v}} + \ddot{\tilde{\boldsymbol{q}}}_{\boldsymbol{\tau net}})\boldsymbol{r}_k, \qquad (6.34)\end{aligned}$$

with $\dot{\hat{W}}_{Mk} = \dot{W}_{Mk} - \dot{\hat{W}}_{Mk}$ and $\dot{W}_{Mk}$ equal to the zero vector, then

$$\dot{\hat{W}}_{Mk} \;=\; \Gamma_{Mk}\big\{\phi_{Mk}(\boldsymbol{q})\big\}(\dot{\boldsymbol{v}} + \ddot{\tilde{\boldsymbol{q}}}_{\boldsymbol{\tau net}})\boldsymbol{r}_k, \tag{6.35}$$

and similarly

$$\dot{\hat{W}}_{Ck} \;=\; \Gamma_{Ck}\big\{\phi_{Ck}(\boldsymbol{z})\big\}(\boldsymbol{v} + \dot{\tilde{\boldsymbol{q}}}_{\boldsymbol{\tau net}})\boldsymbol{r}_k, \tag{6.36}$$

where (6.35) and (6.36) are used to adjust the RBFNN-I weights and Γ_{Mk} and Γ_{Ck} can be used to control the speed at which the weights are adjusted. With (6.32) and (6.33), (6.29) becomes

$$\begin{aligned}
\dot{V} =& -\boldsymbol{r}^T K_p \boldsymbol{r} + (-\boldsymbol{r}^T K sgn(\boldsymbol{r}) + \boldsymbol{r}^T E) + (-\boldsymbol{r}^T K_d sgn(\boldsymbol{r}) + \boldsymbol{r}^T E_d) \\
&+ (-\boldsymbol{r}^T K_{\tau p} sgn(\boldsymbol{r}) + \boldsymbol{r}^T E_p) \\
=& -\boldsymbol{r}^T K_p \boldsymbol{r} + (-\sum_{k=1}^{n} K_k |r_k| + \sum_{k=1}^{n} E_k r_k) + (-\sum_{k=1}^{n} K_{d,k} |r_k| + \sum_{k=1}^{n} E_{d,k} r_k) \\
&+ (-\sum_{k=1}^{n} K_{\tau p,k} |r_k| + \sum_{k=1}^{n} E_{p,k} r_k).
\end{aligned} \tag{6.37}$$

Since $-\boldsymbol{r}^T K_p \boldsymbol{r} \leq 0$ and

$$\begin{aligned}
\sum_{k=1}^{n} K_k |r_k| &\geq \sum_{k=1}^{n} E_k r_k, \\
\sum_{k=1}^{n} K_{d,k} |r_k| &\geq \sum_{k=1}^{n} E_{d,k} r_k, \\
\sum_{k=1}^{n} K_{\tau p,k} |r_k| &\geq \sum_{k=1}^{n} E_{p,k} r_k.
\end{aligned}$$

the derivative of V is then,

$$\dot{V} = -\boldsymbol{r}^T K_p \boldsymbol{r} \leq 0. \tag{6.38}$$

Since $\boldsymbol{r} \in \mathcal{L}_2 \cap \mathcal{L}_\infty$ and $\boldsymbol{r} = \dot{\tilde{\boldsymbol{\xi}}} + \Lambda \tilde{\boldsymbol{\xi}}$ from (6.10), then $\tilde{\boldsymbol{\xi}}$ and $\dot{\tilde{\boldsymbol{\xi}}} \in \mathcal{L}_2^n \cap \mathcal{L}_\infty^n$. According to Barbălat's Lemma as done in §5.3.2

$$\begin{aligned}
\text{if} \quad & \tilde{\boldsymbol{\xi}} \in \mathcal{L}_2^n \cap \mathcal{L}_\infty^n \quad \text{and} \quad \dot{\tilde{\boldsymbol{\xi}}} \in \mathcal{L}^n, \\
& \text{then} \quad \lim_{t\to\infty} \tilde{\boldsymbol{\xi}}(t) = 0,
\end{aligned} \tag{6.39}$$

as desired, which completes the proof.

6.3 RBFNN-I Method Validation with the Single-System 1-DOF Quanser QUBE Robot Testbed

Following a similar validation approach as the ARII control method, the RBFNN-I control method was first validated on a single 1-DOF system in simulation on the identified model of the Quanser QUBE described in §4.4.1. The RBFNN-I control method was then validated experimentally using the Quanser QUBE experimental testbed. The ability of the RBFNN-I method to control a multi-DOF system was then explored using a 2-DOF robotic exoskeleton simulator, after which RBFNN-I telerehabilitation simulations and experiments were studied.

6.3.1 Simulations Validation with the Quanser QUBE Robot

The same desired trajectory used to validate the ARII method was used to validate the RBFNN-I method on the Quanser QUBE single-system 1-DOF simulator. The simulator had the same initial conditions, impedance model, patient-related parameters, and motor and torque saturation blocks as those used for the ARII method in §5.4.1, which enhance system stability by implementing the time domain passivity based strategy of boundedness. Table 6.1 summarizes the impedance model and patient parameters as well as the RBFNN-I controller parameters which were determined using a systematic and extensive manual tuning process that was carried out to achieve desirable and converging tracking errors.

The results of utilizing the RBFNN-I controller with the single-system Quanser QUBE simulator testbed are shown in Fig. 6.3. Fig. 6.3(a) plots the desired joint angular position, the impedance desired joint angular position, and the actual joint angular position as a function of time.

Fig. 6.3(b) shows the corresponding impedance joint angular position tracking error $\tilde{\xi}$ plotted as a function of time which decreases with time as the neural network adjust its weights.

Fig. 6.3(c) plots the patient torque as a function of time, while Fig. 6.4 depicts how the M_{NN} and C_{NN} terms in the RBFNN-I method change with time and are held constant after the learning period of 48 seconds. The corresponding performance metrics for the RBFNN-I simulation validation test are listed in Table 6.2. The maximum $\tilde{\xi}$

Table 6.1: RBFNN-I QUBE Simulation Parameters

Impedance:	J	0.013 Kgm^2
	B	0.667 Nms/rad
	K	2.67 Nm/rad
Patient:	K_p	0.848 Nm/rad
	D_p	0.2 Nms/rad
Controller:	Λ	5.0
	K_p	1.0
	K_i	0.1
	K_r	0.01
	ϵ	0.08
	nodes	5
	Gaussian width	20
	Γ_M	diag(0.05,0.05,0.05,0.05,0.05)
	Γ_C	diag(0.06,0.06,0.06,0.06,0.06)
	M Gaussian function	[-1.0 -0.5 0.0 0.5 1.0]
	C Gaussian function	[-1.0 -0.5 0.0 0.5 1.0 -1.0 -0.5 0.0 0.5 1.0]

was recorded as 2.1° while the $\tilde{\xi}_{RMSE}$ had a value of 0.7°. For reference, Table 6.2 lists the performance metrics for the ARII controller from §5.4.1 when this method was tested on the same simulator. Furthermore, the ARII simulation impedance tracking error $\tilde{\xi}$ results were superimposed on the RBFNN-I simulation $\tilde{\xi}$ results as shown in Fig. 6.5. It can be seen in Table 6.2 and Fig. 6.5 that the RBFNN-I performance is very similar to that of the ARII method when applied to the single-system the Quanser QUBE simulator testbed.

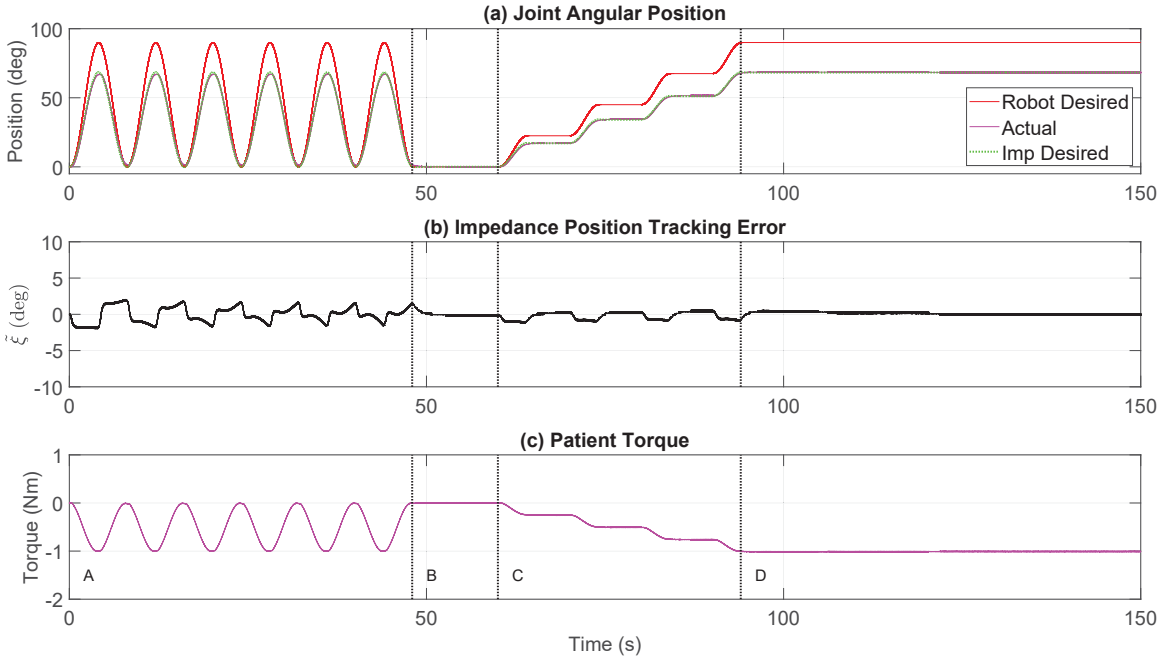


Figure 6.3: RBFNN-I single-system simulation with Quanser QUBE: (a) position trajectory, (b) tracking error, and (c) patient torque

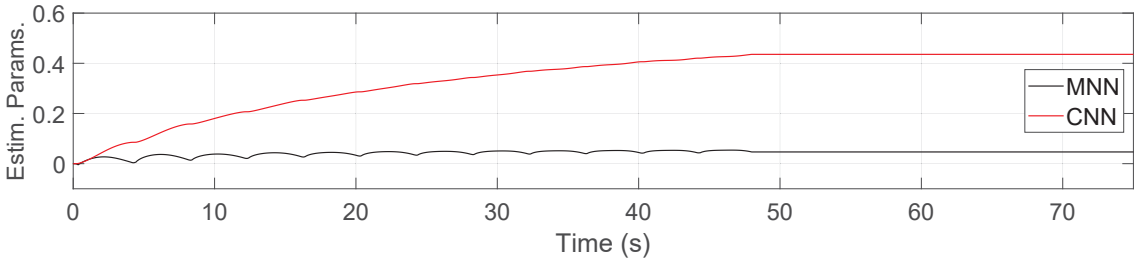


Figure 6.4: Estimated parameters using the RBFNN-I control method with the Quanser QUBE single-system simulator

Table 6.2: Performance Metrics for ARII and RBFNN-I Single-System Simulation with Quanser QUBE

Testbed	**Metrics**	**ARII**	**RBFNN-I**
QUBE simulation	$\tilde{\xi}_{max}$	2.4°	2.1°
	$\tilde{\xi}_{RMSE}$	0.7°	0.7°

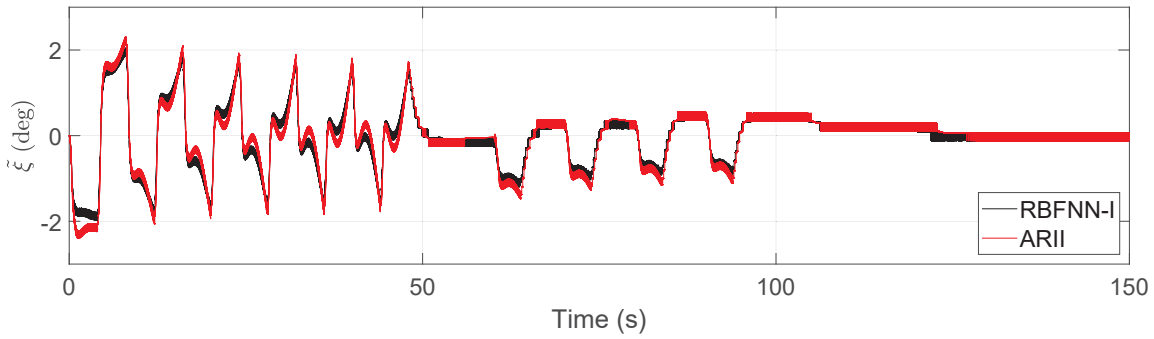


Figure 6.5: The profile of $\tilde{\xi}$ for ARII versus RBFNN-I Quanser QUBE simulation

6.3.2 Experimental Validation with the Quanser QUBE Robot

Having validated the RBFNN-I control approach on a single 1-DOF Quanser QUBE simulator, the RBFNN-I method was then validated on the single-system experimental Quanser QUBE apparatus, following a similar validation approach as ARII. The same desired trajectory and controller parameters used in the previous RBFNN-I simulation test was used for experimental tests with the Quanser QUBE apparatus. Fig. 6.6 shows the resulting experimental results using the RBFNN-I controller with the single-system Quanser QUBE hardware. Similar to Fig. 6.3, Fig. 6.6(a) plots the desired joint angular position, the impedance desired joint angular position, and the actual joint angular position, as functions of time.

Fig. 6.6(b) shows the impedance joint angular position tracking error $\tilde{\xi}$, while Fig. 6.6(c) shows the patient torques applied during the test. Fig. 6.7 shows the neural network terms for the Quanser QUBE RBFNN-I experimental test which, after the 48 seconds learning period, were set to their convergent values for the remainder of the test. The RBFNN-I performance metrics for this experiment are listed in Table 6.3. The maximum $\tilde{\xi}$ was recorded as 2.5°, and the $\tilde{\xi}_{RMSE}$ had a value of 0.8°.

For comparison purposes, Table 6.3 also lists the performance metrics for the ARII controller from §5.4.2 when this method was experimentally tested on the same Quanser QUBE apparatus. Additionally, the ARII experimental impedance tracking error $\tilde{\xi}$ results were superimposed on the RBFNN-I experimental $\tilde{\xi}$ results as shown in Fig. 6.8. It can be seen in Table 6.3 and Fig. 6.8 that the RBFNN-I performance is very similar to that of the ARII method when applied to the single-system Quanser QUBE experimental testbed.

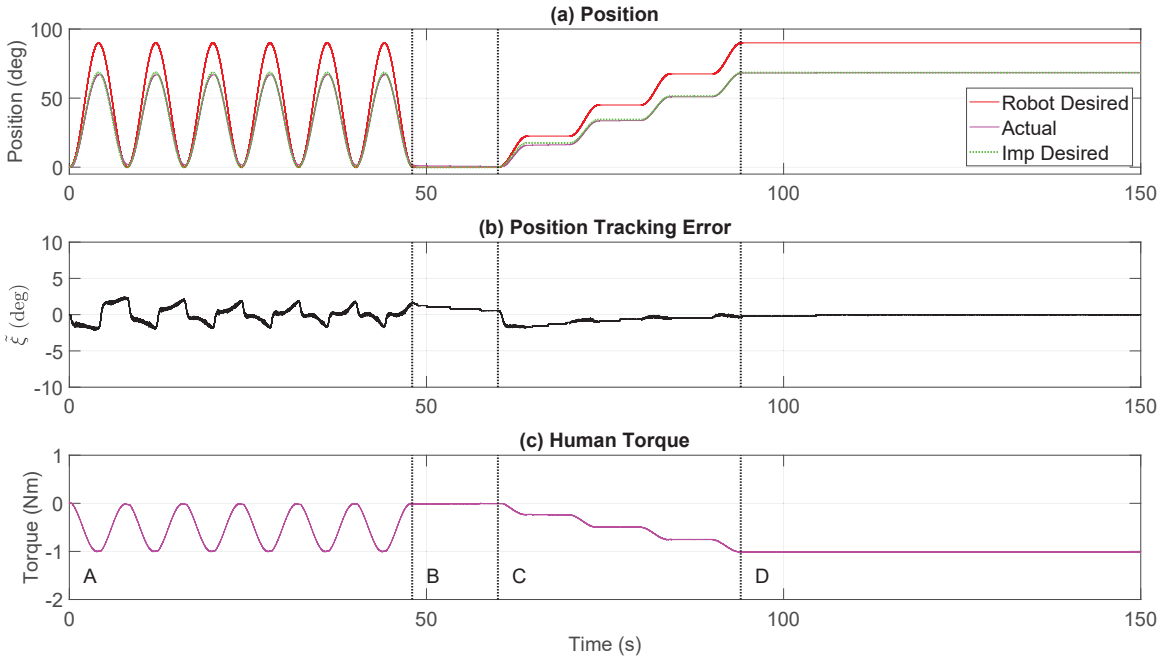


Figure 6.6: RBFNN-I single-system experiment with Quanser QUBE: (a) position trajectory, (b) tracking error, and (c) patient torque

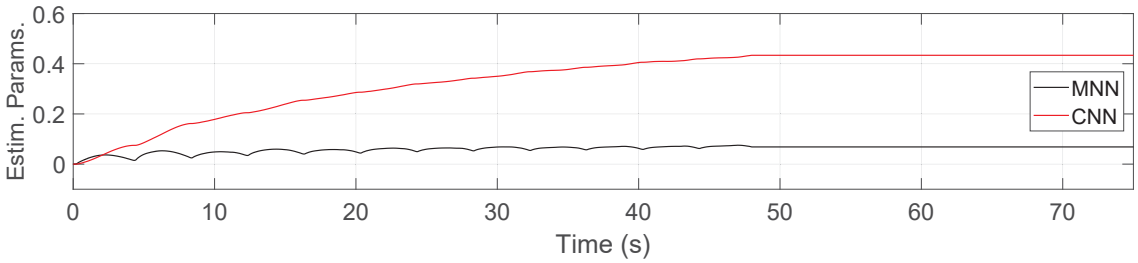


Figure 6.7: Estimated parameters using the RBFNN-I control method with the Quanser QUBE single-system apparatus

Table 6.3: Performance Metrics for ARII and RBFNN-I Single-System Experiment with Quanser QUBE

Testbed	**Metrics**	**ARII**	**RBFNN-I**
QUBE experimental setup	$\tilde{\xi}_{max}$	2.8°	2.5°
	$\tilde{\xi}_{RMSE}$	1.0°	0.8°

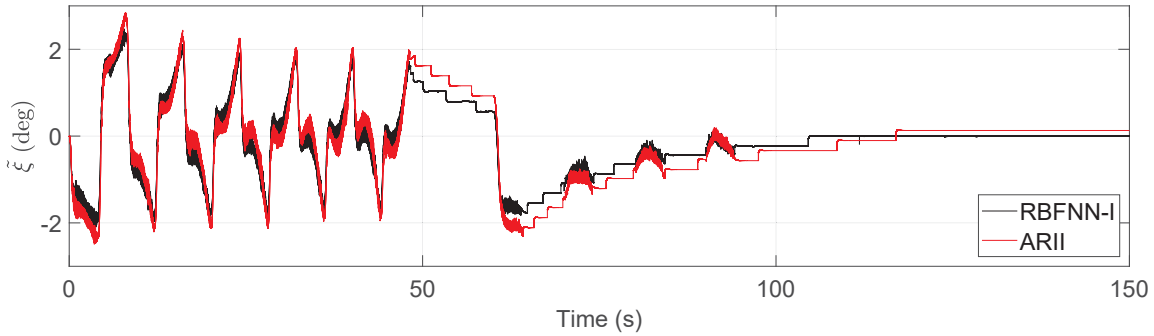


Figure 6.8: The profile of $\tilde{\xi}$ for the ARII and the RBFNN-I Quanser QUBE experiments

Furthermore, the RBFNN-I experimental impedance tracking error $\tilde{\xi}$ results were superimposed on the RBFNN-I simulation $\tilde{\xi}$ results from the previous section as shown in Fig. 6.9. The corresponding performance metrics for the simulated and the experimental tests are listed side-by-side in Table 6.4, for comparison purposes. As can be seen in Fig. 6.9 and Table 6.4, the experiment results are very similar to the simulator results in terms of bound, performance during the oscillatory motion, as well as final holding position and, therefore, effectively validate the simulation results.

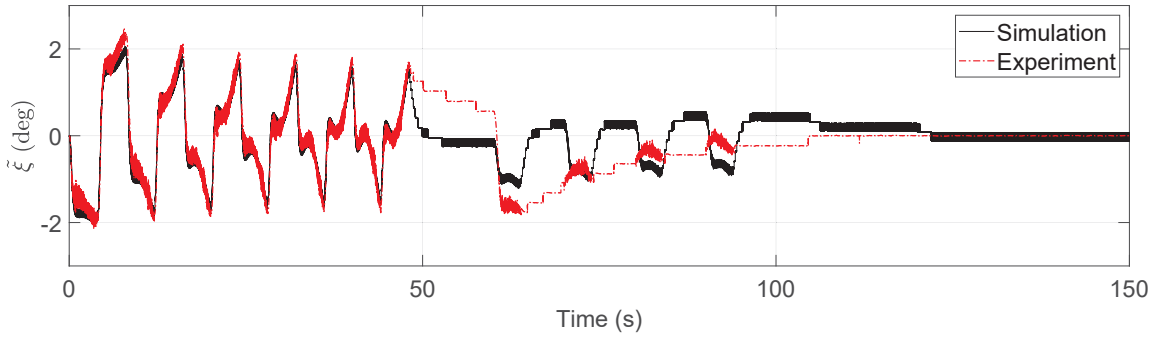


Figure 6.9: The profile of $\tilde{\xi}$ for the RBFNN-I Quanser QUBE simulation and experiment

Table 6.4: Performance Metrics for RBFNN-I Single-System Simulation versus Experiment with Quanser QUBE

Metrics	Simulation	Experiment
$\tilde{\xi}_{max}$	2.1°	2.5°
$\tilde{\xi}_{RMSE}$	0.7°	0.8°

6.3.3 RBFNN-I Method Validation with 2-DOF Robotic Exoskeleton Simulations

Having validated the RBFNN-I simulation results experimentally on a single-system 1-DOF robotic system, the simulator was extended to a single-system 2-DOF robotic exoskeleton involving both elbow and wrist joints to validate the proposed RBFNN-I control methodology on a higher-degree-of-freedom system − similar to the validation approach carried out in §5.5 for the ARII control method. The desired trajectory and the initial positions for the exoskeleton elbow and wrist joints were the same as for the 2-DOF simulations with the ARII proposed control method.

Table 6.5 summarizes the RBFNN-I 2-DOF Exoskeleton simulation parameters, as well as the patient- and impedance-related parameters. The controller parameters were determined through a systematic and extensive manual tuning process to produce desirable and converging tracking errors. Saturation blocks on the control signal were applied to match the motor limits of the conceptualized 2-DOF exoskeleton system described in §5.5, which enhance system stability by implementing the time domain passivity based strategy of boundedness. Figs. 6.10 to Fig. 6.12 show the resulting performance of the RBFNN-I controller on the single-system 2-DOF exoskeleton simulator. Plots on Figs. 6.10 and 6.11 show the desired joint angular position, the impedance desired joint angular position, and the actual joint angular position of joint 1 and joint 2, respectively, plotted as a function of time. Plots (b) on both figures show the impedance joint angular position error $\tilde{\boldsymbol{\xi}}$, while Fig. 6.10(c) and Fig. 6.11(c) reveal the patient torque applied during the tests. Fig. 6.12 shows the adaptation of the M_{NN} and C_{NN} neural network terms, and how, with the selected adaptive law Γ parameters, the neural network terms move towards and oscillate close to their steady state values which are set after the 48 seconds learning period. The corresponding RBFNN-I performance metrics for this test are listed in Table 6.6. For

Table 6.5: RBFNN-I Single-System Simulation with 2-DOF Exoskeleton: Simulation Parameters

Impedance:	J	diag([0.013, 0.013]) Kgm^2
	B	diag([0.667, 0.667]) Nms
	K	diag([2.667, 2.667]) Nm
Patient:	K_p	diag([.848,.424])
	D_p	diag([0.2,.05])
pHMI:	K_{pHMI}	diag([800, 266.667])N/rad
	D_{pHMI}	diag([10, 2.5])Ns/rad
Controller:	Λ	diag([1.0,5.0])
	K_p	diag([12.0,1.0])
	K_i	diag([0.1,0.1])
	K_r	diag([0.01,0.01])
	ϵ	$[0.08,0.08]^T$
	nodes	5
	Gaussian width	20.0
	Γ_M	diag(1.0,1.0,1.0,1.0,1.0)
	Γ_C	diag(0.4,0.4,0.4,0.4,0.4,)
	M Gaussian function	[-1.0 -0.5 0.0 0.5 1.0 -1.0 -0.5 0.0 0.5 1.0]
	C Gaussian function	[-1.0 -0.5 0.0 0.5 1.0 -1.0 -0.5 0.0 0.5 1.0 -1.0 -0.5 0.0 0.5 1.0 -1.0 -0.5 0.0 0.5 1.0]

joint 1, the maximum $\tilde{\xi}$ was recorded as 0.3°, and the $\tilde{\xi}_{RMSE}$ had a value of 0.1°. For joint 2, the maximum $\tilde{\xi}$ was recorded as 0.1°, while the $\tilde{\xi}_{RMSE}$ had a value of 0.0°.

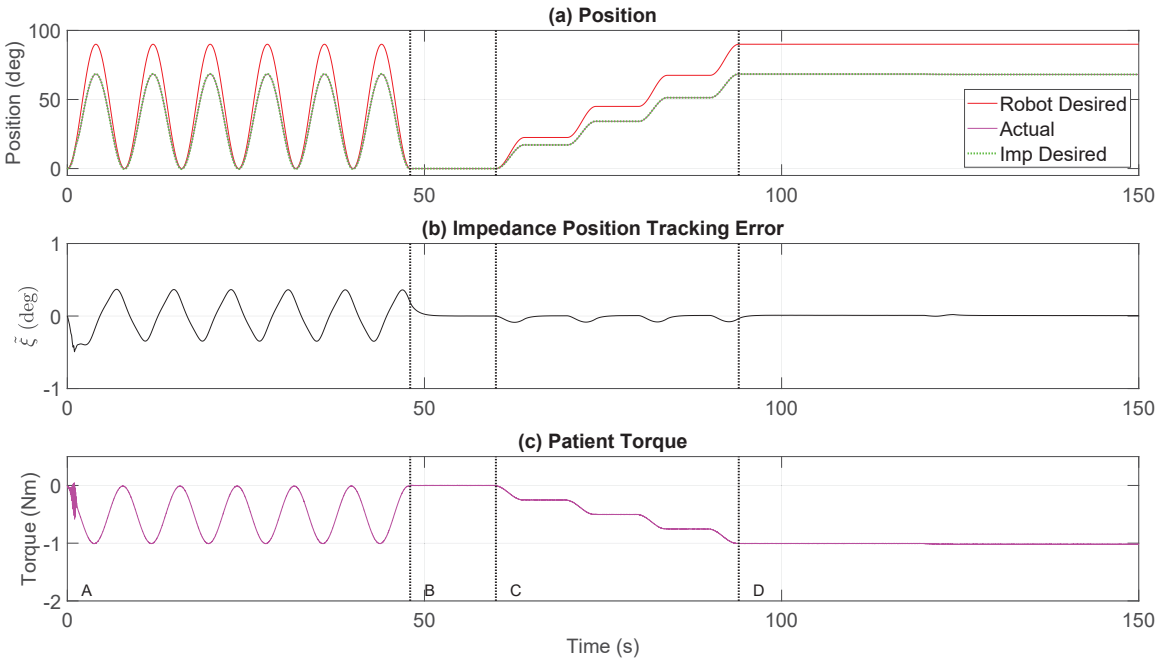


Figure 6.10: RBFNN-I single-system 2-DOF Simulation: (a) position trajectory, (b) tracking error, and (c) patient torque for joint 1

Table 6.6: Performance Metrics for ARII and RBFNN-I 2-DOF Exoskeleton Single-System Simulation

Actuator	Metrics	ARII	RBFNN-I
Joint 1	$\tilde{\xi}_{max}$	0.5°	0.3°
	$\tilde{\xi}_{RMSE}$	0.1°	0.1°
Joint 2	$\tilde{\xi}_{max}$	0.1°	0.1°
	$\tilde{\xi}_{RMSE}$	0.0°	0.0°

For comparison purposes, Table 6.6 additionally lists the performance metrics for the ARII controller from §5.5 when this method was tested on the same 2-DOF robotic exoskeleton single-system testbed. The ARII simulation impedance tracking error $\tilde{\xi}$ results were also superimposed on the RBFNN-I simulation $\tilde{\xi}$ results as shown in Figs. 6.13 and 6.14, respectively. The performance of the two controllers were very

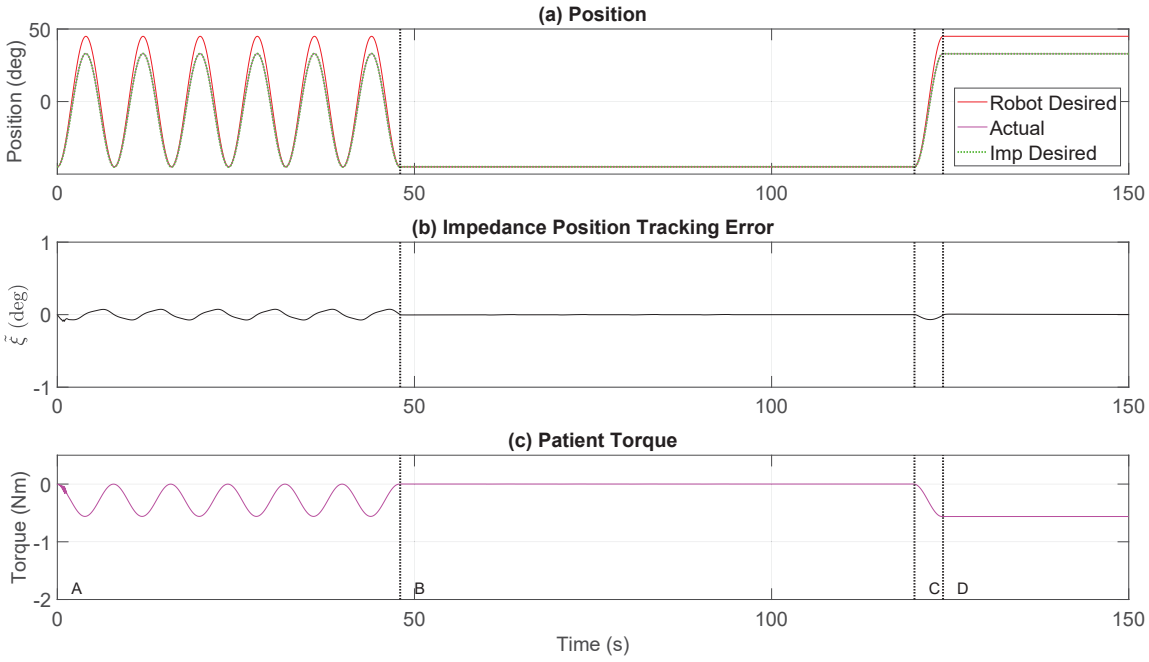


Figure 6.11: RBFNN-I single-system 2-DOF simulation: (a) position trajectory, (b) tracking error, and (c) patient torque for joint 2

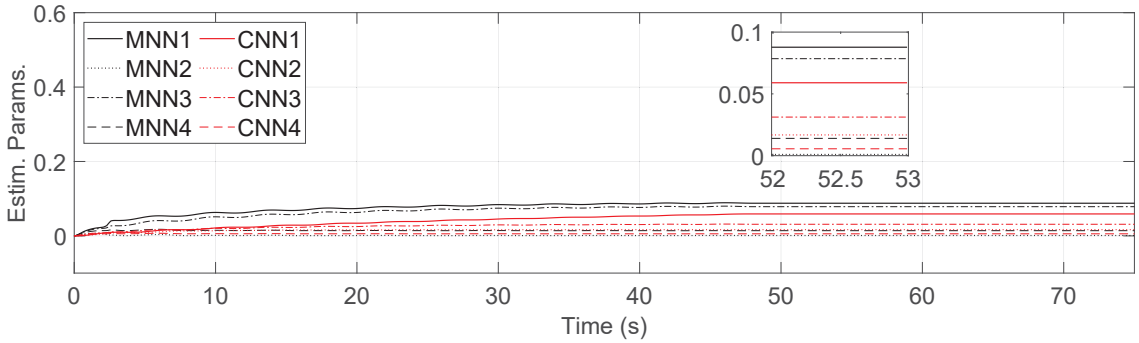


Figure 6.12: Estimated parameters for the RBFNN-I single-system 2-DOF exoskeleton aimulation

similar, with the RBFNN-I performing slightly better with a difference of 0.2° for joint 1 in $\tilde{\xi}_{max}$ and a difference within 0.1° for joint 2 in $\tilde{\xi}_{max}$, when applied to the single-system 2-DOF simulator testbed.

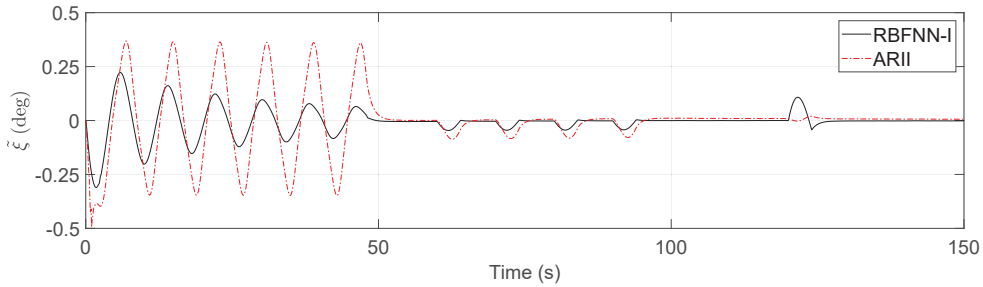


Figure 6.13: The profile of $\tilde{\xi}$ for the ARII and the RBFNN-I 2-DOF exoskeleton simulation joint 1

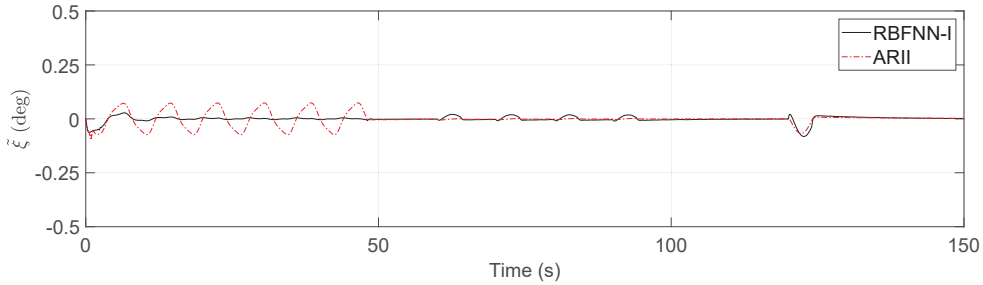


Figure 6.14: The profile of $\tilde{\xi}$ for the ARII and the RBFNN-I 2-DOF exoskeleton simulation Joint 2

6.4 Telerehabilitation Human Torque Synthesis Experiments with Quanser QUBEs

Having validated the proposed RBFNN-I control approach on single systems in both simulation and experiment, experimental tests were carried out with two Quanser QUBE experimental testbeds to validate the performance of the proposed RBFNN-I method in telerehabilitation scenarios and to examine the tracking performance and telepresence fidelity. Similar to the experimental tests performed to validate the proposed ARII control method, the Logitech driving wheel hardware was used to emulate human real-time assistive intervention from the therapist on the slave side. The same motion trajectories were used as with the ARII method to examine the effects of different strategies of combining the human torques (to calculate $\boldsymbol{\tau}_{net}$) on the resulting trajectories and telepresence.

6.4.1 Direct τ_{refl} Addition

Similar to the experiments performed with the proposed ARII control method, Fig. 6.15(a) plots the RBFNN-I experimental position signals as functions of time, while Fig. 6.15(b) shows the RBFNN-I experimental impedance angular position tracking error ξ when τ_{net} is directly added τ_p to calculate τ_{net} as described in (5.52). Fig. 6.15(c) plots the patient torque τ_p in fuchsia and the therapist torque τ_{refl} in black. Fig. 6.15(c) also plots the negative value of the therapist torque ($-\tau_{refl}$) as a dashed black line for comparison, and the net human torque τ_{refl} is shown in green. It can be seen in 6.15(b) that the impedance angular position tracking error improves during the RBFNN-I learning period corresponding to region A. Fig. 6.16 shows how the neural network parameters adapt and converge over time during region A, after which the parameters are held constant for the remainder of the test. Fig. 6.17 focuses on regions E and F and plots the patient torque τ_p and therapist torque τ_{refl} in Fig. 6.17(a) as well as the sum of the two human torques (τ_p+τ_{refl}) in Fig. 6.17(b).

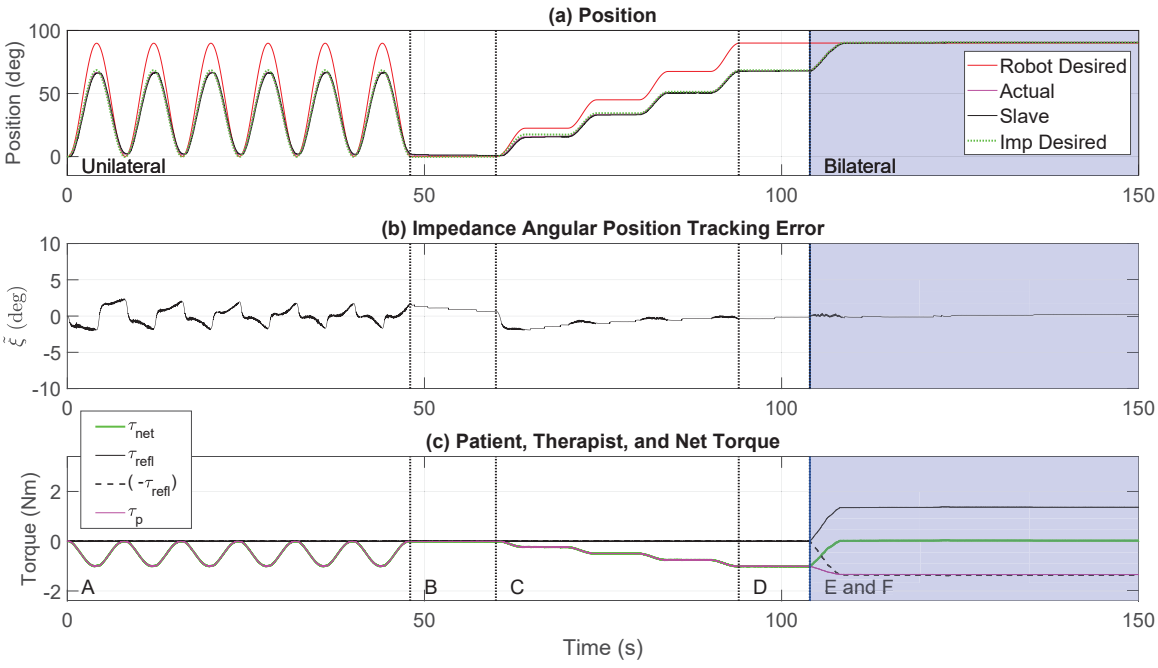


Figure 6.15: RBFNN-I telerehabilitation experiment with Quanser QUBEs with $\alpha = 1$: (a) position trajectory, (b) tracking error, and (c) torques

Note that with the direct addition human torque synthesis strategy, the initial telepresence fidelity is low since the master robot starts to move as soon as the therapist

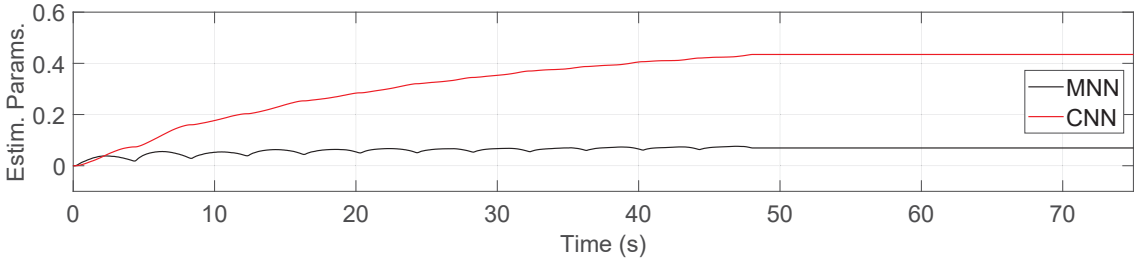


Figure 6.16: Estimated parameters for the RBFNN-I telerehabilitation experiment with Quanser QUBEs with $\alpha = 1$

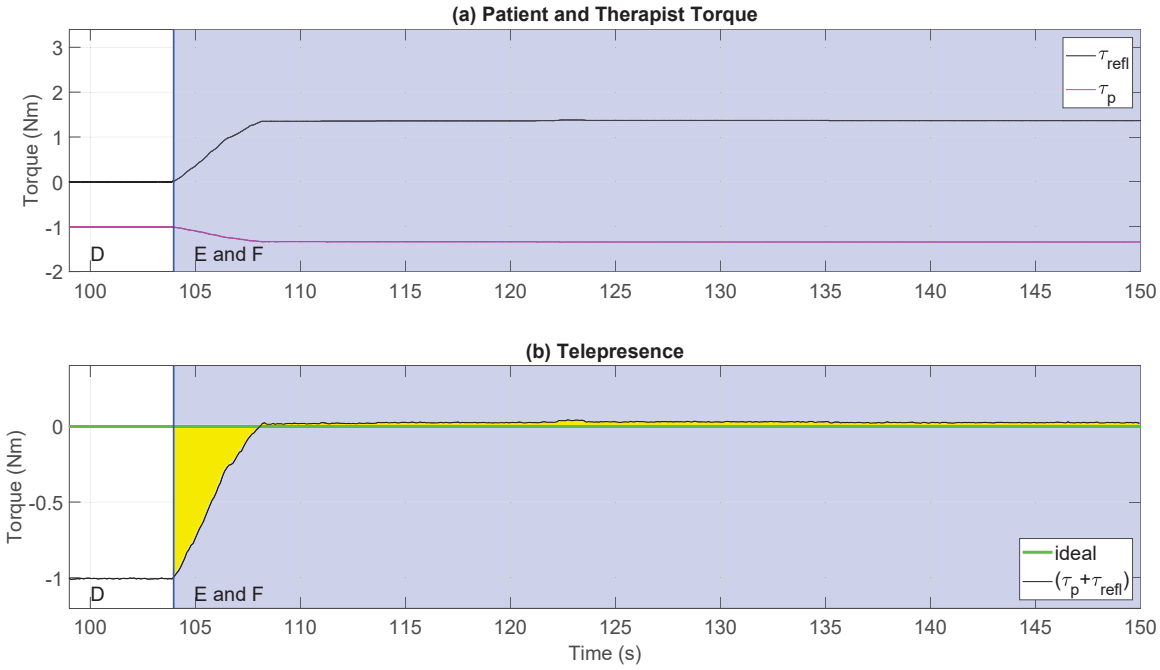


Figure 6.17: Torques for the RBFNN-I telerehabilitation experiment with Quanser QUBEs with $\alpha = 1$ during the bilateral phase

starts to provide small assistive torques, similar to what was observed with the ARII method. This initially poor telepresence can be seen in Fig. 6.17(b) as the relatively large yellow area at the beginning of regions E and F. The therapist's torque then gradually ramps up over the course of about six seconds, and ultimately becomes close to the patient's torque as desired so that $(\tau_p + \tau_{refl})$ is close to the ideal value of zero. The RBFNN-I position tracking and telepresence performance metrics for this $\alpha = 1$ experiment are listed in Table 6.7.

6.4.2 Scaled $\boldsymbol{\tau}_{refl}$ Addition

The scaled $\boldsymbol{\tau}_{refl}$ addition strategy was experimentally implemented with the proposed RBFNN-I proposed control method, similar to how it was applied with the ARII control method using (5.53) in §5.6.2. The results are shown in Figs. 6.18 to 6.20 when the scaling factors in α are equal to three. Similar to the direct $\boldsymbol{\tau}_{refl}$-addition synthesis method described in the previous section, Fig. 6.18 follows a similar format to Fig. 6.15, while Fig. 6.19 shows the neural network parameters learning as a function of time and then being held constant from 48 seconds to the end of the test. Similar to Fig. 6.17, Fig. 6.20 focuses on the torques in the bilateral telerehabilitation-related regions E and F. As can be seen in Fig. 6.20(b), the achieved telepresence hovers around slightly less than -1 Nm when using the scaled therapist's torque, which is consistent with the poor telepresence observed when this scaled τ_{refl} was used with ARII.

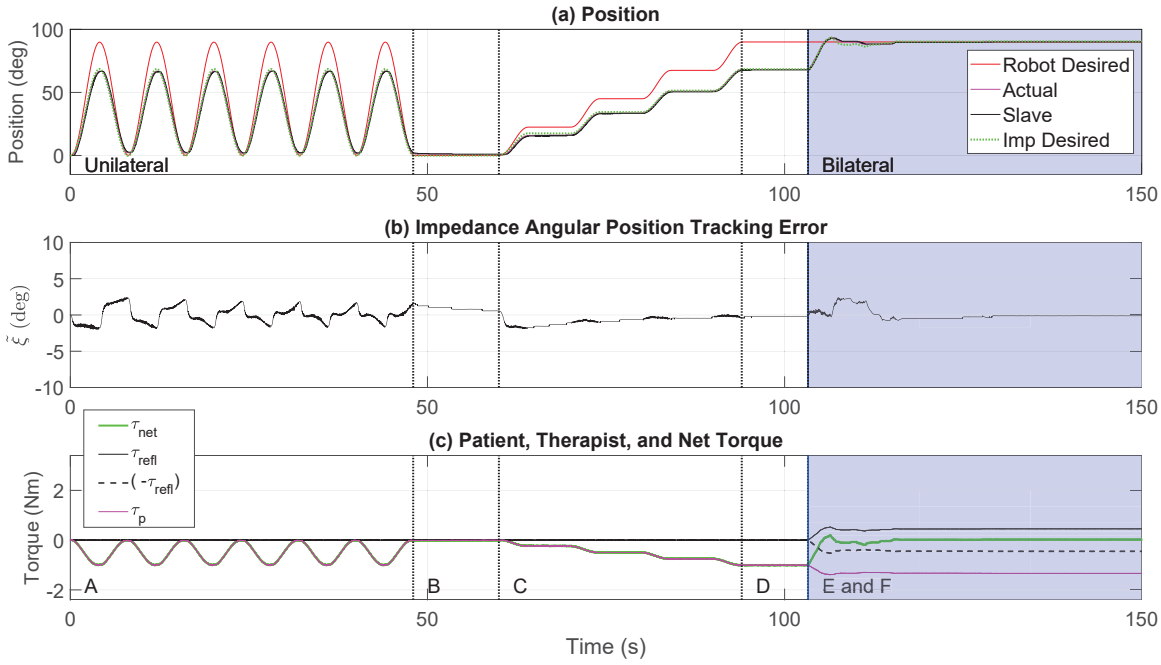


Figure 6.18: RBFNN-I telerehabilitation experiment with Quanser QUBEs with $\alpha = 3$: (a) position trajectory, (b) tracking error, and (c) torques

The relatively large yellow area in Fig. 6.20(b) highlights the poor telepresence achieved when using the $\alpha = 3$ scaled τ_{refl} strategy governed by (5.53), rendering the telepresence virtually non-existent with this strategy. The corresponding RBFNN-I position tracking and telepresence performance metrics for this $\alpha = 3$ experiment are listed in Table 6.7.

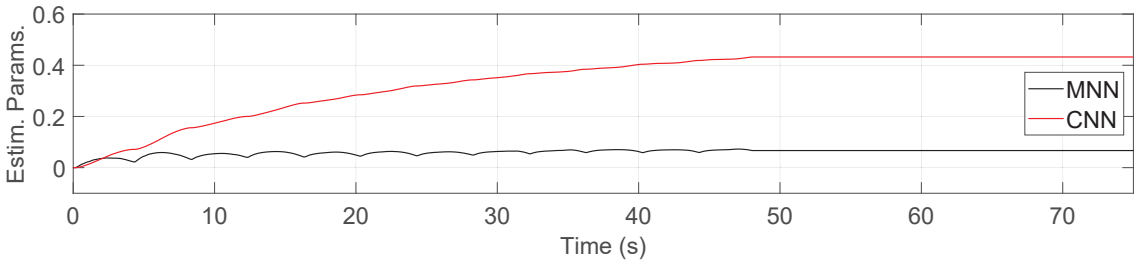


Figure 6.19: Estimated parameters for the RBFNN-I telerehabilitation experiment with Quanser QUBEs with $\alpha = 3$

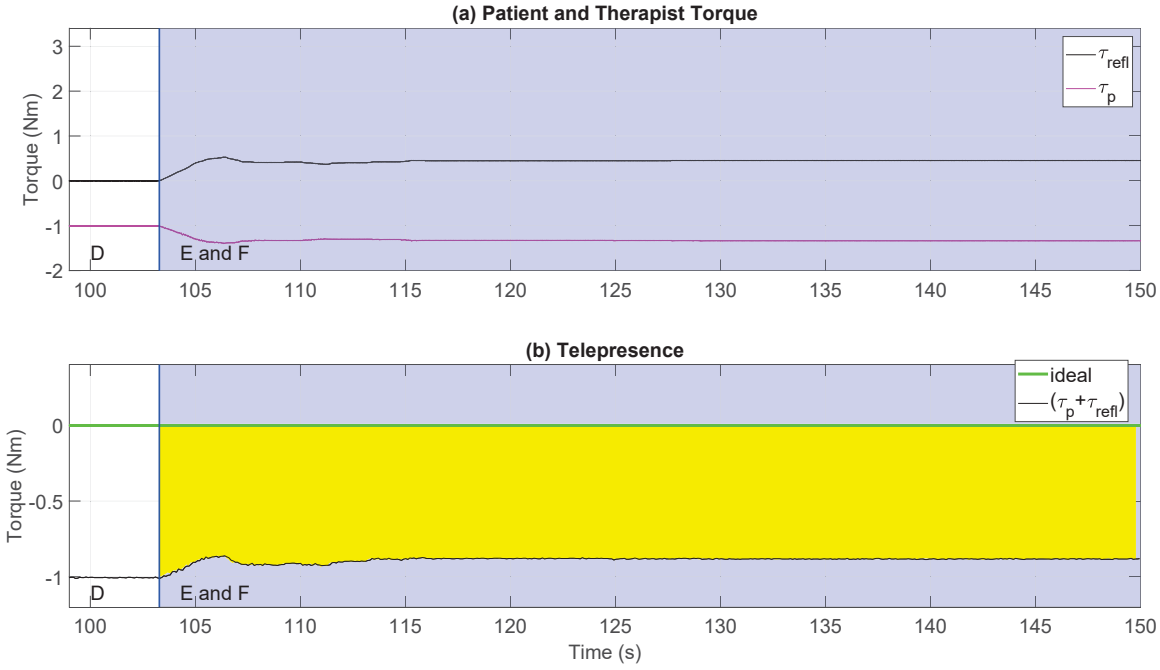


Figure 6.20: Torques for the RBFNN-I telerehabilitation experiment with Quanser QUBEs with $\alpha = 3$ during the bilateral phase

6.4.3 Human Torque Regulator

Figure 6.21 plots the experimental RBFNN-I results when τ_{net} is calculated using the novel Human Torque Regulator (HTR) governed by (5.54) which was proposed in §5.6.3. Fig. 6.21 follows a similar format as Fig. 6.15 and Fig. 6.18, while Fig. 6.23 follows the format shown in Fig. 6.17 and Fig. 6.20. Figure 6.22 shows the convergent neural network parameters and their terminal, constant values from 48 seconds to the end of the test. The trends observed in Figs. 6.21 to 6.23 are similar to those observed when using this proposed HTR approach with ARII in Figs. 5.18 to 5.20.

Table 6.8 additionally lists the performance metrics for the ARII controller from §5.6 when this method was experimentally tested on the same Quanser QUBE testbeds. It can be seen that the RBFNN-I performance is very similar to that of the ARII method.

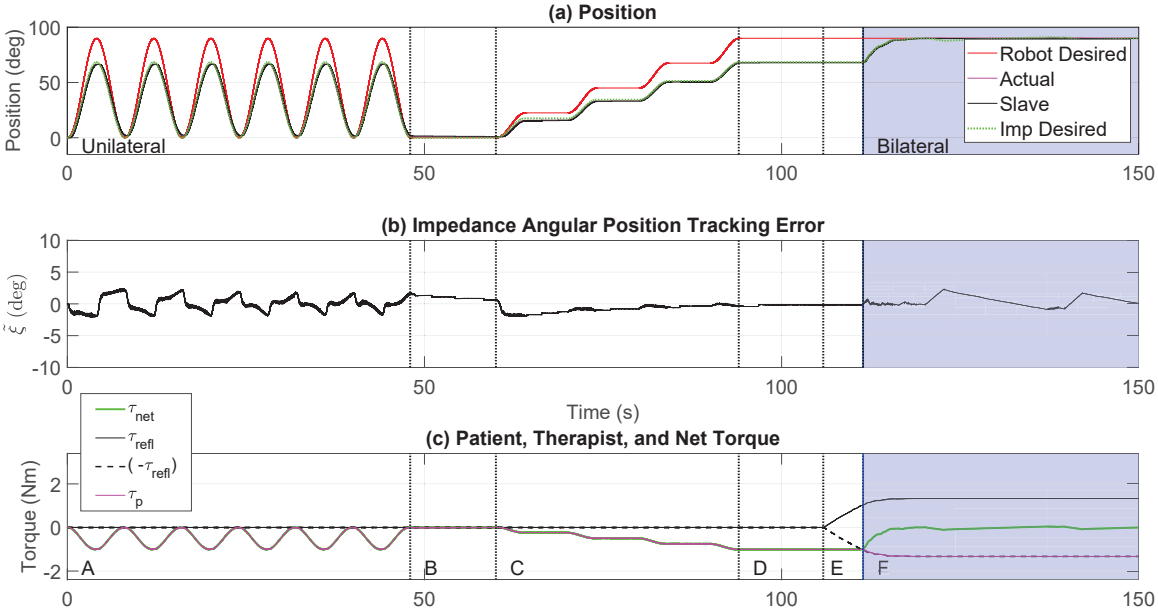


Figure 6.21: RBFNN-I telerehabilitation experiment with Quanser QUBEs with HTR: (a) position trajectory, (b) tracking error, and (c) torques

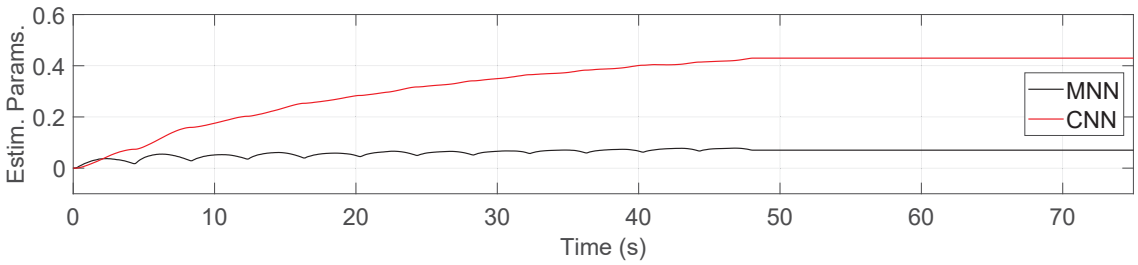


Figure 6.22: Estimated parameters for the RBFNN-I telerehabilitation experiment with Quanser QUBEs with HTR

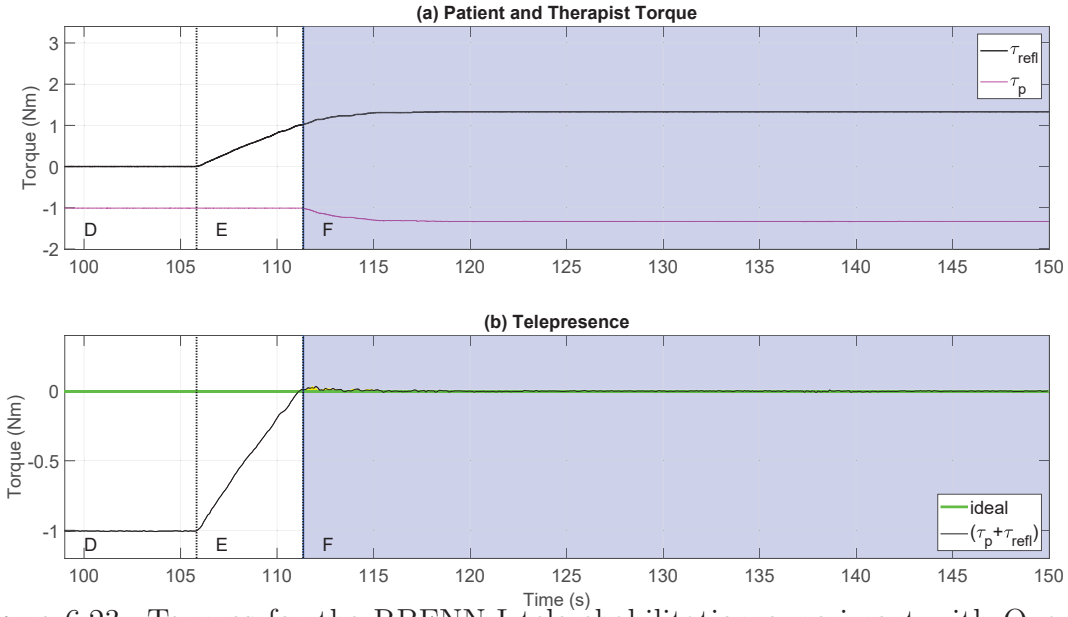


Figure 6.23: Torques for the RBFNN-I telerehabilitation experiment with Quanser QUBEs with HTR during the bilateral phase

Table 6.7: Performance Metrics for RBFNN-I Telerehabilitation Experiments with Quanser QUBEs

Metrics	$\boldsymbol{\alpha = 1}$	$\boldsymbol{\alpha = 3}$	$\boldsymbol{HTR}$
$\tilde{\xi}_{max}$	2.4°	2.4°	2.4°
$\tilde{\xi}_{RMSE}$	0.8°	0.9°	1.0°
TP_{max}	**1.0** Nm	**1.0** Nm	**0.0** Nm
TP_{RMSE}	**0.2** Nm	**0.9** Nm	**0.0** Nm

Table 6.8: Performance Metrics for ARII and RBFNN-I Quanser QUBEs Telerehabilitation Experiments, with α=1, α=3, and HTR

$\boldsymbol{\tau_{net}}$ ***calcs.***	**Metrics**	**ARII**	**RBFNN-I**
$\boldsymbol{\alpha = 1}$	$\tilde{\xi}_{max}$	2.8°	2.4°
	$\tilde{\xi}_{RMSE}$	1.1°	0.8°
	TP_{max}	1.0 Nm	1.0 Nm
	TP_{RMSE}	0.2 Nm	0.2 Nm
$\boldsymbol{\alpha = 3}$	$\tilde{\xi}_{max}$	2.7°	2.4°
	$\tilde{\xi}_{RMSE}$	1.0°	0.9°
	TP_{max}	1.0 Nm	1.0 Nm
	TP_{RMSE}	0.9 Nm	0.9 Nm
HTR	$\tilde{\xi}_{max}$	2.8°	2.4°
	$\tilde{\xi}_{RMSE}$	1.1°	0.9°
	TP_{max}	**0.0** Nm	**0.0** Nm
	TP_{RMSE}	**0.0** Nm	**0.0** Nm

6.5 Telerehabilitation Human Torque Combination Simulations with 2-DOF Robotic Exoskeleton Model

Having validated the proposed RBFNN-I control approach on single DOF bilateral telerehabilitation experiments, the RBFNN-I method is now validated in telerehabilitation scenarios on multi-DOF systems by using the designed 2-DOF robotic exoskeleton simulation testbed from §4.4 and applying the three different net human torque synthesis strategies.

6.5.1 Direct $\boldsymbol{\tau}_{refl}$ Addition

Figs. 6.24 to 6.27 show the results of the simulation when the reflected assistive therapist torque is directly added to the patient's torque using (5.52). Similar to the 2-DOF simulations with the ARII control method in §5.7.1, Fig. 6.24 to Fig. 6.27 follow a similar format as Fig. 5.21 to Fig. 5.24, presenting the 2-DOF robotic exoskeleton simulator's joint 1 and joint 2 angular position, the impedance angular position tracking error, and the torques in the simulation.

As can be seen in Figs. 6.25 and 6.27(b) in regions E and F, the telepresence slowly converges towards the ideal value of zero under the direct addition of the therapist's torque to the patient's torque. Similar to the previous section, the yellow area highlights the relatively large difference between the achieved telepresence and the ideal telepresence. The position tracking and telepresence performance metrics for this $\alpha = 1$ simulation test are summarized in Table 6.9.

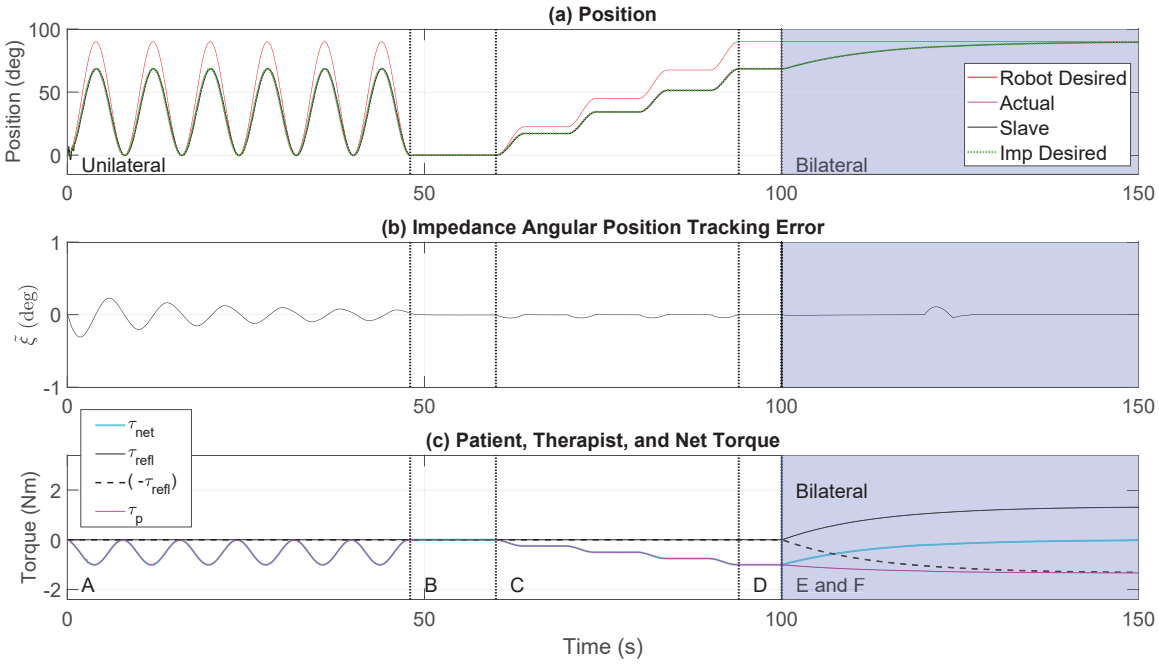


Figure 6.24: Joint 1 RBFNN-I 2-DOF telerehabilitation simulation with $\alpha = 1$: (a) position trajectory, (b) tracking error, and (c) torques

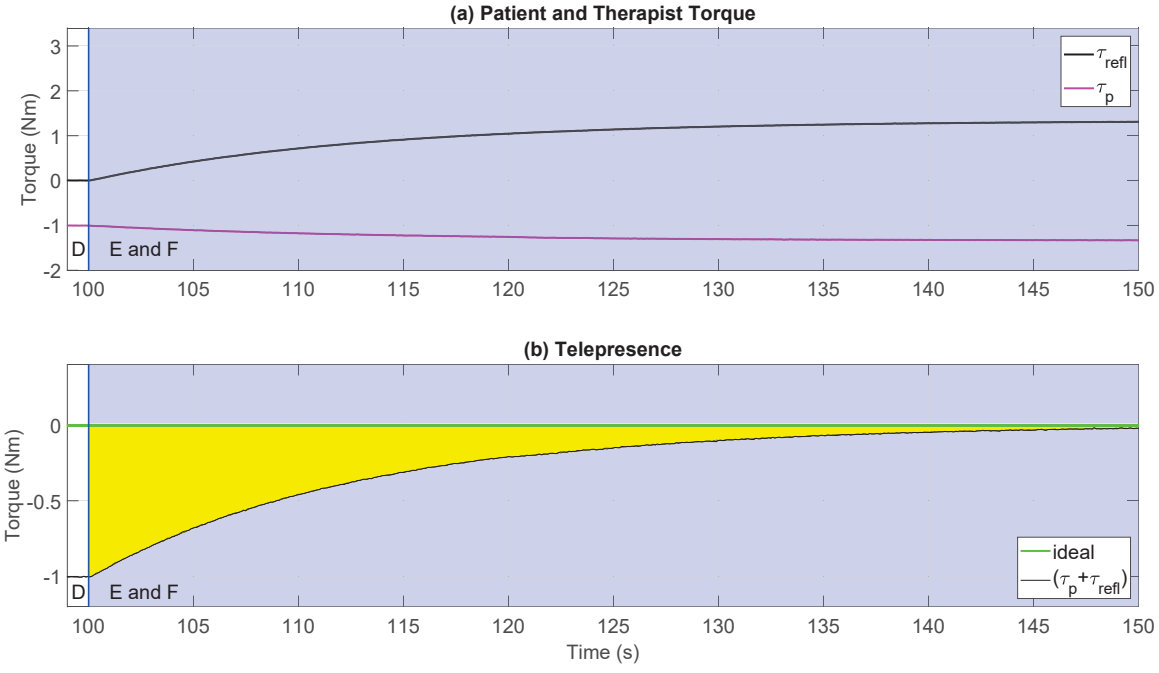


Figure 6.25: Joint 1 torques for the RBFNN-I 2-DOF telerehabilitation simulation with $\alpha = 1$ during the bilateral phase

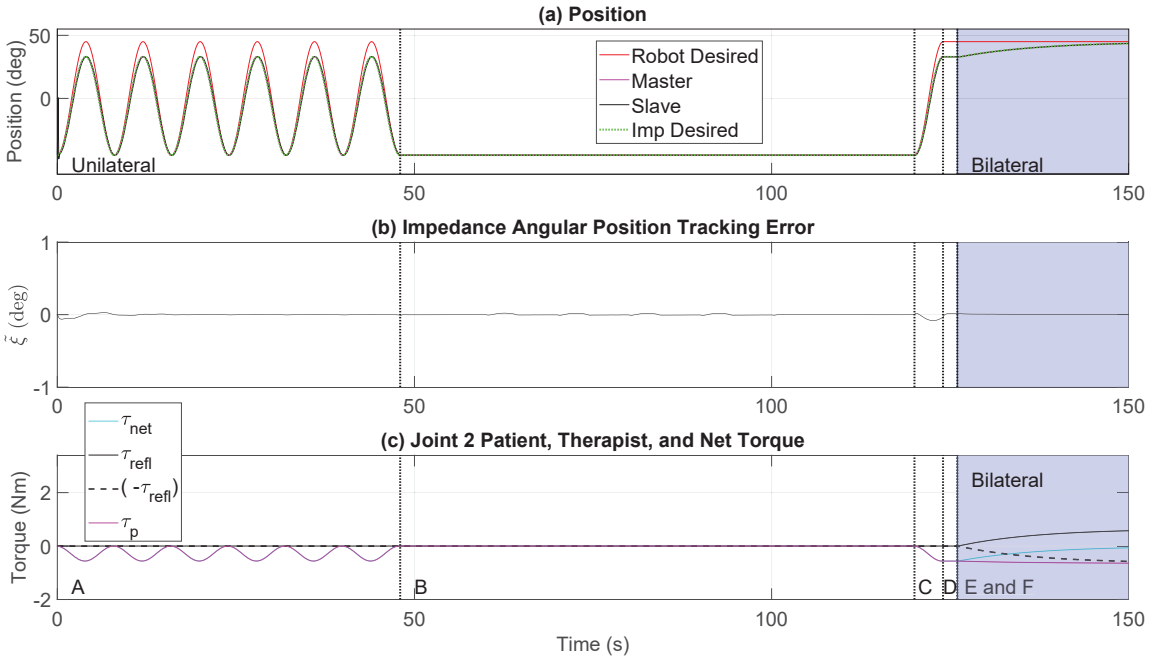


Figure 6.26: Joint 2 RBFNN-I 2-DOF telerehabilitation simulation with $\alpha = 1$: (a) position trajectory, (b) tracking error, and (c) torques

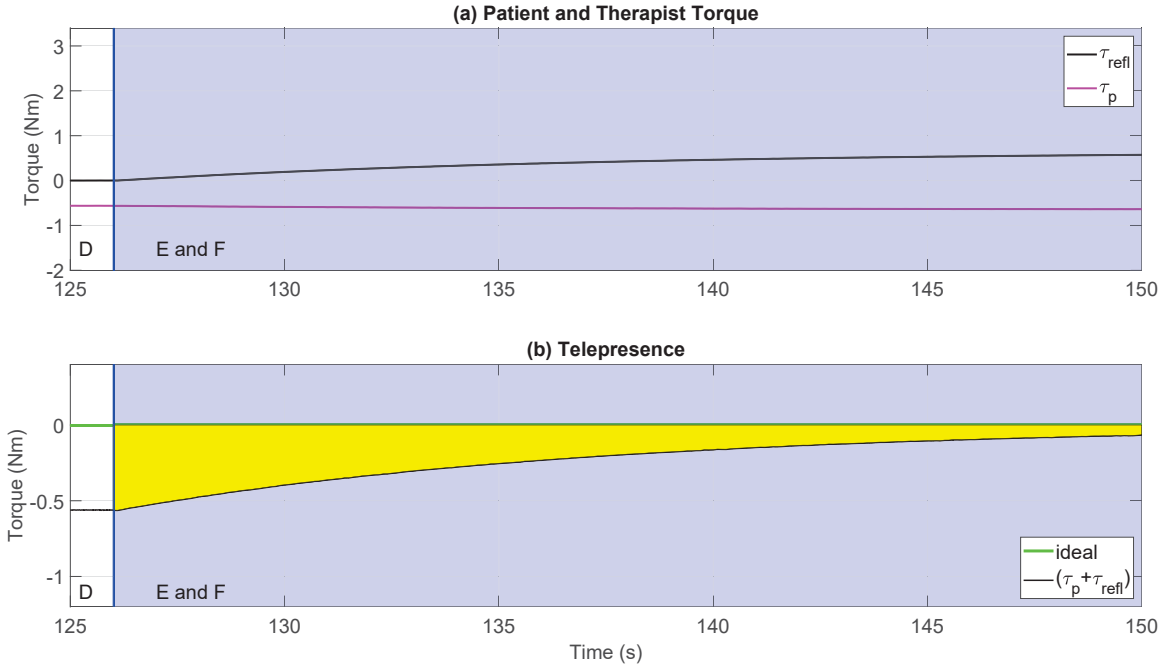


Figure 6.27: Joint 2 torques for the RBFNN-I 2-DOF telerehabilitation simulation with $\alpha = 1$ during the bilateral phase

6.5.2 Scaled $\boldsymbol{\tau}_{refl}$ Addition

Figs. 6.28 to 6.31 show the 2-DOF exoskeleton simulation results from the tests using the $\alpha = 3$ scaled-$\boldsymbol{\tau}_{refl}$ addition strategy given by (5.53). These figures follow a similar format as Figs. 6.24 to 6.27.

As can be seen in Figs. 6.29(b) and 6.31(b) in regions E and F, the telepresence values converge to just under -1 Nm and just under -0.5 Nm, for joint 1 and 2, respectively. The low-fidelity telepresence is evident by the large yellow areas in the telepresence plot between the ideal telepresence green line and the telepresence achieved for this test. Comparing Figs. 6.29(b) with 6.25(b), it is evident that when $\alpha = 3$ the results exhibit worse telepresence behaviour for the therapist than when $\alpha = 1$.

The RBFNN-I position tracking and telepresence performance metrics for this $\alpha = 3$ simulation are listed in Table 6.9 where the worse performance of $\alpha = 3$ compared to $\alpha = 1$ can clearly be seen in the TP_{max} and TP_{RMSE} telepresence metrics.

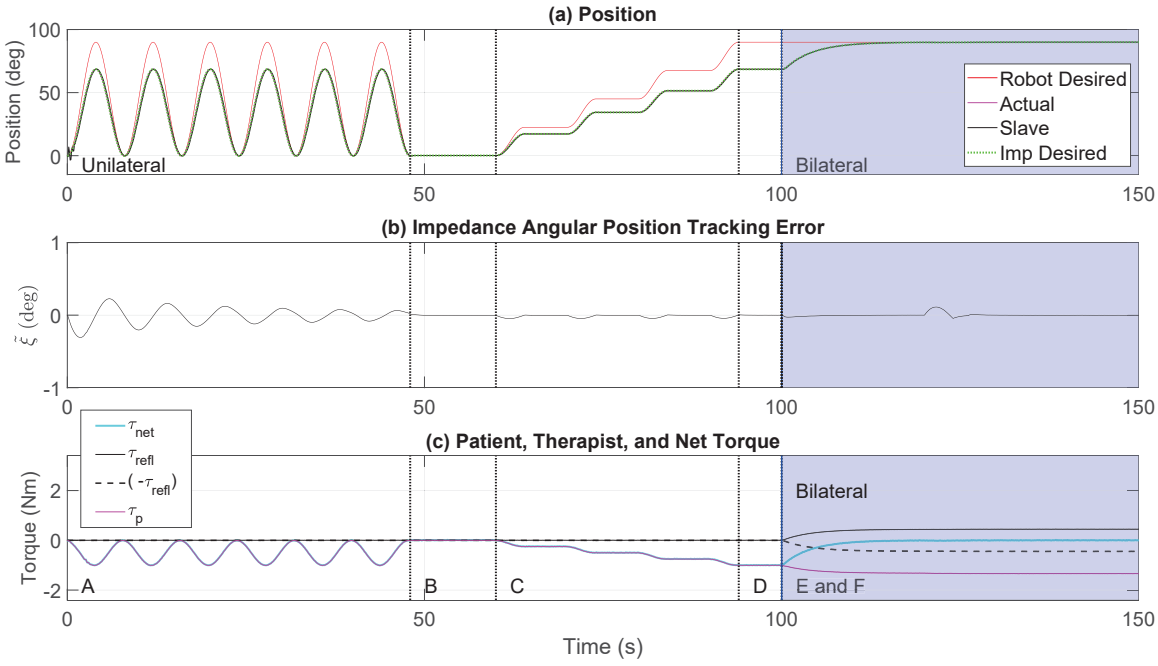


Figure 6.28: Joint 1 RBFNN-I 2-DOF telerehabilitation simulation with $\alpha = 3$: (a) position trajectory, (b) tracking error, and (c) torques

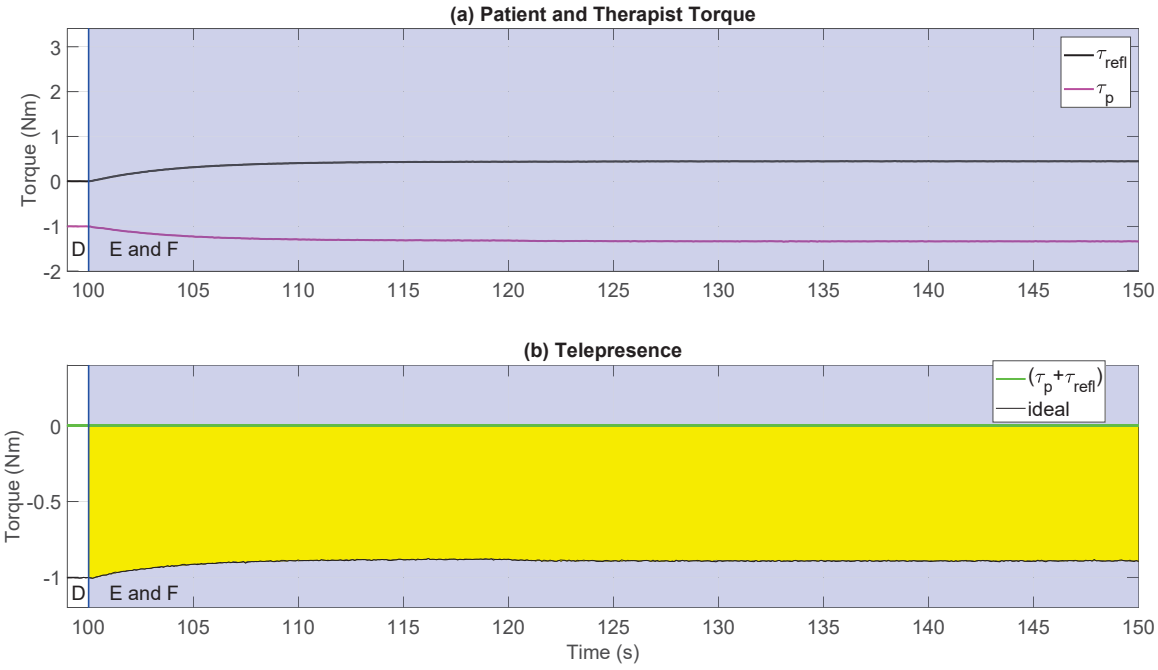


Figure 6.29: Joint 1 torques for the RBFNN-I 2-DOF telerehabilitation simulation with $\alpha = 3$ during the bilateral phase

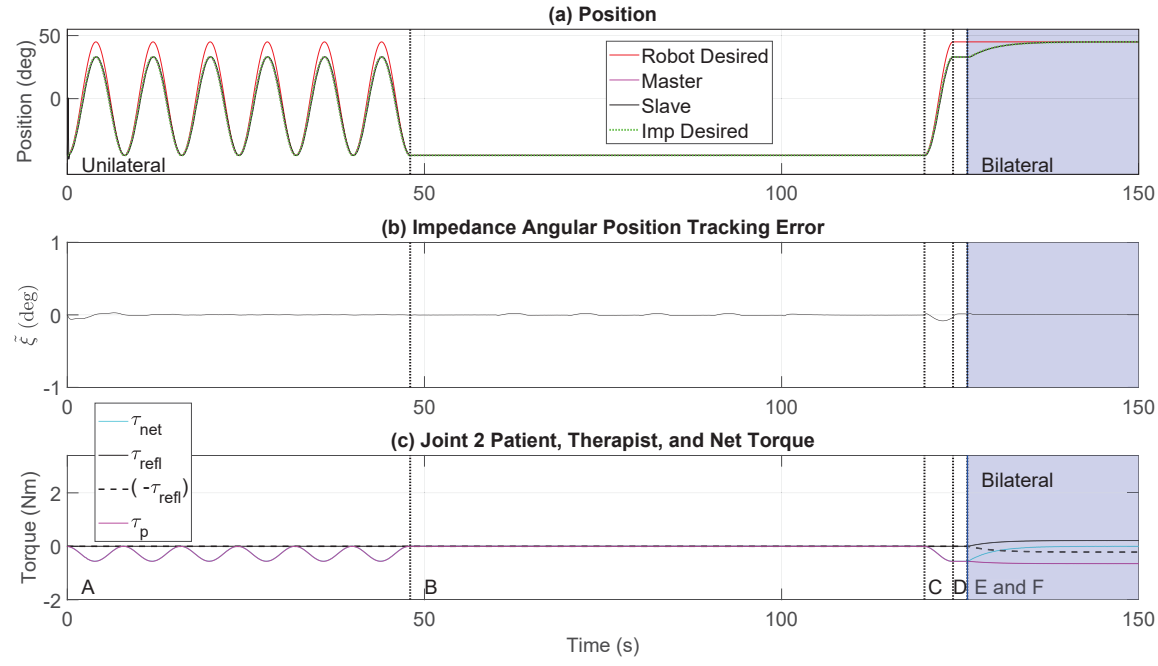


Figure 6.30: Joint 2 RBFNN-I 2-DOF telerehabilitation simulation with $\alpha = 3$: (a) position trajectory, (b) tracking error, and (c) torques

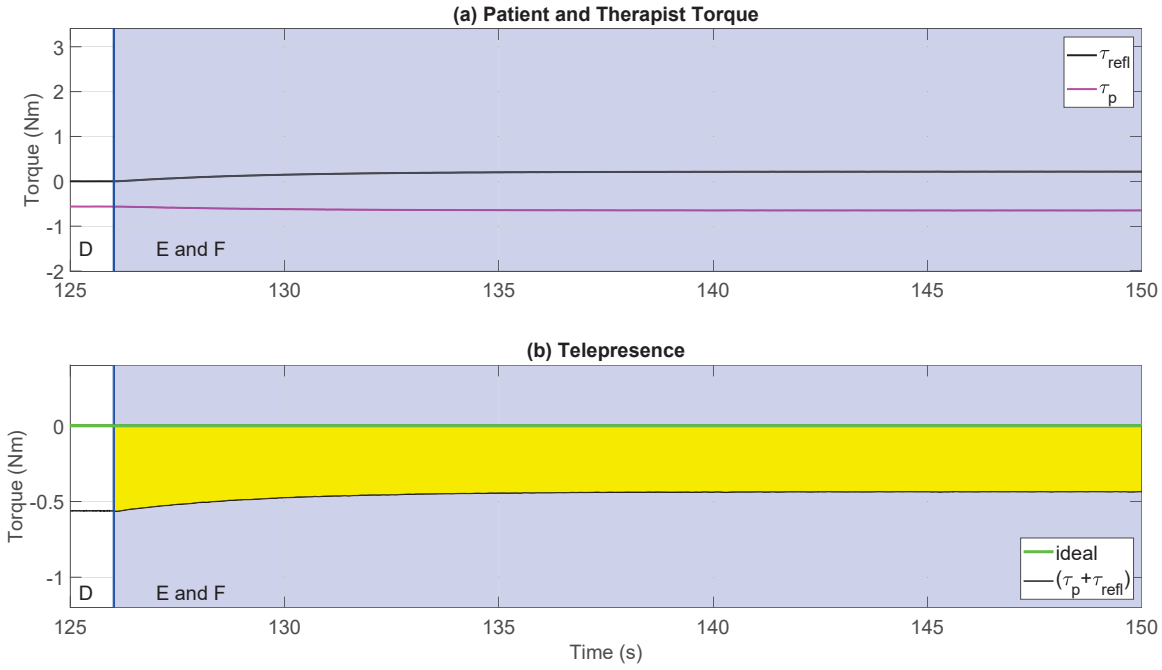


Figure 6.31: Joint 2 torques for the RBFNN-I 2-DOF telerehabilitation simulation with $\alpha = 3$ during the bilateral phase

6.5.3 Human Torque Regulator

Figs. 6.32 to 6.35 show the RBFNN-I simulation results for multi-degrees of freedom implementation of the new HTR strategy introduced in this research in (5.54). These figures again follow a similar format to Figs. 6.24 to 6.27 and Figs. 6.28 to 6.31 in the previous sections.

As can be seen in Figs. 6.33(b) and 6.35(b), once the therapist's torque builds up to the patient's toque in region E, it becomes equal to the patient's torque and bilateral telerehabilitation begins (corresponding to region F). During this bilateral telerehabilitation region, even with 2-DOF the new HTR strategy provides excellent telepresence. This high-fidelity telepresence is evident by the lack of yellow areas in the telepresence plots of Figs. 6.33 and 6.35 for joints 1 and 2, respectively. The position tracking and telepresence performance metrics for this experiment are listed in Table 6.9. It can be seen in this table that the proposed HTR approach using RBFNN-I provides superior telepresence compared with the $\alpha = 1$ and $\alpha = 3$ approaches since the telepresence metrics TP_{max} and TP_{RMSE} are zero Nm for both joint 1 and joint 2 as desired.

Table 6.10 additionally lists the performance metrics for the ARII controller from §5.7 when this control method was tested on the same 2-DOF robotic exoskeleton testbed. It can be seen that the RBFNN-I performance is very similar to that of the ARII method when applied for the 2-DOF simulator testbed.

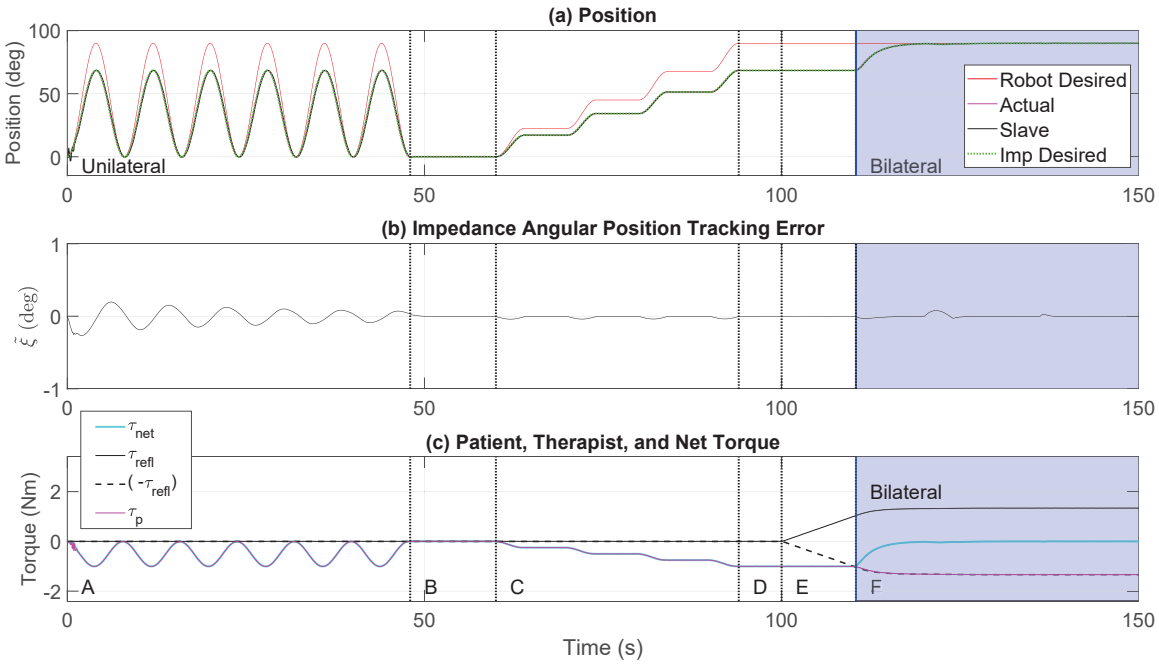


Figure 6.32: Joint 1 RBFNN-I 2-DOF telerehabilitation simulation with HTR: (a) position trajectory, (b) tracking error, and (c) torques

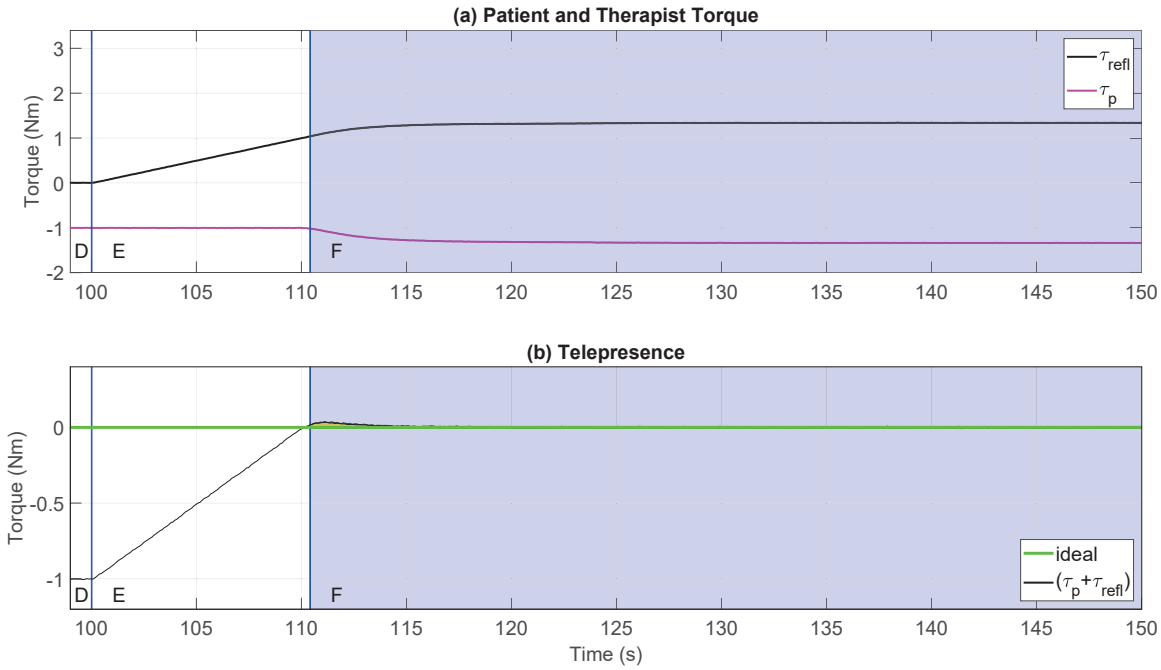


Figure 6.33: Joint 1 torques for the RBFNN-I 2-DOF telerehabilitation simulation with HTR during the bilateral phase

Table 6.9: Performance Metrics for RBFNN-I Telerehabilitation Simulations with 2-DOF Exoskeleton

Actuator	**Metrics**	$\boldsymbol{\alpha = 1}$	$\boldsymbol{\alpha = 3}$	***HTR***
Joint 1	$\tilde{\xi}_{max}$	0.3°	0.3°	0.3°
	$\tilde{\xi}_{RMSE}$	0.1°	0.1°	0.1°
	TP_{max}	**1.0** Nm	**1.0** Nm	**0.0** Nm
	TP_{RMSE}	**0.4** Nm	**0.9** Nm	**0.0** Nm
Joint 2	$\tilde{\xi}_{max}$	0.1°	0.1°	0.1°
	$\tilde{\xi}_{RMSE}$	0.0°	0.0°	0.0°
	TP_{max}	**0.6** Nm	**0.6** Nm	**0.0** Nm
	TP_{RMSE}	**0.3** Nm	**0.5** Nm	**0.0** Nm

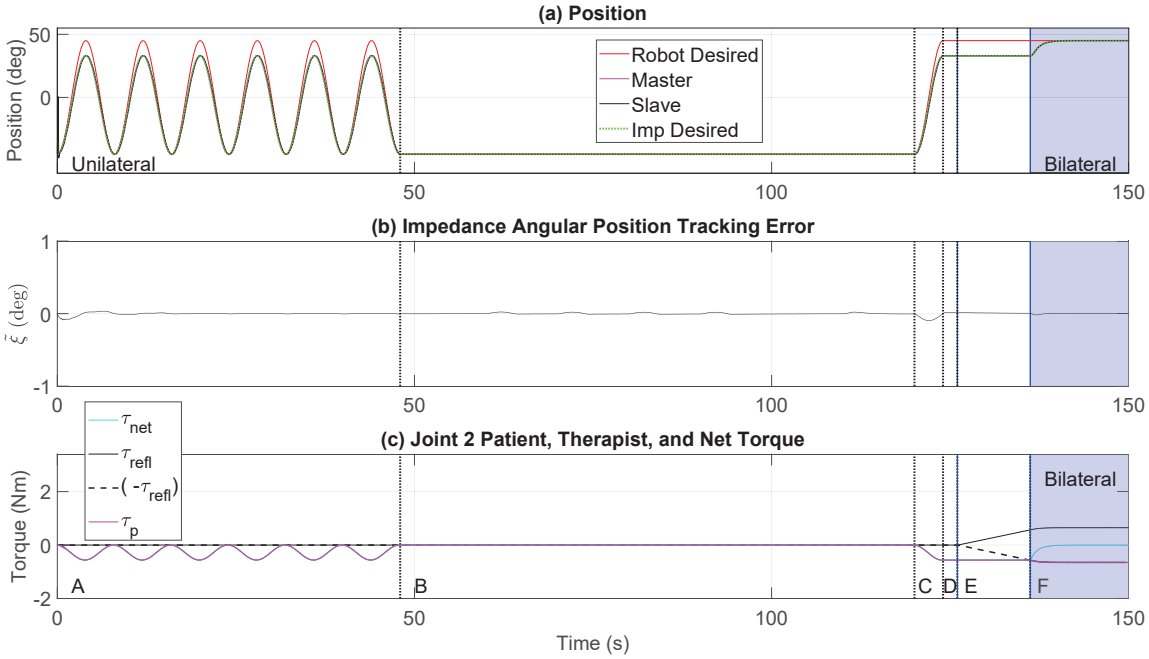


Figure 6.34: Joint 2 RBFNN-I 2-DOF telerehabilitation simulation with HTR: (a) position trajectory, (b) tracking error, and (c) torques

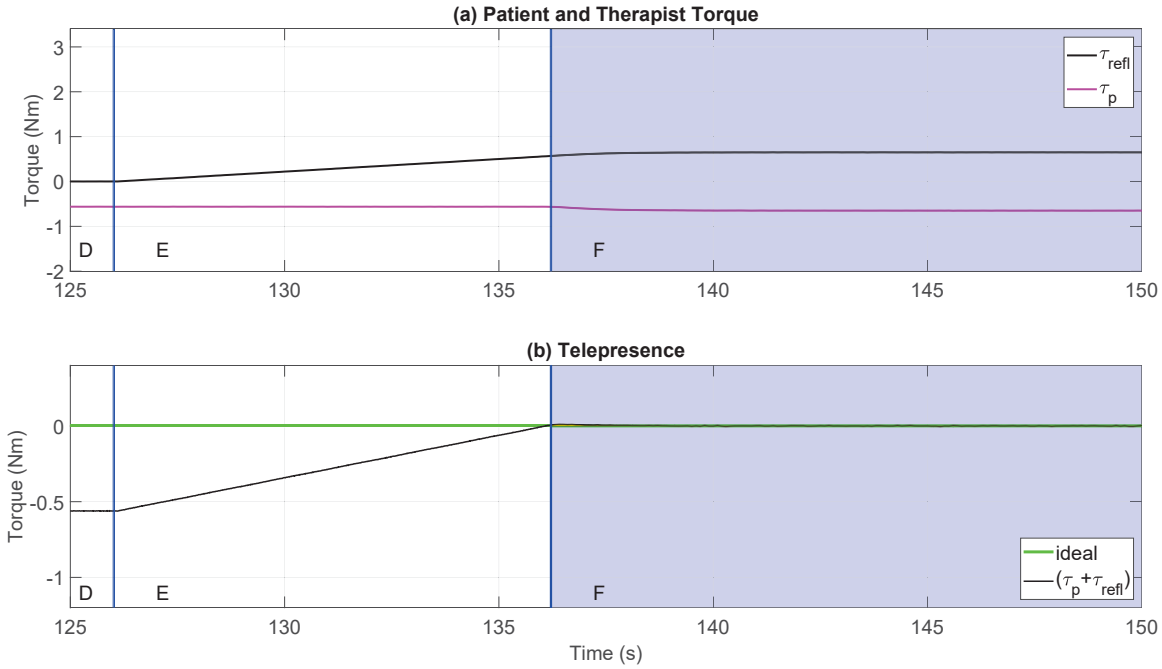


Figure 6.35: Joint 2 torques for the RBFNN-I 2-DOF telerehabilitation simulation with HTR during the bilateral phase

Table 6.10: 2-DOF Exoskeleton Telerehabilitation Simulations, with α=1, α=3, and HTR

$\boldsymbol{\tau_{net}}$	**Actuator**	**Metrics**	**ARII**	**RBFNN-I**
$\boldsymbol{\alpha = 1}$	**Joint 1**	$\tilde{\xi}_{max}$	0.5°	0.3°
		$\tilde{\xi}_{RMSE}$	0.1°	0.1°
		TP_{max}	**1 Nm**	**1 Nm**
		TP_{RMSE}	**0.4 Nm**	**0.4 Nm**
	Joint 2	$\tilde{\xi}_{max}$	0.1°	0.1°
		$\tilde{\xi}_{RMSE}$	0.0°	0.0 °
		TP_{max}	**0.6 Nm**	**0.6 Nm**
		TP_{RMSE}	**0.3 Nm**	**0.3 Nm**
$\boldsymbol{\alpha = 3}$	**Joint 1**	$\tilde{\xi}_{max}$	0.4°	0.3°
		$\tilde{\xi}_{RMSE}$	0.1°	0.1°
		TP_{max}	**1.0 Nm**	**1.0 Nm**
		TP_{RMSE}	**0.9 Nm**	**0.9 Nm**
	Joint 2	$\tilde{\xi}_{max}$	0.1°	0.1°
		$\tilde{\xi}_{RMSE}$	0°	0°
		TP_{max}	**0.6 Nm**	**0.6 Nm**
		TP_{RMSE}	**0.5 Nm**	**0.5 Nm**
HTR	**Joint 1**	$\tilde{\xi}_{max}$	0.5°	0.3°
		$\tilde{\xi}_{RMSE}$	0.1°	0.1°
		TP_{max}	**0.0 Nm**	**0.0 Nm**
		TP_{RMSE}	**0.0 Nm**	**0.0 Nm**
	Joint 2	$\tilde{\xi}_{max}$	0.1°	0.1°
		$\tilde{\xi}_{RMSE}$	0.0°	0.0°
		TP_{max}	**0.0 Nm**	**0.0 Nm**
		TP_{RMSE}	**0.0 Nm**	**0.0 Nm**

6.6 Discussion

As can be seen from the experimental and simulation tests presented in this chapter, the RBFNN-I method is stable and robust in both single-system, unilateral, and bilateral telerehabilitation for both 1-DOF and 2-DOF systems. The method demonstrated robustness in the presence of sensor noise, friction and human interactions, and adapted well to the unknown dynamic parameters. RBFNN-I has the advantage that it is model-free and, as a result, RBFNN-I requires less information and less *a priori* knowledge of the system being controlled than ARII.

6.7 Contributions

The RBFNN-I control method presented in this chapter has several contributions:

1. The RBF neural networks-based control method was modified to be used on the robotic exoskeleton by augmenting it to include an impedance model feature. The present author coined the term RBF Neural Network Adaptive Robust Integral Impedance (RBFNN-I) to refer to this proposed control approach.

2. The theoretical proof of the proposed RBFNN-I method for n-DOF system was developed.

3. The proposed RBFNN-I method was validated in simulation on the 1-DOF Quanser QUBE testbeds and 2-DOF robotic exoskeleton models.

4. The RBFNN-I method was experimentally validated using the 1-DOF Quanser QUBE testbeds and the Logitech driving wheel to provide assistive human-input.

5. The results of an earlier version of the proposed RBFNN-I control method were written in: G. Bauer and Y.J. Pan, "Telerehabilitation with Exoskeletons using Adaptive Robust Integral RBF-Neural-Network Impedance Control under Variable Time Delays," Accepted in ISIE 2021, Japan, Kyoto, Conference Date: June 2021. [66]

Chapter 7

ARII and RBFNN-I Telerehabilitation Case Studies and Comparison

This chapter explores the application of the proposed ARII and RBFNN-I control methods to five different telerehabilitation cases using the experimental Quanser QUBE and the simulated 2-DOF exoskeleton telerehabilitation testbeds. The HTR tests previously shown for both the ARII and RBFNN-I controllers are used as a baseline for comparison.

7.1 Quanser QUBE Experimental Case Studies and Comparison

The following five different case studies were explored: different patient profile, different impedance model, double the frequency of the harmonic motion during the unilateral portion of the test, triple the delays, and double the noise variance (Figs. 7.1-7.6). The human torque synthesis method utilized for all these tests was the novel HTR approach developed in this research. Table 7.1 presents the patient profile, impedance model, harmonic motion frequency, communication time delays, and sensor-noise experimental parameters used for the baseline tests with the HTR along with the parameters that were changed for each of the five case studies being explored. The table entry was left blank where the parameters were kept the same as the baseline test.

Fig. 7.1 shows how for the first case study, a patient profile exhibiting less toning results in an impedance-based desired trajectory that is closer to the ideal desired trajectory. For the second case study, Fig. 7.2 compares the designed desired impedance joint angular position tracking error $\tilde{q}_{\tau net}(t)$ response to a 1 Nm torque using the baseline and the Case 2 impedance model. With these two different impedance models, Fig. 7.3 plots the corresponding baseline impedance-based desired trajectory and the Case 1 impedance-based desired trajectory, along with the original desired trajectory. Note how a smaller $\tilde{q}_{\tau net}(t)$ results in an impedance-based trajectory that is closer to

Table 7.1: Quanser QUBE Telerehabilitation Experiments Case Study: Parameters

Case Study	Patient	Impedance Model	Frequency	Delays	Noise
Baseline	K=0.848Nm D=0.2Nms	J=0.0133kgm^2 B=0.6665Nms K = 2.666Nm	w=$\frac{\pi}{4}$rad/s	$T1$=0.1s $T2$=0.1s	var = 0.000058N^2 $mean$ = 0.0129N
1) Patient Profile	K=0.5Nm D=0.1Nms				
2) Impedance Model: lower inertia, higher stiffness		J=0.0066kgm^2 B=0.6665Nms K=5.3320Nm			
3) Higher frequency			w=$\frac{\pi}{2}$rad/s		
4) Longer delays				$T1$=0.3s $T2$=0.3s	
5) Noise with higher variance					var= 0.000116N^2 $mean$= 0.0129N

the desired angular position trajectory. Fig. 7.4 shows the the desired and impedance-based desired tracking positions for the baseline harmonic motion frequency of $\pi/4$ rad/s and the higher frequency of $\pi/2$ rad/s used for the third case study. Fig. 7.5 shows the master and the slave position curves when the system experiences a 0.1 s delay across the communication channel in each direction as used in the baseline test and compares this delay with the master and slave position when the delay was increased to 0.3 s in each direction as used in the fourth case study. Fig. 7.6 compares the baseline force sensor noise with the sensor noise used in Case 5 where the variance of the noise was double in size.

Table 7.2 shows the resulting performance metrics observed for the baseline experimental tests as well as for the five experimental case studies carried out with the experimental Quanser QUBE telerehabilitation testbed. The results show that the highest impedance angular position tracking error $\tilde{\xi}_{max}$ was 4.2° which was attained

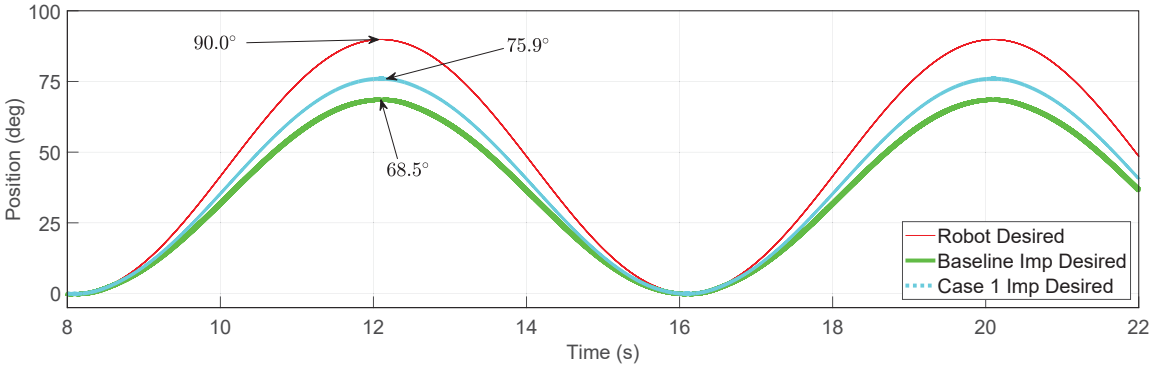


Figure 7.1: Baseline and case 1 position tracking

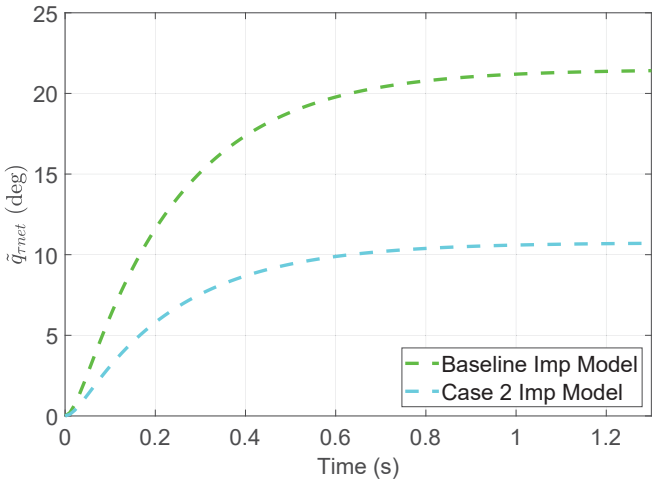


Figure 7.2: Designed impedance responses to a 1 Nm step input

using the ARII method when the harmonic motion frequency was doubled (Case 3). This tracking error corresponds to only a 1.4° difference from the baseline test, which is reasonable considering the higher frequencies used in this case. This test also had the highest $\tilde{\xi}_{RMSE}$ (1.4°). All other cases tested exhibited very similar performance as the baseline test. These observations suggest that for the controller parameters used in this research, the results are generally not very sensitive to different patient profiles, impedance models, motion frequencies, time delays, and sensor-noise levels used in the case studies.

The baseline controller parameters were selected based on an extensive and systematic manual parameter tuning process which was carried out independently with each of the ARII and the RBFNN-I control methods. Using these tuned controller

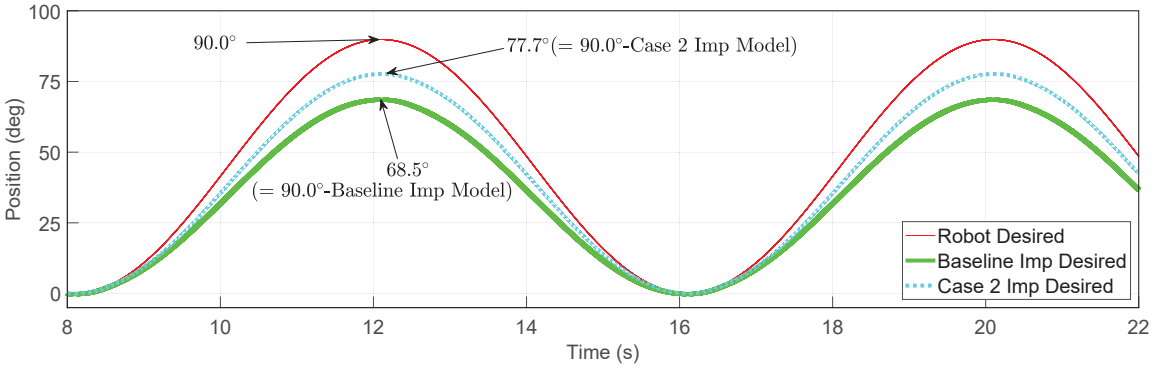


Figure 7.3: Baseline and case 2 position tracking

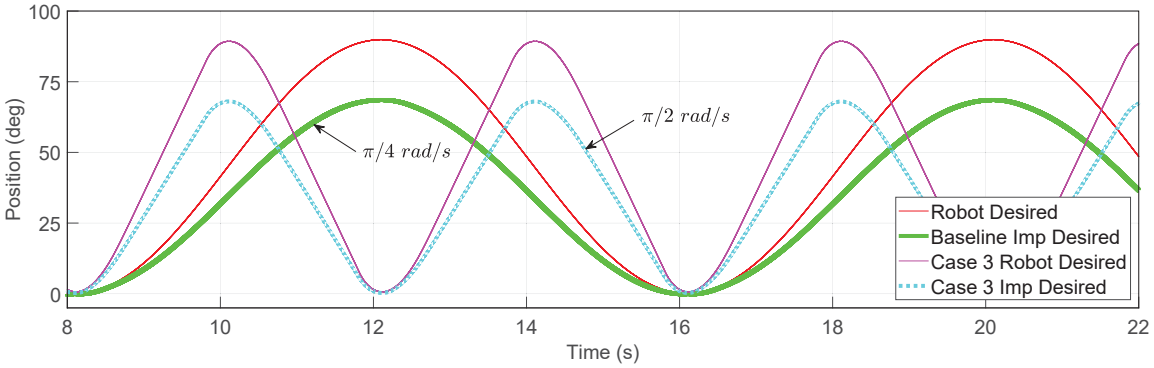


Figure 7.4: Baseline and case 3 position tracking

parameters, it can be observed in Table 7.2 that, for the conditions used in this research, the performances of the two controllers are very similar in all case studies tested. Moreover, note how the TP_{max} and TP_{RMSE} have values of 0.0 Nm in all cases attesting to the ability of the new HTR method to achieve high telepresence fidelity over a wide range of test conditions. The corresponding position and torque plots for these experimental case studies can be found in Appendix A.

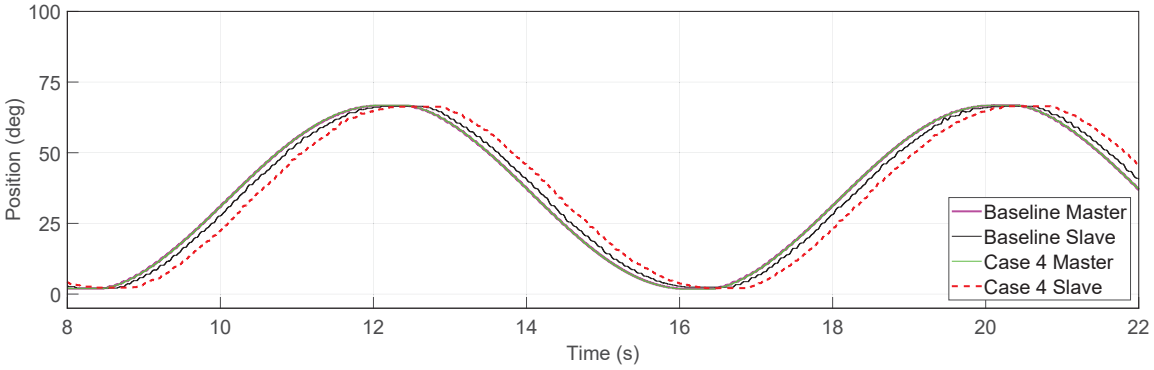


Figure 7.5: Baseline and case 4 position tracking

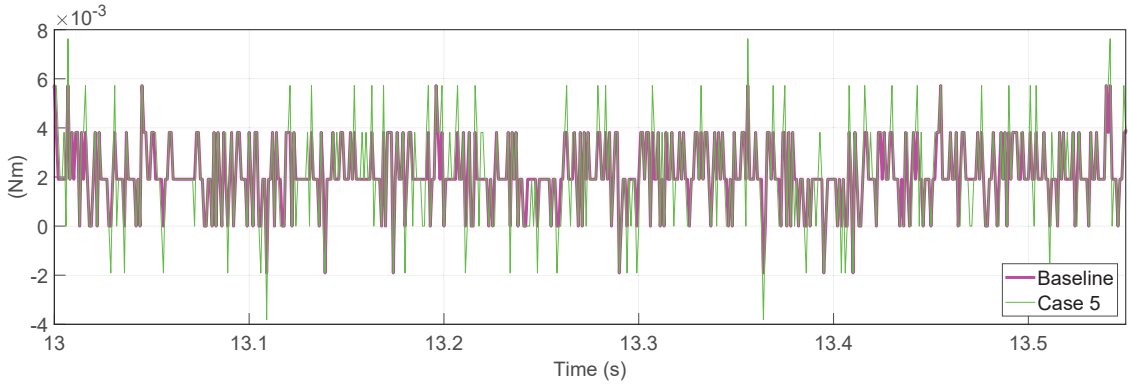


Figure 7.6: Baseline and case 5 signal noise

Table 7.2: Quanser QUBE Telerehabilitation Experiments Case Study: Performance Metrics using ARII and RBFNN-I

Case Study	**Metrics**	**ARII**	**RBFNN-I**
Baseline		Figs. 5.18 and 5.20	Figs. 6.21 and 6.23
	$\tilde{\xi}_{max}$	2.8°	2.4°
	$\tilde{\xi}_{RMSE}$	1.1°	0.9°
	TP_{max}	**0.0** Nm	**0.0** Nm
	TP_{RMSE}	**0.0** Nm	**0.0** Nm
1) Patient Profile		Figs. A.1 and A.2	Figs. A.11 and A.12
	$\tilde{\xi}_{max}$	3.1°	2.7°
	$\tilde{\xi}_{RMSE}$	1.2°	1.0°
	TP_{max}	**0.0 Nm**	**0.0 Nm**
	TP_{RMSE}	**0.0 Nm**	**0.0 Nm**
2) Impedance Model		Figs. A.3 and A.4	Figs. A.13 and A.14
	$\tilde{\xi}_{max}$	3.2°	2.5°
	$\tilde{\xi}_{RMSE}$	1.2°	1.0°
	TP_{max}	**0.0 Nm**	**0.0 Nm**
	TP_{RMSE}	**0.0 Nm**	**0.0 Nm**
3) Frequency		Figs. A.5 and A.6	Figs. A.15 and A.16
	$\tilde{\xi}_{max}$	4.2°	3.9°
	$\tilde{\xi}_{RMSE}$	1.4°	1.1°
	TP_{max}	**0.0 Nm**	**0.0 Nm**
	TP_{RMSE}	**0.0 Nm**	**0.0 Nm**
4) Delays		Figs. A.7 and A.8	Figs. A.17 and A.18
	$\tilde{\xi}_{max}$	3°	2.5°
	$\tilde{\xi}_{RMSE}$	1.1°	1.0°
	TP_{max}	**0.0 Nm**	**0.0 Nm**
	TP_{RMSE}	**0.0 Nm**	**0.0 Nm**
5) Noise		Figs. A.9 and A.10	Figs. A.19 and A.20
	$\tilde{\xi}_{max}$	3°	2.4°
	$\tilde{\xi}_{RMSE}$	1.1°	1.0°
	TP_{max}	**0.0 Nm**	**0.0 Nm**
	TP_{RMSE}	**0.0 Nm**	**0.0 Nm**

7.2 2-DOF Exoskeleton Simulation Case Studies and Comparison

To explore the performance of ARII and RBFNN-I on higher-DOF systems, similar case studies were carried out with the HTR human torque synthesis method using the 2-DOF exoskeleton telerehabilitation simulator. Table 7.3 presents the parameters used in the baseline test with the HTR as well as the parameters for each of the five case studies explored. Similar to the Quanser QUBE tests, the table entry was left blank where the parameters were kept the same as the baseline.

Similar to the experimental Quanser QUBE telerehabilitation case studies, the performance metrics for the baseline and the five telerehabilitation case studies tested with the 2-DOF exoskeleton simulator are summarized in Tables 7.4 and 7.5 for joint 1 and joint 2, respectively. The results show that the highest $\tilde{\xi}_{max}$ was found to be 1.8° and 0.3° for joints 1 and 2, respectively, when using the ARII method for the case where the frequency was doubled (Case 3). This result is 1.3° larger than the baseline test which is reasonable for this higher-frequency case.

For all other case studies, the performance of the ARII and RBFNN-I controllers was similar to their performance in the baseline test. These observations again suggest that, for the tuned controller parameters used in this research, the results are not very sensitive to the parameter changes being explored in these case studies. It can also be seen in Tables 7.4 and 7.5 that the performance of ARII and RBFNN-I is generally similar for all case studies tested. The corresponding position and torque plots for these 2-DOF telerehabilitation simulation case studies can be found in Appendix A.

Table 7.3: 2-DOF Exoskeleton Telerehabilitation Simulation Case Study: Parameters

Case	Patient	Impedance Model	Frequency	Delays	Noise
Baseline	$K_1 = 0.848$Nm $D_1 = 0.2$ Nms $K_2 = 0.424$ m $D_2 = 0.05$Nms	$J_1 = 0.013$ kgm^2 $B_1 = 0.665$ Nms $K_1 = 2.666$Nm $J_2 = 0.013$kgm^2 $B_2 = 0.665$Nms $K_2 = 2.666$Nm	$w_1 = \frac{\pi}{4}$rad/s $w_2 = \frac{\pi}{4}$rad/s	$T1$=0.1s $T2$=0.1s	$var =$ 0.000058N^2 $mean =$ 0.0129N
1) Patient Profile	$K_1 = 0.424$Nm $D_1 = 0.1$Nms $K_1 = 0.212$Nm $D_2 = 0.025$Nms				
2) Imp. Model: lower inertia, higher stiffness		$J_1 = 0.0067$kgm^2 $B_1 = 0.665$Nms $K_1 = 5.332$Nm $J_2 = 0.0067$kgm^2 $B_2 = 0.665$Nms $K_2 = 5.332$Nm			
3) Higher freq.			$w_1 = \frac{\pi}{2}$rad/s $w_2 = \frac{\pi}{2}$rad/s		
4) Longer delays				$T1$=0.3s $T2$=0.3s	
5) Noise with higher variance					$var =$ 0.000116N^2 $mean =$ 0.0129N

Table 7.4: 2-DOF Exoskeleton Telerehabilitation Simulations Case Study: Performance Metrics for Joint 1 using ARII and RBFNN-I

Case Study	**Metrics**	**ARII**	**RBFNN-I**
Baseline		Figs. 5.29 and 5.30	Figs. 6.32 and 6.33
	$\tilde{\xi}_{max}$	0.5°	0.3°
	$\tilde{\xi}_{RMSE}$	0.1°	0.1°
	TP_{max}	**0.0 Nm**	**0.0 Nm**
	TP_{RMSE}	**0.0 Nm**	**0.0 Nm**
1) Patient Profile		Figs. A.21 and A.22	Figs. A.41 and A.42
	$\tilde{\xi}_{max}$	0.5°	0.3°
	$\tilde{\xi}_{RMSE}$	0.2°	0.1°
	TP_{max}	**0.0 Nm**	**0.0 Nm**
	TP_{RMSE}	**0.0 Nm**	**0.0 Nm**
2) Impedance Model		Figs. A.25 and A.26	Figs. A.45 and A.46
	$\tilde{\xi}_{max}$	0.6°	0.3°
	$\tilde{\xi}_{RMSE}$	0.2°	0.1°
	TP_{max}	**0.0 Nm**	**0.0 Nm**
	TP_{RMSE}	**0.0 Nm**	**0.0 Nm**
3) Frequency		Figs. A.29 and A.30	Figs. A.49 and A.50
	$\tilde{\xi}_{max}$	1.8°	0.4°
	$\tilde{\xi}_{RMSE}$	0.2°	0.1°
	TP_{max}	**0.0 Nm**	**0.0 Nm**
	TP_{RMSE}	**0.0 Nm**	**0.0 Nm**
4) Delays		Figs. A.33 and A.34	Figs. A.53 and A.54
	$\tilde{\xi}_{max}$	0.5°	0.3°
	$\tilde{\xi}_{RMSE}$	0.1°	0.1°
	TP_{max}	**0.0 Nm**	**0.0 Nm**
	TP_{RMSE}	**0.0 Nm**	**0.0 Nm**
5) Noise		Figs. A.37 and A.38	Figs. A.57 and A.58
	$\tilde{\xi}_{max}$	0.5°	0.3°
	$\tilde{\xi}_{RMSE}$	0.1°	0.1°
	TP_{max}	**0.0 Nm**	**0.0 Nm**
	TP_{RMSE}	**0.0 Nm**	**0.0 Nm**

Table 7.5: 2-DOF Exoskeleton Telerehabilitation Simulations Case Study: Performance Metrics for Joint 2 using ARII and RBFNN-I

Case Study	**Metrics**	**ARII**	**RBFNN-I**
Baseline		Figs. 5.31 and 5.32	Figs. 6.34 and 6.35
	$\tilde{\xi}_{max}$	0.1°	0.1°
	$\tilde{\xi}_{RMSE}$	0°	0°
	TP_{max}	**0.0 Nm**	**0.0 Nm**
	TP_{RMSE}	**0.0 Nm**	**0.0 Nm**
1) Patient Profile		Figs. A.23 and A.24	Figs. A.43 and A.44
	$\tilde{\xi}_{max}$	0.1°	0.1°
	$\tilde{\xi}_{RMSE}$	0°	0°
	TP_{max}	**0.0 Nm**	**0.0 Nm**
	TP_{RMSE}	**0.0 Nm**	**0.0 Nm**
2) Impedance Model		Figs. A.27 and A.28	Figs. A.47 and A.48
	$\tilde{\xi}_{max}$	0.1°	0.1°
	$\tilde{\xi}_{RMSE}$	0°	0°
	TP_{max}	**0.0 Nm**	**0.0 Nm**
	TP_{RMSE}	**0.0 Nm**	**0.0 Nm**
3) Frequency		Figs. A.31 and A.32	Figs. A.51 and A.52
	$\tilde{\xi}_{max}$	0.3°	0.1°
	$\tilde{\xi}_{RMSE}$	0.1°	0°
	TP_{max}	**0.0 Nm**	**0.0 Nm**
	TP_{RMSE}	**0.0 Nm**	**0.0 Nm**
4) Delays		Figs. A.35 and A.36	Figs. A.55 and A.56
	$\tilde{\xi}_{max}$	0.1°	0.1°
	$\tilde{\xi}_{RMSE}$	0°	0 °
	TP_{max}	**0.0 Nm**	**0.0 Nm**
	TP_{RMSE}	**0.0 Nm**	**0.0 Nm**
5) Noise		Figs. A.39 and A.40	Figs. A.59 and A.60
	$\tilde{\xi}_{max}$	0.1°	0.1°
	$\tilde{\xi}_{RMSE}$	0°	0 °
	TP_{max}	**0.0 Nm**	**0.0 Nm**
	TP_{RMSE}	**0.0 Nm**	**0.0 Nm**

7.3 RBFNN-I Experimental Range of Operation Exploration

There are a multitude of parameters in the system that were tuned to perform the previously-presented tests. It was desired to carry out additional experimental tests to explore the range of operation of the RBFNN-I control method using the Quanser QUBE telerehabilitation testbed. Although there are myriad combinations of parameter variations possible, for these tests the baseline parameter values shown in Table 7.1 were used while exploring the range of the following key parameters which influence the operation of the system: delays, motion frequency, noise, and the HTR telepresence gain ψ.

It was discovered that the system is stable with delays up to 1.1 s in each direction of the communication channel, for a total of 2.2 s delay. The delays were asymmetric and variable. For the conditions used in this research, delays higher than 2.2 s tended to destabilize the system.

Using the asymmetric and variable total time delay of 0.6 s, as in the baseline test, the range of operation for the motion frequency w was found to be up to $5\pi/8$ rad/sec. The tracking error increased with motion frequency, as shown in Fig. 7.7 when compared to the baseline (Fig. 6.21) and case study (Fig. A.15) experimental tests shown in Table 7.6. For the conditions used in this research, increasing the motion frequency beyond $5\pi/8$ rad/sec risked unstable behaviour.

Table 7.6: Quanser QUBE Telerehabilitation Experiments with Different Motion Frequencies using RBFNN-I: Tracking Error Metrics

Frequency w	**Metrics**	**RBFNN-I**
Baseline: $\pi/4$	$\tilde{\xi}_{max}$	2.4°
	$\tilde{\xi}_{RMSE}$	0.9°
Case Study: $\pi/2$	$\tilde{\xi}_{max}$	3.9°
	$\tilde{\xi}_{RMSE}$	1.1°
Max: $5\pi/8$	$\tilde{\xi}_{max}$	5.5°
	$\tilde{\xi}_{RMSE}$	1.4°

The sensor noise variance and mean was amplified by a factor of four and the system was still stable and operational. Noise beyond these levels were not tested.

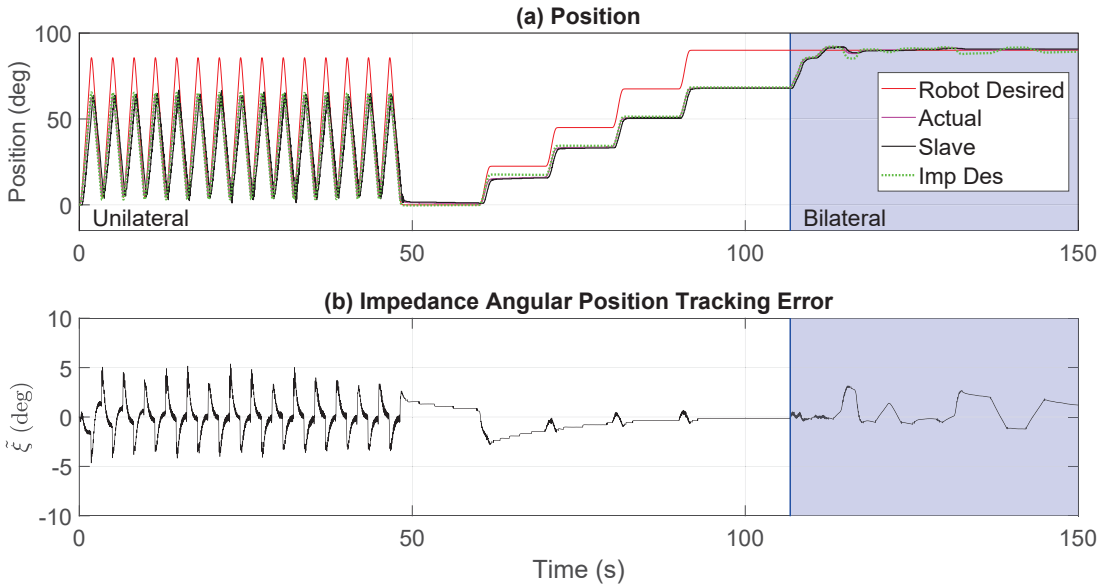


Figure 7.7: Range of operation experimental test with Quanser QUBES, RBFNN-I controller, and w=$5\pi/8$

The range of values for the HTR telepresence gain ψ was also explored and, for the conditions used in this research, it was observed that the system was stable for values between 0.001 and 0.04. The telepresence fidelity started to diminish slightly when using a value of 0.01, and worsened as the value of ψ was reduced. The telepresence performance metrics still revealed relatively high-fidelity, with a TP_{max} value of 0.4Nm and a TP_{RMSE} value of 0.1Nm for a ψ value of 0.001, which can be observed in Fig. 7.8. For values of 0.05 and higher, the system became overreactive and unstable.

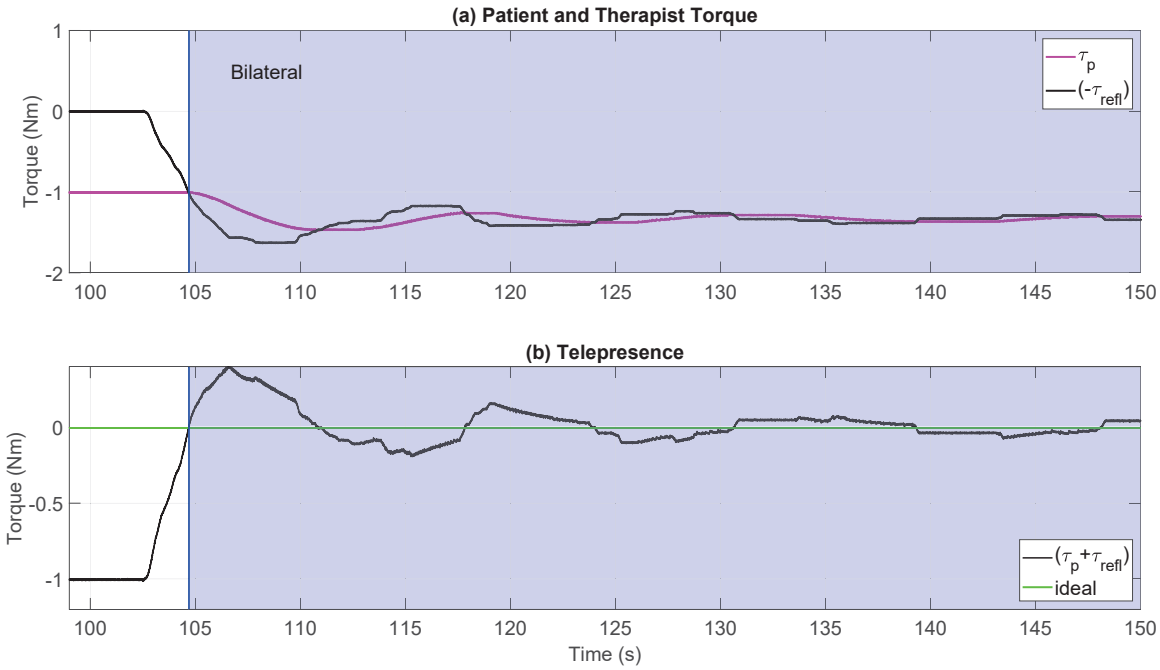


Figure 7.8: Range of operation experimental test with Quanser QUBES, RBFNN-I controller, and ψ=0.001

7.4 Discussion

As can be seen from the different experimental and simulation telerehabilitation case studies presented in this chapter, both ARII and RBFNN-I performed remarkably similar for both the single-joint and the 2-DOF testbeds, with the RBFNN-I controller performing slightly better. The fact that the telepresence values for all the case tests were zero up to the first decimal place, and the tracking errors were minimal, attests to the effectiveness of the HTR strategy and robustness of the system with the tuned parameters.

Chapter 8

Conclusions and Future Work

This chapter summarizes the contributions of the research presented in this thesis and suggests future work in the field of bilateral telerehabilitation with upper-limb robotic exoskeletons.

8.1 Conclusions

The Adaptive Robust Integral Impedance (ARII) controller − a model-based method − was further developed, implemented, and validated to address the unknown dynamic modeling parameters, nonlinearities, sensor noise, and human torque input while providing good tracking, safe, stable, and adjustable compliant motion in the presence of time delays. ARII builds on the state-of-the-art adaptive controllers for upper-limb robotic exoskeletons.

The development, implementation, and validation of a model-free control method called Radial Basis Function Neural Network Integral Impedance (RBFNN-I) was also explored to address the same challenges.

An impedance model controller was incorporated into both proposed control methods to create adjustable compliance, customization for different and changing patients' abilities, and safe motion to accommodate the patient and the therapist in the closed loop system. The impedance model parameters were selected to create an overdamped system and thus enhance stability.

The design and detailed Lyapunov-based stability and convergence analysis of the methods were presented. The time-domain passivity-based strategy of boundedness was utilized to strengthen system stability. Simulated and experimental tests were performed with a single-system single-joint Quanser QUBE testbed. A conceptual design of a novel grounded horizontally-planar 2-DOF upper-limb robotic exoskeleton was developed and simulated to validate the proposed control methods with multi-degree robotic exoskeleton systems. For the conditions used in this research, the

results from the tests performed with both testbeds revealed the desired tracking performance and stability.

For bilateral telerehabilitation, a novel human torque synthesis strategy, called Human Torque Regulator (HTR), was developed which produces high-fidelity telepresence for the therapist while assisting the patient. The current methodologies found in the literature utilized direct or scaled-down values of the reflected therapist torques to be added to the patient-exoskeleton master system, which could be a benefit in cases where therapist fatigue reduction is prioritized; however, they present diminished haptic awareness of the torques experienced by the patient. Three human torque synthesis strategies were implemented using the two proposed control methods, ARII and RBFNN-I, in bilateral telerehabilitation in experiment and simulation using the single-joint Quanser QUBE testbed, and in simulation using the 2-DOF robotic exoskeleton testbed. The bilateral telerehabilitation tests revealed the HTR approach as superior. The HTR method was also implemented in a set of test cases with the two proposed control methods under five different conditions with both the single-joint Quanser QUBE testbed and the simulated 2-DOF robotic exoskeleton testbed, and consistently displayed high-fidelity telepresence.

The proposed methodologies were successfully implemented to demonstrate stable performance with superior telepresence and great tracking performance while under variable asymmetric communication time delays.

8.2 Future Work

The research presented in this thesis contributed to the field of bilateral telerehabilitation with upper-limb robotic exoskeletons by developing a novel human torque synthesis strategy called the human torque regulator (HTR), which improves telepresence. Additionally, two upper limb robotic exoskeleton control methods were developed which build on the existing state-of-the-art to help further the body of knowledge which may aid in the delivery of physiotherapy to those who direly need it.

HTR proved to work very well with the presented upper-limb robotic exoskeletons setup. For future work, it could also be implemented as a high-fidelity telepresence method for systems using end-effector type robots, and lower limb assistive robots.

The presented work showed that the model-based and model-free robot control

methods developed, implemented, and validated work well with the simulated 2-DOF robotic exoskeleton testbed. For future work, it is recommended that they be implemented in experiment with two or higher DOF upper-limb exoskeletons. Furthermore these methods could be further developed and implented in a multitude of applications involving human-machine interface which require human-like, customizable, compliant, and safe motion.

The big-picture recommendations are for rehabilitation robotic devices to be actively integrated into physiotherapy clinics and hospitals around the world so that therapists and patients can benefit from their proven effectiveness. Their integration and acceptance as in-person and remote physiotherapy tools will greatly increase access to care and help mitigate the worldwide exorbitant demand for physical rehabilitation services.

Bibliography

[1] U. Nations, "World population prospects 2019: Highlights," *Department of Economic and Social Affairs*, 2019.

[2] R. Lowe. Rehabilitation 2030 2.4 billion people are in need of rehabilitation services. https://www.physiospot.com/2019/07/11/rehabilitation-2030-2-4-billion-people-are-in-need-of-rehabilitation-services/.

[3] A. Brunier and A. Muchnik. Who reveals leading causes of death and disability worldwide: 2000-2019. https://www.who.int/news/item/09-12-2020-who-reveals-leading-causes-of-death-and-disability-worldwide-2000-2019. World Heath Organization.

[4] M. Landry, S. Jaglal, W. Wodchis, J. Raman, and C. Cott, "Analysis of factors affecting demand for rehabilitation services in ontario, canada: A health-policy perspective," *Disability and rehabilitation*, vol. 30, pp. 1837–47, 02 2008.

[5] Z. Qian and Z. Bi, "Recent development of rehabilitation robots," *Hindawi Publishing Corporation, Advances in Mechanical Engineering*, 2014.

[6] T. Dall, P. Gallo, R. Chakrabarti, T. West, A. Semilla, and M. Storm, "An aging population and growing disease burden will require alarge and specialized health care workforce by 2025," *Health Affairs*, vol. 32:11, 2013.

[7] 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*, vol. 15, no. 1, pp. 7–20, 2003.

[8] A. S. Association. (Viewed 12 March 2021) About stroke. https://www.stroke.org/en/about-stroke.

[9] W. S. Organization, "Learn about stroke," https://www.world-stroke.org/world-stroke-day-campaign/why-stroke-matters/learn-about-stroke, Viewed 25 March 2021.

[10] C. Murray, "The lancet: Latest global disease estimates reveal perfect storm of rising chronic diseases and public health failures fuelling covid-19 pandemic," http://www.healthdata.org/news-release/lancet-latest-global-disease-estimates-reveal-perfect-storm-rising-chronic-diseases-and, Viewed 26 March 2021.

[11] C. O. Johnson, M. Nguyen, G. A. Roth, E. Nichols, T. Alam, D. Abate, F. Abd-Allah, A. Abdelalim, H. N. Abraha, N. M. Abu-Rmeileh *et al.*, "Global, regional, and national burden of stroke, 1990–2016: a systematic analysis for the global burden of disease study 2016," *The Lancet Neurology*, vol. 18, no. 5, pp. 439–458, 2019.

[12] P. Encarnacao and A. Cook, *Robotic Assistive Technologies: Principles and Practice*. CRC Press Taylor and Francis Group, 2017, ch. 3, pp. 71–96.

[13] A. Lioulemes *et al.*, "An intelligent multimodal upper-limb rehabilitation robotic system," Ph.D. dissertation, 2017.

[14] V. Lin, X. Zhang, and P. Dixon, "Occupational therapy workforce in the united states: Forecasting nationwide shortages," *PM&R*, vol. 7, no. 9, pp. 946–954, 2015.

[15] J. K. M. Wadman, J. Couzin-Frankel and C. Matacic, "How does coronavirus kill? clinicians trace a ferocious rampage through the body, from brain to toes," https://www.sciencemag.org/news/2020/ 04/how-does-coronavirus-kill-clinicians-trace-ferocious-rampage-through-body- brain-toes, Viewed 25 March 2021.

[16] J. K. M. Wadman, J. Couzin-Frankel, "Young and middle-aged people, barely sick with covid-19, are dying of strokes," https://www.washingtonpost.com/health/2020/04/24/strokes-coronavirus-young-patients/, 23 April 2020.

[17] B. Berkowitz. Running into trouble: Eager pandemic exercisers rack up injuries. [Online]. Available: https://www.washingtonpost.com/health/running-into-trouble-eager-pandemic-exercisers-rack-up-injuries/2020/05/29/c11c3910-9c73-11ea-ad09-8da7ec214672_story.html

[18] I. Y. Balkaya, J. A. Fernandez, W. Gerguis, M. T. Kaner, M. Lamagna, A. Lekshminarayanan, H. Meng, S. M. Mohar, S. Randev, I. Sanchez, S. Brahmbhatt, M. Islam, M. Frankenthaler, P. T. Diamond, and E. L. Altschuler, "When physical medicine and rehabilitation became medicine-life in the time of coronavirus disease of 2019," *American journal of physical medicine and rehabilitation*, vol. 99, no. 6, p. 480481, June 2020. [Online]. Available: https://europepmc.org/articles/PMC7253035

[19] W. C. for P. Therapy, "Wcpt launches ppe4pt advocacy campaign in support of access to personal protective equipment for physiotherapists," https://www.wcpt.org/node/178216, Viewed 26 April 2020.

[20] J. Kruger, "Jonathon kruger, wcpt chief executive officer, on wcpts advocacy campaign ppe4pt - youtube," https://www.youtube.com/watch?time_continue=117&v=AUcyLtZclsE&feature=emb_logo, Viewed 26 April 2020.

[21] A. P. T. Association, "Impact of covid-19 on the physical therapy profession," https://www.apta.org/contentassets/15ad5dc898a14d02b8257ab1cdb67f46/impact-of-covid-19-on-physical-therapy-profession.pdf, Viewed 27 March 2021.

[22] J. Kruger, "Physiotherapy consultations by phone available from monday," https://timesofmalta.com/articles/view/physiotherapy-consultations-by-phone-available-from-monday.787670, Viewed 26 April 2020.

[23] A. Turolla, G. Rossettini, A. Viceconti, A. Palese, and T. Geri, "Musculoskeletal Physical Therapy During the COVID-19 Pandemic: Is Telerehabilitation the Answer?" *Physical Therapy*, vol. 100, no. 8, pp. 1260–1264, 05 2020. [Online]. Available: https://doi.org/10.1093/ptj/pzaa093

[24] J. C. Fraile, J. Perez-Turiel, E. Baeyens, P. Vinas, R. Alonso, A. Cuadrado, M. Franco-Martin, E. Parra, L. Ayuso, F. García-Bravo *et al.*, "E2rebot: a robotic platform for upper limb rehabilitation in patients with neuromotor disability," *Advances in Mechanical Engineering*, vol. 8, no. 8, p. 1687814016659050, 2016.

[25] J. Medlej, "Human anatomy fundamentals: Flexibility and joint limitations," https://design.tutsplus.com/articles/human-anatomy-fundamentals-flexibility-and-joint-limitations--vector-25401, viewed 3rd May, 2017, created 2015.

[26] H. I. Krebs, L. Dipietro, S. Levy-Tzedek, S. E. Fasoli, A. Rykman-Berland, J. Zipse, J. A. Fawcett, J. Stein, H. Poizner, A. C. Lo *et al.*, "A paradigm shift for rehabilitation robotics," *IEEE engineering in medicine and biology magazine*, vol. 27, no. 4, pp. 61–70, 2008.

[27] M. Longhi, A. Merlo, P. Prati, M. Giacobbi, and D. Mazzoli, "Instrumental indices for upper limb function assessment in stroke patients: a validation study," *Journal of neuroengineering and rehabilitation*, vol. 13, no. 1, pp. 1–11, 2016.

[28] K. Baur, N. Rohrbach, J. Hermsdörfer, R. Riener, and V. Klamroth-Marganska, "The beam-me-in strategy–remote haptic therapist-patient interaction with two exoskeletons for stroke therapy," *Journal of neuroengineering and rehabilitation*, vol. 16, no. 1, p. 85, 2019.

[29] H. M. Van der Loos, D. J. Reinkensmeyer, and E. Guglielmelli, "Rehabilitation and health care robotics," in *Springer handbook of robotics*. Springer, 2016, pp. 1685–1728.

[30] C. J. Mottram, N. L. Suresh, C. Heckman, M. A. Gorassini, and W. Z. Rymer, "Origins of abnormal excitability in biceps brachii motoneurons of spastic-paretic stroke survivors," *Journal of neurophysiology*, vol. 102, no. 4, pp. 2026–2038, 2009.

[31] R. Balliet, B. Levy, and K. Blood, "Upper extremity sensory feedback therapy in chronic cerebrovascular accident patients with impaired expressive aphasia and auditory comprehension," *Arch Phys Med Rehabil.*, vol. 67(5), pp. 304–310, 1986 May.

[32] W. van Niekerk, T. van der Stockt, K. Jackson, and N. O'Reilly. Rehabilitation during a pandemic in people with specific rehabilitation needs. [Online]. Available: https://physio-pedia.com/Rehabilitation_During_a_Pandemic_in_People_with_Specific_Rehabilitation_Needs?utm_source=physiopedia&utm_medium=related_articles&utm_campaign=ongoing_internal

[33] S. P. Buerger, J. J. Palazzolo, H. I. Krebs, and N. Hogan, "Rehabilitation robotics: adapting robot behavior to suit patient needs and abilities," in *Proceedings of the 2004 American Control Conference*, vol. 4. IEEE, 2004, pp. 3239–3244.

[34] L. Marchal-Crespo and D. J. Reinkensmeyer, "Review of control strategies for robotic movement training after neurologic injury," *Journal of neuroengineering and rehabilitation*, vol. 6, no. 1, pp. 1–15, 2009.

[35] R. Calabro, M. Russo, A. Naro, D. Milardi, T. Balletta, A. Leo, S. Filoni, and P. Bramanti, "Who may benefit from armeo power treatment - a neurophysiological approach to predict neurorehabilitation outcomes," *PMandR*, vol. 8, no. 10, pp. 971 – 978, 2016. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1934148216000745

[36] B. B. Johansson, "Brain plasticity and stroke rehabilitation: the willis lecture," *Stroke*, vol. 31, no. 1, pp. 223–230, 2000.

[37] Y. Hara, "Brain plasticity and rehabilitation in stroke patients," *Journal of Nippon Medical School*, vol. 82, no. 1, pp. 4–13, 2015.

[38] S. E. Fasoli, H. I. Krebs, and N. Hogan, "Robotic technology and stroke rehabilitation: translating research into practice," *Topics in stroke Rehabilitation*, vol. 11, no. 4, pp. 11–19, 2004.

[39] 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*, vol. 11, no. 1, p. 3, 2014.

[40] R. Gopura, D. Bandara, K. Kiguchi, and G. K. Mann, "Developments in hardware systems of active upper-limb exoskeleton robots: A review," *Robotics and Autonomous Systems*, vol. 75, pp. 203–220, 2016.

[41] E. Palermo, D. R. Hayes, E. F. Russo, R. S. Calabrò, A. Pacilli, and S. Filoni, "Translational effects of robot-mediated therapy in subacute stroke patients: an experimental evaluation of upper limb motor recovery," *PeerJ*, vol. 6, p. e5544, 2018.

[42] G. Gardiner. Composites in exoskeletons. [Online]. Available: https://www.compositesworld.com/articles/composites-in-exoskeletons

[43] J. Brackenridge, L. V Bradnam, S. Lennon, J. J Costi, and D. A Hobbs, "A review of rehabilitation devices to promote upper limb function following stroke," *Neuroscience and Biomedical Engineering (Discontinued)*, vol. 4, no. 1, pp. 25–42, 2016.

[44] A. Basteris, S. M. Nijenhuis, A. H. Stienen, J. H. Buurke, G. B. Prange, and F. Amirabdollahian, "Training modalities in robot-mediated upper limb rehabilitation in stroke: a framework for classification based on a systematic review," *Journal of neuroengineering and rehabilitation*, vol. 11, no. 1, pp. 1–15, 2014.

[45] J. Laut, M. Porfiri, and P. Raghavan, "The present and future of robotic technology in rehabilitation," *Current physical medicine and rehabilitation reports*, vol. 4, no. 4, pp. 312–319, 2016.

[46] A. Radman, W. Ismail, and M. Bahari, "Robotic devices for upper limb stroke rehabilitation: Potential research trends," in *2017 IEEE International Symposium on Robotics and Intelligent Sensors (IRIS)*. IEEE, 2017, pp. 383–388.

[47] T. Nef, M. Mihelj, and R. Riener, "Armin: a robot for patient-cooperative arm therapy," *Medical & biological engineering & computing*, vol. 45, no. 9, pp. 887–900, 2007.

[48] A. Frisoli, F. Salsedo, M. Bergamasco, B. Rossi, and M. C. Carboncini, "A force-feedback exoskeleton for upper-limb rehabilitation in virtual reality," *Applied Bionics and Biomechanics*, vol. 6, no. 2, pp. 115–126, 2009.

[49] S. S. Conroy, J. Whitall, L. Dipietro, L. M. Jones-Lush, M. Zhan, M. A. Finley, G. F. Wittenberg, H. I. Krebs, and C. T. Bever, "Effect of gravity on robot-assisted motor training after chronic stroke: a randomized trial," *Archives of physical medicine and rehabilitation*, vol. 92, no. 11, pp. 1754–1761, 2011.

[50] InMotion, https://www.bioniklabs.com/products/inmotion-arm, Viewed 29 March 2021.

[51] M. Nova, "The motus hand," https://motusnova.com/hand/, Viewed 29 March 2021.

[52] F. Gaming, "Hand mentor uses game-based therapy to restore hand function in stroke survivors," https://www.fitness-gaming.com/news/health-and-rehab/hand-mentor-uses-game-based-therapy-to-restore-hand-function-in-stroke-survivors.html, Viewed 29 March 2021.

[53] Q. Miao, M. Zhang, J. Cao, and S. Q. Xie, "Reviewing high-level control techniques on robot-assisted upper-limb rehabilitation," *Advanced Robotics*, vol. 32, no. 24, pp. 1253–1268, 2018.

[54] 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*, vol. 9, pp. 4–14, 2016.

[55] F. DeRuyter and D. A. Olson, *Clinician's Guide to Assistive Technology*. Mosby, 2002.

[56] Teprel, "Hocoma - armeo boom," https://teprel.pt/produto/hocoma-armeo-boom/, Viewed 29 March 2021.

[57] R. Q. Van der Linde, P. Lammertse, E. Frederiksen, and B. Ruiter, "The hapticmaster, a new high-performance haptic interface," in *Proc. Eurohaptics*. Edinburgh University, 2002, pp. 1–5.

[58] L. Bishop and J. Stein, "Three upper limb robotic devices for stroke rehabilitation: a review and clinical perspective," *NeuroRehabilitation*, vol. 33, no. 1, pp. 3–11, 2013.

[59] M. Sharifi, S. Behzadipour, H. Salarieh, and M. Tavakoli, "Cooperative modalities in robotic tele-rehabilitation using nonlinear bilateral impedance control," *Control Engineering Practice*, vol. 67, pp. 52–63, 2017.

[60] M. Sharifi, S. Behzadipour, and H. Salarieh, "Assist-as-needed policy for movement therapy using telerobotics-mediated therapist supervision," *Control Engineering Practice*, vol. 101, p. 104481, 2020.

[61] H. Wang and Y. Xie, "Adaptive inverse dynamics control of robots with uncertain kinematics and dynamics," *Automatica*, vol. 45, no. 9, pp. 2114–2119, 2009.

[62] M. Mendoza, I. Bonilla, E. Gonzalez-Galvan, and F. Reyes, "Impedance control in a wave-based teleoperator for rehabilitation motor therapies assisted by robots," *Computer methods and programs in biomedicine*, vol. 123, pp. 54–67, 2016.

[63] X. Wang and J. Li, "A tele-rehabilitation system with bilateral haptic feedback to both the therapist and the patient via time-delay environment," in *2013 World Haptics Conference (WHC)*. IEEE, 2013, pp. 331–334.

[64] D. Buongiorno, D. Chiaradia, S. Marcheschi, M. Solazzi, and A. Frisoli, "Multidofs exoskeleton-based bilateral teleoperation with the time-domain passivity approach," *Robotica*, vol. 37, no. 9, pp. 1641–1662, 2019.

[65] J. Lanini, T. Tsuji, P. Wolf, R. Riener, and D. Novak, "Teleoperation of two six-degree-of-freedom arm rehabilitation exoskeletons," in *2015 IEEE International Conference on Rehabilitation Robotics (ICORR)*. IEEE, 2015, pp. 514–519.

[66] G. Bauer and Y.-J. Pan, "Telerehabilitation with exoskeletons using adaptive robust integral rbf-neural-network impedance control under variable time delays," in *(Accepted) 2021 IEEE International Symposium on Industrial Electronics (ISIE)*. IEEE, 2021.

[67] S. Zhang and S. Guo, "Performance evaluation of a novel telerehabilitation system for the elbow joint training," in *2015 IEEE International Conference on Mechatronics and Automation (ICMA)*. IEEE, 2015, pp. 420–424.

[68] K.-Y. Wu, Y.-Y. Su, Y.-L. Yu, C.-H. Lin, and C.-C. Lan, "A 5-degrees-of-freedom lightweight elbow-wrist exoskeleton for forearm fine-motion rehabilitation," *IEEE/ASME Transactions on Mechatronics*, vol. 24, no. 6, pp. 2684–2695, 2019.

[69] B. Brahmi, M. Driscoll, I. K. El Bojairami, M. Saad, and A. Brahmi, "Novel adaptive impedance control for exoskeleton robot for rehabilitation using a nonlinear time-delay disturbance observer," *ISA transactions*, vol. 108, pp. 381–392, 2021.

[70] G. Bauer, Y.-J. Pan, and H. Shen, "Adaptive impedance control in bilateral telerehabilitation with robotic exoskeletons," in *2020 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. IEEE, 2020, pp. 719–725.

[71] A. Chowdhury, S. S. Nishad, Y. K. Meena, A. Dutta, and G. Prasad, "Hand-exoskeleton assisted progressive neurorehabilitation using impedance adaptation based challenge level adjustment method," *IEEE transactions on haptics*, vol. 12, no. 2, pp. 128–140, 2018.

[72] R. S. Souza, T. de Castro Martins, G. P. Furtado, and A. Forner-Cordero, "Model reference adaptive impedance controller design for modular exoskeleton," *IFAC-PapersOnLine*, vol. 51, no. 27, pp. 345–349, 2018.

[73] J. Liu, *Radial Basis Function (RBF) neural network control for mechanical systems: design, analysis and Matlab simulation*. Springer Science & Business Media, 2013.

[74] G. Bauer and Y.-J. Pan, "Review of control methods for upper limb telerehabilitation with robotic exoskeletons," *IEEE Access*, vol. 8, no. submitted, pp. 203 382–203 397, 2020.

[75] Tyromotion, "Diego tyromotion," Aug. 2019. [Online]. Available: https://tyromotion.com/en/produkte/diego/

[76] E. Pirondini, M. Coscia, S. Marcheschi, G. Roas, F. Salsedo, A. Frisoli, M. Bergamasco, and S. Micera, "Evaluation of a new exoskeleton for upper limb post-stroke neuro-rehabilitation: Preliminary results," in *Replace, Repair, Restore, Relieve–Bridging Clinical and Engineering Solutions in Neurorehabilitation*. Springer, 2014, pp. 637–645.

[77] https://www.hocoma.com/us/solutions/armeo-power/, Viewed 14 April 2017.

[78] G. Bauer, Y.-J. Pan, and R. Adamson, "Analysis of a low-cost sensor towards an emg-based robotic exoskeleton controller," in *CSME 2018 International Congress Proceedings*. CSME-SCGM, 2018.

[79] P. Poli, G. Morone, G. Rosati, and S. Masiero, "Robotic technologies and rehabilitation: new tools for stroke patients therapy," *BioMed Research International*, vol. 2013, 2013.

[80] C. Moro, S. Lin, G. Nejat, and A. Mihailidis, "Social robots and seniors: a comparative study on the influence of dynamic social features on human–robot interaction," *International Journal of Social Robotics*, vol. 11, no. 1, pp. 5–24, 2019.

[81] S. L. Wolf, K. Sahu, R. C. Bay, S. Buchanan, A. Reiss, S. Linder, A. Rosenfeldt, and J. Alberts, "The haapi (home arm assistance progression initiative) trial: a novel robotics delivery approach in stroke rehabilitation," *Neurorehabilitation and neural repair*, vol. 29, no. 10, pp. 958–968, 2015.

[82] A. Vitt, "Machine learning in digital telerehabilitation: Telerehabilitation system based on kinect," in *Dissertion*, 2015.

[83] E. Ivanova, K. Lorenz, M. Schrader, and M. Minge, "Developing motivational visual feedback for a new telerehabilitation system for motor relearning after stroke," in *Proceedings of the 31st International BCS Human Computer Interaction Conference (HCI 2017) 31*, 2017, pp. 1–6.

[84] G. Airò Farulla, D. Pianu, M. Cempini, M. Cortese, L. O. Russo, M. Indaco, R. Nerino, A. Chimienti, C. M. Oddo, and N. Vitiello, "Vision-based pose estimation for robot-mediated hand telerehabilitation," *Sensors*, vol. 16, no. 2, p. 208, 2016.

[85] K. E. Laver, Z. Adey-Wakeling, M. Crotty, N. A. Lannin, S. George, and C. Sherrington, "Telerehabilitation services for stroke," *Cochrane Database of Systematic Reviews*, no. 1, 2020.

[86] S. M. Nijenhuis, G. B. Prange, F. Amirabdollahian, P. Sale, F. Infarinato, N. Nasr, G. Mountain, H. J. Hermens, A. H. Stienen, J. H. Buurke *et al.*, "Feasibility study into self-administered training at home using an arm and hand device with motivational gaming environment in chronic stroke," *Journal of neuroengineering and rehabilitation*, vol. 12, no. 1, pp. 1–13, 2015.

[87] M. Sivan, J. Gallagher, S. Makower, D. Keeling, B. Bhakta, R. J. OConnor, and M. Levesley, "Home-based computer assisted arm rehabilitation (hcaar) robotic device for upper limb exercise after stroke: results of a feasibility study in home setting," *Journal of neuroengineering and rehabilitation*, vol. 11, no. 1, pp. 1–17, 2014.

[88] D. Perez-Marcos, M. Solazzi, W. Steptoe, W. Oyekoya, A. Frisoli, T. Weyrich, A. Steed, F. Tecchia, M. Slater, and M. V. Sanchez-Vives, "A fully immersive set-up for remote interaction and neurorehabilitation based on virtual body ownership," *Frontiers in neurology*, vol. 3, p. 110, 2012.

[89] S. Handelzalts, G. Ballardini, C. Avraham, M. Pagano, M. Casadio, and I. Nisky, "Integrating tactile feedback technologies into home-based telerehabilitation: Opportunities and challenges in light of covid-19 pandemic," *Frontiers in Neurorobotics*, vol. 15, p. 4, 2021.

[90] C. R. Carignan and H. I. Krebs, "Telerehabilitation robotics: bright lights, big future?" *Journal of rehabilitation research and development*, vol. 43, no. 5, p. 695, 2006.

[91] L. J. Consoni, A. A. Siqueira, and H. I. Krebs, "Compensating for telecommunication delays during robotic telerehabilitation," in *2017 International Conference on Rehabilitation Robotics (ICORR)*. IEEE, 2017, pp. 812–817.

[92] M. Shahbazi, S. F. Atashzar, and R. V. Patel, "A systematic review of multilateral teleoperation systems," *IEEE transactions on haptics*, vol. 11, no. 3, pp. 338–356, 2018.

[93] L. Peng, C. Wang, L. Luo, S. Chen, Z.-G. Hou, and W. Wang, "Adaptive modeling and control of an upper-limb rehabilitation robot using rbf neural networks," in *International Conference on Neural Information Processing*. Springer, 2018, pp. 235–245.

[94] J. Zariffa, N. Kapadia, J. Kramer, P. Taylor, M. Alizadeh-Meghrazi, V. Zivanovic, R. Willms, A. Townson, A. Curt, M. Popovic *et al.*, "Feasibility and efficacy of upper limb robotic rehabilitation in a subacute cervical spinal cord injury population," *Spinal Cord*, vol. 50, no. 3, pp. 220–226, 2012.

[95] P. Seitz. Robotic exoskeletons stepping out of science fiction. [Online]. Available: https://www.investors.com/news/technology/robotic-exoskeletons-stepping-out-science-fiction/

[96] Robotic exoskeleton market rising. [Online]. Available: https://www.compositesworld.com/cdn/cms/exo-units.jpg

[97] S. Mulholland, "Stroke arm and hand management," https://www.strokenetworkseo.ca, 2013, accessed: 2020-10-25.

[98] J.-J. E. Slotine, W. Li *et al.*, *Applied nonlinear control*. Prentice hall Englewood Cliffs, NJ, 1991, vol. 199, no. 1.

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

[100] B. Brahmi, A. Brahmi, M. Saad, G. Gauthier, and M. Habibur Rahman, "Robust adaptive tracking control of uncertain rehabilitation exoskeleton robot," *Journal of Dynamic Systems, Measurement, and Control*, vol. 141, no. 12, 2019.

[101] F. Just, K. Baur, R. Riener, V. Klamroth-Marganska, and G. Rauter, "Online adaptive compensation of the armin rehabilitation robot," in *2016 6th IEEE International Conference on Biomedical Robotics and Biomechatronics (BioRob)*. IEEE, 2016, pp. 747–752.

[102] H.-B. Kang and J.-H. Wang, "Adaptive robust control of 5 dof upper-limb exoskeleton robot," *International Journal of Control, Automation and Systems*, vol. 13, no. 3, pp. 733–741, 2015.

[103] M. R. Islam, C. Spiewak, M. H. Rahman, and R. Fareh, "A brief review on robotic exoskeletons for upper extremity rehabilitation to find the gap between research porotype and commercial type," *Adv Robot Autom*, vol. 6, no. 3, p. 2, 2017.

[104] K. D. Young, V. I. Utkin, and U. Ozguner, "A control engineer's guide to sliding mode control," *IEEE transactions on control systems technology*, vol. 7, no. 3, pp. 328–342, 1999.

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

[106] M. H. Rahman, M. Saad, J.-P. Kenné, and P. S. Archambault, "Control of an exoskeleton robot arm with sliding mode exponential reaching law," *International Journal of Control, Automation and Systems*, vol. 11, no. 1, pp. 92–104, 2013.

[107] H. Aschemann and D. Schindele, "Sliding-mode control of a high-speed linear axis driven by pneumatic muscle actuators," *IEEE Transactions on Industrial Electronics*, vol. 55, no. 11, pp. 3855–3864, 2008.

[108] D. Yun, A. M. Khan, R.-J. Yan, Y. Ji, H. Jang, J. Iqbal, K. Zuhaib, J. Y. Ahn, J. Han, and C. Han, "Handling subject arm uncertainties for upper limb rehabilitation robot using robust sliding mode control," *International Journal of Precision Engineering and Manufacturing*, vol. 17, no. 3, pp. 355–362, 2016.

[109] B. Brahmi, M. Saad, C. O. Luna, P. Archambault, and M. Rahman, "Sliding mode control of an exoskeleton robot based on time delay estimation," in *2017 International Conference on Virtual Rehabilitation (ICVR)*. IEEE, 2017, pp. 1–2.

[110] Y. Chen, J. Liu, H. Wang, Z. Pan, and S. Han, "Model-free based adaptive rbf neural network control for a rehabilitation exoskeleton," in *2019 Chinese Control And Decision Conference (CCDC)*. IEEE, 2019, pp. 4208–4213.

[111] W. Yu, X. Li, and R. Carmona, "A novel pid tuning method for robot control," *Industrial Robot: An International Journal*, 2013.

[112] Q. Wu, X. Wang, F. Du, and Q. Zhu, "Fuzzy sliding mode control of an upper limb exoskeleton for robot-assisted rehabilitation," in *2015 IEEE International Symposium on Medical Measurements and Applications (MeMeA) Proceedings*. IEEE, 2015, pp. 451–456.

[113] Z. Chen, F. Huang, W. Sun, J. Gu, and B. Yao, "Rbf-neural-network-based adaptive robust control for nonlinear bilateral teleoperation manipulators with uncertainty and time delay," *IEEE/ASME Transactions on Mechatronics*, vol. 25, no. 2, pp. 906–918, 2019.

[114] D. Richert, C. J. Macnab, and J. K. Pieper, "Adaptive haptic control for telerobotics transitioning between free, soft, and hard environments," *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, vol. 42, no. 3, pp. 558–570, 2011.

[115] P. Agarwal and A. D. Deshpande, "Subject-specific assist-as-needed controllers for a hand exoskeleton for rehabilitation," *IEEE Robotics and Automation Letters*, vol. 3, no. 1, pp. 508–515, 2017.

[116] Q. Cao, C. Cao, F. Wang, D. Liu, and H. Sun, "Robust adaptive full-order tsm control based on neural network," *Symmetry*, vol. 10, no. 12, p. 726, 2018.

[117] M. Hamaya, T. Matsubara, T. Noda, T. Teramae, and J. Morimoto, "Learning assistive strategies from a few user-robot interactions: Model-based reinforcement learning approach," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2016, pp. 3346–3351.

[118] L.-J. Xu, C.-Y. Dong, and Y. Chen, "An adaptive fuzzy sliding mode control for networked control systems," in *2007 International Conference on Mechatronics and Automation*. IEEE, 2007, pp. 1190–1195.

[119] A. Gupta and M. K. OMalley, "Robotic exoskeletons for upper extremity rehabilitation," *Rehabilitation Robotics*, vol. 10, p. 5171, 2007.

[120] R. Riener, L. Lünenburger, and G. Colombo, "Human-centered robotics applied to gait training and assessment." *Journal of Rehabilitation Research & Development*, vol. 43, no. 5, 2006.

[121] K. Kiguchi and Y. Hayashi, "An emg-based control for an upper-limb power-assist exoskeleton robot," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 42, no. 4, pp. 1064–1071, 2012.

[122] J. A. Aderonmu, "Emerging challenges in meeting physiotherapy needs during covid-19 through telerehabilitation," *Bulletin of Faculty of Physical Therapy*, vol. 25, no. 1, pp. 1–4, 2020.

[123] K. D. Knepley, J. Z. Mao, P. Wieczorek, F. O. Okoye, A. P. Jain, and N. Y. Harel, "Impact of telerehabilitation for stroke-related deficits," *Telemedicine and e-Health*, 2020.

[124] G. Niemeyer, C. Preusche, and G. Hirzinger, "Springer handbook of robotics," in *Telerobotics*. Springer, 2008, pp. 741–757.

[125] S. Hirche, M. Ferre, J. Barrio, C. Melchiorri, and M. Buss, "Bilateral control architectures for telerobotics," in *Advances in Telerobotics*. Springer, 2007, pp. 163–176.

[126] T. B. Sheridan, "Teleoperation, telerobotics and telepresence: A progress report," *Control Engineering Practice*, vol. 3, no. 2, pp. 205–214, 1995.

[127] M. Rietzler, G. Haas, T. Dreja, F. Geiselhart, and E. Rukzio, "Virtual muscle force: Communicating kinesthetic forces through pseudo-haptic feedback and muscle input," in *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology*, ser. UIST '19. New York, NY, USA: Association for Computing Machinery, 2019, p. 913922.

[128] H.-S. Park, Q. Peng, and L.-Q. Zhang, "A portable telerehabilitation system for remote evaluations of impaired elbows in neurological disorders," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 16, no. 3, pp. 245–254, 2008.

[129] X. Zhong and Y. ShiQiang, "A review of on-orbit servicing robot teleoperation control system," *International Journal of Signal Processing, Image Processing and Pattern Recognition*, vol. 9, no. 11, pp. 83–92, 2016.

[130] M. Serna, L. G. Garcia-Valdovinos, T. Salgado-Jimenez, and M. Bandala-Sanchez, "Bilateral teleoperation of a commercial small-sized underwater vehicle for academic purposes," in *OCEANS 2015-MTS/IEEE Washington*. IEEE, 2015, pp. 1–5.

[131] W. Wei and Y. Kui, "Teleoperated manipulator for leak detection of sealed radioactive sources," in *IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA'04. 2004*, vol. 2. IEEE, 2004, pp. 1682–1687.

[132] H. Sun, G. Song, Z. Wei, Y. Zhang, and S. Liu, "Bilateral teleoperation of an unmanned aerial vehicle for forest fire detection," in *2017 IEEE International Conference on Information and Automation (ICIA)*. IEEE, 2017, pp. 586–591.

[133] D. Lee, A. Franchi, P. R. Giordano, H. I. Son, and H. H. Bülthoff, "Haptic teleoperation of multiple unmanned aerial vehicles over the internet," in *2011 IEEE International Conference on Robotics and Automation*. IEEE, 2011, pp. 1341–1347.

[134] E. Slawiñski, V. Moya, D. Santiago, and V. Mut, "Force and position–velocity coordination for delayed bilateral teleoperation of a mobile robot," *Robotica*, vol. 37, no. 10, pp. 1768–1784, 2019.

[135] Y. Dong and N. Chopra, "Passivity-based bilateral tele-driving system with parametric uncertainty and communication delays," *IEEE Control Systems Letters*, vol. 3, no. 2, pp. 350–355, 2018.

[136] P. F. Hokayem and M. W. Spong, "Bilateral teleoperation: An historical survey," *Automatica*, vol. 42, no. 12, pp. 2035–2057, 2006.

[137] E. Nuño, L. Basañez, and R. Ortega, "Passivity-based control for bilateral teleoperation: A tutorial," *Automatica*, vol. 47, no. 3, pp. 485–495, 2011.

[138] S. G. Tzafestas and A.-I. Mantelos, "Time delay and uncertainty compensation in bilateral telerobotic systems: State-of-art with case studies," in *Engineering Creative Design in Robotics and Mechatronics*. IGI Global, 2013, pp. 208–238.

[139] E. J. Rodriguez-Seda, D. Lee, and M. W. Spong, "Experimental comparison study of control architectures for bilateral teleoperators," *IEEE Transactions on robotics*, vol. 25, no. 6, pp. 1304–1318, 2009.

[140] G. Niemeyer and J.-J. E. Slotine, "Telemanipulation with time delays," *The International Journal of Robotics Research*, vol. 23, no. 9, pp. 873–890, 2004.

[141] P. Arcara and C. Melchiorri, "Control schemes for teleoperation with time delay: A comparative study," *Robotics and Autonomous systems*, vol. 38, no. 1, pp. 49–64, 2002.

[142] R. Muradore and P. Fiorini, "A review of bilateral teleoperation algorithms," *Acta Polytechnica Hungarica*, vol. 13, no. 1, pp. 191–208, 2016.

[143] P. M. Kebria, H. Abdi, M. M. Dalvand, A. Khosravi, and S. Nahavandi, "Control methods for internet-based teleoperation systems: A review," *IEEE Transactions on Human-Machine Systems*, vol. 49, no. 1, pp. 32–46, 2018.

[144] M. D. Duong, K. Terashima, and T. Miyoshi, "A novel stable teleoperation with haptic feedback by means of impedance adjustment via arbitrary time delay environment for rehabilitation," in *2009 IEEE Control Applications,(CCA) & Intelligent Control,(ISIC)*. IEEE, 2009, pp. 1744–1749.

[145] J. Ware, Y.-J. Pan, and T. Hilliard, "A new bilaterally teleoperated robotic vehicle platform with passivity control," in *Proceedings of the 2011 American Control Conference*. IEEE, 2011, pp. 2837–2842.

[146] M. Zou, "Nonlinear adaptive robust control for bilateral teleoperated robotic manipulators with arbitrary time delays," 2017.

[147] J. H. Park and H. C. Cho, "Sliding-mode controller for bilateral teleoperation with varying time delay," in *1999 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (Cat. No. 99TH8399)*. IEEE, 1999, pp. 311–316.

[148] U. Ahmad and Y.-J. Pan, "A time domain passivity approach for asymmetric multilateral teleoperation system," *IEEE Access*, vol. 6, pp. 519–531, 2017.

[149] Y. Ye, Y.-J. Pan, and Y. Gupta, "A power based time domain passivity control for haptic interfaces," in *Proceedings of the 48h IEEE Conference on Decision and Control (CDC) held jointly with 2009 28th Chinese Control Conference*. IEEE, 2009, pp. 7521–7526.

[150] K. Wladron and J. Schmiedeler, *Kinematics*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 1223–1251. [Online]. Available: http://dx.doi.org/10.1007/978-3-540-30301-5_54

[151] H. K. Khalil, *Nonlinear systems*, 3rd ed. Upper Saddle River, N.J.: Prentice Hall, 2002.

[152] T. Glad and L. Ljung, *Control theory*. CRC press, 2000.

[153] J.-J. E. Slotine, W. Li *et al.*, *Applied nonlinear control*. Prentice hall Englewood Cliffs, NJ, 1991, vol. 199, no. 1.

[154] C. Desoer and M. Vidyasagar, "Feedback systems: Input-output prop-erties,˜: Acadenric," *1975*, 1975.

[155] D. Piovesan, A. Pierobon, P. DiZio, and J. R. Lackner, "Experimental measure of arm stiffness during single reaching movements with a time-frequency analysis," *Journal of neurophysiology*, vol. 110, no. 10, pp. 2484–2496, 2013.

[156] J. Mizrahi, "Erratum to: Review: Mechanical impedance and its relations to motor control, limb dynamics, and motion biomechanics," *Journal of Medical and Biological Engineering*, vol. 36, no. 2, pp. 282–282, 2016.

[157] M. K. Duong, H. Cheng, H. T. Tran, and Q. Jing, "Minimizing human-exoskeleton interaction force using compensation for dynamic uncertainty error with adaptive rbf network," *Journal of Intelligent & Robotic Systems*, vol. 82, no. 3-4, pp. 413–433, 2016.

[158] M. Vidyasagar and M. Spong, *Robot dynamics and control*. Wiley, 1989.

[159] C. Yang, X. Wang, Z. Li, Y. Li, and C.-Y. Su, "Teleoperation control based on combination of wave variable and neural networks," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 47, no. 8, pp. 2125–2136, 2016.

[160] www.https://www.quanser.com/products/qube-servo-2/, Viewed 17 April 2019.

[161] J. C. Lagarias, J. A. Reeds, M. H. Wright, and P. E. Wright, "Convergence properties of the nelder–mead simplex method in low dimensions," *SIAM Journal on optimization*, vol. 9, no. 1, pp. 112–147, 1998.

[162] OptoForce, "3-axis force sensor, omd-20-fg-100n data sheet," 2016.

[163] M. Agarana, O. Ajayi, T. Anake, and S. Bishop, "Mathematical analysis of moment of inertia of human arm at fixed position," in *Proceedings of the World Congress on Engineering and Computer Science*, vol. 2, 2017.

[164] P. De Leva, "Adjustments to zatsiorsky-seluyanov's segment inertia parameters," *Journal of biomechanics*, vol. 29, no. 9, pp. 1223–1230, 1996.

[165] C. Basdogan, C.-H. Ho, M. A. Srinivasan, and M. Slater, "An experimental study on the role of touch in shared virtual environments," *ACM Transactions on Computer-Human Interaction (TOCHI)*, vol. 7, no. 4, pp. 443–460, 2000.

[166] K. Ogata, *Modern control engineering*. Prentice hall, 2010.

[167] J.-J. E. Slotine and W. Li, "On the adaptive control of robot manipulators," *The international journal of robotics research*, vol. 6, no. 3, pp. 49–59, 1987.

[168] Y. Ren, H. S. Park, Y. Li, L. Wang, and L.-Q. Zhang, "A wearable robot for upper limb rehabilitation of patients with neurological disorders," in *2010 IEEE International Conference on Robotics and Biomimetics*. IEEE, 2010, pp. 64–68.

[169] D. T. Pham and X. Liu, "Neural network controllers," in *Neural Networks for Identification, Prediction and Control*. Springer, 1995, pp. 111–130.

Appendix A

ARII and RBFNN-I Case Studies Plots

A.1 Quanser QUBE Experimental Case Studies

A.1.1 Quanser QUBE Experimental Case Studies with ARII Control Methods

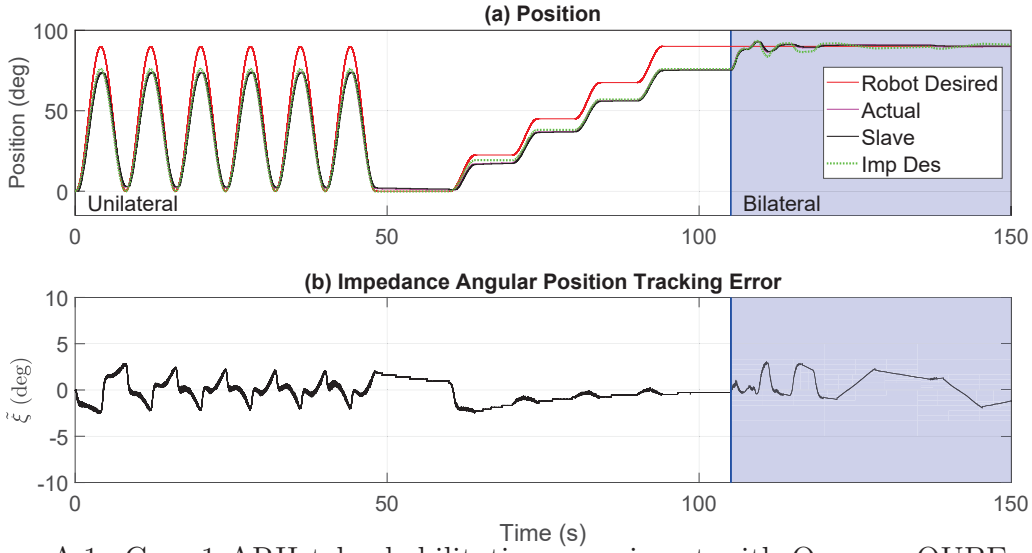


Figure A.1: Case 1 ARII telerehabilitation experiment with Quanser QUBEs using HTR: (a) position trajectory, (b) tracking error

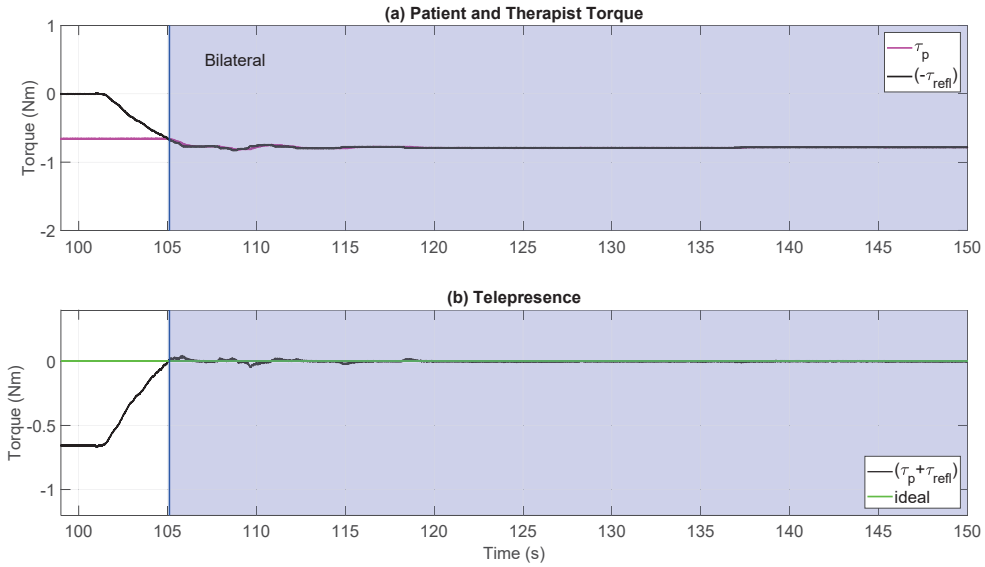


Figure A.2: Torques for case 1 ARII telerehabilitation experiment with Quanser QUBEs with HTR during the bilateral phase

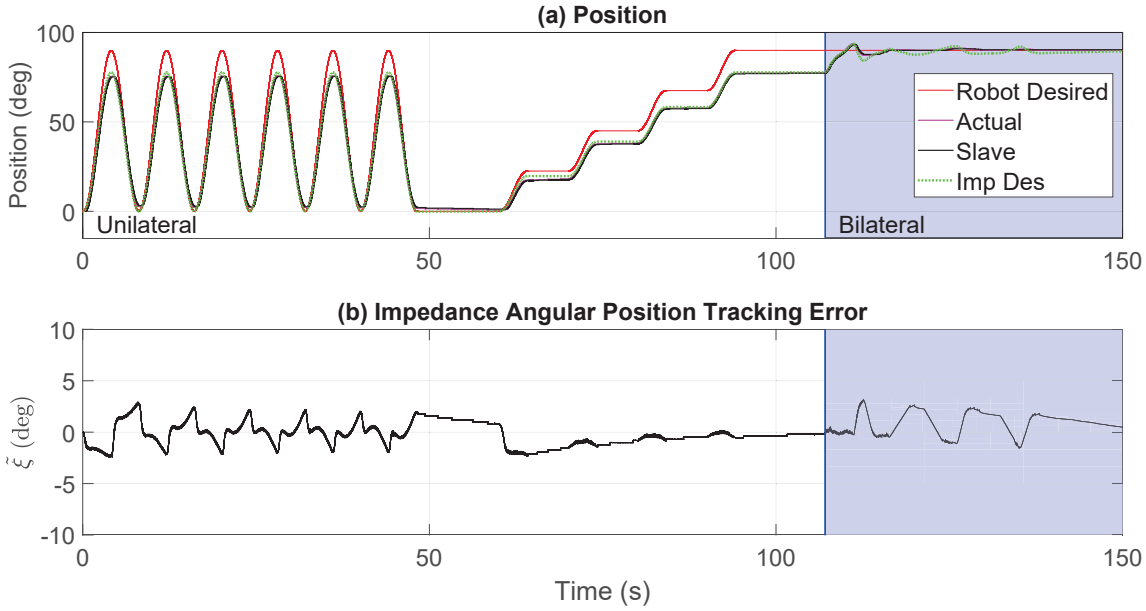


Figure A.3: Case 2 ARII telerehabilitation experiment with Quanser QUBEs using HTR: (a) position trajectory, (b) tracking error

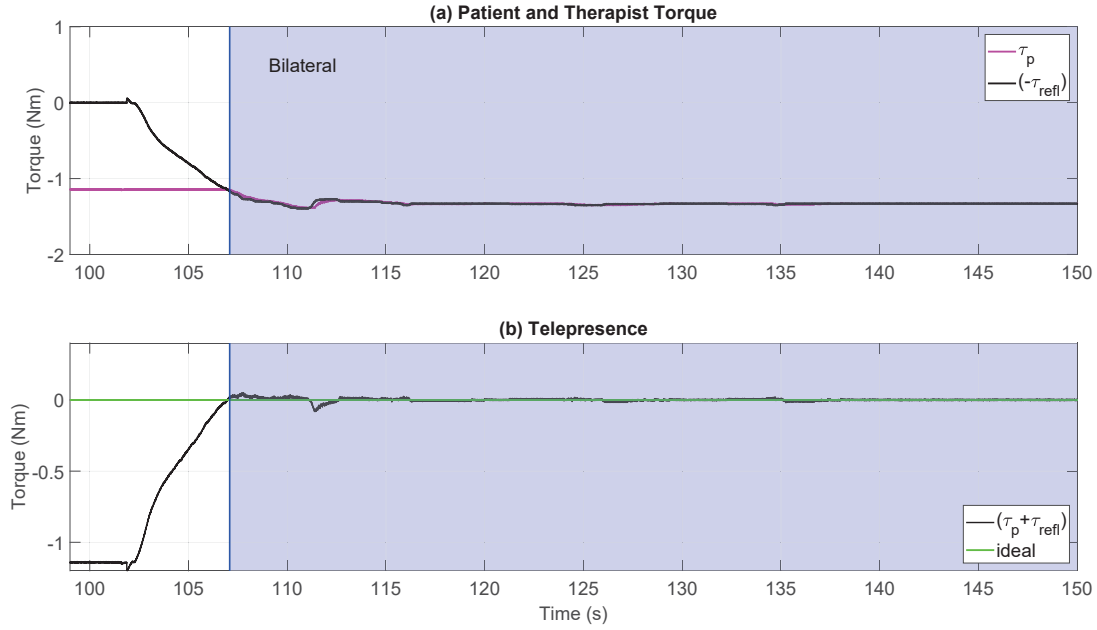


Figure A.4: Torques for case 2 ARII telerehabilitation experiment with Quanser QUBEs with HTR during the bilateral phase

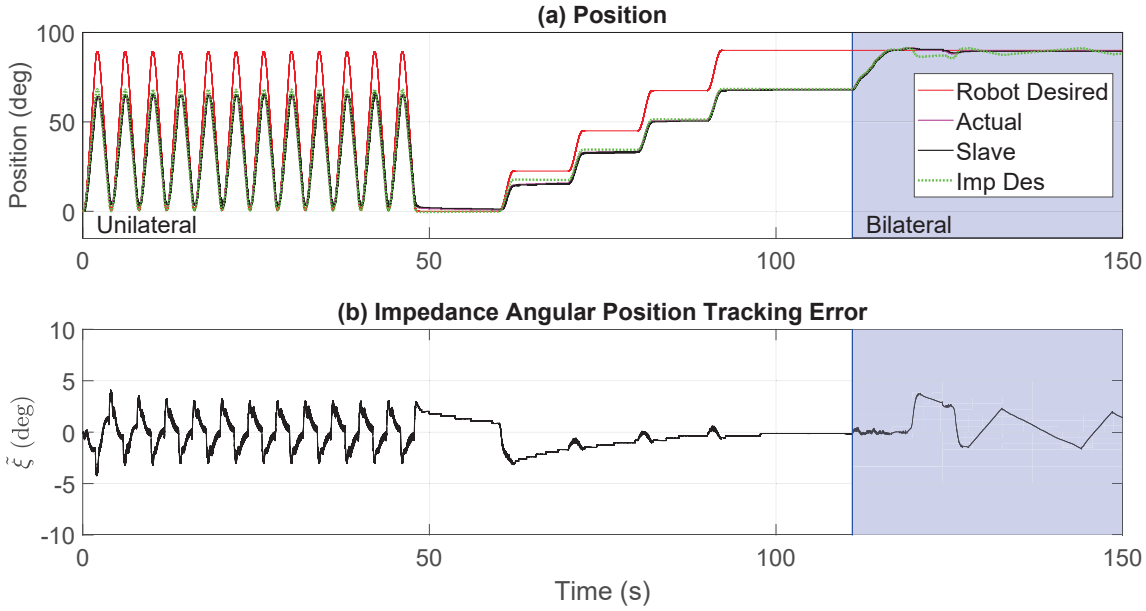


Figure A.5: Case 3 ARII telerehabilitation experiment with Quanser QUBEs using HTR: (a) position trajectory, (b) tracking error

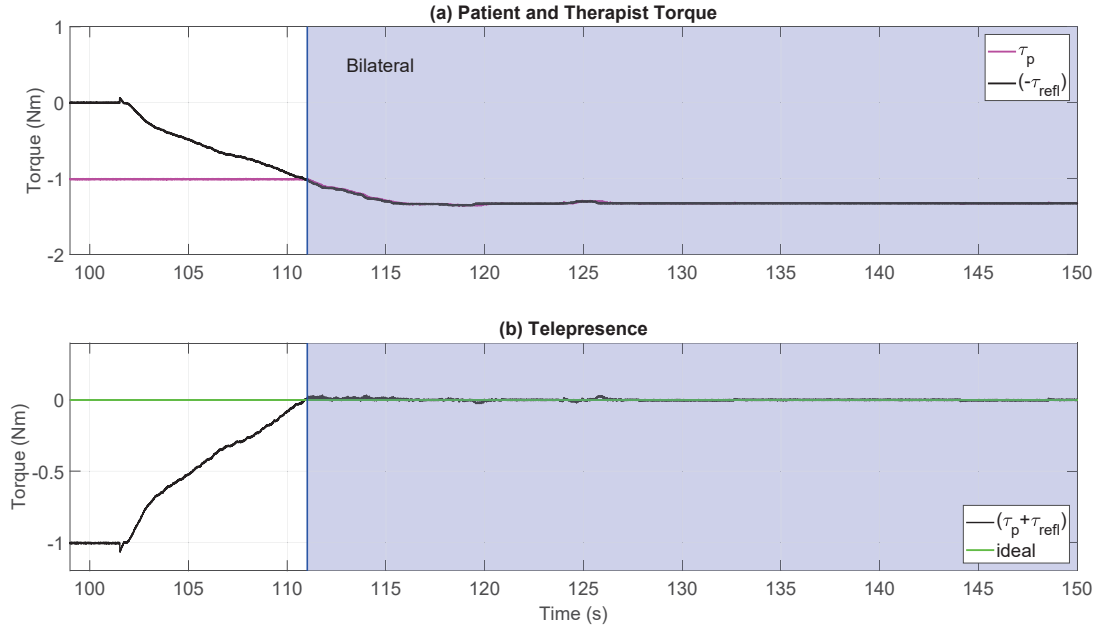


Figure A.6: Torques for case 3 ARII telerehabilitation experiment with Quanser QUBEs with HTR during the bilateral phase

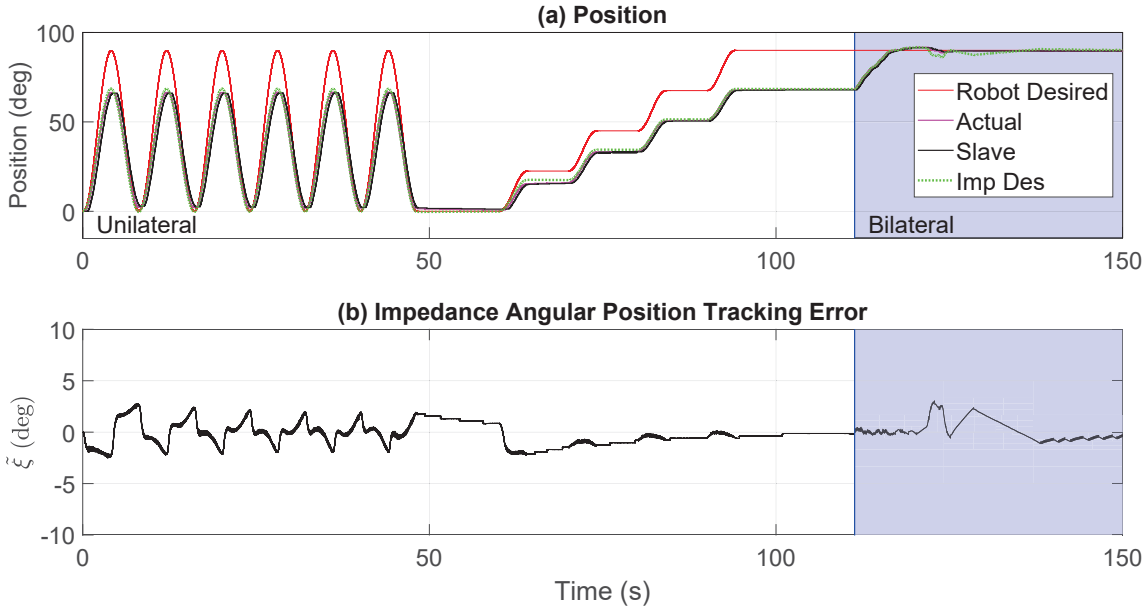


Figure A.7: Case 4 ARII telerehabilitation experiment with Quanser QUBEs using HTR: (a) position trajectory, (b) tracking error

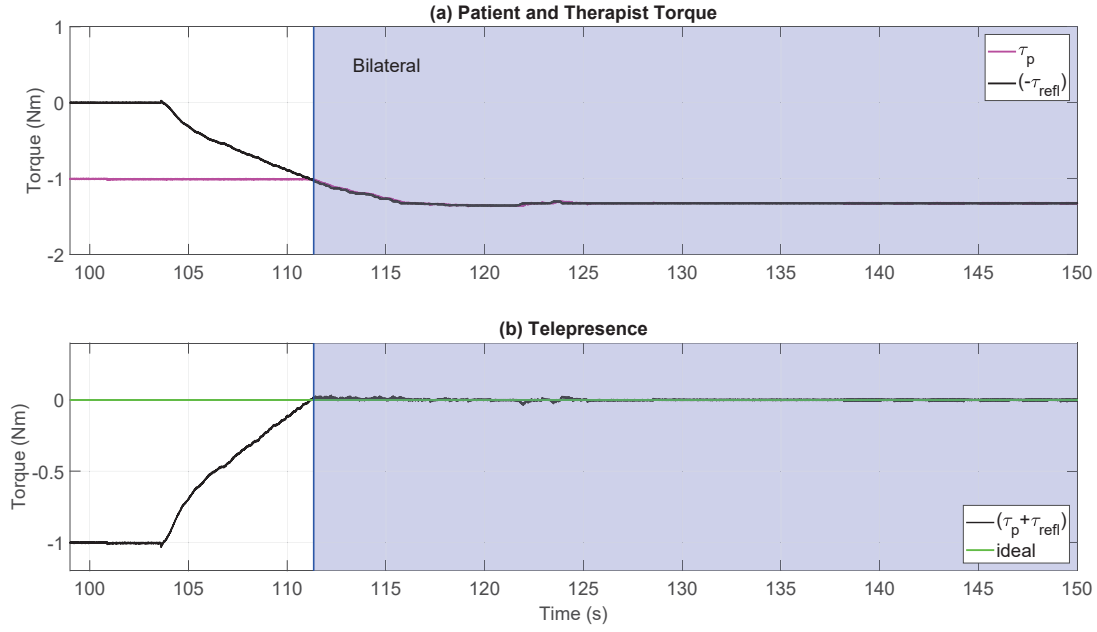


Figure A.8: Torques for case 4 ARII telerehabilitation experiment with Quanser QUBEs with HTR during the bilateral phase

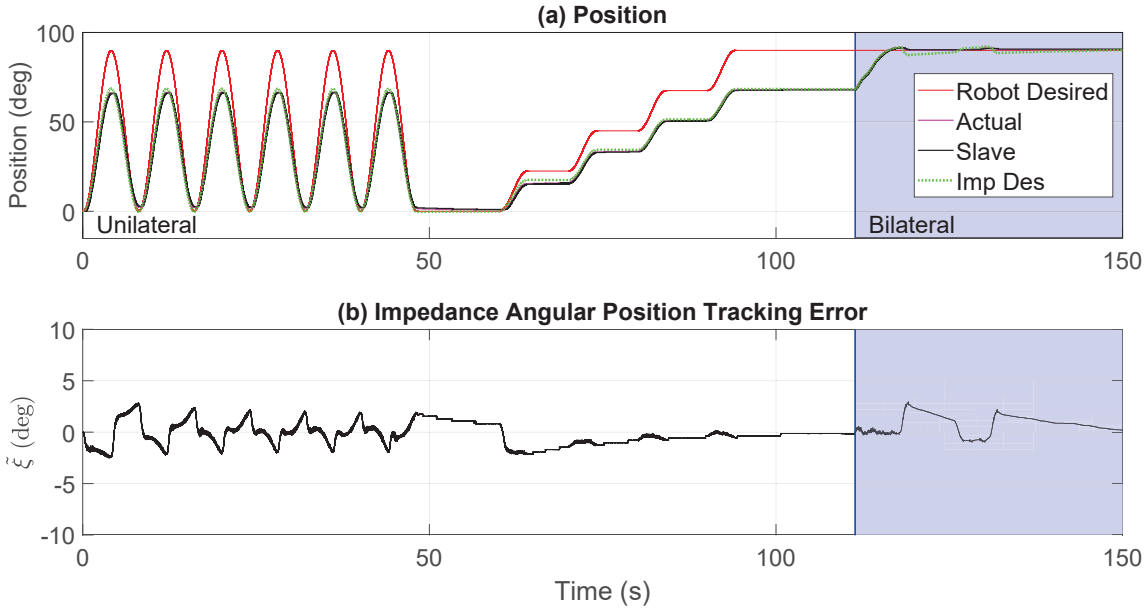


Figure A.9: Case 5 ARII telerehabilitation experiment with Quanser QUBEs using HTR: (a) position trajectory, (b) tracking error

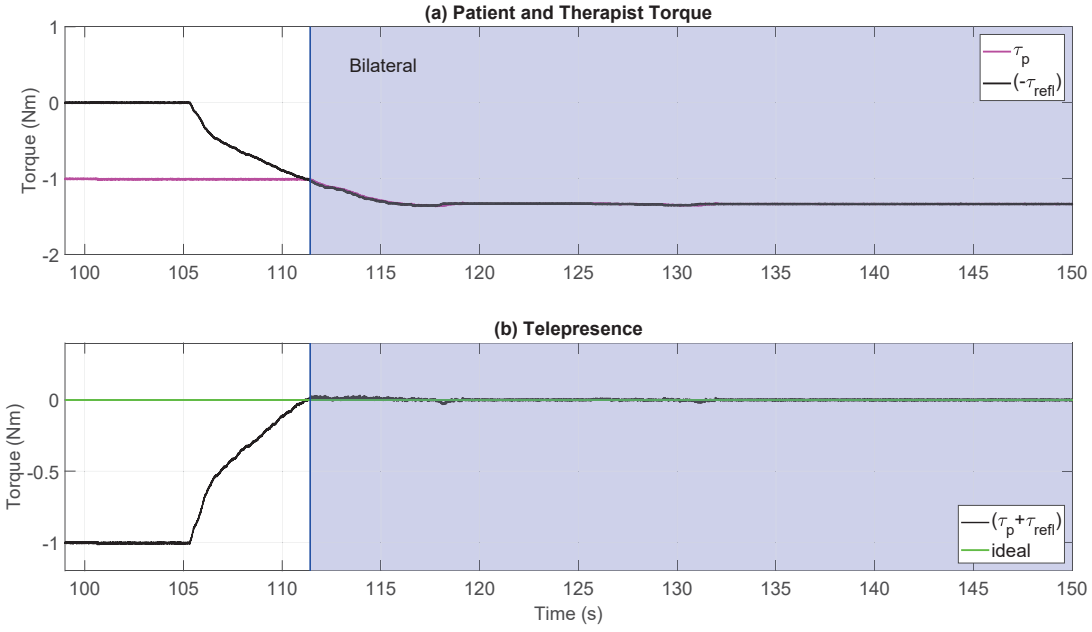


Figure A.10: Torques for case 5 ARII telerehabilitation experiment with Quanser QUBEs with HTR during the bilateral phase

A.1.2 Quanser QUBE Experimental Tests Case Studies with RBFNN-I Control Methods

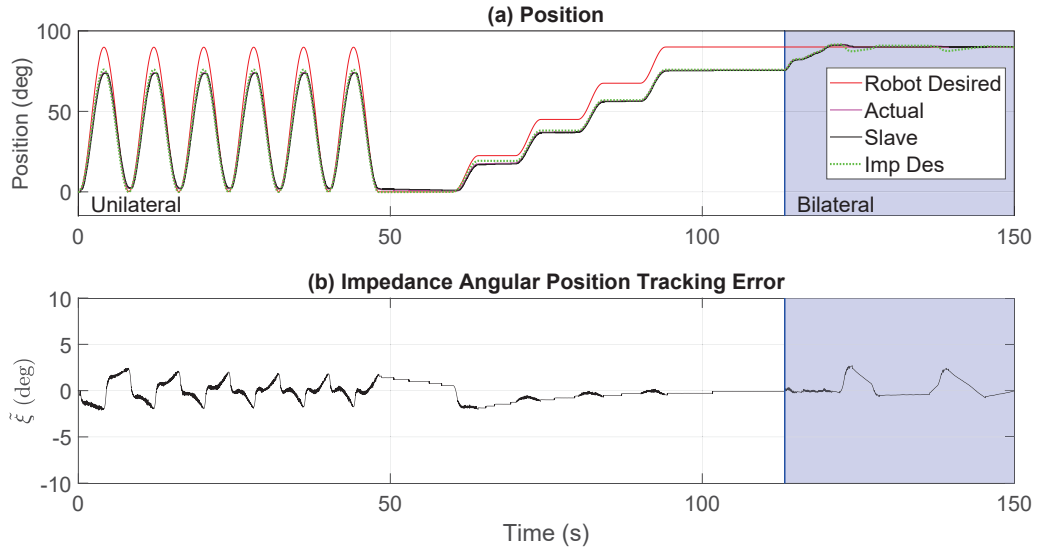


Figure A.11: Case 1 RBFNN-I telerehabilitation experiment with Quanser QUBEs using HTR: (a) position trajectory, (b) tracking error

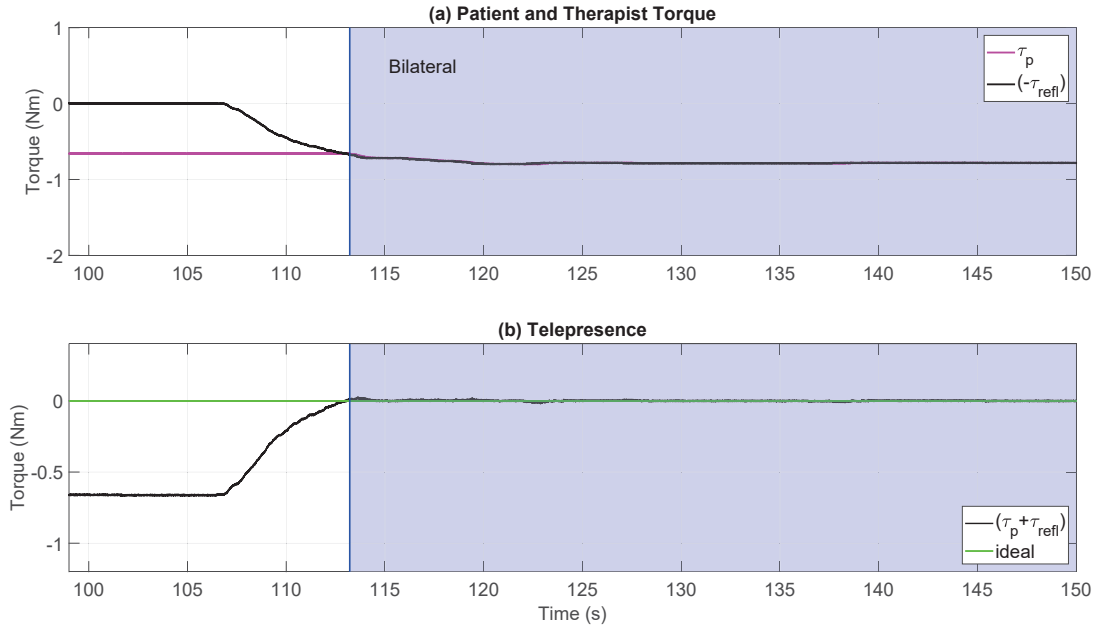


Figure A.12: Torques for case 1 RBFNN-I telerehabilitation experiment with Quanser QUBEs with HTR during the bilateral phase

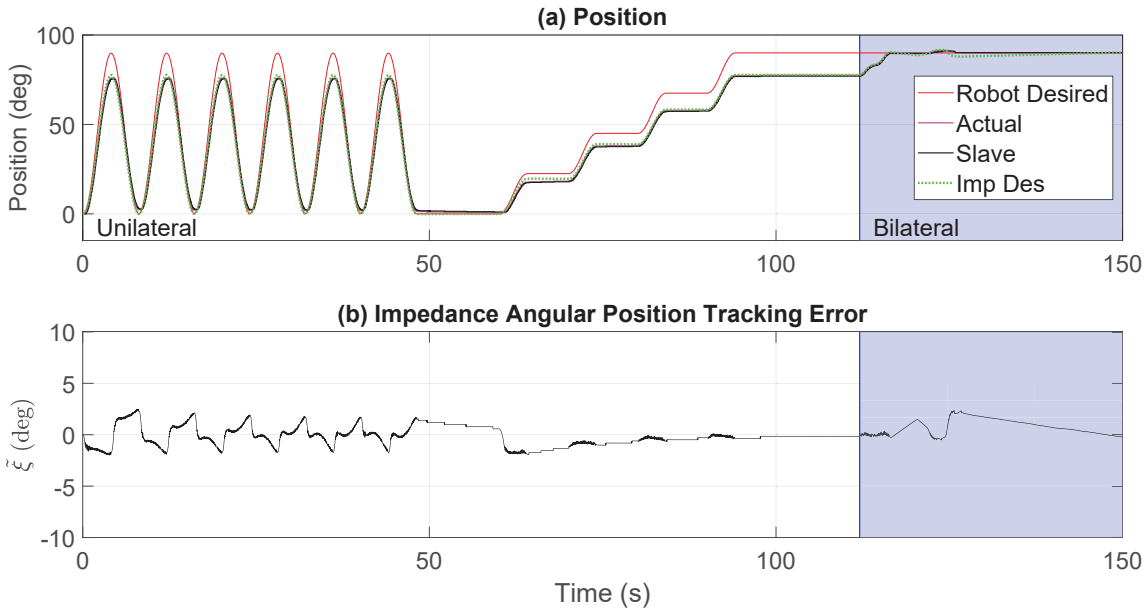


Figure A.13: Case 2 RBFNN-I telerehabilitation experiment with Quanser QUBEs using HTR: (a) position trajectory, (b) tracking error

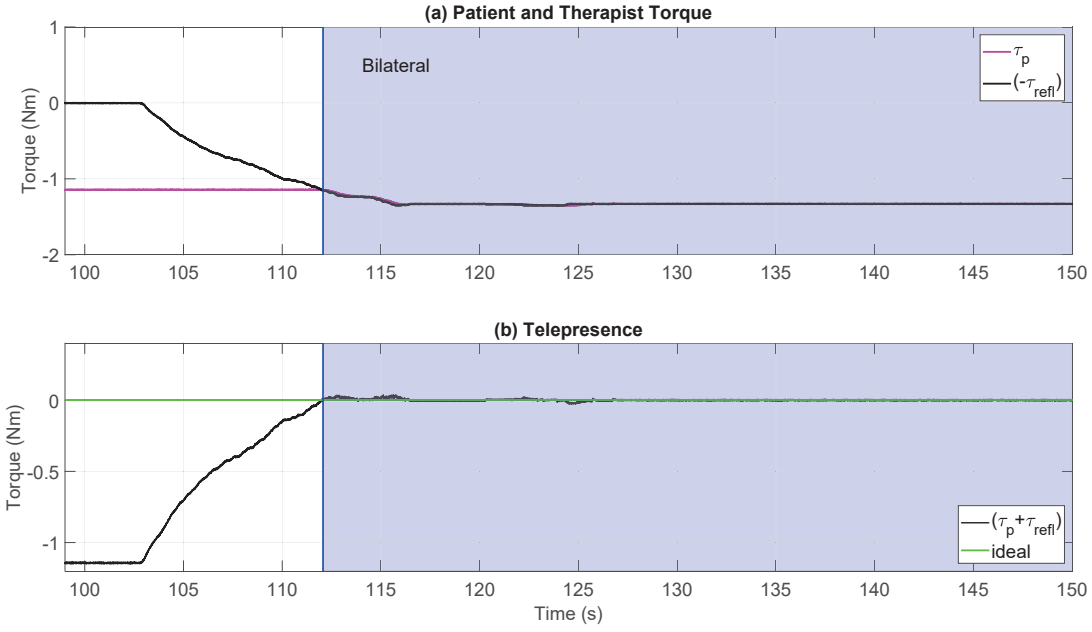


Figure A.14: Torques for case 2 RBFNN-I telerehabilitation experiment with Quanser QUBEs with HTR during the bilateral phase

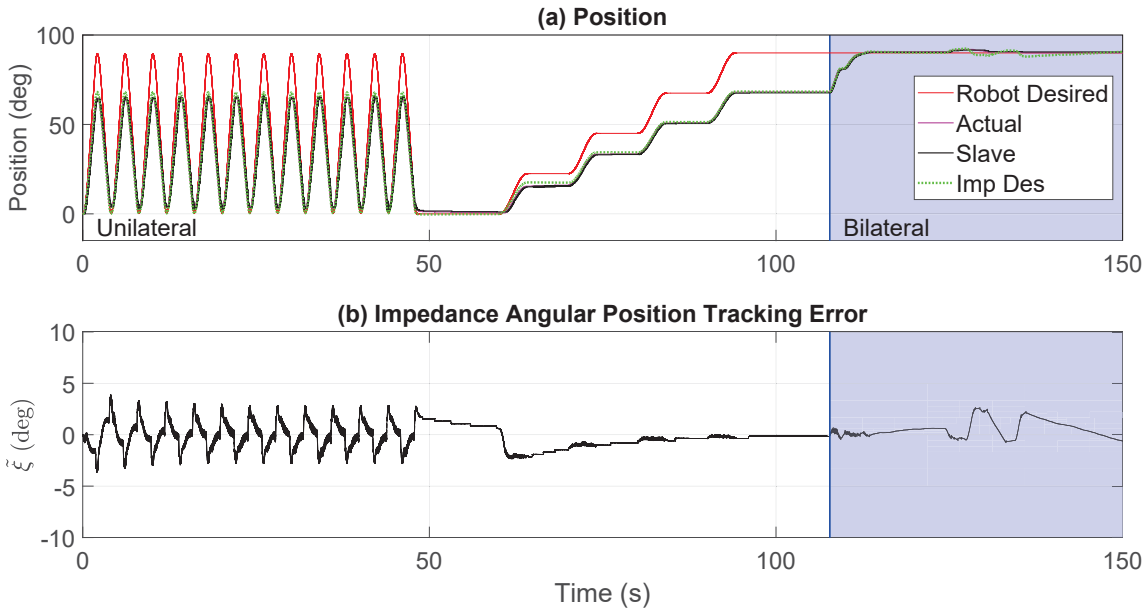


Figure A.15: Case 3 RBFNN-I telerehabilitation experiment with Quanser QUBEs using HTR: (a) position trajectory, (b) tracking error

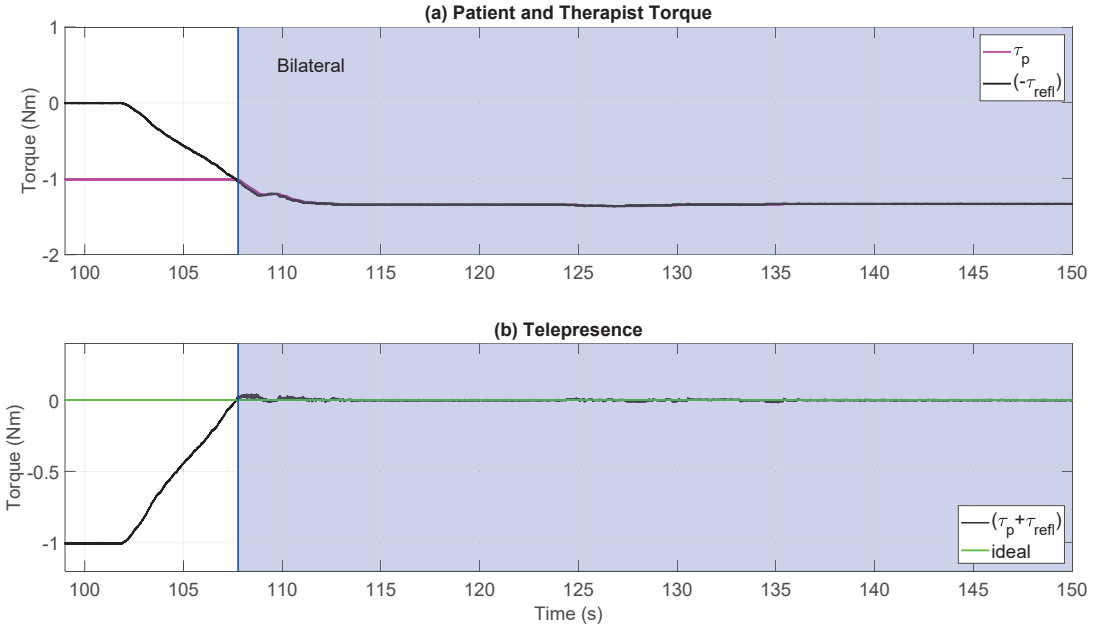


Figure A.16: Torques for case 3 RBFNN-I telerehabilitation experiment with Quanser QUBEs with HTR during the bilateral phase

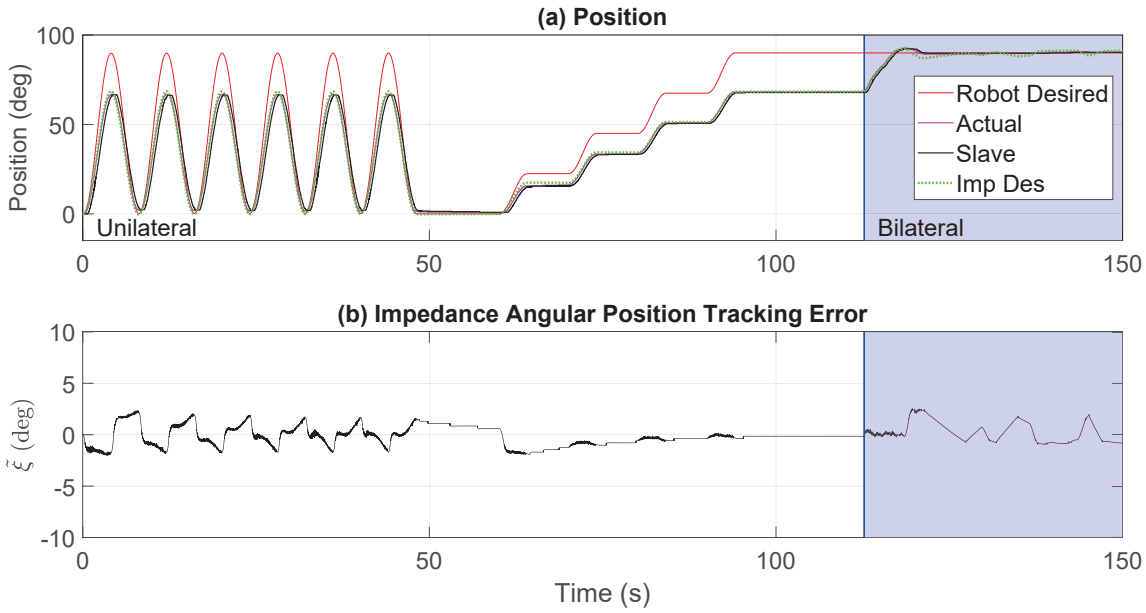


Figure A.17: Case 4 RBFNN-I telerehabilitation experiment with Quanser QUBEs using HTR: (a) position trajectory, (b) tracking error

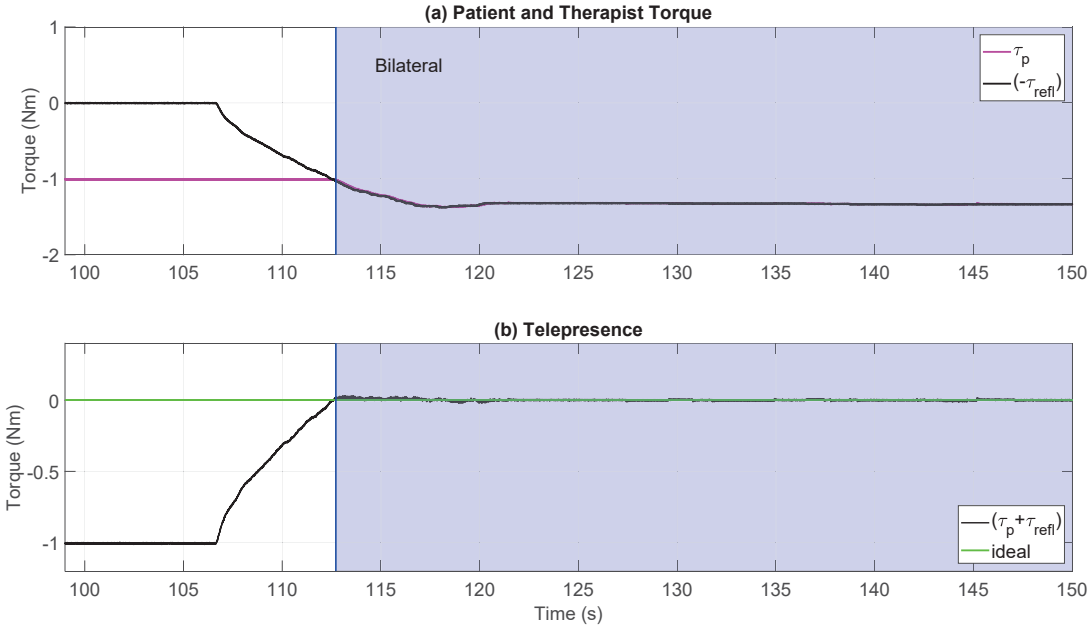


Figure A.18: Torques for case 4 RBFNN-I telerehabilitation experiment with Quanser QUBEs with HTR during the bilateral phase

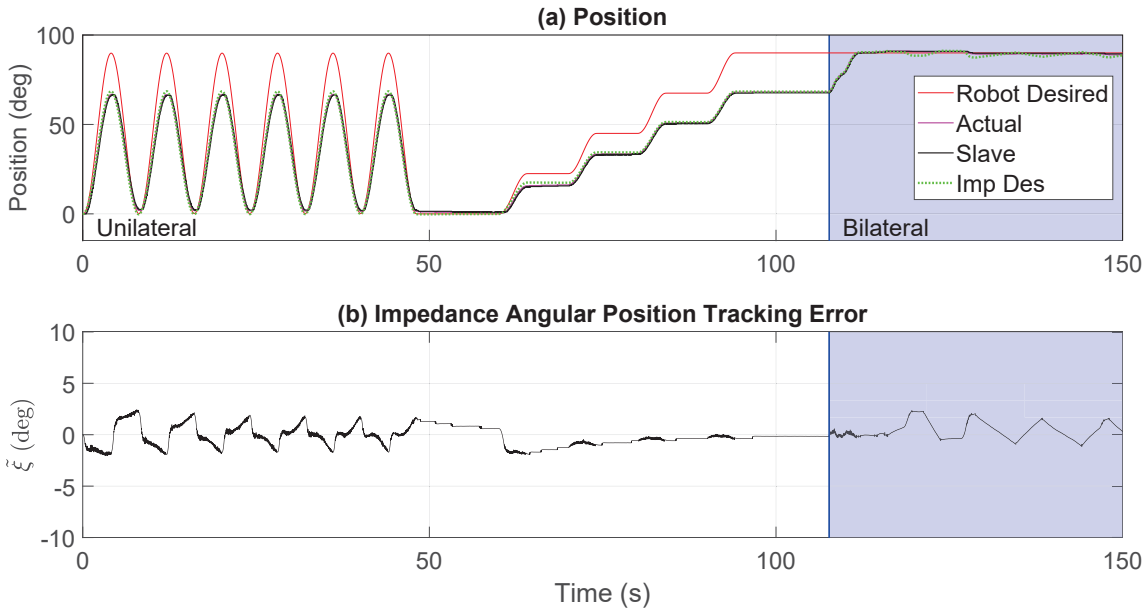


Figure A.19: Case 5 RBFNN-I telerehabilitation experiment with Quanser QUBEs using HTR: (a) position trajectory, (b) tracking error

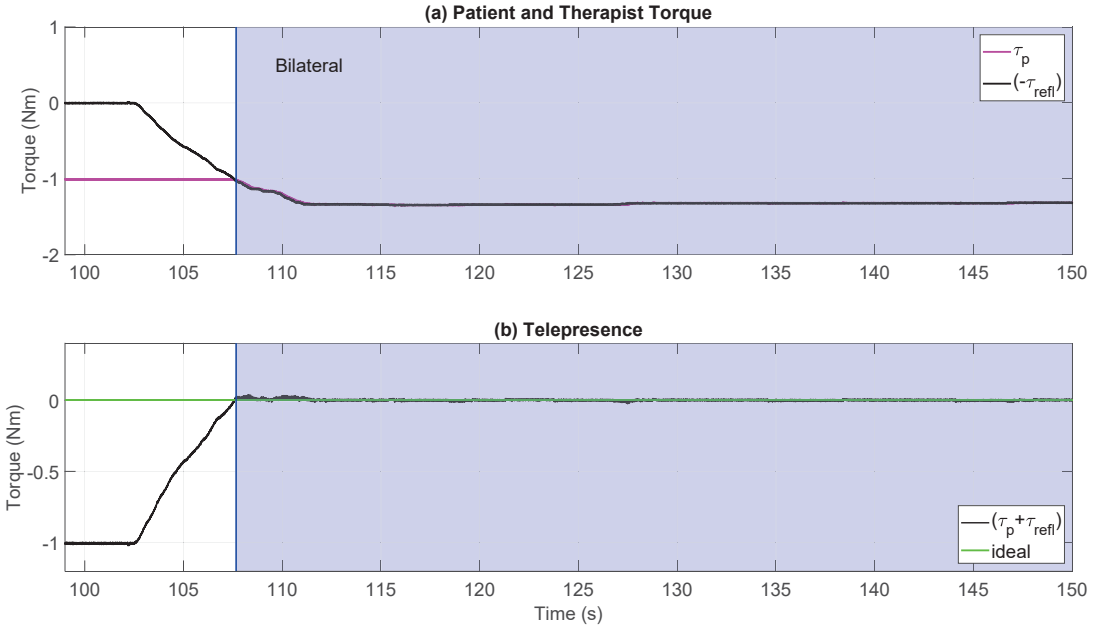


Figure A.20: Torques for case 5 RBFNN-I telerehabilitation experiment with Quanser QUBEs with HTR during the bilateral phase

A.2 2-DOF Exoskeleton Simulation Case Studies

A.2.1 2-DOF Exoskeleton Simulation Case Studies with ARII Control Methods

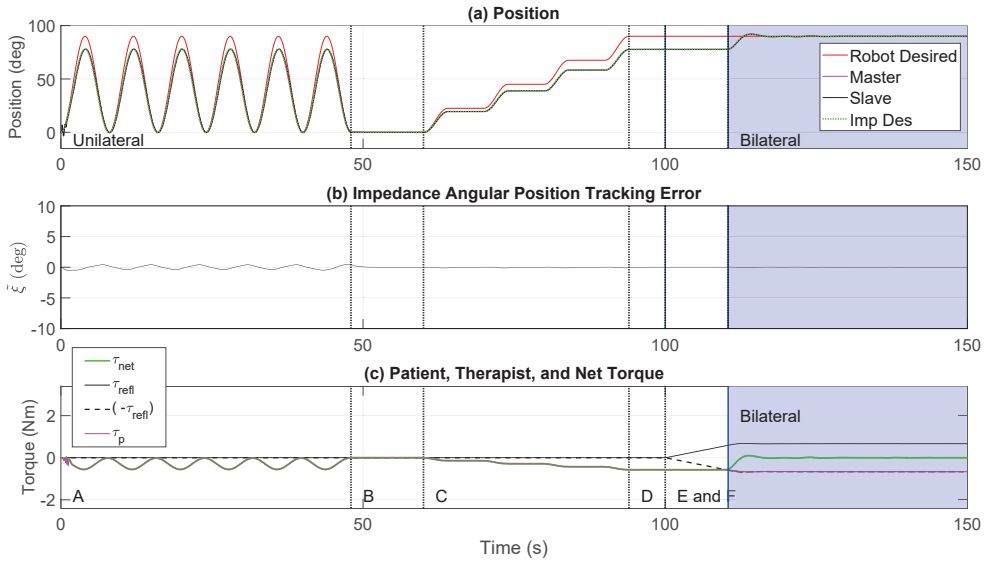


Figure A.21: Joint 1, Case 1 ARII telerehabilitation experiment with 2-DOF Exoskeletons using HTR: (a) position trajectory, (b) tracking error

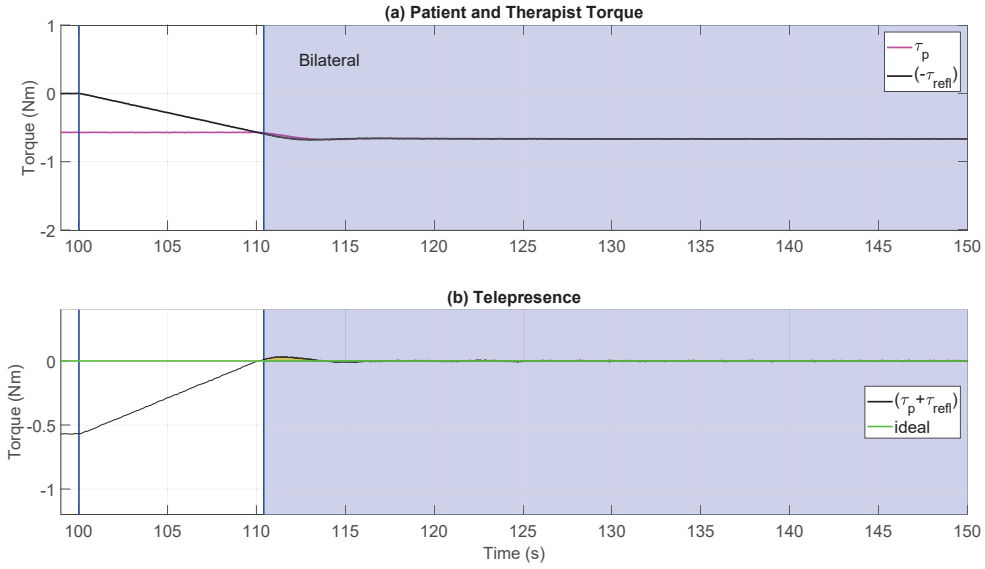


Figure A.22: Joint 1 torques for case 1 ARII telerehabilitation 2-DOF simulation with HTR during the bilateral phase

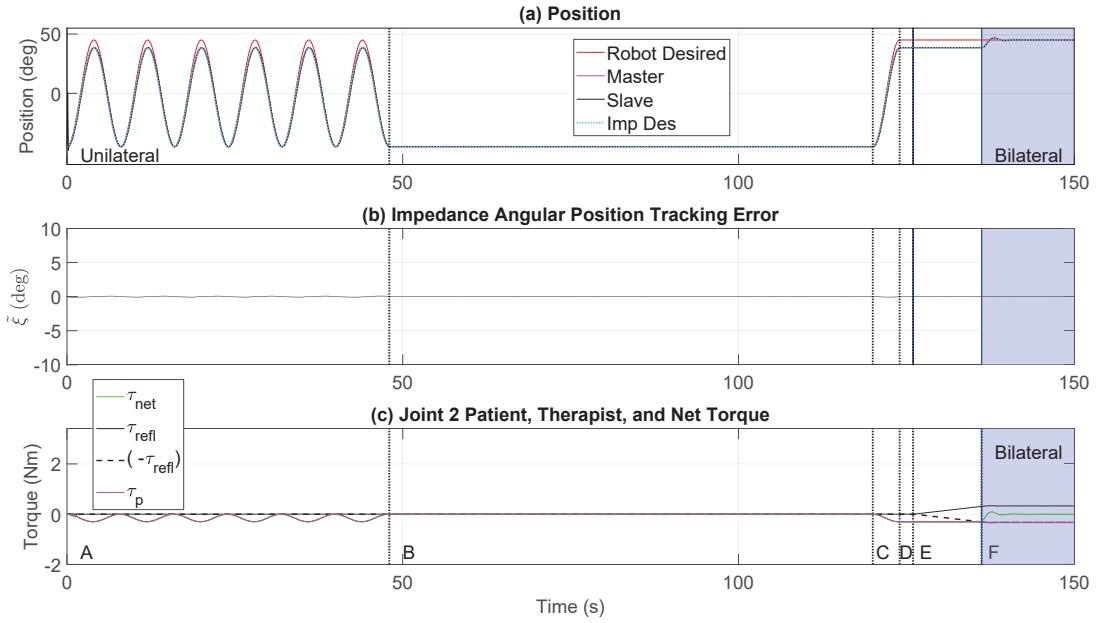


Figure A.23: Joint 2, Case 1 ARII telerehabilitation experiment with 2-DOF Exoskeletons using HTR: (a) position trajectory, (b) tracking error

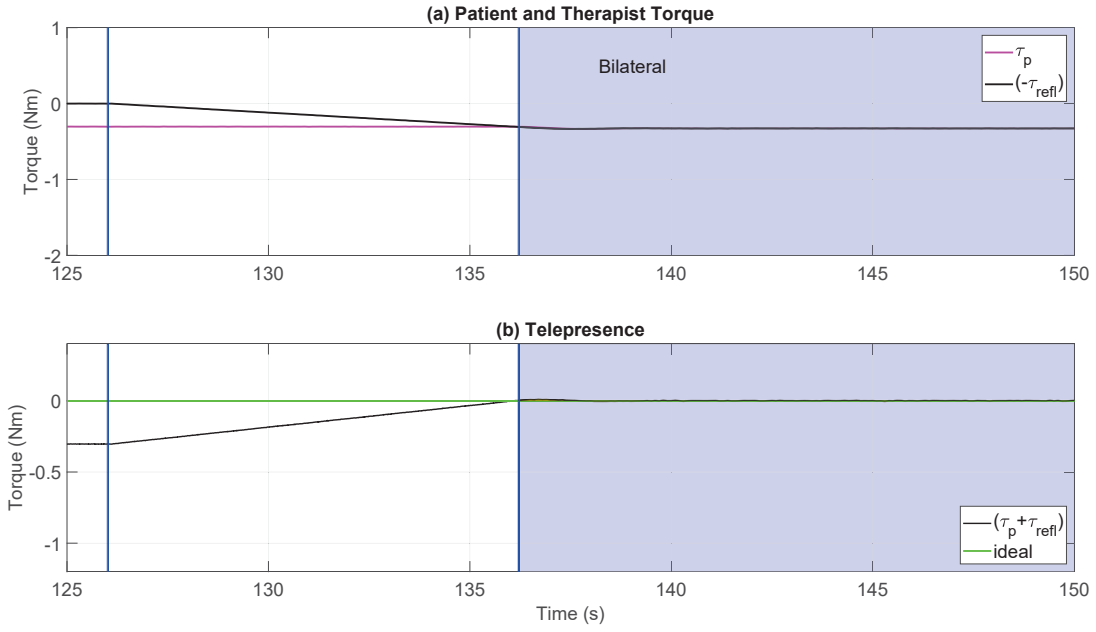


Figure A.24: Joint 2 torques for case 1 ARII telerehabilitation 2-DOF simulation with HTR during the bilateral phase

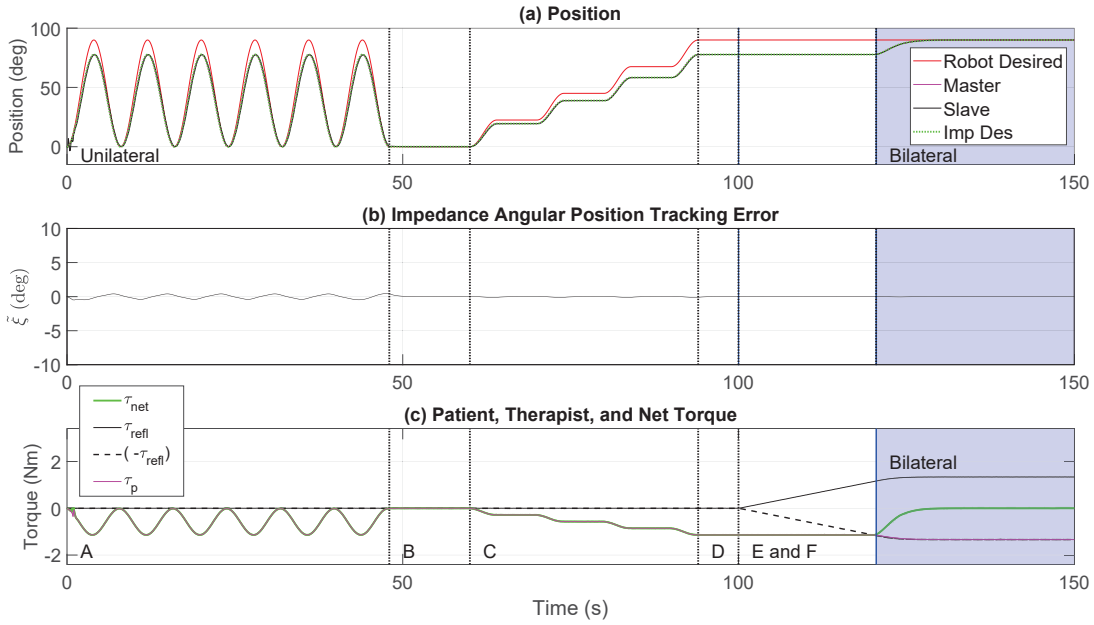


Figure A.25: Joint 1, Case 2 ARII telerehabilitation experiment with 2-DOF Exoskeletons using HTR: (a) position trajectory, (b) tracking error

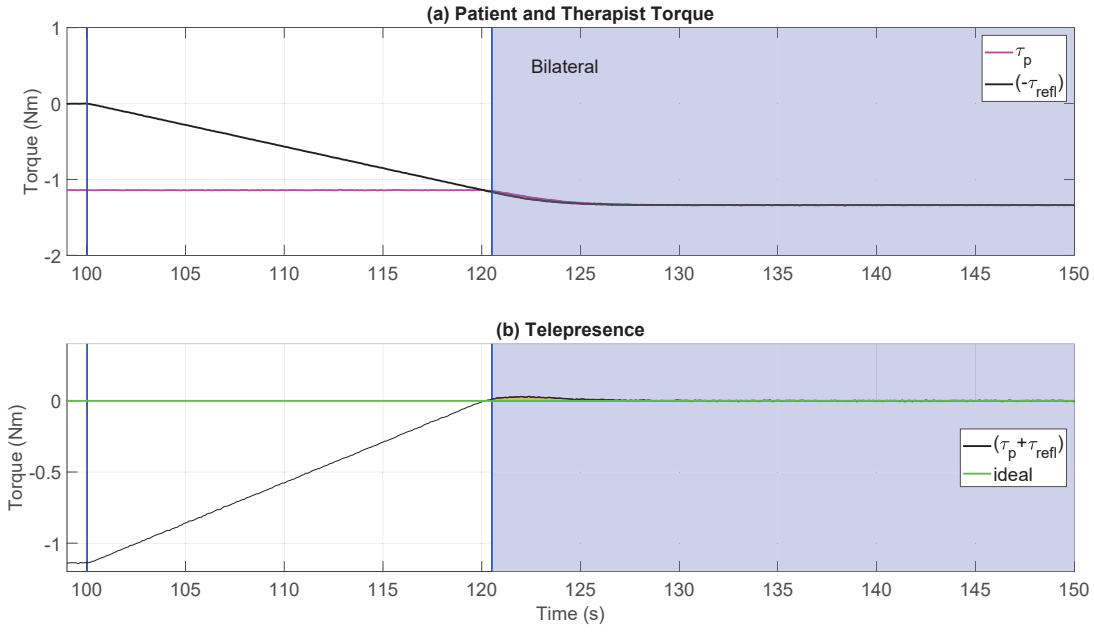


Figure A.26: Joint 1 torques for case 2 ARII telerehabilitation 2-DOF simulation with HTR during the bilateral phase

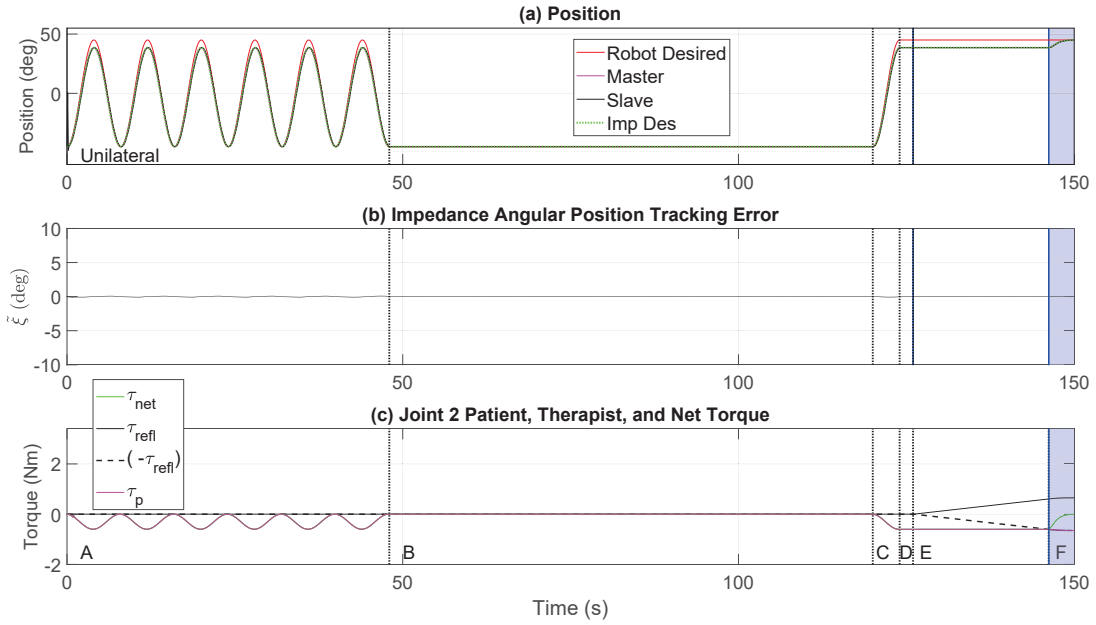


Figure A.27: Joint 2, Case 2 ARII telerehabilitation experiment with 2-DOF Exoskeletons using HTR: (a) position trajectory, (b) tracking error

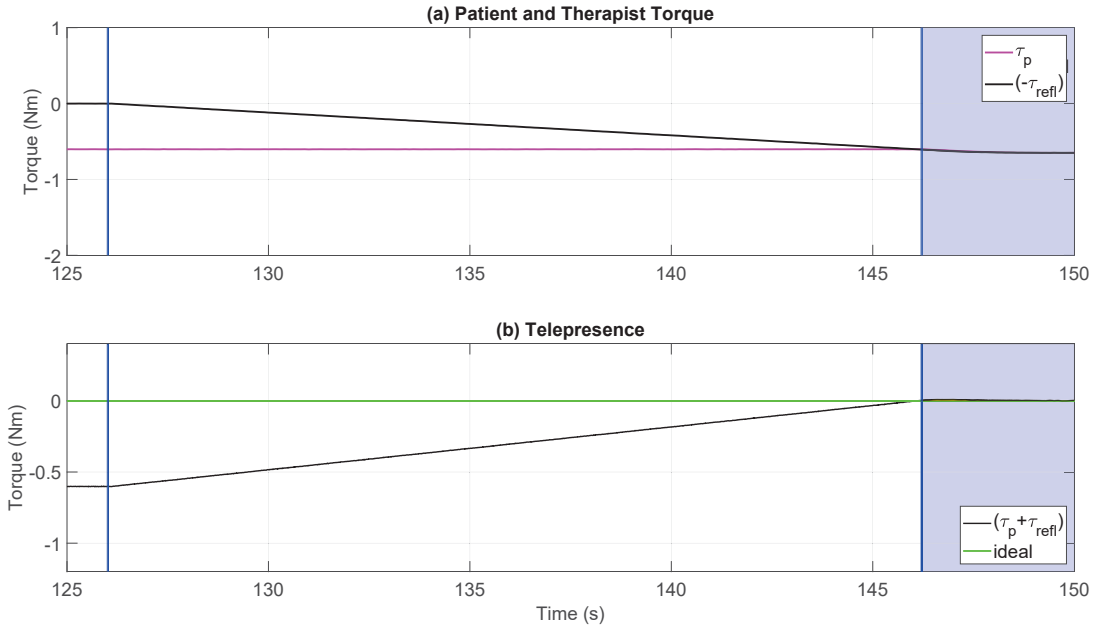


Figure A.28: Joint 2 torques for case 2 ARII telerehabilitation 2-DOF simulation with HTR during the bilateral phase

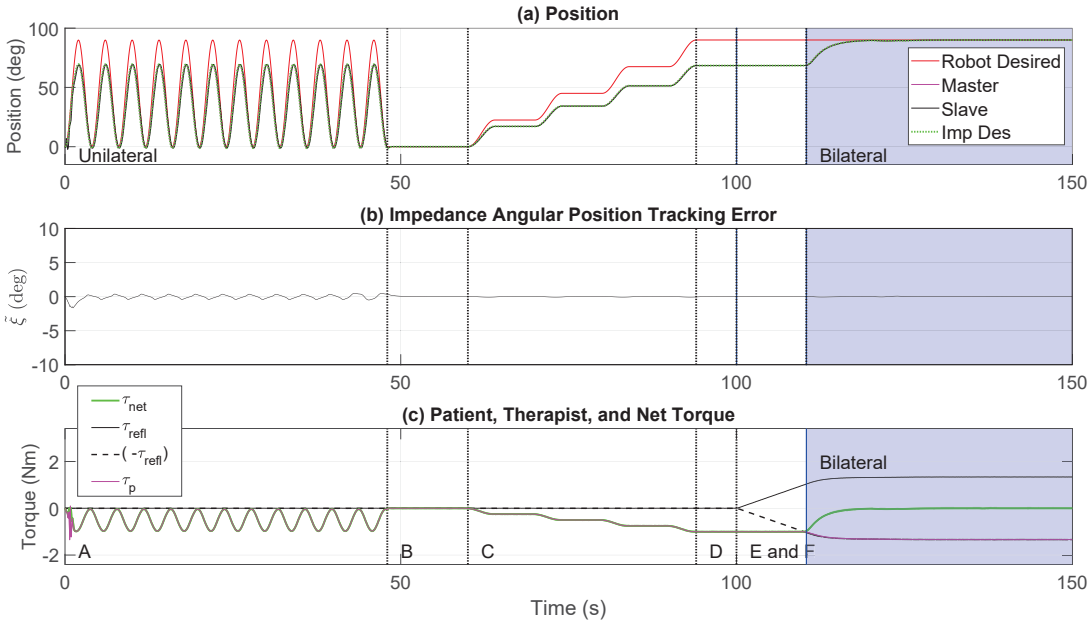


Figure A.29: Joint 1, Case 3 ARII telerehabilitation experiment with 2-DOF Exoskeletons using HTR: (a) position trajectory, (b) tracking error

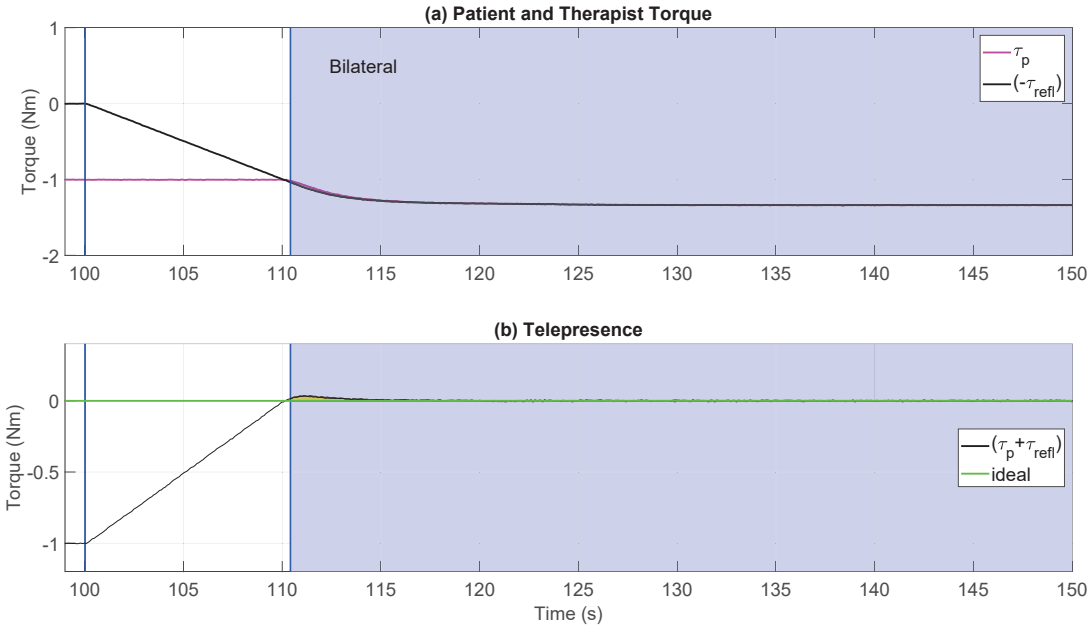


Figure A.30: Joint 1 torques for case 3 ARII telerehabilitation 2-DOF simulation with HTR during the bilateral phase

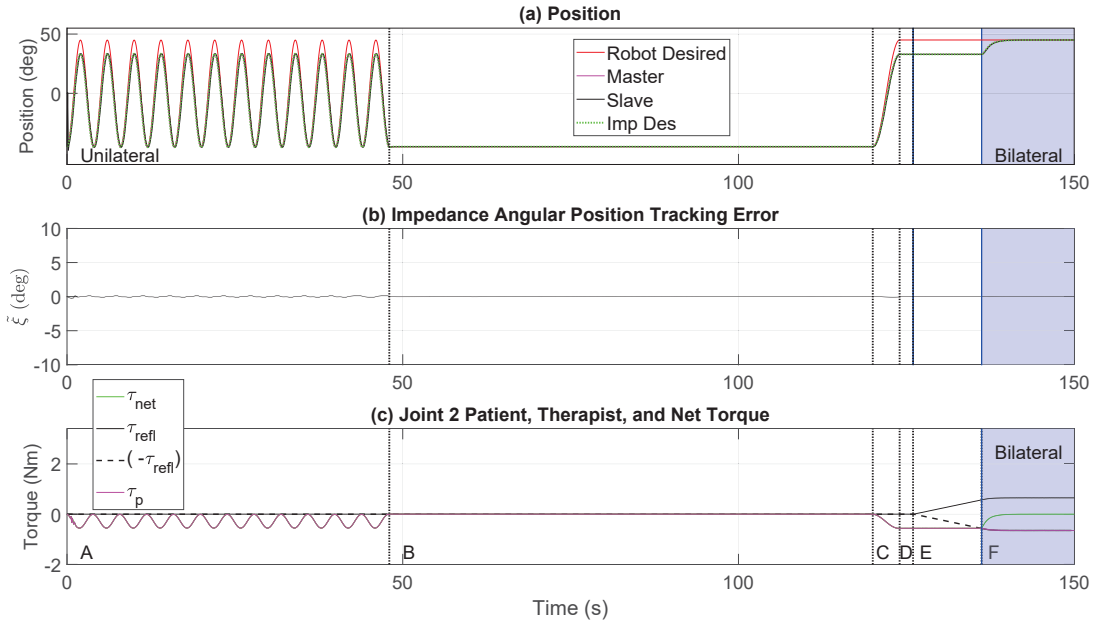


Figure A.31: Joint 2, Case 3 ARII telerehabilitation experiment with 2-DOF Exoskeletons using HTR: (a) position trajectory, (b) tracking error

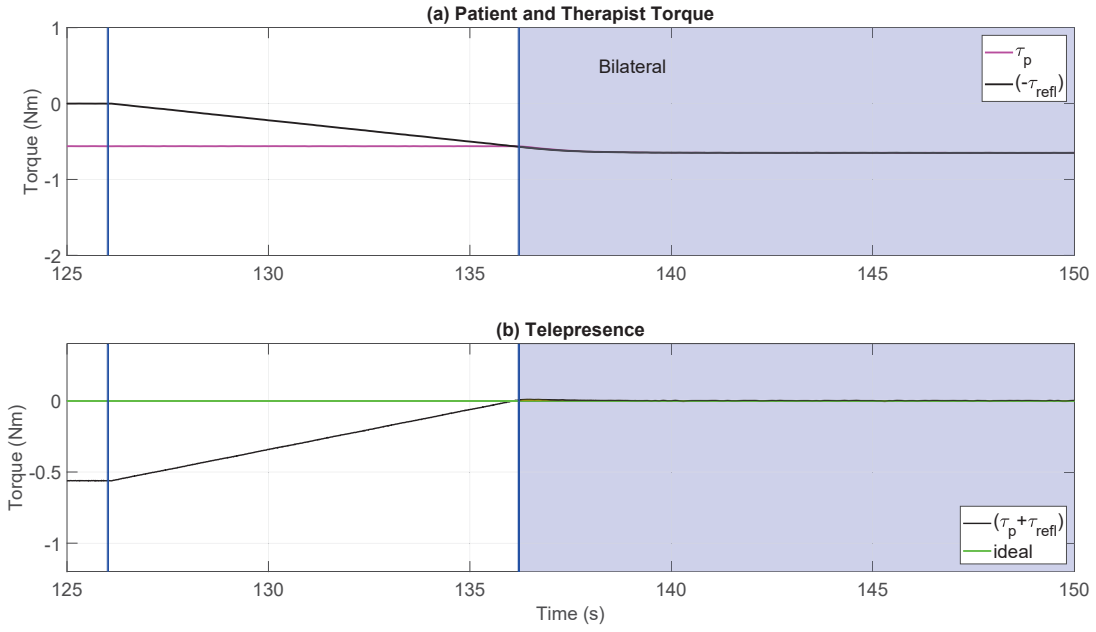


Figure A.32: Joint 2 torques for case 3 ARII telerehabilitation 2-DOF simulation with HTR during the bilateral phase

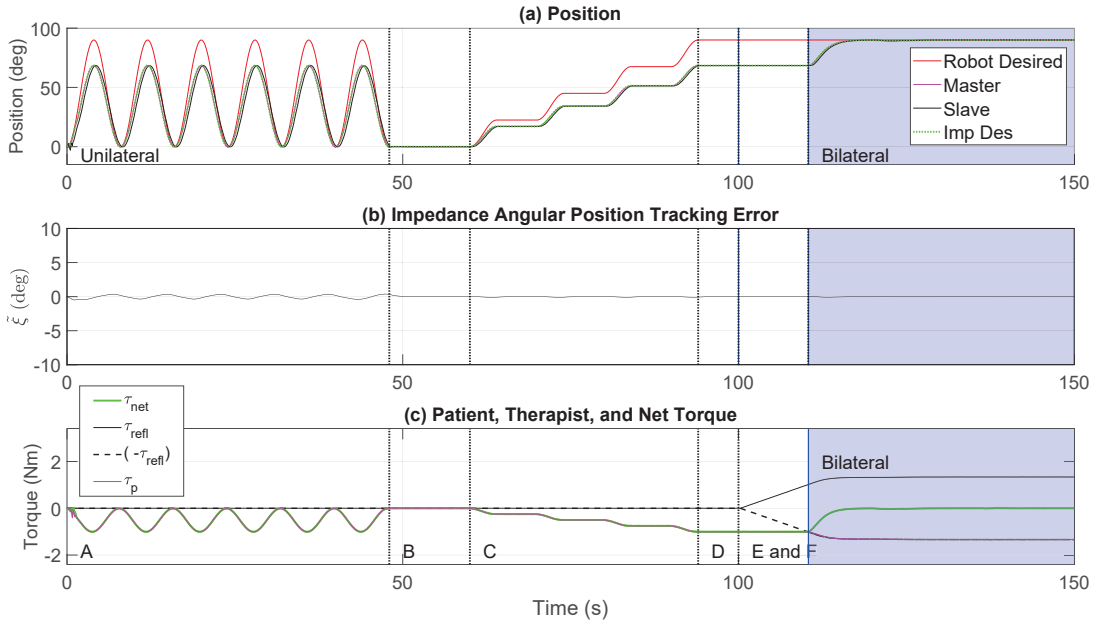


Figure A.33: Joint 1, Case 4 ARII telerehabilitation experiment with 2-DOF Exoskeletons using HTR: (a) position trajectory, (b) tracking error

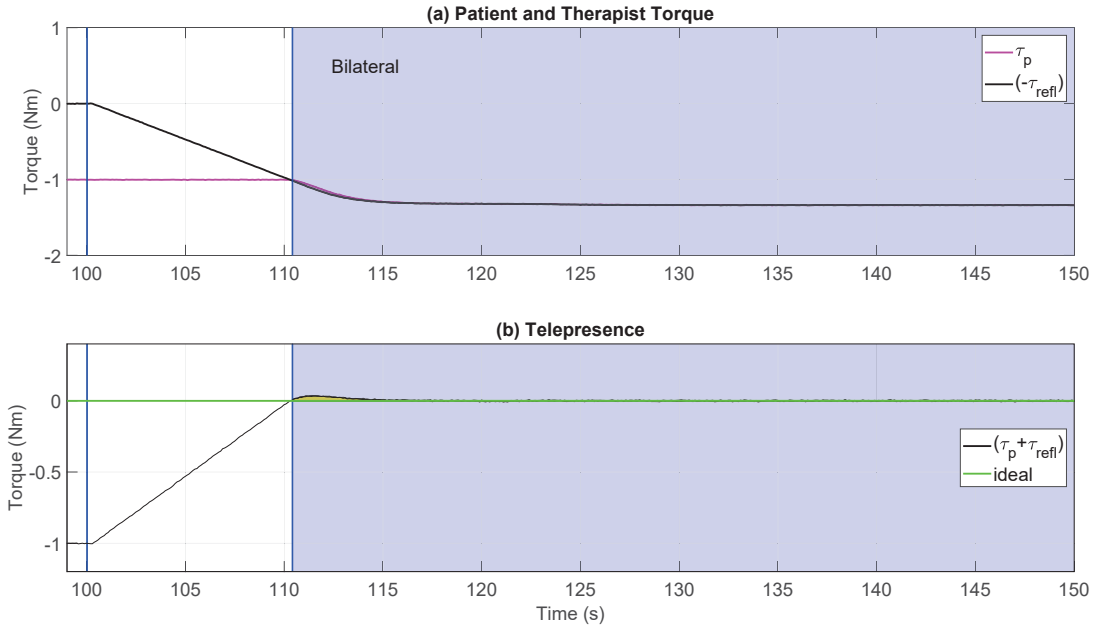


Figure A.34: Joint 1 torques for case 4 ARII telerehabilitation 2-DOF simulation with HTR during the bilateral phase

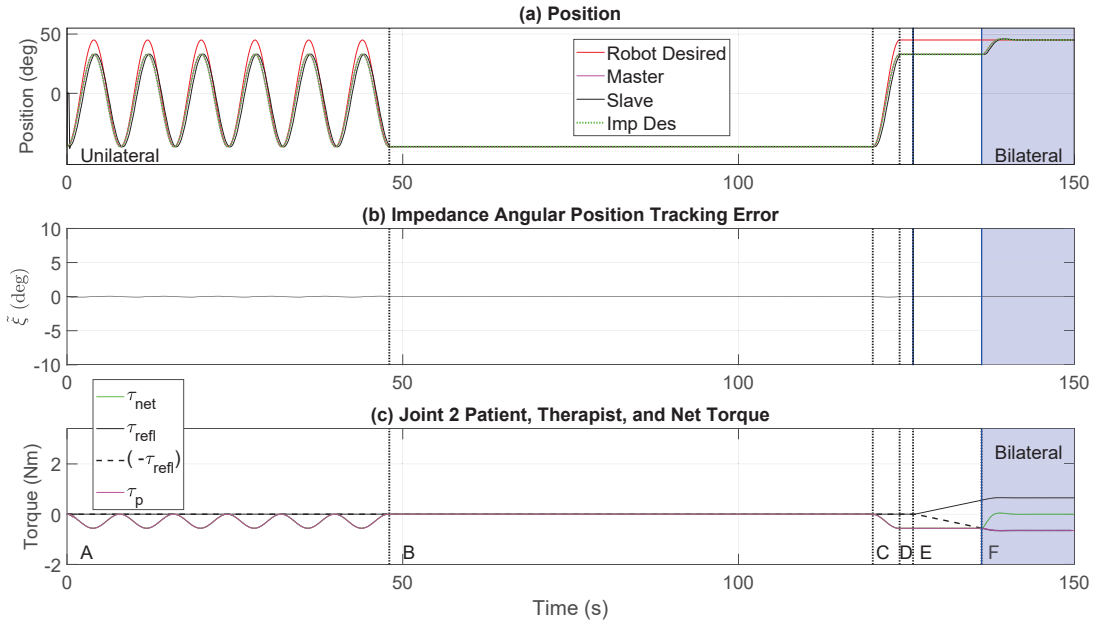


Figure A.35: Joint 2, Case 4 ARII telerehabilitation experiment with 2-DOF Exoskeletons using HTR: (a) position trajectory, (b) tracking error

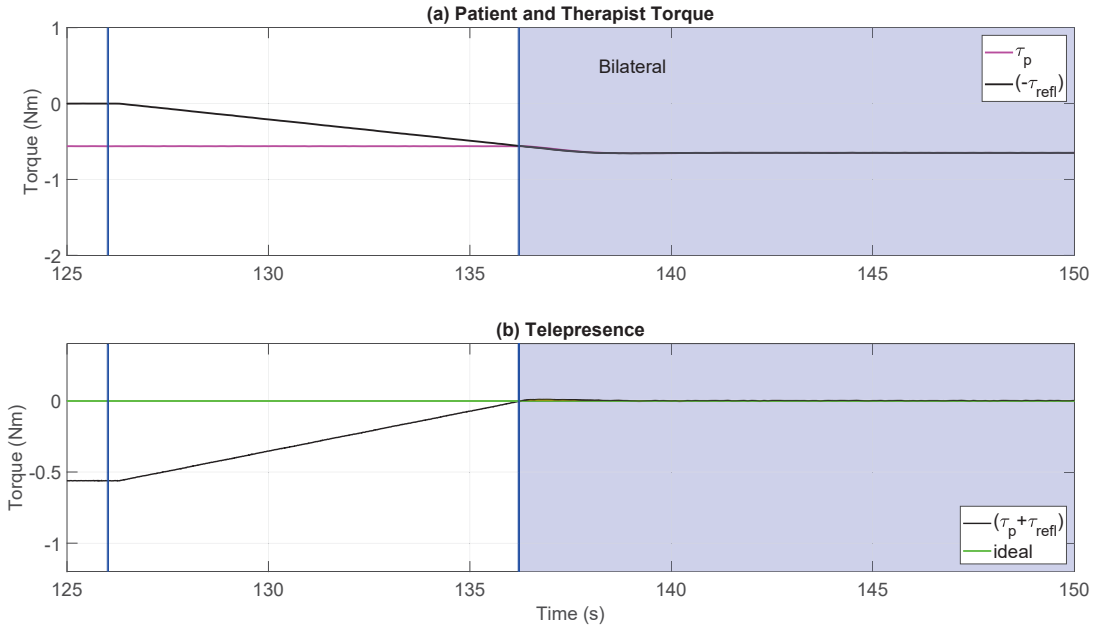


Figure A.36: Joint 2 torques for case 4 ARII telerehabilitation 2-DOF simulation with HTR during the bilateral phase

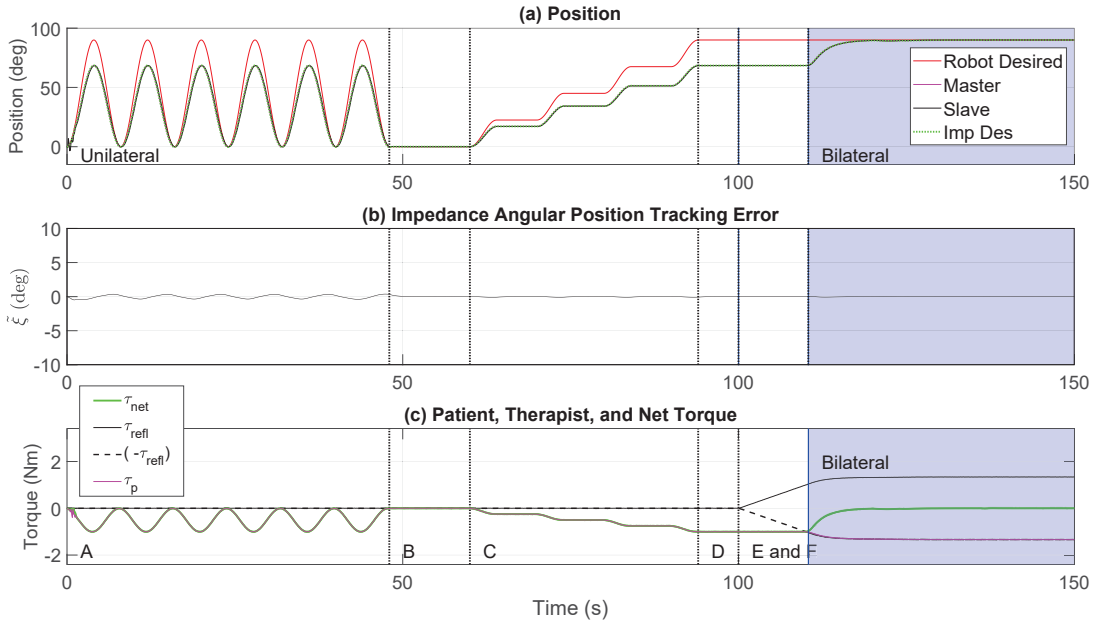


Figure A.37: Joint 1, Case 5 ARII telerehabilitation experiment with 2-DOF Exoskeletons using HTR: (a) position trajectory, (b) tracking error

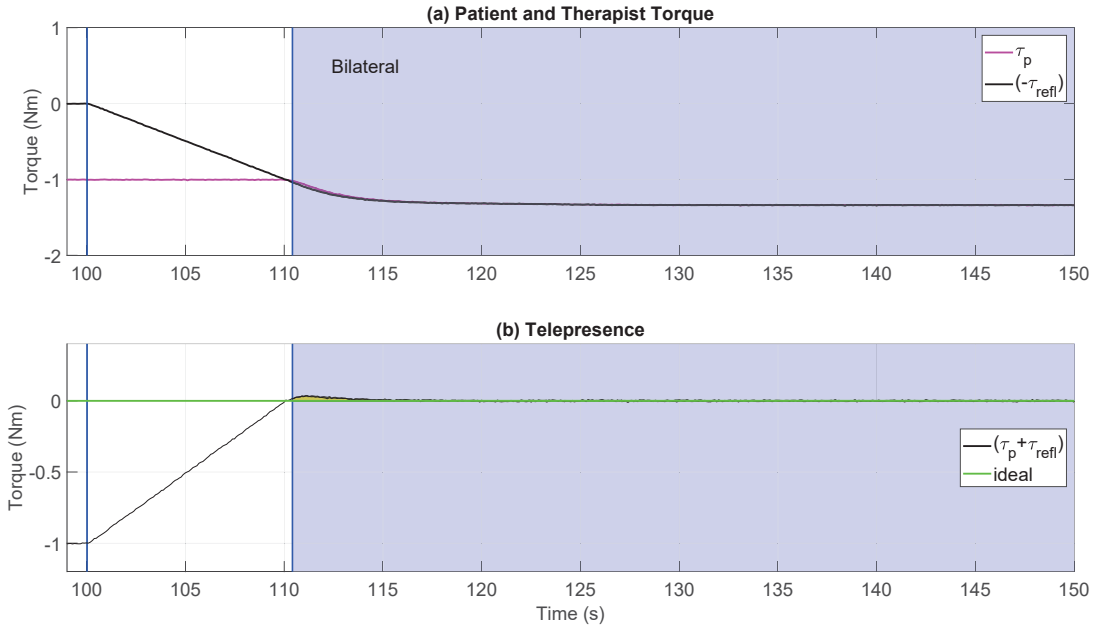


Figure A.38: Joint 1 torques for case 5 ARII telerehabilitation 2-DOF simulation with HTR during the bilateral phase

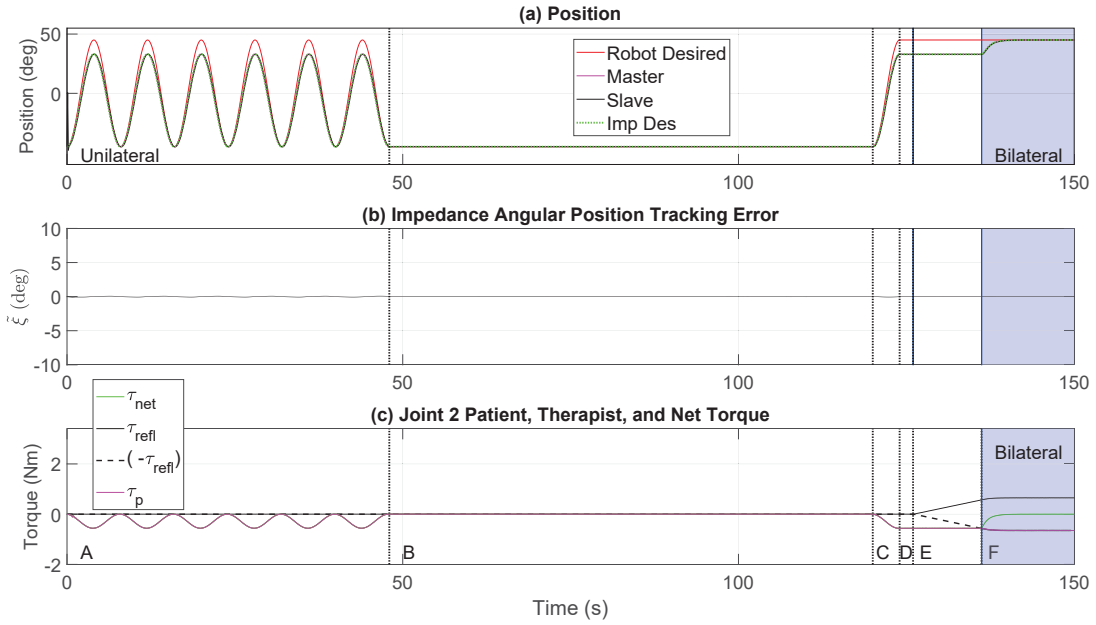


Figure A.39: Joint 2, Case 5 ARII telerehabilitation experiment with 2-DOF Exoskeletons using HTR: (a) position trajectory, (b) tracking error

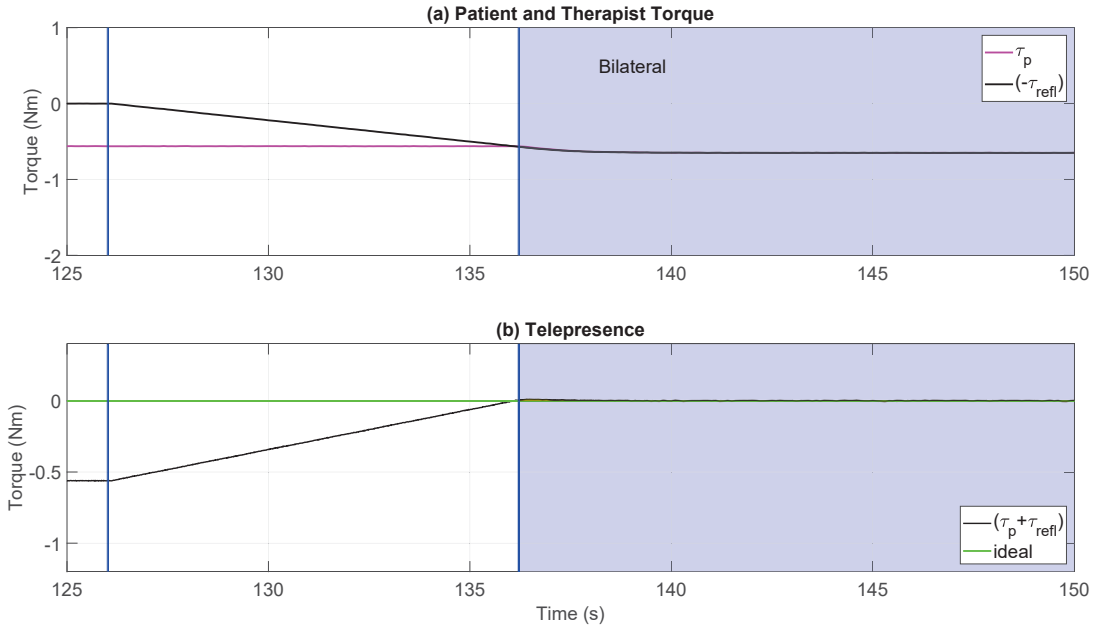


Figure A.40: Joint 2 torques for case 5 ARII telerehabilitation 2-DOF simulation with HTR during the bilateral phase

A.2.2 2-DOF Exoskeleton Simulation Case Studies with RBFNN-I Control Methods

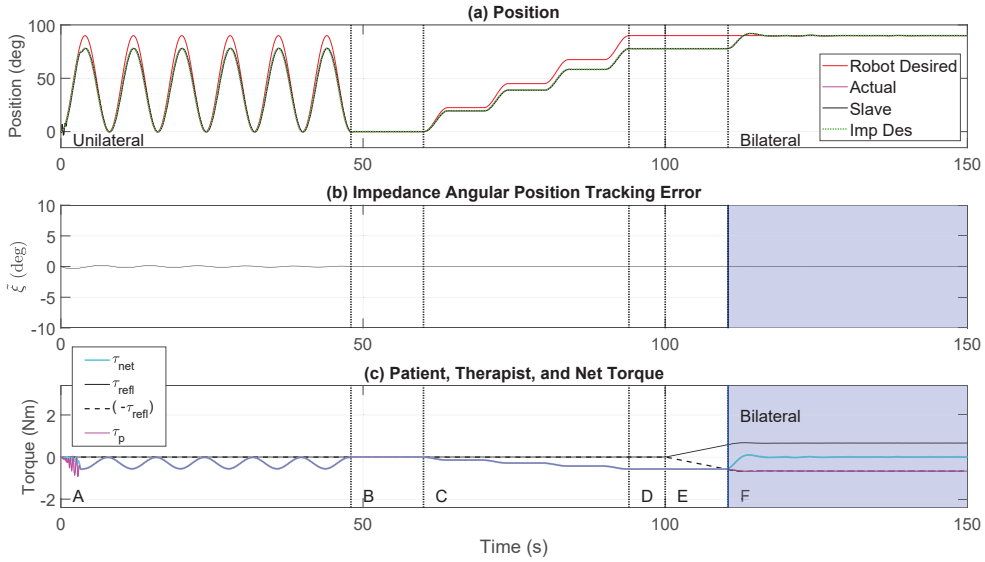


Figure A.41: Joint 1, Case 1 RBFNN-I telerehabilitation experiment with 2-DOF Exoskeletons using HTR: (a) position trajectory, (b) tracking error

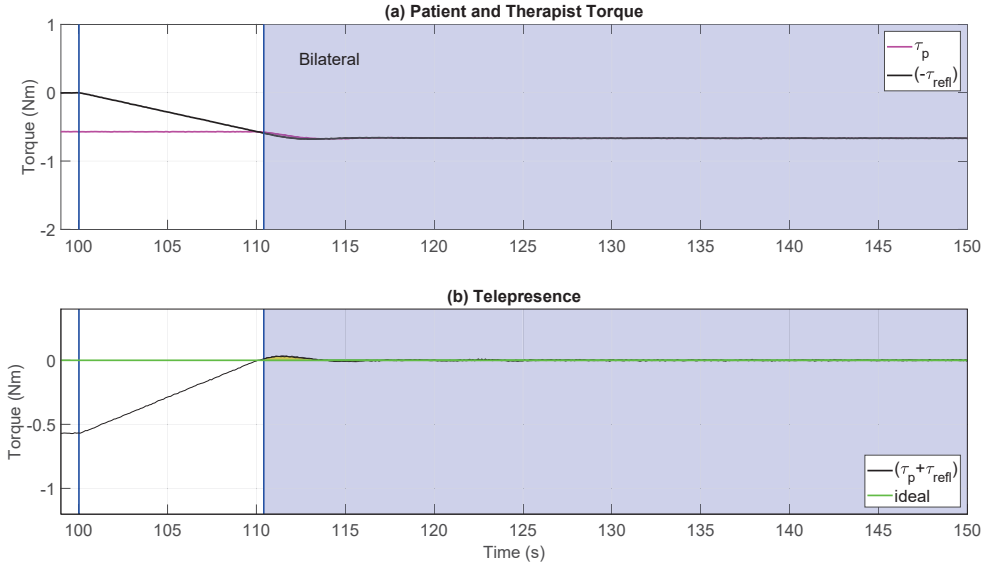


Figure A.42: Joint 1 torques for case 1 RBFNN-I telerehabilitation 2-DOF simulation with HTR during the bilateral phase

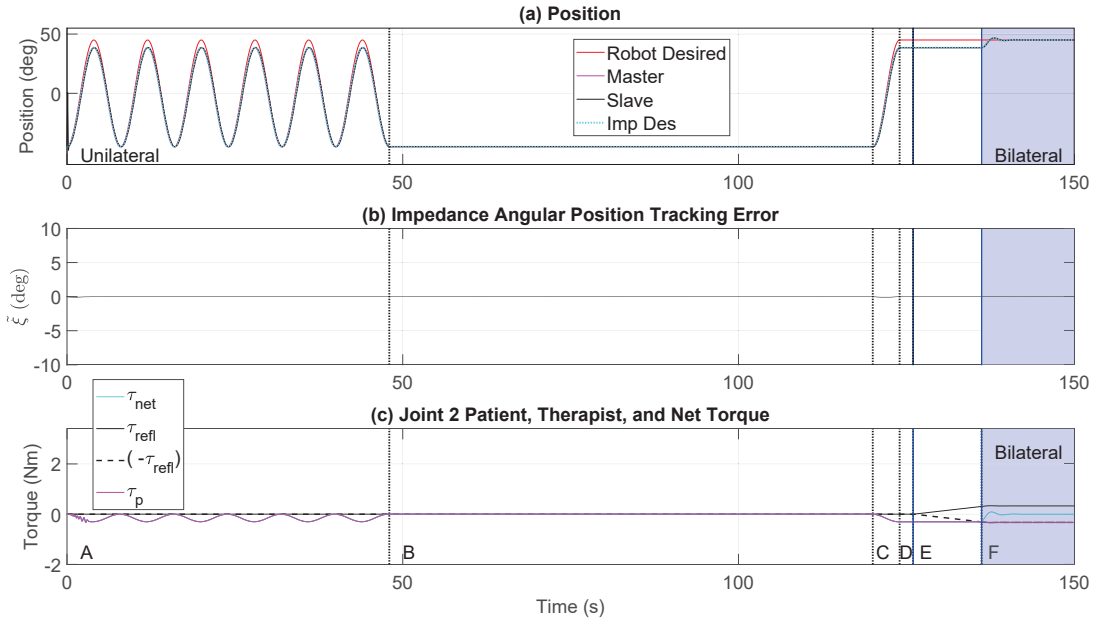


Figure A.43: Joint 2, Case 1 RBFNN-I telerehabilitation experiment with 2-DOF Exoskeletons using HTR: (a) position trajectory, (b) tracking error

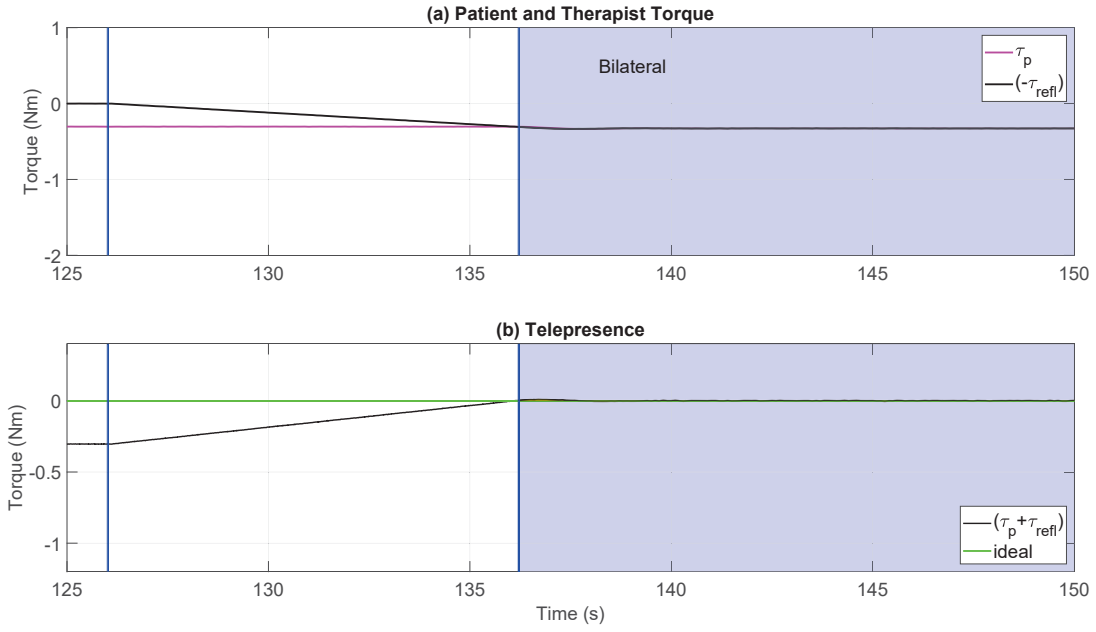


Figure A.44: Joint 2 torques for case 1 RBFNN-I telerehabilitation 2-DOF simulation with HTR during the bilateral phase

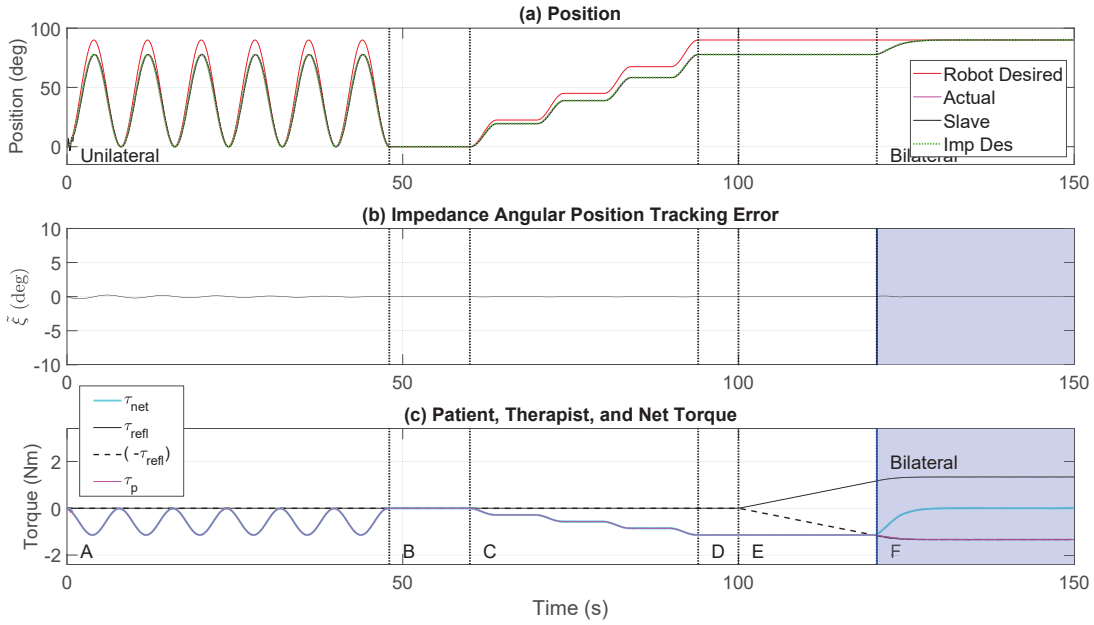


Figure A.45: Joint 1, Case 2 RBFNN-I telerehabilitation experiment with 2-DOF Exoskeletons using HTR: (a) position trajectory, (b) tracking error

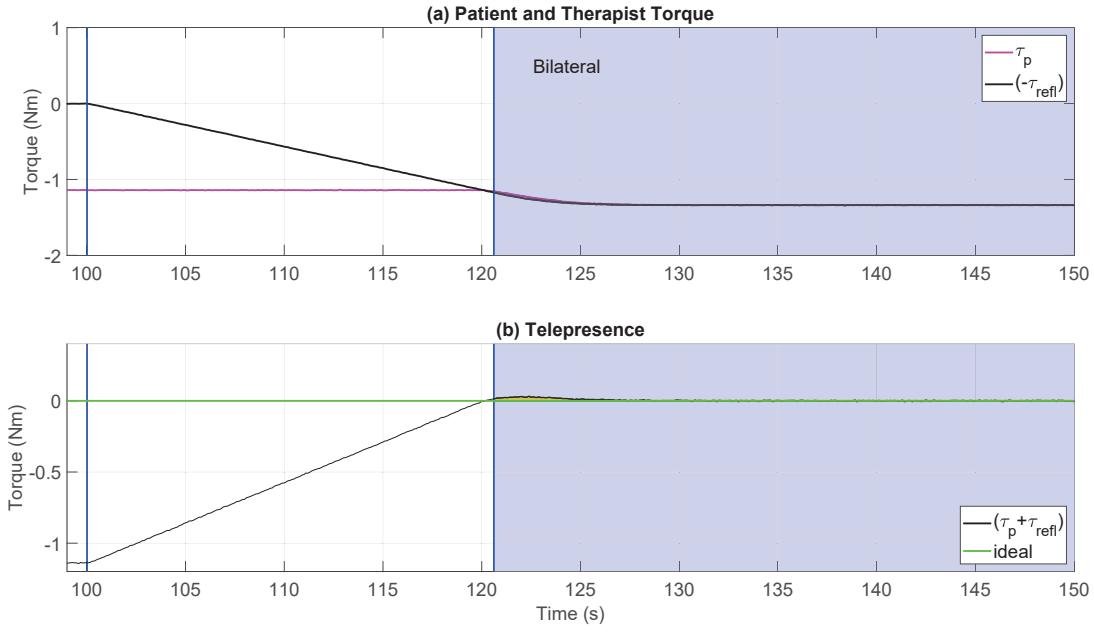


Figure A.46: Joint 1 torques for case 2 RBFNN-I telerehabilitation 2-DOF simulation with HTR during the bilateral phase

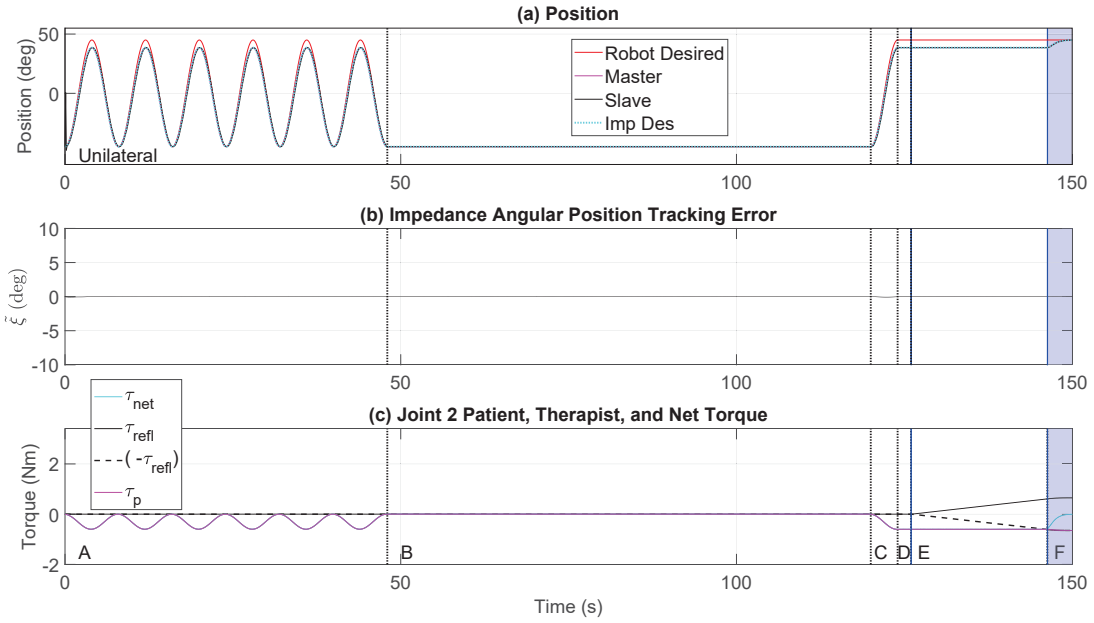


Figure A.47: Joint 2, Case 2 RBFNN-I telerehabilitation experiment with 2-DOF Exoskeletons using HTR: (a) position trajectory, (b) tracking error

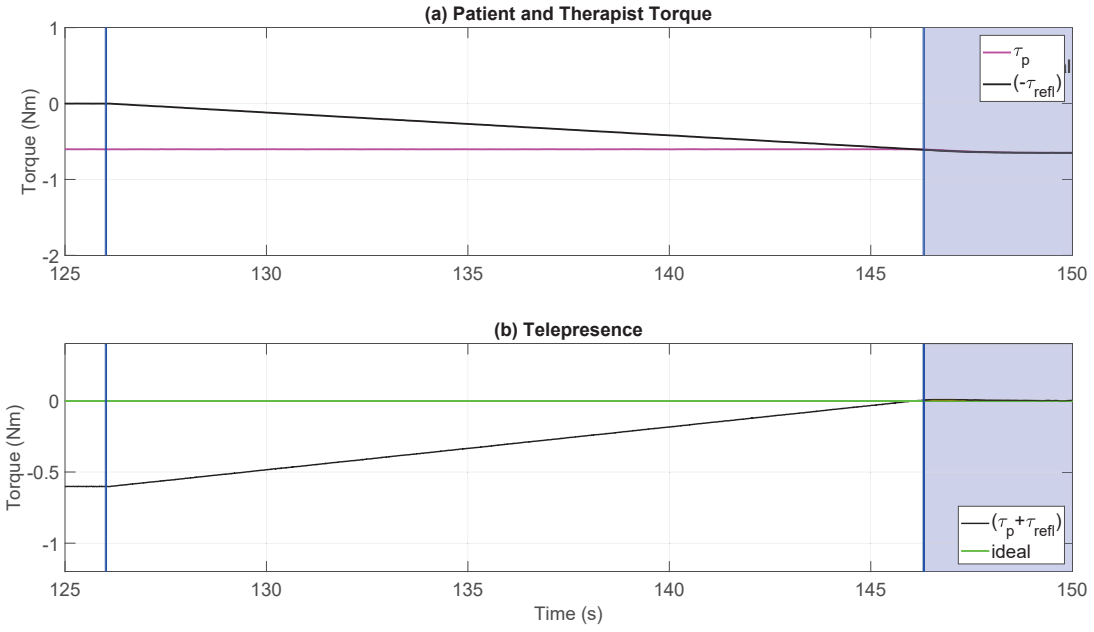


Figure A.48: Joint 2 torques for case 2 RBFNN-I telerehabilitation 2-DOF simulation with HTR during the bilateral phase

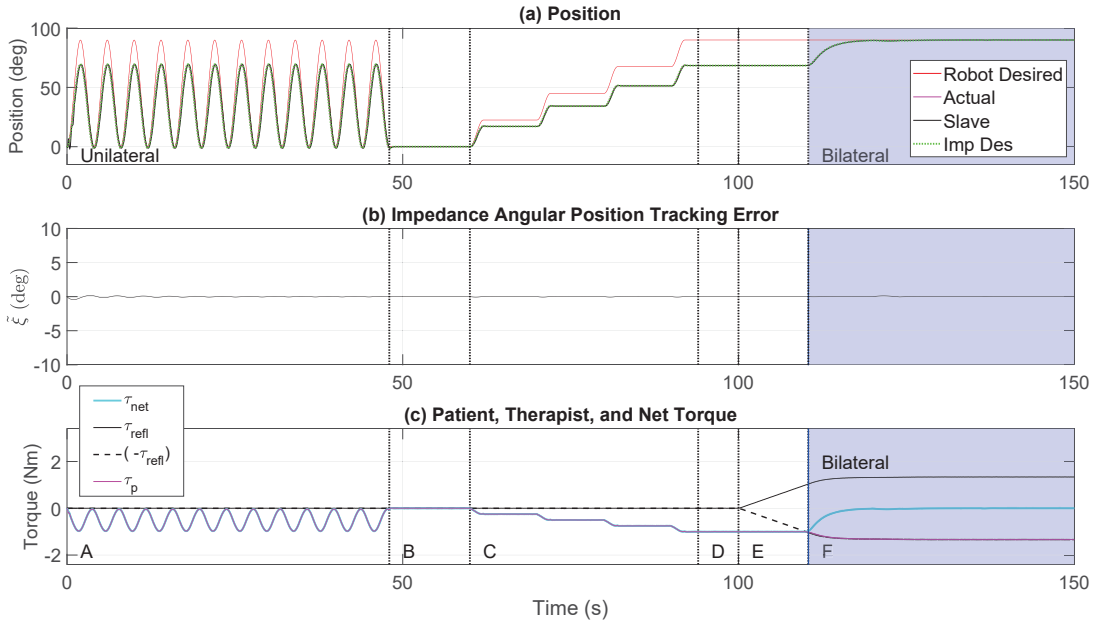


Figure A.49: Joint 1, Case 3 RBFNN-I telerehabilitation experiment with 2-DOF Exoskeletons using HTR: (a) position trajectory, (b) tracking error

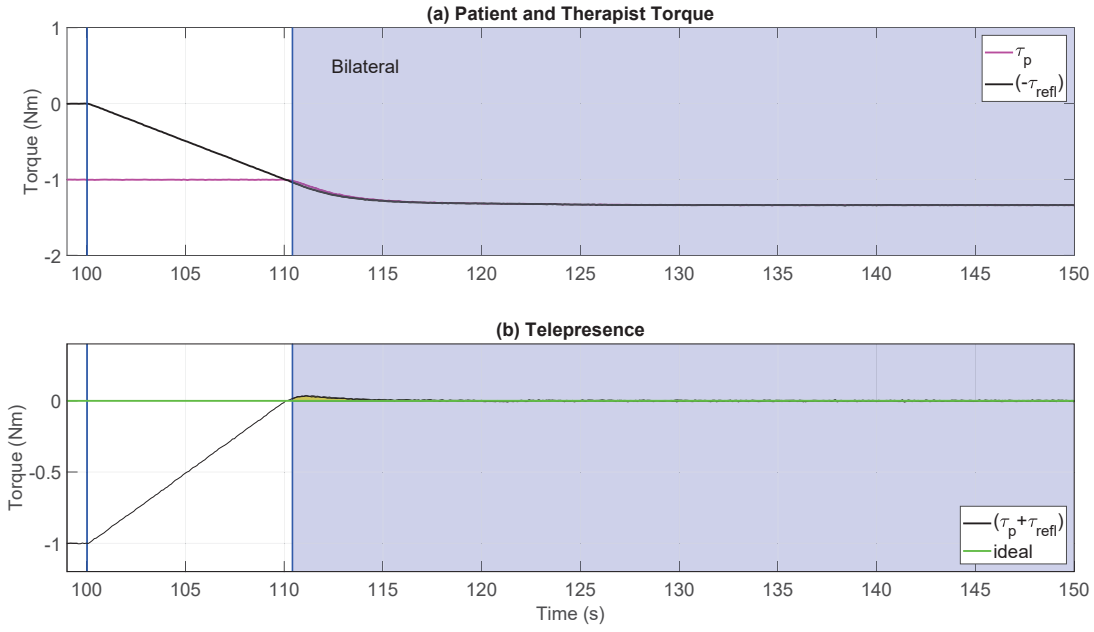


Figure A.50: Joint 1 torques for case 3 RBFNN-I telerehabilitation 2-DOF simulation with HTR during the bilateral phase

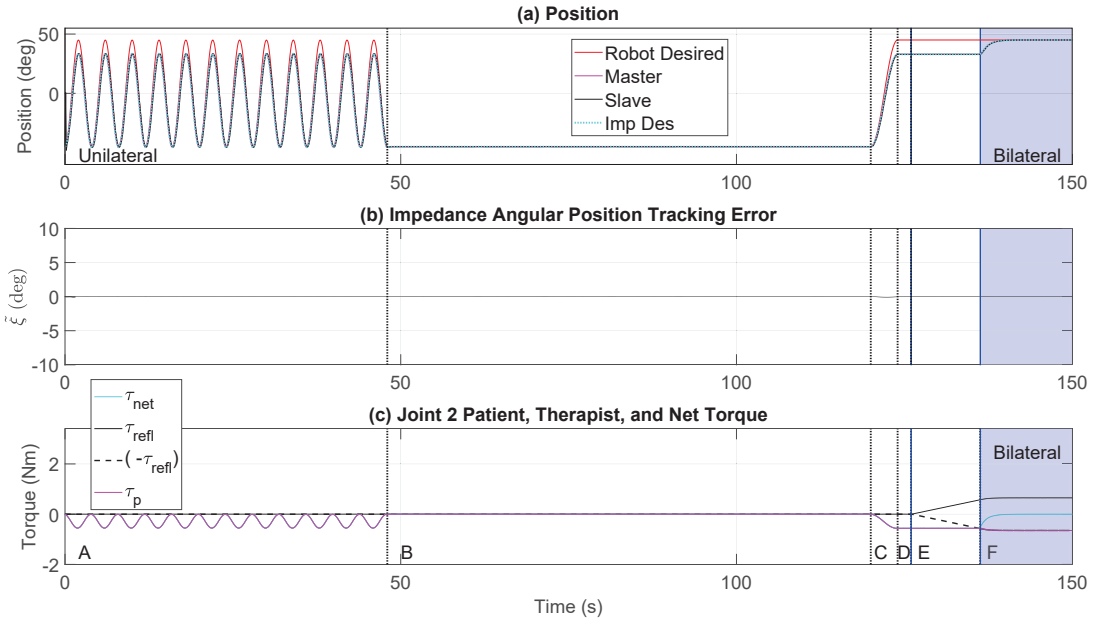


Figure A.51: Joint 2, Case 3 RBFNN-I telerehabilitation experiment with 2-DOF Exoskeletons using HTR: (a) position trajectory, (b) tracking error

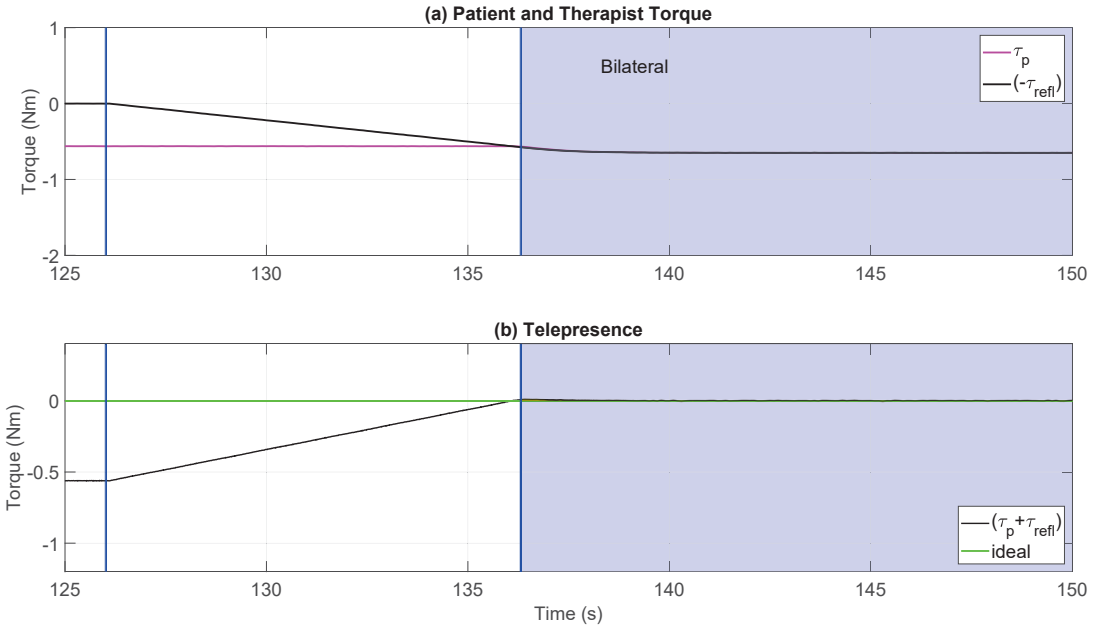


Figure A.52: Joint 2 torques for case 3 RBFNN-I telerehabilitation 2-DOF simulation with HTR during the bilateral phase

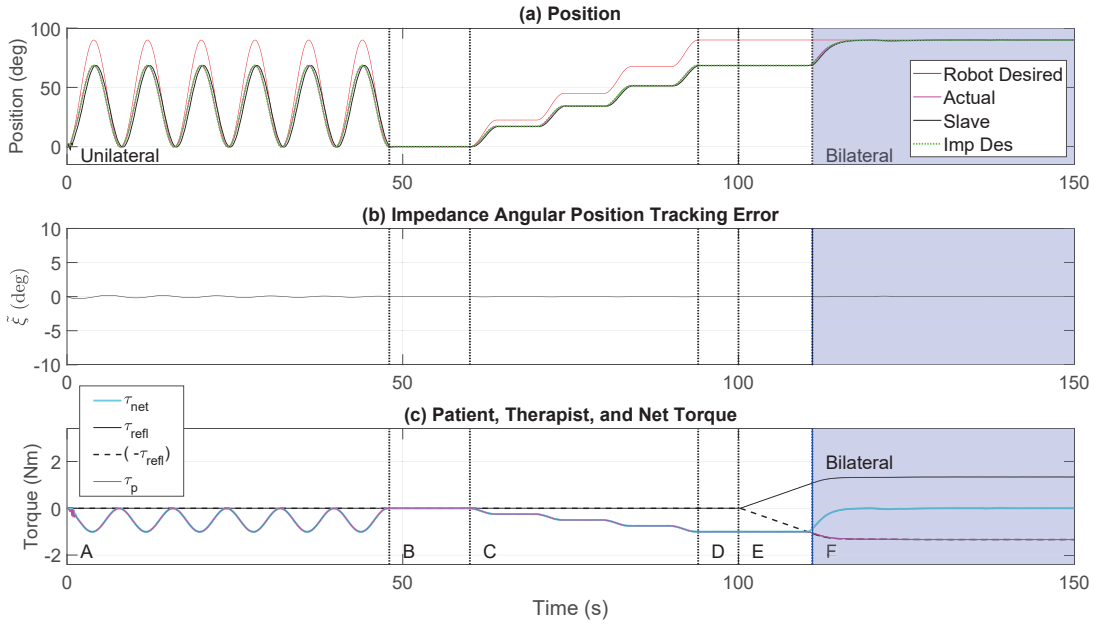


Figure A.53: Joint 1, Case 4 RBFNN-I telerehabilitation experiment with 2-DOF Exoskeletons using HTR: (a) position trajectory, (b) tracking error

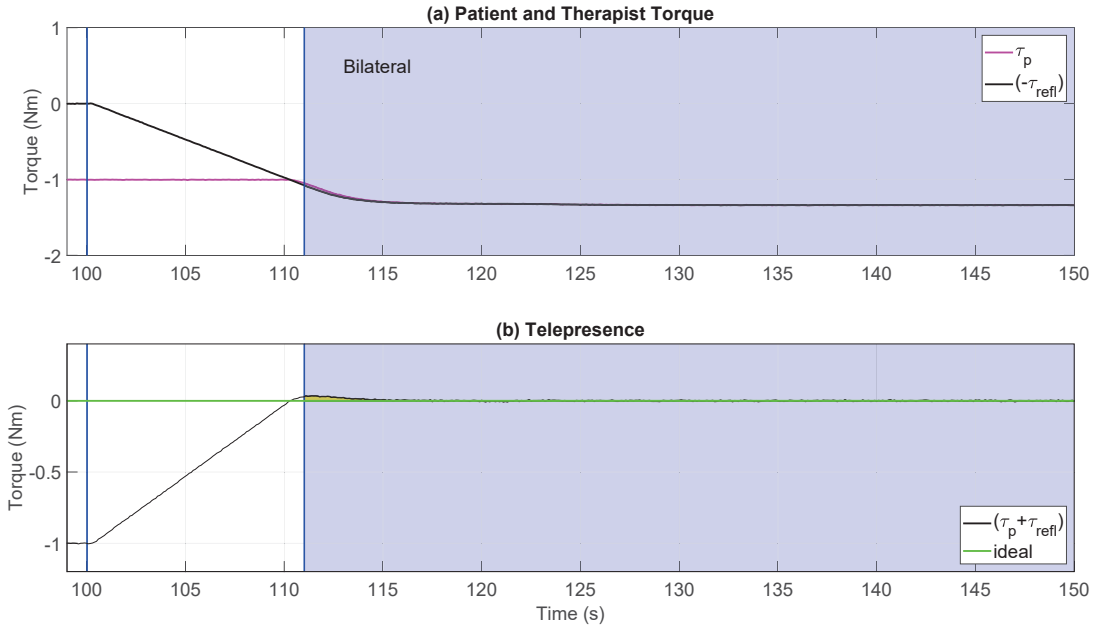


Figure A.54: Joint 1 torques for case 4 RBFNN-I telerehabilitation 2-DOF simulation with HTR during the bilateral phase

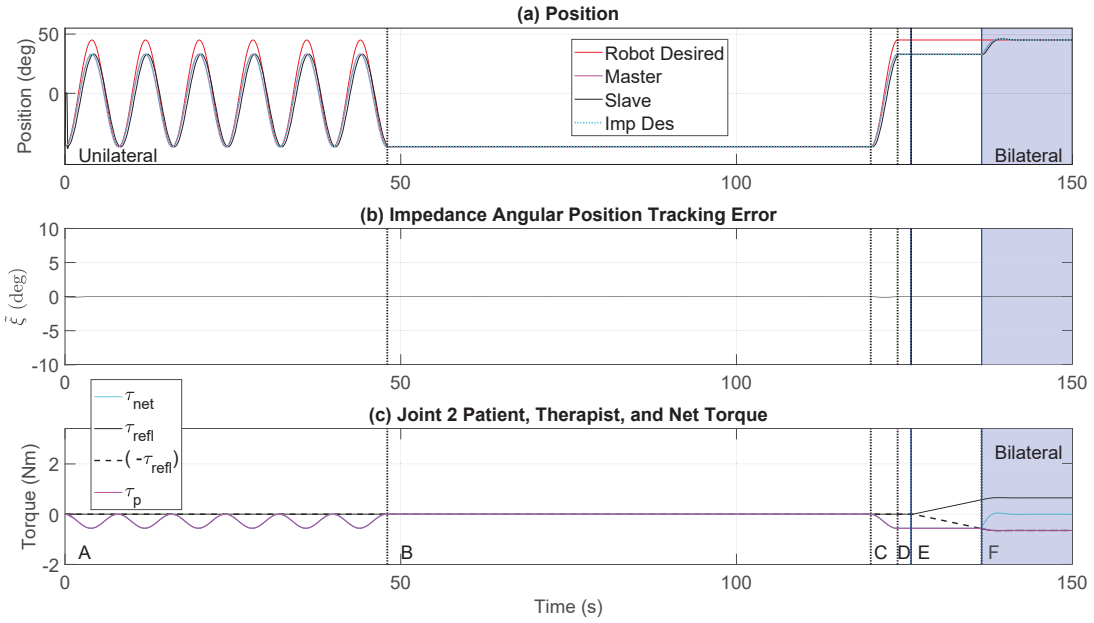


Figure A.55: Joint 2, Case 4 RBFNN-I telerehabilitation experiment with 2-DOF Exoskeletons using HTR: (a) position trajectory, (b) tracking error

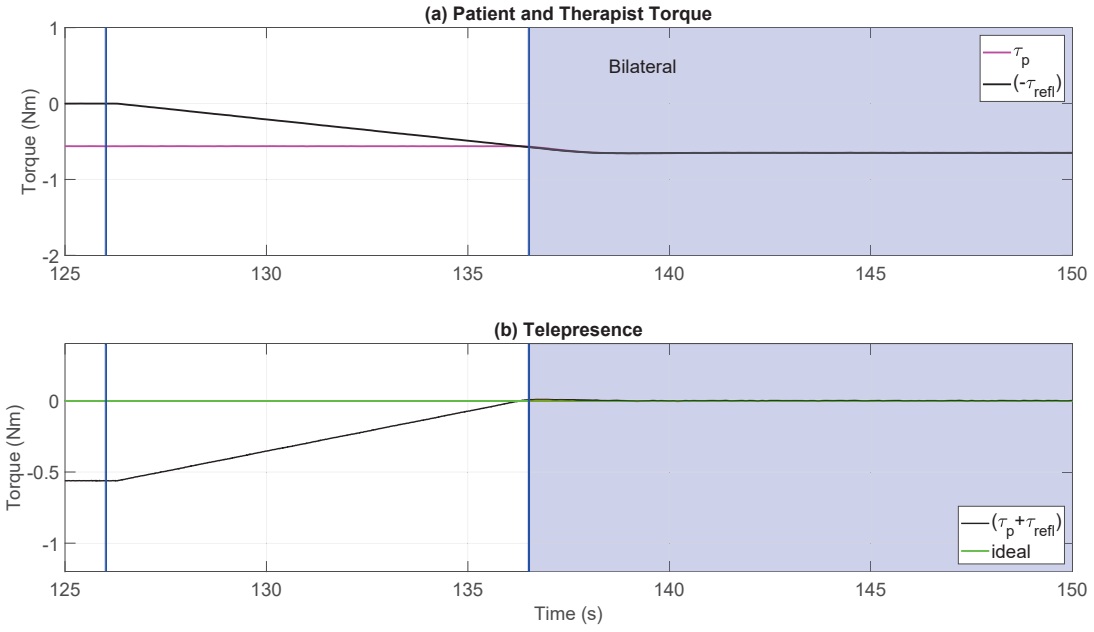


Figure A.56: Joint 2 torques for case 4 RBFNN-I telerehabilitation 2-DOF simulation with HTR during the bilateral phase

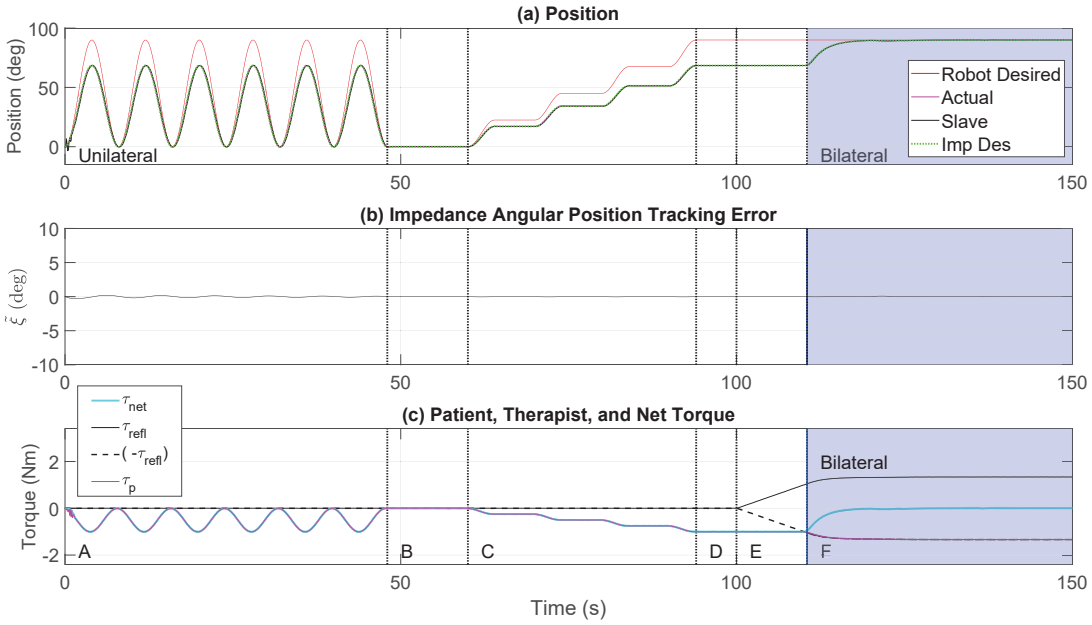


Figure A.57: Joint 1, Case 5 RBFNN-I telerehabilitation experiment with 2-DOF Exoskeletons using HTR: (a) position trajectory, (b) tracking error

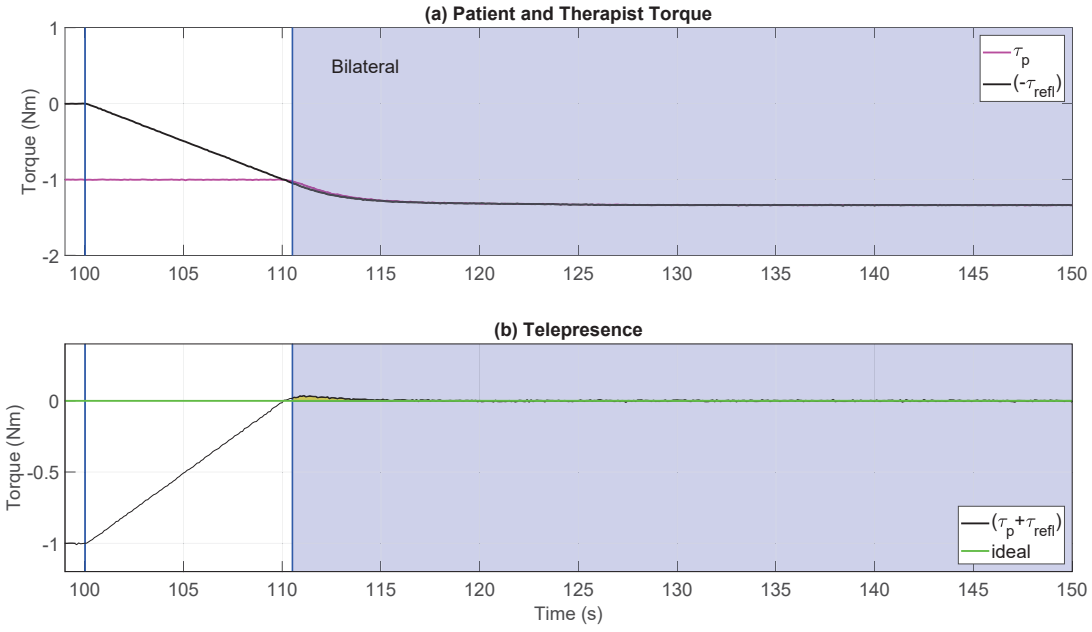


Figure A.58: Joint 1 torques for case 5 RBFNN-I telerehabilitation 2-DOF simulation with HTR during the bilateral phase

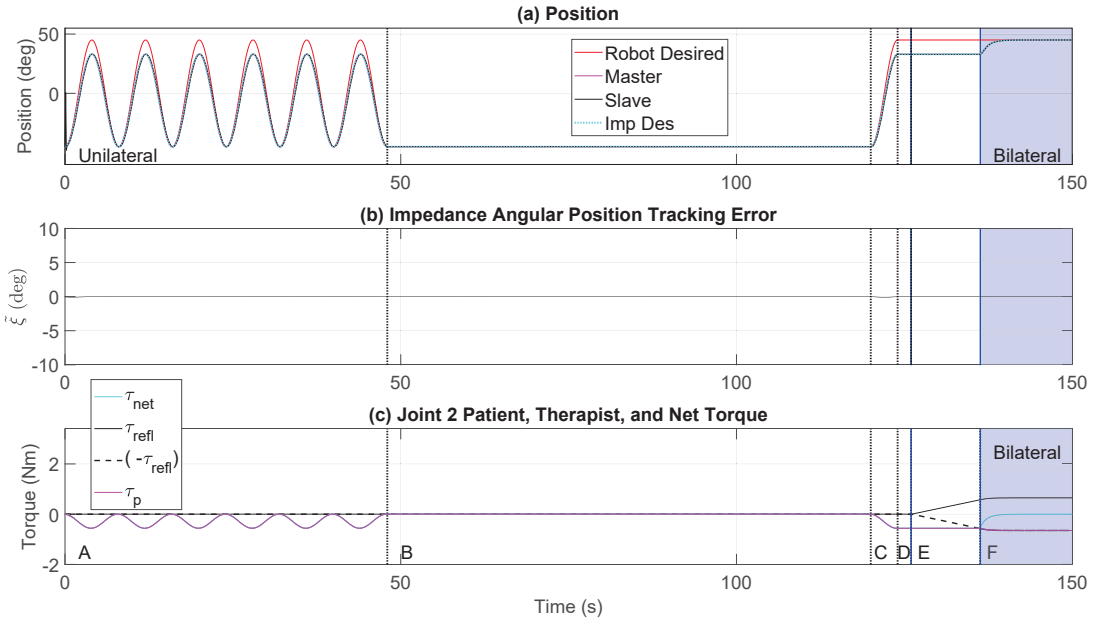


Figure A.59: Joint 2, Case 5 RBFNN-I telerehabilitation experiment with 2-DOF Exoskeletons using HTR: (a) position trajectory, (b) tracking error

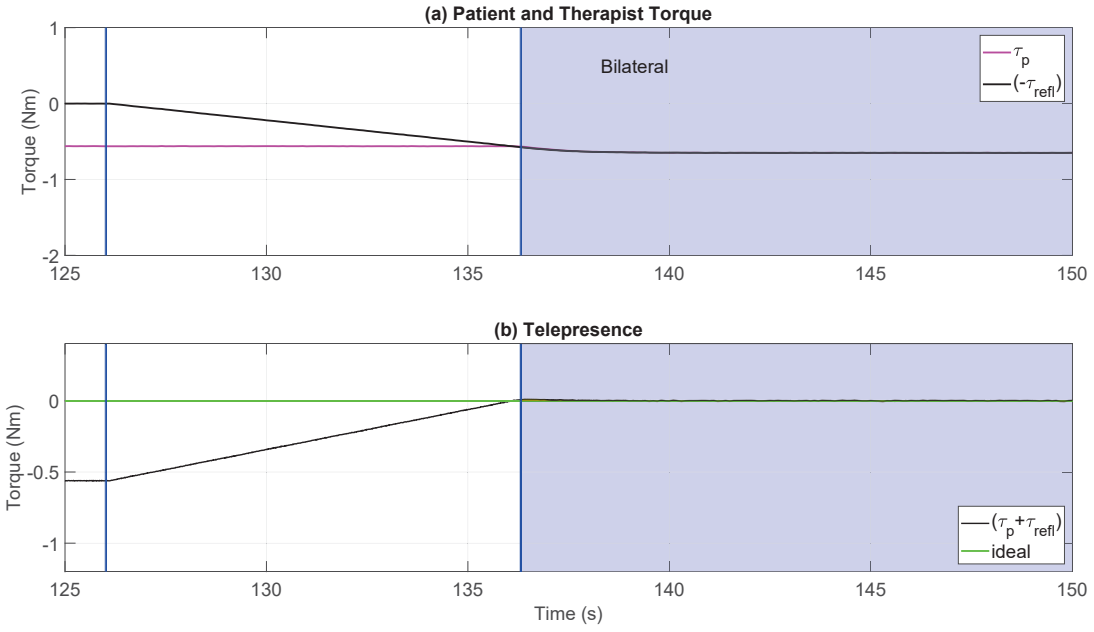


Figure A.60: Joint 2 torques for case 5 RBFNN-I telerehabilitation 2-DOF simulation with HTR during the bilateral phase

Appendix B

Author's Publications

1. **G. Bauer**, Y.J. Pan, R. Adamson, "Analysis of a Low-cost Sensor Towards an EMG-based Robotic Exoskeleton Controller," in *CSME 2018 International Congress Proceedings*, Canada, Toronto, (2018), Conference Date: May 2018.

2. H.H. Shen, Y.J. Pan, **G. Bauer**, "Manipulability-Based Load Allocation and Kinematic Decoupling in Cooperative Manipulations," in *2019 IEEE 28th International Symposium on Industrial Electronics (ISIE)*, Page Range: 1168-1173, Conference Date: June 2019.

3. **G. Bauer**, Y.J. Pan and H.H. Shen, "Adaptive Impedance Control in Bilateral Telerehabilitation with Robotic Exoskeletons," in *2020 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, Canada, Toronto, Page Range: 719-725, Conference Date: October 2020.

4. H.H. Shen, Y.J. Pan and **G. Bauer**, "Online Noise-Estimation-based Neighbor Selection for Multi-Manipulator Systems," in *Proceedings of the 21st IFAC World Congress*, Germany, Berlin, Conference Date: July 2020.

5. **G. Bauer** and Y.J. Pan, "Review of Control Methods for Upper Limb Telerehabilitation with Robotic Exoskeletons," in *IEEE Access*, doi: 10.1109/ACCESS.2020.3036596, Nov 2020.

6. **G. Bauer** and Y.J. Pan, "Telerehabilitation with Exoskeletons using Adaptive Robust Integral RBF-Neural-Network Impedance Control under Variable

Time Delays," Accepted in *2021 IEEE International Symposium on Industrial Electronics (ISIE)*, Japan, Kyoto, Conference Date: June 2021.