ROBOTIC DEVICES AND ADAPTIVE CONTROL STRATEGIES FOR ROBOTIC REHABILITATION AFTER STROKE

A Dissertation

Presented in Partial Fulfillment of the Requirements for the

Degree of Doctorate of Philosophy

with a

Major in Mechanical Engineering

in the

College of Graduate Studies

University of Idaho

by

Hossein Taheri

August 2014

Major Professor: Eric T. Wolbrecht, Ph.D.

Authorization to Submit Dissertation

This dissertation of Hossein Taheri, submitted for the degree of Doctorate of Philosophy with a Major in Mechanical Engineering and titled "Robotic Devices and Adaptive Control Strategies for Robotic Rehabilitation after Stroke," has been reviewed in final form. Permission, as indicated by the signatures and dates below, is now granted to submit final copies to the College of Graduate Studies for approval.

Major Professor:

_____ Date: _____
Eric T. Wolbrecht, Ph.D.

Committee Members:

_____ Date: _____
Touraj Assefi, Ph.D.

_____ Date: _____
Michael J. Anderson, Ph.D.

_____ Date: _____
Matthew Riley, Ph.D.

Department Administrator:

_____ Date: _____
John C. Crepeau, Ph.D.

Discipline's College Dean:

_____ Date: _____
Larry Stauffer, Ph.D.

Final Approval and Acceptance

Dean of the College of Graduate Studies:

_____ Date: _____
Jie Chen, Ph.D.

Abstract

Stroke is the leading cause of neurological disability in the US and stroke patients typically require an extensive rehabilitation therapy to regain some of their lost neuromuscular functionality. Various robotic devices have been developed for post-stroke rehabilitation to reduce the labor intensity of therapists and rehabilitation cost and provide therapists with quantitative information about rehabilitation procedure and patients recovery. The robot-patient interaction plays an important role in effectiveness of robotic therapy. Different strategies have therefore been proposed and employed to control rehabilitation robots in order to improve the therapy outcome. However, since the underlying neural mechanisms of motor recovery after stroke are not completely understood, and the effect of each stroke is unique and can be very different from that of other strokes, it is not clear what control strategy is the best.

Adaptive assist-as-needed (AAN) control is a movement training methodology with very desirable characteristics for rehabilitation robotic applications. It can adaptively modulate the level of robotic assistance to promote patient active involvement in therapy.

In this research, the evolution of two robotic devices for stroke rehabilitation are presented. The FINGER (Finger Individuating Grasp Exercise Robot) rehabilitation robot was designed to assist with hand and finger rehabilitation. A discrete performance-based adaptive control was implemented on FINGER that could provide patients with a suitable assistance level in order to modulate success during therapy game play. In a separate experiment, an inertial and directionally dependent AAN controller was implemented and tested on the FINGER robot.

Additionally, the design and development of a 4-DOF parallel robot for upper extremity impairment rehabilitation is presented. This end-effector type rehabilitation robot has low end-effector inertia, is very backdrivable, and is designed to counter-balance a significant portion of its own weight in order to reduce the need for the robot's actuators to overcome gravitational forces. Finally, an inertial adaptive AAN controller is proposed and tested using dynamic simulations.

Acknowledgements

This research was made possible by the grant NIH-R01HD062744 from the National Center for Medical Rehabilitation Research at the National Institute of Child Health and Human Development, and the National Center for Research Resources and the National Center for Advancing Translational Sciences, National Institutes of Health, through Grant UL1 TR000153.

I would like to thank the Mechanical Engineering faculty and staff and my committee members for their help and support. I also want to thank Dr. Dean Edwards and Dr. Touraj Assefi for their financial support while I was a graduate assistant at MRCI.

Finally, I would like to thank Dr. Eric Wolbrecht and ME students: Curtis Bower, Kyle Gray, David Gardner, Stephen Goodwin, Kevin Witkoe and James Tigue, as well as Dr. David Reinkensmeyer, Justin Rowe and Vicky Chan of UC Irvine who collaborated on this research.

Table of Contents

Table of Figures

List of Tables

Chapter 1. Introduction

Robots have long been used in automation and industrial applications for tasks that require high repeatability and accuracy. There is a growing tendency to use robots in healthcare industry. Healthcare quality is an essential factor in improving quality of life and there is an increasing demand for automation in healthcare due to problems such as population aging and increase in healthcare costs [1, 2]. Extensive research has been dedicated to medical robots to fulfill this demand. One major application of robots is in stroke rehabilitation [3]. Due to the adaptive nature of the central nervous system as well as a need for compliant rehabilitation robots, using adaptive control methods can be potentially very beneficial. The overall goal of this research therefore is to investigate adaptive control strategies for robotic rehabilitation as well as robotic devices that are suitable for both movement training therapy and adaptive control. In this chapter, an overview of previous rehabilitation robots in literature is provided followed by examples of adaptive control in rehabilitation robotics. This is followed by a description of chapters of this thesis.

1.1. Rehabilitation Robots

The early rehabilitation robots were focused on replacing lost motor functions of the disabled individuals using devices such as feeding devices, robotic wheelchairs and orthotic robots [4]. During the last several decades, there has been an increase in amount of research into physical therapy through robots [3, 5-7]. Rehabilitation robots can be considered as devices that use active feedback control to provide patients with assistance or resistance during rehabilitation exercise. In this section motivation and some notable robotic systems used for rehabilitation will be discussed

1.1.1. Motivation for Rehabilitation Robotics

Traditional physical therapy typically requires moving patient's impaired limb manually by a therapist. This process can be very labor intensive [8-10]. Therefore, conventional therapy usually leads to fatigue in therapist that consequently limits the efficiency of therapy sessions. Methods that can overcome this limitation can be very beneficial because repetition and intensity of therapy is a key factor in patient's functional improvement. A key characteristic of robots is that they are well suited for repetitive tasks. Robots that are designed to have suitable spatial and force capabilities can partially take on the physical role of the therapists, enabling longer therapy sessions with higher intensity. Moreover, therapists can potentially conduct treatment of multiple patients simultaneously a supervisory manner to increase the cost efficiency of therapy [7, 11].

Another advantage of using robots in rehabilitation is their ability to collect large amounts of quantitative data. Unlike human therapists that mainly operate based on their qualitative and subjective assessment of patient's performance, robots that are equipped with appropriate sensors can provide therapists with more objective data [12, 13]. It makes it easier to monitor patient's progress over time and also makes it easier to compare effect of different methods and strategies among different patients. So, besides the potential to improve rehabilitation practices, robots can offer a lot to rehabilitation research and neuroscience.

Robotic systems usually communicate with patients through graphical user interfaces. In many cases the graphical interface has the form of video games [14-17]. In such games, the desired motion is presented to patients by displaying the desired trajectory on the screen and asking patients to use their effected limbs to follow it. Video games have made rehabilitation more engaging and patients feel motivated to complete intensive therapy tasks.

Several rehabilitation robots have passed clinical trials and are commercially available. Clinical trials have confirmed the effectiveness of rehabilitation robots, suggesting that intensive robotic therapy can improve functional recovery and motor learning [18]. Many rehabilitation hospitals use these robots as an effective tool in their treatment plans. Examples of robotic devices being largely used at rehabilitation hospitals include Lokomat, ArmeoSpring and ArmeoPower [19-24].

Figure 1.1. Some commercially available rehabilitation robots. Lokomat (top-left). ArmeoSpring (top-right). ArmePower (bottom).

1.1.2. Assistance Provided by Rehabilitation Robots

There are two main purposes of using robots for rehabilitation: using robots to provide support and assistance for patients to perform some activities of daily living (ADL) [25-30] and utilizing robots as physical therapy tools [31-35]. The types of motion/force assistance provided by rehabilitation robots can be categorized into four main groups of active devices,

passive devices, haptic devices and coaching devices [3]. Most devices lie in the category of active devices. These are devices that can actively exert forces and move the effected limbs through use of actuators. A forced assistance from the robot is usually needed when the patients are weak and cannot complete the tasks by themselves. Passive devices, on the other hand, are those with passive force elements such as springs or brakes [34, 36, 37]. Another group of rehabilitation robots are haptic devices that provide patients with somatosensory input during patient-robot interaction [38, 39]. These devices can be active or passive, but their main characteristic is their ability to provide tactile sensation to users. Coaching devices are those that don't generate any forces but are equipped with sensors to record patient's performance [40]. These signals can be used to communicate with video games to provide patients with some sort of visual feedback or for measurement and analysis of patient's progress over time. Devices that use video-based motion capture, e.g. Microsoft Kinect®, wearable sensor networks and sensor gloves belong to this group.

1.1.3. Physical Interaction between Patients and Robots

Considering types of mechanical interactions between motion therapy robots and patients, two major mechanical structure paradigms stand out: end-effector-based and exoskeleton-based. The two groups are different in how they are connected to patients. End-effector-based devices are connected to patient through their most distal limb. For example, in end-effector-based upper extremity motion therapy robots, a patient may grasp a handle that is connected to robot's end-effector and performs therapy tasks while patients hand and the robot's end-effector move together. Examples of end-effector-based systems are the MIT-Manus [31], ARM guide [32], ACRE [41], CRAMER [42] and NeReBot [33]. Although some sort of wrist of forearm support might be provided, the main port of interaction

between patient and robot is robot's end-effector. The main advantage of end-effector-based systems is their simplicity and it makes design and manufacturing much easier.

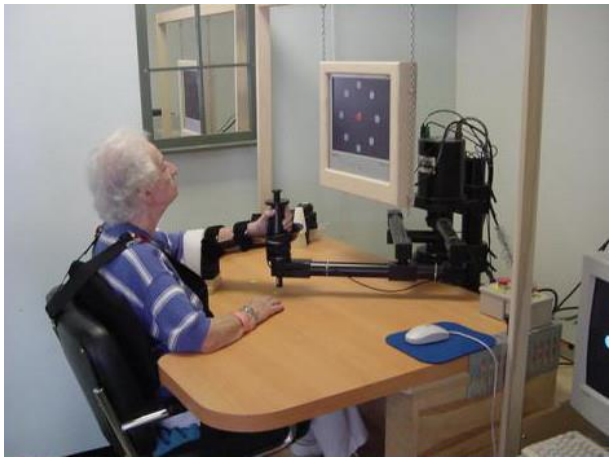

Figure 1.2. MIT-Manus.

However, since the patient's limb is constrained at only one point, it is impossible to force all the joints of patient's effected limb to move in a predetermined desired trajectory that stems from inherent redundancy of human limbs such as arms and hands. Exoskeleton-based robots can be used to overcome this disadvantage. Exoskeletons are designed to mimic human's skeletal structure and hence make it possible to control individual joints of the effected limb. Exoskeleton-based systems normally require complicated geometric and kinematic design. One of the main issues in designing exoskeleton-based robots is joint alignment as rotation axes of many of the joints in human body are complex and different between patients. It is also hard to make the exoskeleton-based robots adjustable to limps of patients with different body sizes. Examples of exoskeleton-based rehabilitation robots include Lokomat [21], Pneu-WREX [35], T-WREX [40] and CADEN-7 [43]. There are also devices that incorporate both end-effector interaction and exoskeleton structure such as BONES [44] and MIME-Rice Wrist rehabilitation system [45].

Figure 1.3. Pneu-WREX.

1.1.4. Actuation

While many researchers have designed robots specifically for rehabilitation purposes, some researchers have used industrial robots in their robotic rehabilitation research. For example, REHABROB system [46] uses the ABB industrial robots IRB140 and IRB1400H, and MIME [47] uses the PUMA560 robot. Using industrial robots can be more straightforward and less expensive [3], but they have a significantly higher impedance compared to human limbs and hence it can be potentially dangerous for patients to interact with such rigid and powerful devices [48]. Also, it is desired that the patient's force can move the robot when the robot is powered off. For this reason, many robots used for rehabilitation applications are backdrivable, e.g. FINGER [49], HWARD [50] and RehabExos [51]. Backdrivability is a qualitative property and usually the system should have low inertia, friction and gear ratio to be considered backdrivable.

Most rehabilitation robots are powered by electric actuators. Electric actuators are clean and it is typically easy to provide electrical energy to them. They can produce high power and are available with various sizes. However, the main problem of electric actuators is that they typically require high gear ratios and hence are not very backdrivable. Pneumatic actuators

are also frequently used in rehabilitation robots. They main advantage of pneumatic systems is their high power to weight ratio and low impedance and hence good backdrivability. Examples of pneumatic actuation in rehabilitation robots include Pneu-WREX [35] and ASSIST [52]. Pneumatic Artificial Muscles are also used in rehabilitation robotics. These actuators are lightweight and backdrivable, however, they only produce force in one direction and have non-linear behavior and are not easily controllable.

Hydraulic actuators are also used in rehabilitation robotics. These systems are heavy and have very high impedance. Moreover, they require frequent maintenance, can be noisy and may exhibit fluid leakage.

Functional Electrical Stimulation (FES) is a type of actuation that doesn't require external power generation [53, 54]. In this method, electrical stimulus signals are sent to users muscles to activate them. Since an external actuator doesn't exist in such systems, all the work is done by user's muscles and it can improve the muscle strength and reduce the chance of muscle atrophy that can be consequently beneficial [55]. However, FES can cause strong involuntary muscle reactions and pain and also it is very hard to control due to nonlinear behavior of muscles.

1.2. Control Systems

In this section, the typical control signals and control strategies in rehabilitation robotics are discussed. Much of the rehabilitation robotics research has been focused on developing complex, multiple-degree-of-freedom mechanisms that can assist or support patients with complicated movements during motion therapy. There has been also research on finding effective control strategies for rehabilitation robots [15]. By using suitable control strategies, it is possible to control how robots interact with patients that can be beneficial. For instance,

selecting a control strategy that stimulates motor plasticity can improve motor recovery. Unfortunately, the relationship between rehabilitative control strategies and motor recovery are not fully understood. As such, findings and concepts from rehabilitation, motor learning, neurophysiology, etc. are used to design appropriate control systems.

1.2.1. Control signals

Many systems with relatively simple control structures use on-off switches as control signals. These signals can be activated, for instance, when the user reaches to a certain region in the task space. More complicated control systems, however, require higher level signals such as position or velocity, typically measured at actuator joints. Force and torque signals are also widely used in force control strategies. Magnitude of force can be directly provided by actuators or special load cells can be used to measure the interaction between the robot and user.

Surface Electromyography (sEMG) is also widely used as a control signal. In this method muscle activation signals received from the central nervous system is recorded at the skin surface and can be used to predict person's intention to perform a particular movement. These signals, however, are very noisy and hard to analyze. Moreover, it is not easy to record sEMG of small muscles and does that are not exposed to skin.

1.2.2. Control Strategies

According to [15], the control strategies used in rehabilitation robotics can be categorized into two main groups of high-level and low-level strategies. High-level strategies are those that are specifically designed to induce motor plasticity. These strategies then utilize a series

of low-level control strategies in their implementation such as PID position control, impedance control, or admittance control.

The most common high-level control strategy is the assistive strategy. In assistive control strategy, the robot is used to provide assistance to patient in order to complete a task that the patient is unable to completely accomplish without assistance. The four main types of assistive control strategy are impedance-based [56, 57], counterbalance-based [58, 59], EMG-based [60, 61] and performance-based adaptive control [62-64].

Unlike assistive strategies, challenge-based strategies are high-level control strategies that, instead of making the task easier for patients, make it more challenging to complete. For instance, some robotic systems use resistive strategies based on practiced clinical rehabilitation techniques [56, 65].

Constraint-induced strategies are another type of challenge-based strategies. This strategy is based on the notion that constraining the un-effected limbs forces the patient to use the effected-limb more frequently, which can lead to faster motor learning and hence improve recovery. Examples of constraint-based systems include [66, 67].

Another type of challenge-based strategies is error-amplification. This approach argues that kinematic errors are the main signals that drive motor learning and hence amplifying them during rehabilitation exercises can be beneficial. Patton and colleagues [68, 69] and Reinkensmeyer and colleagues [70, 71] have incorporated this strategy in their studies.

Other types of high-level control include haptic simulation [72] and non-contacting coaching [73].

1.2.3. Adaptive Control

Adaptive control is a control method in which controller parameters are not static and adapt over time. This method can be useful when the plant parameters vary or are uncertain. Using adaptive control in rehabilitation robotics has the advantage that controller can be tuned according to user's needs online. For example, a robot that uses an adaptive controller can take into account patient's intention instead of imposing a fixed desired motion [16, 63, 74].

Parameter adaptation in adaptive control typically has the following form

$$P_{i+1} = fP_i - ge_i \tag{1.1}$$

where P_i is the adaptive control parameter at time i , e_i is the error signal at time i , f is the forgetting factor and g is the adaptation rate. Adaptation in this form of control is based on user's performance during each trial and control parameters can be controller gains, parameters of desired motion planning such as reaching time or maximum velocity or parameters that determine interaction characteristics between user and robot such as impedance.

The main reason for including a forgetting factor in this form of adaptation is to prevent patient slacking. If there is no forgetting ($f = 1$), regardless of patient's participation, the robot always adapt to parameters that minimize performance error. This may encourage participants to back off and let the robot take over completing task. However, with inclusion of some forgetting ($0 < f < 1$), the adaptation algorithm tends to reduce the parameters even if the performance error is very small. This can systematically reduce the assistance force and consequently increase the voluntary participation of patients.

Use of adaptive controllers in rehabilitation robotics can also be looked at from an Assist-as-Needed (AAN) framework point of view. It can provide a good mechanical compliance and

can calculate the needed amount of force to cancel effects tone, weakness and lack of coordinated motor control. Wolbrecht et. al. proposed an adaptive AAN controller that uses a model of patient's impairment based on radial-basis functions and uses a passivity-based adaptation algorithm [62, 75]. In this method, a feedforward term, comprised of RBF's and adaptation parameters, is used to model existing effects of impairment. A schematic of the method is shown in Figure 1.4.

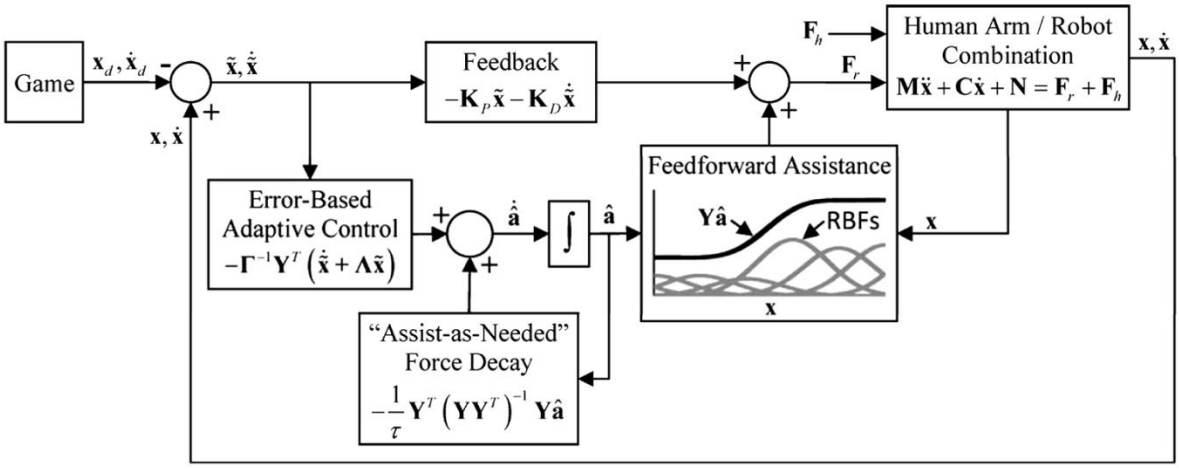

Figure 1.4. Schematic of the adaptive assist-as-needed control proposed by Wolbrecht et. al. [62].

The robot force at the location of interface is determined using

$$\mathbf{F}_r = \mathbf{Y}\hat{\mathbf{a}} - \mathbf{K}_P\tilde{\mathbf{x}} - \mathbf{K}_D\dot{\tilde{\mathbf{x}}} \tag{1.2}$$

where $\tilde{\mathbf{x}}$ and $\dot{\tilde{\mathbf{x}}}$ are vectors of position and velocity errors, $\mathbf{K}_P$ and $\mathbf{K}_D$ are symmetric, constant, positive definite matrices, $\hat{\mathbf{a}}$ is the vector of parameter estimates and $\mathbf{Y}$ is the regressor matrix defined as

$$\mathbf{Y} = \begin{bmatrix} \mathbf{g}^T & 0 & \cdots & 0 \\ 0 & \mathbf{g}^T & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \mathbf{g}^T \end{bmatrix} \tag{1.3}$$

where $\mathbf{g}$ is an $p \times 1$ vector of RBF's defined as

$$\mathbf{g}_n = \exp\left(-\frac{\left\| \mathbf{x} - \mu_n \right\|^2}{2\sigma^2} \right) \tag{1.4}$$

where μ_n is the location of n^{th} RBF and σ is the dilation of RBF's. The following adaptation mechanism is used to adapt parameter values.

$$\dot{\hat{\mathbf{a}}} = -\frac{1}{\tau} \mathbf{Y}^T \left(\mathbf{Y}\mathbf{Y}^T \right)^{-1} \mathbf{Y}\hat{\mathbf{a}} - \Gamma^{-1}\mathbf{Y}^T \left(\dot{\tilde{\mathbf{x}}} + \Lambda\tilde{\mathbf{x}} \right) \tag{1.5}$$

Where Γ and Λ are symmetric $n \times n$ positive definite matrices and τ is the forgetting time constant. The forgetting term reduces the robots actuator output when error is small, thus limiting the robot's force and encouraging the patients to participate while receiving just adequate assistance to complete tasks. For more details and stability proofs see [62].

1.3. Stroke

Stroke is any brain malfunction due to disturbance in blood supply to the brain. It can be due to either rapture of blood vessels (hemorrhage) or blockage of a blood vessel (ischemia). Stroke can lead to death of brain tissues that can consequently cause severe sensory and motor impairments.

There are about of 6.8 million Americans who have had a stroke and are living with the aftereffects [76]. Each year, over 795,000 people experience a new stroke. It is equal to an average of one stroke every 40 seconds, making stroke the leading cause of neuromuscular disability in the US. By 2030 an additional 3.4 million people will have a stroke that is a 20.5% increase in prevalence. The total cost of stroke in 2010 was $36.5 billion. The direct medical cost of stroke will be three times higher by 2030.

Among ischemic stroke survivors, 50% suffer some hemiparesis, 30% are unable to walk without assistance and 26% are dependent in activities of daily living. Figure 1.5 shows some statistics about severity and prevalence of stroke among different ages and genders.

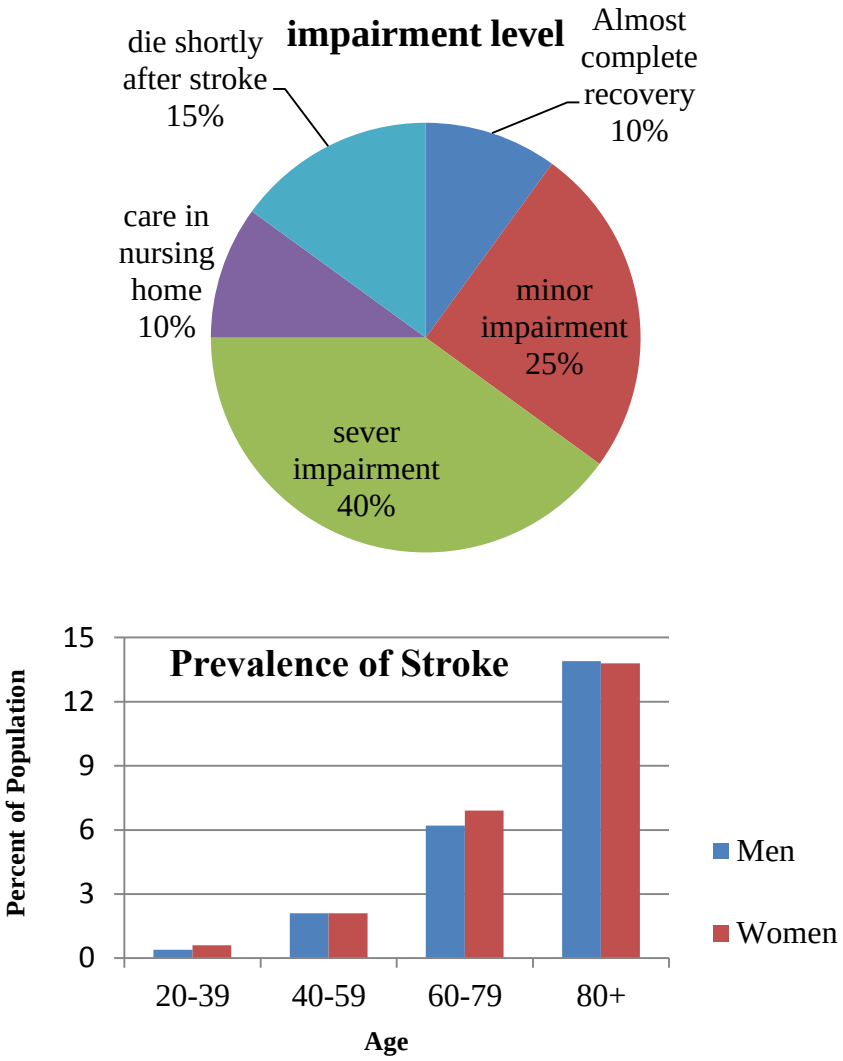

Figure 1.5. Stroke statistics: Severity of post stroke impairment (top) and prevalence of stroke (bottom).

Although traditional post stroke rehabilitation therapy has shown to be effective, due to high cost of stroke and aging problem in the years to come, tools that automate the rehabilitation process can be very beneficial. Based on reasons discussed in section 1.1.1, rehabilitation robotics can be a potential solution. It is automated, provides clinicians with quantitative

information about rehabilitation process and patient's progress and can bring fun and engagement to post-stroke rehabilitation exercises.

1.4. Thesis Outline

This thesis details the work carried out by the author on rehabilitation robotics. Chapter 2 details the design procedure, and studies done on FINGER rehabilitation robot. Content of Chapter 2 was previously published in the journal of Neuro-Engineering and Rehabilitation. Chapter 3 that is also from a previous publication details studies on a state-dependent adaptive control strategy that include inertial effects as well as directional-dependent impairment effects. The proposed control system was tested on 5 healthy subjects using FINGER robot. A 4 DOF parallel robot for upper extremity rehabilitation exercises is proposed in Chapter 4 and its kinematics and dynamics modeling is explained. In Chapter 5, an adaptive control strategy that takes into account inertial assistance is proposed. The results of this controller is then compared to results of the adaptive AAN controller in [62]. Finally, discussion and conclusions are provided in Chapter 6.

Chapter 2. Design and Preliminary Evaluation of the FINGER Rehabilitation Robot: Controlling Challenge and Quantifying Finger Individuation during Musical Computer Game Play

Hossein Taheri[1], Justin B Rowe[2], David Gardner[1], Vicki Chan[3], Kyle Gray[1], Curtis Bower[1], David J Reinkensmeyer[2,4,5], Eric T Wolbrecht[1]

Journal of NeuroEngineering and Rehabilitation, vol. 11, p. 10, 2014.

2.1. Abstract

Background-This paper describes the design and preliminary testing of FINGER (Finger Individuating Grasp Exercise Robot), a device for assisting in finger rehabilitation after neurologic injury. We developed FINGER to assist stroke patients in moving their fingers individually in a naturalistic curling motion while playing a game similar to Guitar Hero®[6]. The goal was to make FINGER capable of assisting with motions where precise timing is important.

Methods - FINGER consists of a pair of stacked single degree-of-freedom 8-bar mechanisms, one for the index and one for the middle finger. Each 8-bar mechanism was designed to control the angle and position of the proximal phalanx and the position of the middle phalanx. Target positions for the mechanism optimization were determined from trajectory data collected from 7 healthy subjects using color-based motion capture. The resulting robotic device was built to accommodate multiple finger sizes and finger-to-finger

[1] Mechanical Engineering Department, University of Idaho, Moscow, ID, USA
[2] Department of Biomedical Engineering, University of California, Irvine, CA, USA
[3] Department of Rehabilitation, University of California, Irvine, CA, USA
[4] Department of Mechanical and Aerospace Engineering, University of California, Irvine, CA, USA
[5] Department of Anatomy and Neurobiology, University of California, Irvine, CA, USA
[6] Guitar Hero® is a trademark of Activision Publishing, Inc.

widths. For initial evaluation, we asked individuals with a stroke (n=16) and without impairment (n=4) to play a game similar to Guitar Hero® while connected to FINGER.

Results - Precision design, low friction bearings, and separate high speed linear actuators allowed FINGER to individually actuate the fingers with a high bandwidth of control (-3dB at approximately 8 Hz). During the tests, we were able to modulate the subject's success rate at the game by automatically adjusting the controller gains of FINGER. We also used FINGER to measure subjects' effort and finger individuation while playing the game.

Conclusions-Test results demonstrate the ability of FINGER to motivate subjects with an engaging game environment that challenges individuated control of the fingers, automatically control assistance levels, and quantify finger individuation after stroke.

Keywords-Robotic rehabilitation, Stroke, motor control, finger individuation.

2.2. Background

Over the past several decades, researchers have developed robotic devices for rehabilitation therapy after stroke. This is in response to a sizable need, with nearly 800,000 people per year suffering a stroke in the United States alone [76]. Of the survivors, approximately two-thirds experience long-term impairment of their affected upper-extremity [77]. Robotic therapy devices can automate the repetitive and strenuous aspects of conventional physical therapy. Furthermore, robotic therapy devices can serve as scientific instruments for quantifying the recovery process, and thus may provide insight that is not normally available with conventional therapy alone.

Robot assisted therapy of the upper extremity following stroke has been shown to be as effective as, and in some cases modestly more effective than, conventional therapy (for reviews see [9, 56, 78-80]). Research with robotic therapy devices supports the contention

that motor recovery increases with therapy intensity [80], i.e. more practice is better. What remains unclear, however, is how a rehabilitation robot should interact with the patient in order to optimize recovery during practice. One approach is to help patients practice movements that they cannot complete without assistance, which may foster somatosensory stimulation that induces brain plasticity [81]. Indeed, most rehabilitation robots are strong enough to complete movements even when patients are completely impaired and/or when tone and spasticity act in opposition. However, care must be taken so that the robot does not "take over" the movement practice from the patient, which may cause the patient to "slack" and reduce their effort at the task being practiced [62, 82]. Patient effort is considered crucial to increasing motor-plasticity during rehabilitation therapy [83, 84]. Thus, it seems important for robotic rehabilitation devices to simultaneously enable movement practice and encourage patient effort during therapy.

Numerous control strategies for robotic therapy have been successfully implemented and tested, as summarized in [15]. Of specific interest are "assist-as-needed" control strategies, which change assistance in response to perceived effort, typically correlated in some way to performance error (tracking error or similar). These controllers alter the assistance level by modifying controller parameters (e.g. feedback gains, desired trajectory shape and/or timing, model based terms, etc.) [16, 62, 63, 74, 85, 86]. Tests with these controllers suggest that increased error encourages patient effort, and vice-versa, although the relationship remains unclear. Additional experiments may clarify this and other relationships affecting motor recovery during rehabilitation therapy, although the ultimate validation clearly depends on therapeutic efficacy.

Effectively exploring the factors that promote functional recovery during movement therapy and evaluating "assist-as-needed" and other control strategies depends on the control fidelity of the robotic platform. To quantify baseline motor ability, ideally, the robot should be able to appear both massless and frictionless to the patient, and should be highly compliant and backdriveable. However, it is also important to have a high bandwidth of force control, as to not limit the response of robot during interaction with the patient. Improving the control and impedance characteristics of a rehabilitation robot has the potential to make such devices better scientific instruments as well as allowing more precise investigation of motor learning and the mechanisms of neuroplasticity, as suggested by [15].

Another critical consideration for understanding the mechanisms by which rehabilitation robots promote recovery is the limb of application of the robot. As an integral part of activities of daily living (ADLs), rehabilitation of the hand is particularly important, and a significant need exists for improved hand rehabilitation, as most of those who have suffered a stroke experience some impairment in hand function [87]. Furthermore, the hand and fingers have a highly developed neuro-muscular system to which the brain has dedicated a large portion of resources.

Designing a robot to actuate the hand or finger is a significant challenge, as evidenced by the large variety of robotic devices that have been developed for hand and finger therapy. Previous work has focused often on re-creating the complexity of hand and finger movements, often at the expensive of actuation and control. These hand robots typically fall into the category of end-effector or exoskeleton (for review see [88].) End-effector devices attach distally and do not attempt to align with the joint axis of the patient, as exoskeleton devices typically do.

In the work presented here, we sought to maximize controller fidelity and minimize the mechanical impedance of the device, at the expense of the robot's degrees-of-freedom. Although each finger in the human hand has multiple degrees-of-freedom, most ADLs incorporate a simple finger curling motion, similar to a power grasp [89]. Thus, an opportunity existed to create a finger-curling robot with one degree-of-freedom, high control fidelity, and low friction.

FINGER, the finger curling robot presented here, is capable of individually assisting both the index and middle fingers through a natural grasping motion. Each finger is individually guided by an 8-bar mechanism that controls the orientation and position of the proximal phalanx and the position of the middle phalanx. Each 8-bar mechanism has a single degree-of-freedom and is actuated by a high bandwidth and low-friction linear electric actuator. Further friction reduction is achieved through feed-forward control compensation.

In the sections that follow, we present the design, controller development, and preliminary testing of FINGER. We present the mechanism synthesis, which is based on motion capture of finger grasping motion, first. We then describe the mechanical design, including sizing adjustments and patient-robot interface. In the third section, we describe the actuation including controller development and friction compensation. Finally, we present some results from pilot testing with several subjects who have suffered a stroke. Portions of this work have appeared previously in conference paper format [17, 90].

2.3. Methods

2.3.1. Mechanical Design

2.3.1.1. Finger Curling Data Acquisition and Analysis

This section describes motion capture and data analysis used to characterize a basic finger curling motion, similar to a power grasp [89]. Although the human hand can perform many differing grasps and grips in order to manipulate objects, the basic curling motion is the most common and therefore we focused on it for administering and studying finger movement therapy.

Color-based motion tracking [91] was used to record the path of the index finger as the hand performed a curling motion. Subjects' hands were filmed with a single camera from above, where the index and middle fingers curled in a plane perpendicular to the camera. Any out-of-plane motion during curling was small and treated as noise. The back of the hand was placed against a rest aligning it with the x-axis. Four brightly colored felt dots of differing colors were attached to the index finger, two each on the center-lines of the proximal and middle phalanges, which are the defined attachment points for the 8-bar mechanism. Likewise, the colored dots were attached using hook-and-loop straps that were the same thickness as the planned mechanism straps. The placement of the felt dots allowed the centerlines of the proximal and middle phalanges to be recorded throughout the curling motion. See Figure 2.1.

Seven healthy adult subjects were asked to curl their hand, meeting the thumb in a circle, for a minimum of 10 times. They were not given any further instructions regarding how to curl their hand, in order to produce the most natural motion. Dimensions of the index and middle fingers and hand were recorded for each subject using calipers. The lengths of the proximal

and middle phalanges for the index and middle fingers were recorded in a similar fashion as [92]. The distances between creases for both the index and middle fingers were also recorded in the same manner as [93].

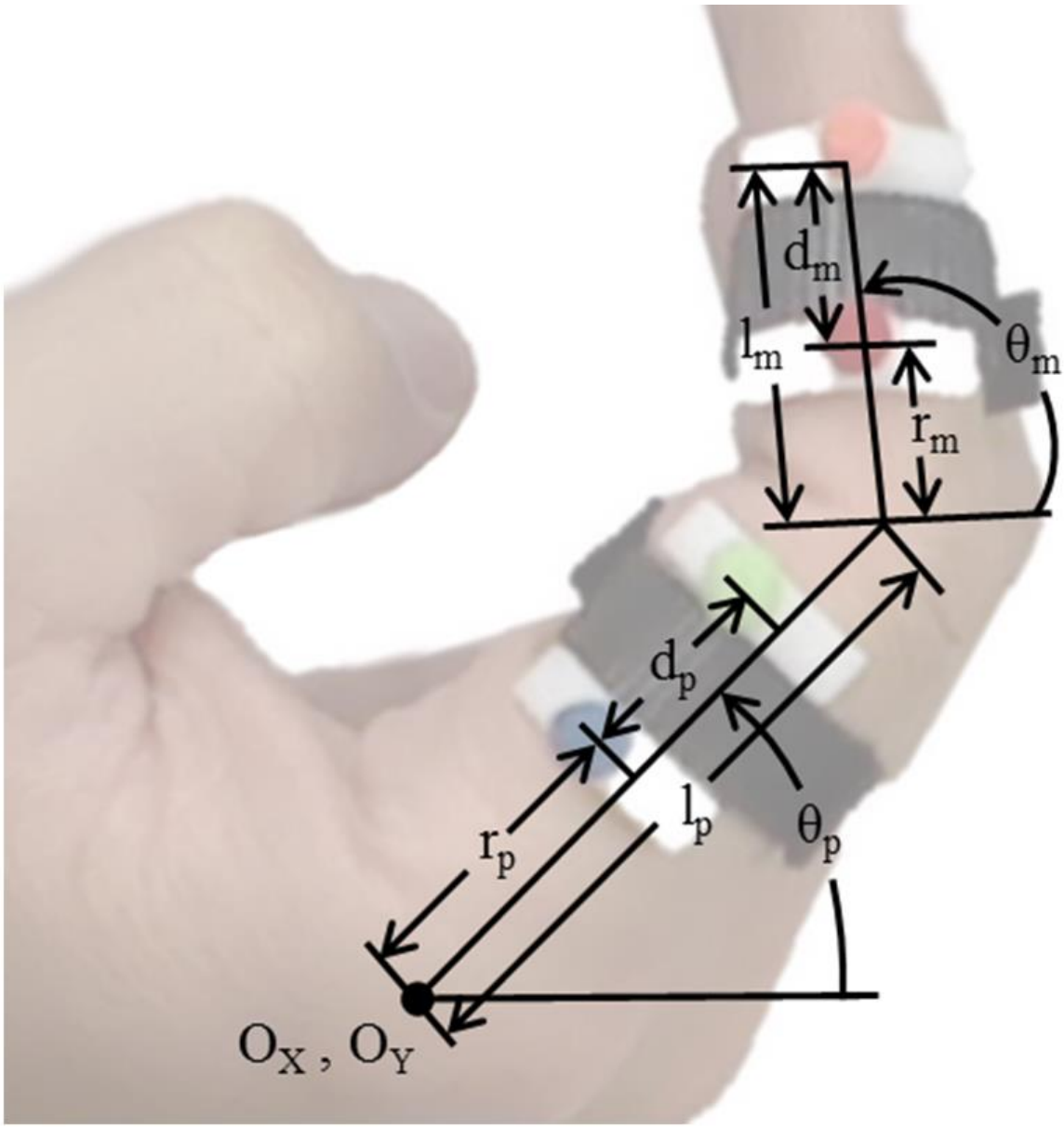

Figure 2.1. The setup and dimensions collected from motion capture and regression

The path of the four felt dots was regressed against a planar, two revolute joint model, shown in Figure 2.1. The regression model determines 5 dimensions using the system of

equations given below in (2.1). These equations are defined for each frame of the motion capture.

$$\begin{bmatrix} 1 & 0 & c_p & 0 & 0 \\ 0 & 1 & s_p & 0 & 0 \\ 1 & 0 & c_p & 0 & 0 \\ 0 & 1 & s_p & 0 & 0 \\ 1 & 0 & 0 & c_m & c_p \\ 0 & 1 & 0 & s_m & s_p \\ 1 & 0 & 0 & c_m & c_p \\ 0 & 1 & 0 & s_m & s_p \end{bmatrix} \begin{bmatrix} O_x \\ O_y \\ r_p \\ r_m \\ l_p \end{bmatrix} = \begin{bmatrix} (\mathbf{p}_1)_x \\ (\mathbf{p}_1)_y \\ (\mathbf{p}_2)_x - d_p c_p \\ (\mathbf{p}_2)_y - d_p s_p \\ (\mathbf{m}_1)_x \\ (\mathbf{m}_1)_y \\ (\mathbf{m}_2)_x - d_m c_m \\ (\mathbf{m}_2)_y - d_m s_m \end{bmatrix}, \tag{2.1}$$

where

$$c_p = \left((\mathbf{p}_2)_x - (\mathbf{p}_1)_x\right)/d_p \quad s_p = \left((\mathbf{p}_2)_y - (\mathbf{p}_1)_y\right)/d_p \quad c_m = \left((\mathbf{m}_2)_x - (\mathbf{m}_1)_x\right)/d_m$$
$$s_m = \left((\mathbf{m}_2)_y - (\mathbf{m}_1)_y\right)/d_m \qquad d_p = \|\mathbf{p}_2 - \mathbf{p}_1\| \qquad d_m = \|\mathbf{m}_2 - \mathbf{m}_1\|.$$

In (2.1) above, $\mathbf{m}_1$, $\mathbf{m}_2$, $\mathbf{p}_1$ and $\mathbf{p}_2$ are the positions of the four markers, and O_x and O_y define the location of the metacarpophalangeal (MCP) joint of the index finger. During mechanism design, this point is assumed to be the origin. The length of the proximal phalanx is denoted l_p and the final two parameters are the distances to the proximal and middle strap attachment from the previous joint, referred to as the proximal, r_p, and middle, r_m, radii. It may seem that these radii should center the straps along the proximal and middle phalanges, but in practice the position along the proximal phalanges that is most comfortable for a strap is significantly forward of the center of the phalanx. For example, in Figure 2.1 it can be seen that the hook and loop strap holding the felt dots to the proximal phalanx sits comfortably at more than half the distance along the proximal phalanx from the MCP joint. The same relationship is true for the middle phalanx.

The mean length of the index finger proximal phalanx determined by the motion capture and regression analysis was 42 mm, with a standard deviation of 3 mm. This mean value was compared to [92] which contains a statistical analysis of 4000 hand samples. The ratio of this mean proximal length to the mean proximal length reported in [93] was multiplied by the standard deviation also reported in [93], producing a scaled standard deviation 3 mm, which is close to the standard deviation of the small data sample used.

Following the approach in [94], the length change of the proximal phalanx between successive mechanism sizes was chosen to be two standard deviations. By scaling the other variables accordingly, the complete dimensions for the other finger sizes may be found, as given in Table 2.1. This range of finger sizes provides an acceptable coverage of the population of hand sizes.

Table 2.1. Dimensions determined for different hand sizes. The dimension l_p is the proximal length, r_p is the proximal radius, and r_m is the middle radius.

	l_p (mm)	r_p (mm)	r_m (mm)
Extra-Small	28.68	16.28	10.78
Small	35.13	20.32	13.21
Medium	41.58	24.57	15.63
Large	48.03	28.40	18.06

This approach to regression has the advantage of determining the dimensions of the finger independent of the motion type. Thus, the angular relationship between the phalanges can be used independently to define the finger motion. Specifically, regression is used to determine the middle phalanx angle, θ_m , as a function of the proximal phalanx angle, θ_p, using a second-order polynomial equation.

Figure 2.2 shows the relationship between the middle phalanx angle and the proximal phalanx angle for the curling motion collected from all 7 motion capture subjects. The black line is a quadratic curve-fit. With significant variance in hand sizes, the relationship between the two angles appears uncorrelated to hand size.

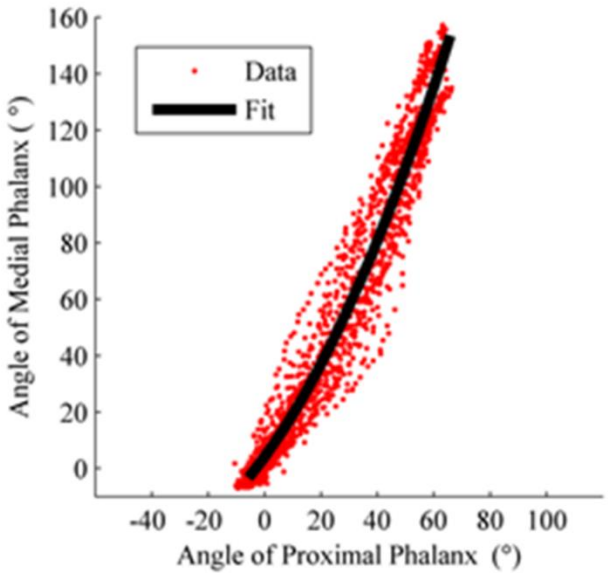

Figure 2.2. The angular relationship between the middle and proximal phalanges during finger curling. The red points were collected from motion capture, and the black line is the regressed quadratic curve-fit.

The 2nd order equation used to fit the data as shown in Figure 2.2 is:

$$\theta_m = 0.77660232\theta_p{}^2 + 1.37397306\theta_p + 0.07324267. \quad (2.2)$$

This equation and finger dimensions given in Table 2.1 were used to generate the 15 target points for each mechanism size, consisting of 15 desired points and angles of the proximal phalanx and 15 desired points of the middle phalanx, repeated for each of three sizes: extra-small, small, medium, and large. This number of points was chosen in order to maintain a good spatial resolution while keeping computational complexity of the design reasonable. The target points were created using the 2 revolute joint planar model with the angle of proximal phalanx, θ_p, varied from 0° to 60° discretized into 15 evenly spaced target angles.

The angle of the middle phalanx, θ_m, was determined from (2.2) and the target points for both the proximal and middle phalanges were defined at 19 mm behind the center-line of the finger to allow for a means of connecting the robot to the hand.

2.3.1.2. Linkage Selection

Designing a mechanism to reach multiple end-effector configurations, known as mechanism synthesis, is a well-studied research area [95]. This particular application, however, has a unique twist. In this case, there is not a single desired configuration but rather two that are correlated; one for the proximal phalanx (position and angle), and one for the middle phalanx (position only). Furthermore, the design specifies a planar grasping motion with a single DOF for each finger. Planar mechanisms, with their multiple varieties of single DOF configurations, provide an adequate solution base for this design problem.

Initial mechanism synthesis attempts explored multiple configurations of Watt type six-bar chains [96], but were ultimately unsuccessful in reproducing the desired output configuration. The final design uses an eight-bar mechanism (Chain 1 from [96]) with revolute joints (see Figure 2.3). The goal configurations consist of the position ($\mathbf{P}_G$) and angle (μ_P) of the proximal phalanx and the position of the middle phalanx ($\mathbf{M}_G$). The mechanism is made up of 10 revolute joints ($\mathbf{G}$, $\mathbf{G}_1$, $\mathbf{W}$, $\mathbf{W}_1$, $\mathbf{W}_2$, $\mathbf{H}$, $\mathbf{H}_2$, $\mathbf{Y}$, $\mathbf{Y}_1$, and $\mathbf{Y}_2$) and 7 links defined by the kinematic chains $\mathbf{GW}$, $\mathbf{WHW}_1$, $\mathbf{G}_1\mathbf{W}_1\mathbf{W}_2$, $\mathbf{HPYH}_2$, $\mathbf{W}_2\mathbf{H}_2\mathbf{Y}_2$, $\mathbf{Y}_1\mathbf{Y}_2$, and $\mathbf{YMY}_1$. These links are defined by seven structural angles (α, α_2, δ, δ_2, γ, γ_2, and μ) and 13 structural lengths ($d_{1\text{–}11}$, m, and m_2). Figure 2.3 also shows the seven configuration angles (θ, θ_1, ϕ, ϕ_1, ϕ_2, ψ, and ψ_1) that changes as the mechanism moves.

The mechanism has 1 DOF so that specifying one of these configuration angles specifies the complete configuration of the mechanism.

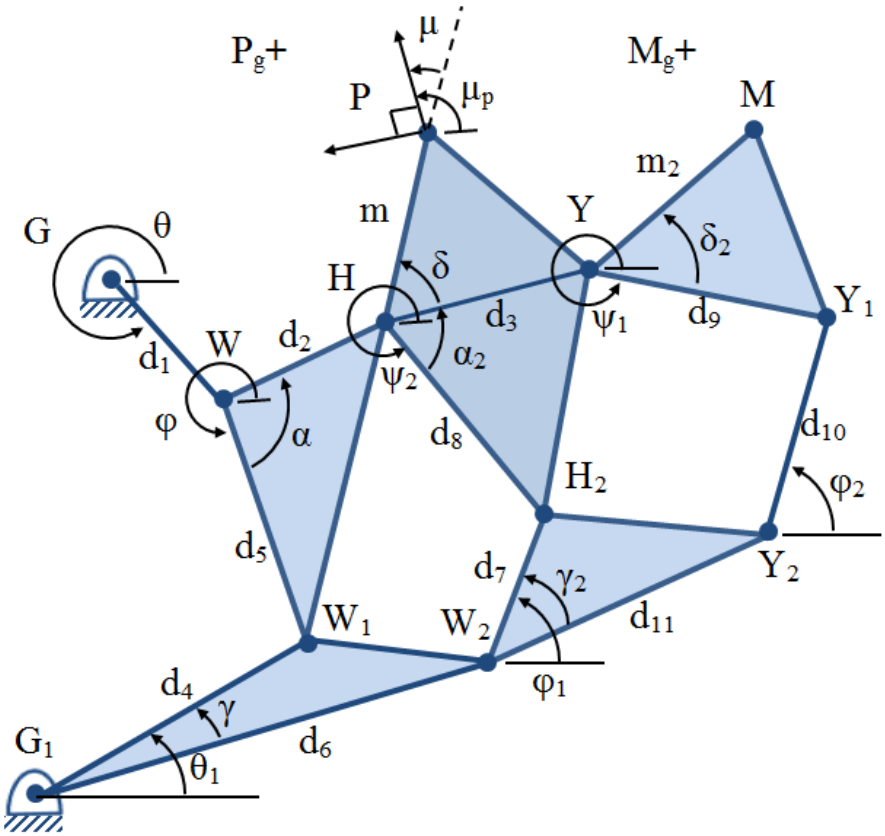

Figure 2.3. Structural dimensions and configuration angles of the 8-bar mechanism. Goal positions for the proximal and middle phalanges are shown as $\mathbf{P}_G$ and $\mathbf{M}_G$, respectively. The goal angle of the proximal phalanx is μ_p.

The preliminary optimization of this mechanism was presented in [90]. The approach here is similar, but here the mechanism configuration has changed so that the middle phalanx end-effector is connected to link $\mathbf{Y}_1\mathbf{Y}_2$ rather than $\mathbf{YY}_1$. This change improved the ability of the optimization process to reach desired middle phalanx target points, and also made the resulting mechanism easier to manufacture.

2.3.1.3. Mechanism design equations

The design equations that define the 8-bar mechanism consist of both path and loop equations. The path equations consist of three separate kinematic paths through the mechanism from one of the fixed pivots to the each of the end effectors ($\mathbf{P}$ and $\mathbf{M}$). The path equations are similar to those presented in [90] but have changed to improve optimization and manufacturability, based on a trial and error design process. For the proximal phalanx, the shortest path equation is:

$$\mathbf{G} + d_1 \begin{Bmatrix} c\theta \\ s\theta \end{Bmatrix} + d_2 \begin{Bmatrix} c(\alpha+\phi) \\ s(\alpha+\phi) \end{Bmatrix} + m \begin{Bmatrix} c(\psi+\alpha_2+\delta) \\ s(\psi+\alpha_2+\delta) \end{Bmatrix} = \mathbf{P}, \tag{2.3}$$

where c and s stand for cosine and sine, respectively, and the other parameters are previously defined and shown in Figure 2.3. The other two path equations for the proximal phalanx are defined by the kinematic chains $\mathbf{G_1W_2H_2HP}$ and $\mathbf{G_1W_2Y_2Y_1YHP}$. As previously mentioned, the middle phalanx path equations have changed more significantly. Using the same notation, the shortest path to the middle phalanx is

$$\mathbf{G}_1 + d_6 \begin{Bmatrix} c\theta_1 \\ s\theta_1 \end{Bmatrix} + d_{11} \begin{Bmatrix} c(\varphi_1-\gamma_2) \\ s(\varphi_1-\gamma_2) \end{Bmatrix} + m_2 \begin{Bmatrix} c(\varphi_2-\delta_2) \\ s(\varphi_2-\delta_2) \end{Bmatrix} = \mathbf{M}, \tag{2.4}$$

The other two kinematic paths to the middle phalanx are defined by the kinematic chains $\mathbf{G_1W_2Y_2M}$ and $\mathbf{GWHYY_1Y_2M}$. The design equations also include 3 internal loop constraint equations which must be satisfied to keep the design viable. The inner loop equation is

$$\mathbf{G} + d_1 \begin{Bmatrix} c\theta \\ s\theta \end{Bmatrix} + d_5 \begin{Bmatrix} c\varphi \\ s\varphi \end{Bmatrix} - d_4 \begin{Bmatrix} c\theta_1 \\ s\theta_1 \end{Bmatrix} = \mathbf{G}_1. \tag{2.5}$$

The other two equations, for the middle and outer loops are defined by the kinematic chains $\mathbf{GWHH_2W_2G_1}$ and $\mathbf{GWHYY_1Y_2W_2G_1}$.

As mentioned before, the design specifies a goal angle of the proximal phalanx, μ_P, in addition for each goal position of the proximal phalanx, $\mathbf{P}_g$. Using the relationship

$$\psi = \mu_P - \alpha_2 - \delta - \mu, \tag{2.6}$$

the goal angle for the proximal phalanx is substituted into the previously presented path and loop equations to constrain the configuration angle ψ to the goal angle of the proximal phalanx, μ_P, and the structural angles α_2, δ, and μ.

With the path and loop equations defined, the design problem becomes a function minimization problem. The objective is to find the structural variables and the set of $n = 15$ configuration angles that best reach the 15 desired configurations. To achieve this, a cost function is created consisting of the sum of the squares of the distance between the desired end-effector points $\mathbf{P}$ and $\mathbf{M}$ and the goal positions $\mathbf{P}_G$ and $\mathbf{M}_G$ for each of the 15 desired configurations. The cost function is defined

$$J = \sum_{n=1}^{15} \left(\left(\mathbf{P}_n - \mathbf{P}_{\mathbf{G},n} \right)^T \left(\mathbf{P}_n - \mathbf{P}_{\mathbf{G},n} \right) + \left(\mathbf{M}_n - \mathbf{M}_{\mathbf{G},n} \right)^T \left(\mathbf{M}_n - \mathbf{M}_{\mathbf{G},n} \right) \right), \tag{2.7}$$

where $\mathbf{P}_n$ is the position of point $\mathbf{P}$ at angle θ_n and $\mathbf{P}_{\mathbf{G},n}$ is the n^{th} desired configuration (with similar definitions for $\mathbf{M}$ and $\mathbf{M}_G$). Only one configuration angle is necessary as the mechanism has only 1 DOF, and θ was arbitrarily chosen (the other possibility was θ_1), even though in the final design we selected θ_1 as the input angle for connecting the actuator (based on the locations of $\mathbf{G}$ and $\mathbf{G}_1$).

2.3.1.4. Mechanism design equation constraints

In addition to the cost function, a large set of constraints are required based on the overall design goals and manufacturing considerations. Our preliminary approach to constraints was presented in [90]. Some constraints require only upper and lower bounds, such as the location of the base points, $\mathbf{G}$ and $\mathbf{G}_1$, and structure variables $d_{1\text{–}11}$, m, and m_2. Others constraints require additional calculations, such as the link dimensions not specifically specified by the structural dimensions (the distance $\overline{\mathbf{HW}}_1$, for example). These constraints are summarized, with brief explanations, in Table 2.2.

Table 2.2. Mechanism Structural Design Constraints

Dimension(s)	Bounds (mm)	Purpose
$\mathbf{G}_x$, $\mathbf{G}_{1,x}$	$\{-76.2,\ 25.4\}$	Keep fixed pivots located behind wrist/hand.
$\mathbf{G}_y$, $\mathbf{G}_{1,y}$	$\{-10.2,\ 2.54\}$	Keep fixed pivots located behind wrist/hand.
$d_{1\text{–}7}$	$\{1.91,\ 12.7\}$	Min. distance to manufacture joints, keep mechanism compact.
$d_{8\text{–}11}$	$\{1.91,\ 7.62\}$	Min. distance to manufacture joints, keep mechanism compact.
m	$\{1.27,\ 12.7\}$	Min. distance to manufacture proximal phalanx end-effector, keep mechanism compact.
m_2	$\{1.91,\ 12.7\}$	Min. distance to manufacture middle phalanx end-effector (including room for rotating joint), keep mechanism compact.
$\overline{\mathbf{GG}_1}$, $\overline{\mathbf{PY}}$, $\overline{\mathbf{HW}}_1$, $\overline{\mathbf{W}_1\mathbf{W}_2}$, $\overline{\mathbf{H}_2\mathbf{Y}}$, $\overline{\mathbf{H}_2\mathbf{Y}_2}$, and $\overline{\mathbf{MY}_1}$	$\{1.91,\ 7.62\}$	Min. distance to manufacture joints, keep mechanism compact.

These constraints govern the structural dimensions of the 8-bar mechanism, but do not limit the location of the free joints ($\mathbf{W}$, $\mathbf{W}_1$, $\mathbf{W}_2$, $\mathbf{H}$, $\mathbf{H}_2$, $\mathbf{Y}$, $\mathbf{Y}_1$, and $\mathbf{Y}_2$) as the mechanism moves through the desired configurations. For example, any two joints should not overlap

during the motion of the mechanism. This requires that the location of each joint be calculated at each of the 15 goal configurations. In total there are 29 joint pairs with the potential to overlap which are constrained to keep the distance between joints from going below a manufacturable distance. The complete list of joint-to-joint distances to calculate at each of the 15 goal configurations is given in Table 2.3.

Table 2.3. 8-bar Mechanism Joint Distance Constraints

Joint-to-joint distance(s), calculated at each of the 15 goal configurations	Bounds (mm)	Purpose
$\overline{\mathbf{WG}_1}$, $\overline{\mathbf{W}_1\mathbf{G}}$, $\overline{\mathbf{W}_2\mathbf{G}}$, $\overline{\mathbf{W}_2\mathbf{W}}$, $\overline{\mathbf{W}_2\mathbf{H}}$, $\overline{\mathbf{HG}}$, $\overline{\mathbf{HG}_1}$, $\overline{\mathbf{H}_2\mathbf{G}}$, $\overline{\mathbf{H}_2\mathbf{G}_1}$, $\overline{\mathbf{H}_2\mathbf{W}}$, $\overline{\mathbf{H}_2\mathbf{W}_1}$, $\overline{\mathbf{YG}}$, $\overline{\mathbf{YG}_1}$, $\overline{\mathbf{YW}}$, $\overline{\mathbf{YW}_1}$, $\overline{\mathbf{YW}_2}$, $\overline{\mathbf{Y}_1\mathbf{G}}$, $\overline{\mathbf{Y}_1\mathbf{G}_1}$, $\overline{\mathbf{Y}_1\mathbf{W}}$, $\overline{\mathbf{Y}_1\mathbf{W}_1}$, $\overline{\mathbf{Y}_1\mathbf{W}_2}$, $\overline{\mathbf{Y}_1\mathbf{H}}$, $\overline{\mathbf{Y}_1\mathbf{H}_2}$, $\overline{\mathbf{Y}_2\mathbf{G}}$, $\overline{\mathbf{Y}_2\mathbf{G}_1}$, $\overline{\mathbf{Y}_2\mathbf{W}}$, $\overline{\mathbf{Y}_2\mathbf{W}_1}$, $\overline{\mathbf{Y}_2\mathbf{H}}$, $\overline{\mathbf{Y}_2\mathbf{Y}}$	$\{19.1,\ 254\}$	Keep joints from colliding during motion, and make joints manufacturable.

One of the most important set constraints concerns the location of the joints with respect to the hand and fingers throughout the motion. One of the main goals for the mechanism was that it be entirely located behind the hand during operation. This goal was chosen to allow the hand to be easily attached to the robot, to allow stacking of mechanisms for individual fingers, and to facilitate providing sensory stimulation to the volar surface of the hand. For instance, a soft object can be mounted towards the palm of users' hands to be touched while practicing grasp motions. In the presented experiment, however, this feature is not utilized. This requires all of the joints to be behind the hand and fingers at each of the 15 goal configurations. Thus the constraint area is constantly changing. We implemented this constraint area by creating three separate unit vectors, one at the back of the middle phalanx, $\mathbf{u}_m$, one at the back of the proximal phalanx, $\mathbf{u}_p$, and one at the back of the wrist, $\mathbf{u}_w$. These

vectors are illustrated in Figure 2.4. These unit vectors at the phalanges point away from the hand at an angle perpendicular to the phalanx and are different for each of the 15 goal configurations. The unit vector at the back of the wrist also points away from the hand, but does not move for different goal configurations. Furthermore, three vectors are created for each of the free joints at each of the 15 goal configurations. These three vectors, $\mathbf{v}_m$, $\mathbf{v}_p$, and $\mathbf{v}_w$, all point to the aforementioned joint and originate from the base of the unit vectors $\mathbf{u}_m$, $\mathbf{u}_p$, $\mathbf{u}_w$ (see Figure 2.4). The maximum of the dot products between the corresponding $\mathbf{u}$ and $\mathbf{v}$ vectors gives the distance, as a positive value, that the joint is from the back of the hand/finger. This value is constrained to between 12.7 and 76.2 mm, so that the joints clear the back of the hand during motion but are also kept from begin excessively away from the hand.

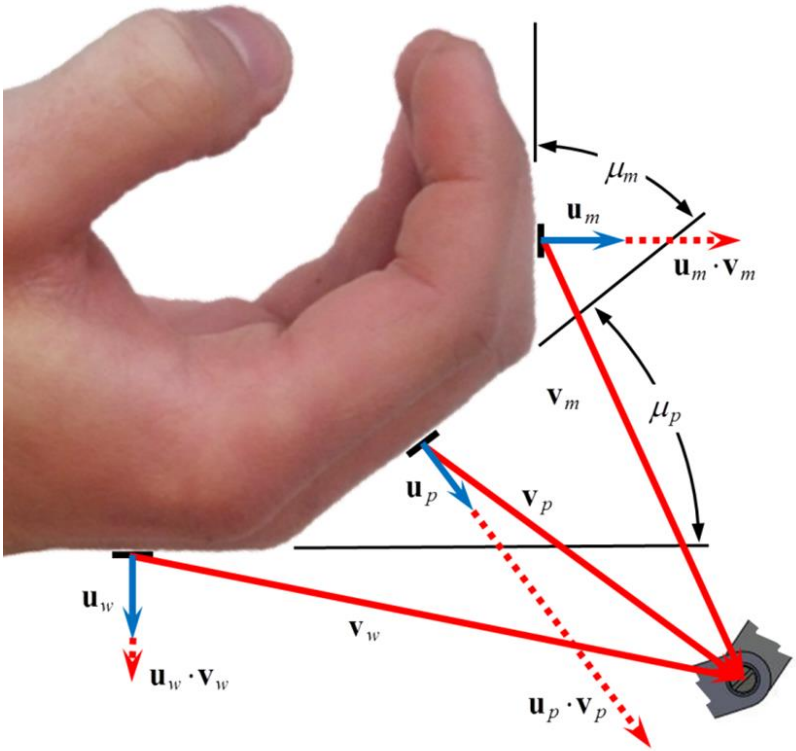

Figure 2.4. Vectors for constraining the location of the free joints to the back of the hand and finger

2.3.1.5. Mechanism cost function minimization

The cost function (2.7) was minimized within the bounds of the constraints detailed in the previous section using a constrained multivariable minimization optimizer (Matlab function "fmincon"). The minimization process was repeated numerous times with randomized initial conditions until viable solutions were found. The minimization was first performed on the goal configurations generated from the dimensions of the medium size given in Table 2.1. The resulting structural parameters of the medium mechanism are given in Table 2.4.

Table 2.4. Structural parameters for the medium finger curling mechanism

Length (mm)					Angle (°)		
$\mathbf{G}_x$	-40.93	d_6	+70.30		α	=	-135.2
$\mathbf{G}_y$	-28.68	d_7	+92.68		α_2	=	+59.18
$\mathbf{G}_{1x}$	-59.64	d_8	+76.20		δ	=	+9.520
$\mathbf{G}_{1y}$	-25.40	d_9	+69.80		δ_2	=	-24.25
d_1	+36.28	d_{10}	+76.20		γ	=	-39.29
d_2	+49.12	d_{11}	+55.30		γ_2	=	2.057
d_3	+19.05	m	+47.24		μ	=	-19.67
d_4	+34.43	m_2	+101.20				
d_5	+20.18						

The total error after cost function minimization was 2.14 mm for all 30 of the target points for the 15 medium finger size goal configurations. This is an average error of 0.071 mm per target point.

After the structural parameters for the medium finger size were determined, the cost function was re-minimized for the large, small, and extra small finger sizes using the medium structural parameters as the initial conditions. During re-minimization, all of the structural parameters were fixed except for five (m, m_2, δ, δ_2, & μ) which allow the

new mechanism it to reach the goal configurations of the other finger sizes. Although other choices of parameters also allow the minimization to reach the trajectories for the other finger sizes, changing these five parameters can be accomplished by only modifying the shape the two end-effector links. This allows the re-sizing to be accomplished without the need to disassemble the mechanism when changing between sizes. The resulting values of these structural parameters and the minimized cost function for the other mechanism sizes are given in Table 2.5. A visual depiction of the ability of the four different mechanism sizes to reach the four sets of 15 desired configurations is shown in Figure 2.5.

Table 2.5. Values of the changing structural parameters for different mechanism sizes and the resulting cost function

Parameter	Extra-Small	Small	Medium	Large
m (mm)	+41.5	+44.2	+47.2	+50.1
m_2 (mm)	+94.5	+67.6	+101.2	+105.7
δ (°)	+145.7	+120.6	+95.2	+74.2
δ_2 (°)	-32.79	-28.46	-24.25	-20.45
μ (°)	-19.93	-19.99	-19.67	-19.13
Cost Function, J (mm)	7.58	3.42	2.14	5.01
Per 30 Points, J / 30 (mm)	0.25	0.11	0.07	0.17

2.3.1.6. Mechanical design

The current version of FINGER has two identical planar 8-bar mechanisms to individually curl the index and middle fingers through a naturalistic motion. The mechanism, actuators, and adjustment assemblies are located behind the hand. As mentioned above, this allows contact of the volar surface of the hand with objects during therapy, and makes it easier to attach the hand of a subject to the robot.

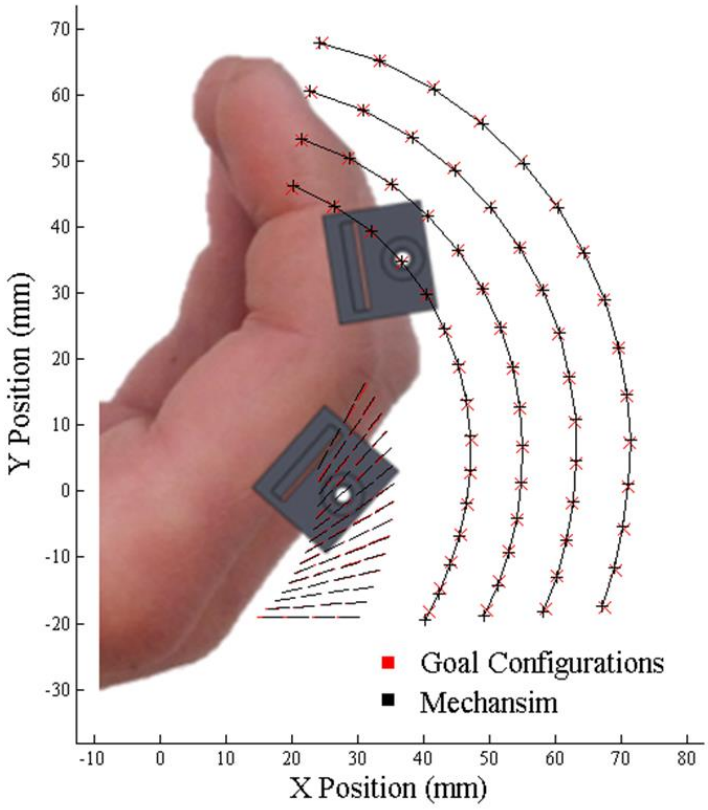

Figure 2.5. Optimized mechanism paths four different finger sizes. There are 15 goal configurations for each size, including target points (red exes) for the middle phalanx (for controlling position) and target lines (read lines) for the proximal phalanx (for controlling position and angle). The ability of the mechanism to reach these 15 configurations is demonstrated with black crosses and black lines.

Each 8-bar mechanism was designed with alternating inner links and outer link pairs, overlapping at joints to balance bearing forces and keep friction low. Two ABEC 5 bearings and one precision shoulder bolt were used for each joint. The links were designed in Solidworks to the dimensions determined from the mechanism synthesis and machined from aluminum using a three-axis, computer numerical control (CNC) milling machine. The linkage design includes mechanical hard stops to limit the range of motion to the desired range.

Finger cups with custom ratcheting straps (see Figure 2.6) are located at the two end effectors of each mechanism to attach the robot to the subjects' proximal and middle phalanges. The middle phalanx finger-cup allows for rotation while the proximal finger cup

is fixed, as per desired kinematic design. Each of the 8-bar mechanisms includes adjustability for different finger lengths. After inspecting the results of the mechanism synthesis, it was apparent that the locations of the end effector **M** as defined for the four hand sizes (Table 2.5) are very nearly located on a line with respect to link **MY**1**Y**2. The same is true for the proximal phalange end effector on **PHY**. This fact simplified the mechanical design, allowing for infinite positioning of the finger cups over the full adjustment range. The middle phalanx length adjustment is shown in Figure 2.6.

Figure 2.6. Finger cups with ratcheting straps for the middle phalanx (top left) and the proximal phalanx (top right) and finger length adjustment (bottom)

The location of each of the 8-bar mechanisms may be adjusted vertically to align them with the plane of the subjects' index and middle fingers (see Figure 2.7). Furthermore, the wrist of the subjects is secured in a wrist cuff, of which the height and angle may be adjusted as necessary for alignment and comfort during gameplay (see Figure 2.7).

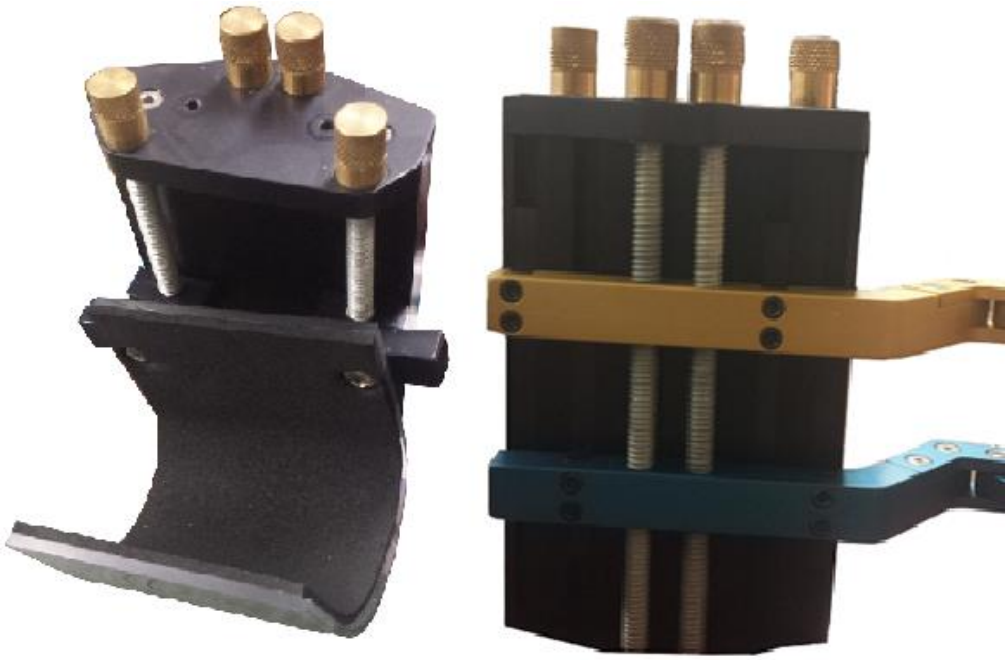

Figure 2.7. Wrist cuff with height adjustment assembly (left) and mechanism height adjustment (right)

Each 8-bar mechanism is independently actuated. The two linear actuators are mounted on top of each other with a fixed vertical distance from the base plate while they can freely rotate about an axis normal to it. The detailed specifications of the actuators are explained in the sections that follow. The entire assembly is shown in Figure 2.8.

2.3.2. Robotic actuation and performance

2.3.2.1. Actuation Hardware

FINGER uses two brushless linear motors ("Servo Tube" actuators, Dunkermotoren STA116-168-S-S03C) to independently actuate the 8-bar finger curling mechanisms. These actuators were chosen for their unique combination of high speed, low friction, and large stroke length. Because they lack any gearing or cables, they exhibit good backdrivability. This is an important feature for robot assisted therapy; the ideal rehabilitation actuator would be able to apply any force at any point during the desired motion, including zero-force, allowing the subject to see the results of their efforts.

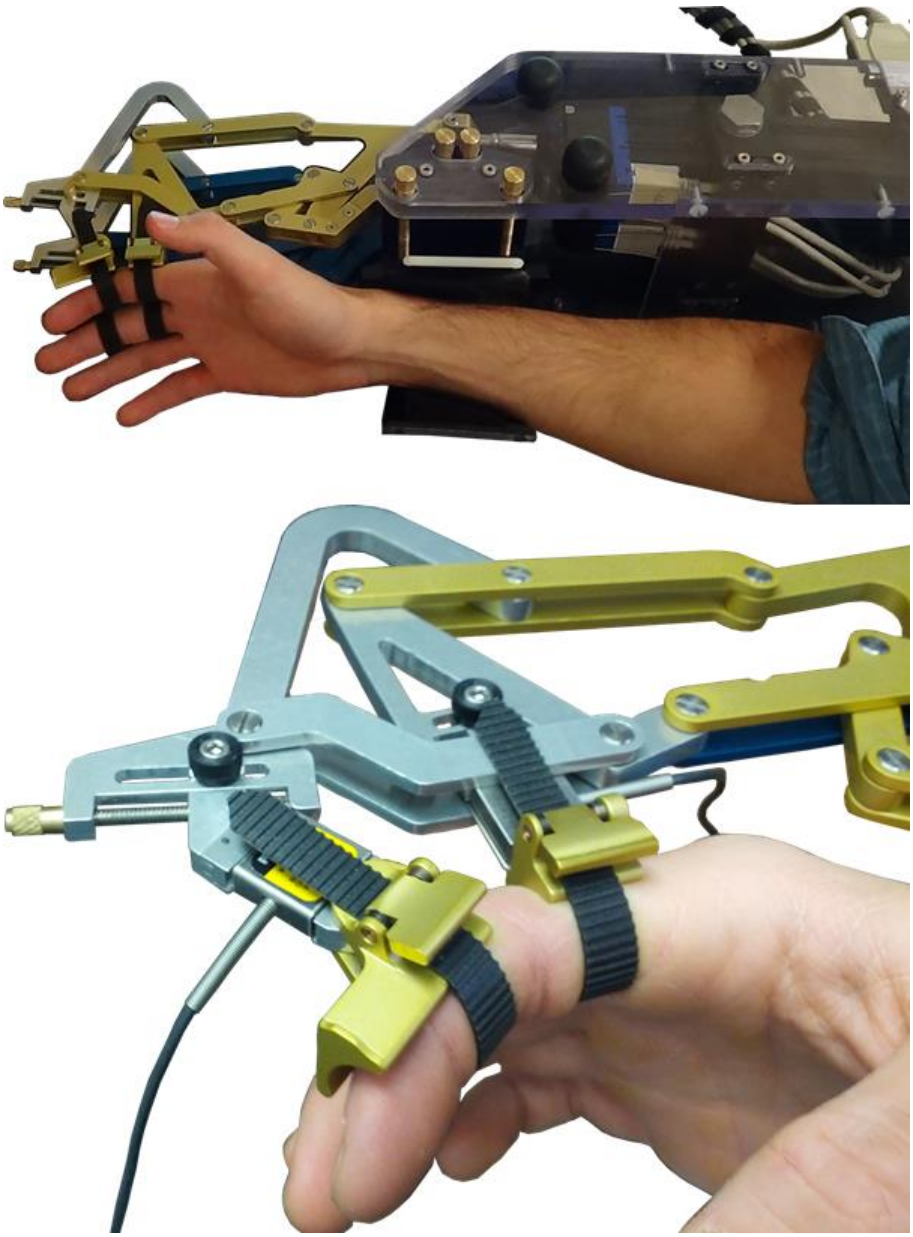

Figure 2.8. FINGER assembly. FINGER robot with two 8-bar finger curling mechanisms and two actuators (top), and close-up of index and middle fingers attached to the robot (bottom). The proximal phalanx finger-cup is fixed at an angle but the middle phalanx is free to rotate.

This particular model of Servo Tube actuator can produce a continuous force of 26.75N with a peak of 91.9N. Current to the actuator is controlled by an amplifier (Copely Controls ACJ-055-09-S), which allows a voltage or PWM setpoint signal. The Servo Tube actuator has built-in Hall Effect sensors and outputs an emulated quadrature encoder position signal of up

to 8 microns of resolution. Accelerometers (Analog Devices ADXL325EB) mounted to the end of the actuator rod measures actuator accelerations with a range of ±6 g.

The controller is implemented on a PC using Matlab® xPC Target, with a sampling frequency of 1000 Hz. A National Instruments 6221 DAQ card (16-Bits, 250kS/s) is used to acquire voltage signals from the accelerometers, read the quadrature outputs from the Servo Tube actuators, and send the forces commands to the actuators.

2.3.2.2. Control fidelity

To evaluate the control fidelity of FINGER, we conducted a closed loop frequency response test. A Proportional-Integral-Derivative (PID) controller was used to follow desired sinusoidal trajectories with a magnitude of 75% of the range of motion and frequencies from 0.15 to 100 Hz. The PID controller gains, chosen by trial and error, for this test were KP= 8 N/m, KI = 8 N/m·s and KD = 2 N·s/m, respectively. The results, shown in Figure 2.9, show a -3 db magnitude reduction at approximately 8 Hz. The corresponding jump in phase lag indicates the nonlinearity in the system at high speeds.

2.3.2.3. Velocity estimation

Although the built-in position sensor of the Servo Tube actuator has a very high resolution, using a discreet derivative of the position signal can be very noisy, especially at low velocities. In order to obtain a smooth velocity estimate, a Kalman-Filter was designed that uses the actuator's position signal and an acceleration signal from an accelerometer mounted at the end of the actuator rod. The Kalman-Filter gains were calculated using the Matlab LQR function (Linear Quadratic Regulator). The Kalman-Filter design is similar to the one used in [97].

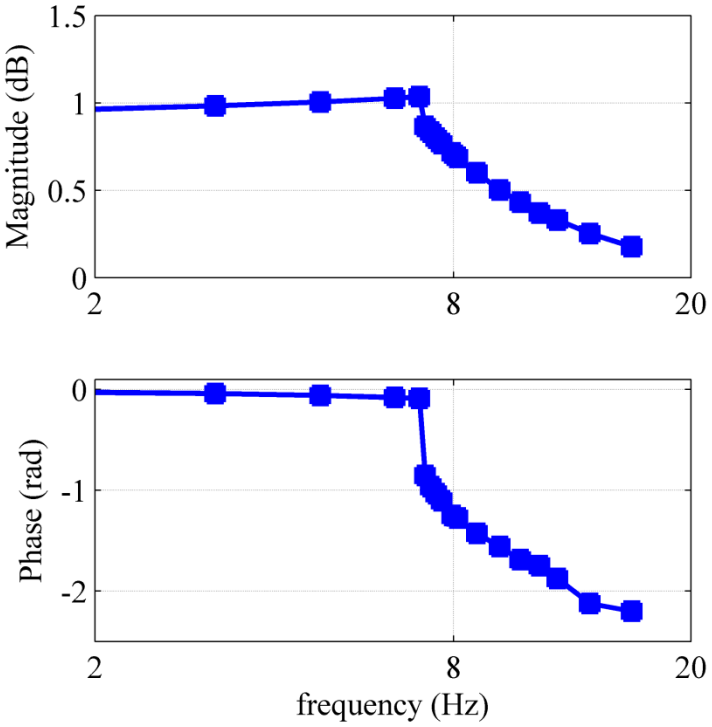

Figure 2.9. Bode plots of the robot under PID control.

2.3.2.4. Friction compensation

Minimizing friction was a top priority during the design and manufacturing of FINGER. This goal guided the mechanism design, manufacturing process, and the selection of bearings and actuators. Figure 2.10 shows the static friction force for one of the 8-bar mechanisms as a function of actuator stroke. These static friction forces were determined experimentally as the force required by the actuator to move the mechanism from a rest position. Because the position dependency in the static friction is minimal, the average static friction force (0.0137±0.0015N SD) was used to construct a feedforward friction compensator. Assuming a simple Coulomb friction model, the compensator adds this average friction force along the direction of the estimated velocity. To prevent chattering, the compensator only applied the static force after a minimum velocity magnitude is achieved (see the dotted box in Figure 2.11).

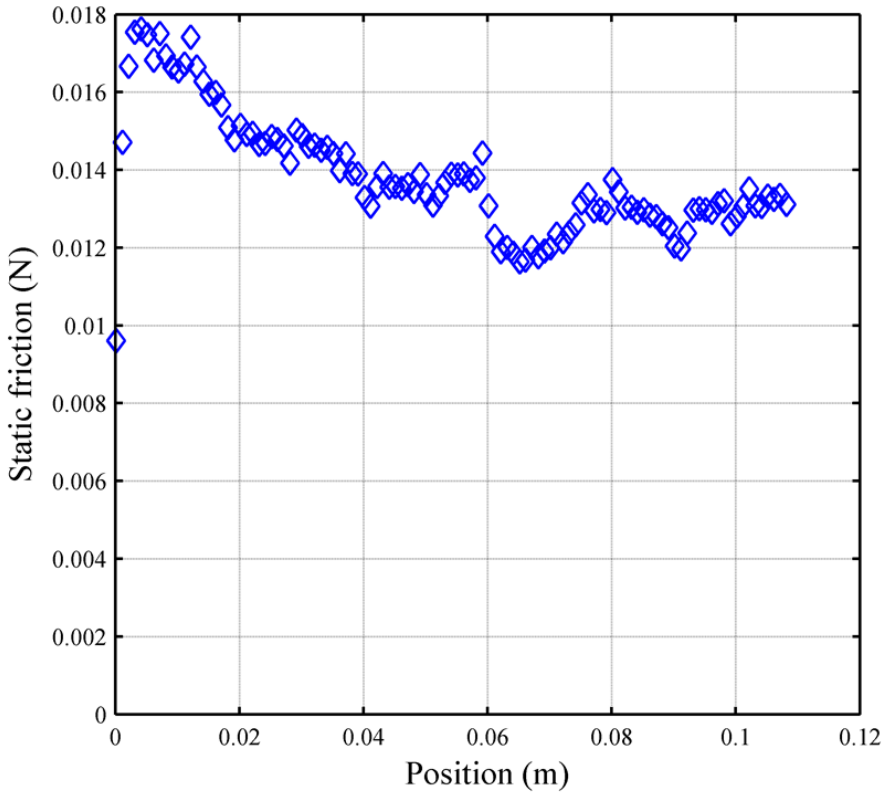

Figure 2.10. Static friction of the 8-bar finger curling mechanism

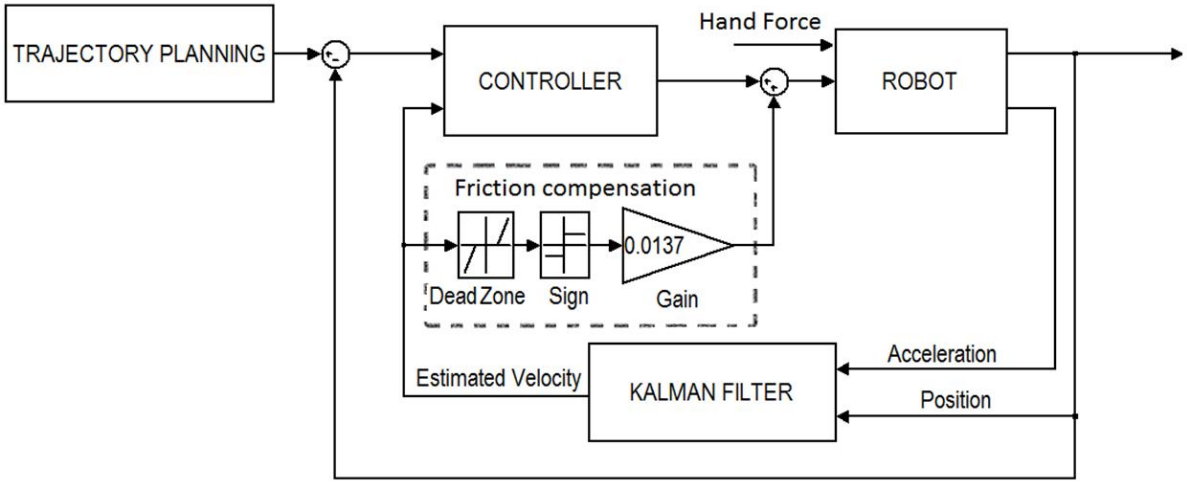

Figure 2.11. Block diagram of the control system

2.3.2.5. Control architecture

Figure 2.11 shows the block diagram of the robot control system. The trajectory planning block includes the computer game with predefined desired trajectories, sent to the robot

controller. Based on therapeutic preferences, different games can be used or designed as an interface between the subjects and the robot. For each game, the subjects are instructed to move the robot that is attached to their fingers according to the tasks dictated by the game. The robot moves by the combination of subject and actuator forces (Figure 2.11). The actuator force is a function of the controller type; hence, the controller structure determines how the subject will be assisted by the robot. Various controller types with different characteristics have been used in assistive devices to fulfil different therapeutic hypotheses [15]. The controller used for the testing described herein is a linear Proportional-Derivative (PD) controller, whose gains vary during the gameplay according to an algorithm that will be described in the following sections.

2.3.3. Pilot testing with individuals with stroke

2.3.3.1. Human subjects

Eleven male and five female volunteers with stroke related motor impairment on the right side participated in the study. The average age of the subjects was 57.8 +/- 12.5 SD and they were 3.3+/-1.8 SD years post stroke. Eight subjects reported that their stroke was ischemic; three reported that their stroke was haemorrhagic; and five did not know. Level of impairment was assessed using both the upper extremity Fugl-Meyer (FM) test and the box and blocks (BB) test [98, 99]. For the FM test, a trained therapist asked subjects to perform 33 test movements and scored them 0 (can't do), 1 (can do partially), or 2 (can do), then summed the scores. For the BB test, subjects moved as many blocks as possible over a divider in a one minute period. The average FM scores for the group were found to be 41.6±15.8 SD out of 66, and average BB scores were found to be 25.1±21.9 (compared to a score of 75.2±11.9 reported in literature for healthy subjects) [98]. Based on these scores,

nine of the subjects were classified as highly impaired (FM < 40 & BB< 20), and the remaining seven subjects were classified as moderately impaired. For comparison, four healthy subjects (3 male/1 female, average age 33.5 ± 9.4 SD) were also included in the study. All subjects provided informed consent, and all procedures were approved by the institutional review board at U.C. Irvine.

2.3.4. Therapeutic game play

To demonstrate its potential as both a rehabilitative tool and a platform for exploring the factors that promote functional recovery, FINGER was used to test the hypothesis that subjects will be most engaged in the rehabilitation therapy presented to them when they are at their optimal challenge level. To test this hypothesis, FINGER was used to assist subjects in playing a custom-designed game similar to Guitar Hero ®, which is the third largest video game franchise in history. Prior to gameplay, the subjects were asked to put their hand in FINGER, and the proximal and middle phalange attachment points were adjusted to finger size until the subjects were able to comfortably curl their fingers through the full range of motion. Additional support under elbow was provided as needed to put the subject's arm and wrist in a comfortable orientation. This game requires subjects to play along with a song by attempting to hit notes streaming down a visual display as shown in Figure 2.12. In order to hit these notes, the subjects were required to flex their fingers to a desired angle and stop at the correct time while receiving performance-based assistance from the robot. During the game, subjects were presented with three types of notes corresponding to flexion of the index finger, the middle finger and both fingers together. After successfully hitting a note, the subjects were required to extend their fingers back to a neutral position before the game would credit them with hitting future notes. During extension to the neutral position,

subjects received the same amount of assistance as they received during flexion. While subjects attempted to flex their fingers to the correct positions, small dark balls hovering above the fretboard were displayed to provide the subjects with visual feedback of their finger position (see Figure 2.12). The song used in this experiment was “Happy Together” by the Turtles, and it required 104 notes to be hit over the course of a 160 second game. Timing of the notes was the same during all the experiments and only the level of assistance changed to modulate subjects’ success rates as will be discussed later. Portions of this experiment have been published in conference paper format [17].

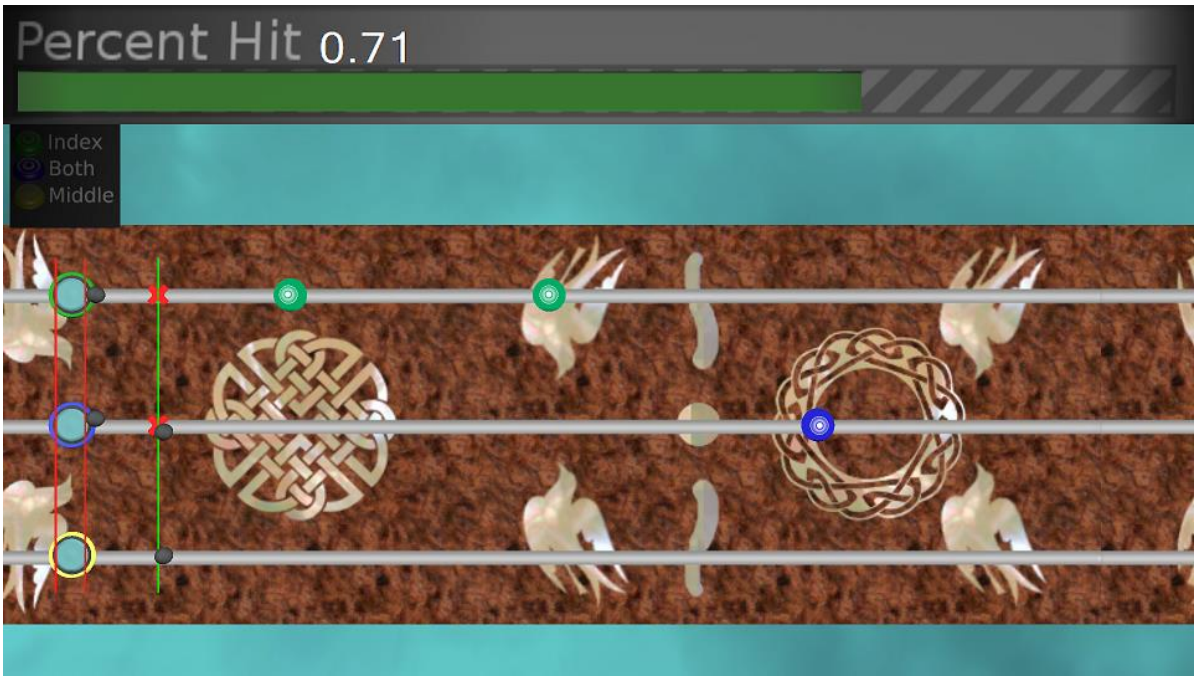

Figure 2.12. Screen-shot of the game, which is similar to Guitar Hero®. The green target was controlled by the index finger, the yellow target by the middle finger, and the blue target by both fingers together. The other two targets were not used. As subjects move a finger, a corresponding dark ball moves along the fret board. The desired locations of the fingers are displayed by fixed circles on the fret board that stream towards the bottom of the screen. To hit a note, the subjects were required to move the ball to the fixed desired position at the time the streaming note passes through it. After hitting a note, the subjects must return the finger(s) to neutral position in order to hit the upcoming notes.

2.3.4.1. Success rate algorithm

During the game, FINGER was used to both assist the subjects in completing the desired task and to monitor their performance. Although FINGER can be operated under a variety of

control paradigms, this experiment used a PD controller whose gains were intermittently updated by an algorithm which attempts to control the subjects' probability of hitting notes successfully [100]. Our contention is that by controlling subjects' success rate, we will be able to control their challenge level. According to the challenge point framework (CPF), determining the optimal challenge level is crucial to optimality of motor learning, particularly in rehabilitation [101]. CPF states there is an ideal amount of information which when presented to the learner will optimize the learning process. In other words, to achieve the best learning rate, the task shouldn't be too easy or too difficult. This ideal amount of information varies with the skill level of the learner. By controlling the controller gains, we can control the game difficulty and hence the level of challenge the subjects experience, regardless of their impairment level.

Determining the optimal challenge point for a particular task is difficult because it requires measuring long-term learning at a variety of challenge levels in a large number of subjects. However, one determinant of the optimal challenge point is likely effort – i.e. the more engaged a subject is, the more learning will likely occur. Effort can be measured in real-time and thus has the potential to serve as a means to identify when conditions are at least partially conducive for learning. Thus, we studied how effort, quantified by how much force the subjects exerted during the game (see below), varied with success rate.

The success rate algorithm mentioned above works as follows: For each successful note, the algorithm reduced the gains on the corresponding finger by an amount ρ, and for every missed note the gains on the corresponding finger were increased by an amount $\alpha \cdot \rho$. As shown in [100], this simple algorithm eventually forces the subjects' probability of success to converge on a value dependent only on α as shown in Equation (2.8) below.

$$\overline{P}_{i\to\infty} = \frac{\alpha}{\alpha+1} \tag{2.8}$$

2.3.4.2. Experimental protocol

Subjects were seated in front of a visual display, and the proximal and middle phalanges of their index and middles fingers were securely attached to the end effectors of the FINGER robot. Subjects were then instructed how to play the game and were asked to familiarize themselves with the task by playing through a song at a success rate of 75%. Data from this initial trial were excluded from the final analysis.

After the familiarization task, the robot was used to measure the subjects' range of motion and maximum isometric force in both flexion and extension. Measurements were taken from the index and middle fingers both individually and together. These measurements were repeated at the end of the experiment. Then subjects were asked to play through the same song twice at each of the three randomly presented success rates (50%, 75%, and 99%).

On a randomly selected subset consisting of roughly 15% of the notes in every song, the robot's gains were set to a fixed value and the robot was used to block the subject's movements instead of assisting them. During these blocked trials, the amount of force exerted against the robot was taken as a measure of the subject's effort in the task. Subject performance during these trials was not used to adapt the robot's gains, and once the blocked notes passed the control gains were returned to their previous values.

2.3.4.3. Data analysis

The instantaneous success rate at each note was calculated by dividing the number of successful trials within a moving window containing the 25 preceding notes by the size of the window. The peak force applied against the robot during blocked trials was used to quantify subject effort by normalizing it to the subject's maximum force for the

corresponding finger as measured during isometric trials. An unbalanced 2 factor mixed measures ANOVA with repeated measures applied to the success rate variable was used to test the effects of success rate and impairment level on subjects' effort.

During blocked notes for the index and middle fingers, the robot restricted the motion of both the correct and the incorrect fingers. An estimate of finger individuation was thus obtained by comparing the force generated by the finger that was supposed to move to the force generated by the finger that was not. Forces measured from both fingers were first normalized by their corresponding maximum force values from isometric trials. A measure of individuation was then calculated by dividing the average maximum normalized force applied by the incorrect finger by that of the correct finger. For blocked notes in which the force applied by the incorrect finger was greater than 1.25 times the force applied by the correct finger, it was assumed that the subject accidentally tried to hit the wrong note. Similarly, for trials in which the subjects did not apply any measurable force with either finger, it was assumed the notes were completely missed. These blocked notes were not included in the individuation analysis. An unbalanced three factor mixed measures ANOVA with repeated measures on the finger variable and the success rate variable was used to determine whether finger, success rate, or impairment level had any significant effect on the subject's individuation value.

2.4. Results

Average probability of success in hitting correct notes during gameplay versus time for the sixteen impaired and the four healthy subjects is shown in Figure 2.13. At the desired success rates of 50%, 75% and 99% the impaired subjects converged to the average actual success rates of 47.7+/-9.6%, 73.8+/-7.1%, and 97.6+/-1.9%. However, the unimpaired

subjects converged to the average actual success rates of 72.2+/-19.5%, 79.3+/-4%, and 99+/-1.1%. This result shows that the algorithm explained in 2.3.4.1 is successful in assisting subjects to achieve a desired success rate. It is not surprising that the healthy subjects could achieve success rates higher than algorithm's desired success rate, because the algorithm doesn't prevent subjects from hitting more correct notes than desired. In order to effectively challenge the unimpaired subject, the algorithm would need to have been able to make the game more difficult than it would naturally be with the assistance turned completely off. This is not necessary for the impaired subjects, whose reduced neuromuscular ability provided the increased difficulty.

We also measured how success rate and impairment level affected the subjects' effort while playing the game. Success rate was found to have a significant effect on subjects' effort ($p = 0.0024$, degrees of freedom=2). The effects of impairment level on effort, approached but did not achieve significance ($p =0.0785$, degrees of freedom=2). As shown in Figure 2.14, effort decreased when subjects' success rate increased.

Figure 2.15 shows the effects of impairment level and the finger being used on finger individuation. Both the finger being used and impairment level were found to have a significant effect on finger individuation ($p = .0001$, degrees of freedom=1 and $p = .0062$, degrees of freedom=2, respectively). As can be seen in Figure 2.15, individuation scores of the index finger were consistently better than those of the middle finger. This means that when the subject tried to move the index finger, he was more successful at moving the index finger only, as compared to when he tried to move the middle finger. Success rate was not found to have a significant effect on finger individuation, and so we combined data across success levels, resulting in Figure 2.16, which shows the effect of subjects' impairment level

on finger individuation. Subjects' with higher impairment had lower individuation ability. The ability to individuate the index finger was higher than the ability to individuate the middle finger.

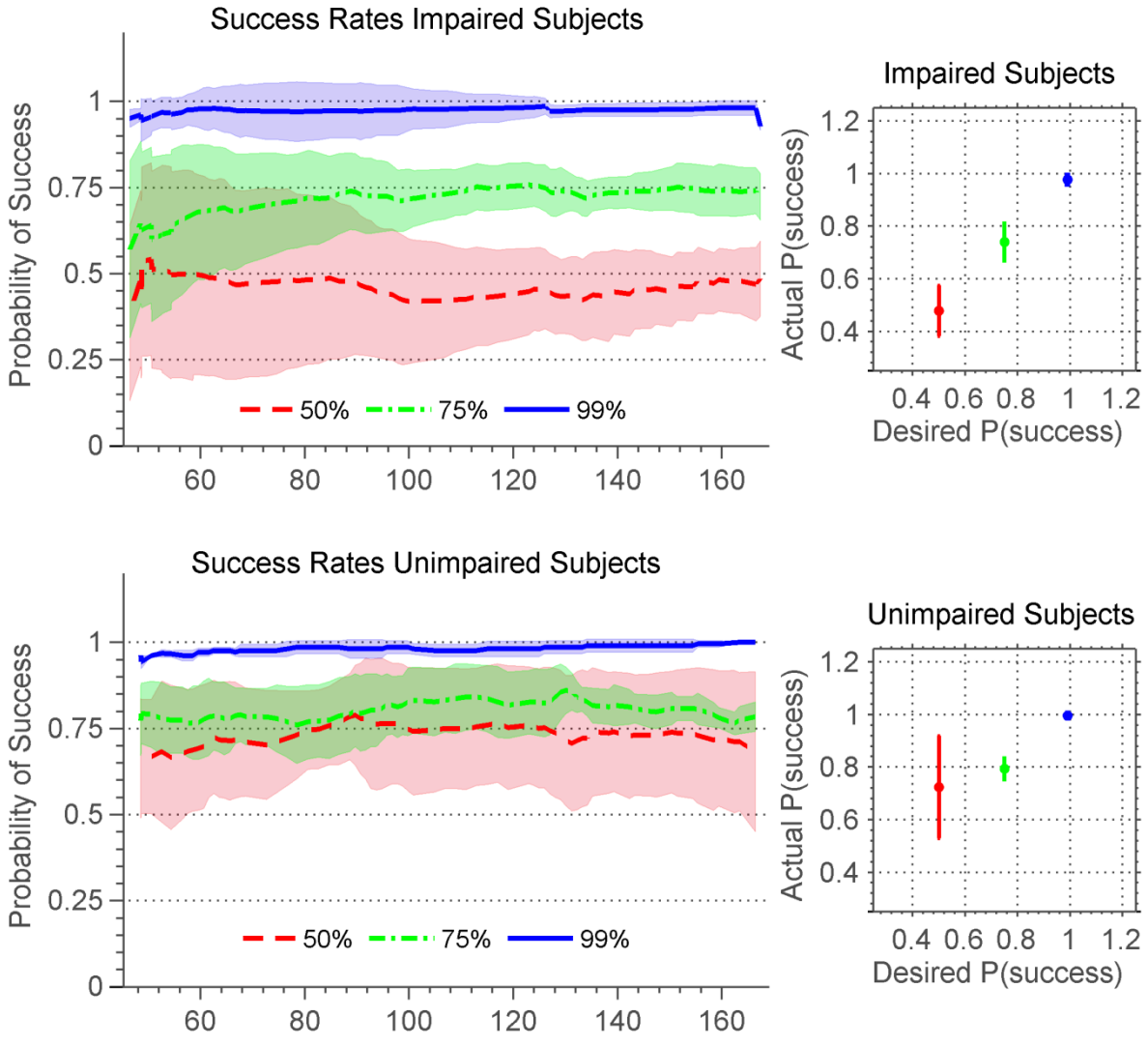

Figure 2.13. Actual success rates of stroke and unimpaired subjects. Actual success rates of stroke (top) and unimpaired (bottom) subjects for songs with desired success rates of 50% (red), 75% (green), and 99% (blue). Plots to the left show time progression of success rates. Lines are the moving window average over subjects and the shaded area is the standard deviation. Plots to the right show mean and standard deviation of desired vs. actual success rates at convergence.

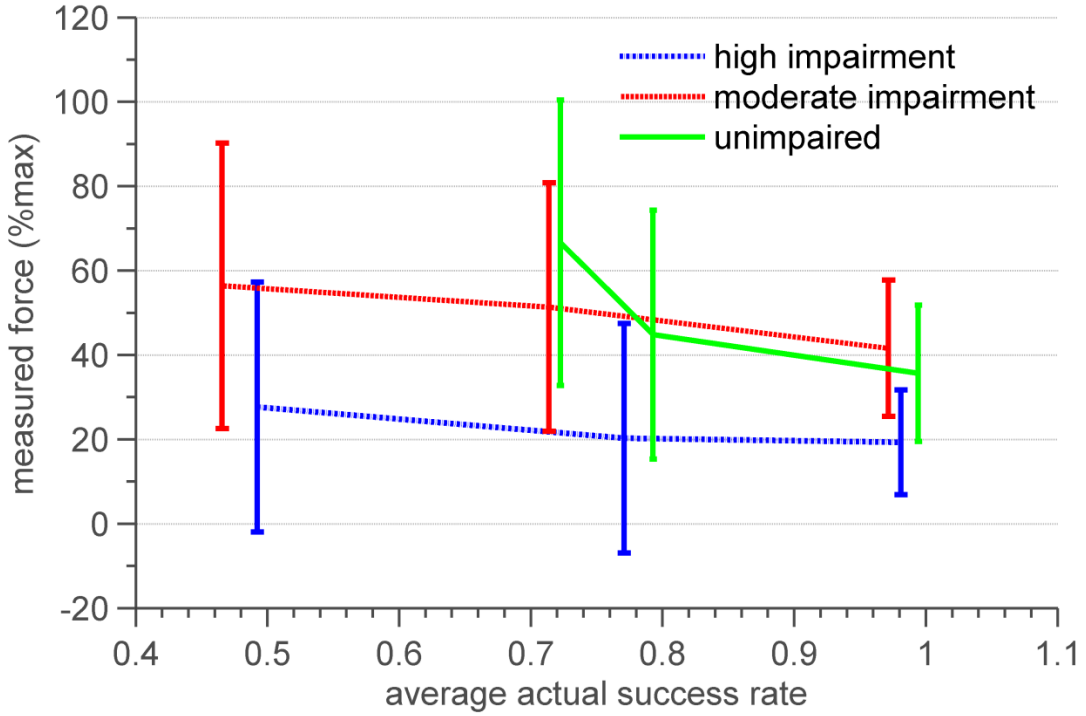

Figure 2.14. Average and standard deviation of effort versus average actual success rates for three groups of highly impaired, moderately impaired and unimpaired subjects. Effort was quantified as the peak force subjects generated during blocked notes. This peak force was then normalized to each patient's maximum force generated during isometric test.

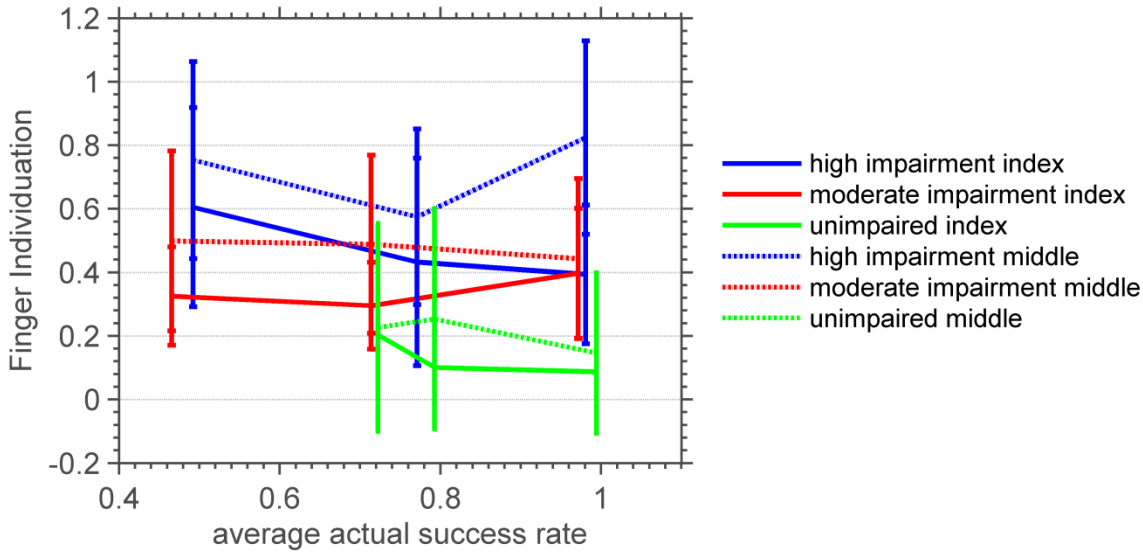

Figure 2.15. Average and standard deviation of the index and middle finger individuations versus average actual success rates of three groups of highly impaired, moderately impaired and unimpaired subjects. Finger individuation was quantified as the ratio of the normalized force generated by the finger that was not supposed to move to that of the finger that was during blocked notes.

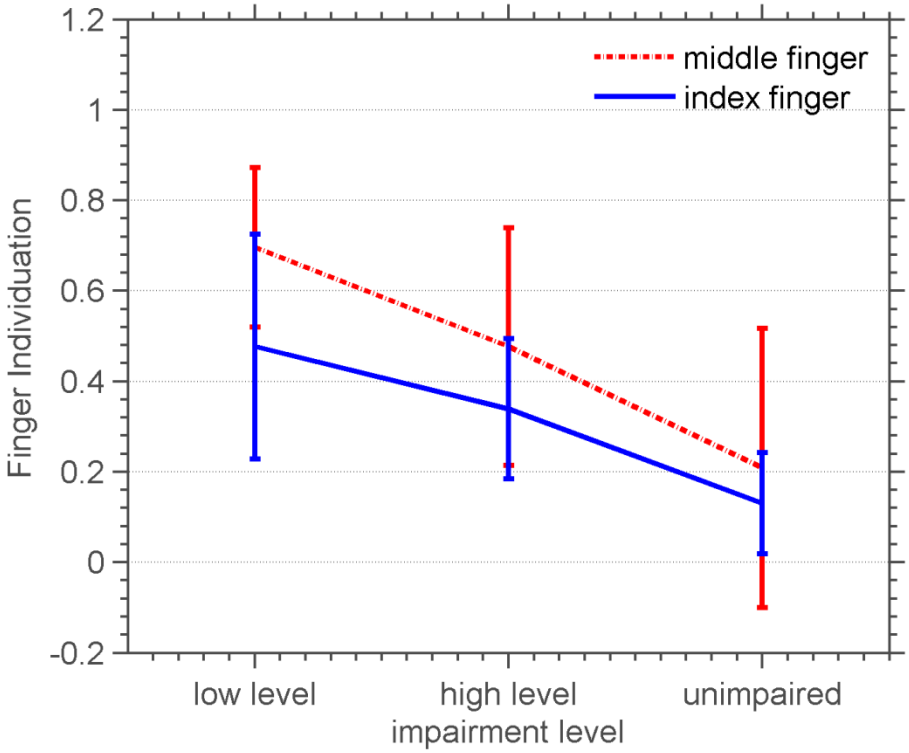

Figure 2.16. Average and standard deviation of finger individuation versus impairment level of three groups of highly impaired, moderately impaired and unimpaired subjects. Average finger individuation across success levels of both middle and index fingers.

2.5. Conclusions

This paper described the design and preliminary evaluation of FINGER (Finger Individuating Grasp Exercise Robot). FINGER makes use of individual single degree-of-freedom 8-bar mechanisms to assist patients in making a naturalistic grasping motion with different fingers, together or separately. The kinematic and mechanical design work was guided by the overall goal of creating a robot with high-control fidelity as an instrument for testing and implementing the widest possible range of control strategies. Thus, we paired the lightweight, low-friction mechanism with high-speed and un-geared linear actuators. The resulting robotic mechanism has a closed loop frequency response of approximately of -3 dB at approximately 8 Hz. The fast speed and frequency response of FINGER make it a good

candidate for evaluating control algorithms and therapy tasks that require fast movements and/or precise timing.

Another unique feature of FINGER is that, in contrast to most exoskeleton designs that attempt to align the joints of the robot with the joints of the body, the joints of the 8-bar mechanisms of FINGER are kept to the back of the hand and wrist throughout the curling motion. This facilitates easy attachment to the user and stacking of the mechanisms for multiple fingers, and allows for the possibility of applying sensory stimuli to the volar surface of the palm, for example by having individuals grasp real objects with assistance from FINGER.

The physical parameters of the 8-bar mechanism were determined through a mechanism synthesis process that achieved desired end-effector locations using cost function minimization. Four different sets of desired end-effector locations were created to generate a mechanism that could be easily adjusted to accommodate four different hand sizes. Using low-friction bearings and a balanced joint design, we were able to achieve smooth, low-friction 8-bar mechanisms that are easily backdriven. Further design features include finger-to-finder width adjustment, finger length adjustment, and wrist alignment.

Future upgrades to FINGER are currently under development. Possible upgrades and improvements include adding direct force sensing and impedance control, implementing unstructured "assist-as-needed" adaptive control, and adding a thumb exoskeleton mechanism.

The preliminary tests of FINGER showed that it can allow individuals with a range of impairment levels to play an engaging video game similar to Guitar Hero®. We used FINGER with a simple gain-adaptation algorithm to test the hypothesis that we can assist

subjects as needed in achieving predefined success levels at the game, which we confirmed. We also found that the effort of both high level and low level subjects decreased when their success rate increased; this is consistent with previous observations of slacking when a robotic device over-assists its user [62, 82].

According to CPF, there is an intermediate success rate in which learning is maximal. We do not find a success level at which effort was optimal. One possibility is that effort may not decrease unless success is below 50%, the lowest level we tested. Determining the relationship between measures of effort and the optimal challenge point is an important direction for future research.

These tests also demonstrated the ability of FINGER to quantify finger individuation. Using measurements during blocked trials based on patients' force applied by the wrong finger, we found that patients with higher impairment levels individuated less than those with lower levels of impairment. This result supports the findings in the previous literature on individuation that found that stroke reduced the ability to perform selective individuated finger motions, and specifically that the independence of the middle finger is more impaired than that of the index finger [102, 103]. A significant result is that we were able to quantify individuation during the normal course of game play of the game similar to Guitar Hero®. The possibility of generating quantitative measures of movement ability while therapy is delivered may increase the frequency at which these measures can be obtained [19].

The results of the preliminary tests with FINGER demonstrate its unique capabilities to study and implement finger therapy after stroke. Additional testing with FINGER may add insight to the effects of success rate on motor learning and finger movement recovery. We also plan to further explore the mechanisms of finger individuation in subjects with

impairment due to stroke. Such knowledge will guide the use of FINGER for post-stroke movement therapy.

2.6. Acknowledgements

The authors gratefully acknowledge the support for the work described herein by NIH-R01HD062744 from the National Center for Medical Rehabilitation Research at the National Institute of Child Health and Human Development, and the National Center for Research Resources and the National Center for Advancing Translational Sciences, National Institutes of Health, through Grant UL1 TR000153. The content is solely the responsibility of the authors and does not necessarily represent the official views of the NIH.

Chapter 3. Adaptive Control with State-Dependent Modeling of Patient Impairment for Robotic Movement Therapy

Curtis Bower[1], Hossein Taheri[1], Eric T Wolbrecht[1]

Rehabilitation Robotics (ICORR), 2013 IEEE International Conference on, 2013, pp. 1-6

3.1. Abstract

This paper presents an adaptive control approach for robotic movement therapy that learns a state-dependent model of patient impairment. Unlike previous work, this approach uses an unstructured inertial model that depends on both the position and direction of the desired motion in the robot's workspace. This method learns a patient impairment model that accounts for movement specific disability in neuro-muscular output (such as flexion vs. extension and slow vs. dynamic tasks). Combined with assist-as-needed force decay, this approach may promote further patient engagement and participation. Using the robotic therapy device, FINGER (Finger Individuating Grasp Exercise Robot), several experiments are presented to demonstrate the ability of the adaptive control to learn state-dependent abilities.

Keywords—adaptive control, assist-as-needed, rehabilitation robotics, movement therapy

3.2. Introduction

The use of robotic devices for post-stroke movement therapy continues to be an important and growing research area. Rehabilitation robots have previously demonstrated the ability to administer therapy in a consistent and prescribed manner, with therapeutic efficacy equal to or marginally exceeding conventional therapy (for reviews see [78, 79]). However, exactly how these devices should interact with patients during movement therapy remains an open

[1] Department of Mechanical Engineering, College of Engineering, University of Idaho, Moscow, Idaho, USA

and active research area. The goal is to create protocols and algorithms by which robotic therapy may maximize the functional recovery experienced by a patient.

Previous works suggest that too much assistance may limit or reverse the effects of therapy. For example, [62, 82] demonstrated that patients will reduce their effort when given the opportunity to do so without drastically affecting the desired motion. That is, if the robot is able to "take over" the movement, patients are willing to allow it to do so. And because patient effort is known to promote motor-plasticity during therapy [83, 84], it is important for robotic devices to only assist as much as necessary for a prescribed therapy motion.

To promote patient effort and engagement, assist-as-needed control strategies aim to restrain the power of a rehabilitation robot in a way that maximizes patient effort while simultaneously completing therapy movements. As noted in [15], a wide-variety of implementation approaches for assist-as-needed control have been investigated, including both static (examples include [31, 56, 85, 104]) and adaptive (examples include [16, 62, 74]). Unlike static approaches, adaptive strategies change control parameters during and/or between movement therapies in order to modulate assistance based on the patient's ability.

In previous work by the authors [62], the adaptive assist-as-needed approach was implemented using passivity-based adaptive control [75]. This implementation included an unstructured adaptive model of patient abilities and a force decay term that limited the robotic assistance needed to complete the prescribed movements. The resulting controller demonstrated the ability to modulate patient effort while keeping tracking errors small.

The need for an unstructured model stems from the nature of disability after stroke; each person who has suffered a stroke has unique neuro-muscular impairment. Because passivity-based adaptive control assumes that the system parameters are constants and correlated

directly to the geometry and kinematics of the robot, [62] implemented an unstructured model using radial basis functions to adapt to patient's neuro-muscular abilities. This use of radial basis functions allows the controller to create a model of each individual patient's impairment as a function of position. While other unstructured modeling approaches are possible, radial basis functions are linear in term of their magnitudes, and may thus be incorporated into the passivity-based adaptive control structure.

The work presented in [62] included two significant shortcomings. First, the adaptive model only represented patient force output and excluded inertial and viscous forces and is thus unable to properly assist with dynamic movements requiring higher velocities and specific timing. Increasing the rate of adaptation can improve the dynamic response, but at the expense of decreasing the compliance of the robot. This problem is noted in [105], which also presents a potential solution based on separate models for each target motion.

The second noted problem with the approach in [62] is that the unstructured model assumed patient ability was only a function of position and not a function of velocity or movement direction. In fact, the ability to model patient impairment by movement direction may be significant for rehabilitation. In the hand, for example, it has been shown that stroke survivors with hand impairment often have unequal impairment in metacarpophalangeal (MCP) joint movement. Often patients exhibit a deficiency in extension that is significantly greater than in flexion [106-109]. This phenomenon appears to be caused by an inherent weakness in extension activation signal [107] or inappropriate muscle co-activation [106] rather than passive mechanisms such as stiffness and spasticity. Thus, it is important to model directional impairment for providing movement assistance to stroke survivors.

In this paper we present an unstructured model for adaptively learning patient abilities that includes both inertial terms and state dependence (position and direction). By modeling patient impairment with better state-dependent resolution, the robot will be able to further minimize assistance to the minimal amount needed. In the following section, we describe the control algorithm and its implementation on a robotic device developed for finger rehabilitation. We then present experimental results that demonstrate the ability of the presented approach to learn a direction specific inertial model in order to assist with dynamic movements with unbalanced, directionally dependent force disturbances.

3.3. Methods

3.3.1. Robotic Therapy Device

The robotic therapy device, FINGER (Finger Individuating Grasp Exercise Robot) was used for experimental validation of the presented adaptive control approach. FINGER consists of two stacked, planar 8-bar mechanisms each with a single degree-of-freedom (DOF) [108, 109]. FINGER is capable of guiding the index and middle fingers individually through a naturalistic grasping motion that was based on motion capture of unimpaired subjects. Range of motion is limited with hard stops and the grasping mechanisms incorporate several easily adjustable components to accommodate physical differences between patients.

Several characteristics of FINGER make it an excellent test platform for the control strategies outlined in this paper. First, the combination of low-friction bearings, precision machining, lightweight components, and high speed linear actuators produce a high bandwidth (about 8Hz at -3dB) of direct force control. Furthermore, because the linear actuators lack any gearing or other high-friction components, FINGER is highly back-drivable.

3.3.2. Passivity Based Adaptive Control

One successful model-based adaptive controls system was implemented by [62] and expanded by [105]. This approach follows [75], which defines the adaptive control as

$$\begin{aligned}\mathbf{F}_r &= \mathbf{Y}\hat{\mathbf{a}} - \mathbf{K}_P\left(\mathbf{x}-\mathbf{x}_d\right) - \mathbf{K}_D\left(\dot{\mathbf{x}}-\dot{\mathbf{x}}_d\right) \\ &= \mathbf{Y}\hat{\mathbf{a}} - \mathbf{K}_P\left(\tilde{\mathbf{x}}\right) - \mathbf{K}_D\left(\dot{\tilde{\mathbf{x}}}\right),\end{aligned} \tag{3.1}$$

where $\mathbf{F}_r$ is the assistive force applied by the robot, $\mathbf{x}$ is an $n\times 1$ vector of generalized coordinates of the robot (subscript d denotes desired), $\mathbf{Y}$ is an $n\times m$ matrix of functions of known parameters and system dynamics $\left(\mathbf{x},\dot{\mathbf{x}},\mathbf{x_d},\dot{\mathbf{x}}_\mathbf{d},\ddot{\mathbf{x}}_\mathbf{d}\right)$, $\hat{\mathbf{a}}$ is an $m\times 1$ estimate of system parameters $\mathbf{a}$, and $\mathbf{K}_P$ and $\mathbf{K}_D$ are symmetric, positive-definite gain matrices. In (3.1), the terms $-\mathbf{K}_P\tilde{\mathbf{x}}$ and $-\mathbf{K}_D\dot{\tilde{\mathbf{x}}}$ are the proportional and derivative feedback portions of the control and $\mathbf{Y}\hat{\mathbf{a}}$ is a model of system dynamics. In this application, $\mathbf{Y}\hat{\mathbf{a}}$ is used to model state-dependent neuro-muscular impairment. The regressor matrix, $\mathbf{Y}$ is a sparsely populated, quasi-diagonal matrix with the form of

$$\mathbf{Y}^{n\times m} = \begin{bmatrix} \mathbf{g}^T & \cdots & 0 \\ \vdots & \ddots & 0 \\ 0 & 0 & \mathbf{g}^T \end{bmatrix}, \tag{3.2}$$

and $\mathbf{g}$ is a $p\times 1$ vector ($m = np$) of radial basis functions (RBFs) that spans the workspace of the robot with elements defined by

$$g_n = \exp\left(-\left\|\mathbf{x}-\boldsymbol{\mu}_\mathbf{n}\right\|^2/\sigma^2\right), \tag{3.3}$$

where the center of the n^{th} radial basis function is $\boldsymbol{\mu}_n$. The standard deviation, σ, specifies the width of the RBFs and may be selected to allow proper overlap. Adaptation of the amplitudes of the RBFs is achieved by

$$\dot{\hat{\mathbf{a}}} = -(1/\tau)\,\mathbf{Y}^T\left(\mathbf{Y}\mathbf{Y}^T\right)^{-1}\mathbf{Y}\hat{\mathbf{a}} - \boldsymbol{\Gamma}^{-1}\mathbf{Y}^T\mathbf{s}, \tag{3.4}$$

where τ is the decay time constant and $\boldsymbol{\Gamma}$ is a diagonal, positive-definite gain matrix. The first term governs the "assistance-as-needed"; it exponentially decays the weights in order to address the tendency of patients to allow a stiff controller to take over [62]. The second term in (3.4) updates the parameter estimates according to the error defined by the sliding surface, $\mathbf{s}$, which is defined as

$$\mathbf{s} = \dot{\tilde{\mathbf{x}}} + \boldsymbol{\Lambda}\tilde{\mathbf{x}}, \tag{3.5}$$

where $\boldsymbol{\Lambda}$ is a diagonal constant gain matrix that specifies the ratio of position error to velocity error. The adaptive controller described above was modified in [105] by splitting up the desired trajectory into multiple segments and utilizing a separate $\hat{\mathbf{a}}$ vector for each, thus reducing the need for fast adaptation which increases controller stiffness. The adaptive control approach presented in this paper also improves controller compliance, but unlike [75], it does so with a continuous workspace model of patient impairment, that is state-dependent. This impairment model represents the inability of a patient to generate forces during movement as a function of both movement position and direction.

3.3.3. Inertial Adaptive Control Model

In [62] the regressor matrix $\mathbf{Y}$ does not include viscous or inertial components. This limits complexity, but it also restricts the types of dynamic movements that can be effectively assisted. The following describes the implementation of the inertial and viscous components.

Since physical validation will be completed on FINGER, the model can be reduced to a single DOF. This is for experimental validation and is not a limitation of the approach, which can be expanded for multi- DOF robotic devices. For the single DOF robotic device FINGER, we define an impairment model according to

$$\mathbf{Y} = \begin{bmatrix} g_1\dot{w} & \cdots & g_n\dot{w} & \dot{x} & x \end{bmatrix}, \tag{3.6}$$

where g_{1-n} are n RBFs distributed from full extension to full flexion of the finger, and $w = \dot{x}_d - \Lambda\tilde{x}$. By using radial basis functions, the system mass is not required to be constant across the range of motion, and more closely match the inertial ability (as mass) of the patient as a function of position. The corresponding system parameter vector is

$$\hat{\mathbf{a}} = \begin{bmatrix} \gamma_1 & \cdots & \gamma_n & \mathrm{b}_\gamma & \mathrm{k}_\gamma \end{bmatrix}^T, \tag{3.7}$$

where γ_{1-n} are the amplitudes of the respective radial basis functions and k_γ and b_γ are adapted stiffness and damping, respectively.

3.3.4. State-Dependent Adaptive Control

Many stroke survivors exhibit significant weakness in one direction (typically extension) over the other [106-109]. Due to this phenomenon, (3.6) is modified to account for possible variation in patient impairment between flexion and extension. To achieve this, a separate parameter vector is used for flexion and extension, such that (3.6) becomes

$$\mathbf{Y} = \begin{bmatrix} (.5\tanh 80\dot{x}_d + .5)\mathbf{y} \\ (-.5\tanh 80\dot{x}_d + .5)\mathbf{y} \end{bmatrix}^T \tag{3.8}$$

where

$$\mathbf{y} = \begin{bmatrix} g_1\dot{w} & \cdots & g_n\dot{w} & \dot{x} & x \end{bmatrix}^T. \tag{3.9}$$

The system parameter vector was expanded from (3.7) to

$$\hat{\mathbf{a}} = \begin{bmatrix} \alpha_1 & \cdots & \alpha_n & \mathrm{b}_\alpha & \mathrm{k}_\alpha & \beta_1 & \cdots & \beta_n & \mathrm{b}_\beta & \mathrm{k}_\beta \end{bmatrix}^T \tag{3.10}$$

where α_n represents the amplitude of the n^{th} radial basis function in flexion and β_n represents the corresponding amplitude in extension. Combining the amplitudes with the RBFs creates distribution models of the effective "mass" in flexion and extension that describes the inability of the subject to create forces necessary to accelerate their fingers according to a desired trajectory. These "mass" distribution models are a function of position and defined for a single mass distribution model, *m*, a flexion specific model, m_F, and an extension specific model, m_E, according to:

$$\begin{aligned} m &= \begin{bmatrix} \gamma_1 & \ldots & \gamma_n \end{bmatrix} \begin{bmatrix} g_1 & \ldots & g_n \end{bmatrix}^T \\ m_F &= \begin{bmatrix} \alpha_1 & \ldots & \alpha_n \end{bmatrix} \begin{bmatrix} g_1 & \ldots & g_n \end{bmatrix}^T \\ m_E &= \begin{bmatrix} \beta_1 & \ldots & \beta_n \end{bmatrix} \begin{bmatrix} g_1 & \ldots & g_n \end{bmatrix}^T . \end{aligned} \tag{3.11}$$

The hyperbolic tangent functions included in (3.8) activate either the first or second set of radial basis functions based on the direction of the desired velocity. They also remove the discontinuity that occurs as the desired velocity crosses zero. Thus, in the small region around zero (approximately ±.02 m/s) both the flexion and extension sets of parameters will be proportionally updated and will both influence output. This method could be expanded to a spatial robot by using two sets of radial basis functions for each Euclidean direction.

3.3.5. Experimental Protocol

Four experiments were conducted to validate the presented adaptive controller. For each experiment, four unimpaired subjects, securely connected to FINGER (see Figure 3.1), were guided by a simple visual interface that displayed two markers on a radial path (see

Figure 3.2). One marker corresponded to the position of the user's finger and the other marker represented the desired location. The participants were instructed to follow the desired marker which followed a minimum jerk trajectory between two points pseudo-randomly distributed between the bounds of FINGER's functional workspace. This point-to-point trajectory operated at 1.5 Hz, regardless of movement size. Each of the three experiments was repeated with adaptation based on a direction specific (using the model defined by (3.8), (3.9), and (3.10)) and a single mass distribution (using the model defined by (3.6) and (3.7)) model.

These four experiments were designed to investigate the ability of the proposed approach to adapt to directionally dependent patient impairment. In order to simulate impairment, directional viscous force fields were superimposed with the robot.

In the first experiment, only the inertia and friction of the device (both relatively low) impeded the performance of the participants. For the second procedure, the desired trajectory and visual feedback were unchanged. However, a significant viscous field was superimposed in the flexion direction. The experiment was then repeated with the viscous field superimposed in the extension direction. This impeded the ability of the subject to perform the flexion or extension task.

The third experiment follows the same procedure as the second with the inclusion of the forgetting term with a force decay rate of $\tau = 1$. The fourth experiment followed the same procedure as the second, except that the artificial viscous field was removed suddenly after 40 seconds. All four experiments are summarized in Table 3.1.

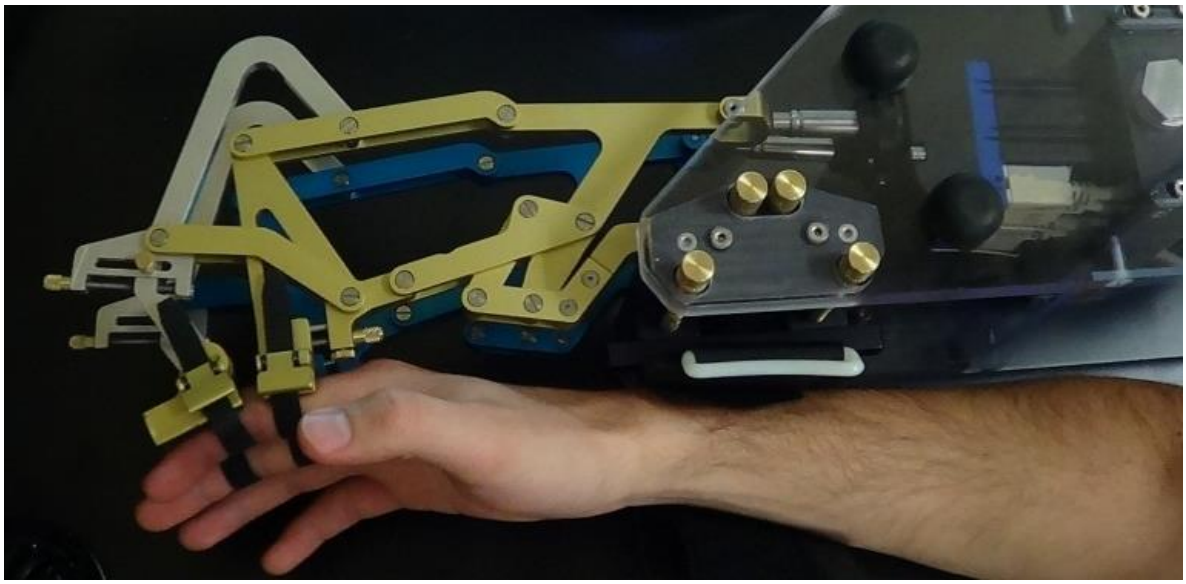

Figure 3.1. Finger INdividuating Grasp Exercise Robot (FINGER) viewed from the top. FINGER can individually guide the index and pointer fingers through a naturalistic grasping motion.

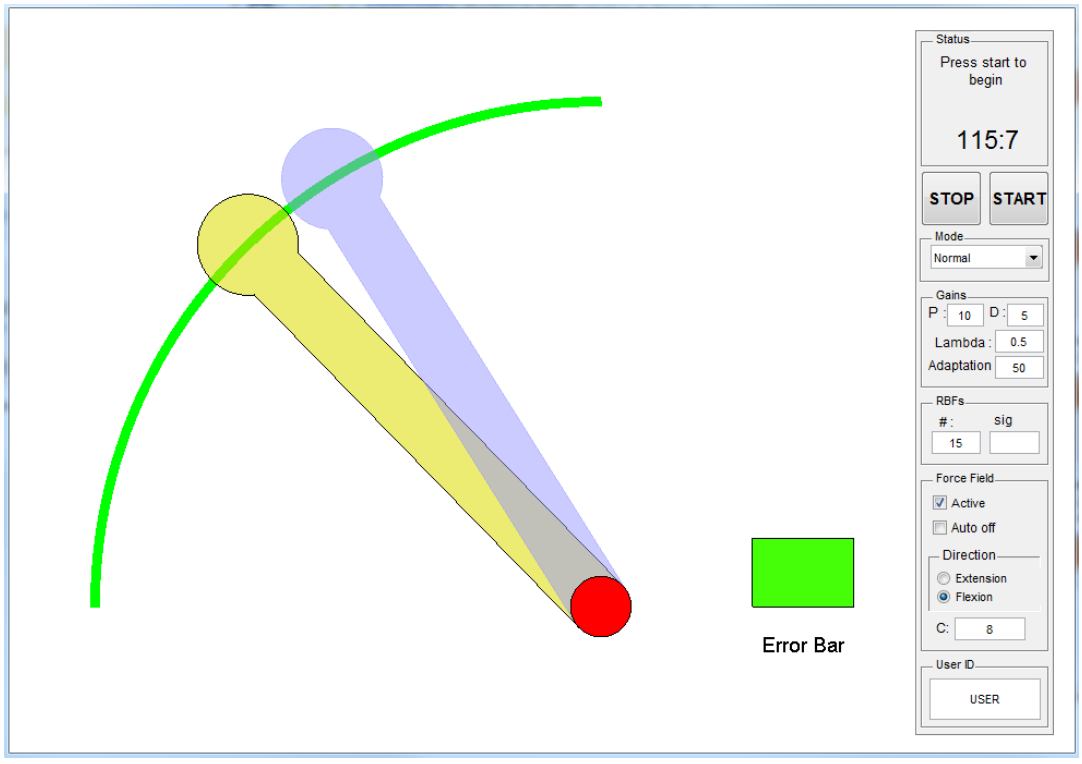

Figure 3.2. The visual interface during experiments. The blue marker shows the desired angle/position and the yellow marker shows the actual angle/position.

Table 3.1. Summary of the experiments.

Test	**Task**	**Controller**	τ	**Force Field**
1	Min. jerk tracking at 1.5 Hz	Both with and without direction specific adaptation.	∞	None
2	Min. jerk tracking at 1.5 Hz	Both with and without direction specific adaptation.	∞	Viscous field applied opposing either flexion or extension.
3	Min. jerk tracking at 1.5 Hz	Both with and without direction specific adaptation.	1	Viscous field applied opposing either flexion or extension.
4	Min. jerk tracking at 1.5 Hz	Both with and without direction specific adaptation.	∞	Viscous field applied opposing flexion or extension for the first 40 seconds.

During testing the controller gains and adaptation parameters were set to $K_P = 1.15$, $K_D = 0.575$, $\Gamma = 0.02$, and $\Lambda = 0.5$. Furthermore, $n = 15$ RBFs were evenly distributed with $\sigma = 0.007$ m^2. The magnitude of the superimposed viscous field was $C = 1 \frac{N \cdot s}{m}$ in all pertinent experiments.

3.4. Results

The results from the first experiment shown in Figure 3.3 illustrate how RBF coefficients adapt with the either the direction specific or a single mass distribution models. The figure shows the four subject average of the converged mass models for both flexion, m_F, and extension, m_E, using the direction specific model for adaptation. It also shows the converged mass model, m, when adaptation was based on a single mass distribution. In all cases, adaptation is limited near both ends of the workspace (full flexion and full extension)

because the desired trajectories lack significant velocity and acceleration in those regions (and thus mass adaptation is limited).

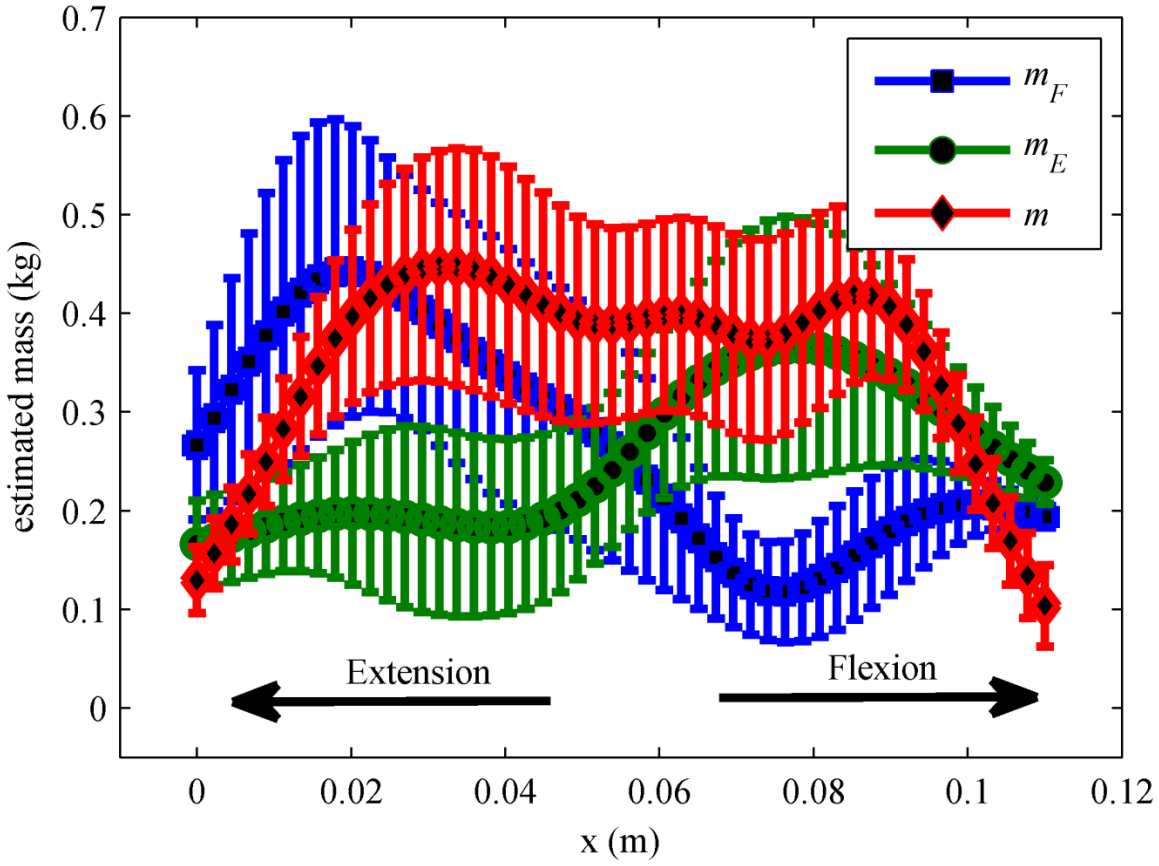

Figure 3.3. Estimated average mass distribution of four subjects from the first experiment. The blue and green markers are the mass distribution in the direction specific model and red is the single mass distribution.

With careful inspection of Figure 3.3, it is clear that when direction specific adaptation is present, the mass model for flexion has greater amplitude near full extension ($x = 0\,m$). Conversely, the mass model for extension has greater amplitude near full flexion ($x = 0.114\,m$). A possible explanation for this phenomenon is that larger motions require higher accelerations which cause faster adaptation, as evident in (3.6), and larger flexion motions must start near full extension. Thus, the largest adaptation for the flexion model appears near full extension. The same argument is valid for extension motions and the

extension mass model. However, adaptation with a single mass distribution does not take advantage of this propensity and all weights increase/decrease regardless of the direction of motion.

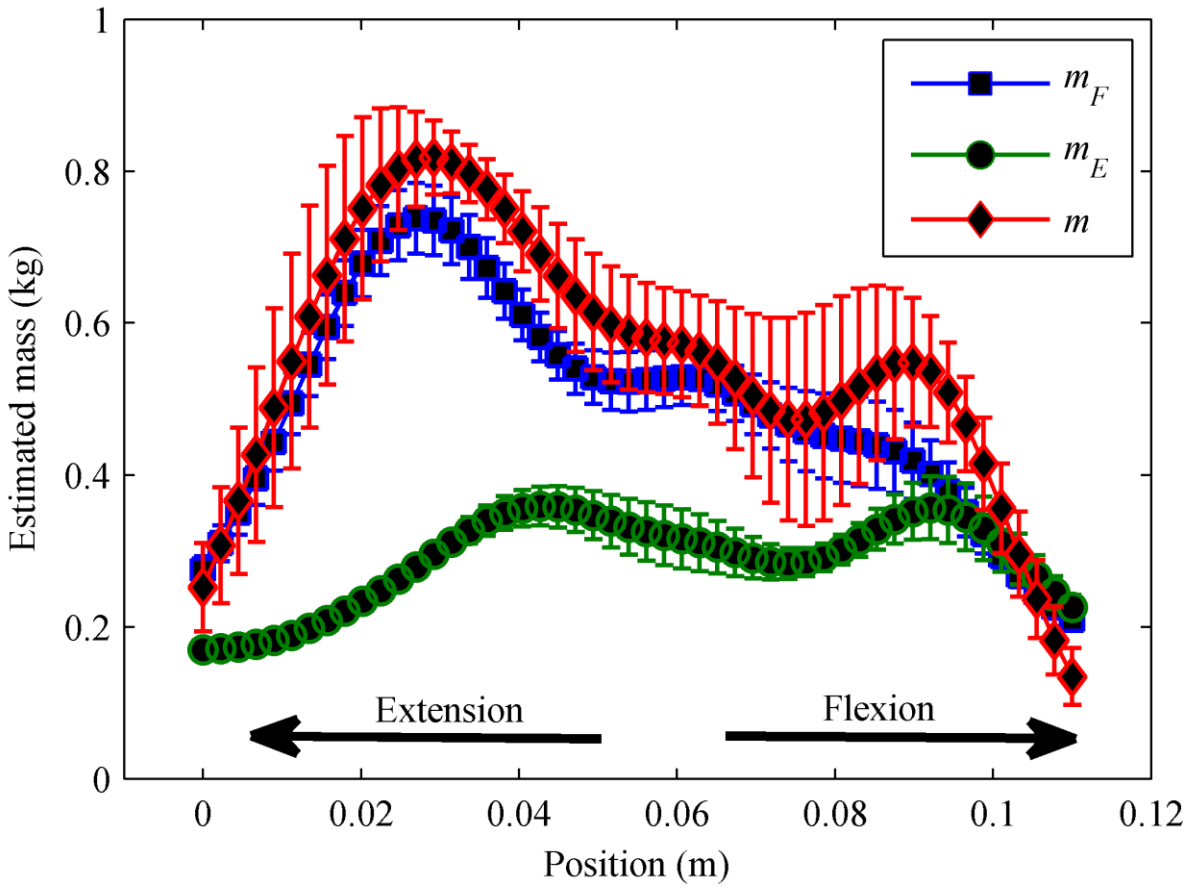

Figure 3.4. Estimated average mass distribution of four subjects with a viscous field applied in flexion from the second experiment. The blue and green markers are the mass distribution in the direction specific model and red is the single mass distribution.

In the second experiment, a viscous force field was superimposed on the subjects separately during either flexion or extension. Figure 3.4 is analogous to Figure 3.3 and shows the mass distribution using the direction specific and single mass distribution models with the viscous field applied in flexion. Because the force field was applied in only one direction, the results show a clear difference in the direction specific modeling versus the single mass distribution.

Figure 3.5 shows the position error of a subject under the two adaptive control schemes. Position errors are slightly smaller when the direction specific adaptive control is used. Since the viscous field was applied in flexion, the position error doesn't have a zero mean; the subjects have to overcome the force field and are behind the desired trajectory. This causes the error to be biased toward positive numbers. The results from extension were similar.

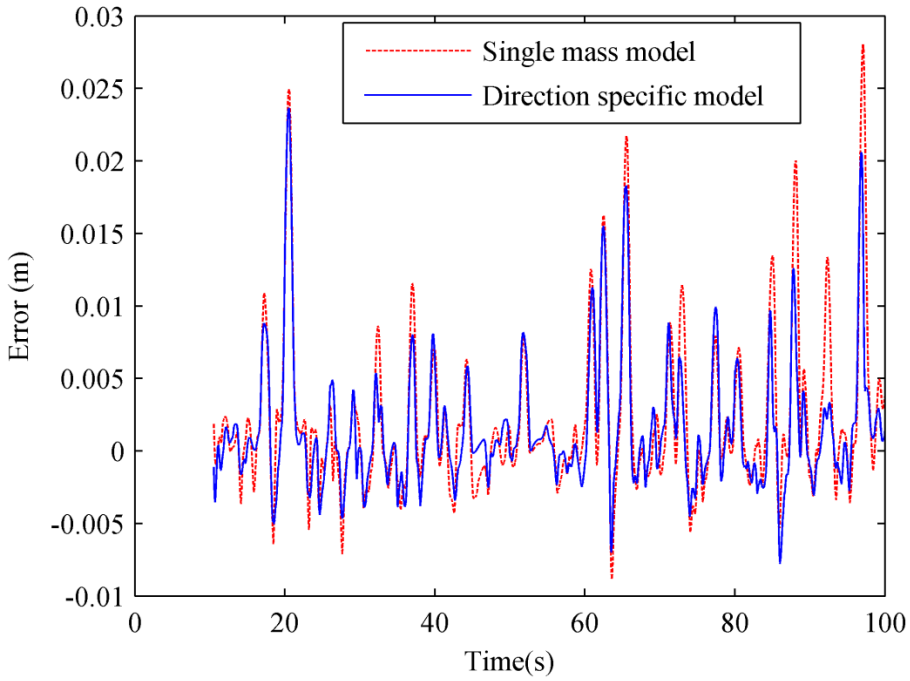

Figure 3.5. Position error using the two adaptive control schemes from the second experiment for one patient.

Figure 3.6 demonstrates how the adaptation gains evolve throughout a typical trial when the viscous field was applied to subject in the flexion direction. The two adaptive controller schemes were used to assist the subject to follow the same desired trajectory. Figure 3.6 (a) shows the instantaneous mean of the mass parameters, $\bar{\gamma}_M$, $\bar{\alpha}_M$ and $\bar{\beta}_M$, versus time. It is observed that, as suggested by Figure 3.4, in the direction specific controller, the average mass parameters are smaller than the ones in the controller with single mass distribution.

Since there is no apparent stiffness in the system, one could predict a zero estimated stiffness. However, this is not the case when the single mass distribution adaptation is used. In contrast, the direction specific adaptation settles to the predicted zero stiffness as shown in Figure 3.6 (b).

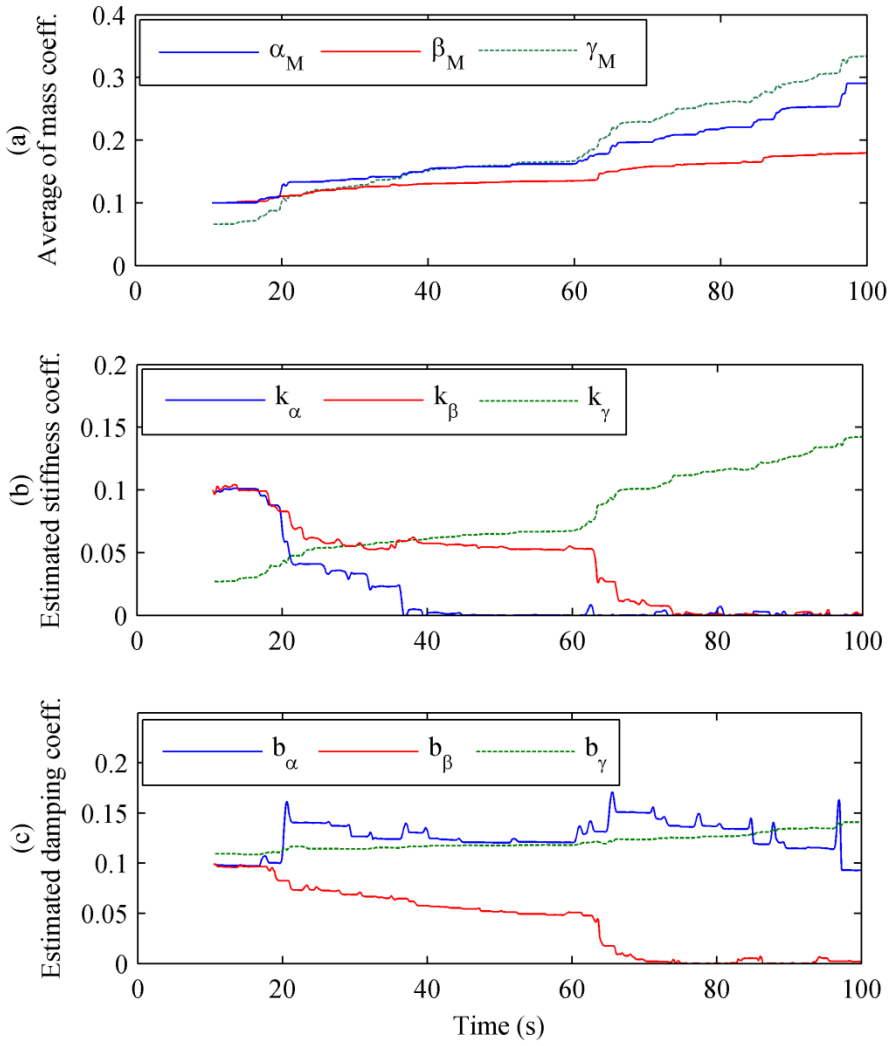

Figure 3.6. Parameter adaptation for the second experiment with a viscous field applied in flexion. The solid lines are the directional specific adaptation and the dashed lines are the directional independent adaptation.

Figure 3.6 (c) shows how the two damping parameters are independently adapted in the direction specific adaptation. The parameter along the direction that the force field was

applied (flexion) continues to adapt while the coefficient associated with extension goes to zero.

In the third experiment, the procedure from the second experiment was repeated with the forgetting portion of the control included ($\tau = 1$). Figure 3.7 shows the parameter adaptation of a trial with a viscous field applied in extension.

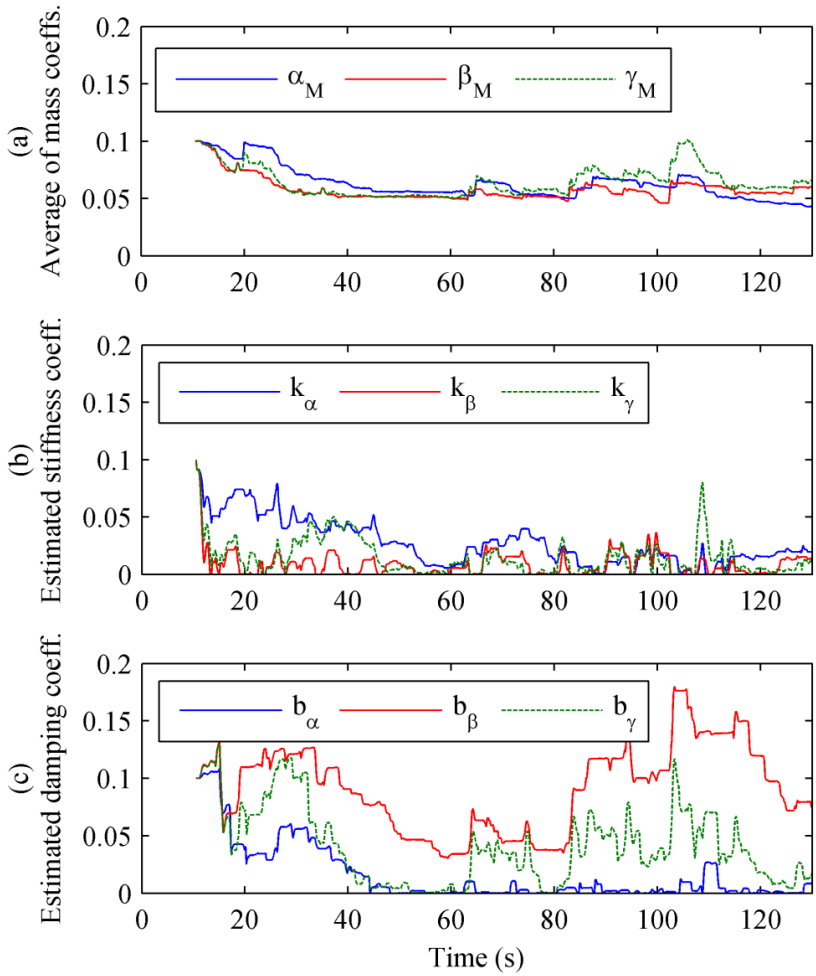

Figure 3.7. Parameter adaptation for the third experiment with $\tau = 1$ and a viscous field applied in extension. The solid lines are the direction specific model and the dashed lines are the single mass distribution.

In the final experiment, the direction specific adaption and a directional force field was applied separately in both directions. However, the force field was removed after forty

seconds. As a result, the damping coefficient estimate corresponding to the direction that the force field was applied increased until forty seconds and then suddenly went to zero as force field turned off as predicted by the model. This is shown in Figure 3.8 for a case where the viscous field was applied in flexion.

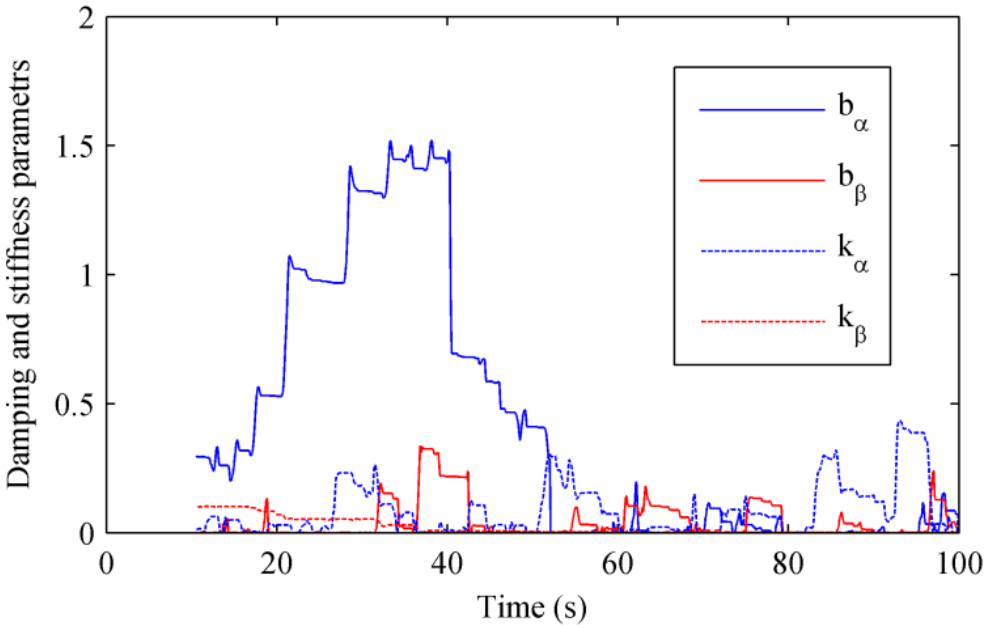

Figure 3.8. Direction specific model parameter adaptation for the fourth experiment with a viscous field applied in flexion for the first 40 seconds. The solid lines are the damping coefficients and the dashed lines are the stiffness coefficients.

3.5. Discussion and Conclusion

This paper describes an adaptive control algorithm for robotic assisted movement therapy that includes direction specific mass distributions for modeling neuro-muscular impairment. Previous work in impairment modeling lacks inertial terms or directional dependence which limits its ability to effectively assist with dynamic motions and direction specific impairment. By including direction specific mass distributions modeled with RBFs the proposed adaptive controller can tailor its assistance to individual patients. Combined with the assistance-as-needed force decay, this approach decreases control effort, which is important to promote participation and effort from the patient.

The direction specific model was compared with the single mass distribution, performing with less error. Furthermore, in the presence of a superimposed asymmetric force field, the single mass distribution increased stiffness due to its inability to learn the direction specific behavior. However, with the proposed controller, which includes a direction specific model, the adapted stiffness did not exhibit this increase.

Future efforts will focus on implementing the adaptive control approach on a multiple DOF system. In addition, an expansion of the RBFs to consider the full state space (variable velocity rather than movement direction only) will also be implemented. Future experiments are also planned with impaired and unimpaired subjects to better understand how to model neuro-muscular impairment after stroke. Final validation will include therapeutic evaluation of the proposed approach with impaired subjects.

Chapter 4. Design and Integration of a 4DOF Parallel Robot for Upper Extremity Motion Therapy

Various robots have been developed by researchers for motion therapy of the arm and hand as discussed in Chapter 1. These devices can be categorized into two main groups of end-effector based and exoskeleton based robots. Although using exoskeleton based robots can help with guiding patients in particular joint space trajectories, design of such devices can be complex due to joint alignment issues. End-effector based systems, on the other hand, have simpler designs and standard robot designs or industrial robot arms can be employed to reduce design complexity. In this research, a four-degrees-of-freedom (DOF) parallel robot was designed and optimized that can assist with arm motions in a large workspace through its end-effector. The parallel topology was chosen because unlike serial robots, the actuators don't move with robot and hence lower impedance robots can be achieved with the same output power as serial robots.

4.1. Mechanical Design

4.1.1. Robot topology

The robot design is based on the parallel robot proposed in [110, 111] that has three translational degrees of freedom and one rotational degree of freedom about a fixed axis. This robot architecture can provide high performance in terms of power transmission with no passive links. A 3DOF version of this mechanism, called Delta platform, has been widely used in different industrial and research settings. The schematic of the robot is shown in Figure 4.1, where R and U represent revolute (single rotating joint) and universal (two rotating joints with intersecting axes), respectively. The mechanism is based on four

independent kinematic chains between ground and the end effector. Link 1 on each chain is actuated independently with an actuator that is directly connected to the base.

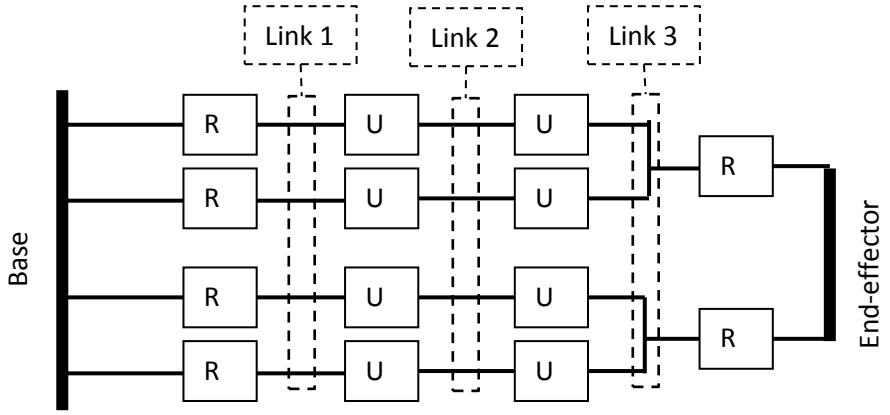

Figure 4.1. Schematic of the robot. The vertical line on the left is the ground and the vertical line on the right is the robot end-effector.

Figure 4.2 shows a schematic of the robot. The special location of links relative to each other is shown in Figure 4.2 (a). The first link is connected to the base through a revolute joint. This link is then connected to the second link through a universal joint. The second link is connected to the third link through another universal joint and the third link is connected to the end-effector through a revolute joint Figure 4.2 (c). Each wing has its own first and second link and the third link is shared between two of the chains as can be seen in Figure 4.2. A handle, equipped with suitable wrist and arm support will be attached to the end-effector to interface with patients. The kinematics of the robot is such that the end-effector, and hence patient's hand, can move in three directions and rotate about z axis as shown in Figure 4.2 (b) [111]. The two third links move parallel to each other with a fixed orientation at all times as can be seen in Figure 4.2 (d). In other words, the rotation axis of the revolute joint between Link 3 and the end-effector always remains vertical. It is possible to replace link 2 with a parallelogram [111] as shown in Figure 4.2 (bottom-right). Using

parallelograms can increase the structural stability and make manufacturing easier as it requires revolute joints only.

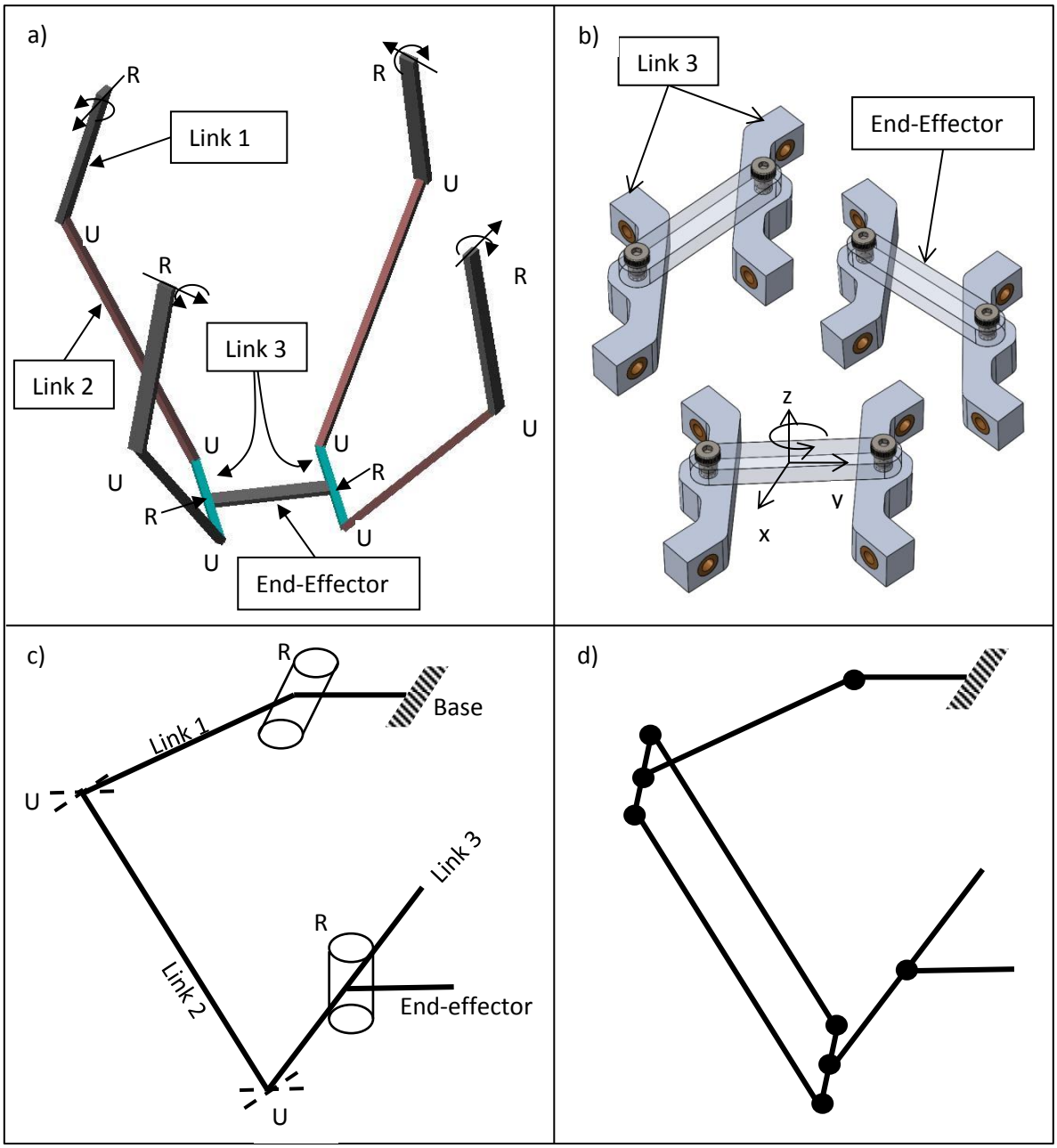

Figure 4.2. Schematics or the robot. a) The four gray links are Link1, the red links are Link 2, blue links are Link 3. b) end-effector motion. c) Structure of one chains of the robot between the base and the end-effector. d) The equivalent structure with a parallelogram. Black circles are revolute joints.

The possible degrees of freedom of the patient's hand when attached to the robot end-effector with the current configuration are illustrated in Figure 4.3. Although the current

downward facing design of the robot only allows for wrist extension-flexion, it is possible to rotate the entire robot around x-axis or y-axis to allow for wrist pronation-supination or radial deviation-Ulnar deviation instead.

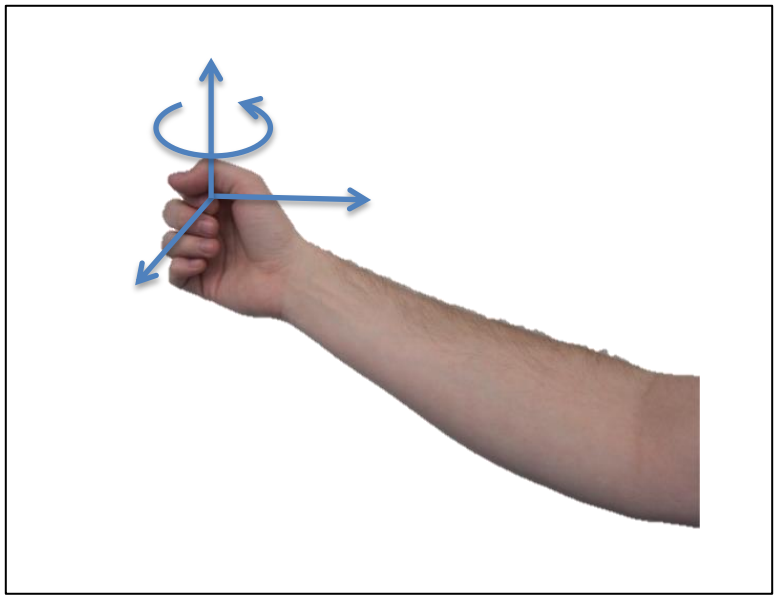

Figure 4.3. Possible motions of arm and hand when attached to the robot's end-effector, i.e. translation in x, y and z and wrist flextion-extension.

4.1.2. Weight Counter-balance

As can be seen in Figure 4.2 (a), robot's weight always pushes the end-effector down. It can put extra load on the actuators and a portion of their power always goes to cancel robot's own weight. This issue can be partially addressed by using linear actuators instead of rotational actuators in a configuration shown in Figure 4.4. In this configuration, Link 1 is extended back and forms a slider-crank mechanism with the linear actuator. If the linear actuator has a heavy-enough thrust rod and a correct geometry is chosen for the robot, some of the robot's weight can be counter-balanced by the actuators weight. The full cad model of the robot is shown in Figure 4.5.

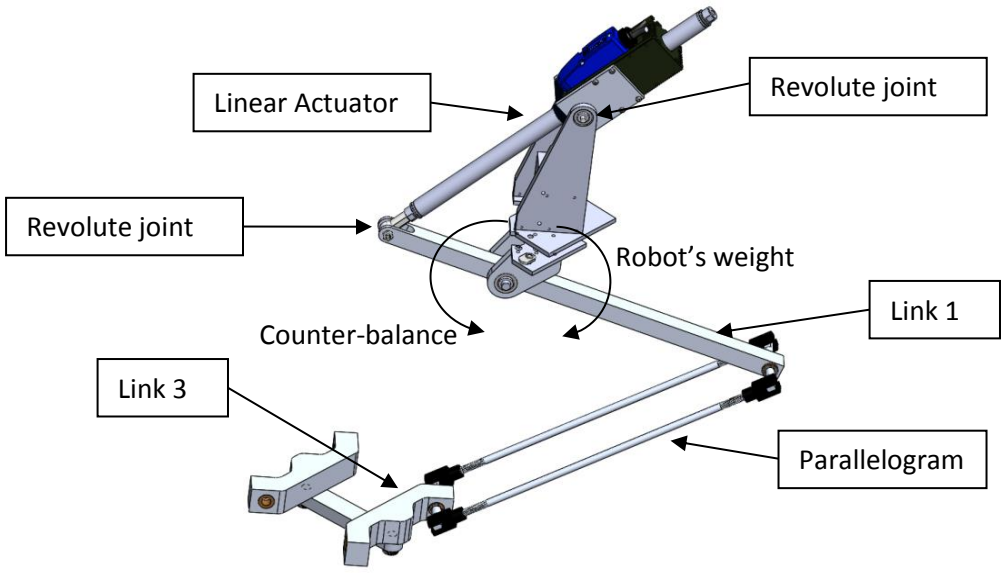

Figure 4.4. One of the kinematic chains of the robot and the counter-balance mechanism. The counter-balance torque is provided by the weight of the actuator thrust rod and the extension of Link1.

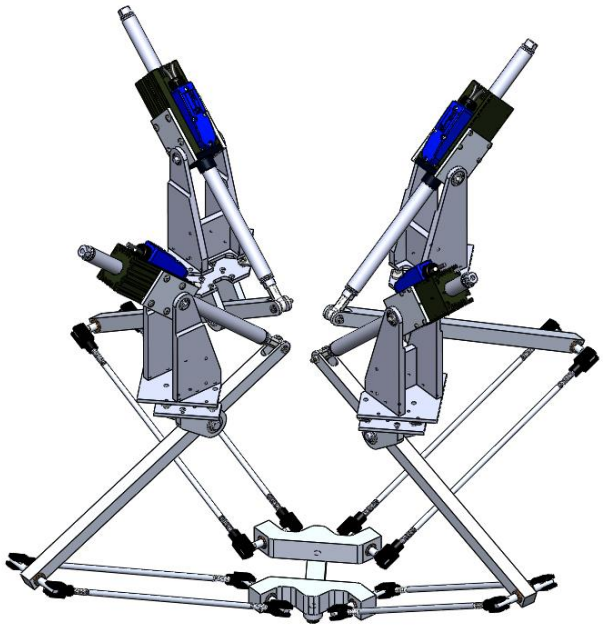

Figure 4.5. The CAD model of the robot.

4.1.3. Actuation

Each chain can be actuated individually by using rotational actuators to turn the revolute joint between the base and the first link. To achieve good power output, these actuators require high gear ratio that can consequently increase robot impedance. As mentioned in Chapter 1, a key characteristic of rehabilitation robots is backdrivability. Using such gearing systems can severely decrease the backdrivability of the robot. To address this issue, a linear brushless electric actuator (Dunkermotoren STA2504) was used. This actuator doesn't have any gearing and is very backdrivable. It also is directly force controllable and has a very accurate built-in position encoder. To drive the actuators, we used the manufacturer's recommended current drive (Copley Controls XSJ-230-10). Table 4.1 lists some mechanical and electrical specifications of the actuator.

Table 4.1. Actuator specifications.

Specification	Value	Units
Peak force	156	N
Continuous stall force	51.2	N
Peak current	20	Apk
Continuous current	4.62	Arms
Force constant	11	N/Arms
Stroke	309	mm
Thrust rod density	3.5	Kg/m

4.2. Kinematics

4.2.1. Forward and inverse kinematics

Figure 4.6 shows i^{th} quarter of the robot. Each quarter has an internal and an external kinematic loop. The internal loop, $OH_iY_iS_iR_iO'$ is the same as the kinematic chains shown in Figure 4.1 and includes the origin, Link 1, Link 2, Link 3 and the end-effector. The internal loop equation can be written as

$$\mathbf{P}_i + \mathbf{A}_i + \mathbf{L}_i + \mathbf{D}_i + \mathbf{E}_i = \mathbf{X}, \tag{4.1}$$

where $\mathbf{P}_i$ is the position vector of the Link 1 revolute joint of the i^{th} chain as shown in Figure 4.6 (bottom-right), $\mathbf{X}$ is the end effector position vector, $\mathbf{D}_i$ is the vector between the i^{th} chain's third links universal joint and revolute joint (see bottom-right of Figure 4.6), and $\mathbf{E}_i$ is the vector between the end-effector center and its revolute joint as shown in Figure 4.6.

The external loop $O_iH_iQ_i$, i.e. the actuator slider-crank mechanism, includes the base, the actuator thrust rod and the first link's extension (see Figure 4.6). The loop equation can be written as

$$\mathbf{B}_i + \mathbf{C}_i = \mathbf{q}_i, \tag{4.2}$$

where $\mathbf{q}_i$ is the thrust rod vector, $\mathbf{C}_i$ is the vector of the first link's extension and $\mathbf{B}_i$ is the vector from the actuator pivot to the first link's pivot (see Figure 4.6). Equation (4.1) and (4.2) can be simplified to the following algebraic equations

$$\begin{array}{c} f_i\left(x, y, z, \theta, \varphi_i\right) = 0 \\ q_i = g\left(\varphi_i\right) \end{array} \quad i = 1, \ldots, 4 \tag{4.3}$$

where φ_i is the rotation angle of the i^{th} the angle of H_iY_i as shown in Figure 4.6, x, y, z and θ are the coordinates of the end-effectors and its angle and $q_i = \|\mathbf{q}_i\|$. Equation (4.3) can be written as

$$\begin{array}{c} \left(a_i - A\sin\left(\alpha_i\right)\cos\left(\varphi_1\right)\right)^2 + \left(b_i + A\cos\left(\alpha_i\right)\cos\left(\varphi_i\right)\right)^2 + \left(z + A\sin\left(\varphi_i\right)\right)^2 - L^2 = 0 \\ q_i^{\,2} = a^2 + b^2 + C^2 + 2C\sqrt{a^2 + b^2}\cos\left(\varphi_i + \tan^{-1}\left(\frac{b}{a}\right)\right) \end{array} \tag{4.4}$$

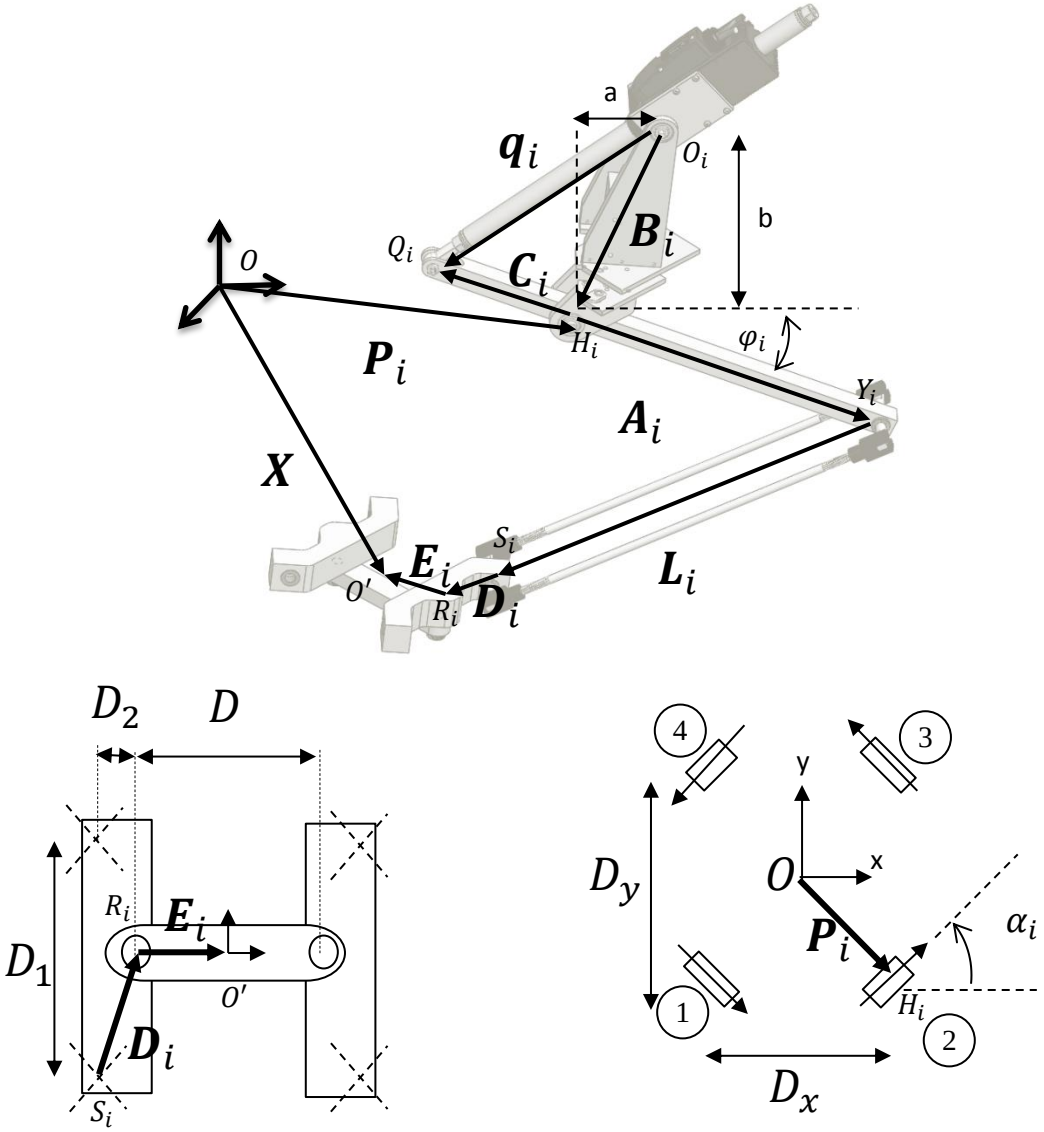

Figure 4.6. Each kinematics chain's internal and external kinematic loops (top) and the details of vectors (bottom). Chain numbers are marked in the bottom-right diagram.

where $A=\|\mathbf{A}_i\|, L=\|\mathbf{L}_i\|$, $C=\|\mathbf{C}_i\|$, α_i is the angle of rotation vector of the i^{th} Link1, $i=1,\ldots,4$ and a and b are shown in Figure 4.6 (top). Using the chain numbering shown in Figure 4.6 (bottom-right), the values of a_i and b_i can be found to be:

$$\begin{aligned} a_1 &= x + \frac{D_x}{2} - D_2 - \frac{D}{2}\cos(\theta) \quad b_1 = y + \frac{D_y}{2} - \frac{D_1}{2} - \frac{D}{2}\sin(\theta) \\ a_2 &= x - \frac{D_x}{2} + D_2 + \frac{D}{2}\cos(\theta) \quad b_2 = y + \frac{D_y}{2} - \frac{D_1}{2} + \frac{D}{2}\sin(\theta) \\ a_3 &= x - \frac{D_x}{2} + D_2 + \frac{D}{2}\cos(\theta) \quad b_3 = y - \frac{D_y}{2} + \frac{D_1}{2} + \frac{D}{2}\sin(\theta) \\ a_4 &= x + \frac{D_x}{2} - D_2 - \frac{D}{2}\cos(\theta) \quad b_4 = y - \frac{D_y}{2} + \frac{D_1}{2} - \frac{D}{2}\sin(\theta) \end{aligned} \quad . \tag{4.5}$$

The forward kinematics can be calculated directly using (4.4). Since there is no coupling between $q_1,\ldots,q_4$, given x, y, z and θ, one can solve for the actuator motions $q_1,\ldots,q_4$ individually using (4.4). However, due to x, y, z and θ coupling in (4.4), the forward kinematics problem is not trivial and numerical methods should be employed to solve for x, y, z and θ when only $q_1,\ldots,q_4$ are known. The following iterative Newton's method can be used :

$$\mathbf{x}_{n+1} = \mathbf{x}_n - \left(\left(\frac{\partial \mathbf{F}}{\partial \mathbf{x}} \right)^{-1} \mathbf{F} \right)_{\mathbf{x}_n, \boldsymbol{\varphi}} \tag{4.6}$$

where $\mathbf{x} = \begin{bmatrix} x & y & z & \theta \end{bmatrix}^T$, $\boldsymbol{\varphi} = \begin{bmatrix} \varphi_1 & \cdots & \varphi_4 \end{bmatrix}^T$ and $\mathbf{F} = \begin{bmatrix} f_1 & \cdots & f_4 \end{bmatrix}^T$.

4.2.2. Manipulator Jacobian

Jacobian is the mapping between the joint-space and work-space velocities. It can also be used to find the actuator forces that are equivalent of the external forces exerted to the end-effector. Since the only rotation of end-effector is about the fixed z-axis, the manipulator Jacobian is the same as the analytical Jacobian. It can be derived by taking the derivative of (4.3):

$$\begin{aligned} \frac{\partial \mathbf{F}}{\partial \mathbf{x}}\dot{\mathbf{x}} &= -\frac{\partial \mathbf{F}}{\partial \boldsymbol{\varphi}}\dot{\boldsymbol{\varphi}} \\ \dot{\mathbf{q}} &= \frac{\partial \mathbf{G}}{\partial \boldsymbol{\varphi}}\dot{\boldsymbol{\varphi}} \end{aligned} \Leftrightarrow \begin{aligned} \mathbf{J}_x\dot{\mathbf{x}} &= -\mathbf{J}_\varphi\dot{\boldsymbol{\varphi}} \\ \dot{\mathbf{q}} &= \mathbf{J}_q\dot{\boldsymbol{\varphi}} \end{aligned} \tag{4.7}$$

where $\mathbf{G} = \begin{bmatrix} g_1 & \cdots & g_4 \end{bmatrix}$. Using the above equations the Jacobian can be found as:

$$\mathbf{J} = -\mathbf{J}_x^{-1}\mathbf{J}_\varphi\mathbf{J}_q^{-1} = \left(\frac{\partial \mathbf{F}}{\partial \mathbf{x}}\right)^{-1}\frac{\partial \mathbf{F}}{\partial \boldsymbol{\varphi}}\left(\frac{\partial \mathbf{G}}{\partial \boldsymbol{\varphi}}\right)^{-1}. \tag{4.8}$$

4.3. Dynamics Model

Robot dynamics can be determined using Lagrange's method. In order to use Lagrange's method, one should define the kinetic and potential energy of the robot as a function of some generalized coordinates and their time derivatives. Here, the vector of the end-effector position coordinates and orientation $\mathbf{x} = \begin{bmatrix} x & y & z & \theta \end{bmatrix}^T$ is used as the vector of generalized coordinates.

4.3.1. Kinetic Energy

To find the kinetic energy, we must find velocities of the end-effector, Link1, Link2, Link3 and actuator of each kinematic chain. Velocity of the end effector is $\dot{\mathbf{X}}$ and velocity of the left and right Link3's are $\dot{\mathbf{X}} \pm \boldsymbol{\omega} \times \mathbf{E}_i$, where $\boldsymbol{\omega} = \dot{\theta}\mathbf{k}$ is the angular velocity of the end-effector. Hence, the combined kinetic energy of the end effector and the two Link3's is

$$KE_e = \frac{1}{2}\dot{\mathbf{x}}^T\mathbf{M}_e\dot{\mathbf{x}} \tag{4.9}$$

where

$$\mathbf{M}_e = \begin{bmatrix} (m_e + 2m_s)\mathbf{I}_{3\times 3} & \mathbf{0} \\ \mathbf{0} & I_e + \dfrac{m_s D^2}{2} \end{bmatrix},$$

m_e and m_s are masses of the end-effector and Link3 respectively, I_e is the end-effector polar moment of inertia and $\mathbf{I}_{3\times 3}$ is identity matrix. Taking the time derivative of (4.1), it is possible to show that the velocity of center of Link 2 is

$$(\mathbf{V}_L)_i = \frac{1}{2}\left(\dot{\mathbf{X}} - \boldsymbol{\omega}\times\mathbf{E}_i + \dot{\varphi}_i\mathbf{u}_i\times\mathbf{A}_i\right). \tag{4.10}$$

Assuming that the center of mass of each parallelogram is at its center, the kinetic energy of the four parallelograms can be written as

$$KE_L = \frac{1}{2}m_L\sum_{i=1}^{4}(\mathbf{V}_L)_i^T(\mathbf{V}_L)_i, \tag{4.11}.$$

where m_L is the mass of the parallelogram assembly. This kinetic energy can be simplified to

$$KE_L = \frac{1}{2}m_L\dot{\mathbf{x}}^T\left(\frac{1}{4}\mathbf{M}_r + \frac{A^2}{4}\mathbf{J}_x^T\mathbf{J}_\varphi^{-T}\mathbf{J}_\varphi^{-1}\mathbf{J}_x + \frac{A}{4}\left(\mathbf{J}_x^T\mathbf{J}_\varphi^{-T}\mathbf{J}_L^T + \mathbf{J}_L\mathbf{J}_\varphi^{-1}\mathbf{J}_x\right)\right)\dot{\mathbf{x}} = \frac{1}{2}\dot{\mathbf{x}}^T\mathbf{M}_L\dot{\mathbf{x}} \tag{4.12}$$

where

$$\mathbf{M}_r = \begin{bmatrix} \mathbf{I}_{3\times 3} & \mathbf{0} \\ \mathbf{0} & \dfrac{D^2}{4} \end{bmatrix}, \tag{4.13}$$

$$\mathbf{J}_L = \begin{bmatrix} s\alpha_1 s\varphi_1 & s\alpha_2 s\varphi_2 & s\alpha_3 s\varphi_3 & s\alpha_4 s\varphi_4 \\ -c\alpha_1 s\varphi_1 & -c\alpha_2 s\varphi_2 & -c\alpha_3 s\varphi_3 & -c\alpha_4 s\varphi_4 \\ c\varphi_1(c\alpha_1 + s\alpha_1) & c\varphi_2(c\alpha_2 + s\alpha_2) & c\varphi_3(c\alpha_3 + s\alpha_3) & c\varphi_4(c\alpha_4 + s\alpha_4) \\ Ds\varphi_1\dfrac{c(\alpha_1-\theta)}{2} & -Ds\varphi_2\dfrac{c(\alpha_2-\theta)}{2} & -Ds\varphi_3\dfrac{c(\alpha_3-\theta)}{2} & Ds\varphi_4\dfrac{c(\alpha_4-\theta)}{2} \end{bmatrix}, \tag{4.14}$$

and $s\theta$ and $c\theta$ represent $\sin(\theta)$ and $\cos(\theta)$ respectively. The kinetic energy of each Link1 and its extension is

$$KE_A = \frac{1}{2}\dot{\mathbf{x}}^T \left(\frac{m_A A^2 + m_C C^2}{4} + I_A + I_C \right) \mathbf{J}_x^T \mathbf{J}_\varphi^{-T} \mathbf{J}_\varphi^{-1} \mathbf{J}_x \dot{\mathbf{x}} = \frac{1}{2}\dot{\mathbf{x}}^T \mathbf{M}_A \dot{\mathbf{x}}, \tag{4.15}$$

where m_A and m_C are the masses and I_A and I_C are the moment of inertia of Link1 and its extension, respectively. Finally the kinetic energy of the four actuators can be calculated as

$$KE_a = \frac{1}{2} m_a \dot{\mathbf{x}}^T \mathbf{J}^T \mathbf{J} \dot{\mathbf{x}} = \frac{1}{2}\dot{\mathbf{x}}^T \mathbf{M}_a \dot{\mathbf{x}} \tag{4.16}$$

where m_a is the mass of each actuator thrust rod. Here, the small rotational velocity of the actuators is neglected. The total kinetic energy is now

$$KE = KE_e + KE_L + KE_A + KE_a = \frac{1}{2}\dot{\mathbf{x}}^T \left(\mathbf{M}_e + \mathbf{M}_L + \mathbf{M}_A + \mathbf{M}_a \right) \dot{\mathbf{x}} \tag{4.17}$$

4.3.2. Potential Energy

The end-effector and the two Link3's always have the same elevation of z. So, their potential energy can be written as

$$PE_e = \left(m_e + 2m_s \right) gz \tag{4.18}$$

where g is the acceleration of gravity. The potential energy of the parallelograms can be obtained from (4.1) as

$$PE_L = \frac{m_L}{2} \sum_{i=1}^{4} \left(\mathbf{A}_i + \mathbf{X}_i - \mathbf{E}_i - \mathbf{D}_i \right) \cdot \mathbf{g}, \tag{4.19}$$

where $\mathbf{g} = \begin{bmatrix} 0 & 0 & g \end{bmatrix}^T$. It can be rewritten as

$$PE_L = \frac{m_L g}{2} \sum_{i=1}^{4} \left\{ z - \sin\left(\varphi_i\right) \right\}. \tag{4.20}$$

The potential energy of the four Link1's and their extensions is

$$PE_A = \frac{g}{2} \sum_{i=1}^{4} \left(m_C C - m_A A \right) \sin\left(\varphi_i\right). \tag{4.21}$$

The potential energy of the four actuator thrust rods can be obtained using (4.2) as

$$PE_a = m_a g \sum_{i=1}^{4} \left(\mathbf{C}_i + \frac{L_a}{2} \frac{\mathbf{B}_i - \mathbf{C}_i}{\left\| \mathbf{B}_i - \mathbf{C}_i \right\|} \right) \cdot \mathbf{g}, \tag{4.22}$$

that can be simplified to

$$PE_a = m_a g \sum_{i=1}^{4} \left(C \sin\left(\varphi_i\right) + \frac{L_a \left(b - C \sin\left(\varphi_i\right) \right)}{2\sqrt{\left(b - C \sin\left(\varphi_i\right) \right)^2 + \left(a + C \cos\left(\varphi_i\right) \right)^2}} \right) \tag{4.23}$$

where La is the length of the actuator thrust rod. The total potential energy is

$$PE_a = PE_e + PE_L + PE_A + PE_a \tag{4.24}$$

4.3.3. Lagrange Equation

The Lagrangian of a multi-body mechanical system is defined as

$$\mathcal{L} = KE - PE \tag{4.25}$$

For an n-degrees-of-freedom holonomic system with a vector of generalized coordinates $\mathbf{x}_{n \times n}$, the equations of motion are

$$\frac{d}{dt}\left(\frac{\partial \mathcal{L}}{\partial \dot{x}_i} \right) + \frac{\partial \mathcal{L}}{\partial x_i} = U_i, \tag{4.26}$$

where U is the vector of generalized forces. Using the above equation, one can show that robot's equations of motion have the following form:

$$\mathbf{M}(\mathbf{x})\ddot{\mathbf{x}}+\mathbf{C}(\mathbf{x},\dot{\mathbf{x}})\dot{\mathbf{x}}+\mathbf{N}(\mathbf{x})=\mathbf{F}_x+\mathbf{J}^{-T}\mathbf{F}_q \tag{4.27}$$

where

$$\mathbf{M}=\mathbf{M}_e+\mathbf{M}_L+\mathbf{M}_A+\mathbf{M}_a\,, \tag{4.28}$$

$$\mathbf{C}_{ij}=\frac{1}{2}\sum_{k=1}^{4}\left(\frac{\partial\mathbf{M}_{ij}}{\partial\mathbf{x}_k}+\frac{\partial\mathbf{M}_{ik}}{\partial\mathbf{x}_j}-\frac{\partial\mathbf{M}_{kj}}{\partial\mathbf{x}_i}\right)\dot{\mathbf{x}}_k\,, \tag{4.29}$$

and

$$\mathbf{N}=\frac{\partial PE}{\partial\mathbf{x}}\,. \tag{4.30}$$

4.4. Optimization

The structural dimensions of the robot can be determined by finding the solution of a constrained optimization problem. The optimization parameters are listed in Table 4.1. Throughout this chapter the robot is assumed to be fully symmetrical with $\alpha_1=-\pi/4$, $\alpha_2=\pi/4$, $\alpha_3=3\pi/4$, $\alpha_4=5\pi/4$ and $D_x=D_y$. The cost function is defined considering the following four criteria:

- Force isotropy
- Mechanical advantage
- Workspace size
- Counter-balancing

The values of robot parameter that result in a good performance in terms of the above criteria, that at the same time satisfy a set of geometric constraints can be used to build the robot. Both the cost function and the constraints will be discussed in the following sections.

Table 4.2. Optimization variables.

Parameter	Description
L	Parallelogram length
D	End-effector length
D_1	Link3 length
D_2	Link3 width
D_x	Horizontal distance between actuator
A	Link1 length
C	Link1 extension length
a	Actuator horizontal offset from Link1 pivot
b	Actuator vertical offset from Link1 pivot
z_c	z of the center of the work-space
W_x	Work-space width
W_y	Work-space length
W_z	Work-space height

4.4.1. Cost Function

A robot configuration is called isotropic if it can develop the same amount of force or velocity in all work-space directions in that configuration. In configurations that are close to robot singularities, the robot loses isotropy. At singular configurations, the robot cannot develop any force/velocity in certain directions. It means that singular configurations are not isotropic. A good measure of workspace isotropy is condition number of the Jacobian. It is defined as

$$k = \left\|\mathbf{J}\right\|\left\|\mathbf{J}^{-1}\right\| = \frac{\sigma_{\max}\left(\mathbf{J}\right)}{\sigma_{\min}\left(\mathbf{J}\right)} \tag{4.31}$$

where $\|.\|$ represents matrix induced 2-norm and $\sigma_{\max}$ and $\sigma_{\min}$ are the maximum and minimum singular values of $\mathbf{J}$. Condition number is always positive and greater than one, with one meaning fully isotropic and larger numbers meaning less isotropy. As mentioned above, each configuration or robot pose has a condition number. An optimized robot has a small condition number almost everywhere in its active workspace. Therefore, one can use the average of condition number as a measure of a good robot isotropy. Choosing N points in the active workspace of the robot, we can calculate the average condition number as

$$K = \frac{1}{N}\sum_{i=1}^{N} k(\mathbf{x}_i). \tag{4.32}$$

Another criterion considered here is the robots mechanical advantage. The actuators have limited force/torque capacity and this capacity can be very low in highly backdrivable actuators. So, the robot should have a high mechanical advantage. The work-space to join-space force mapping is described by the robot Jacobian as

$$\mathbf{F}_q = \mathbf{J}^T \boldsymbol{\tau}_x \tag{4.33}$$

where $\mathbf{F}_q = \begin{bmatrix} F_1 & F_2 & F_3 & F_4 \end{bmatrix}^T$ is the vector of actuator forces and $\boldsymbol{\tau}_x = \begin{bmatrix} F_x & F_y & F_z & \tau_z \end{bmatrix}^T$ is the vector of end-effector forces in x, y and z direction and torque about z axis. The highest mechanical advantage at any robot configuration is achieved when the ratio of the output force to the input force is maximized:

$$\max_{\mathbf{F}_q} \frac{\|\tau_x\|}{\|\mathbf{F}_q\|} = \max_{\mathbf{F}_q} \frac{\|\mathbf{J}^{-T}\mathbf{F}_q\|}{\|\mathbf{F}_q\|} = \|\mathbf{J}^{-T}\|. \tag{4.34}$$

So, the optimization problem should maximize the average 2-norm of the inverse Jacobian. It is well-known that size of the workspace has an inverse relationship with mechanical

advantage of the robot. So, if only mechanical advantage is used as cost function, the resulting robot will have a very small usable workspace. To prevent this situation, the optimization cost function should have a work-space volume term. Now, the solution will be a trade-off between work-space size and the mechanical advantage. Finding the workspace boundary is not simple and calculating the volume of entire workspace can be computationally expensive. So, the calculations are carried out over a cubic subspace of the workspace with width W_x and W_y in x and y directions and W_z in z direction. Also, the desired end-effector angle bounds is chosen to be $\theta \in \left[-\frac{\pi}{4}, \frac{\pi}{4} \right]$. Due to robot symmetry, the center of this designated work-space is constrained to lie at $x = y = 0, z = z_c$.

The robot should not fall off in case it turns off unintentionally. Also, it would be very beneficial in terms of actuator power if the robot can counter-balance some of its own weight. The optimization cost function includes a term to improve the counter-balance ability of the robot. Under no actuator force, the robot goes to an equilibrium state where robot weight is balanced. The distance from the location of end-effector at equilibrium and the center of workspace can be used as a function that needs to be minimized. Due to symmetry of the robot, the equilibrium lies on the line of $x = y = \theta = 0$. Now, using (4.30), it is possible to find the equivalent weight of the robot as a function of z at $x = y = \theta = 0$. To parameterize this equivalent weight function, the mass of links can be written as a function of the parameters listed in Table 4.2. The masses are

$$\begin{aligned} m_e &= \rho A_e D \\ m_s &= \rho t_s D_1 D_2 \\ m_L &= 2\rho\pi r_L^2 L \\ m_A &= \rho A_A A \\ m_C &= \rho_C A_C C \end{aligned} \tag{4.35}$$

where ρ is the density of aluminum, r_L is the radius of each parallelogram link, t_s is the thickness of link3, A_e, A_A and A_C are the cross section areas of the end-effector, Link1 and its extension, respectively. Now, the equivalent weight can be calculated as

$$w_e(z) = \mathbf{N}_3\big|_{x=y=\theta=0}, \tag{4.36}$$

where $\mathbf{N}$ is the same as (4.30). The equilibrium is at the point z^* where $w_e(z^*) = 0$. The optimization problem cost function then can be expressed as

$$H = \frac{1}{N}\sum_{i=1}^{N}\left(C_1 k_i + C_2 \|\mathbf{J}\|_i\right) + \frac{C_3}{W_x W_y W_z} + C_4\left(z_c - z^*\right)^2, \tag{4.37}$$

where C_1, C_2, C_3 and C_4 are constant weights corresponding to isotropy, mechanical advantage, work-space volume and counter-balance, respectively.

4.4.2. Constraints

The upper and lower limits of the optimization variables are listed in the right column of Table 4.3. These constraints are either due to geometric or manufacturability limitations.

The linear constraints are listed in the middle column of Table 4.3. The first two constraints are defined in order to avoid results in which the end-effector is larger than the base. The third constraint prevents possibility of the Link1 extensions to hit each other. The forth

constraint is imposed to keep the entire cubic work-space below the robot. The remaining linear constraints are required for reasons such as manufacturability.

There are also some nonlinear constraints that need to be met that are listed in the left column of Table 4.3. Due to mechanical limitations, the angles $\varphi_{1\ldots4}$ should be larger than zero and smaller than $\frac{\pi}{2}$ everywhere in the designated workspace. The minimum and maximum of $\varphi_{1\ldots4}$ can be found intuitively. For example, φ_1 is minimum when the end-effector is at $x=-W_x/2$, $y=-W_y/2$, $z=z_c+W_z/2$ and $\theta=\pi/4$, and is maximum when the end-effector is at $x=W_x/2$, $y=W_y/2$, $z=z_c-W_z/2$ and $\theta=-\pi/4$. So, the first two nonlinear constraints oversee these minimum and maximum angles. Other than software safety features, it is very important for the robot to have mechanical safety features that prevent the robot hitting the user. One such limitation can be achieved by constraining the configuration in which the end-effector is closest to the user, i.e. y is minimum. It is possible to show that

$$y_{\min}=-\sqrt{L^2-\left(D_x-D\right)^2}-D_1+\frac{D_x}{2}. \tag{4.38}$$

The next nonlinear constraint, therefore, can be defined as $y_{\min}>-0.5m$.

Since the linear actuators have a limited stroke, a constraint is needed to guarantee that this limitation will not be violated anywhere in the designated work-space. The motion of the i^{th} actuator is maximum when $\varphi_i=0$ and is minimum when $\varphi_i=\frac{\pi}{2}$. So, the required stroke can be calculated as

$$S=\sqrt{\left(a+C\right)^2+b^2}-\sqrt{a^2+\left(b-C\right)^2}\,. \tag{4.39}$$

Therefore, the next nonlinear constraint is $S \le S_{act}$, where S_{act} is the stroke of the linear actuators.

The elevation of the designated work-space cannot be so low that the robot cannot reach. The end-effector reaches its lowest configuration when $\varphi_i = \frac{\pi}{2}$, $i = 1,\ldots,4$, and the minimum elevation is

$$z_{\min} = -A - \sqrt{L^2 - \left(\frac{D}{2} + D_2\right)^2 - \left(\frac{D_1}{2}\right)^2} \tag{4.40}$$

The lowest point in the designated work-space is $z_c - \frac{W_z}{2}$. Therefore, the next nonlinear constraint can be defined as $z_{\min} \le z_c - \frac{W_z}{2}$. The next two constraints are defined so that the robot has good counter-balancing behavior: The equilibrium should lie somewhere in the designated work-space, so, $\left|z_c - z^*\right| \le \frac{W_z}{2}$. Moreover, there might be a solution with two equilibrium points where the second one is higher than the desired equilibrium. This second equilibrium might act as an attractor and push the robot up. To avoid the optimization program to reach to such a solution, we should constrain the robot to always have one equilibrium point. It is possible to show that always

$$w_e\left(z_{\min}\right) < 0 . \tag{4.41}$$

Since the equilibrium is the point where $w_e\left(z\right) = 0$, adding the constraint $w_e\left(z_{\max}\right) > 0$ guarantees existence of only one equilibrium. $z_{\max}$ is the highest elevation of the end-effector and can be found to be:

$$z_{\max} = -\sqrt{L^2 - \left(A\frac{\sqrt{2}}{2} - \frac{D}{2} - D_2\right)^2 - \left(A\frac{\sqrt{2}}{2} - \frac{D_1}{2}\right)^2} \tag{4.42}$$

Table 4.3. Optimization constraints.

Bound constraints	Linear constraints	Nonlinear constraints
$0 < L < 1$	$D + D_2 < D_x$	$0 < \varphi_i < \frac{\pi}{2}$
$0.1 < D < 1$	$D_1 < D_x$	$y_{\min} < -0.5$
$0.1\sqrt{2} < D_1 < 1$	$C\sqrt{2} < D_x$	$S \le S_{act}$
$D_2 < 1$	$z_c < -\frac{W_z}{2}$	$\sqrt{(C+a)^2 + b^2} < 0.42$
$0 < D_x < 1$	$A < L$	$\left\|z_c - z^*\right\| \le \frac{W_z}{2}$
$0 < A < 1$	$C < b$	$w_e\left(z_{\min}\right) < 0$
$0 < C < 1$	$-2D_2 < D$	
$a < 0.5$	$-a < C$	
$0 < b < 0.5$		
$-1 < z_c < 0$		
$0.4 < W_x < 1$		
$0.4 < W_y < 1$		
$0.3 < W_z < 1$		

4.4.3. Optimization Method and Results

The uniform grid spacing used to calculate (4.37), was 4 cm in the x, y, and z directions and $\frac{\pi}{8}$ in θ direction. The kinematics, cost function and all the constraints were programmed in Matlab® and the "fmincon" function was employed to solve the problem using the interior point method [112]. The second column of Table 4.4 shows the admissible initial values and the third column shows the optimized results. The constants used in (4.37) were $C_1 = C_2 = 1$,

$C_3 = 20$ and $C_4 = 100$. The value of C_3 is chosen to normalize the portion of the cost function pertaining to the designated work-space volume to the minimum admitted volume of $0.4cm \times 0.4cm \times 0.3cm$. Similarly, the value of C_4 is chosen to normalize the distance between the workspace center to equilibrium point to 10 cm.

Figure 4.7 shows the Jacobian condition number values (top) and the Jacobian norm (bottom) in the designated workspace for $\theta = 0$. The four lower corners of the designated work-space have the highest condition number and norm. These two values, however, are uniform on the rest of the work-space.

Figure 4.8 shows the equivalent weight of the end effector as a function of z. The distance between z_c and the equilibrium is 24.84 cm and it is located at the lowest level of the designated work-space.

Table 4.4. The initial and optimized variables.

Parameter	Initial Value (m)	Optimized Value (m)
L	0.5050	0.5899
D	0.1524	0.1090
D_1	0.1614	0.1414
D_2	0.0050	0.2959
D_x	0.3048	0.4049
A	0.400	0.5672
C	0.1778	0.1869
a	0.0762	0.0371
b	0.2908	0.3620
z_c	-0.6000	-0.7566
W_x	0.4000	0.5666
W_y	0.4000	0.400
W_z	0.3000	0.4968

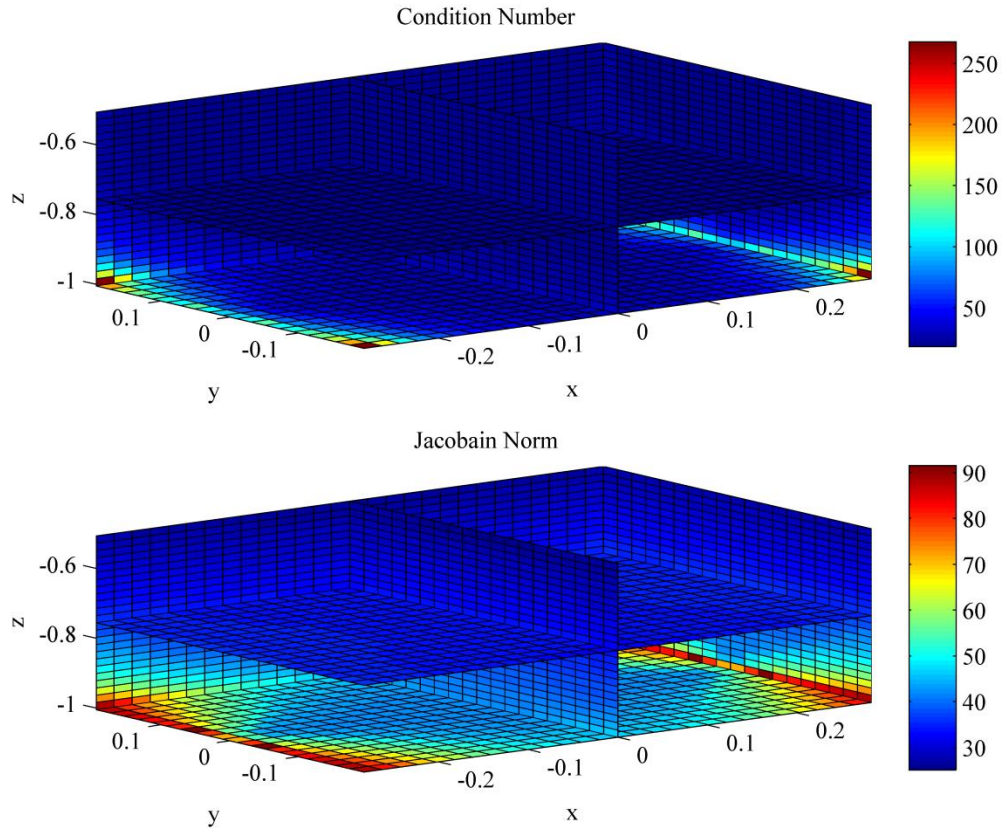

Figure 4.7. The condition number (top) and norm of Jacobian matrix throughout the designated work-space at $\theta = 0$.

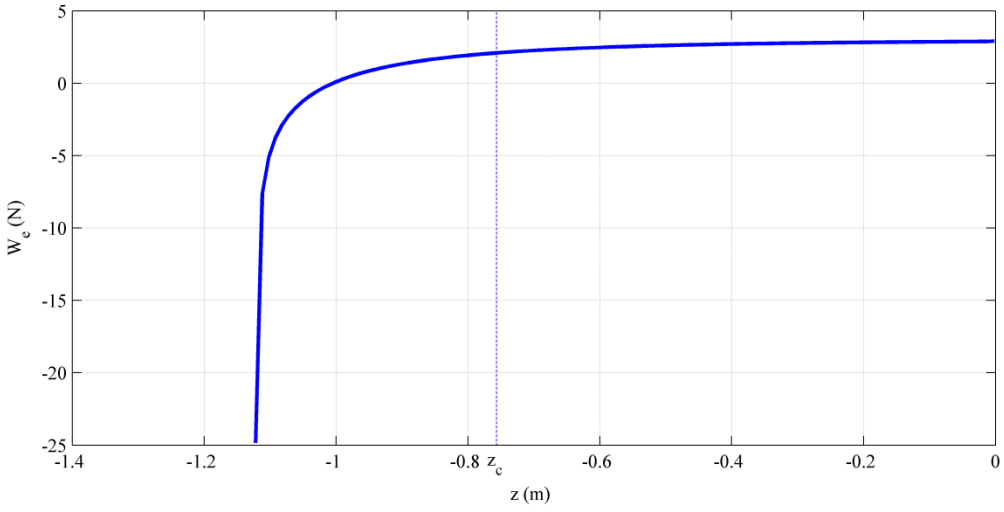

Figure 4.8. Equivalent weight w_e as a function of robot elevation z. The height of center of the designated work-space z_c is shown by a dashed line.

4.5. Velocity Estimation

Many robotic rehabilitation strategies require a smooth online velocity signal for their control systems to operate. The simplest method that can be used to find the end-effector velocity of the robot is to take numerical time derivative of the actuator position signals and use (4.7) and (4.8) to obtain the end-effector velocity. Although the actuators used in this robot can output a very high resolution position signal, direct time derivate of these signals is prone to noise due to the measurement and discretization errors. An extended Kalman-filter (EKF) can be used to obtain fine and dependable velocity signals. This EKF works based on the relationship between position, velocity and acceleration, as well as the robot kinematics. The input to the EKF can be obtained from an inertial measurement unit (IMU) installed at the center of the end-effector. IMU's, however, are expensive and we can use two simple analog 3-zxis accelerometers installed on the two Link3's so that the z-axis of the accelerometers are along the rotation axis of the Link3 revolute joints as shown in Figure 4.6. The position vector of the left and right accelerometers are $\mathbf{x}_L = \begin{bmatrix} x_L & y_L & z_L \end{bmatrix}^T$ and $\mathbf{x}_R = \begin{bmatrix} x_R & y_R & z_R \end{bmatrix}^T$, respectively.

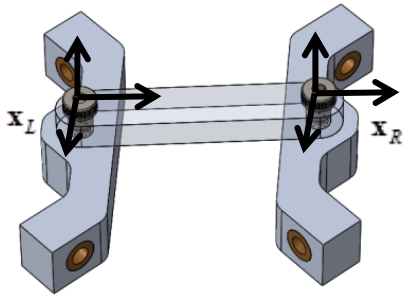

Figure 4.9. The location of the 3-axis accelerometers mounted on the two Link3's.

Using the kinematics of solid bodies,

$$\begin{aligned} \dot{\mathbf{x}}_L &= \dot{\mathbf{x}} - \boldsymbol{\omega} \times \mathbf{E}_1 \\ \dot{\mathbf{x}}_R &= \dot{\mathbf{x}} - \boldsymbol{\omega} \times \mathbf{E}_2 \end{aligned} \tag{4.43}$$

Where $\boldsymbol{\omega}$ is the angular velocity vector of the end-effector and $\mathbf{x}$, $\mathbf{E}_1$ and $\mathbf{E}_2$ are shown in Figure 4.6. As mentioned before, the two Link3's don't rotate and hence the two accelerometers only have a translational motion. Since $\mathbf{E}_2 = -\mathbf{E}_1$, adding and subtracting (4.43) yields:

$$\begin{aligned} \dot{\mathbf{x}} &= \frac{\dot{\mathbf{x}}_L + \dot{\mathbf{x}}_R}{2} \\ \boldsymbol{\omega} \times \mathbf{E}_1 &= \frac{\dot{\mathbf{x}}_L - \dot{\mathbf{x}}_R}{2} \end{aligned} \tag{4.44}$$

The above equations can be simplified to

$$\begin{aligned} \dot{x} &= \frac{\dot{x}_L + \dot{x}_R}{2} \\ \dot{y} &= \frac{\dot{y}_L + \dot{y}_R}{2} \\ \dot{z} &= \frac{\dot{z}_L + \dot{z}_R}{2} \\ \dot{\theta} &= \frac{1}{D}\left(\left(\dot{y}_R - \dot{y}_L\right)\cos\left(\theta\right) + \left(\dot{x}_L - \dot{x}_R\right)\sin\left(\theta\right)\right). \end{aligned} \tag{4.45}$$

The relationship between the acceleration and output voltage of the accelerometers is:

$$a_i = \alpha V_i + b_i \tag{4.46}$$

Where a_i is the acceleration along the i^{th} axis relative to ground, V_i is the output voltage of the i^{th} channel, α is the accelerometer sensitivity and b_i is a constant or slowly varying bias. Assuming process and measurement noise covariance are $\mathbf{Q}(t)$ and $\mathbf{R}(t)$, respectively, and defining the estimation state vector and input vector as

$$\begin{aligned} \mathbf{s} &= \begin{bmatrix} b_1 & \cdots & b_6 & x & y & z & \theta & \dot{x} & \dot{y} & \dot{z} & \dot{\theta} \end{bmatrix}^T \\ \mathbf{u} &= \begin{bmatrix} V_1 & \cdots & V_6 \end{bmatrix}^T \end{aligned}, \tag{4.47}$$

the process dynamics can be found as

$$\begin{cases} \dot{\mathbf{s}} = \boldsymbol{f}\left(\mathbf{s}(t), \mathbf{u}(t)\right) + \mathbf{w}(t) \\ \mathbf{y} = \boldsymbol{h}\left(\mathbf{s}(t)\right) + \mathbf{v}(t) \end{cases}, \tag{4.48}$$

where $\mathbf{w}(t) \sim N\left(\mathbf{0}, \mathbf{Q}(t)\right)$ and $\mathbf{v}(t) \sim N\left(\mathbf{0}, \mathbf{R}(t)\right)$ and $\mathbf{y}$, i.e. EKF output, is the vector of the actuator position measurements. Taking the time derivative of (4.45) and using (4.46) and (4.48), the function $\boldsymbol{f}\left(\mathbf{s}(t), \mathbf{u}(t)\right)$ can be found as

$$\boldsymbol{f}\left(\mathbf{s}(t), \mathbf{u}(t)\right) = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ \dot{x} \\ \dot{y} \\ \dot{z} \\ \dot{\theta} \\ \dfrac{\alpha\left(V_1 + V_4\right) + b_1 + b_4}{2} \\ \dfrac{\alpha\left(V_2 + V_5\right) + b_2 + b_5}{2} \\ \dfrac{\alpha\left(V_1 + V_4\right) + b_1 + b_4}{2} \\ \dfrac{\alpha\left(\cos(\theta)\left(V_2 + b_2 - V_5 - b_5\right) + \sin(\theta)\left(V_4 + b_4 - V_1 - b_1\right)\right)}{D} \end{bmatrix}, \tag{4.49}$$

And $\boldsymbol{h}$ is the inverse kinematic equation relating the end-effector position and orientation to actuator positions as given by (4.4). The EKF predict-update law is

$$\begin{aligned} \dot{\hat{\mathbf{s}}}(t) &= \boldsymbol{f}\left(\hat{\mathbf{s}}(t), \mathbf{u}(t)\right) + \mathbf{K}(t)\left(\mathbf{y}(t) - \boldsymbol{h}(\hat{\mathbf{s}})\right) \\ \dot{\mathbf{P}} &= \mathbf{F}(t)\mathbf{P}(t) + \mathbf{P}(t)\mathbf{F}(t)^T - \mathbf{K}(t)\mathbf{H}(t)\mathbf{P}(t) + \mathbf{Q} \\ \mathbf{K}(t) &= \mathbf{P}(t)\mathbf{H}(t)^T \mathbf{R}(t)^{-1} \end{aligned} \tag{4.50}$$

where

$$\mathbf{F}(t) = \left.\frac{\partial \boldsymbol{f}}{\partial \mathbf{s}}\right|_{\hat{\mathbf{s}}(t),\mathbf{u}(t)} \qquad \mathbf{H}(t) = \left.\frac{\partial \boldsymbol{h}}{\partial \mathbf{s}}\right|_{\hat{\mathbf{s}}(t)} . \tag{4.51}$$

Chapter 5. Internal Model Based Adaptive Control for Robotic Movement Therapy after Stroke

To be published.

Abstract— This paper presents an adaptive control approach for robotic movement therapy that learns defects in patient's neuromuscular output and assists accordingly. Unlike previous control strategies, this approach uses a healthy internal model as a template for model-based movement therapy. The controller adaptively learns and fills the gaps in patient's ability to generate inertial forces impaired by stoke so the assistance received by the patient mimics missing motor control forces that would be generated by a healthy person. To test this method, a two dimensional model of a human arm with impaired neuromuscular control was simulated performing reaching movements in the horizontal plane. The results from simulation demonstrate the potential efficacy of the proposed controller.

5.1. Introduction

Using robotic devices for therapy after stroke has been a rapidly growing field of research during the last several decades. It has been demonstrated that rehabilitation robots can be effective if used in a consistent manner and their therapeutic efficiency is comparable to that of conventional therapy methods [3, 78, 79].

Recent research suggests that patient voluntary participation and effort is very important in enhancing motor plasticity [113]. Also, receiving excessive assistance may limit patient motor output [82], i.e. patients tend to let the robot take over completing task. On the other hand, if the robot's assistance level is too low, the patient may not be able to complete the desired motion. So, providing only the minimum assistance required for a specific therapy task is a key feature of a robotic rehabilitation device.

To address these issues, researchers have developed assist-as-needed (AAN) control strategies that limit the assistance level while maintaining task completion and engagement [15]. These control strategies include static [85] and adaptive approaches [2, 114, 115]. For a review of control strategies for robot assisted rehabilitation, see [15].

An internal model is a neural mechanism that predicts the outcome of motor commands [116]. In presence of biological feedback delay, existence of such a model is necessary for performing fast and coordinated movements. It has been suggested that motor learning cannot be achieved solely based on rote learning [117]. After neural injuries such as stroke, the existing internal model is no longer appropriate as it is not able to predict disturbances caused by damage. So, an important goal of robotic rehabilitation is re-establishing the internal model [117].

A passivity-based adaptive AAN controller was implemented in [75] that learned an unstructured model of patient impairment. The controller included a force decay term that reduced the impairment model when tracking errors were small, preventing the robot from "taking over" the desired motions [115]. One shortcoming of this approach was its inability to generate inertial forces. This limited the robot's ability to appropriately assist with fast and coordinated movements. It is possible to overcome this issue by increasing adaptation rate, but this results in higher robot stiffness and reduced compliance. An alternative solution was proposed in [105] that used separate models for each target motion.

In [118], a controller was proposed that included inertial terms as well as directional dependency in the unstructured model of impairment. The main shortcoming of the proposed controller was that it only allowed for single degree-of-freedom motions. Moreover, although the mass model used in this method was a function of position and

direction of motion, it was an incomplete model that was unable to accurately represent the inertial forces required for moving articulated limbs such as human arm.

In both [105] and [118], radial basis functions (RBF's) were used to form the unstructured model of a patient's impairment, i.e. their ability to generate forces. Using an unstructured model, as opposed to the typical serial-chain manipulator model structure used in adaptive control, is necessary due to the nature of impairment after stroke; each stroke is unique in and impaired motor control varies from patient to patient.

In this paper, we present an adaptive controller that learns a model of stroke-impaired human motor control. The impairment model includes representation of neuromuscular output for generating forces both statically (to overcome gravity, etc.) and dynamically (for creating limb acceleration). In the following sections, we describe the control algorithm and its mathematical basis. Next, several simulated planar reaching experiments are presented in order to validate the presented controller.

5.2. Methods

5.2.1. Motor Control Model

During robot-assisted movement therapy, the human limb and the robot become connected in some way, and as such have combined dynamics. In the case of an end-effector type upper-extremity robot, the connection point is the human hand. The physical system of hand and robot can be modeled as

$$M(x)\ddot{x}+C(x,\dot{x})\dot{x}+G(x,\dot{x})=f_h+f_r \tag{5.1}$$

where M and C are the system's generalized mass and centrifugal-Coriolis matrices, G is the gravitational and viscous force field of the system, f_r is the robot generalized force

vector and f_h is the force applied by the person's hand. Equation (5.1) is expressed in the hand coordinate system x, which is the normal earth-referenced Cartesian frame.

We can assume that the hand force is the sum of three components: a stabilizing feedback term, an internal model for inertial dynamics, and an internal model of the external environment [116, 119]. The resulting motor output is

$$f_h = \hat{M}(x)\ddot{x}^d + \hat{C}(x,\dot{x})\dot{x} + \hat{G}(x,\dot{x}) - K_p\tilde{x} - K_d\dot{\tilde{x}} \tag{5.2}$$

where $\hat{M}, \hat{C}$ and $\hat{G}$ are approximations of M, C and G of the hand with the same mathematical structure, $\ddot{x}^d$ is the desired hand trajectory, $\tilde{x}$ is the position error and K_p and K_d are stabilizing proportional-derivative (PD) gains. Ideally, for a healthy person, the aforementioned approximations converge to the actual matrices during motor learning, i.e, $\hat{M} \to M$, $\hat{C} \to C$ and $\hat{G} \to G$ so that

$$f_h = M(x)\ddot{x}^d + C(x,\dot{x})\dot{x}^d + G(x,\dot{x}) - K_p\tilde{x} - K_d\dot{\tilde{x}} \tag{5.3}$$

Stability of this model can be verified by substituting (5.3) in (5.1) and assuming no robot control force. This model can be extended to represent impairment by distorting the parameters of the model.

It is well known that the model in (5.3) is linear in its parameters [120] and can be written as

$$M(x)\ddot{x} + C(x,\dot{x})\dot{x} + G(x,\dot{x}) = Y(x,\dot{x},\ddot{x})\theta \tag{5.4}$$

where Y is a regressor matrix with a known structure and θ is a vector containing parameters. Similarly, the first three terms on the right hand side of (5.3) can be rewritten as

$$M(x)\ddot{x}^d + C(x,\dot{x})\dot{x}^d + G(x,\dot{x}) = Y(x,\dot{x},\dot{x}^d,\ddot{x}^d)\theta^*, \tag{5.5}$$

where θ^* is the parameters estimated by the internal model. For a healthy person, one can assume that there are only modest limitation on θ^* in (5.5) so that it can converge close to θ, so that a healthy person can reasonably estimate the dynamics of their environment in order to achieve a functional internal model. Note that viscous, elastic or other types of environments could be included in the model, but without loss of generality, only inertial effects of environment are considered. For instance, this model includes learning inertial dynamics of a robot connected to hand or heavy objects manipulated by person.

Impaired people, on the other hand, are more severely limited in their ability to estimate the internal model parameters. This can be modeled by imposing limitation on θ^* in (5.5). This limitation, for instance, can be a modeled as a set of constraints on individual parameter magnitudes. These limitations can be a function of hand position and/or velocity to represent patients' lack of control of limb motions in certain areas of space or certain directions.

Considering the deficiencies mentioned above, the motor controller can be modeled as

$$\begin{aligned} f_h &= \left(M - M_d\right)\ddot{x}^d + \left(C - C_d\right)\dot{x} + \left(G - G_d\right) \\ -K_p\tilde{x} - K_d\dot{\tilde{x}} &= Y\left(x, \dot{x}, \dot{x}^d, \ddot{x}^d\right)\theta_d, \end{aligned} \tag{5.6}$$

where M_d, C_d, G_d represent the deficiencies in mass, Coriolis and gravity components of the internal model and θ_d is the vector of limited estimated parameters. An appropriate robot controller can have the following form:

$$\begin{aligned} f_r &= \hat{M}_d\ddot{x}^d + \hat{C}_d\dot{x} + \hat{G}_d - K_{rp}\tilde{x} - K_{rd}\dot{\tilde{x}} \\ &= Y\widehat{\theta}_d - K_{rp}\tilde{x} - K_{rd}\dot{\tilde{x}} \end{aligned} \tag{5.7}$$

where $\hat{M}_d, \hat{C}_d$ and $\hat{G}_d$ are deficiency estimates and K_{rp} and K_{rd} are the stabilizing proportional and derivative gain matrices. Substituting (5.6) and (5.7) into (5.1) yields the following closed loop dynamics

$$M\ddot{\tilde{x}} + C\dot{\tilde{x}} = \tilde{M}_d \ddot{x}^d + \tilde{C}_d \dot{x} + \tilde{G}_d - \left(K_p + K_{rp}\right)\tilde{x} - \left(K_d + K_{rd}\right)\dot{\tilde{x}}, \tag{5.8}$$

where $\tilde{M}_d$, $\tilde{C}_d$, $\tilde{G}_d$ are estimation errors and $\tilde{\theta}_d$ is the vector of internal model parameter estimation errors. The above dynamics can be shown to be linear in parameter and can written in the form

$$M\ddot{\tilde{x}} + C\dot{\tilde{x}} + \left(K_p + K_{rp}\right)\tilde{x} + \left(K_d + K_{rd}\right)\dot{\tilde{x}} = Y\left(\ddot{x}^d, \dot{x}, x\right)\tilde{\theta}_d, \tag{5.9}$$

which is the suitable form for adaptive control design. For instance, one can use gradient adaptation law:

$$\dot{\hat{\theta}}_d = -\Gamma^{-1} Y^T s, \tag{5.10}$$

where Γ is the adaptation rate and is a positive definite matrix and $\dot{\tilde{x}} + \Lambda\tilde{x}$ is the sliding surface with Λ a diagonal matrix with positive diagonal entries. Furthermore, to increase compliance and voluntary effort, and make the controller AAN, a decay term similar to the one in [62] can be added to (5.10):

$$\dot{\hat{\theta}}_d = -\Gamma^{-1} Y^T s - \lambda^{-1} Y^T \left(YY^T\right)^{-1} Y\hat{\theta}, \tag{5.11}$$

where λ is the forgetting rate.

5.2.2. Unstructured Modeling

To implement the controller described above, one can use (5.7) with an explicit structure of the regressor matrix Y, obtained based on dynamics of the system (5.1). It is shown to be a

good model for predicting force fields generated by hand during reaching movements [121]. In the approach presented in this paper, however, an explicit structure is not used for two main reasons. Firstly, in a structured representation, internal model parameter vector θ is not a function of position and/or velocity that makes it hard to include position/velocity specific impairments in the model. The second reason is that structured representation of (5.2) is accurate if the robot is very light-weight and its dynamics can be easily modeled. If the robot has considerable inertia compared to hand, user's CNS forms an internal model of the robot's inertial forces and the structured representation of (5.2) will not be accurate anymore.

To create an unstructured model, we used radial basis functions (RBF). Using enough number of RBF's, it is possible to accurately approximate any function [122]. Let's start modeling by considering the symmetric $n \times n$ mass matrix M. Using R RBF's scattered appropriately in the space, each entry of M can be approximated as

$$M_{ij}(x) = \sum_{k=1}^{R} \phi_k(x) m_{ij,k} = \phi m_{ij} \tag{5.12}$$

where M_{ij} is the entry of M in the i^{th} row and j^{th} column, $m_{ij,k}$ is the k^{th} weight function corresponding to M_{ij}, $\phi_k(x)$ is the k^{th} radial basis function defined as

$$\phi_k(x) = \exp\left(-\frac{\|x - \mu_k\|^2}{2\sigma}\right) \tag{5.13}$$

where $\phi = [\phi_1 \quad \cdots \quad \phi_R]$, $m_{ij} = [m_{ij,1} \quad \cdots \quad m_{ij,R}]^T$, μ_k is the location of the k^{th} RBF and σ is the RBF dilation that is considered to be the same for all RBF's. Note that M is symmetric and i.e. $m_{ij} = m_{ji}$.

It is known that the centrifugal/Coriolis terms can be obtained as the product of the Christoffel symbol of the first kind of M and the generalized velocity [120]:

$$C_{ij} = \sum_{k=1}^{n} \frac{1}{2} \left\{ \frac{\partial M_{ij}}{\partial x_k} + \frac{\partial M_{ik}}{\partial x_j} - \frac{\partial M_{kj}}{\partial M_i} \right\} \dot{x}_k . \tag{5.14}$$

Substituting (5.5) into (5.7), C_{ij} can be rewritten as

$$C_{ij} = \frac{-1}{2\sigma} \phi \sum_{k=1}^{n} \left\{ L_k m_{ij} + L_j m_{ik} - L_i m_{jk} \right\} \dot{x}_k \tag{5.15}$$

where L_i is an $n \times n$ diagonal matrix defined as

$$\left(L_i \right)_{jk} = \begin{cases} x_i - \left(\mu_i \right)_j & , j = k \\ 0 & , j \neq k \end{cases} . \tag{5.16}$$

The same procedure can be applied to the gravitational matrix G as follows:

$$G_i \left(x \right) = \sum_{k=1}^{R} \phi_k \left(x \right) g_{ik} = \phi g_i \tag{5.17}$$

As mentioned before, it is assumed that M_d, C_d and G_d have the same structure as M, C and G, and only their parameters are limited, hence the estimates $\hat{M}_d$, $\hat{C}_d$ and $\hat{G}_d$ can be modeled using equations (5.12)-(5.17) and only by hatting the parameters. To find the regressor matrix Y, the above models can be substituted in (5.7):

$$\left(f_r \right)_i = \phi \left\{ \sum_{j=1}^{n} \left(\begin{array}{l} \left(\ddot{x}_j^d I_{R \times R} - \dfrac{1}{2\sigma} \displaystyle\sum_{k=1}^{n} L_k \dot{x}_j{}^d \dot{x}_k \right) \hat{m}_{ij} \\ + \dfrac{1}{2\sigma} \displaystyle\sum_{k=1}^{n} \dot{x}_k \left(L_i \dot{x}_j{}^d \hat{m}_{jk} - L_j \dot{x}_i{}^d \hat{m}_{ik} \right) \end{array} \right) + \hat{g}_i \right\} . \tag{5.18}$$

By factoring the $\hat{m}_{ij}$ and $\hat{g}_i$ terms and putting them in the parameter vector

$$\hat{\theta}_d = \left[\hat{m}_{11}^T \hat{m}_{12}^T \ldots \hat{m}_{1n}^T \hat{m}_{21}^T \hat{m}_{22}^T \ldots \hat{m}_{nn}^T \hat{g}_1^T \hat{g}_2^T \ldots \hat{g}_n^T \right]^T \tag{5.19}$$

the regressor matrix Y_i can be obtained.

5.2.3. Experimental Procedure

A two DOF arm was modelled to be capable of performing reaching movements to eight target points, each 15 cm from the center as shown in Figure 5.1. The motor control law was the same as (5.3), but was a function of joint space variables. Since motion is planar, gravitational model terms in (5.1), (5.3) and (5.18) were excluded. Center of mass of both upper arm and forearm was assumed to be in the middle of each segment. Simulation parameters are listed in Table 5.1. Both the controllers and the equations of motion of the 2 DOF arm were simulated in the Simulink® graphical programming environment.

The home position of desired trajectories is 0.35 m in front of the shoulder and the hand reaches to targets each 0.15 m in eight different directions as shown by blue lines in Figure 5.1. The desired trajectories are simple minimum jerk reaches with 1 sec duration. Impairment is simulated by multiplying the unimpaired motor control model (5.3) by a weight function that gradually varies from 0.1 to 1 as a function of hand position. A zero-mean band limited Gaussian noise with variance of 0.001 N.m was also added to the calculated joint torques in the impaired motor control model. The gray area in Figure 5.1 shows the impairment region.

During simulated experiments, the arm reaches to each of the 8 target points in a randomized order.

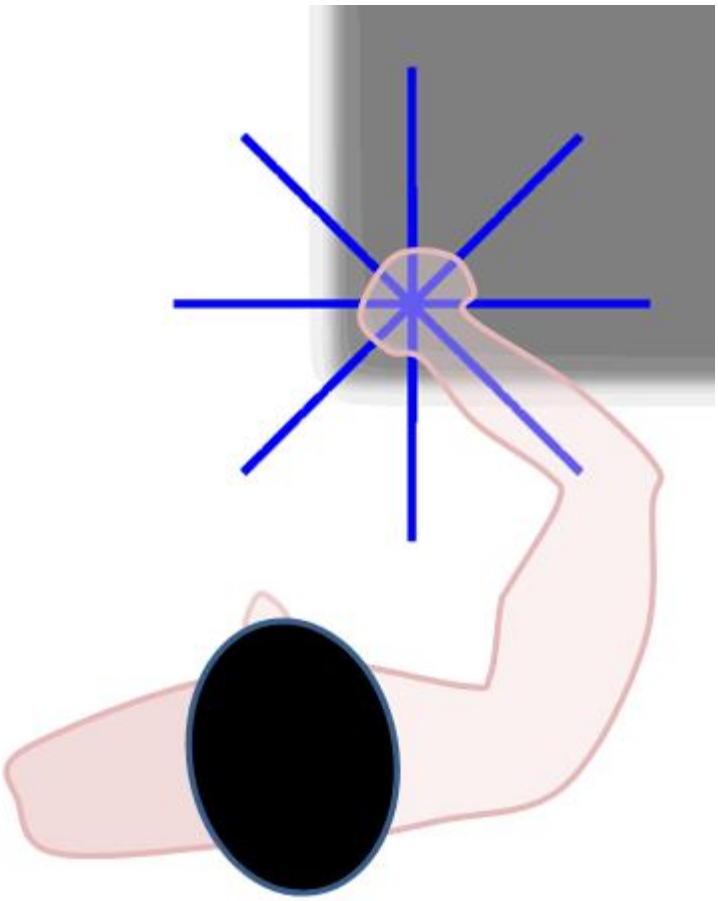

Figure 5.1. Illustration of the simulated reaching motions. The gray area is where impairment exists in the model.

Table 5.1. Arm simulation parameters.

	Upper arm	**Forearm**
Mass	2 kg	1.5 kg
Moment of inertia	0.015 $kg.m^2$	0.018 $kg.m^2$
Length	0.33 m	0.35 m
Stiffness $\begin{bmatrix} 15 & 6 \\ 6 & 15 \end{bmatrix}$ N.m/rad	**Viscosity** $\begin{bmatrix} 2.5 & 1 \\ 1 & 2.5 \end{bmatrix}$ N.m.s/rad	

5.3. Results

The results of reaching with un-impaired and impaired motor control models is shown in Figure 5.2. The effect of impairment on the accuracy of motion is clearly visible in the affected grey area.

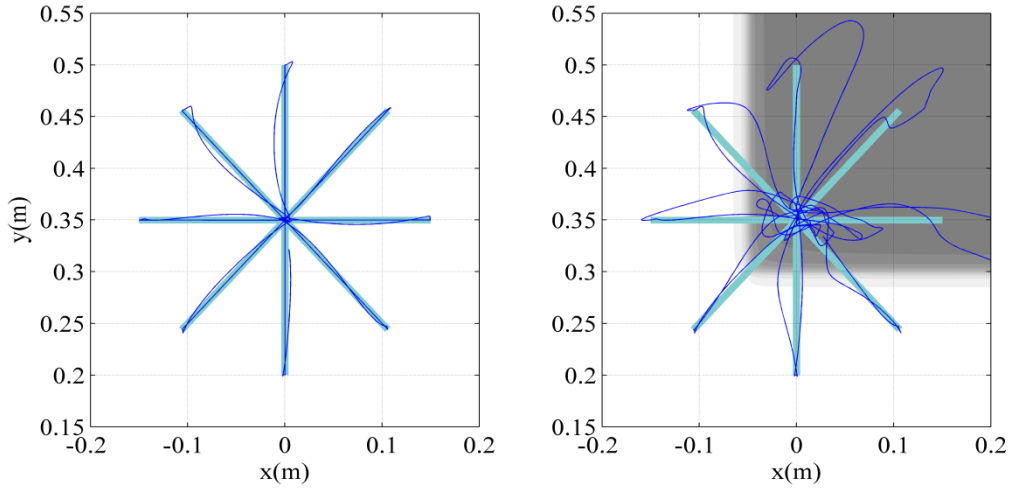

Figure 5.2. Performance of simulated unimpaired (left) and partially impaired motor control (right). Gray region represents impairment region.

Figure 5.3 shows results of the proposed controller applied to the simulated impaired arm for different values of learning time constant $\Gamma = diag(\tau)$. In all simulations, K_{rp} and K_{rd} were $diag(18)\, N.m^{-1}$ and $diag(6)\, N.s.m^{-1}$, respectively.

Performance of the proposed controller was compared to that of the adaptive control presented in [62] that doesn't include explicit learning of inertial effects. All the controller parameters are the same and are $K_{rp} = diag\left(1.8\right) N.m^{-1}$, $K_{rd} = diag\left(0.3\right) N.s.m^{-1}$ and $\Gamma = diag\left(10\right)$ s. The gains are chosen very small to compare performance of the two controllers with a very low effective stiffness. The results are shown in Figure 5.4.

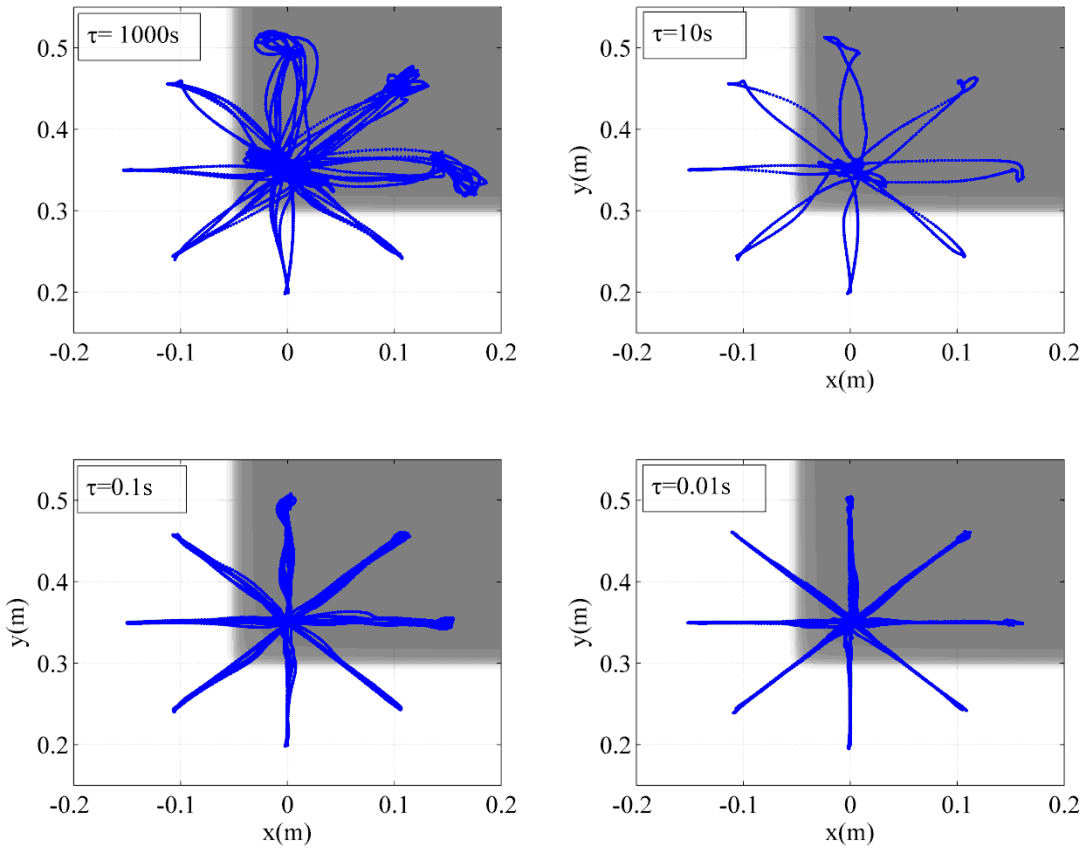

Figure 5.3. Simulation results of the controller assisting the impaired arm with different learning rates.

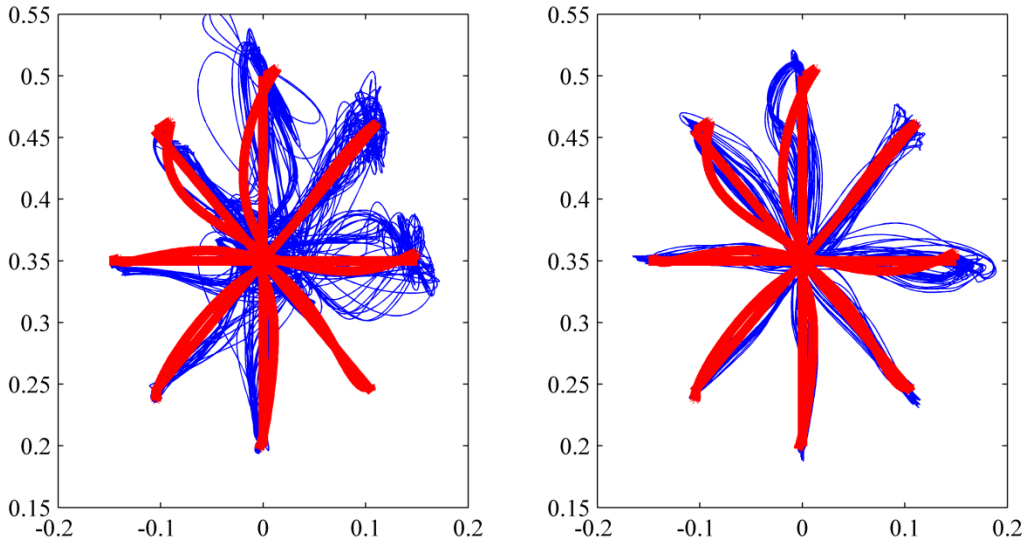

Figure 5.4. Simulation results of the proposed controller (right) and the non-inertial AAN [62] (left). The red line shows the simulation of the healthy motor performance (5.3).

5.4. Conclusion

This paper presents an adaptive control algorithm for robotic movement therapy that includes inertia learning components. By including these inertial terms, a controller structure was formed that has a mathematical structure similar to mathematical structure of the internal model that CNS uses to generate forces required for fast and coordinated notions. A computer model of arm and motor control that was impaired in a certain region was used to test the proposed controller. It can be verified that this model works as predicted and since it is capable of learning inertial forces, it is capable of assisting patients during fast movements without increasing its apparent stiffness.

Chapter 6. Discussion and Conclusions

Various aspects of this research had been discussed in previous chapters. In this chapter, the main concepts, contributions and conclusions of this research are discussed. New devices and adaptive control methods for robotic rehabilitation after stroke were developed. This research includes the following major works: development of a single DOF finger rehabilitation robot FINGER, implementation and results of an adaptive run-to-run control to modulate patient engagement, analyzing patient finger individuation ability after stroke, development of a state-dependent adaptive control for robotic rehabilitation therapy, development of a 4 DOF parallel robot for upper extremity movement therapy including modeling and optimization and finally development of an adaptive control for rehabilitation robotics that takes inertial effects into account.

6.1. FINGER Robot

FINGER was designed to have a very low friction and inertia and be very backdrivable to be suitable for implementing many rehabilitation therapy control architectures. The structural dimensions of the planar mechanism of FINGER were obtained by solving an optimization problem. This mechanism was an 8-bar mechanism and was optimized so that its output has the minimum tracking error relative to a predetermined desired trajectory obtained from motion capture of natural motions of multiple healthy subjects. Also, many constraints were imposed to guarantee safety and manufacturability of the robot. A graphical user interface was coupled with FINGER that was basically a video game similar to Guitar Hero®. To successfully hit each not, the patient should flex their finger corresponding to the steaming not to a certain location and at a certain time.

6.2. Modulating Success Rate

To keep the patient's engaged, it is very important to be able to modulate patients' level of engagement during therapy. One potential way of doing so, is by controlling the success rate through the level of assistance received from the robot and assuming that engagement is maximum a certain success rate. The success rate can be changed by making the task more difficult to do by the subjects, e.g. by imposing limited time constraints, or by decreasing the assistance level provided by the robot. In this research, the second method was used. In case of the game similar to Guitar Hero®, after each note, depending on if the patient's success in hitting the note, the controller gain was increased or decreased by a certain value that guarantees a final probability of success.

The above method was tested on multiple stroke patients with hemiparesis and its ability to modulate success rate was observed. Patient effort was also estimated by measuring the force generated by them at some intentionally blocked control notes. It was observed that patient effort has a reverse correlation with their success that was in accordance with motor learning theories. A similar method was used to quantify finger individuation ability and its correlation with impairment level was observed.

6.3. State-Dependent Adaptive Control

In Chapter 3, a state-dependent adaptive control was proposed and tested on FINGER. Both inertial effects and directional dependent impairment effects were modeled in the control system. To include directional dependency, two distributed mass models, corresponding to finger flexion and extension, were introduced to the model. It was observed that the controller adapts two these two models depending on severity of impairment at each

direction. In this experiment, healthy subjects were tested and artificial viscous fields were superimposed to the robot input to emulate impairment.

6.4. Parallel Robot for Upper Extremity Impairment

In Chapter 4, development of a 4 DOF parallel robot for hand and arm movement therapy was presented. This robot's architecture is very similar to that of Delta platform, but has an additional end-effector rotation about a fixed axis. It can be used for large translational motions of the hand and wrist flexion/extension. One of the main advantages of this robot is that it can be made very light weight and backdrivable. Also, it can be manufactured relatively easily because of its symmetry and that most of the joints are not actuated. Only the joints that are directly connected to the ground are actuated. It also increases the output power density of the robot and decreases the robot inertia as the actuators are fixed on the ground. The robot was modified to be able to counter-balance a good portion of its own weight. It can reduce the load on the actuators at and smaller actuators can be used to get the same power from the robot.

The kinematics of the robot was derived as well as its dynamics model using the Lagrange's method for simulation purposes. To find the structural dimensions of the robot, an optimization program was used. This program used the interior point method to find the best dimensions that result in a good combination of robot isotropy, mechanical advantage, work-space size and counter-balance. Also, multiple constraints were considered in solving the optimization problem to control different aspects of the solution. For example, lengths of links cannot exceed a certain value to prevent very large robot size and cannot be so small to ensure manufacturability; the workspace size had a minimum that is needed for practical movement therapy tasks; actuator stroke was limited by the maximum stroke of the selected

linear actuators; and the minimum distance between the robot end-effector and used was limited as a safety factor.

6.5. Adaptive Control with Modeling Inertial effects

In this research an adaptive control algorithm for robotic movement therapy was proposed that includes inertia learning components. It is well known that the CNS uses an internal model of person's limbs in order to estimate the muscle forces needed for fast and coordinated motions. The controller proposed in this research includes has unstructured inertial terms that can emulate the internal model formation and execution. A computer model of arm and motor control that was impaired in a certain region was used to test the proposed controller. It can be verified that this model works as predicted.

References

[1] N. Tejima, "Rehabilitation robotics: a review," *Advanced Robotics,* vol. 14, pp. 551-564, 2001.

[2] H. Krebs, J. Palazzolo, L. Dipietro, M. Ferraro, J. Krol, K. Rannekleiv, *et al.*, "Rehabilitation robotics: performance-based progressive robot-assisted therapy," *Auto. Rob.,* vol. 15, pp. 7-20, 2003.

[3] J. Eschweiler, K. Gerlach-Hahn, A. Jansen-Toy, and S. Leonhardt, "A survey on robotic devices for upper limb rehabilitation," *Journal of neuroengineering and rehabilitation,* vol. 11, p. 3, 2014.

[4] W. S. Harwin, T. Rahman, and R. A. Foulds, "A review of design issues in rehabilitation robotics with reference to North American research," *Rehabilitation Engineering, IEEE Transactions on,* vol. 3, pp. 3-13, 1995.

[5] S. Hesse, H. Schmidt, C. Werner, and A. Bardeleben, "Upper and lower extremity robotic devices for rehabilitation and for studying motor control," *Current Opinion in Neurology,* vol. 16, pp. 705–710, 2003.

[6] H. I. Krebs, B. T. Volpe, M. L. Aisen, W. Hening, S. Adamovich, H. Poizner, *et al.*, "Robotic applications in neuromotor rehabilitation," *Robotica,* vol. 21, pp. 3-11, 2003.

[7] H. I. Krebs, B. T. Volpe, M. L. Aisen, and N. Hogan, "Increasing productivity and quality of care: Robot-aided neuro-rehabilitation," *Journal of Rehabilitation Research and Development,* vol. 37, pp. 639-52, 2000.

[8] M. S. Nash, "Exercise as a health-promoting activity following spinal cord injury," *J Neurol Phys Ther,* vol. 29, pp. 87-103, 106, Jun 2005.

[9] T. Platz, "Evidence-based arm rehabilitation - a systematic review of the literature," *Nervenarzt,* vol. 74, pp. 841-+, OCT 2003 2003.

[10] A. Sunderland, D. J. Tinson, E. L. Bradley, D. Fletcher, R. Langton Hewer, and D. T. Wade, "Enhanced physical therapy improves recovery of arm function after stroke. A randomized controlled trial," *J. Neurology, Neurosurgery, and Psychiatry,* vol. 55, pp. 530-535, 1992.

[11] R. Riener, M. Frey, M. Bernhardt, T. Nef, and G. Colombo, "Human-centered rehabilitation robotics," in *Rehabilitation Robotics, 2005. ICORR 2005. 9th International Conference on*, 2005, pp. 319-322.

[12] A. Roy, H. I. Krebs, S. L. Patterson, T. N. Judkins, I. Khanna, L. W. Forrester, *et al.*, "Measurement of Human Ankle Stiffness Using the Anklebot," in *Rehabilitation Robotics, 2007. ICORR 2007. IEEE 10th International Conference on*, 2007, pp. 356-363.

[13] W. Gallagher, T. McPherson, J. Huggins, M. Shinohara, G. Dalong, R. Menassa, *et al.*, "An improved human-robot interface by measurement of muscle stiffness," in *Biomedical Robotics and Biomechatronics (BioRob), 2012 4th IEEE RAS & EMBS International Conference on*, 2012, pp. 177-182.

[14] T. Nef and R. Riener, "ARMin – Design of a Novel Arm Rehabilitation Robot," *Proceedings of the 2005 IEEE International Conference on Rehabilitation Robotics, June 28-July 1, Chicago, Illinois*, pp. 57-60, 2005.

[15] 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, p. 20, 2009.

[16] H. I. Krebs, J. J. Palazzolo, L. Dipietro, M. Ferraro, J. Krol, K. Rannekleiv, *et al.*, "Rehabilitation robotics: Performance-based progressive robot-assisted therapy," *Autonomous Robots*, vol. 15, pp. 7-20, 2003.

[17] H. Taheri, J. B. Rowe, D. Gardner, V. Chan, D. J. Reinkensmeyer, and E. T. Wolbrecht, "Robot-assisted Guitar Hero for finger rehabilitation after stroke," in *Engineering in Medicine and Biology Society (EMBC), 2012 Annual International Conference of the IEEE*, 2012, pp. 3911-3917.

[18] D. J. Reinkensmeyer, J. L. Emken, and S. C. Cramer, "Robotics, motor learning, and neurologic recovery," *Annual Review of Biomedical Engineering*, vol. 6, pp. 497-525, 2004.

[19] J. Zariffa, N. Kapadia, J. Kramer, P. Taylor, M. Alizadeh-Meghrazi, V. Zivanovic, *et al.*, "Relationship between clinical assessments of function and measurements from an upper-limb robotic rehabilitation device in cervical spinal cord injury," *Neural Systems and Rehabilitation Engineering, IEEE Transactions on*, pp. 1-1, 2011.

[20] J. Zariffa, N. Kapadia, J. L. K. Kramer, P. Taylor, M. Alizadeh-Meghrazi, V. Zivanovic, *et al.*, "Feasibility and efficacy of upper limb robotic rehabilitation in a subacute cervical spinal cord injury population," *Spinal cord*, vol. 50, pp. 220-226, 2012.

[21] J. Hidler, D. Nichols, M. Pelliccio, K. Brady, D. D. Campbell, J. H. Kahn, *et al.*, "Multicenter randomized clinical trial evaluating the effectiveness of the Lokomat in subacute stroke," *Neurorehabilitation and Neural Repair*, vol. 23, pp. 5-13, 2009.

[22] R. Riener, M. Guidali, U. Keller, A. Duschau-Wicke, V. Klamroth, and T. Nef, "Transferring ARMin to the Clinics and Industry," *Topics in Spinal Cord Injury Rehabilitation*, vol. 17, pp. 54-59, 2011.

[23] L. Bishop and J. Stein, "Three upper limb robotic devices for stroke rehabilitation: A review and clinical perspective," *NeuroRehabilitation*, vol. 33, pp. 3-11, 2013.

[24] *Hocoma*. Available: http://www.hocoma.com/

[25] L. Lucas, M. DiCicco, and Y. Matsuoka, "An EMG-controlled hand exoskeleton for natural pinching," *Journal of Robotics and Mechatronics*, vol. 16, pp. 482-488, 2004.

[26] K. Kiguchi, M. H. Rahman, M. Sasaki, and K. Teramoto, "Development of a 3DOF mobile exoskeleton robot for human upper-limb motion assist," *Robotics and Autonomous Systems*, vol. 56, pp. 678-691, 2008.

[27] H. Kobayashi and H. Nozaki, "Development of muscle suit for supporting manual worker," in *Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on*, 2007, pp. 1769-1774.

[28] R. A. R. C. Gopura and K. Kiguchi, "A human forearm and wrist motion assist exoskeleton robot with EMG-based Fuzzy-neuro control," in *Biomedical Robotics and Biomechatronics, 2008. BioRob 2008. 2nd IEEE RAS & EMBS International Conference on*, 2008, pp. 550-555.

[29] R. Gopura, K. Kiguchi, and Y. Li, "SUEFUL-7: A 7DOF upper-limb exoskeleton robot with muscle-model-oriented EMG-based control," *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 1126-1131, 2009.

[30] A. Pedrocchi, S. Ferrante, E. Ambrosini, M. Gandolla, C. Casellato, T. Schauer, *et al.*, "MUNDUS project: MUltimodal Neuroprosthesis for daily upper limb support," *J Neuroeng Rehabil*, vol. 10, p. 66, 2013.

[31] H. I. Krebs, N. Hogan, M. L. Aisen, and B. T. Volpe, "Robot-aided neurorehabilitation," *Rehabilitation Engineering, IEEE Transactions on*, vol. 6, pp. 75-87, 1998.

[32] D. J. Reinkensmeyer, L. E. Kahn, M. Averbuch, A. McKenna-Cole, B. D. Schmit, and W. Z. Rymer, "Understanding and treating arm movement impairment after chronic brain injury: progress with the ARM guide," *J Rehabil Res Dev*, vol. 37, pp. 653-62, 2000.

[33] G. Rosati, P. Gallina, and S. Masiero, "Design, implementation and clinical tests of a wire-based robot for neurorehabilitation," *IEEE Trans Neural Syst Rehabil Eng*, vol. 15, pp. 560-569, 2007.

[34] D. Gijbels, I. Lamers, L. Kerkhofs, G. Alders, E. Knippenberg, and P. Feys, "The Armeo Spring as training tool to improve upper limb functionality in multiple sclerosis: a pilot study," *J Neuroeng Rehabil*, vol. 8, p. 5, 2011.

[35] E. T. Wolbrecht, J. Leavitt, D. J. Reinkensmeyer, and J. E. Bobrow, "Control of a pneumatic orthosis for upper extremity stroke rehabilitation," in *Engineering in Medicine and Biology Society, 2006. EMBS '06. 28th Annual International Conference of the IEEE*, New York, New York, 2006, pp. 2687-2693.

[36] K. Oda, S. Isozumi, Y. Ohyama, K. Tamida, T. Kikuchi, and J. Furusho, "Development of isokinetic and iso-contractile exercise machine MEM-MRB using MR brake," *Proc. IEEE Int. Conf. on Rehabilitation Robotics (ICORR)*, pp. 6-11, 2009.

[37] R. C. V. Loureiro, J. M. Belda-Lois, E. R. Lima, J. L. Pons, J. J. Sanchez-Lacuesta, and W. S. Harwin, "Upper limb tremor suppression in ADL via an orthosis incorporating a controllable double viscous beam actuator," *Proc. 9th Int. Conf. on Rehabilitation Robotics ICORR*, pp. 119-122, 2005.

[38] M. Mihelj, J. Podobnik, and M. Munih, "HEnRiE - Haptic environment for reaching and grasping exercise," *Proc. 2nd IEEE RAS & EMBS International Conference on Biomedical Robotics and Biomechatronics (BioRob)*, pp. 907-912, 2008.

[39] R. Van Der Linde and P. Lammertse, "HapticMaster-a generic force controlled robot for human interaction," *Industrial Robot: Int J*, vol. 30, pp. 515-524, 2003.

[40] R. Sanchez, D. Reinkensmeyer, P. Shah, J. Liu, S. Rao, R. Smith, *et al.*, "Monitoring functional arm movement for home-based therapy after stroke," *Engineering in Medicine and Biology Society, 2004. EMBC 2004. Conference Proceedings. 26th Annual International Conference of the*, 2004.

[41] M. Schoone, P. van Os, and A. Campagne, "Robot-mediated Active Rehabilitation (ACRE) A user trial," *Proc. IEEE 10th International Conference on Rehabilitation Robotics (ICORR)*, pp. 477-481, 2007.

[42] S. J. Spencer, J. Klein, K. Minakata, V. Le, J. E. Bobrow, and D. J. Reinkensmeyer, "A low cost parallel robot and trajectory optimization method for wrist and forearm rehabilitation using the Wii," *Proc. 2nd IEEE RAS & EMBS International Conference on Biomedical Robotics and Biomechatronics (BioRob)*, pp. 869-874, 2008.

[43] J. C. Perry, J. Rosen, and S. Burns, "Upper-limb powered exoskeleton design," *Mechatronics, IEEE/ASME Trans*, vol. 12, pp. 408-417, 2007.

[44] J. Klein, S. Spencer, J. Allington, J. E. Bobrow, and D. J. Reinkensmeyer, "Optimization of a parallel shoulder mechanism to achieve a high-force, low-mass, robotic-arm exoskeleton," *Robotics, IEEE Trans*, vol. 26, pp. 710-715, 2010.

[45] A. Gupta, M. O'Malley, V. Patoglu, and C. Burgar, "Design, control and performance of RiceWrist: a force feedback wrist exoskeleton for rehabilitation and training," *Int J Robot Res*, vol. 27, p. 233, 2008.

[46] A. Toth, G. Fazekas, G. Arz, M. Jurak, M. Horvath, and R. Passive robotic movement therapy of the spastic hemiparetic arm with, "report of the first clinical test and the follow-up

system improvement," *Proc. 9th International Conference on Rehabilitation Robotics (ICORR)*, pp. 127-130, 2005.

[47] C. G. Burgar, P. S. Lum, P. C. Shor, and H. der Loos, "Development of robots for rehabilitation therapy: the Palo Alto VA/Stanford experience," *J Rehabil Res Dev*, vol. 37, pp. 663-673, 2000.

[48] H. I. Krebs, N. Hogan, W. Durfee, and H. Herr, "REHABILITATION ROBOTICS, ORTHOTICS, AND PROSTHETICS," *Textbook of Neural Repair and Rehabilitation*, 2006.

[49] H. Taheri, J. Rowe, D. Gardner, V. Chan, K. Gray, C. Bower, *et al.*, "Design and preliminary evaluation of the FINGER rehabilitation robot: controlling challenge and quantifying finger individuation during musical computer game play," *Journal of NeuroEngineering and Rehabilitation*, vol. 11, p. 10, 2014.

[50] C. D. Takahashi, L. Der-Yeghiaian, V. Le, R. R. Motiwala, and S. C. Cramer, "Robot-based hand motor therapy after stroke," *Brain*, vol. 131, p. 425, 2008.

[51] R. Vertechy, A. Frisoli, A. Dettori, M. Solazzi, and M. Bergamasco, "Development of a new exoskeleton for upper limb rehabilitation," *Proc. IEEE International Conference on Rehabilitation Robotics (ICORR)*, pp. 188-193, 2009.

[52] D. Sasaki, T. Noritsugu, and M. Takaiwa, "Development of Active Support Splint driven by Pneumatic Soft Actuator (ASSIST)," *Proc. IEEE International Conference on Robotics and Automation (ICRA)*, pp. 520-525, 2005.

[53] C. T. Freeman, A. M. Hughes, J. H. Burridge, P. H. Chappell, P. L. Lewin, and E. Rogers, "A robotic workstation for stroke rehabilitation of the upper extremity using FES," *Med Eng Phys*, vol. 31, pp. 364-373, 2009.

[54] R. Li, X. L. Hu, and K. Y. Tong, "Combined Electromyography(EMG)-driven system with functional electrical stimulation (FES) for poststroke rehabilitation," *Proc. 2nd IEEE RAS & EMBS International Conference on Biomedical Robotics and Biomechatronics (BioRob)*, pp. 642-646, 2008.

[55] S. Hamid and R. Hayek, "Role of electrical stimulation for rehabilitation and regeneration after spinal cord injury: an overview," *Eur Spine J*, vol. 17, pp. 1256-1269, 2008.

[56] P. S. Lum, C. G. Burgar, S. P.C., M. Majmundar, and M. Van der Loos, "Robot-assisted movement training compared with conventional therapy techniques for the rehabilitation of upper limb motor function following stroke," *Arch. Phys. Med. Rehabil.*, vol. 83, pp. 952-9, 2002.

[57] E. Rocon, J. M. Belda-Lois, A. F. Ruiz, M. Manto, J. C. Moreno, and J. L. Pons, "Design and validation of a rehabilitation robotic exoskeleton for tremor assessment and suppression," *IEEE Trans Neural Syst Rehabil Eng*, vol. 15, pp. 367-378, 2007.

[58] R. J. Sanchez, J. Liu, S. Rao, P. Shah, R. Smith, T. Rahman, *et al.*, "Automating Arm Movement Training Following Severe Stroke: Functional Exercises With Quantitative Feedback in a Gravity-Reduced Environment," *Neural Systems and Rehabilitation Engineering, IEEE Transactions on [see also IEEE Trans. on Rehabilitation Engineering]*, vol. 14, pp. 378-389, 2006.

[59] A. H. A. Stienen, E. E. G. Hekman, F. C. T. Van der Helm, G. B. Prange, M. J. A. Jannink, A. M. M. Aalsma, *et al.*, "Freebal: dedicated gravity compensation for the upper extremities," in *Rehabilitation Robotics, IEEE 10th International Conference on*, 2007, pp. 804-808.

[60] R. Song, K. yu Tong, X. Hu, and L. Li, "Assistive control system using continuous myoelectric signal in robot-aided arm training for patients after stroke," *IEEE Trans Neural Syst Rehabil Eng*, vol. 16, pp. 371-379, 2008.

[61] T. Hayashi, H. Kawamoto, and Y. Sankai, "Control method of robot suit HAL working as operator's muscle using biological and dynamical information," in *Intelligent Robots and Systems, 2005. (IROS 2005). 2005 IEEE/RSJ International Conference on*, 2005, pp. 3063-3068.

[62] E. T. Wolbrecht, V. Chan, D. J. Reinkensmeyer, and J. E. Bobrow, "Optimizing compliant, model-based robotic assistance to promote neurorehabilitation," *IEEE Trans Neural Syst Rehabil Eng*, vol. 16, pp. 286-297, 2008.

[63] L. E. Kahn, W. Z. Rymer, and D. J. Reinkensmeyer, "Adaptive assistance for guided force training in chronic stroke," *Engineering in Medicine and Biology Society, 2004. EMBC 2004. Conference Proceedings. 26th Annual International Conference of the*, vol. 1, 2004.

[64] J. von Zitzewitz, M. Bernhardt, and R. Riener, "A Novel Method for Automatic Treadmill Speed Adaptation," *Neural Systems and Rehabilitation Engineering, IEEE Transactions on*, vol. 15, pp. 401-409, 2007.

[65] A. Stienen, E. Hekman, G. Prange, M. Jannink, A. Aalsma, F. van der Helm, *et al.*, "Dampace: Design of an exoskeleton for force-coordination training in upper-extremity rehabilitation," *J Med Devices*, vol. 3, p. 031003, 2009.

[66] M. J. Johnson, H. F. M. Van der Loos, C. G. Burgar, P. Shor, and L. J. Leifer, "Design and evaluation of Driver's SEAT: A car steering simulation environment for upper limb stroke therapy," *Robotica*, vol. 21, pp. 13-23

[67] A. M. Simon, R. Brent Gillespie, and D. P. Ferris, "Symmetry-based resistance as a novel means of lower limb rehabilitation," *Journal of Biomechanics,* vol. 40, pp. 1286-1292, // 2007.

[68] J. L. Patton, M. E. Stoykov, M. Kovic, and F. A. Mussa-Ivaldi, "Evaluation of robotic training forces that either enhance or reduce error in chronic hemiparetic stroke survivors," *Experimental Brain Research,* vol. 168, pp. 368-383, 2006.

[69] Y. Wei, P. Bajaj, R. Scheidt, and J. L. Patton, "Visual error augmentation for enhancing motor learning and rehabilitative relearning," *Proceedings 2005 IEEE International Conference on Rehabilitation Robotics,* pp. 505-510, 2005.

[70] J. L. Emken and D. J. Reinkensmeyer, "Robot-enhanced motor learning: Accelerating internal model formation during locomotion by transient dynamic amplification," *Neural Systems and Rehabilitation Engineering, IEEE Transactions on,* vol. 13, pp. 33-39, 2005.

[71] J. L. Emken and D. J. Reinkensmeyer, "Human-robot cooperative movement training: Learning a novel sensory motor transformation during walking with robotic assistance-as-needed," *Journal of Neuroengineering and Rehabilitation,* p. (in press), 2007.

[72] J. Patton, G. Dawe, C. Scharver, F. Mussa-Ivaldi, and R. Kenyon, "Robotics and virtual reality: the development of a life-sized 3-D system for the rehabilitation of motor function," *26th Annual International Conference of the IEEE Engineering in Medicine and Biology Society,* pp. 4840-3, 2004.

[73] J. Eriksson, M. Mataric, and C. Winstein, "Hands-off assistive robotics for post-stroke arm rehabilitation," *Proceedings of the 2005 IEEE International Conference on Rehabilitation Robotics, June 28-July 1, Chicago, Illinois,* pp. 21-24, 2005.

[74] R. Riener, L. Lunenburger, S. Jezernik, M. Anderschitz, G. Colombo, and V. Dietz, "Patient-cooperative strategies for robot-aided treadmill training: first experimental results," *Neural Systems and Rehabilitation Engineering, IEEE Transactions on [see also IEEE Trans. on Rehabilitation Engineering],* vol. 13, pp. 380-394, 2005.

[75] J. J. E. Slotine and W. Li, *Applied nonlinear control*: Prentice Hall Englewood Cliffs, NJ, 1991.

[76] A. S. Go, D. Mozaffarian, V. L. Roger, E. J. Benjamin, J. D. Berry, M. J. Blaha, *et al.*, "Heart Disease and Stroke Statistics—2014 Update: A Report From the American Heart Association," *Circulation,* 2013.

[77] J. Broeks, G. Lankhorst, K. Rumping, and A. Prevo, "The long-term outcome of arm function after stroke: results of a follow-up study," *Disability and Rehabilitation,* vol. 21, pp. 357-364, AUG 1999 1999.

[78] G. Kwakkel, B. J. Kollen, and H. I. Krebs, "Effects of Robot-Assisted Therapy on Upper Limb Recovery After Stroke: A Systematic Review," *Neurorehabilitation and Neural Repair,* vol. 22, p. 111, 2008.

[79] G. B. Prange, M. J. A. Jannink, C. G. M. Groothuis, H. J. Hermens, and M. J. IJzerman, "Systematic review of the effect of robot-aided therapy on recovery of the hemiparetic arm after stroke," *J. Rehabil. Res. Develop,* vol. 43, pp. 171-184, 2006.

[80] J. van der Lee, I. Snels, H. Beckerman, G. Lankhorst, R. Wagenaar, and L. Bouter, "Exercise therapy for arm function in stroke patients: a systematic review of randomized controlled trial," *Clinical Rehabilitation,* vol. 15, pp. 20-31, 2001.

[81] P. M. Rossini and G. Dal Forno, "Integrated technology for evaluation of brain function and neural plasticity," *Physical medicine and rehabilitation clinics of North America,* vol. 15, pp. 263-306, 2004.

[82] J. F. Israel, D. D. Campbell, J. H. Kahn, and T. G. Hornby, "Metabolic Costs and Muscle Activity Patterns During Robotic- and Therapist-Assisted Treadmill Walking in Individuals With Incomplete Spinal Cord Injury," *Physical Therapy,* vol. 86, pp. 1466-1478, November 2006 2006.

[83] M. Lotze, C. Braun, N. Birbaumer, S. Anders, and L. G. Cohen, "Motor learning elicited by voluntary drive," *Brain,* vol. 126, pp. 866-872, 2003.

[84] A. Kaelin-Lang, L. Sawaki, and L. G. Cohen, "Role of Voluntary Drive in Encoding an Elementary Motor Memory," ed: Am Physiological Soc, 2005.

[85] L. L. Cai, A. J. Fong, C. K. Otoshi, Y. Liang, J. W. Burdick, R. R. Roy, *et al.*, "Implications of assist-as-needed robotic step training after a complete spinal cord injury on intrinsic strategies of motor learning," *J Neuroscience,* vol. 26, pp. 10564-8, 2006.

[86] R. Colombo, F. Pisano, S. Micera, A. Mazzone, C. Delconte, M. C. Carrozza, *et al.*, "Robotic techniques for upper limb evaluation and rehabilitation of stroke patients," *Neural Systems and Rehabilitation Engineering, IEEE Transactions on [see also IEEE Trans. on Rehabilitation Engineering],* vol. 13, pp. 311-324, 2005.

[87] D. Nowak, "The impact of stroke on the performance of grasping: Usefulness of kinetic and kinematic motion analysis," *Neuroscience and Biobehavioral Reviews,* vol. 32, pp. 1439-1450, OCT 2008 2008.

[88] S. Balasubramanian, J. Klein, and E. Burdet, "Robot-assisted rehabilitation of hand function," *Current Opinion in Neurology,* vol. 23, pp. 661-670, DEC 2010 2010.

[89] M. R. Cutkosky and I. Kao, "Computing and controlling compliance of a robotic hand," *Robotics and Automation, IEEE Transactions on,* vol. 5, pp. 151-165, 1989.

[90] E. T. Wolbrecht, D. J. Reinkensmeyer, and A. Perez-Gracia, "Single degree-of-freedom exoskeleton mechanism design for finger rehabilitation," *IEEE ... International Conference on Rehabilitation Robotics : [proceedings]*, vol. 2011, p. 5975427, 2011 Jun 29-Jul 2011.

[91] B. Heisele, U. Kressel, and W. Ritter, "Tracking non-rigid, moving objects based on color cluster flow," in *Computer Vision and Pattern Recognition, 1997. Proceedings., 1997 IEEE Computer Society Conference on*, 1997, pp. 257-260.

[92] N. A. S. Book, "Volume II: A Handbook of Anthropometric Data," *NASA Reference Publication*, vol. 1024, 1978.

[93] I. Serban, M. Baritz, I. C. Rosca, and L. D. Cotoros, "Statistical Analysis of Anthropometric and Physiologic Performance of the Hand," in *International Conference on Advancements of Medicine and Health Care through Technology*. vol. 36, S. Vlad and R. Ciupa, Eds., ed: Springer Berlin Heidelberg, 2011, pp. 380-383.

[94] H. Dreyfuss and A. Tilley, "Measure of Man and Woman: Human Factors in Design," ed: Watson-Guptill, 1993.

[95] G. N. Sandor and A. G. Erdman, *Advanced mechanism design. Analysis and synthesis. Vol. 2*: Englewood Cliffs, NJ, Prentice-Hall, 1984.

[96] G. S. Soh and J. M. McCarthy, "The synthesis of six-bar linkages as constrained planar 3R chains," *Mechanism and Machine Theory*, vol. 43, pp. 160-170, 2008.

[97] E. T. Wolbrecht, D. J. Reinkensmeyer, and J. E. Bobrow, "Pneumatic Control of Robots for Rehabilitation," *International Journal of Robotics Research*, vol. 29, pp. 23-38, Jan 2010.

[98] A. R. Fugl-Meyer, L. Jaasco, L. Leyman, S. Olsson, and S. Steglind, "The post-stroke hemiplegic patient," *Scand. Journal Rehab. Med.*, vol. 7, pp. 13-31, 1975.

[99] V. Mathiowetz, G. Volland, N. Kashman, and K. Weber, "Adult norms for the Box and Block Test of manual dexterity," *Am J Occup Ther*, vol. 39, pp. 386-91, 1985.

[100] S. J. Spencer, "Movement Training and Post-Stroke Rehabilitation Using a Six Degree of Freedom Upper-Extremity Robotic Orthosis and Virtual Environment," ed. UC Irvine, 2012.

[101] M. A. Guadagnoli and T. D. Lee, "Challenge point: a framework for conceptualizing the effects of various practice conditions in motor learning," *Journal of Motor Behavior*, vol. 36, pp. 212-224, 2004.

[102] C. E. Lang and M. H. Schieber, "Differential impairment of individuated finger movements in humans after damage to the motor cortex or the corticospinal tract," *Journal of neurophysiology*, vol. 90, pp. 1160-1170, 2003.

[103] C. E. Lang and M. H. Schieber, "Reduced muscle selectivity during individuated finger movements in humans after damage to the motor cortex or corticospinal tract," *Journal of neurophysiology*, vol. 91, pp. 1722-1733, 2004.

[104] T. Nef, M. Mihelj, and R. Riener, "ARMin: a robot for patient-cooperative arm therapy," *Medical and Biological Engineering and Computing*, vol. 45, pp. 887-900, 2007.

[105] G. Rosati, J. E. Bobrow, and D. J. Reinkensmeyer, "Compliant control of post-stroke rehabilitation robots: using movement-specific models to improve controller performance," presented at the ASME International Mechanical Engineering Congress and Exposition, excepted for publication in, Boston, Massachusetts, 2008.

[106] D. Kamper and W. Rymer, "Impairment of voluntary control of finger motion following stroke: role of inappropriate muscle coactivation," *Muscle & Nerve*, vol. 24, pp. 673-681, 2001.

[107] D. G. Kamper, H. C. Fischer, E. G. Cruz, and W. Z. Rymer, "Weakness is the primary contributor to finger impairment in chronic stroke," *Archives of physical medicine and rehabilitation*, vol. 87, pp. 1262-1269, 2006.

[108] M. O. Conrad and D. G. Kamper, "Isokinetic strength and power deficits in the hand following stroke," *Clinical Neurophysiology*, vol. 123, pp. 1200-1206, 2012.

[109] D. G. Kamper, R. L. Harvey, S. Suresh, and W. Z. Rymer, "Relative contributions of neural mechanisms versus muscle mechanics in promoting finger extension deficits following stroke," *Muscle & nerve*, vol. 28, pp. 309-318, 2003.

[110] F. Pierrot, F. Marquet, O. Company, and T. Gil, "H4 parallel robot: modeling, design and preliminary experiments," in *Robotics and Automation, 2001. Proceedings 2001 ICRA. IEEE International Conference on*, 2001, pp. 3256-3261 vol.4.

[111] F. Pierrot and O. Company, "H4: a new family of 4-DOF parallel robots," in *Advanced Intelligent Mechatronics, 1999. Proceedings. 1999 IEEE/ASME International Conference on*, 1999, pp. 508-513.

[112] S. Boyd and L. Vandenberghe, "Convex optimization. 2004," *Cambridge Univ. Pr*, 2004.

[113] A. Kaelin-Lang, L. Sawaki, and L. G. Cohen, "Role of voluntary drive in encoding an elementary motor memory," *Journal of neurophysiology*, vol. 93, pp. 1099-1103, 2005.

[114] C. Bower, H. Taheri, and E. Wolbrecht, "Adaptive control with state-dependent modeling of patient impairment for robotic movement therapy," pp. 1-6.

[115] E. T. Wolbrecht, V. Chan, D. J. Reinkensmeyer, and J. E. Bobrow, "Optimizing compliant, model-based robotic assistance to promote neurorehabilitation," *Neural Systems and Rehabilitation Engineering, IEEE Transactions on*, vol. 16, pp. 286-297, 2008.

[116] R. Shadmehr and F. A. Mussa-Ivaldi, "Adaptive representation of dynamics during learning of a motor task," *Journal of Neuroscience*, vol. 14, pp. 3208-3224, 1994.

[117] J. Hidler and R. Sainburg, "Role of robotics in neurorehabilitation," *Topics in spinal cord injury rehabilitation*, vol. 17, pp. 42-49, 2011.

[118] C. Bower, H. Taheri, and E. Wolbrecht, "Adaptive control with state-dependent modeling of patient impairment for robotic movement therapy," in *Rehabilitation Robotics (ICORR), 2013 IEEE International Conference on*, 2013, pp. 1-6.

[119] M. Berniker, D. W. Franklin, J. R. Flanagan, D. M. Wolpert, and K. Kording, "Motor learning of novel dynamics is not represented in a single global coordinate system: evaluation of mixed coordinate representations and local learning," *Journal of neurophysiology*, vol. 111, pp. 1165-1182, 2014.

[120] M. W. Spong and M. Vidyasagar, *Robot dynamics and control*. New York: Wiley, 1989.

[121] R. Shadmehr and S. Mussa-Ivaldi, *Biological learning and control: how the brain builds representations, predicts events, and makes decisions*: Mit Press, 2012.

[122] M. D. Buhmann, *Radial basis functions: theory and implementations* vol. 5: Cambridge university press Cambridge, 2003.