2020

# Achieving Practical Functional Electrical Stimulation-driven Reaching Motions In An Individual With Tetraplegia

Derek nathaniel Wolf
*Cleveland State University*

# ACHIEVING PRACTICAL FUNCTIONAL ELECTRICAL STIMULATION-DRIVEN REACHING MOTIONS IN AN INDIVIDUAL WITH TETRAPLEGIA

DEREK NATHANIEL WOLF

Bachelor of Science in Mechanical Engineering

University of Notre Dame

May 2013

Submitted in partial fulfillment of requirements for the degree

DOCTOR OF PHILOSOPHY IN MECHANICAL ENGINEERING

at the

CLEVELAND STATE UNIVERSITY

DECEMBER 2020

**We hereby approve the dissertation of**

**DEREK WOLF**

**Candidate for the Doctor of Philosophy in Engineering degree.**

**This dissertation has been approved for the specialization of**
**Mechanical Engineering**
**and CLEVELAND STATE UNIVERSITY'S**
**College of Graduate Studies by**

_____

Dr. Eric Schearer
Dissertation Committee Chairperson
Department of Mechanical Engineering

_____

Dr. Hanz Richter
Dissertation Committee Member
Department of Mechanical Engineering

_____

Dr. Antonie van den Bogert
Dissertation Committee Member
Department of Mechanical Engineering

_____

Dr. Deborah Espy
Dissertation Committee Member
Department of Health Sciences

_____

Dr. Levi Hargrove
Dissertation Committee Member
Department of Biomedical Engineering, Northwestern University

December 3, 2020
Student's Date of Defense

This student has fulfilled all requirements for the Doctor of Philosophy in Engineering degree.

_____

Chandra Kothapalli, Doctoral Program Director

## ACKNOWLEDGMENTS

First and foremost, I would like to thank my advisor, Dr. Eric Schearer, for his guidance during this journey. I have looked forward to coming to work with you every day for the past five and a half years. You have made me a better student, a better researcher, and most importantly, a better person. You may have even convinced me to love Gaussian processes for the sake of the math. I will greatly miss our road trips to conferences, brainstorming sessions, experiments at the hospital, and just generally spending time together. Thank you for taking a chance on me and for all you have done for me.

I would also like to thank the members of my committee: Dr. Hanz Richter, Dr. Ton van den Bogert, Dr. Debbie Espy, and Dr. Levi Hargrove. Thank you so much for your time, your thoughts, and your guidance as I pursued this journey. Either in classes or in meetings, you have all taught me so much and given so much of your time to me. I cannot thank you enough.

I would not be where I am today without the guidance of Dr. Jim Schmiedeler. You welcomed me into the world of research nearly eight years ago and have been a trusted mentor ever since. I cannot thank you enough for your continued guidance and support.

To all the members of my lab and CHMS, I could not have picked a better group of people to work with. You all have made my time at CSU the most enjoyable time of my life. You stretched the possibilities of my research with every discussion we had and made every day of work enjoyable. I know that great research will continue to come out of CHMS because it is full of great people. I would like to specifically thank Kyra, Andrew, Cameron, Jack, Anne, Sandy, and Huawei. Whether it was planning my research or planning my wedding, you all have been there for me throughout my time at CSU. I would not be where I am now without your friendship. I will greatly miss working with you all.

I would also like to thank the team at MetroHealth OBC. Bill, Chris, Anne, Ron, and Emma: the work that I was able to complete was only possible due to your support, knowledge, and effort. You have all contributed to my development as a researcher and person,

and I can only hope to achieve even a small portion of the impact on the spinal cord injury community that you all achieve every day.

To my many friends in Cleveland from all groups including the Cleveland Broomball Association and the Notre Dame Club of Cleveland: I may have graduated faster without you all, but I would not have enjoyed my time nearly as much. Cleveland will forever be our home because of you all.

Pete and Ellen, you have become some of my closest friends in the world. Thank you for welcoming Kayla and I to Cleveland and for being so supportive. Moving to Cleveland was a big decision in my life, and from day one, you made it clear that it was the right one. I am so lucky to have you all in my life as family, and, more importantly, as close friends.

I'd like to thank my family. My parents have always been so supportive of my journey, and I would not be where I am without them. Dad, thank you for always checking in on me and for distracting me with discussions about how bad the Browns are. Mom, thank you for your support, guidance, and for setting the example that the goal of obtaining a Ph.D. is achievable. Thank you to my brother-in-law, Tim, for providing me endless distraction by talking fantasy football, playing Nintendo Switch, and discussing "Daniel and Jorge". Thank you to the Tinsley family for always being supportive even as I drag Kayla away from home. Finally, I'd like to thank my sister, Erin. You've been my best friend my whole life. I will soon be half the doctor that you are, and if I can live up to half the person that you are, I will be doing pretty great.

Most importantly, I'd like to thank my wife, Kayla. I cannot thank you enough for your support and willingness to come on this journey with me. Whether it is completing my Ph.D. or simply surviving the pandemic together, I love every moment with you, and I could not imagine my life without you (and, of course, Mulligan too). I cannot wait for the next adventure of our journey together.

Finally, thank you to Oliver Bjorkstrand, Stipe Miocic, Francisco Lindor, and, of course, LeBron James for 2016, a year I will never forget!

ACHIEVING PRACTICAL FUNCTIONAL ELECTRICAL STIMULATION-DRIVEN REACHING MOTIONS IN AN INDIVIDUAL WITH TETRAPLEGIA

DEREK NATHANIEL WOLF

## ABSTRACT

Functional electrical stimulation (FES) is a promising technique for restoring the ability to complete reaching motions to individuals with tetraplegia due to a spinal cord injury (SCI). FES has proven to be a successful technique for controlling many functional tasks such as grasping, standing, and even limited walking. However, translating these successes to reaching motions has proven difficult due to the complexity of the arm and the goal-directed nature of reaching motions. The state-of-the-art systems either use robots to assist the FES-driven reaching motions or control the arm of healthy subjects to complete planar motions. These controllers do not directly translate to controlling the full-arm of an individual with tetraplegia because the muscle capabilities of individuals with spinal cord injuries are unique and often limited due to muscle atrophy and the loss of function caused by lower motor neuron damage. This dissertation aims to develop a full-arm FES-driven reaching controller that is capable of achieving 3D reaching motions in an individual with a spinal cord injury.

Aim 1 was to develop a complete-arm FES-driven reaching controller that can hold static hand positions for an individual with high tetraplegia due to SCI. We developed a combined feedforward-feedback controller which used the subject-specific model to automatically determine the muscle stimulation commands necessary to hold a desired static hand position.

Aim 2 was to develop a subject-specific model-based control strategy to use FES to drive the arm of an individual with high tetraplegia due to SCI along a desired path in the subject's workspace. We used trajectory optimization to find feasible trajectories which explicitly account for the unique muscle characteristics and the simulated arm dynamics of our subject with tetraplegia. We then developed a model predictive control controller to

control the arm along the desired trajectory.

The controller developed in this dissertation is a significant step towards restoring full-arm reaching function to individuals with spinal cord injuries.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

# CHAPTER I

# INTRODUCTION

## 1.1 Motivation

There are approximately 294,000 individuals in the United States that are living with a spinal cord injury (SCI) with nearly 60% of these injuries resulting in some level of incomplete (47.2%) or complete tetraplegia (12.3%) [1]. The repercussions of the injury on the quality of life for these individuals are vast. Financially, the lifetime cost for an individual with high tetraplegia is $5.1 million [1]. Additionally, employment rates remain less than half of the pre-injury rates even decades post injury. A significant factor leading to the financial struggles and loss of occupation is the need for frequent healthcare aid and a lack of ability to independently complete activities of daily living. The development of rehabilitation and assistive technologies that can restore daily independence to individuals with paralysis due to spinal cord injuries is of critical importance to alleviating these concerns and improving their quality of life.

According to individuals with tetraplegia, restoring hand and arm function is the highest priority for improving their quality of life [2]. The ability to reach is the way in which we, as humans, manipulate our environment and is critical to completing many activities of daily living such as self-feeding and grooming. Unfortunately, the options available to help these individuals regain this critical ability are limited. To date, there is no complete medical cure for a high spinal cord injury. For individuals with some residual muscle control, there

have been advances and successes in rehabilitation therapies [3]. Additionally, there are some surgical options available to individuals with tetraplegia to improve their hand and arm function include tendon transfers [4, 5] and nerve transfers [6]. The viability of these procedures depends on the level and extent of the spinal cord injury. For individuals who lack residual muscle control, or for those who the above procedures do not fully achieve their personal goals, the development of assistive technologies is critical to provide the necessary hand and arm function for independent daily life.

There are two main assistive technology approaches to restoring reaching motions to individuals with high tetraplegia, robotics and functional electrical stimulation (FES), also known as neuromuscular electrical stimulation (NMES). Robotic solutions include exoskeletons that assist and move the subject's arm directly [7], robotic arm supports [8], and stand-alone robots that the subject uses in place of their arms such as the one presented by [9]. These devices are often bulky, complex, and require large amounts of power to drive the actuators making them difficult to use in daily life. FES, on the other hand, is able to move the arm using comparatively little hardware and power by taking advantage of the natural actuation of the arm, the muscles. Successful development of FES-driven reaching controllers will allow for easier adoption of the technology and thus greater independence in daily life for individuals with tetraplegia. Even in individuals where FES alone cannot achieve all functional reaching goals, accurate FES control in combination with robotic actuation can create a much more viable system for use in everyday life (see [10] for a review of hybrid FES-robotic systems).

## 1.2 A brief background on functional electrical stimulation

FES systems restore functional movement by using electrical stimulation to create action potentials in the peripheral nervous system which propagate across the neuromuscular junction to cause the paralyzed muscles to contract [11, 12]. (Stimulation of the central nervous system is another feasible technique [13], but I will focus on peripheral stimulation in this

research.) The stimulation can be applied either via an implanted system [14, 15], percutaneous electrodes where the electrodes are inserted through the skin via a needle [16], or surface electrode arrays [17]. The stimulation is delivered via either monophasic or biphasic pulses with the amount of muscular activation controlled by varying either the pulse-width, amplitude, or the frequency of the stimulation. Biphasic pulses are generally used in implanted systems to balance the charge and minimize the amount of damage that can occur at the electrode-tissue interface [11].

There are important limitations of FES when trying to use it to restore functional movements to individuals with paralyzed limbs. First, all motor function FES applications operate by activating the nerve instead of directly activating the muscle because the electrical threshold for activation is lower [11]. Therefore, the lower motor neuron must be undamaged in order for FES to produce muscle contractions. This is notable for the work presented in the current document because in cases of high tetraplegia, there is often lower motor neuron damage in muscles critical to controlling the arm. These muscles often include the biceps, supraspinatus, and deltoids and sometimes include the pectoralis, triceps, and lattisiumus dorsi [18]. Additionally, activating muscles by electrical stimulation recruits muscles in the opposite order of the physiological size principle [11]. This leads to more rapid fatigue of the muscles when electrically activated instead of physiologically activated, although there are some methods of mitigating this issue via different stimulation patterns [19]. Lastly, there are time delays which are introduced by the low frequency of stimulation, generally 12-15 Hz, which limits the speed at which a control input can be implemented and the frequency with which corrective control action can occur. Despite these shortcomings, FES is a promising technology that has already demonstrated significant success in restoring functional movements to individuals with spinal cord injuries and individuals with residual deficits following a stroke.

FES has proven to be a successful technology in restoring many functional movements to individuals with paralysis due to a spinal cord injury including bladder control [20],

standing [21], and some walking [22–24]. For individuals with spinal cord injuries at the C5-C6 level, FES has demonstrated success in improving pinch force and grasping functions [14], and the Freehand system was briefly available as a commercial product [25]. These hand functions have the ability to drastically improve the quality of life for individuals with a lower level of SCI. However, for individuals with high tetraplegia due to high cervical (C1-C4) SCI, the need to control the full-arm is critical to improving their daily independence.

### 1.3 Functional electrical stimulation for controlling reaching

To unlock the full potential of FES to restore independence to individuals with spinal cord injuries, there is a critical need to develop controllers capable of controlling FES-driven reaching motions. It has been proposed in [26], that the best control method for FES uses a hierarchical hybrid structure where high level decisions are then implemented by low level controllers. To implement an FES controller, Lynch and Popovic [27] have proposed the following framework that must be met for clinical use of an FES system, "The FES system must:

1. compensate for the nonlinear, time-varying, and coupled nature of the muscle being controlled, including the effects of fatigue and training.

2. be stable in the presence of the time delays and perturbations (reflex contractions) that are inherenet to the system.

3. be implemented in portable, battery powered electronics, and should be designed for at least 16 hours of operation each day...

4. be compatible with efficient setup and calibration procedures that are simple enough to be performed by a therapist or patient..."

Additionally, they specify that the control strategy must be tested with individuals with SCI as their response to electrical stimulation will vary greatly from those of healthy subjects.

To date, FES controllers for reaching motions have had limited success in achieving these goals along with accurate control of desired motions. The purpose of my dissertation research presented here is to develop a low-level controller for FES-driven reaching motions that moves the field closer to meeting these desired goals.

Control of grasping function was achieved with a relatively simple implementation of applying a stimulation pattern in a proportional manner to an EMG signal [14]. Expanding these techniques to individuals with high tetraplegia due to high cervical (C1-C4) SCI for the purpose of achieving full reaching function to the complete arm has proven difficult. Controlling reaching motions requires coordinating the shoulder, elbow, wrist, and hand to complete reaching motions throughout the subject's workspace. Additionally, subjects with SCI who are using FES present unique muscular characteristics - muscles which cannot be activated due to lower motor neuron damage, rapid fatigue, and general weakness due to muscle atrophy - which make restoring full-arm motions with FES difficult. More complex control strategies have been implemented in lower limb functional restoration, but these controllers typically take advantage of the cyclic nature of the motions for activities like walking and cycling [28, 29]. For goal-directed, non-repetitive reaching motions, these control strategies are difficult to implement.

There have been many attempts to extend these control strategies, both simple and complex, to control one or two degrees of freedom of the arm. For example, elbow flexion has been controlled using reinforcement learning [30], co-activation of antagonist muscles [31], as well as position and torque based feedback controllers [32]. In rehabilitation settings, model-based, input-output linearization and iterative learning control has been used to control the arms of individuals with stroke [33–35]. The results of these systems are very exciting and prove the ability of a controller to accurately drive reaching motions. However, these systems generally actively control only two degrees of freedom and again rely on the repetitive nature of a rehabilitation setting which allows for a significant learning period. For a reaching controller to be effective in everyday life, the controller must be

able to complete novel, non-repetitive tasks.

The move to controlling the complete arm system in a non repetitive environment, instead of just a couple of degrees of freedom in a rehabilitation environment, has proven difficult. Many different techniques for controlling full-arm reaching have been implemented in simulation including threshold control [36], optimized PID control [37, 38], reinforcement learning [39], combined feedforward-feedback control [40], a feedforward-feedback controller that accounts for the electromechanical delays of the FES system [41], and multi-muscle control with co-contraction [42]. Although many of these ideas have guided the practical implementation of FES-reaching controller, few of these controllers have been successfully implemented due to the significant differences between simulation studies and the muscular dynamics and characteristics of individuals with SCI.

To date (and to my knowledge), there have been only four main groups to practically implement full-arm FES controllers: the MUNDUS/ReTrainer team [30, 43, 44], the BrainGate2 clinical trial team [16, 45], Razavian and McPhee [46–48], and the Cleveland FES Center group of which my lab is a part of [15, 20, 49]. Each of these projects have had significant successes and moved the field closer to achieving full-arm FES-driven reaching in individuals with SCI. However, each group's methods have significant limitations.

A seminal work to achieving FES-driven reaching motions, the MUltimodal Neuroprosthesis for daily Upper limb Support, MUNDUS, project [43, 44] used a passive, lockable exoskeleton in combination with FES to drive reaching motions in individuals with high-level SCI (see Fig. 1 to view the experimental set-up). The system used surface electrodes to apply stimulation to four channels in order to control the biceps (elbow flexion/extension), and the deltoids muscles (shoulder rotation, and shoulder flexion/extension). The system could be donned in less than 10 minutes and required about two minutes for calibration thus eliminating the need for a long learning period. The passive exoskeleton provided gravity compensation as well as the ability to lock joints once their desired position was achieved. The system used a sequential control strategy driving one degree of

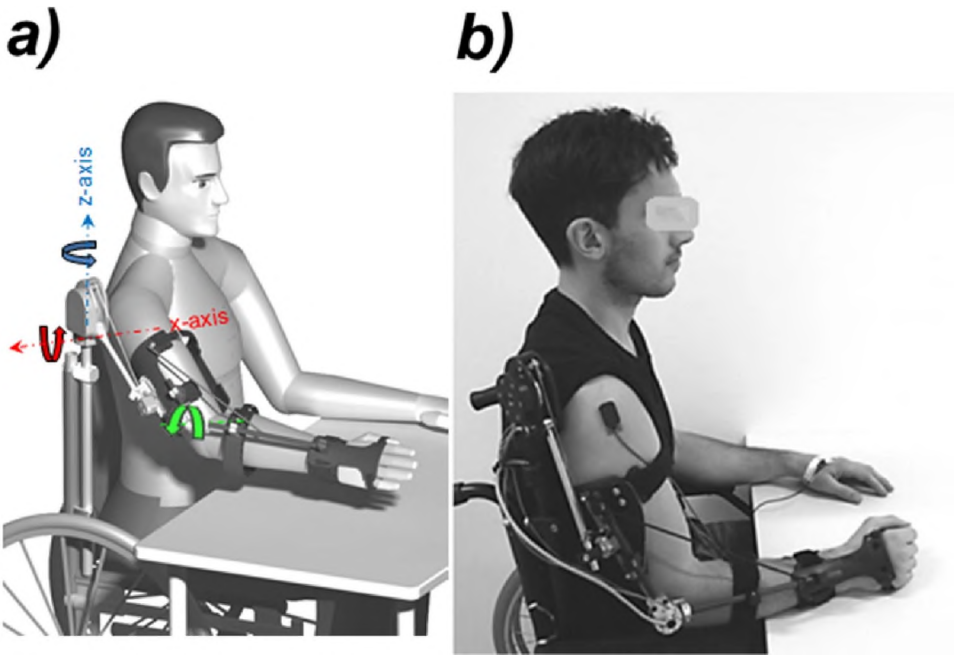



Figure 1: Experimental setup for the MUNDUS project showing a lockable exoskelton with motion driven by FES. Image is from [43] and is used with permission

freedom at a time while the exoskeleton provided braking to lock the other joints. This results in a completely decoupled system that is significantly easier to control. Using the system, the subject was able to place the hand within 2 cm of the desired position which allowed the subject to drink from a cup with some head movement to compensate for small errors in the placement of the straw. While the accuracy and functionality produced by the relatively simple controller are tremendous successes, by controlling each joint sequentially, the system required, on average, 71.4 seconds to complete a drinking task and produced motions which appear unnatural. Additionally, significant errors arose because the exoskeleton could not prevent the shoulder from slipping in the horizontal rotation degree of freedom. By controlling the arm as a complete system, instead of sequential decoupled joints, more physiologically natural movements can be achieved, and the shoulder rotation can be actively controlled to produce more accurate reaches.

The BrainGate2 clinical study [16, 45] is the current state-of-the-art system for reaching motions in individuals with high tetraplegia (see Fig. 2(A) for an illustration of the system).

The system used an implanted brain control interface to send commands to percutaneous FES electrodes to control the triceps and biceps in the arm (elbow flexion/extension) as well as control of hand and arm function. Unlike the MUNDUS project where all motion was driven by FES, in the BrainGate2 system, shoulder ab/adduction was controlled via a robotic mobile arm support. The FES system decodes neural commands to select a stimulation command setting from along a predetermined stimulation pattern to produce a desired motion (see 2(B)). The system is able to produce more natural motions, and a subject able to drink from a cup and self-feed in a much faster, though still relatively slow, period of time, 20-40 seconds. The BrainGate2 system again demonstrated tremendous advancement for an individual with high tetraplegia due to SCI to be able to complete functional tasks independently. However, when controlling multiple joints, the largest failure mode (63% of all failures) were categorized as control interface challenges. These failures were largely caused by the motion of the arm support leading to undesired coupled motions in the other degrees of freedom. This also created a greater cognitive burden on the subject as they tried to control multiple degrees of freedom with coupled dynamics. A low-level controller, with high-level inputs from the decoded neural signals, which controlled the arm as a complete system with knowledge of the coupled dynamics could allow for fewer failures. Additionally, in individuals where stimulation of the muscle in the shoulder produces tangible muscle contractions, removal of the robotic arm support could significantly decrease the size, complexity, and power requirements of the system.

The Cleveland FES Center IST-12 system is a surgically implanted system for controlling the arm and shoulder of an individual with SCI [15, 20]. The system has 12 stimulation channels to send stimulation to the muscles and two EMG channels for recording signals that can be used for control (see Fig. 3). Communication to the device is via a radio-frequency link in the subject's abdomen. When using the system, subjects typically require a passive mobile arm support because their stimulated shoulder muscles are unable to support the arm against gravity. However, all motion is actively controlled by FES and the

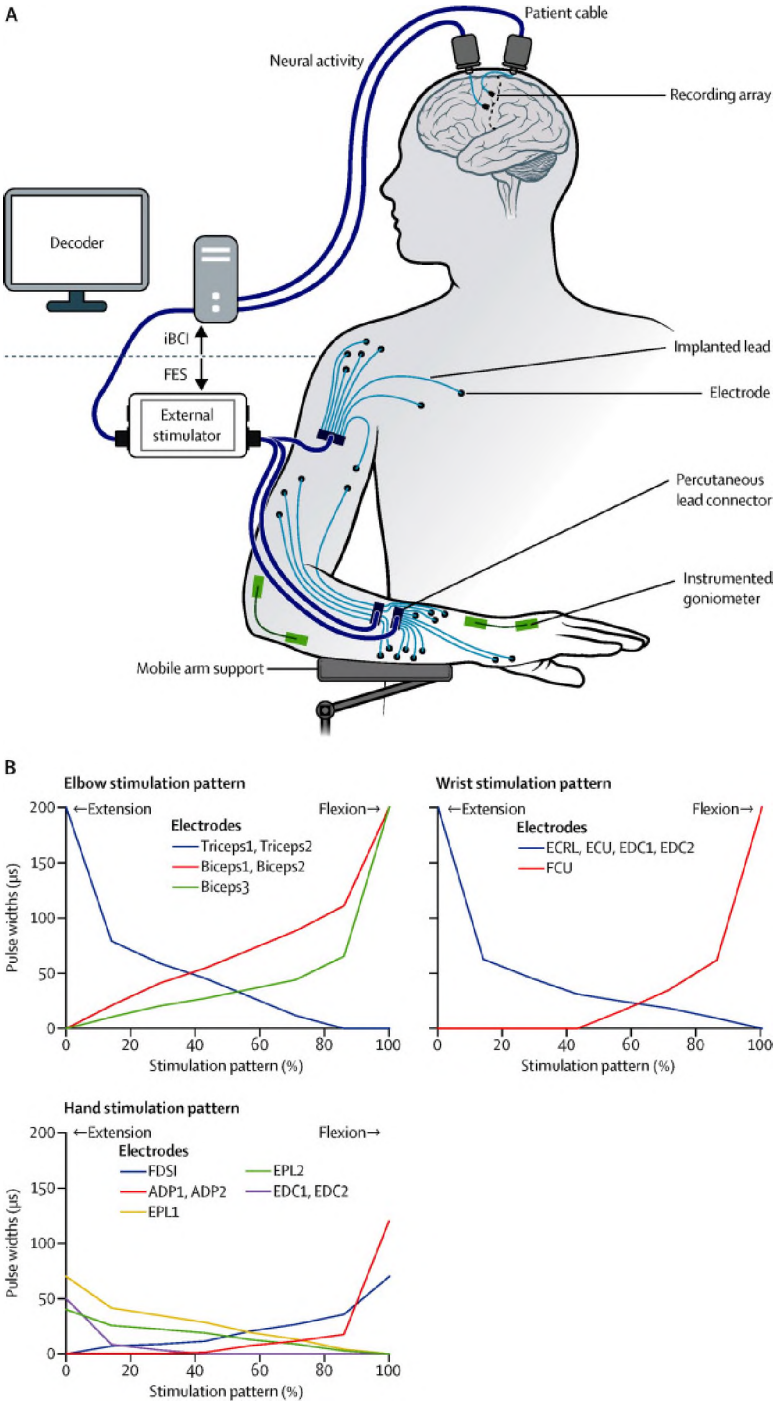



Figure 2: (A) Experimental setup for the BrainGate2 clinical trial a robotic arm support for controlling the shoulder motion and percutaneous FES electrodes for controlling elbow flexion/extension and wrist/hand motions. (B) Stimulation patterns from which the neural signals are decoded to select a desired stimulation to drive the arm to a desired position. Image is from [16] and is used with permission

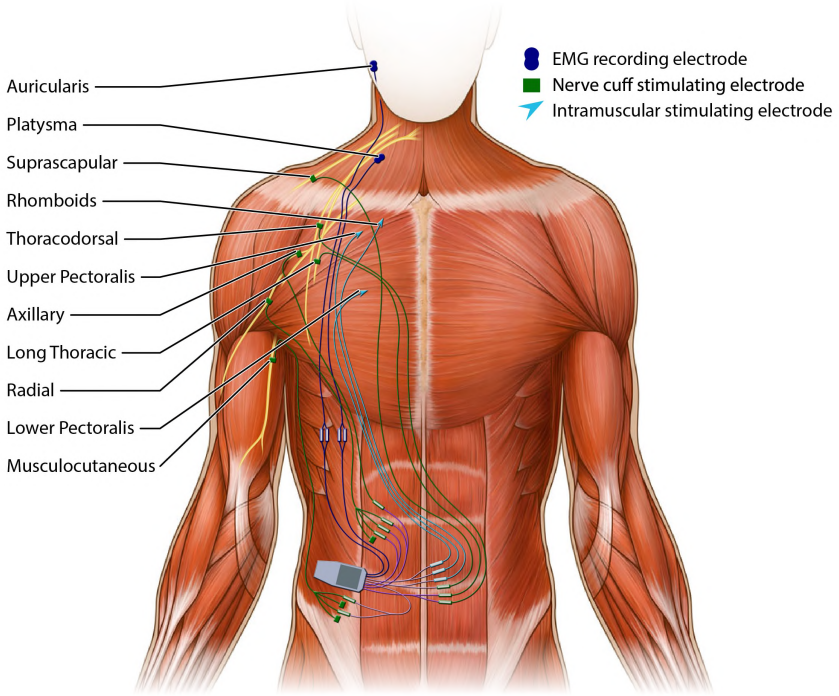



Figure 3: Illustration of the IST-12 system showing the muscles and nerves which can be stimulated by the device. Image is courtesy of the Cleveland FES Center.

device can control the arm as a complete system. Reaching motions have been achieved in two subjects with tetraplegia via a set of preprogrammed reaching motions. These motions were designed in simulation and then tuned by an engineer to account for differences between the simulation and real subject. The preprogrammed stimulation patterns were a great proof of concept that the shoulder and elbow can be completely controlled by FES. However, preprogrammed patterns are not feasible to complete the large number of possible reaching patterns necessary for every day life. Even if all motions could be developed in simulation, the requirement of an engineer to tune the patterns to account for the unique muscular characteristics of each individual with tetraplegia makes it an overwhelming task to apply this technique for an everyday implementation of FES-driven reaching. For practical implementation of an FES-driven reaching controller, the control scheme must be able to achieve novel reaching motions that it has not been explicitly trained on.

The use of model learning techniques to develop subject-specific models of a subject's muscular abilities and their arm's response to electrical stimulation has been shown to be

one of the most promising techniques to control the complete arm system. These methods require a model-learning phase where the electrical stimulation is mapped to either the force or torque produced by each muscle group. Razavian and McPhee [46–48] used artificial neural networks to create a hand position dependent mapping of the stimulation of muscles to the isometric 2D force produced. Their system uses this model to determine the surface stimulation needed to activate five muscles (anterior deltoid, posterior deltoid, biceps, triceps, and pectoralis major) to drive the shoulder and elbow to create desired positioning of the subject's hand in a 2D planar task-space. The models are used to decompose the muscle forces at each point into muscle synergies which produce force that span the space. A feedback controller is used to select a desired force and a linear combination of muscle synergies is selected to produce that force. The system produced reaches with a tracking error between 1-4 cm for planar motions. Additionally, the controller was implemented in healthy subjects which do not exhibit the aforementioned difficult issues in muscle actuation seen in subjects with SCI. Expanding this controller to 3D reaching motions in individuals with SCI significantly increases the complexity of the problem and has yet to be reported as successfully implemented.

In prior work from my own lab, they have also worked to develop subject-specific models of a subject's arm when driven with FES [49–51]. Working with the IST-12 system, they have used semiparametric Gaussian Process Regression (GPR) models to map the electrical stimulation input to the amount of isometric torque and/or force produced by each muscle group. The models have successfully been used to control endpoint forces in the subject's workspace. Semiparametric GPR models have also been developed to model the dynamic torques produced by stimulating the muscles of the arm [52]. However, due to the muscle weakness and denervation seen in subjects with SCI, planning trajectories with these dynamic models was difficult due to there being many states which were uncontrollable. In [52], uncontrollable states were solved using a robot, but for daily reaching motions without robotic assistance, the controllability of the system at each state was be explic-

itly accounted for. To date, none of these techniques have been successful at controlling complete arm, 3D reaching motions in individuals with high tetraplegia.

## 1.4 Research goals

The overall goal of the research presented in this dissertation is to build upon the previous successes of model-based FES controllers and develop a subject-specific model based controller that is capable of achieving accurate complete-arm FES-controlled reaching motions in individuals with high tetraplegia due to SCI. The research is broken into two main aims:

**Aim 1:** Develop a complete-arm FES-driven reaching controller that is capable of holding static hand positions for an individual with high tetraplegia due to SCI.

In this aim, I demonstrate the effectiveness of using a subject specific model based feedback controller to hold the arm in a static position. Previous controllers have demonstrated the ability to achieve desired torques and forces [52, 53]. Additionally, many controllers have demonstrated the ability to control single joints [31] or a couple joints in a rehabilitation setting [33]. However, to date, there has not been a controller which controls both the shoulder and elbow degrees of freedom to achieve 3D hand positions in an individual with tetraplegia. To complete this aim, we used model-learning methods to identify a configuration dependent model that mapped the muscle activation to the amount of force/torque produced by the muscles. We then used this model as the basis of an open-loop controller to hold 3D static wrist positions. After verifying that the open-loop controller, we added feedback via a PID controller to improve the performance throughout the workspace.

**Aim 2:** Develop a subject-specific model-based control strategy to use FES to drive the arm of an individual with high tetraplegia due to SCI along a desired path in the subject's workspace.

In this aim, I present a novel control strategy that uses the subject-specific model presented in Aim 1 as the basis of a control strategy for 3D reaching motions. To date, there

has been limited success in controlling 3D reaching motions with FES especially when accounting for the difficult muscular dynamics associated with individuals with SCI. Most examples have required robotic assistance to control the shoulder [16, 43], worked in a rehabilitation environment where the reaching tasks are repetitive [33], or have been planar motions with healthy subjects [48]. In the work presented here, I demonstrate the need to plan trajectories to ensure controllability along the desired path by accounting for the subject-specific muscular capabilities of the individual who is being stimulated. I then compare a feedforward-feedback control scheme to a model predictive control scheme in a simulation of the arm. Finally, I implemented the model predictive controller in an individual with high tetraplegia due to SCI and demonstrate its capability in controlling the shoulder and elbow through 3D reaching motions.

## 1.5 Outline

This dissertation is divided into nine chapters.

- *Chapter I* discusses the background, motivation, and main aims of this dissertation.
- *Chapter II* presents a summary of the main methods used in the dissertation.
- *Chapters III-IV* address Aim 1 of the dissertation.
  - *Chapter III* presents an open-loop control strategy for holding static 3D wrist positions. The results validate the subject-specific model of the arm and its response to stimulation and demonstrate the feasibility to use the models to control full-arm reaching motions.
  - *Chapter IV* presents a feedback control strategy for full-arm control of static 3D wrist positions. The results demonstrate the feasibility to use a subject-specific model the arm and its response to stimulation as the basis of a controller capable of achieving 3D reaching motions in an individual with high tetraplegia due to SCI.

- *Chapters V-VI* address a series of practical improvements that led to the development of the final controller presented in this dissertation.
  - *Chapter V* compares a feedforward-feedback controller with a feedback only controller to drive the arm to a desired wrist position without a planned path. The large overshoot and oscillations, along with relatively low accuracy led to the need to plan paths. Additionally, the large overshoot of the feedforward-feedback controller began to demonstrate the need for an open-loop controller which directly accounts for the dynamics of the system. To build on these results, we then present a simulation study to compare controlling the arm using a quasi-static path. The results again demonstrated the need for path planning. Additionally, due to the nonlinearities and time delays inherent to an FES driven system, derivative gain was shown to increase oscillation and instead it was shown that adding damping to the mobile arm support would create a more suitable environment for accurate reaching.
  - *Chapter VI* presents an attempt at completing straight line paths in a damped environment. The overall results demonstrate the need for smarter path planning than straight line paths as the arm was unable to reach all sections of the workspace with equal accuracy (notably with low accuracy to the right half of the workspace). Two additional advances to arise from this chapter: 1) development of a faster, day-of modeling procedure which finds a subject-specific model in approximately 30 minutes by using the previously trained model hyperparameters, and 2) the need for and a first attempt at handling feedback overcompensation, when the controller asks for more torque than the muscles can produce which can be a common problem in individuals with SCI due to increased muscle weakness.
- *Chapter VII* addresses Aim 2 of the dissertation. It presents a simulation study which

uses trajectory optimization methods to plan feasible trajectories and muscle activation patterns that account for the specific capabilities of the subjects muscles. A feedforward-feedback controller and a model predictive controller are then used to control the arm through the desired trajectory. Using the methods developed in simulation, we then practically implement a model predictive control scheme to control full-arm 3D reaching motions in an individual with high tetraplegia due to spinal cord injury.

- *Chapter VIII* concludes to the dissertation and points to future directions of research.

## 1.6 REFERENCES


[1] National Spinal Cord Injury Statistical Center, Facts and Figures at a Glance. *Birmingham, AL: University of Alabama at Birmingham*, 2020.

[2] Kim D. Anderson. Targeting Recovery: Priorities of the Spinal Cord-Injured Population. *Journal of Neurotrauma*, 21(10):1371–1383, oct 2004. ISSN 0897-7151. doi: 10.1089/neu.2004.21.1371.

[3] Nuray Yozbatiran and Gerard E Francisco. Robot-assisted Therapy for the Upper Limb after Cervical Spinal Cord Injury. *Physical medicine and rehabilitation clinics of North America*, 30(2):367–384, May 2019. ISSN 1558-1381 (Electronic). doi: 10.1016/j.pmr.2018.12.008.

[4] Michael W. Keith, Kevin L. Kilgore, P. Hunter Peckham, Kathryn Stroh Wuolle, Graham Creasey, and Michel Lemay. Tendon transfers and functional electrical stimulation for restoration of hand function in spinal cord injury. *The Journal of Hand Surgery*, 21(1):89–99, Jan 1996. ISSN 03635023. doi: 10.1016/S0363-5023(96) 80160-2.

[5] Jennifer A. Dunn, K. Anne Sinnott, Alastair G. Rothwell, Khalid D. Mohammed, and Jeremy W. Simcock. Tendon Transfer Surgery for People With Tetraplegia: An Overview. *Archives of Physical Medicine and Rehabilitation*, 97(6):S75–S80, Jun 2016. ISSN 00039993. doi: 10.1016/j.apmr.2016.01.034.

[6] Natasha Van Zyl, Jodie B. Hahn, Catherine A. Cooper, Michael D. Weymouth, Stephen J. Flood, and Mary P. Galea. Upper limb reinnervation in C6 tetraplegia using a triple nerve transfer: Case report. *Journal of Hand Surgery*, 39(9):1779–1783, Sep 2014. ISSN 15316564. doi: 10.1016/j.jhsa.2014.06.017.

[7] R. A.R.C. Gopura, D. S.V. Bandara, Kazuo Kiguchi, and G. K.I. Mann. Developments in hardware systems of active upper-limb exoskeleton robots: A review. *Robotics and*

*Autonomous Systems*, 75:203–220, Jan 2016. ISSN 09218890. doi: 10.1016/j.robot.2015.10.001.

[8] Arun Kumar and Margaret Frances Phillips. Use of powered mobile arm supports by people with neuromuscular conditions. *Journal of rehabilitation research and development*, 50(1):61, 2013. ISSN 0748-7711.

[9] Leigh R. Hochberg, Daniel Bacher, Beata Jarosiewicz, Nicolas Y. Masse, John D. Simeral, Joern Vogel, Sami Haddadin, Jie Liu, Sydney S. Cash, Patrick Van Der Smagt, and John P. Donoghue. Reach and grasp by people with tetraplegia using a neurally controlled robotic arm. *Nature*, 485(7398):372–375, 2012. ISSN 00280836. doi: 10.1038/nature11076.

[10] Nathan Dunkelberger, Eric M. Schearer, and Marcia K. O'Malley. A review of methods for achieving upper limb movement following spinal cord injury through hybrid muscle stimulation and robotic assistance. *Experimental Neurology*, 328(February): 113274, Jun 2020. ISSN 00144886. doi: 10.1016/j.expneurol.2020.113274.

[11] P. Hunter Peckham and Jayme S. Knutson. Functional Electrical Stimulation for Neuromuscular Applications. *Annual Review of Biomedical Engineering*, 7(1):327–360, 2005. ISSN 1523-9829. doi: 10.1146/annurev.bioeng.6.040803.140103.

[12] Lynne R. Sheffler and John Chae. Neuromuscular electrical stimulation in neurorehabilitation. *Muscle & Nerve*, 35(5):562–590, 2007. ISSN 0148639X. doi: 10.1002/mus.20758.

[13] Rubia Van den Brand, Janine Heutschi, Quentin Barraud, Jack DiGiovanna, Kay Bartholdi, Michèle Huerlimann, Lucia Friedli, Isabel Vollenweider, Eduardo Martin Moraud, and Simone Duis. Restoring voluntary control of locomotion after paralyzing spinal cord injury. *science*, 336(6085):1182–1185, 2012. ISSN 0036-8075.

[14] Kevin L. Kilgore, Harry A. Hoyen, Anne M. Bryden, Ronald L. Hart, Michael W. Keith, and P. Hunter Peckham. An Implanted Upper-Extremity Neuroprosthesis Using Myoelectric Control. *The Journal of Hand Surgery*, 33(4):539–550, Apr 2008. ISSN 03635023. doi: 10.1016/j.jhsa.2008.01.007.

[15] William D. Memberg, Katharine H. Polasek, Ronald L. Hart, Anne M. Bryden, Kevin L. Kilgore, Gregory A. Nemunaitis, Harry A. Hoyen, Michael W. Keith, and Robert F. Kirsch. Implanted neuroprosthesis for restoring arm and hand function in people with high level tetraplegia. *Archives of Physical Medicine and Rehabilitation*, 95(6):1201–1211.e1, Jun 2014. ISSN 1532821X. doi: 10.1016/j.apmr.2014.01.028.

[16] A. Bolu Ajiboye, Francis R. Willett, Daniel R. Young, William D. Memberg, Brian A. Murphy, Jonathan P. Miller, Benjamin L. Walter, Jennifer A. Sweet, Harry A. Hoyen, Michael W. Keith, P. Hunter Peckham, John D. Simeral, John P. Donoghue, Leigh R. Hochberg, and Robert F. Kirsch. Restoration of reaching and grasping movements through brain-controlled muscle stimulation in a person with tetraplegia: a proof-of-concept demonstration. *The Lancet*, 389(10081):1821–1830, May 2017. ISSN 01406736. doi: 10.1016/S0140-6736(17)30601-3.

[17] Dejan B Popović and Mirjana B Popović. Automatic determination of the optimal shape of a surface electrode: selective stimulation. *Journal of neuroscience methods*, 178(1):174–181, 2009. ISSN 0165-0270.

[18] M J Mulcahey, B T Smith, and R R Betz. Evaluation of the lower motor neuron integrity of upper extremity muscles in high level spinal cord injury. *Spinal Cord*, 37 (8):585–591, 1999. ISSN 1476-5624.

[19] Ryan J. Downey, Matthew J. Bellman, Hiroyuki Kawai, Chris M. Gregory, and Warren E. Dixon. Comparing the Induced Muscle Fatigue Between Asynchronous and Synchronous Electrical Stimulation in Able-Bodied and Spinal Cord Injured Popula-

tions. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 23(6): 964–972, 2015. ISSN 15344320. doi: 10.1109/TNSRE.2014.2364735.

[20] Ronald L. Hart, Niloy Bhadra, Fred W. Montague, Kevin L. Kilgore, and P. Hunter Peckham. Design and testing of an advanced implantable neuroprosthesis with myoelectric control. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 19(1):45–53, 2011. ISSN 15344320. doi: 10.1109/TNSRE.2010.2079952.

[21] Raviraj Nataraj, Musa L. Audu, and Ronald J. Triolo. Restoring standing capabilities with feedback control of functional neuromuscular stimulation following spinal cord injury. *Medical Engineering & Physics*, 42(1):13–25, Apr 2017. ISSN 13504533. doi: 10.1016/j.medengphy.2017.01.023.

[22] Vahab Nekoukar and Abbas Erfanian. A decentralized modular control framework for robust control of FES-activated walker-assisted paraplegic walking using terminal sliding mode and fuzzy logic control. *IEEE transactions on biomedical engineering*, 59(10):2818–2827, 2012. ISSN 0018-9294.

[23] Elizabeth Hardin, Rudi Kobetic, Lori Murray, Michelle Corado-Ahmed, Gilles Pinault, Jonathan Sakai, Stephanie Nogan Bailey, Chester Ho, and Ronald J. Triolo. Walking after incomplete spinal cord injury using an implanted FES system: A case report. *Journal of Rehabilitation Research & Development*, 44(3):333–346, 2007. ISSN 07487711. doi: 10.1682/JRRD.2006.02.0013.

[24] Naji A. Alibeji, Nicholas Andrew Kirsch, and Nitin Sharma. A Muscle Synergy-Inspired Adaptive Control Scheme for a Hybrid Walking Neuroprosthesis. *Frontiers in Bioengineering and Biotechnology*, 3(December):1–13, 2015. ISSN 2296-4185. doi: 10.3389/fbioe.2015.00203.

[25] J Hobby, P N Taylor, and J Esnouf. Restoration of tetraplegic hand function by use

of the neurocontrol freehand system. *Journal of Hand Surgery*, 26(5):459–464, 2001. ISSN 0266-7681.

[26] Dejan B Popović. Advances in functional electrical stimulation (FES). *Journal of Electromyography and Kinesiology*, 24(6):795–802, 2014. ISSN 1050-6411. doi: https://doi.org/10.1016/j.jelekin.2014.09.008.

[27] C L Lynch and M R Popovic. Closed-loop control for FES: Past work and future directions. In *10th Annual Conference of the International FES Society*, pages 2–4, 2005.

[28] Naji Alibeji, Nicholas Kirsch, Shawn Farrokhi, and Nitin Sharma. Further Results on Predictor-Based Control of Neuromuscular Electrical Stimulation. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 23(6):1095–1105, Nov 2015. ISSN 1534-4320. doi: 10.1109/TNSRE.2015.2418735.

[29] Emilia Ambrosini, Simona Ferrante, Thomas Schauer, Giancarlo Ferrigno, Franco Molteni, and Alessandra Pedrocchi. Design of a Symmetry Controller for Cycling Induced by Electrical Stimulation: Preliminary Results on Post-Acute Stroke Patients. *Artificial Organs*, 34(8):663–667, Aug 2010. ISSN 0160-564X. doi: https://doi.org/10.1111/j.1525-1594.2009.00941.x.

[30] E. Ambrosini, S. Ferrante, J. Zajc, M. Bulgheroni, W. Baccinelli, E. D'Amico, T. Schauer, C. Wiesener, M. Russold, M. Gfoehler, M. Puchinger, M. Weber, S. Becker, K. Krakow, M. Rossini, D. Proserpio, G. Gasperini, F. Molteni, G. Ferrigno, and A. Pedrocchi. The combined action of a passive exoskeleton and an EMG-controlled neuroprosthesis for upper limb stroke rehabilitation: First results of the RETRAINER project. In *2017 International Conference on Rehabilitation Robotics (ICORR)*, pages 56–61. IEEE, Jul 2017. ISBN 978-1-5386-2296-4. doi: 10.1109/ICORR.2017.8009221.

[31] Antônio Padilha L. Bó, Lucas O. da Fonseca, and Ana Carolina C. de Sousa. FES-induced co-activation of antagonist muscles for upper limb control and disturbance rejection. *Medical Engineering & Physics*, 38(11):1176–1184, Nov 2016. ISSN 13504533. doi: 10.1016/j.medengphy.2016.07.004.

[32] Christian A. Cousin, Courtney A. Rouse, Victor H. Duenas, and Warren E. Dixon. Position and torque control via rehabilitation robot and functional electrical stimulation. In *2017 International Conference on Rehabilitation Robotics (ICORR)*, pages 38–43. IEEE, Jul 2017. ISBN 978-1-5386-2296-4. doi: 10.1109/ICORR.2017.8009218.

[33] Christopher T Freeman. Upper Limb Electrical Stimulation Using Input-Output Linearization and Iterative Learning Control. *IEEE Transactions on Control Systems Technology*, 23(4):1546–1554, Jul 2015. ISSN 1063-6536. doi: 10.1109/TCST.2014.2363412.

[34] Katie L Meadmore, Timothy A Exell, Emma Hallewell, Ann-Marie Hughes, Chris T Freeman, Mustafa Kutlu, Valerie Benson, Eric Rogers, and Jane H Burridge. The application of precisely controlled functional electrical stimulation to the shoulder, elbow and wrist for upper limb stroke rehabilitation: a feasibility study. *Journal of NeuroEngineering and Rehabilitation*, 11(1):105, 2014. ISSN 1743-0003. doi: 10.1186/1743-0003-11-105.

[35] Mustafa Kutlu, Chris Freeman, Ann-Marie Hughes, and Matthew Spraggs. A Home-based FES System for Upper-limb Stroke Rehabilitation with Iterative Learning Control. *IFAC-PapersOnLine*, 50(1):12089–12094, 2017. ISSN 24058963. doi: 10.1016/j.ifacol.2017.08.2153.

[36] L. LAN, K. Y. ZHU, and C. Y. WEN. *Journal of Mechanics in Medicine and Biology*, December . ISSN 0219-5194. doi: 10.1142/S0219519409003127.

[37] Kathleen M. Jagodnik and Antonie J. van den Bogert. Optimization and evaluation of

a proportional derivative controller for planar arm movement. *Journal of Biomechanics*, 43(6):1086–1091, Apr 2010. ISSN 00219290. doi: 10.1016/j.jbiomech.2009.12.017.

[38] Kathleen M. Jagodnik, Dimitra Blana, Antonie J. van den Bogert, and Robert F. Kirsch. An optimized proportional-derivative controller for the human upper extremity with gravity. *Journal of Biomechanics*, 48(13):3692–3700, oct 2015. ISSN 00219290. doi: 10.1016/j.jbiomech.2015.08.016.

[39] Kathleen M. Jagodnik, Philip S. Thomas, Antonie J. Van Den Bogert, Michael S. Branicky, and Robert F. Kirsch. Training an Actor-Critic Reinforcement Learning Controller for Arm Movement Using Human-Generated Rewards. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 25(10):1892–1905, oct 2017. ISSN 15344320. doi: 10.1109/TNSRE.2017.2700395.

[40] Dimitra Blana, Robert F. Kirsch, and Edward K. Chadwick. Combined feedforward and feedback control of a redundant, nonlinear, dynamic musculoskeletal system. *Medical & Biological Engineering & Computing*, 47(5):533–542, May 2009. ISSN 0140-0118. doi: 10.1007/s11517-009-0479-3.

[41] P. Cooman and R. F. Kirsch. Control of a time-delayed 5 degrees of freedom arm model for use in upper extremity functional electrical stimulation. In *2012 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pages 322–324. IEEE, Aug 2012. ISBN 978-1-4577-1787-1. doi: 10.1109/EMBC.2012.6345934.

[42] Yu-Wei Liao, Eric M. Schearer, Eric J. Perreault, Matthew C. Tresch, and Kevin M. Lynch. Multi-muscle FES control of the human arm for interaction tasks - Stabilizing with muscle co-contraction and postural adjustment: A simulation study. In *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, number Iros,

pages 2134–2139. IEEE, Sep 2014. ISBN 978-1-4799-6934-0. doi: 10.1109/IROS.2014.6942849.

[43] Alessandra Pedrocchi, Simona Ferrante, Emilia Ambrosini, Marta Gandolla, Claudia Casellato, Thomas Schauer, Christian Klauer, Javier Pascual, Carmen Vidaurre, Margit Gföhler, Werner Reichenfelser, Jakob Karner, Silvestro Micera, Andrea Crema, Franco Molteni, Mauro Rossini, Giovanna Palumbo, Eleonora Guanziroli, Andreas Jedlitschka, Marco Hack, Maria Bulgheroni, Enrico D'Amico, Peter Schenk, Sven Zwicker, Alexander Duschau-Wicke, Justinas Miseikis, Lina Graber, and Giancarlo Ferrigno. MUNDUS project: MUltimodal Neuroprosthesis for daily Upper limb Support. *Journal of NeuroEngineering and Rehabilitation*, 10(1):66, 2013. ISSN 1743-0003. doi: 10.1186/1743-0003-10-66.

[44] Christian Klauer, Thomas Schauer, Werner Reichenfelser, Jakob Karner, Sven Zwicker, Marta Gandolla, Emilia Ambrosini, Simona Ferrante, Marco Hack, Andreas Jedlitschka, Alexander Duschau-Wicke, Margit Gföhler, and Alessandra Pedrocchi. Feedback control of arm movements using Neuro-Muscular Electrical Stimulation (NMES) combined with a lockable, passive exoskeleton for gravity compensation. *Frontiers in Neuroscience*, 8(Sep):1–16, 2014. ISSN 1662453X. doi: 10.3389/fnins.2014.00262.

[45] A Bolu Ajiboye, Leigh R Hochberg, and Robert F Kirsch. Restoring Functional Reach-to-Grasp in a Person with Chronic Tetraplegia Using Implanted Functional Electrical Stimulation and Intracortical Brain-Computer Interfaces BT - Brain–Computer Interface Research: A State-of-the-Art Summary 8. pages 35–45. Springer International Publishing, Cham, 2020. ISBN 978-3-030-49583-1. doi: 10.1007/978-3-030-49583-1_4.

[46] Reza Sharif Razavian, Naser Mehrabi, and John McPhee. A model-based approach to predict muscle synergies using optimization: application to feedback control. *Fron-*

*tiers in Computational Neuroscience*, 9(October):1–13, 2015. ISSN 1662-5188. doi: 10.3389/fncom.2015.00121.

[47] Reza Sharif Razavian, Borna Ghannadi, and John McPhee. Feedback control of functional electrical stimulation for arbitrary upper extremity movements. In *2017 International Conference on Rehabilitation Robotics (ICORR)*, pages 1451–1456. IEEE, Jul 2017. ISBN 978-1-5386-2296-4. doi: 10.1109/ICORR.2017.8009452.

[48] Reza Sharif Razavian, Borna Ghannadi, Naser Mehrabi, Mark Charlet, and John McPhee. Feedback Control of Functional Electrical Stimulation for 2-D Arm Reaching Movements. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 26(10):2033–2043, oct 2018. ISSN 1534-4320. doi: 10.1109/TNSRE.2018.2853573.

[49] Eric M. Schearer, Yu-Wei Liao, Eric J. Perreault, Matthew C. Tresch, William D. Memberg, Robert F. Kirsch, and Kevin M. Lynch. System identification for 3D force control of a human arm neuroprosthesis using functional electrical stimulation. *2012 IEEE International Conference on Robotics and Automation*, pages 3698–3705, 2012. ISSN 10504729. doi: 10.1109/ICRA.2012.6224981.

[50] Eric M. Schearer, Yu Wei Liao, Eric J. Perreault, Matthew C. Tresch, William D. Memberg, Robert F. Kirsch, and Kevin M. Lynch. Multi-muscle FES force control of the human arm for arbitrary goals. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 22(3):654–663, 2014. ISSN 15344320. doi: 10.1109/TNSRE.2013.2282903.

[51] Eric M. Schearer, Yu Wei Liao, Eric J. Perreault, Matthew C. Tresch, William D. Memberg, Robert F. Kirsch, and Kevin M. Lynch. Evaluation of a semi-parametric model for high-dimensional FES control. *International IEEE/EMBS Conference on*

*Neural Engineering, NER*, 2015-July(April):304–307, 2015. ISSN 19483554. doi: 10.1109/NER.2015.7146620.

[52] Eric M. Schearer, Yu-Wei Liao, Eric J. Perreault, Matthew C. Tresch, William D. Memberg, Robert F. Kirsch, and Kevin M. Lynch. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, December . ISSN 1534-4320. doi: 10.1109/TNSRE.2016.2535348.

[53] Eric M. Schearer, Yu-Wei Liao, Eric J. Perreault, Matthew C. Tresch, and Kevin M. Lynch. Optimal sampling of recruitment curves for functional electrical stimulation control. In *2012 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, number 1, pages 329–332. IEEE, Aug 2012. ISBN 978-1-4577-1787-1. doi: 10.1109/EMBC.2012.6345936.

# CHAPTER II

# A DESCRIPTION OF THE METHODS USED IN THIS DISSERTATION

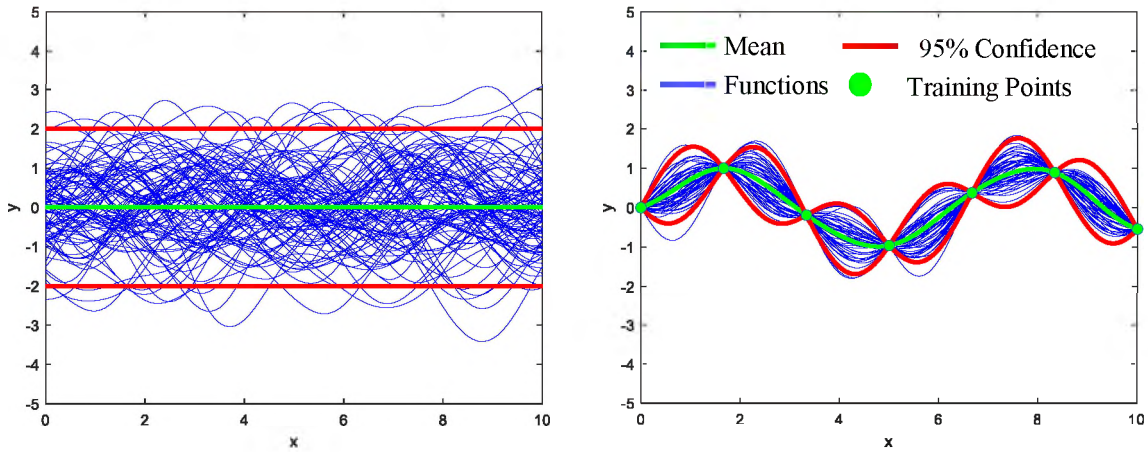
*This chapter serves to provide the reader with background knowledge of the underlying methodology for the research presented in this dissertation. The research in this dissertation uses model-learning methods including Gaussian process regression to learn the statics of the subject's arm and its response to stimulation. Trajectory optimization is used to find feasible trajectories while accounting for the subject-specific muscle capabilities and arm dynamics. Model predictive control is then used to control the arm through FES-driven reaching motions. In this chapter, we summarize these techniques which are the basis of the dissertation research.*

Throughout the research presented in this dissertation, three main techniques are used to model and control the arm: 1. Gaussian process regression, 2. Trajectory optimization, and 3. model predictive control. We present a brief description of each of these methods here.

## 2.1 Gaussian process regression

Throughout this dissertation, we train Gaussian process regression (GPR) models of the arm's statics and response to electrical stimulation. GPR is a "black-box", Bayesian machine learning method which creates a mapping from a query input, $\boldsymbol{x}_q$ (which in our work is the arm configuration or position), to the predicted output, $\boldsymbol{f}_*$ (which in our models is the torque or force measured by the robot), given a set of training inputs, $\boldsymbol{X}$, and training outputs, $\boldsymbol{y}$. We use GPR over other nonparametric methods, such as artificial neural notworks, due to the automated nature of determining the complexity of the model by maximizing the marginal likelihood (see [1] for details on the quality of the model). Additionally, GPR offers an explicit calculation of the uncertainty of the model at given locations in the input space which may be useful for model update techniques or in determining better system identification methods to better map the entire workspace. The methods presented here are implemented using the GPML MATLAB toolbox which is based on [2]. I present a brief summary of GPR here to provide background to the reader.

For a general regression problem, we select a subset of functions and attempt to determine the parameters that best fit the data. Especially when considering complex functions for which the expected shape of the function is difficult to predict, this technique can lead to the regression model not being flexible enough to encompass all the detail of the function if the selected function set is not complex enough or can lead to overfitting if the selected function is more complex than that underlying data. GPR offers a solution to this issue in that it considers all possible functions and then weights the functions which are more likely based on the training data. To consider all possible functions, Gaussian processes are used.




Prior distribution

Posterior Distribution

Figure 4: This image shows an example of Gaussian process regression with sample data. The prior model uses a mean function of zero and squared exponential covariance. Once conditioned on the training data (shown with zero noise), the posterior mean is drawn to the training outputs and the uncertainty at points near the training data shrinks.

Gaussian processes are defined as "a collection of random variables, any finite number of which have a joint Gaussian distribution." GPR models create a joint distribution based on the training data to determine a posterior distribution which can be used to predict the value of the function. The act of predicting an output for a query is simply completed by conditioning the joint Gaussian distribution on the observations. Fig. 4 illustrates this technique. The prior distribution is often considered to have a mean of zero. By conditioning the prior distribution on the training observations, the posterior distribution includes only data which fits the training data. The mean of the function posterior distribution corresponds to the predicted output.

Mathematically, the joint distribution of the training observations with system noise, $(\boldsymbol{X}, \boldsymbol{y})$, with Gaussian noise of variance $\sigma_n^2$ and the query data points, $(\boldsymbol{X}_*, \boldsymbol{f}_*)$, is described by

$$\begin{bmatrix} \boldsymbol{y} \\ \boldsymbol{f}_* \end{bmatrix} \sim \mathcal{N}\left( 0 \quad \begin{bmatrix} \boldsymbol{K}(\boldsymbol{X}, \boldsymbol{X}) + \sigma_n^2 \boldsymbol{I}) & \boldsymbol{K}(\boldsymbol{X}, \boldsymbol{X}_*) \\ \boldsymbol{K}(\boldsymbol{X}_*, \boldsymbol{X}) & \boldsymbol{K}(\boldsymbol{X}_*, \boldsymbol{X}_*) \end{bmatrix} \right) \tag{2.1}$$

where $\boldsymbol{K}(\boldsymbol{X}, \boldsymbol{X})$ represents the covariance matrix for the training inputs with themselves, $\boldsymbol{K}(\boldsymbol{X}, \boldsymbol{X}_*)$ is the covariance matrix of the training inputs and the query inputs, and $\boldsymbol{K}(\boldsymbol{X}_*, \boldsymbol{X}_*)$ is the covariance matrix of the query inputs with themselves. By conditioning the distribution on the training data, the query outputs, $\boldsymbol{f}_*$ can be predicted by

$$\boldsymbol{f}_* = \boldsymbol{K}(\boldsymbol{X}_*, \boldsymbol{X})[\boldsymbol{K}(\boldsymbol{X}, \boldsymbol{X}) + \sigma_n^2 \boldsymbol{I})]^{-1} \boldsymbol{y}. \tag{2.2}$$

The covariance matrix, $\boldsymbol{K}$, represents the relationship between outputs of the function relative to inputs. The standard covariance function used in this dissertation is the squared exponential covariance function where each element of $\boldsymbol{K}$ is defined by

$$k(x, x') = p_1 e^{\frac{(x-x')^T (x-x')}{2p_2^2}} \tag{2.3}$$

where $p_1$ is the vertical scale and $p_2$ is the length scale. These parameters are known as hyperparameters, and they determine the effect on the predicted output the training data has. The vertical scale determines how much the function can vary in magnitude. The length scale affects how quickly the function can vary. Practically, the length scale weights which training observations are "close enough" to effect the prediction. The act of training a GPR model is to determine the optimal hyperparameters to best make predictions. To determine the optimal hyperparameters, we select the hyperparameters which minimize the log marginal likelihood which balances model complexity with fitting the training data. In our experiments, we often use the GPML toolbox's *squared exponential with automatic relevance detection* covariance function. This function includes a separate length scale for each dimension of the input space.

We also use semiparametric models in this dissertation [1]. These models follow much the the nonparametric GPR method described above, but they incorporate an explicit basis function to the prediction of the model. This parametric model is often a simple linear model and allows predictions to be made in areas of the workspace with minimal training

data. The nonparametric GPR model is then trained on the difference between the parametric model and the training observations. By simply adding the parametric model prediction with the GPR prediction, a predicted output can be found for any query point.

## 2.2 Trajectory optimization

To find feasible reaching trajectories, we use the trajectory optimization methods developed in [3]. I present a summary of those methods here.

The method of direct collocation is used to solve the trajectory optimization. Direct collocation transforms the dynamics problem into a constrained nonlinear optimization problem. As opposed to the shooting method which optimizes the initial conditions and open-loop controls of the trajectory, direct collocation attempts to optimize the entire state and control trajectory at one time. This allows for a more computationally efficient solution. In this method, the trajectory is discretized over $n$ nodes equally spaced in the time domain.

The trajectory optimization seeks to calculate the state of the system (joint angles and joint velocities in this dissertation), and input to the system (in our case, muscle activations), at each node. An objective function, $f(\boldsymbol{x}, \boldsymbol{u})$, is developed where $\boldsymbol{x}$ represents that vector of states at all nodes and $\boldsymbol{u}$ represents the vector of inputs at all nodes.

The constraints of the optimization problem must then be defined. The state and input to the system can be bounded. Additionally, task constraints are defined. For example, these constraints include starting and ending at the desired arm configuration. The final requirement for trajectory optimization is the addition of dynamics constraints. A model of the system dynamics is required for this constraint. Between each node, the semi-implicit Euler method is used to predict the dynamics of the system based on the state and inputs at the nodes. The dynamics constraint ensures that the predicted state at each node based on the state, dynamics, and input need to be consistent with the dynamical model.

Once the problem is defined, IPOPT [4] was used to solve the constrained nonlinear program for the optimal trajectory. The optimal trajectory consists of the states and in-

puts at each node which minimize the objective function while meeting the bounds, task constraints, and dynamical constraints.

## 2.3 Model predictive control

Model predictive control (MPC) is a powerful tool for controlling dynamical systems. It seeks to find the optimal control inputs which best produce the desired trajectory based on a model of the system dynamics. With an accurate model, MPC is especially well suited to handle situations where there are system constraints on the inputs or states because those constraints can be explicitly defined in the optimization problem. It is this ability to handle input constraints which makes it a valuable tool in controlling reaching motions in individuals with spinal cord injuries. The methods used in this dissertation are developed in [5]. I present a summary for completeness here.

For a given state, $x$, and input, $u$, at time $t = k$ and a discretized state-space model of the system dynamics, $A, B, C, D$, the next state and output of the system at $t = k + 1$ of the system can be predicted by

$$x(k+1) = Ax(k) + Bu(k) \tag{2.4}$$

$$y(k) = Cx(k) + Du(k) \tag{2.5}$$

. For the controller developed in this dissertation, it is assumed that $D = 0$. Model predictive control uses these equations to predict the state of the system based on the inputs applied. To add integral action to the controller, the state is augmented to include the current control input and the state-space matrices are augmented as well. The new control input is defined as the change in control input from time-step $k$ to $k + 1$, $\Delta u$, and the

augmented state-space system becomes

$$\begin{bmatrix} x(k+1) \\ u(k) \end{bmatrix} = \begin{bmatrix} A & B \\ 0 & I \end{bmatrix} \begin{bmatrix} x(k) \\ u(k-1) \end{bmatrix} + \begin{bmatrix} B \\ I \end{bmatrix} \Delta u(k) \tag{2.6}$$

$$y(k) = [C\ D] \begin{bmatrix} x(k) \\ u(k-1) \end{bmatrix} + D\Delta u(k). \tag{2.7}$$

For a given state and desired trajectory, the state-space matrices are found by linearizing a nonlinear dynamic model of the system about the current state of the system. These matrices are then augmented as above and assumed constant for the control calculations. The controller aims to select the input commands which minimize the objective function

$$J = \sum_{i=1}^{n_y} e(k+i)^T e(k+i) + \lambda \sum_{i=0}^{n_u-1} \Delta u(k+i)^T \Delta u(k+i). \tag{2.8}$$

The first term of the equation minimizes the error, $e(k+i)$, for a given time-step which is defined as the estimated output as calculated by equations (2.7) subtracted from a desired reference trajectory. The prediction horizon, $n_y$, determines for how many time steps forward the model predicts states and system error. In an ideal situation, the controller would use an infinite horizon, however, computational limits require a smaller horizon. The second term minimizes the change in control input and functions as an effort limiting and control input smoothing term. The control horizon, $n_u$, determines the number of time steps forward that the controller optimizes control inputs. For time steps $n_u < i < n_y$, $\Delta u = 0$. The lumped scalar weighting $\lambda$ is used to weight the importance to the optimization of minimizing the change in system input. Increasing $\lambda$ results in less variability in the system inputs.

The optimization problem is formulated as a quadratic program with the required constraints on the inputs and state. When the optimal change in control input for the entire control horizon is determined, the change in control input for the next time step is applied

to the system. The new control input is then calculated by $u(k) = u(k-1) + \Delta u$. The system applies the new control input and the optimization begins again to find the optimal control input for the next time-step.

There are a few limitations of an MPC controller. The accuracy of the model is critical to the controller performance. As described in [5], the MPC controller presented here acts as a state feedback controller with integral action. However, the controller is not robust to large errors in the model. If the nonlinear model is not an accurate representation of the real system or there are large errors produced by linearizing the system, the controller will not produce accurate control. It is possible to perform MPC control directly with the nonlinear model, but the computational burden makes it currently impractical to implement with our system. Even with the linearized state-space matrices, the computational burden can be high as a quadratic program needs to be solved at every time-step. The control and prediction horizons must be tuned to achieve desired performance in the amount of time available between time steps.

## 2.4 REFERENCES


[1] Eric M. Schearer, Yu-Wei Liao, Eric J. Perreault, Matthew C. Tresch, William D. Memberg, Robert F. Kirsch, and Kevin M. Lynch. Semiparametric Identification of Human Arm Dynamics for Flexible Control of a Functional Electrical Stimulation Neuroprosthesis. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 24(12): 1405–1415, Dec 2016. ISSN 1534-4320. doi: 10.1109/TNSRE.2016.2535348.

[2] CE Rasmussen and CKI Williams. *Gaussian processes for machine learning*. 2006.

[3] Marko Ackermann and Antonie J van den Bogert. Optimality principles for model-based prediction of human gait. *Journal of Biomechanics*, 43(6):1055–1060, 2010. ISSN 0021-9290. doi: https://doi.org/10.1016/j.jbiomech.2009.12.012.

[4] Andreas Wächter. An interior point algorithm for large-scale nonlinear optimization with applications in process engineering, 2002.

[5] Hanz Richter. *Advanced control of turbofan engines*. Springer Science & Business Media, 2011. ISBN 1461411718.

# CHAPTER III

# EVALUATING AN OPEN-LOOP FUNCTIONAL ELECTRICAL STIMULATION CONTROLLER FOR HOLDING THE SHOULDER AND ELBOW CONFIGURATION OF A PARALYZED ARM

*Function electrical stimulation is a promising solution to restore reaching motions to individuals with paralyzed limbs due to spinal cord injuries. In this chapter, I present our method of modeling the arm and its response to electrical stimulation. We develop a subject-specific, data-driven model that is capable of predicting the required joint torques needed to hold a desired arm configuration and the torques produced by stimulating the muscles in a given configuration. This model will form the basis of all controllers moving forward in this dissertation. To validate the model, we demonstrated its capabilities to be used as an open-loop controller to hold static wrist positions.*

**Conference publication:**

1. Derek N Wolf and Eric M Schearer. Evaluating an open-loop functional electrical stimulation controller for holding the shoulder and elbow configuration of a paralyzed arm. In *2017 International Conference on Rehabilitation Robotics (ICORR)*, pages 789–794. IEEE, 2017.

**ABSTRACT**

*Functional electrical stimulation (FES) is a promising solution for restoring functional motion to individuals with paralysis, but the potential for achieving full-arm reaching motions with FES for various desired tasks has not been realized. We present an open-loop controller capable of calculating and applying the necessary muscle stimulations to hold the wrist of an individual with high tetraplegia at any desired position. We used the controller to hold the wrist at a series of static positions. The controller was capable of discriminating between different wrist positions. The average distance to the target wrist position, or accuracy, was* $7.7\,\mathrm{cm}$. *The average radius of the 95% confidence ellipsoid for a set of trials with the same muscle stimulations, or precision, was* $6.7\,\mathrm{cm}$. *Adding feedback or online model updates will likely improve the accuracy for tasks requiring finer control. The controller is a good first step to controlling full-arm motions with FES.*

## 3.1 Introduction

There are approximately 282,000 individuals with spinal cord injuries (SCI) in the United States, of which 58.8% have some level of tetraplegia [1]. For these individuals with tetraplegia, the loss of functional motion in their upper extremities severely limits their quality of life, and restoring arm and hand function is their greatest priority for improving their quality of life [2]. Functional electrical stimulation (FES) is a promising technology for restoring function to individuals with SCI.

FES restores function in individuals with SCI by stimulating the paralyzed muscles to activate in desired patterns. FES has demonstrated success in restoring functions to subjects with SCI including standing [3], bowel control [4], and hand function [5]. These functions are typically achieved using fixed stimulation patterns. We aim to build upon these achievements in FES control to restore shoulder and arm function to patients with paralysis of their upper limbs due to high-level tetraplegia.

Full-arm reaching has been achieved in simulation using various controllers including artificial neural networks, PD, and PID controllers [6][7]. Practical implementation of these strategies with human subjects with high tetraplegia has proven difficult due to the complexity of the arm and shoulder systems and differences between the computer model and human subjects. Notable successes in FES-controlled full-arm reaching are the MUNDUS project [8] and the neuroprosthesis developed by the Cleveland FES Center [9]. Both projects significantly improved the subjects' abilities to perform activities of daily living, providing functions such as wiping the nose or lifting a cup to drink through a straw. MUNDUS achieves joint motion through controlling a single degree of freedom at a time while using an exoskeleton to lock the other motions. This technique does not exploit the redundancy of the arm to achieve different paths to the same target or modulate stiffness, thus limiting the flexibility of the tasks to be achieved. For the Cleveland FES Center neuroprosthesis, expert-tuned, predefined stimulation patterns are used to achieve reaching tasks. These specific motion patterns are useful for exercise routines and specific motions, but the lack of flexibility limits everyday practicality.

Full-arm functions are goal directed, and to achieve these goals, an individual must be able to move their arm to any place within their workspace to account for variations in the goal. For example, while eating a meal, the food and utensils will not be in the same place every day. The sheer number of possible goal positions makes retraining the controller for each goal improbable. Therefore, there exists a need for the development of a controller which can determine the stimulation commands necessary to achieve any desired task.

As a solution to this need, we, along with our colleagues, have developed a method for identifying a data-driven, subject-specific model of an arm driven by FES [10]. The model uses semiparametric Gaussian regression to estimate the arm's dynamics and response to stimulation. We aim to use this model as part of a controller to move the wrist of a subject with tetraplegia along any desired path. Given a desired goal path, our model can be used to determine the stimulation commands to achieve the goal. Our working hypothesis is that

reaching can be achieved by quasi-static tracking control, e.g. moving from static point to static point along a path from starting position to goal position. In this study, as a first step to this quasi-static control, we have focused on achieving static wrist positions. We use the model to determine the open-loop stimulation inputs necessary to achieve the desired static wrist position. These inputs are used to control the arm of a person with tetraplegia.

We completed this study as an initial step in developing a practical FES control strategy for functional reaching in an individual with tetraplegia. The overall goal was to measure the efficacy of our model as an open-loop controller for holding static wrist positions. Specifically, we quantified the accuracy and the precision of the controller.

Preliminary results of this study were presented in [11].

## 3.2 Materials and Methods

To evaluate the controller, we identified the model for a subject with high tetraplegia and an implanted neuroprosthesis, selected target wrist positions, and calculated and executed the stimulations necessary to achieve the targets. The model was identified on day one, and the accuracy and precision of the controller were tested over two subsequent days. The three-day process was completed twice.

### 3.2.1 Experimental Setup

We performed the experiments with a single human subject with tetraplegia. The subject was a 59-year-old female who sustained a hemisection of the spinal cord at the C1-C2 level. She is unable to move her right arm but does have sensation. She experiences hypertonia in some of the arm muscles. Additionally, the subject's wheelchair is equipped with a passive arm support to assist against the force of gravity. More details can be found in [12] (subject 1).

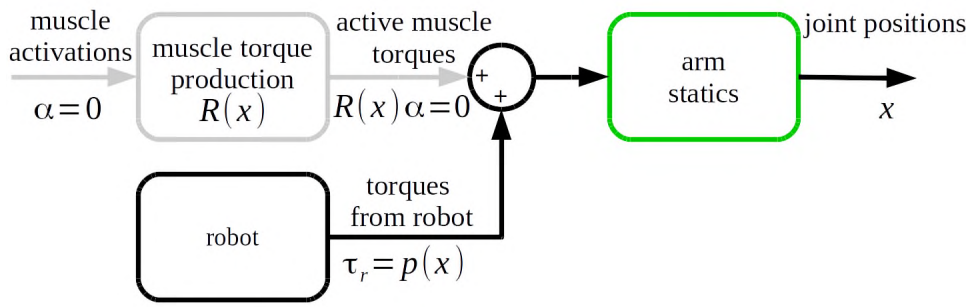
The subject is implanted with a stimulator-telemeter in her abdomen [13][14][15]. This device has leads which carry current to intramuscular electrodes [16] and nerve cuff elec-

trodes [17] to activate muscles in her right arm and shoulder complex. We refer to each muscle or group of muscles stimulated by a single electrode as a muscle group. In this experiment, we controlled the nine muscle groups. Power and control signals are sent to the implanted device via an inductive radio-frequency link. Muscle stimulation uses bi-phasic, charge balanced pulses delivered at 13 Hz. The amplitude of the pulses is determined for each muscle group. The force generated by each muscle group is controlled by varying the pulse-width (referred to as the stimulation input) from 0-250 $\mu$s. The vector of stimulation inputs for every muscle group is the control input. We send stimulation commands to the implant using real-time control code on a computer. Appropriate stimulation and amplitude limits were determined for subject safety. The controller commands cannot exceed these limits. Protocols used for this research were approved by the institutional review boards at Cleveland State University (IRB NO. 30213-SCH-HS) and MetroHealth Medical Center (IRB NO. 04-00014).

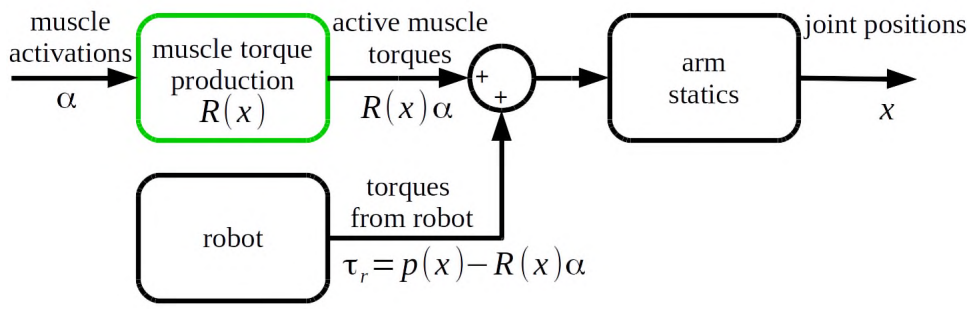
Data for developing the model were gathered using a HapticMaster (Moog FCS) robot with three degrees of freedom. The robot records the 3D forces and position of its end effector. An Optotrak Certus Motion Capture System (Northern Digital, Inc.) was used to capture data which was used to estimate the arm's configuration by adding smoothing to the extended Kalman filter method and MATLAB® code from [18]. The arm's configuration was defined by the joint positions of the shoulder elevation plane, shoulder elevation, shoulder rotation, elbow flexion, and elbow pronation as defined in [19].

### 3.2.2 Model Identification

We developed a three-part model consisting of: 1. inverse statics (the mapping from configuration to joint torque) 2. muscle torque production (the mapping from configuration and activation to the torques produced) and 3. recruitment curves (the mapping from muscle group stimulation to activation). The model was identified using the approach defined in [10]. A summary is presented here for completeness.




(a) Identification of arm statics; No muscle stimulation




(b) Identification of muscle torque production; Stimulation of one muscle group

Figure 5: This shows an illustration of the model identification. The torques from the robot, $\boldsymbol{\tau}_r$, are the shoulder and elbow torques which would produce the same static position as the force applied by the robot during a trial. For a given joint configuration, $x$, when no muscle groups are stimulated (a), the robot torques are equal to the torques, $\mathbf{p}(x)$, required to hold the arm in the configuration. With one muscle group stimulated at 100%, the robot torques are equal to the difference between $\mathbf{p}(x)$ and the torques produced by the muscles. For each trial, we choose the muscle group activation, compute the robot torques, and use our identification technique to determine the arm statics and muscle torque production blocks.

To gather data for the model identification, a robot held the subject's wrist at a series of positions within the subject's workspace. The subject's wrist was connected to the robot via a ball-in-socket joint that does not transmit torque. The robot was equipped with a three-dimensional force sensor at its end-effector. For each position, the kinematic Jacobian of the arm at the wrist was used to transform the force recorded by the robot into the joint torques, $\boldsymbol{\tau}_r$, about the shoulder and elbow which would produce the equivalent force.

To determine the arm statics, the robot held the arm in a position with zero muscle stimulation (Fig. 5(a)), as such all muscle activations, $\boldsymbol{\alpha}$, are zero. Therefore,

$$\boldsymbol{\tau}_r = \mathbf{p}(x) \tag{3.1}$$

where $\mathbf{p}(x)$ are the the torques necessary to hold the arm in the static configuration, $x$, determined by the shoulder elevation plane, shoulder elevation, shoulder rotation, elbow flexion, and elbow pronation as defined in [19].

To determine the torque production of a single muscle group, the muscle group was stimulated at its maximum stimulation command (Fig. 5(b)) so that $\boldsymbol{\alpha}$ was a vector of all zeros except for an activation of one for a single muscle group. The calculated joint torques from the robot are then defined by the difference of the robot torques with zero stimulation and during stimulation,

$$\boldsymbol{\tau}_r = \mathbf{p}(x) - R(x)\boldsymbol{\alpha}, \tag{3.2}$$

where $R(x) \in \mathbb{R}^{4\times 9}$ is the linear mapping of muscle activation to joint torque and $\mathbf{p}(x)$ are the torques when stimulating no muscles. The assumptions of the forces produced by muscles combining in a linear fashion was validated in [20]. Each row of $R(x)$ represents the torque about each degree of freedom. Elbow pronation torque is omitted as it does not affect the position of the wrist which is the focus of our study. Each column of $R(x)$ represents the amount of torque produced in each degree of freedom by 100% activation of a muscle group. The elements in $R(x)$ are determined by subtracting the total torque, $\mathbf{p}(x) - R(x)\boldsymbol{\alpha}$, during stimulation from the previously identified inverse static torques.

This process of determining $\mathbf{p}(x)$ and $R(x)$ for a joint configuration, $x$, was completed for 27 positions within the subject's workspace. The sets of 27 positions were repeated as many times as possible in the allotted time (5 for maximization experiments and 6 for minimization experiments as defined in section 3.2.4). Within each set, the order of positions was randomized, and the order of muscle group activations was randomized for each position. The data was used to train a semiparametric Gaussian process regression (GPR) [21] which can be used to determine $\mathbf{p}(x)$ and $R(x)$ for any configuration, $x$, within the subject's workspace.

The mapping from the stimulation input to the muscle group activation, known as the recruitment curve, was identified using the deconvolved ramp method [22].

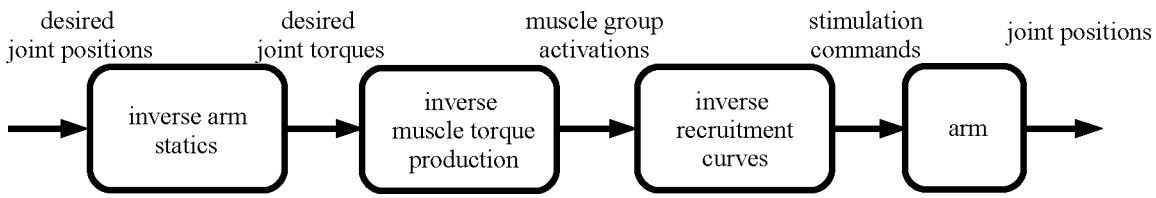



Figure 6: Controller block diagram

**Model methodology adjustments throughout the dissertation**

For the model presented in this chapter, the input to the model is the configuration of the arm defined as the joint angles. The model is then used to determine the torques produced by each muscle. Throughout the dissertation, we worked with different definitions of the arm configuration and with force-space vs torque-space models. These changes will be discussed in each chapter as they are used. The general modeling process is largely the same regardless of these adjustments.

### 3.2.3 Controller

Using the model presented in section 3.2.2, our controller determines and applies the open-loop stimulation inputs necessary to hold a desired set of joint angles. An illustration of the controller is seen in Fig. 6. The controller first maps the desired joint positions to the joint torques necessary to hold those positions. The muscle group activations necessary to achieve the desired torques are then determined and mapped to the stimulation inputs and applied to the arm.

The inputs to the controller are the desired joint positions, $x_* \in \mathbb{R}^5$, that correspond to the desired wrist position. The joint positions are the shoulder elevation plane, shoulder elevation, shoulder rotation, elbow flexion, and elbow pronation. The first block of the controller uses the GPR model of arm statics to calculate the desired joint torques, $\mathbf{p}(x_*)$, necessary to hold the configuration. The next block of the controller uses the GPR model of muscle torque production to identify the elements of the mapping from joint torques to muscle group activations, $R(x_*)$.

Knowing the desired joint torques, $\mathbf{p}(x_*)$, and the mapping, $R(x_*)$, we calculate the muscle activations, $\boldsymbol{\alpha}$, by solving $R(x_*)\boldsymbol{\alpha} = \mathbf{p}(x_*)$. $R(x_*)$ is not square as there are more muscle groups than degrees of freedom. Therefore, this becomes an optimization problem of the form

$$\begin{aligned} \underset{\boldsymbol{\alpha}}{\text{minimize:}} \quad & ||\boldsymbol{\alpha}||_2^2 \\ \text{subject to:} \quad & R(x_*)\boldsymbol{\alpha} = \mathbf{p}(x_*) \\ & \alpha_i \in [0,1] \quad \forall i \in \{1,2,\ldots,9\} \end{aligned} \quad . \tag{3.3}$$

A quadratic programming routine is used to solve (3.3). The routine returns a set of muscle activations if a feasible solution is found or a flag if a solution is not found.

Equation (3.3) can be solved for both minimizing and maximizing the muscle activations. A good starting point would be to minimize the activations to limit the energy usage and fatigue. However, minimizing the muscle activations limits co-contraction. Co-contraction of antagonist muscles has been observed in able bodied movements and results in benefits such as increased damping to limit overshoot [23]. In the present study both maximization and minimization of muscle activations were tested.

If a feasible solution to (3.3) is found, the recruitment curves are inverted (third block of the controller) to determine the stimulation inputs to achieve the desired muscle activations. These stimulation inputs are sent to the stimulator to be applied to the arm.

### 3.2.4 Static Hold Experiments

To evaluate the controller's ability to hold static positions, we quantified the accuracy and precision of the controller at various targets in the subject's workspace during four sessions held on separate days. For each individual trial, the robot moved the subject's wrist to the desired target position. With the robot holding the wrist stationary, the calculated stimulation input was applied to the arm. After holding for one second, the robot allowed the arm to move freely depending on the stimulation of the muscles for 5 seconds. The final wrist position was recorded. A perfect controller would result in a stationary wrist position for

the entire trial, while a less than perfect model would result in movement away from the starting position.

The targets were selected as the nearest feasible positions (as determined by the ability of the controller to solve (3.3)) to the 27 targets used in the model identification. For each target, the stimulation input to hold each position was calculated. For one set of experiments, the maximum muscle activations were found (referred to as maximization experiments). For the second set of experiments, the minimum muscle activations were found (minimization experiments). The experiments were repeated twice, 2 days after model identification and 7 days after model identification, as determined by subject availability. For each set, the targets were completed in a random order. The number of sets completed each day was determined by the allotted time.

To start each day of experimentation, every target position was tested for subject comfort. If the subject reported any discomfort in a target position, the target was removed from all future trials. Due to the varying nature of the subject's muscles, especially muscle tone, the comfortably reachable workspace varied from day to day. For a single set of experiments, the targets on the second day of testing were the same as the first day except for those targets which were removed due to subject discomfort on the second day. For the maximization experiments, we tested 13 sets of 15 targets on day one and 10 sets of 13 targets on the second day. For the minimization experiments, we tested 12 sets of 22 targets on day one and 13 sets of 20 targets on day two.

### 3.2.5 Data Analysis

The accuracy of each trial of the static hold experiments was defined as the Euclidean distance from the final wrist position and the target wrist position. For a set of trials, the accuracy was the average of all trials in the set. The precision, $r$, for a set of trials is defined by

$$r = \sqrt{\chi \lambda_{max}}, \tag{3.4}$$

where $\lambda_{max}$ is the maximum eigenvalue of the covariance matrix for the end positions of the trials and represents the largest spread of the points in any direction. $\chi$ is the inverse of the chi-squared cumulative distribution function. In this case, for three dimensions and a 95% confidence, $\chi = 7.8147$. $r$ is thus equal to half the length of the maximum axis for the 95% confidence ellipsoid of the data. Therefore, $r$ represents the radius of a sphere which will encompass 95% of the final positions from the distribution of the set.

A 1-way ANOVA was completed to determine if the accuracy and precision were significantly different for maximizing and minimizing the muscle activations.

### 3.3 Results

The controller is able to hold and discriminate between a variety of wrist positions. This can be seen in Fig. 7 which shows the 95% confidence ellipsoid calculated using the standard error of the mean of the final positions for each target of a representative set of trials. For the most part, the ellipsoids do not overlap demonstrating the mean wrist position for one set of muscle stimulations is different than for another set of muscle stimulations.

A representative example of a target with 12 repetitions from the first day of minimization experiments is shown in Fig. 8(a). The accuracy was $11.5\,\mathrm{cm}$, and the precision was $14.0\,\mathrm{cm}$. Fig. 8(b) shows a representative example of a target with 13 repetitions from the first day of maximization experiments. The accuracy was $2.6\,\mathrm{cm}$, and the precision was $3.7\,\mathrm{cm}$.

The accuracy and precision were quantified for all targets. For the maximization experiments, the mean accuracy (standard deviation) was $6.3\,\mathrm{cm}$ ($2.7\,\mathrm{cm}$) for day 1 and $6.5\,\mathrm{cm}$ ($3.5\,\mathrm{cm}$) for day 2. For the minimization experiments, the mean accuracy was $8.5\,\mathrm{cm}$ ($5.2\,\mathrm{cm}$) for day 1 and $8.5\,\mathrm{cm}$ ($3.8\,\mathrm{cm}$) for day 2. The average accuracy for all trials was $7.7\,\mathrm{cm}$ ($4.2\,\mathrm{cm}$).

The mean precision (standard deviation) was $6.0\,\mathrm{cm}$ ($2.0\,\mathrm{cm}$) for day 1 of the maximization experiments and $8.9\,\mathrm{cm}$ ($2.7\,\mathrm{cm}$) for day 2. For the minimization experiments,

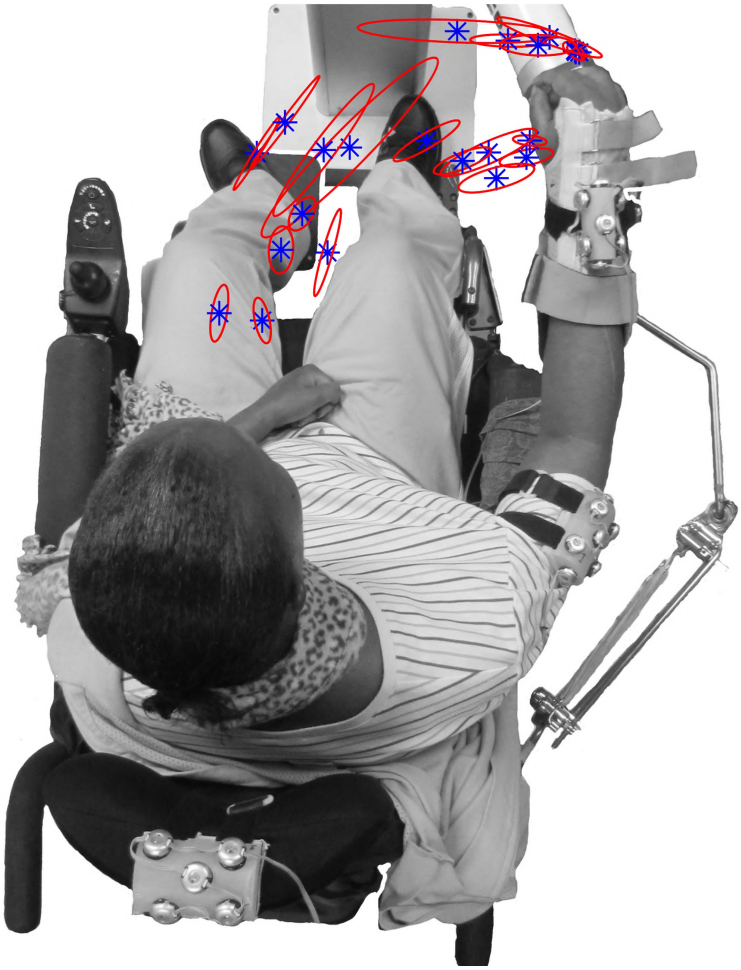


Figure 7: Representative example showing the mean final positions and 95% confidence ellipsoids for the mean for the targets in day one of the minimization experiments. This image shows the ability of the controller to discriminate between positions in the workspace.

the mean precision was $6.1\,\mathrm{cm}$ ($4.1\,\mathrm{cm}$) for day 1 and $6.6\,\mathrm{cm}$ ($3.3\,\mathrm{cm}$) for day 2. The average precision for all targets was $6.7\,\mathrm{cm}$ ($3.4\,\mathrm{cm}$).

Maximizing the muscle activations resulted in a significantly better accuracy than minimizing the muscle activation ($p = 4.3e^{-13}$). There was not a significant difference in the precision of maximization experiments and the minimization experiments ($p = 0.2$).

The final positions of some trials were limited by the workspace of the robot and not by the subject or stimulation. For maximization experiments, 8.3% of the trials finished at the limit of the robot workspace. For the minimization experiments, 44.1% of the trials finished at the limit of the robot workspace.

## 3.4 Discussion and Conclusion

In this study, we have presented an open-loop controller for holding any feasible static wrist position with an FES-controlled paralyzed human arm. Most importantly, the controller is capable of holding and discriminating between wrist positions.

The accuracy of $7.7\,\mathrm{cm}$ was similar to the accuracy achieved by [24]. Controlling elbow joint angle tracking using feedback and co-activation of antagonist muscles, the authors achieved a root-mean-square error of approximately $9°$ for trajectories with no disturbance. Using the measured length of our subject's arm, 57 cm, and translating the error to the shoulder joint, the same joint angle error would result in a wrist position error of $9.0\,\mathrm{cm}$. While our controller has only been demonstrated for static positions, having similar accuracy without feedback while including the degrees of freedom in the shoulder is encouraging moving forward.

The accuracy found in this study is useful in many applications similar to those achieved in other studies [8][9]. If the accuracy of $7.7\,\mathrm{cm}$ (about the length of a finger) was maintained for an entire trajectory, a person could successfully comb one's hair or move food from a plate to their mouth with some head movement to account for the error in final position. For finer movements, such as picking up food with a fork, improved accuracy may be necessary. The controller presented in this study achieved this accuracy without limiting the flexibility of the achievable tasks in that it is capable of determining the stimulation inputs for any feasible wrist position and has the potential to modulate the stiffness by solving (3.3) with an objective other than the maximization and minimization of muscle activations.

The average precision of $6.7\,\mathrm{cm}$ implies that applying the same stimulation inputs to the muscles produces similar results during a single day of experiments. The experiment has not been completed over enough days to make conclusions about the the results of a stimulation pattern over several days. The controller's achieved precision is encouraging for its continued use moving forward. For performing quasi-static control of the wrist, this

precision means the controller can place the wrist at unique points $6.7\,\mathrm{cm}$ apart. This may not be fine enough for some tasks such as picking up a small piece of food with a fork.

For fine motions, further strategies are necessary to improve the overall performance of the controller. The most obvious place to start would be the addition of feedback to our controller. A similar combination of a nonlinear model and feedback has performed well in simulation [6]. Additionally, model identification the day of the experiment may result in better results. At the current time, the model identification process takes too long to both identify the model and perform control experiments in a single day. Simpler methods for identification have been tested [24], but they use large assumptions (such as linearity) which may result in errors and differences between subjects. There exists a need for identifying or updating the models to be used the same day.

Some trials resulted in a final wrist position at the limits of the robot's workspace. The error we measured for these trials is likely smaller than if the movement was not constrained by the robot. Additionally, trials constrained by the robot's limits may result in a tighter grouping of final wrist positions. These facts likely have some effect on the overall accuracy and precision numbers we present, especially for the minimization experiments, where 44% of all trials ended at the limits of the robot's workspace. During these experiments, it was observed that the arm frequently fell to the lower limits of the robot. Additionally, the minimization experiments frequently ended up with the subject's arm extended to the right edge of the robot's workspace. Minimizing muscle activations results in less co-contraction. Without co-contraction, there is not an antagonist muscle to offset the movement that occurs due to model error. This is similar to the results in [23] and [24] showing co-contraction increases the stiffness of the joint.

This study presents a method of open-loop control of an FES-controlled paralyzed human arm. Our controller is able to accurately and consistently hold feasible static wrist positions. For fine movements, improved performance may be necessary. Adding feedback to the controller may provide the best opportunity to do so. Using this controller,

FES-controlled full-arm reaching can be achieved by commanding a sequence of static positions along a path connecting a starting position to a goal position.

**Acknowledgments**

This work was supported by NIH NINDS grant N01-NS-5-2365 and the Cleveland State University Faculty Research and Development Plan. The authors would also like to thank Dr. Robert Kirsch at Case Western Reserve University for access to laboratory space and for referring us to the human subject who participated in this research.

## 3.5 REFERENCES

[1] NSCISC. Spinal cord injury facts and figures at a glance. National Spinal Cord Injury Statistical Center, 2016.

[2] K. D. Anderson. Targeting recovery: Priorities of the spinal cord-injured population. *Jounral of Neurotrauma*, 21(10):1371–1383, 2004.

[3] H. Kobravi and A. Erfanian. A decentralized adaptive fuzzy robust strategy for control of upright standing posture in paraplegia using functional electrical stimulation. *Medical Engineering & Physics*, 34(1):28–37, 2012.

[4] K. T. Ragnarsson. Functional electrical stimulation after spinal cord injury: Current use, therapeutic effects and future directions. *Spinal Cord*, 46:255–274, 2008.

[5] P. H. Peckham and J. S. Knutson. Functional electrical stimulation for neuromuscular applications. *Annual Reviews in Biomedical Engineering*, 7:327–360, 2005.

[6] D. Blana, R. F. Kirsch, and E. K. Chadwick. Combined feedforward and feedback control of a redundant, nonlinear, dynamic musculoskeletal system. *Medical & Biological Engineering & Computing*, 47:533–542, 2009.

[7] Kathleen M Jagodnik and Antonie J van den Bogert. Optimization and evaluation of a proportional derivative controller for planar arm movement. *Journal of Biomechanics*, 43, 2010.

[8] A. Pedrocchi, S. Ferrante, E. Ambrosini, M. Gandolla, C. Casellato, T. Schauer, C. Klauer, J. Pascual, C. Vidaurre, M. Gföhler, et al. MUNDUS project: MUltimodal Neuroprosthesis for daily Upper limb Support. *Journal of NeuroEngineering and Rehabilitation*, 10(1):1–20, 2013.

[9] W. D. Memberg, K. H. Polasek, R. L. Hart, A. M. Bryden, K. L. Kilgore, G. A. Nemunaitis, H. A. Hoyen, M. W. Keith, and R. F. Kirsch. Implanted neuroprosthesis for restoring arm and hand function in people with high level tetraplegia. *Archives of Physical Medicine and Rehabilitation*, 95(6):1201–1211, 2014.

[10] Eric M. Schearer, Yu-Wei Liao, Eric J. Perreault, Matthew C. Tresch, William D. Memberg, Robert F. Kirsch, and Kevin M. Lynch. Semiparametric Identification of Human Arm Dynamics for Flexible Control of a Functional Electrical Stimulation Neuroprosthesis. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 24(12):1405–1415, Dec 2016. ISSN 1534-4320. doi: 10.1109/TNSRE.2016.2535348.

[11] Eric M Schearer, Derek N Wolf, and Robert F Kirsch. Quasi-static control of whole-arm motions with fes. In *Converging Clinical and Engineering Research on Neurorehabilitation II*, pages 673–677. Springer, 2017.

[12] K. H. Polasek, H. A. Hoyen, M. W. Keith, R. F. Kirsch, and D. J. Tyler. Stimulation stability and selectivity of chronically implanted multicontact nerve cuff electrodes in the human upper extremity. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 17(5):428–437, 2009.

[13] B. Smith, P. H. Peckham, M. W. Keith, and D. D. Roscoe. An externally powered,

multichannel, implantable stimulator for versatile control of paralyzed muscle. *IEEE Transactions on Biomedical Engineering*, 34(7):499–508, 1987.

[14] B. Smith, Z. Tang, M. W. Johnson, S. Pourmehdi, M. M. Gazdik, J. R. Buckett, and P. H. Peckham. An externally powered, multichannel, implantable stimulator-telemeter for control of paralyzed muscle. *IEEE Transactions on Biomedical Engineering*, 45(4):463–475, 1998.

[15] R. L. Hart, N. Bhadra, F. W. Montague, K. L. Kilgore, and P. H. Peckham. Design and testing of an advanced implantable neuroprosthesis with myoelectric control. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 19(1):45–53, 2011.

[16] W. D. Memberg, P. H. Peckham, and M. W. Keith. A surgically-implanted intramuscular electrode for and implantable neuromuscular stimulation system. *IEEE Transactions on Rehabilitation Engineering*, 2(2):80–91, 1994.

[17] G. G. Naples and J. T. Mortimer. A spiral nerve cuff electrode for peripheral nerve stimulation. *IEEE Transactions on Biomedical Engineering*, 35(11):905–916, 1988.

[18] E. Todorov. Probabilistic inference of multijoint movements, skeletal parameters and marker attachments from diverse motion capture data. *IEEE Transactions on Biomedical Engineering*, 54(11):1927–1939, 2007.

[19] G. Wu, F. C. van der Helm, H. E. Veeger, M. Makhsous, P. Van Roy, C. Anglin, J. Nagels, A. R. Karduna, K. McQuade, X. Wang, F. W. Werner, and B. Buchholz. ISB recommendation on definitions of joint coordinate systems of various joints for the reporting of human joint motion – part II: Shoulder, elbow, wrist and hand. *Journal of Biomechanics*, 38(5):981–992, 2005.

[20] E. M. Schearer, Y. Liao, E. J. Perreault, M. C. Tresch, W. D. Memberg, R. F. Kirsch, and K. M. Lynch. Indentifying inverse human arm dynamics using a robotic testbed.

In *International Conference on Intelligent Robots and Systems*, pages 3585–3591, 2014. URL `http://ieeexplore.ieee.org/document/6943064/`.

[21] C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. The MIT Press, Cambridge, MA, 2006.

[22] W. K. Durfee and K. E. MacLean. Methods for estimating isometric recruitment curves of electrically stimulated muscle. *IEEE Transactions on Biomedical Engineering*, 36(7):654–666, 1989.

[23] Cheryl L Lynch, Dimitry Sayenko, and Milos R Popovic. Co-contraction of antagonist muscles during knee extension against gravity: Insights for functional electrical stimulation control design. In *2012 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pages 1843–1846. IEEE, 2012.

[24] Antônio Padilha L Bó, Lucas O da Fonseca, and Ana Carolina C de Sousa. Fes-induced co-activation of antagonist muscles for upper limb control and disturbance rejection. *Medical Engineering & Physics*, 38(11):1176–1184, 2016.

(a) A target with an accuracy of 11.5 cm and precision of 14.0 cm

(b) A target with an accuracy of 2.5 cm and precision of 3.7 cm

Figure 8: Representative examples showing the accuracy and precision for a single target (Cyan represents the target, blue represents the mean final position, green represents the final position of the individual trials, and red represents the 95% confidence ellipsoid used to calculate the precision).

# CHAPTER IV

# HOLDING STATIC ARM CONFIGURATIONS WITH FEEDBACK CONTROL OF FUNCTIONAL ELECTRICAL STIMULATION

*Functional electrical stimulation offers promise as a solution to restore the ability to complete reaching motions to individuals with paralyzed limbs due to spinal cord injury. As a step to achieving FES-controlled reaching motions, we present a controller that is capable of selecting and applying the required stimulation to hold a subject's wrist at a desired location. I describe our modeling procedure in detail again, and then use the model as the basis of a combined feedforward and feedback controller. With the addition of feedback, our controller is able to compensate for errors in the model to hold static, three-dimensional wrist positions significantly better than with open-loop control. The main contribution of this chapter is the development of a data-driven-model-based feedback controller for 3-dimensional wrist-position control of an FES-controlled paralyzed human arm.*

**Publication:**

1. Derek N. Wolf and Eric M. Schearer. Holding Static Arm Configurations With Functional Electrical Stimulation: A Case Study. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 26(10):2044–2052, Oct 2018. ISSN 1534-4320.doi: 10.1109/TNSRE.2018.2866226.

**ABSTRACT**

*Functional electrical stimulation (FES) is a promising solution for restoring functional motion to individuals with paralysis, but the potential for achieving any desired full-arm reaching motion has not been realized. We present a combined feedforward-feedback controller capable of automatically calculating and applying the necessary muscle stimulations to hold the wrist of an individual with high tetraplegia in a desired static position. We used the controller to hold a complete arm configuration to maintain a series of static wrist positions. The average distance to the target wrist position, or accuracy, was 2.9 cm. The precision is defined as the radius of the 95% confidence ellipsoid for the final positions of a set of trials with the same muscle stimulations and starting position. The average precision was 3.7 cm. The control architecture used in this study to hold static positions has the potential to control arbitrary reaching motions.*

## 4.1 Introduction

For approximately 166,000 individuals in the United States living with some level of tetraplegia, the loss of functional motion in their upper extremities limits their ability to self-feed, groom themselves, and perform other activities of daily living [1]. For these individuals, their greatest priority for functional recovery is the restoration of arm and hand function [2]. Functional electrical stimulation (FES) is a promising technology for restoring full-arm reaching function to individuals with spinal cord injuries (SCI).

FES restores function in individuals with SCI by stimulating paralyzed muscles to activate in desired patterns. FES has demonstrated success in restoring functions to individuals with SCI including standing [3], bowel control [4], and hand function [5]. These functions have typically been achieved using fixed stimulation patterns. Implementing fixed stimulation patterns to control full-arm reaching has been attempted [6], but these methods lack the flexibility to achieve any goal-directed task and to account for the complexity of the

shoulder and arm mechanics.

More flexible methods have been developed to select the stimulation commands required to control the arm's joint or wrist position. Many strategies have been implemented in computer simulations including using an optimized proportional-derivative controller [7], combined feedforward-feedback controllers [8], reinforcement learning [9], and threshold control [10]. While these, and other controllers, have proven successful in simulation, in practice, application has been limited due to the differences between the models and constantly changing real-world arm dynamics.

Most practically applied control strategies for reaching motions have, to this point, treated the joints independently instead of as a complete arm system. The MUltimodal Neuroprosthesis for daily Upper limb Support (MUNDUS) project successfully achieved some reaching tasks by using an exoskeleton to lock all degrees of freedom (DOF) except for the single joint currently being actively controlled [11]. However, this method does not take advantage of the kinematic redundancy of the arm which allows an individual to reach points in their workspace following different trajectories. Additionally, this system results in slower, less smooth movements than standard reaching motions.

The most advanced FES-controlled reaching system, demonstrated as part of the BrainGate2 clinical trial, used a percutaneous FES system controlled via an intracortical brain-computer interface [12]. The system controlled each joint simultaneously, but still treated the joints as independent. Using a low level controller which independently controlled each joint, it was difficult for the participant to accurately control the multiple degrees of freedom necessary to complete full-arm reaching motions. The system also did not control the shoulder using FES, which would significantly increase the difficulty of control due to the increased degrees of freedom.

Model-based controllers which seek to control the entire arm system have been developed to overcome these obstacles. Parameterized models have had some success in controlling two muscles in rehabilitation of stroke patients [13, 14], but assessing the pa-

rameters of all muscles necessary for complete arm control requires significantly larger amounts of data. Nonparametric models have thus been developed to eliminate the need of direct parameter identification. We have previously demonstrated that these methods, used in open-loop control, are capable of holding and differentiating between desired wrist positions in the reachable workspace [15]. However, feedback is necessary to achieve the accuracy required for many reaching tasks.

Feedback control of planar arm tasks has been achieved in healthy individuals using a model-based controller [16]. The authors used an artificial neural network to map the configuration in task space to the forces the muscles produce. The shift to a task-space (as opposed to a joint-space) controller makes planning and feedback more intuitive as this is the space in which the reaching is occurring. Overall, this technique was very successful in planar reaching and may be useful for some tasks, but many other tasks require three-dimensional movements (for example, moving food from a plate to the mouth). Removing the constraints of a planar workspace significantly complicates the problem.

To apply these ideas to practical, three-dimensional control of an impaired arm driven by FES, we propose using a similar model-based method that isn't subject to planar constraints and controls the whole-arm system instead of individual joints. We present a combined feedforward-feedback task-space controller. We identify a data-driven, person-specific model of an arm driven by FES which provides a feedforward aspect of the controller. Feedback is added to the system via a positional PID controller. The controller then uses the model to calculate the muscle stimulations necessary to achieve the desired wrist position.

We completed this study to test the feasibility of the presented control architecture for controlling full-arm reaching movements with FES. The main goal for the project was to evaluate the performance of the combined feedforward-feedback controller for holding static wrist positions with an FES driven arm.

## 4.2 Methods

To assess the controller, we identified the model for an individual with high tetraplegia and an implanted neuroprosthesis and then used the model as the basis of a feedforward-feedback controller (referred to as the feedback+ controller) to calculate and execute the muscle stimulation commands necessary to achieve a series of desired wrist target positions. For the set of experiments, the model was identified over the course of a day, and the controller was tested over two additional days. For simplicity, the two days of controller testing will be referred to as Day One and Day Two respectively.

Each day of the experiments took place during a four-hour time block. Approximately one hour was used to set up the motion capture system and the participant. The participant would then take a half-hour break to eat lunch. The experimentation took place during the remaining 2.5 hours with short breaks whenever the participant requested.

### 4.2.1 Experimental Setup

We completed the experiments with a single human participant who has high tetraplegia. The participant was a 60-year-old female who sustained a hemisection of the spinal cord at the C1-C2 level. She is unable to voluntarily move her right arm (the arm with which we performed our experiments) but does have sensation. She experiences hypertonia in some of the arm muscles. The participant's wheelchair is equipped with a passive arm support which produces a comfortable and achievable workspace by using elastic bands to assist against the force of gravity. The arm support results in a resting equilibrium position with the wrist approximately at the height of the participant's chest. More details can be found in [17] (Subject 1).

The participant is implanted with a stimulator-telemeter in her abdomen [18–20]. The device has leads which transmit current to intramuscular electrodes [21] and nerve cuff electrodes [22] activating muscles in her right arm and shoulder complex. We refer to each muscle or group of muscles stimulated by a single electrode as a muscle group. In this

experiment, we controlled the nine muscle groups shown in Table I. Power and control signals are sent by a computer to the implanted device via an inductive radio-frequency link. Muscle stimulation uses bi-phasic, charge balanced pulses delivered at 13 Hz. The amplitude of the pulses is constant for each muscle group. The force generated by each muscle group is controlled by varying the pulse-width (referred to as the stimulation input) from 0-250 $\mu$s. The maximum stimulation input for each muscle was determined as the point when no additional muscle force was achieved or the participant reported discomfort (shown in the last column of Table I). The vector containing the stimulation inputs for every muscle group is the control input. Stimulation commands are sent to the implant using real-time control code on a computer. Protocols used for this research were approved by the institutional review boards at Cleveland State University (IRB NO. 30213-SCH-HS) and MetroHealth Medical Center (IRB NO. 04-00014).

Table I: Summary of stimulation electrodes used

| Electrode Placement | Muscles Targeted | Approximate Function | Type | Current Amplitude (mA) | Max Pulse Width ($\mu$s) |
|---|---|---|---|---|---|
| radial nerve | triceps | elbow extension | nerve cuff | 2.1 | 250 |
| axillary nerve | deltoids | arm abduction | nerve cuff | 2.1 | 23 |
| thoracodorsal nerve | latissimus dorsi | arm adduction | nerve cuff | 0.8 | 10 |
| long thoracic nerve | serratus anterior | scapular abduction | nerve cuff | 1.4 | 20 |
| musculocutaneous nerve | biceps, brachialis | elbow flexion | nerve cuff | 0.8 | 49 |
| suprascapular nerve | supraspinatus, infraspinatus | shoulder stability, humeral roation | nerve cuff | 1.4 | 62 |
| rhomboids | rhomboids | scapular adduction | intramuscular | 18.0 | 107 |
| lower pectoralis | lower pectoralis | shoulder horizontal flexion | intramuscular | 18.0 | 22 |
| upper pectoralis | upper pectoralis | shoulder horizontal flexion | intramuscular | 20.0 | 25 |

To identify the model, we gathered data using a HapticMaster (Moog FCS) robot with three degrees of freedom. The robot records the 3D forces and positions of its end-effector. An Optotrak Certus Motion Capture System (Northern Digital, Inc.) captured data used to estimate the arm's configuration. The arm's configuration was defined by the position and orientation of the wrist relative to the thorax. The motion capture system was also used to

measure the real-time position of the wrist to be used for feedback during the static hold experiments. A third order-moving average filter was used on the wrist position signal to achieve smooth velocities.

The experiment was controlled using MATLAB xPC target on a Dell Dimension 8400 PC with a Pentium 4 3.20 GHz processor. The control and data collection occurred at 52 Hz, but stimulation inputs were updated at the stimulation frequency of 13 Hz.

### 4.2.2 Model Identification

We developed a three-part model consisting of: 1. inverse arm statics (the mapping from configuration to the forces needed to hold the wrist in a position), 2. muscle force production (the mapping from configuration and activation to the forces produced at the wrist by each muscle), and 3. recruitment curves (the mapping from muscle group stimulation input to activation). Our controller uses the model of the inverse arm statics, the inverse of the model of muscle force production, and the inverse recruitment curves as shown in Fig. 9. A similar model identification procedure using a joint space configuration was defined in [23]. Following the ideas of [16], we developed our model using the wrist position and orientation because it produces a more intuitive system by working directly in the space where the task is occurring without reducing the amount of information in the model (Our joint space controller in [15] has five dimensions while the workspace of the controller presented in this paper is six dimensional.). Additionally, by working in the task-space, we are able to eliminate the need to accurately track the joint angles of the shoulder which is difficult. We present a complete summary of our task-space model identification here.

To gather data for the model identification, a robot held the participant's wrist at a series of static positions within the participant's comfortably reachable workspace. The connection of the participant's wrist to the robot was via a ball-in-socket joint that does not transmit torque. The robot was equipped with a three-dimensional force sensor at its end-effector, and the force needed to hold the wrist static, $\mathbf{f_r} \in \mathbb{R}^3$, was recorded.

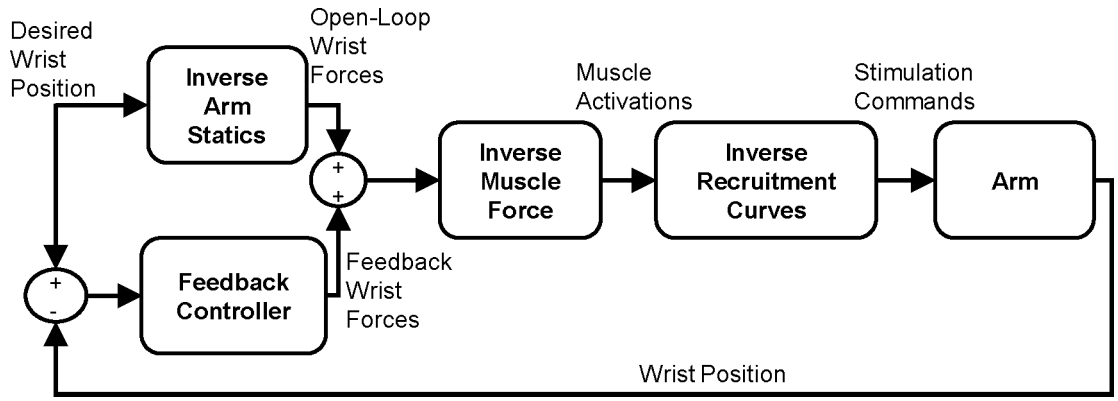



Figure 9: Controller block diagram

To determine the inverse arm statics, the robot held the arm in a position with zero muscle stimulation, and, thus, all muscle activations, $\boldsymbol{\alpha} \in \mathbb{R}^9$, were zero. Therefore,

$$\mathbf{f}_{\mathbf{r}_{\mathbf{static}}} = \mathbf{p}(\mathbf{q}) \tag{4.1}$$

where $\mathbf{p}(\mathbf{q}) \in \mathbb{R}^3$ are the forces necessary to hold the arm in the static configuration, $\mathbf{q} \in \mathbf{SE}(3)$, determined by wrist position and orientation. The wrist position is defined as $x$, $y$, and $z$ coordinates of the center of the wrist relative to the thorax coordinate frame. The wrist orientation is defined as the orientation of the forearm coordinate frame relative to the thorax. The thorax and forearm coordinate frames are defined by [24].

To determine the force production of the $j^{\text{th}}$ muscle group, the muscle group was stimulated at its maximum stimulation command so that $\boldsymbol{\alpha}$ was a vector of all zeros except for an activation of one for the selected muscle group. The forces applied by the robot, $\mathbf{f}_{\mathbf{r}_{\mathbf{stim}}\mathbf{j}}$, are then defined by the difference of the robot forces with zero stimulation (i.e. the required static forces) and the forces produced by the muscle group,

$$\mathbf{f}_{\mathbf{r}_{\mathbf{stim}}} = \mathbf{p}(\mathbf{q}) - \mathbf{R}(\mathbf{q})\boldsymbol{\alpha}, \tag{4.2}$$

where $\mathbf{R}(\mathbf{q}) \in \mathbb{R}^{3\times 9}$ is the linear mapping of muscle activation to forces at the wrist and

$\mathbf{p}(\mathbf{q})$ are the forces when stimulating no muscles. Each row of $\mathbf{R}(\mathbf{q})$ represents the force at the wrist in each Cartesian direction. Each column of $\mathbf{R}(\mathbf{q})$ represents the amount of force produced in each degree of freedom by 100% activation of the corresponding muscle group. The $j^{\text{th}}$ column of $\mathbf{R}(\mathbf{q})$ is determined by subtracting $\mathbf{f}_{\mathbf{r}_{\mathbf{stim}}\mathbf{j}}$, the recorded total force during stimulation of muscle group $j$, from the previously identified inverse static forces, $\mathbf{f}_{\mathbf{r}_{\mathbf{static}}}$,

$$\mathbf{R}(\mathbf{q})_{\mathbf{j}} = \mathbf{f}_{\mathbf{r}_{\mathbf{static}}} - \mathbf{f}_{\mathbf{r}_{\mathbf{stim}}\mathbf{j}}. \tag{4.3}$$

This process of identifying $\mathbf{p}(\mathbf{q})$ and $\mathbf{R}(\mathbf{q})$ for a wrist configuration $\mathbf{q}$, was completed for 27 positions within the participant's workspace. The set of 27 positions was repeated five times as determined by the allotted time. Within each set, the order of positions was randomized, and the order of activating muscle groups was randomized for each position. The data was used to train 30 Gaussian process regression (GPR) models [25]. The inputs for each GPR model were the wrist position and orientation, and the output was the forces recorded by the robot. One GPR model was used for arm statics in each Cartesian direction (three total models). For each muscle, a separate GPR model was used to determine the forces in each Cartesian direction required to hold the wrist in place when the muscle is stimulated (27 total models). Thus, using the GPR models, we can determine $\mathbf{p}(\mathbf{q})$ and $\mathbf{R}(\mathbf{q})$ for any desired wrist configuration $\mathbf{q}$ within the participant's workspace. When used in the controller (Fig. 9), the GPR models form the basis of the "Inverse Arm Statics" and "Inverse Muscle Force" blocks.

Relative to a parametric model, GPR does not have requirements on identifiability. Compared to other nonparametric methods, such as artificial neural networks, we chose GPR due to the automated nature of determining the complexity of the model by maximizing the marginal likelihood (see [23] for details on the quality of the model). The hyper-parameters for each model were selected by maximizing the marginal likelihood. The kernel function used in the GPR was the squared exponential function using the distance metric for rigid bodies defined in [26].

The recruitment curves, the mapping from stimulation input to muscle group activation, for each muscle group were identified using the deconvolved ramp method [27].

### 4.2.3 Controller

Our controller aims to determine the muscle stimulation commands necessary to maintain a desired static wrist position. It does so by building upon the model presented in section 4.2.2 which requires the wrist position and orientation as inputs. The controller (Fig. 9) uses the model to map the desired wrist position and orientation to the forces necessary to hold the wrist statically at the desired position. The muscle group activations necessary to achieve the desired forces are then determined and mapped to the stimulation inputs which are applied to the arm.

The input to the controller (see Fig. 9) is the desired wrist configuration (position and orientation), $\mathbf{q}_* \in \mathbf{SE}(\mathbf{3})$, that corresponds to the desired wrist position. The controller calculates the desired open-loop forces at the wrist, $\mathbf{p}(\mathbf{q}_*)$, necessary to hold the position by using the GPR model of the inverse arm statics. Feedback is added using a positional PID controller which outputs corrective forces in each degree of freedom ($x$, $y$, and $z$ directions). These forces are added to the open-loop forces to get the required force necessary to maintain the wrist position.

Next, the controller uses the GPR model of muscle force production to determine the force produced by each muscle group. Equation (4.3) is then used to identify the elements of the mapping from muscle group activations to wrist forces, $\mathbf{R}(\mathbf{q}_*)$.

It is important to reiterate that after the feedback controller is added to the output of the inverse arm statics model, it still requires the model of the inverse muscle force to calculate the desired muscle activations. After determining the desired forces and the muscle-force mapping, $\mathbf{R}(\mathbf{q}_*)$, we calculate the muscle activations, $\boldsymbol{\alpha}$ which will produce the desired forces. $\mathbf{R}(\mathbf{q}_*)$ is not square as there are more muscle groups than degrees of freedom. We resolve this redundancy and determine the muscle activations by solving the following

optimization problem,

$$\begin{aligned} \underset{\boldsymbol{\alpha}}{\text{minimize:}} \quad & ||\boldsymbol{\alpha}||_2^2 \\ \text{subject to:} \quad & \mathbf{R}(\mathbf{q}_*)\boldsymbol{\alpha} = \mathbf{p}(\mathbf{q}_*) \\ & \alpha_i \in [0,1] \quad \forall i \in \{1,2,\ldots,9\} \end{aligned} \quad . \tag{4.4}$$

For feedback control, (4.4) must be solved in real-time as the desired forces, $\mathbf{p}(\mathbf{q}_*)$, are being updated. We used the quasi-Newton method to minimize the penalty function,

$$\begin{aligned} & ||\boldsymbol{\alpha}||_2^2 + c_1\|\mathbf{R}(\mathbf{q}_*)\boldsymbol{\alpha} - \mathbf{p}(\mathbf{q}_*)\|_2^2 + \mathbf{c_2K} \\ & K = \sum k_i \text{ where } k_i = \begin{cases} \boldsymbol{\alpha}_i^2 \text{ if } \boldsymbol{\alpha}_i < 0 \\ (\boldsymbol{\alpha} - 1)^2 \text{ if } \boldsymbol{\alpha}_i > 1 \\ 0 \text{ if } 0 \leq \boldsymbol{\alpha}_i \leq 1 \end{cases} \end{aligned} \quad , \tag{4.5}$$

where $||\boldsymbol{\alpha}||_2^2$ minimizes the muscle activations, $c_1\|\mathbf{R}(\mathbf{q}_*)\boldsymbol{\alpha} - \mathbf{p}(\mathbf{q}_*)\|_2^2$ penalizes activations that do not produce the desired force, and $c_2K$ penalizes activations which do not belong to $\alpha_i \in [0,1]$. $c_1$ and $c_2$ were chosen to be 500 and 50,000 respectively because they produced the same solution as the MATLAB function `quadprog` found for (4.4).

Equation (4.4) can be solved for a number of objective functions. As a starting point, we chose to minimize the muscle activations as a way to limit energy usage and fatigue. Once a feasible solution to (4.4) is found, the recruitment curves are inverted (inverse recruitment curves block of Fig. 9) to determine the stimulation inputs to achieve the desired muscle activations. These stimulation inputs are sent to the stimulator to be applied to the arm.

### 4.2.4 Static Hold Experiments

To evaluate the controller's ability to hold static positions, we quantified the accuracy of the controller at various targets in the participant's workspace during two sessions held on separate days. For each individual trial, the robot moved the participant's wrist to the desired target position. With the robot holding the wrist stationary, the stimulation input

calculated by the controller was applied to the arm. Each individual trial lasted seven seconds. To avoid transient dynamics of the muscle groups affecting the results, the robot held the participant's wrist in place with decreasing stiffness for the first two seconds. For the next five seconds, the arm moved freely depending on the stimulation of the muscles. The average wrist position over the final second of each trial was recorded. A perfect controller would result in a stationary wrist position for the entire trial, while a less than perfect controller would result in movement away from the starting position.

To select the targets, a $3\times3\times3$ grid of points was developed within the space of the training positions, thus ensuring a wide spread of targets distinct from the training positions. For each point, the nearest feasible wrist position was selected. Feasibility is determined by the ability to solve (4.4) using `quadprog`. From these points, 13 targets were selected based on participant comfort (as reported by the participant) while maintaining positions throughout the workspace.

A single target near the center of the workspace was selected to tune the PID controller. The controller was tuned with the goal of improving accuracy while limiting oscillation which could be disconcerting to the participant. After tuning was complete, every target was tested once, and the tuning was adjusted if oscillations occurred at any of the targets. The final proportional gain was 0.025 N/mm, derivative gain was 0.01 N-s/mm, and integral gain was 0.1 N/mm-s. These gains were the same for all Cartesian directions and were used for all targets and all trials across both days of static hold experiments.

For each set during testing, each target was repeated twice, once with the feedback+ controller and once with open-loop control (zero feedback forces) resulting in a total of 26 targets during a set. The order of the 26 targets was randomized in each set. The number of sets completed each day was determined by the scheduled time (5 sets on Day One and 11 sets on Day Two).

### 4.2.5 Data Analysis

The accuracy of each trial of the static hold experiments was defined as the Euclidean distance from the target wrist position to the average wrist position over the final second of a trial. For a set of trials, the accuracy was the average of all trials in the set.

The precision, $r$, for a set of trials is defined by

$$r = \sqrt{\chi \lambda_{max}}, \tag{4.6}$$

where $\lambda_{max}$ is the maximum eigenvalue of the covariance matrix for the mean wrist positions over the last second of the trials and represents the largest spread of the points in any direction. For three dimensions and a 95% confidence, $\chi$, the inverse of the chi-squared cumulative distribution function, is equal to 7.8147. Thus, $r$ is equal to half the length of the maximum axis for the 95% confidence ellipsoid of the data. Therefore, $r$ represents the radius of a sphere which will encompass 95% of the final positions.

To quantify the response of the system, the maximum error and 5% settling time for each trial was recorded. The 5% settling time was defined as the time after which the distance between the wrist and the target position remained within 5% of the accuracy for the trial.

The study was analyzed as a randomized complete blocked design where the blocks were each set of 26 targets. 1-way ANOVAs were completed to determine if the accuracy, settling time, and maximum error were significantly different for the feedback+ controller than for the open-loop controller. A 2-sample t-test was completed to determine if the controller affected the precision. A 2-sample t-test was also completed to determine if the accuracy of the controllers changed from day to day.

## 4.3 Results

The feedback+ controller generally performed with better accuracy and less maximum error than the open-loop controller. Feedback control typically had a significant effect on the overall controller during a trial and was dominated by the integral portion of control. To illustrate these results in detail, we present a representative example (Fig. 10-11) along with the numerical results from all trials.

Figure 10 shows the position of the wrist relative to the target, the desired forces (the input to the inverse muscle force block of the controller), and the stimulation commands for a representative trial of the experiments. The target shown is representative of the overall accuracy of the controllers, the time history of the controllers, the contributions of the feedback controller, and the complex relationship of the muscles.

Relative to the resting position of the participant (which is determined by the arm support), the target was a wrist position away from the participant (negative $x$ direction), to the participant's left (negative $y$) and slightly higher (positive $z$). To achieve the target position, the elastic properties of the arm support must be overcome, and thus our model predicts open-loop forces in the negative $x$, negative $y$, and positive $z$ directions.

As the trial begins, the wrist was gradually released from the target position over the first two seconds. At two seconds, there was an immediate movement away from the target position, most notably in the positive $y$ direction. To compensate for this movement, the feedback controller calculated forces in the negative $y$ direction. Due to this, the controller increased the upper pectoralis stimulation command to 100%, and the lower pectoralis quickly followed as more negative $y$ force was needed. Additionally, the need for increased $x$ force led to an increase in activation of the biceps/brachialis and a slight decrease in the triceps stimulation command. This new combination of muscles and stimulation commands led the $y$ position of the wrist to move back to the negative side of the target, while the $x$ position of the wrist moved very near the target. The $y$ desired force began to increase just after three seconds, and so the lower pectoralis stimulation command decreased. At

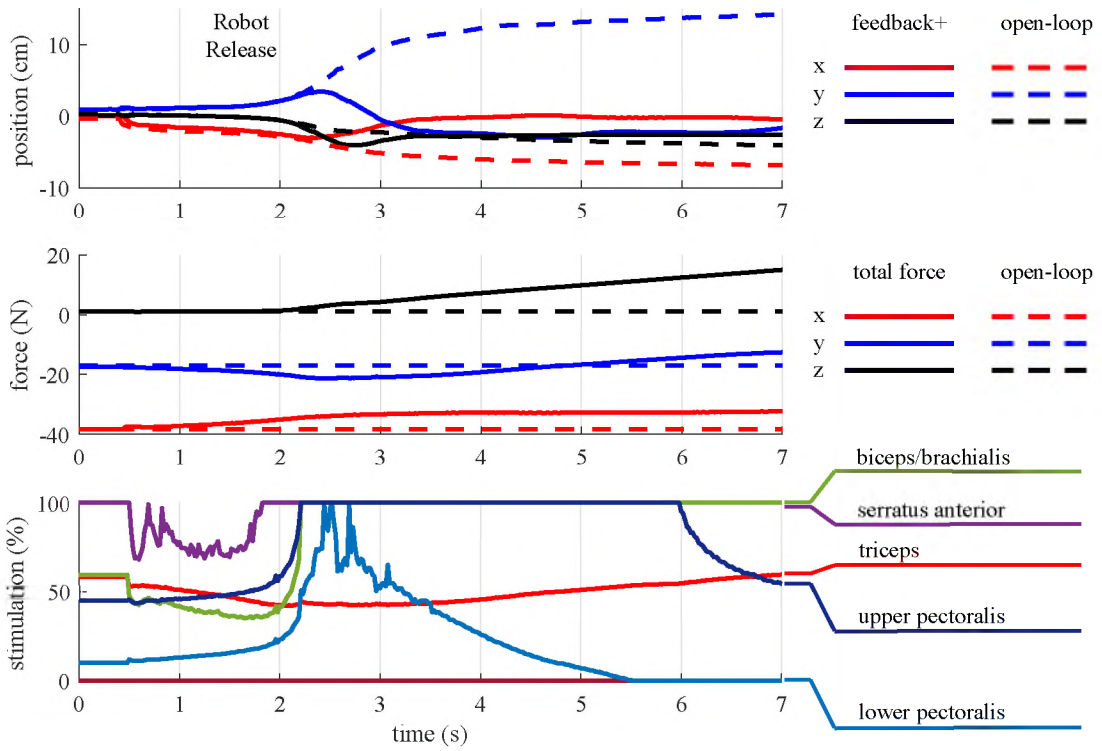



Figure 10: This figure shows the controller performance during a trial of a representative target. The top plot shows the time response of the position of the wrist (adjusted so the target is at 0) for a single target with the open-loop (dashed lines) and the feedback+ (solid lines) controllers. The middle plot shows the total force input to the inverse muscle force block (see Fig. 9) along with the open-loop force commands (dashed lines). The bottom plot shows the stimulation levels (as a percentage of the maximum pulse-width defined in Table I) for all muscles during the trial. (The latissimus dorsi, deltoids, supraspinatus/infraspinatus, and rhomboids were not active during this trial.)

six seconds, as the $y$ desired force continued to increase, the upper pectoralis stimulation command began to decrease since the lower pectoralis was already at 0% stimulation.

Overall, the feedback+ controller held static wrist positions with better accuracy than the open-loop controller. As seen in Fig. 10 (top), the open-loop position moved away from the target to a new final position while the feedback would drive the wrist back towards the target. The mean accuracy and precision results for all trials of all targets are seen in Table II. For the open-loop controller, the mean accuracy (standard deviation) was $12.3\,\mathrm{cm}$ ($9.5\,\mathrm{cm}$). The mean accuracy of the feedback+ controller was $2.9\,\mathrm{cm}$ ($2.2\,\mathrm{cm}$). There was a

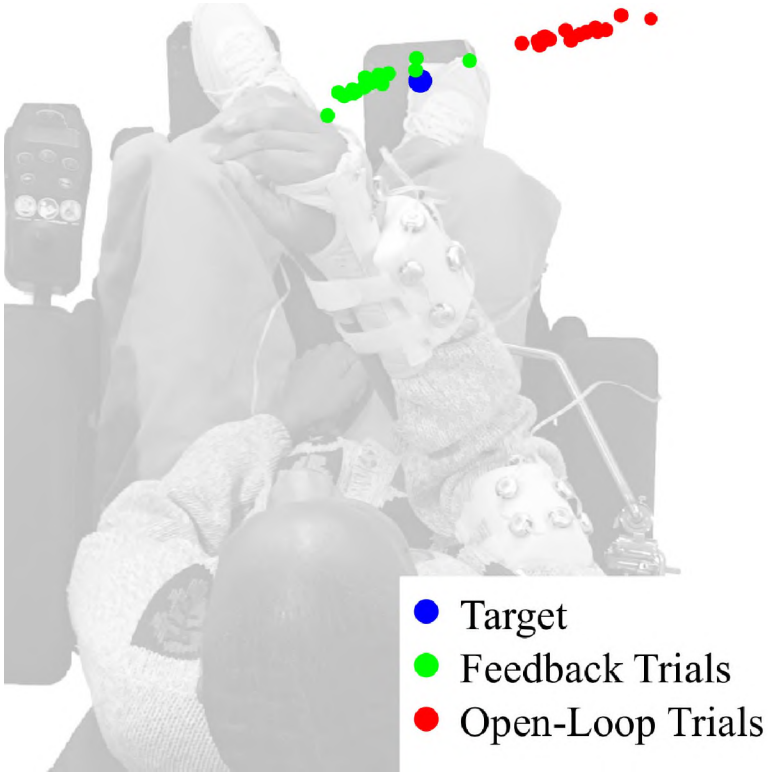



Figure 11: Representative example showing the final positions for each trial for a single target (blue). The open-loop trials (red) had an average accuracy of $10.4\,\mathrm{cm}$ and precision of $7.1\,\mathrm{cm}$. The feedback+ trials (green) had an average accuracy of $3.7\,\mathrm{cm}$ and precision of $7.0\,\mathrm{cm}$.

significant improvement in the accuracy of the feedback+ controller compared to the open-loop controller ($p < 0.001$). The example in Fig. 10 had a similar performance with an accuracy of $16.0\,\mathrm{cm}$ for the open-loop controller and $3.3\,\mathrm{cm}$ for the feedback+ controller. The complete set of trials for this target are shown spatially in Fig. 11 with an average open-loop accuracy of $10.4\,\mathrm{cm}$ and feedback+ accuracy of $3.7\,\mathrm{cm}$.

Table II: Comparison of Controllers

| Mean (standard deviation) | Open-loop | Feedback+ | p-value |
|---|---|---|---|
| Accuracy (cm) | 12.3 (9.5) | 2.9 (2.2) | <0.001 |
| Precision (cm) | 7.7 (8.7) | 3.7 (1.9) | 0.13 |
| Maximum error (cm) | 12.7 (9.6) | 6.1 (3.4) | <0.001 |
| 5% settling time (s) | 4.3 (1.3) | 6.3 (1.1) | <0.001 |

The mean precision (standard deviation) for the open-loop controller was 7.7 cm (8.7 cm). The mean precision for the feedback+ controller was $3.7\,\mathrm{cm}$ ($1.9\,\mathrm{cm}$). There was not a significant improvement in the precision of the feedback+ controller compared to the open-loop controller ($p = 0.13$). Fig. 11 shows trials spatially with an open-loop precision of $7.1\,\mathrm{cm}$ and a feedback+ precision of $7.0\,\mathrm{cm}$.

Figure 10 shows the representative contribution of feedback in a trial. Over all trials, the feedback controller produced a median change of 97% from the open-loop forces. For example, this means that a trial starting with a desired open-loop force of $10\,\mathrm{N}$ would end with a desired force of $19.7\,\mathrm{N}$. The feedback was dominated by the integral component with it, on average, accounting for 75% of the maximum amount of force change desired due to the feedback controller. Additionally, as shown in the figure, the feedback+ controller was able to produce a significantly smaller maximum error (overshoot). The average time response for each controller was quantified (defined by the settling time and maximum error) as seen in Table II. The open-loop controller had a significantly lower settling time ($p < 0.001$). The feedback+ controller had a significantly lower maximum error ($p < 0.001$).

There was not a significant difference in the performance of the open-loop controller on Day One vs Day Two ($p = 0.61$). There was also not a significant difference in the performance of the feedback+ controller on Day One vs Day Two ($p = 0.076$).

## 4.4 Discussion and Conclusion

We have presented a combined feedforward-feedback (feedback+) controller for holding any feasible static wrist position of a paralyzed human arm controlled by FES and have quantified its performance throughout the workspace. Overall, the addition of feedback to the controller produced better performance.

The accuracy of $2.9\,\mathrm{cm}$ was an improvement to the single joint accuracy achieved by [28]. The authors of this paper controlled the elbow joint angle over a trajectory using feedback and co-activation of antagonist muscles. An rms error of approximately $9°$ was achieved for trajectories with no disturbances. With the length of our participant's arm, $57\,\mathrm{cm}$, and translating the error to the shoulder joint, this error would result in a wrist position error of $9\,\mathrm{cm}$. Our controller has been demonstrated for only static purposes, but the improved accuracy while including the degrees of freedom at the shoulder is encouraging to applying our controller to full reaching trajectories.

The accuracy found in the study was also an improvement over our previous study using open-loop control [15] and is useful in similar applications to those achieved in the BrainGate2 study [12]. An accuracy of $2.9\,\mathrm{cm}$ maintained over a trajectory would be good enough for many reaching tasks including combing one's hair or picking up a large piece of food on a plate like a sandwich. Finer movements, such as picking up a small vegetable with a fork, would require improved accuracy. The BrainGate2 study used a set of stimulation patterns for each joint, and the participant used an intracortical brain-computer interface (iBCI) to select the position on the stimulation pattern and achieve the desired arm motion. The main failure mode was due to control interface challenges which demonstrates the challenge of controlling joint dynamics directly. A low-level controller is necessary to

account for these joint dynamics and allow the participant to focus on high-level goal inputs such as a target position in Euclidean space. The ability to focus on high-level control inputs also allows for additional control interfaces, such as an eye-gaze system, for individuals who cannot or do not wish to use an iBCI due to the required brain surgery. This paper demonstrates that our controller, with some improvement, has the potential to be a low-level controller for FES-controlled arm motions for a high-level control input such as the iBCI used in the BrainGate2 study.

Our accuracy of $2.9\,\mathrm{cm}$ was worse than the tracking accuracy of less than $2\,\mathrm{cm}$ found in [16] where they completed arbitrary planar movements with a healthy participant. Removing the planar constraints, however, makes the control more difficult due to the increased degrees of freedom. The relative performance of our controller in a 3D workspace while working with an SCI participant is promising for moving forward with the controller to full-arm reaching.

Many methods of identifying muscle models have been proposed throughout the years. The vast majority of such literature has focused on identifying the models for a single muscle acting on a single degree of freedom. Examples of this include the use of such methods in identifying the parameters of a muscle model about the knee [29, 30]. These types of methods have been expanded upon to identify muscle models for two muscles in the upper extremity [13]. Other upper-limb system identification methods have been performed for single degrees of freedom [28] or in a restricted workspace [16]. While the speed and accuracy of these methods have improved, the requirements to model the entire arm still make them impractical for full-arm reaching. Our method defined in this paper rises to the challenge of identifying a model of the entire arm of a person with a spinal cord injury using a limited amount of data. The model can immediately be used as a controller to be used for full-arm reaching tasks.

Achieving arbitrary, 3-dimensional reaching motions requires an accurate model. The work in [16] shows that a model of the muscles and their actions is necessary for good con-

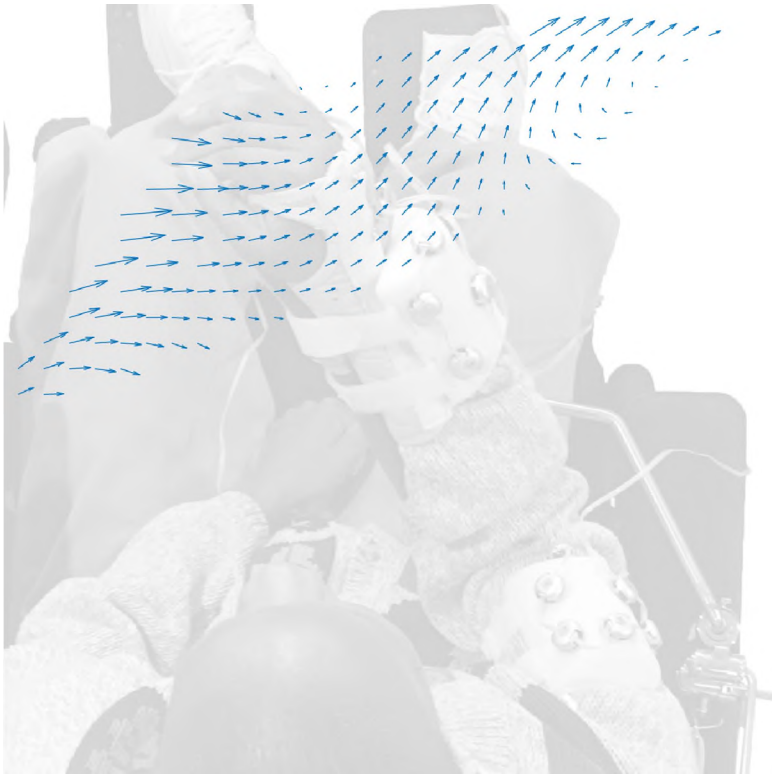


Figure 12: Image showing the modeled direction of force produced at the wrist by the deltoids throughout the workpspace. The direction of force changes based on the position and orientation of the wrist.

trol. For feedback to work correctly, our controller must know the correct direction of force induced at the wrist by each muscle. In a 2D workspace, this is relatively easier as each muscle essentially acts about a single degree of freedom. However, in 3-dimensions, many muscles (especially in the shoulder) act about multiple degrees of freedom. If we consider the deltoids, the action of the muscle of arm abduction would lead to an expected positive force in the $z$ direction. Fig.12 shows the direction of the force produced by the deltoids in the $x$-$y$ plane according to our model. In configurations to the left side of the workspace, the deltoids produce a force almost entirely perpendicular to the participant's chest, but towards the right side of the workspace the deltoids produce a force which pushes away from the participant's chest. It is necessary to know the force produced in all directions to accurately control reaching.

This accurate model is critical to having a controller in 3-dimensional space which can

automatically select the muscle stimulation levels. In the trial shown in Fig. 10, it is not completely intuitive which muscles were selected to achieve the desired forces. For example, it is not clear as to why the biceps increased in activation instead of the triceps decreasing activation since they are often considered simple antagonist muscles about the elbow. The stimulation pattern selected by the controller was most likely due to the muscle actions in other degrees of freedom. Without an accurate model and a method of automatically selecting the muscles for a given reach, it would not be possible to intuitively make these muscle choices.

The required complexity for 3-dimensional control of the entire upper-limb demands a significant amount of time to complete the system identification. System identification methods which require less time have been presented in works such as [16, 28], but most focus on single degrees of freedom or constrained workspaces. The data gathering for our model identification took place over the course of approximately 2.5 hours. Our work models a 6-dimensional workspace of the wrist position and orientation. This large increase in dimensionality requires significantly more information compared to single joint control methods (1D workspace) and planar methods (2D workspace). Our system also requires modeling for controlling nine muscle groups as opposed to only two muscles in single joint systems or even five muscles in [16]. This increase in control inputs requires more data to accurately model. Additionally, an individual with SCI requires more frequent breaks than a healthy individual which increases the amount of time required to gather the data.

A drawback of the amount of time required for our system is that, for real-world use, it is difficult to complete the identification frequently to account for day-to-day changes in the model. Additionally, there could be rapid changes to the system in real-time (for example, if the individual picks up an object) which would lead to errors in the model. However, as our controller has demonstrated, the addition of feedback is able to account for errors in the modeling or changes in the system over several days. Therefore, the system identification will need to be performed at less frequent intervals as opposed to daily and the system can

account for changes due to picking up objects.

To improve controller performance, improved modeling or model adaptations may still be necessary. For several open-loop targets, the wrist would start at the target, drift slowly away for a second or two, and then quickly accelerate to the far right extreme of the participant's workspace. It was noticed that this seemed to occur due to the triceps causing elbow extension when other muscles caused internal rotation of the shoulder. Internal rotation of the shoulder would cause the triceps direction of force to change from one that is pushing forward, to a force pushing to the right. This internally rotated shoulder does not passively occur and therefore is not seen during the muscle identification procedure (the triceps model is developed with only the triceps active). It is likely that performance could be improved by using a richer amount of data which could better include the changes in orientation which occur when multiple muscles are activated. However, compared to our current method of identifying joints individually, stimulating multiple muscles would not leverage the independence of muscles and would require significantly more time. To improve our modeling without adding more identification time, we aim to develop a system of updating the muscle models during control tasks to improve the system performance.

More advanced controllers may also be necessary to improve the system's performance. Our controller has a relatively slow response as shown by the high settling time because it is driven strongly by integral control (Fig. 10). This slow response leads to the wrist moving an average of 6.1 cm away from the target before the feedback pushes the wrist back towards the target. The controller gains were selected to improve the accuracy of the controller while limiting oscillations which can be uncomfortable to the participant. Due to the system dynamics and time delays in the system, increasing the proportional and derivative gains led to oscillations. Techniques for accounting for these issues, including electromechanical delay, have been developed but generally only for single joint systems [31]. Developing and applying these techniques to our complete arm system may help further improve the controller performance.

A common issue in FES control is the rapid fatigue in the muscles which have been controlled. During this experiment, we did not notice any significant changes in the performance of the controller over the course of a day (though this was not explicitly tested for). For a single trial (or trials spaced out over time), the controller seems to be able to account for changes in muscle dynamics due to fatigue (or other disturbances) as demonstrated in the performance found in this study.

The goal of this paper was to develop a controller capable of achieving reaching tasks. Though demonstrated for static positions, our control architecture shows promise in achieving full reaching tasks. We propose using the controller (with the stated improvements) as a quasi-static controller. The wrist will move along a path of feasible static wrist positions connecting a starting position to the end goal position. The path of feasible points will be selected from the set of feasible configurations as defined by the model in this paper. The controller presented in this paper demonstrated the capability of achieving the static wrist positions. By shifting the desired static position, we will be able to move the wrist along any desired path.

The main contribution of this paper is the development of a data-driven-model-based feedback controller for 3-dimensional wrist-position control of an FES-controlled paralyzed human arm. Our controller accurately and consistently holds feasible static wrist positions while maintaining the muscular redundancy of the arm. However, improved performance may be necessary for finer motions. Improved modeling and model updates are the clearest opportunity to do so. Using this controller, FES-controlled full-arm reaching motions can be achieved by commanding a sequence of static positions along a path connecting a starting position to a goal position.

## Acknowledgments

This work was supported by NIH NINDS grant N01-NS-5-2365, Veteran's Affairs grant B2359-C, NSF grant 1751821, and the Cleveland State University Faculty Research and

Development Program.

## 4.5 REFERENCES


[1] NSCISC. Spinal cord injury facts and figures at a glance. National Spinal Cord Injury Statistical Center, 2016.

[2] K. D. Anderson. Targeting recovery: Priorities of the spinal cord-injured population. *Jounral of Neurotrauma*, 21(10):1371–1383, 2004.

[3] H. Kobravi and A. Erfanian. A decentralized adaptive fuzzy robust strategy for control of upright standing posture in paraplegia using functional electrical stimulation. *Medical Engineering & Physics*, 34(1):28–37, 2012.

[4] K. T. Ragnarsson. Functional electrical stimulation after spinal cord injury: Current use, therapeutic effects and future directions. *Spinal Cord*, 46:255–274, 2008.

[5] P. H. Peckham and J. S. Knutson. Functional electrical stimulation for neuromuscular applications. *Annual Reviews in Biomedical Engineering*, 7:327–360, 2005.

[6] W. D. Memberg, K. H. Polasek, R. L. Hart, A. M. Bryden, K. L. Kilgore, G. A. Nemunaitis, H. A. Hoyen, M. W. Keith, and R. F. Kirsch. Implanted neuroprosthesis for restoring arm and hand function in people with high level tetraplegia. *Archives of Physical Medicine and Rehabilitation*, 95(6):1201–1211, 2014.

[7] Kathleen M Jagodnik and Antonie J van den Bogert. Optimization and evaluation of a proportional derivative controller for planar arm movement. *Journal of Biomechanics*, 43, 2010.

[8] D. Blana, R. F. Kirsch, and E. K. Chadwick. Combined feedforward and feedback control of a redundant, nonlinear, dynamic musculoskeletal system. *Medical & Biological Engineering & Computing*, 47:533–542, 2009.

[9] Kathleen Jagodnik, Philip Thomas, Antonie van den Bogert, Michael Branicky, and Robert Kirsch. Training an actor-critic reinforcement learning controller for arm movement using human-generated rewards. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 2017.

[10] L Lan, K Y Zhu, and C Y Wen. System for Arm Movement. 9(4):449–479, 2009.

[11] A. Pedrocchi, S. Ferrante, E. Ambrosini, M. Gandolla, C. Casellato, T. Schauer, C. Klauer, J. Pascual, C. Vidaurre, M. Gföhler, et al. MUNDUS project: MUltimodal Neuroprosthesis for daily Upper limb Support. *Journal of NeuroEngineering and Rehabilitation*, 10(1):1–20, 2013.

[12] A Bolu Ajiboye, Francis R Willett, Daniel R Young, William D Memberg, Brian A Murphy, Jonathan P Miller, Benjamin L Walter, Jennifer A Sweet, Harry A Hoyen, Michael W Keith, et al. Restoration of reaching and grasping movements through brain-controlled muscle stimulation in a person with tetraplegia: a proof-of-concept demonstration. *The Lancet*, 389(10081):1821–1830, 2017.

[13] C T Freeman. Upper Limb Electrical Stimulation Using Input-Output Linearization and Iterative Learning Control. *Control Systems Technology, IEEE Transactions on*, PP(99):1, 2014. ISSN 1063-6536. doi: 10.1109/TCST.2014.2363412.

[14] Mustafa Kutlu, Chris Freeman, Ann-Marie Hughes, and Matthew Spraggs. A Home-based FES System for Upper-limb Stroke Rehabilitation with Iterative Learning Control. *IFAC-PapersOnLine*, 50(1):12089–12094, 2017. ISSN 24058963. doi: 10.1016/j.ifacol.2017.08.2153.

[15] Derek N Wolf and Eric M Schearer. Evaluating an open-loop functional electrical stimulation controller for holding the shoulder and elbow configuration of a paralyzed arm. In *2017 International Conference on Rehabilitation Robotics (ICORR)*, pages 789–794. IEEE, 2017.

[16] Reza Sharif Razavian, Borna Ghannadi, and John McPhee. Feedback control of functional electrical stimulation for arbitrary upper extremity movements. In *Rehabilitation Robotics (ICORR), 2017 International Conference on*, pages 1451–1456. IEEE, 2017.

[17] K. H. Polasek, H. A. Hoyen, M. W. Keith, R. F. Kirsch, and D. J. Tyler. Stimulation stability and selectivity of chronically implanted multicontact nerve cuff electrodes in the human upper extremity. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 17(5):428–437, 2009.

[18] B. Smith, P. H. Peckham, M. W. Keith, and D. D. Roscoe. An externally powered, multichannel, implantable stimulator for versatile control of paralyzed muscle. *IEEE Transactions on Biomedical Engineering*, 34(7):499–508, 1987.

[19] B. Smith, Z. Tang, M. W. Johnson, S. Pourmehdi, M. M. Gazdik, J. R. Buckett, and P. H. Peckham. An externally powered, multichannel, implantable stimulator-telemeter for control of paralyzed muscle. *IEEE Transactions on Biomedical Engineering*, 45(4):463–475, 1998.

[20] R. L. Hart, N. Bhadra, F. W. Montague, K. L. Kilgore, and P. H. Peckham. Design and testing of an advanced implantable neuroprosthesis with myoelectric control. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 19(1):45–53, 2011.

[21] W. D. Memberg, P. H. Peckham, and M. W. Keith. A surgically-implanted intramuscular electrode for and implantable neuromuscular stimulation system. *IEEE Transactions on Rehabilitation Engineering*, 2(2):80–91, 1994.

[22] G. G. Naples and J. T. Mortimer. A spiral nerve cuff electrode for peripheral nerve stimulation. *IEEE Transactions on Biomedical Engineering*, 35(11):905–916, 1988.

[23] Eric M Schearer, Yu-Wei Liao, Eric J Perreault, Matthew C Tresch, William D Memberg, Robert F Kirsch, and Kevin M Lynch. Semiparametric identification of human

arm dynamics for flexible control of a functional electrical stimulation neuroprosthesis. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 24(12): 1405–1415, 2016.

[24] G. Wu, F. C. van der Helm, H. E. Veeger, M. Makhsous, P. Van Roy, C. Anglin, J. Nagels, A. R. Karduna, K. McQuade, X. Wang, F. W. Werner, and B. Buchholz. ISB recommendation on definitions of joint coordinate systems of various joints for the reporting of human joint motion – part II: Shoulder, elbow, wrist and hand. *Journal of Biomechanics*, 38(5):981–992, 2005.

[25] C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. The MIT Press, Cambridge, MA, 2006.

[26] Muriel Lang and Sandra Hirche. Computationally Efficient Rigid-Body Gaussian Process for Motion Dynamics. *IEEE Robotics and Automation Letters*, 2(3):1601–1608, 2017. ISSN 2377-3766. doi: 10.1109/LRA.2017.2677469.

[27] W. K. Durfee and K. E. MacLean. Methods for estimating isometric recruitment curves of electrically stimulated muscle. *IEEE Transactions on Biomedical Engineering*, 36(7):654–666, 1989.

[28] Antônio Padilha L Bó, Lucas O da Fonseca, and Ana Carolina C de Sousa. Fes-induced co-activation of antagonist muscles for upper limb control and disturbance rejection. *Medical Engineering & Physics*, 38(11):1176–1184, 2016.

[29] Maurizio Ferrarin, Francesco Palazzo, Robert Riener, and Jochen Quintern. Model-Based Control of FES-Induced Single Joint Movements. 9(3):245–257, 2001.

[30] Identification of electrically stimulated quadriceps muscles in paraplegic subjects. *IEEE Transactions on Biomedical Engineering*, 46(1):51–61, 1999. ISSN 00189294.

[31] N. Alibeji, N. Kirsch, S. Farrokhi, and N. Sharma. Further results on predictor-based control of neuromuscular electrical stimulation. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 23(6):1095–1105, 2015.

# CHAPTER V

# CONTROLLING SIMPLE QUASI-STATIC MOTIONS AND PRACTICAL IMPLEMENTATION SOLUTIONS FOR FUNCTIONAL ELECTRICAL STIMULATION-CONTROLLED REACHING MOTIONS

*Functional electrical stimulation is a promising solution to achieving reaching motions in individuals with tetraplegia. In this chapter, we present a simple model-based feedback controller that uses no path planning to drive the arm towards a desired wrist position. The controller demonstrates the potential of our model-based controller to achieve reaching motions with an individual with a spinal cord injury. However, the controller saw significant oscillation and improved accuracy was needed. Building on the results of the simple controller, we develop a simulation study to determine the conditions for which a quasi-static controller can best control reaching motions with functional electrical stimulation. The main contribution of this chapter is developing a set of practical implementation requirements for successful functional electrical stimulation-driven reaching motions. The two main improvements recommended by this chapter are the need for intermediate path planning and the need to use external damping to control oscillations. These results inform the straight-line reaching controller described in VI.*

**Conference publication:**

1. Derek N Wolf and Eric M Schearer. Simple quasi-static control of functional electrical stimulation-driven reaching motions. In *2019 9th International IEEE/EMBS Conference on Neural Engineering (NER)*, parges 211-214. IEEE 2019.

2. Derek N Wolf and Eric M Schearer. Developing a quasi-static controller for a paralyzed human arm: a simulation study. In *2019 16th IEEE International Conference on Rehabilitation Robotics (ICORR)*. IEEE, 2019.

## 5.1 Introduction

Functional electrical stimulation (FES) is a promising technology that restores movement to paralyzed muscles by delivering electrical current to the nerves and muscles directly. Using FES to control reaching motions could allow individuals with paralysis from spinal cord injuries to regain their independence.

Achieving reaching motions with FES has proven difficult due to the complexity of human arm motions. For repetitive tasks such as standing [1] and hand function [2], the complexities of the nonlinear, redundant musculoskeletal system have been overcome with predetermined fixed stimulation patterns. The goal-directed nature of reaching motions, however, requires different and potentially new stimulation patterns for every reach.

Many different control strategies have been proposed to achieve FES reaching. The state-of-the-art strategy, demonstrated in the BrainGate2 clinical trial, simultaneously controlled each joint independently [3]. For each joint, the controller selected (based on user intent recorded by an intracortical brain computer interface) a position along a predefined stimulation pattern. While the user's intent was accurately perceived, it was difficult to control multi-joint movements because the independent joint control could not account for the interactions of the joints. To accurately control reaching motions, it is necessary to treat the arm as a complete system.

In simulations, optimized proportional-derivative control [4], combined feedforward-feedback control [5], reinforcement learning [6], and threshold control [7] have all proven successful in controlling reaching motions. Practical implementation of these methods has proven difficult due to the real-world arm dynamics differing from the simulation.

To control the arm as a complete system and determine the real-world arm dynamics necessary for accurate control, model-based methods have been proposed in previous works. Physics-based models have shown some success in controlling two muscles for rehabilitation after stroke [8]. However, identifying the physical parameters of the whole arm requires significant amounts of data. Black-box model-based control methods have

been developed to help solve this issue. One such method achieved feedback control of planar arm tasks using an artificial neural network to produce a map of the task space configuration to the forces the muscles can produce [9]. We have used similar nonparametric and semiparametric concepts to produce a model-based controller capable of holding three-dimensional static arm configurations [10][11]. However, extending these methods to full-arm model-based three-dimensional reaching control has not yet been achieved.

Previous work has demonstrated that a semiparametric Gaussian Process Regression (GPR) model could form the basis of a controller for achieving three-dimensional dynamic trajectories [12]. However, including velocity in the controller made it difficult to select trajectories that could be physically achieved with FES. Instead, we have developed a combined feedforward-feedback controller for holding static wrist positions [11]. To achieve reaching movements, we propose using a version of this controller to move the arm through a quasi-static reaching motion. We define a quasi-static path as a series of intermediate static positions which connect the starting arm configuration to the target arm configuration. The controller will shift the desired static target wrist position to the next position in the path until the final target is reached.

As an initial attempt at achieving quasi-static reaching motions and to guide the development of our controller, we first implemented a simple quasi-static controller which only consisted of the target position and no intermediate points. The results of this study demonstrated the need for intermediate path planning as well as a reduction in oscillation. Therefore, we decided to try to implement the full quasi-static controller with intermediate static positions.

Before implementing the quasi-static controller with a human subject, we conducted a simulation study presented here. The simulation study was designed to replicate the experimental conditions which we work with during our research with human subjects [11]. This includes the use of an elastic arm support which people with high tetraplegia use to assist against gravity to produce a more effective reachable workspace.

Our simple quasi-static controller work demonstrated that the elasticity of the arm support can combine with the time delays inherent to an FES system (the relatively slow frequency of switching the stimulation signal and muscular activation dynamics) to produce oscillations (which can be uncomfortable for the subject) that basic derivative control is unable to eliminate. Advanced control techniques have been developed to account for these delays in single joint movements [13] as well as in a model-based controller of a simulated arm [14]. Applying these techniques to an FES-controlled human arm, where the parameters of the model are not easily determined, has yet to be implemented. As a simple solution to practically solving the oscillation issue, we propose the addition of damping to the arm support.

The goals of the simulation study are to demonstrate the feasibility of a quasi-static controller for controlling reaching and to determine the conditions for successful reaching. A secondary goal is to study the effects of adding physical damping to the arm support.

## 5.2 Simple Quasi-static Control of Functional Electrical Stimulation-Driven Reaching Motions

### 5.2.1 Methods

We assessed the efficacy of our model-based controller over a single day of experiments. The controller used a model of the arm of an individual with tetraplegia to automatically determine the stimulation commands necessary to achieve a desired wrist position. The experiments took place over approximately two hours.

#### Experimental Setup

A single human participant who has high tetraplegia participated in our experiments. The participant was a 60-year-old female who sustained a hemisection of the spinal cord at the C1-C2 level. She cannot voluntarily move her right arm (the arm with which we performed our experiments) but does have sensation. She experiences hypertonia in some of the arm

muscles. A passive arm support produces a comfortable and achievable workspace by using elastic bands to assist against the force of gravity. The arm support creates a resting equilibrium position with the wrist approximately at the height of and centered upon the participant's chest. More details can be found in [15] (Subject 1).

The participant is implanted with a stimulator-telemeter in her abdomen [16][17][18]. The device has leads which transmit current to intramuscular electrodes [19] and nerve cuff electrodes [20] activating muscles in her right arm and shoulder complex. We refer to each muscle or group of muscles stimulated by a single electrode as a muscle group. In this experiment, we controlled nine muscle groups including the triceps, deltoids, latissimus dorsi, serratus anterior, biceps and brachialis, supraspinatus and infraspinatus, rhomboids, lower pectoralis, and upper pectoralis. A computer sends power and control signals to the implanted device via an inductive radio-frequency link. Muscle stimulation uses bi-phasic, charge balanced pulses delivered at 13 Hz. The amplitude of the pulses is constant for each muscle group. The force generated by each muscle group is controlled by varying the pulse-width (referred to as the stimulation command) from 0-250 $\mu$s. The maximum stimulation command for each muscle was determined as the point when no additional muscle force was achieved or the participant reported discomfort. The control input is the vector containing the stimulation command for every muscle group. Stimulation commands are sent to the implant using real-time control code on a computer. Protocols used for this research were approved by the institutional review boards at Cleveland State University (IRB NO. 30213-SCH-HS) and MetroHealth Medical Center (IRB NO. 04-00014).

An Optotrak Certus Motion Capture System (Northern Digital, Inc.) captured data used to estimate the arm's configuration. The arm's configuration was defined by the position and orientation of the wrist relative to the thorax. The motion capture system was also used to measure the real-time position of the wrist to be used for feedback during the static hold experiments. A third-order moving-average filter was used on the wrist position signal to achieve smooth velocities.

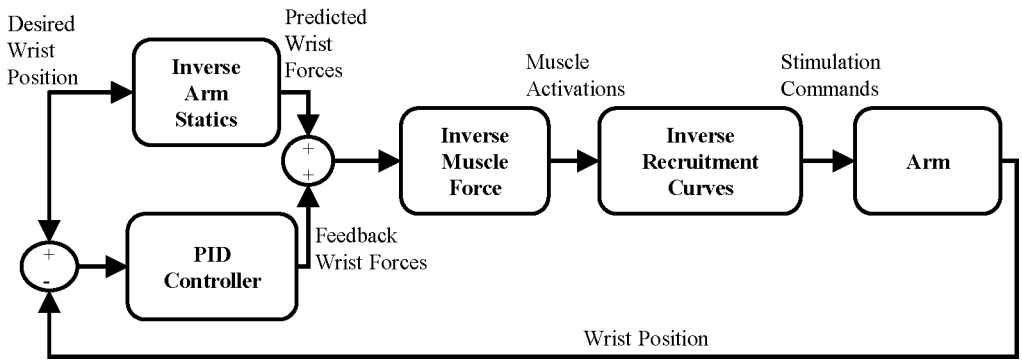



Figure 13: Controller block diagram

The experiment was controlled using MATLAB xPC target on a Dell Dimension 8400 PC with a Pentium 4 3.20 GHz processor. The control and data collection occurred at 52 Hz, but stimulation inputs were updated at the stimulation frequency of 13 Hz.

### Controller

Our controller (Fig. 13) aims to automatically determine the stimulation commands necessary to achieve a desired static wrist position. The controller was developed in detail in [11], but some details are repeated here for clarity. While the controller was developed and has been tested for static wrist positions, this study used the same control structure to achieve movement from a starting position to a different goal position.

The controller uses a subject specific, data-driven model of the arm's statics and response to stimulation. The model identification details are presented in [11]. The resulting model consists of three parts: the inverse arm statics (the mapping from a wrist position and orientation to the forces needed to maintain that wrist position), muscle force production (the amount of force each muscle can produce in a given configuration), and recruitment curves (the mapping from stimulation command to the muscle group activation). This model forms the basis for the blocks in the controller.

The input to the controller is the desired wrist configuration (defined as the position and orientation), $q$, which corresponds to the desired wrist position. The inverse arm stat-

ics model is then used to predict the wrist forces necessary to hold the desired static wrist position. The forces are then added to the compensation forces determined by a PID controller based on the error of the wrist position. The resulting output of the summation are the desired forces, $\mathbf{f}^*$, at the wrist needed to achieve the desired wrist position.

The desired force is mapped to the muscle activations needed to achieve the desired force in the "inverse muscle force" block. The muscle force production model forms the basis of this block. The output of the model is the linear mapping, $\mathbf{R}(\mathbf{q}) \in \mathbb{R}^{\mathbf{3}\times\mathbf{9}}$, of the muscle activation to the force at the wrist in each Cartesian direction for a given arm configuration. The $j^{\text{th}}$ column of $\mathbf{R}(\mathbf{q})$ represents the force produced in each Cartesian direction by 100% activation of the $j^{\text{th}}$ muscle group. To determine the required muscle activations, $\boldsymbol{\alpha} \in \mathbb{R}^{9\times 1}$ to achieve the desired force, we must find a solution to the equation

$$\mathbf{f}^* = \mathbf{R}(\mathbf{q})\boldsymbol{\alpha}. \tag{5.1}$$

$\mathbf{R}(\mathbf{q})$ is not square because there are more muscle groups than degrees of freedom. We resolve the redundancy in real-time using the quasi-Newton method to find the $\boldsymbol{\alpha}$ that minimizes the penalty function,

$$\begin{aligned} &||\boldsymbol{\alpha}||_2^2 + c_1\|\mathbf{R}(\mathbf{q}_*)\boldsymbol{\alpha} - \mathbf{p}(\mathbf{q}_*)\|_{\mathbf{2}}^{\mathbf{2}} + c_2 K \\ &K = \sum k_i \text{ where } k_i = \begin{cases} \boldsymbol{\alpha}_i^2 \text{ if } \boldsymbol{\alpha}_i < 0 \\ (\boldsymbol{\alpha} - 1)^2 \text{ if } \boldsymbol{\alpha}_i > 1 \\ 0 \text{ if } 0 \leq \boldsymbol{\alpha}_i \leq 1 \end{cases}, \end{aligned} \tag{5.2}$$

where $||\boldsymbol{\alpha}||_2^2$ minimizes the muscle activations, $c_1\|\mathbf{R}(\mathbf{q}_*)\boldsymbol{\alpha} - \mathbf{p}(\mathbf{q}_*)\|_{\mathbf{2}}^{\mathbf{2}}$ penalizes activations that do not produce the desired force, and $c_2 K$ penalizes activations which do not belong to $\alpha_i \in [0, 1]$. $c_1$ and $c_2$ were chosen to be 500 and 50,000 respectively.

Once the required activations are found, the modeled recruitment curves are inverted to map the muscle group activations to the stimulation commands which produce those

activations. These stimulation commands are then applied to the arm and the resulting wrist motion is tracked.

### Desired Wrist Position Experiments and Analysis

We quantified the performance of the controller to achieve desired wrist positions at various targets throughout the participant's workspace. For each trial, the participant's arm began at the resting equilibrium. A desired wrist target position was selected and used as the input for the controller. It is important to point out that the predicted wrist forces and $\mathbf{R}(\mathbf{q})$ within the inverse muscle force block are based on the target configuration and are constant for a trial. The controller applied the determined stimulation commands to the arm for seven seconds with the goal of driving the wrist to the target wrist position. The position of the wrist was recorded throughout the trial.

Thirteen targets were selected from a $3\times3\times3$ grid of targets which filled the subject's reachable workspace. The chosen targets were selected based on subject comfort as well as maintaining a targets spread throughout the workspace. Within each set of thirteen targets, each target was repeated twice, once with the predicted wrist forces all equal to zero (referred to as the feedback trials) and once with a combined feedforward-feedback controller (referred to as feedback+ trials). The experiment was repeated over four sets with a random order of the targets in each one.

To analyze the performance of the two controllers, the accuracy for each trial was determined by the Euclidean distance of the mean wrist position over the final second of the trial and target wrist position. The accuracy for a controller was calculated as the mean accuracy across all trials using that control strategy. Additionally, the system response was quantified by calculating the 5% settling time and the maximum error for each trial. The 5% settling time was determined as the time after which the distance between the wrist position and the target wrist position stayed within 5% of the final accuracy. The performance of the feedback and feedback+ controllers were compared using t-tests.

Table III: Comparison of Controllers

| Mean (standard deviation) | Feedback | Feedback+ | p-value |
|---|---|---|---|
| Accuracy (cm) | 4.3 (3.3) | 4.9 (3.5) | 0.41 |
| Maximum error (cm) | 9.5 (4.5) | 119.5 (5.3) | 0.02 |
| 5% settling time (s) | 5.7 (1.8) | 5.8 (1.4) | 0.84 |

### 5.2.2 Results

The feedback+ and feedback controllers generally performed with similar accuracy and settling time. The feedback controller performed with a lower maximum error than the feedback+ controller. The maximum error is defined as the largest Euclidean distance of the wrist position from the target during a trial.

A time history of the wrist position during representative trials using each controller for a single target is shown in figure 14. As seen, the feedback+ controller initially drove the wrist away from the target position, but the feedback was able to compensate for that movement and drive the wrist back in the direction of the target. The feedback controller is able to more immediately drive the wrist towards the target without the large initial movement. Both controllers moved towards the desired position and finished with similar accuracy.

The overall performances of each controller across all trials are compared in Table III. As seen, the accuracy of 4.3 cm for the feedback and 4.9 cm for the feedback+ were not significantly different ($p = 0.41$). The settling time of 5.7 s for the feedback and 5.8 s for the feedback+ were also not significantly different ($p = 0.84$). The only significant difference between the controllers was the maximum error of 9.5 cm for the feedback and 119.5 cm for the feedback+ ($p = 0.02$).

### 5.2.3 Discussion and Conclusion

Overall, the control architecture presented in this study shows promise in controlling reaching movements, but some improvements are necessary. The average accuracy of the feed-

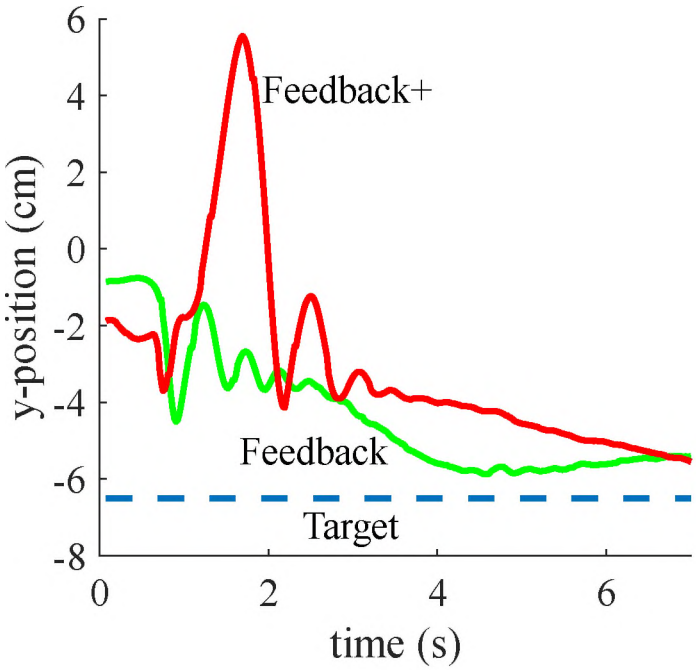



Figure 14: Representative time history for a single target.

back controller of 4.3 cm is slightly worse than the accuracy of 2 cm which has been achieved in 2-dimensional reaching motions with a similar controller [21]. The additional degrees of freedom in 3-dimensional reaching motions, however, significantly increases the complexity of the problem. Thus, the limited decrease in accuracy with a significant increase in difficulty is encouraging moving forward with this control structure.

When comparing the two controllers presented in this study, the feedback+ controller often caused more aggressive initial movements which was seen by the overall difference in maximum error as well as in Fig. 14. This rapid movement away from the target is due to errors in the model. Due to the fact that the accuracy and settling times of the two controllers were similar, it may make sense to move forward with the feedback controller to eliminate the large error caused by the predicted forces in the feedback+ controller. However, it has been shown in simulation that using a combined feedforward-feedback controller results in smoother muscle activation time histories during a trial and better performance in general [5]. Smooth activation profiles are preferable as they would be more comfortable for the

subject. Therefore, improving the feedback+ controller is a worthwhile pursuit.

A major reason for the error in the feedback+ controller may be because the configuration was assumed to be at the goal configuration. Since the feedback was able to push the wrist in the correct direction after a while in both controllers, this assumption seems valid for the inverse muscle force production block. However, the initial movement away from the target position seen in the feedback+ controller would be dominated by the predicted forces from the inverse arm statics block. It is likely that assuming the target configuration led to large errors in the predicted forces (i.e. the forces our model expects are necessary to achieve the position) relative to what was actually needed to move towards the target wrist position. Using the same control structure to maintain static wrist positions after starting in the desired configuration produced a better accuracy of 3.7 cm [11]. This also points to the fact that the controller performs better when the wrist is in the configuration expected by the model.

To improve the controller moving forward, we plan to use a quasi-static control method. In this method, a path of static points will be chosen from the starting position to the goal wrist position. By creating the path of points, the assumed configuration will be closer to the true current wrist position and thus the model should produce more accurate forces and thus better performance. Further research is necessary to determine the required distance between each static position and the amount of time necessary to move between positions. While dynamic trajectories could be useful and may improve the performance, dynamic models of the arm require significantly more data and model complexity. While the velocities must remain low, moving forward with a quasi-static controller would allow for many reaching motions to be achieved with a much simpler model and less demanding system identification.

Overall, we have demonstrated that our control architecture, with the stated improvements, is capable of moving the wrist to a desired wrist position.

## 5.3 Developing a Quasi-Static Controller for a Paralyzed Human Arm: A Simulation Study

### 5.3.1 Methods

The goal of the experiment was to simulate the experimental setup presented in [11] where we used an implanted neuroprosthesis to actuate the arm of an individual with high tetraplegia (We will refer to this experiment as the laboratory study). We used a MATLAB based dynamic simulation of the arm to recreate the conditions of the laboratory study. As in human experiments, we assume that our controller does not have access to the true dynamics of the arm, and we identified a non-parametric model of the response of the simulated human arm to muscle activation inputs. We used this model as the basis of a feedforward-feedback controller. We tuned the controller and added damping to the arm-support to achieve good performance at static positions. The controller was then used to move the arm to points of varying distances to determine the best distance and time between intermediate wrist positions in a quasi-static path. The best parameters were used to control the arm through complete reaching motions and compared to a controller without a path of intermediate static positions.

#### Simulation Experiment Setup

The goal of the simulation study was to recreate the conditions of the laboratory study. The computer simulation thus consisted of a musculoskeletal model of the arm, an elastic arm support, and a robot.

To simulate the subject's arm, we used the Dynamic Arm Simulator, a MATLAB based dynamic model of the arm [22]. The model has seven links, eleven degrees of freedom, and 138 muscle elements. The model includes the multibody dynamics of the links as well as muscle activation dynamics. In all parts of the present study, the time step used in the simulation was 3 ms.

The Dynamic Arm Simulator model is actuated by inputting the neural excitations, $\mathbf{u}$,

which correspond to the desired muscle activations. The neuroprosthesis used in the laboratory study applies stimulation at a frequency of 13 Hz to induce the desired muscle activations. Though the desired control input can be calculated at every time step, switching the input can only occur at the stimulation frequency. In the Dynamic Arm Simulator simulation, we modeled this by restricting $\mathbf{u}$ to only change at 13 Hz. We controlled only the muscle elements which are able to be controlled by the neuroprosthesis in the laboratory study. In the Dynamic Arm Simulator, we controlled the muscle elements related to the triceps, deltoids, latissimus dorsi, serratus anterior, biceps/brachialis, supraspinatus/infraspinatus, rhomboids, lower pectoralis, and upper pectoralis. When a neural excitation is applied to a muscle group, all elements in the group receive the same excitation.

We simulated a passive arm support that is typically used by individuals with spinal cord injuries to assist against the force of gravity and create a more functional workspace. The support used in the laboratory study uses elastic bands to create a resting equilibrium position of the wrist approximately at the top of the thorax slightly forward from the body, and we aimed to simulate the behavior of this arm support. The stiffness of the support were 0 N/m in the X direction and 30 N/m in both the Y and Z directions (due to the orientation of the elastic bands in the laboratory study's arm support, the majority of force is applied in the Y and Z directions). The equilibrium point for the support was placed at [0 m, 0.3 m, -0.15 m] (see the coordinate frame in Fig. 16). Choosing the damping of the support is one of the goals of the study.

In the laboratory study, a robot is used during model identification and at the start of control to hold the wrist in the desired position. In the computer simulation, a PID controller was used to apply force at the wrist to mimic this robot.

We used the included Dynamic Arm Simulator OpenSim model to visualize all trials and experiments [23][24].

### Model Identification

The model identification procedures are developed in detail in [11] and [12]. We present a summary here.

We developed a two-part model consisting of the inverse statics (the mapping from arm configuration to joint torque required to hold the configuration) and the muscle torque production (the mapping from configuration and muscle activation to the joint torques produced). In the same way as the laboratory study, to gather data for the model, the robot held the wrist of the simulated arm in a series of 27 positions throughout the workspace. At each position, each muscle group was individually activated at 100% neural excitation for 0.5 seconds, and for one period of 0.5 seconds, no muscles were activated, $u = 0$. The force required for the robot controller to hold the wrist at a static position and the configuration of the arm were recorded. The configuration of the arm, $\mathbf{q}$, is defined by the angles between the thorax and the humerus (shoulder elevation plane, shoulder elevation, and shoulder rotation) as defined in [25]) as well as the elbow flexion and pronation angles. The force required to hold the wrist position and joint configurations were averaged over the last 10% of each trial. The kinematic Jacobian was used to transform the recorded robot controller force to the joint torques, $\boldsymbol{\tau}_j$, about the shoulder and elbow which produce the equivalent force (j represents the muscle group being activated with 0 representing no muscles being active).

The torques needed to hold the wrist in a static position, $\mathbf{p}(\mathbf{q}) \in \mathbb{R}^{4\times 1}$, (The torque about elbow pronation is not included as it does not affect the position of the wrist.) with no muscles activated represent the arm statics, and therefore,

$$\boldsymbol{\tau}_0 = \mathbf{p}(\mathbf{q}). \tag{5.3}$$

The difference between the torques recorded with no muscles active and the torques recorded with muscle group j active represents the amount of torque produced by muscle group j.

The amount of torque produced by a muscle being activated is represented by $\mathbf{R}(\mathbf{q})\boldsymbol{\alpha}$ where $\boldsymbol{\alpha} \in \mathbb{R}^{9\times 1}$ is the vector of muscle group activations and $\mathbf{R}(\mathbf{q}) \in \mathbb{R}^{4\times 9}$ is the mapping from muscle group activation to joint torque. The $j^{\text{th}}$ column of $\mathbf{R}(\mathbf{q})$, $\mathbf{R}_j(\mathbf{q})$, represents torques about the shoulder elevation plane, shoulder elevation, shoulder rotation, and elbow flexion produced by muscle group j. Therefore,

$$\mathbf{R}_j(\mathbf{q}) = \boldsymbol{\tau}_0 - \boldsymbol{\tau}_j. \tag{5.4}$$

The data from the set of 27 training positions was used to train semiparametric GPR models [26] which are used to predict $\boldsymbol{\tau}_j$ for $j = 0, 1...9$ for a given configuration. The models can therefore be used to determine the static arm torques $\mathbf{p}(\mathbf{q})$ and the muscle force mapping $\mathbf{R}(\mathbf{q})$ for a desired arm configuration.

### Controller

Our controller (see Fig. 15) automatically determines the muscle activations required to hold a static wrist position. The input to the controller is the desired arm configuration, $\mathbf{q}_*$, defined by the three angles between the humerus and the thorax (shoulder elevation plane, shoulder elevation, and shoulder rotation) and elbow flexion and pronation. This controller maps the desired arm configuration to the predicted torques needed to maintain the configuration. These torques are modified by the feedback controller to produce the desired torques necessary to be achieved by the muscles. The output of the controller is the set of neural excitations which correspond to the muscle activations that produce the desired torques, and these excitations are applied to the Dynamic Arm Simulator arm.

The inverse arm statics block uses the model developed in section 5.3.1 to calculate the open-loop joint torques, $\mathbf{p}(\mathbf{q}_*)$, about the shoulder and elbow to hold wrist at the desired position. A PID controller calculates corrective forces in each cardinal direction (X, Y, and Z directions) required to hold the wrist at the desired position. The kinematic Jacobian is

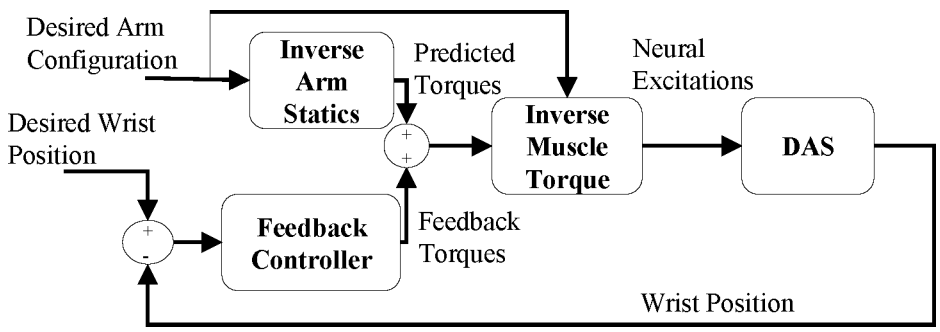



Figure 15: The block diagram for our controller. The controller uses model-based blocks and a feedback controller to automatically calculates the neural excitations to apply to the dynamic arm simulator (DAS) to to achieve the desired wrist position that corresponds to the desired arm configuration.

used to transform these forces to the equivalent torques about the shoulder and elbow. These feedback torques, $\boldsymbol{\tau}_{FB}$, are added to the open-loop torques to produce the total desired torque, $\boldsymbol{\tau}_{total} = \mathbf{p}(\mathbf{q}_*) + \boldsymbol{\tau}_{FB}$.

The controller next maps the total torque to the muscle group activations which produce the desired torques. The controller uses the GPR models of muscle torque production to determine the muscle-torque mapping, $\mathbf{R}(\mathbf{q}_*)$, for the desired configuration. (Note: The inverse arm statics and inverse muscle torques are found for the desired configuration and not the current configuration.) Since the system is redundant, we determine the muscle activations, $\alpha \in \mathbb{R}^{9\times 1}$, which produce the desired total torques by solving an optimization problem which minimizes the muscle activations such that the desired torques are produced and $\boldsymbol{\alpha} \in [0, 1]$. The neural excitations input to the Dynamic Arm Simulator model are equivalent to the desired muscle activations, $u_j = \boldsymbol{\alpha}_j$ for the $j^{\text{th}}$ muscle group. The actual activation achieved is determined by the muscle activation dynamics of the Dynamic Arm Simulator.

The model was used to determine a set of feasible wrist positions throughout the workspace. A grid of wrist positions within the positions used during model identification with 1 cm spacing was produced. The feasibility of each wrist position was determined by the ability of the model to find a set of muscle activations which can produce the required open-loop

torques for the configuration that corresponds to the desired wrist position. The map of feasible points is shown in Fig. 16.

To tune the controller, 15 wrist positions were randomly selected from the map of feasible points. The open-loop portion of the controller was used with the goal of holding the static wrist position for three seconds. The final distance from the desired position was recorded for all trials. The feedback controller was tuned at the position with the median error. The controller was tuned with the goal of improving the accuracy of holding a static position for three seconds while minimizing oscillations. The gains of the PID controller were tuned to increase the accuracy of the holding the static position while also limiting oscillations. As previously discussed, the existence of time delays in the system caused by the 13 Hz frequency of changing neural excitations as well as the muscle activation dynamics meant that simple derivative gain in the FES controller was unable to eliminate oscillations (see Fig. 18). Instead, damping was added to the arm support. The final FES controller was a PI controller with tuned parameters of 250 N/m for the proportional gain and 80 N/m-s for the integral gain. 120 N-s/m was chosen for the damping of the arm support. The gains were the same for the X, Y, and Z directions.

### Quasi-static Path Following

The controller was used to control the wrist to follow a quasi-static path. A series of static wrist positions was selected from the feasibility map (Fig. 16) which connected the starting position to the target position. As the arm moved along the path, the desired configuration shifted to the next static point in the path and was input to the controller.

To determine a suitable distance and amount of time between points in a quasi-static path, point to point reaches of different distances were completed. A starting wrist position and target wrist position a set distance, $d$, away from the start position were randomly selected from the feasibility map. The corresponding target arm configuration was input to the controller and the controller drove the arm for three seconds. The final distance from

the target wrist position and the 80% rise time were recorded for each point. This process was repeated at 100 start wrist positions for distances, $d = 2, 4, 6...16$ cm for a total of 800 reaches. The mean accuracy and mean rise time of all targets of a single $d$ were recorded.

From the single point to point studies, the best parameters (the distance between positions, $d_*$ and the time between switching positions, $t_{switch}$) were selected to test on complete quasi-static reaching paths. The required distance between positions, $d_*$ was selected based upon the accuracy of the point to point reaches. The required time between switching positions, $t_{switch}$, was determined by the average rise time of the trials. The reasoning for using the 80% rise time was to switch to the next target after being close enough to the current target to still have accurate control. If switching occurred fast enough, the motion could become smoother (less stops and starts).

Using the selected parameters, complete quasi-static reaching paths were completed. Start and end positions at least 30 cm apart were randomly selected from the feasibility map. A path of wrist positions, each a maximum distance of $d_*$ cm from the previous point, was selected which connected the two points. As the arm moved along the path, the desired wrist position shifted to the next point in the path every $t_{switch}$. The final position is held for $2t_{switch}$ to allow for the controller to settle at the position. The quasi-static paths were compared to the same controller but with no path of intermediate positions (the goal position is the final target from the initial time step) for the same average speed of reach as the final controller.

The final accuracy (the distance from the average hand position over the final 0.3 seconds of a trial to the final target position) was recorded for each trial. For a grouping of trials, the overall accuracy was determined by the mean of the accuracy for all trials in the group. A t-test was used to compare the quasi-static controller to a simple PI controller with no intermediate positions (referred to as the simple controller).

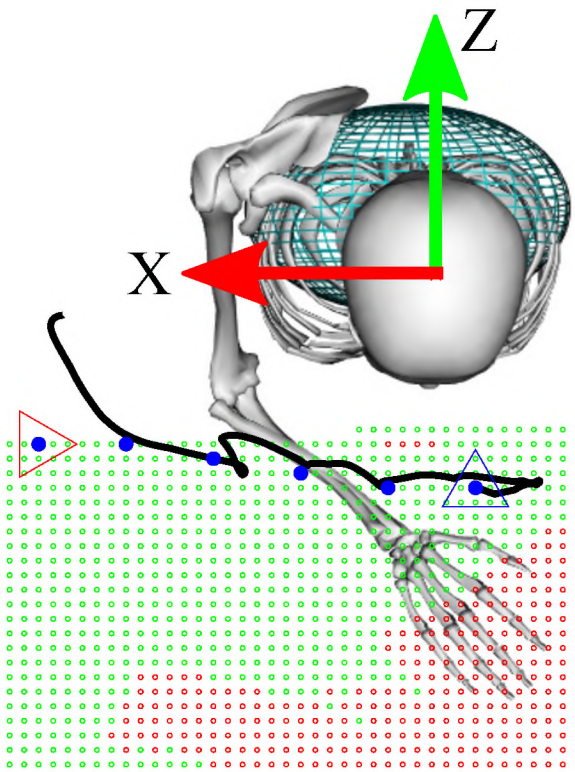



Figure 16: A 2-D projection of the 3-D map of the feasible wrist positions in the modeled workspace. Feasible points (green) are where the controller is able to determine predict a set of muscle activations capable of achieving the model predicted static arm torques. The map was produced with a 1 cm spaced grid of wrist positions. The image also shows an example reaching motion completed by the final quasi-static controller. The intermediate points are shown by blue circles and the triangles represent the start (blue) and end target positions (red). The reach shown was a 32 cm reach with an accuracy of 9.5 cm. As seen, the reach has fairly good accuracy until the arm moves to the edge of the feasible workspace.

### 5.3.2 Results

We have developed a quasi-static controller that is capable of moving the wrist of a simulated arm along a path between a starting position and target position. Using a controller with a distance between positions of 6 cm and a time between switching points of 1.3 seconds, we were able to achieve a median accuracy of 6.8 cm (mean: 10.5 cm, standard deviation: 9.4 cm) over a series of 200 reaches longer than 30 cm. This is significantly better than the simple controller that does not use intermediate points (mean: 20 cm, $p < 0.01$). Fig. 16 shows an example trial of the final controller achieving a 31 cm reach with an accuracy of 0.6 cm.

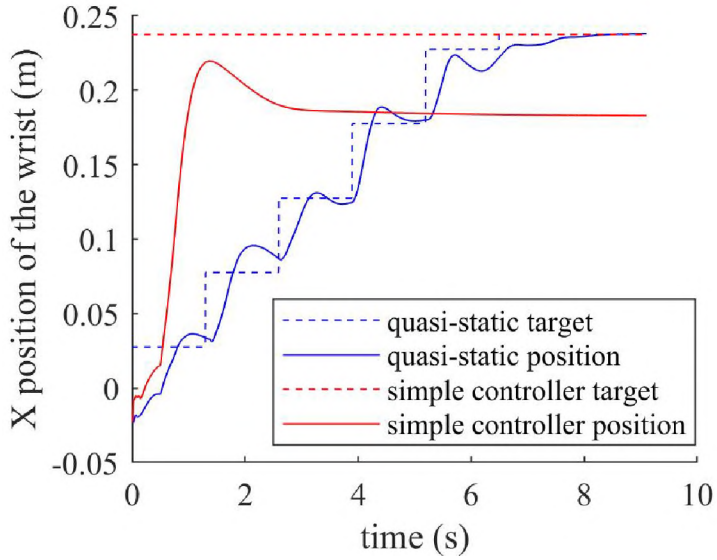



Figure 17: An example of a time history of the X position of the wrist for a reaching motion controlled by the quasi-static controller (blue) and the simple controller where the target position was always the final goal wrist position (red). The accuracy of the quasi-static controller was 1.1 cm and the accuracy of the simple controller was 27.4 cm.

Fig.17 shows the time histories in the X direction of both the quasi-static controller and the simple controller for achieving a 32 cm reach with accuracy of 1.1 cm and 27.4 cm respectively. The quasi-static controller achieved better accuracy, but required a significantly longer period of time to reach its final position (nearly 10 s vs 3 s). Also, the quasi-static controller results in a motion that stops and starts at each position.

To achieve the final controller, we first had to tune the parameters of the controller. Tuning was performed at a static position with the goal of improving accuracy while limiting oscillations. As seen in Fig. 18, while tuning the controller, oscillations arose due to the proportional gain. Adding derivative gain to the controller was unable to eliminate the oscillations because of the 13 Hz frequency for changing neural excitations as well as the muscle activation dynamics. Instead, 120 N-s/m of damping was added to the arm support in each direction. This was effective in limiting oscillations during tuning.

We used the tuned controller to move the wrist over single point reaches of different distances. The average accuracy and rise times of each distance are shown in Table IV.

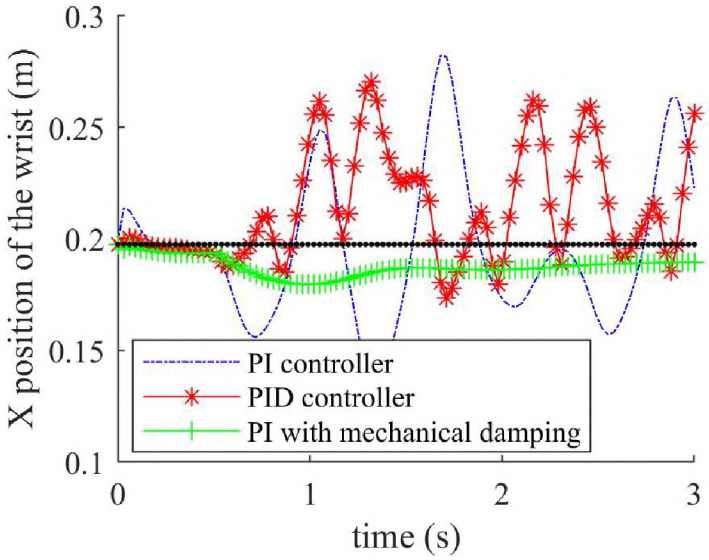



Figure 18: This figure compares the effect of using a PID controller (red) vs a PI controller with mechanical damping (green). Both controllers use a damping constant of 120 N-s/m. The controller used had a proportional gain of 250 N/m and an integral gain of 80 N/m-s and the goal position was the starting position (black). Adding derivative control is unable to eliminate oscillations compared to the undamped system (blue) because of the time delays in switching neural excitation inputs and the muscle activation dynamics. The addition of physical damping to the arm support eliminates the oscillations.

Table IV: Average results for point to point reaches

| Reach distance (cm) | Accuracy (cm) | 80% rise time (s) |
|---|---|---|
| 2 | 2.2 | 1.2 |
| 4 | 3.5 | 1.0 |
| 6 | 3.5 | 1.2 |
| 8 | 4.1 | 1.4 |
| 10 | 6.6 | 1.3 |
| 12 | 3.3 | 1.3 |
| 14 | 5.5 | 1.5 |
| 16 | 10.3 | 1.5 |

Since the rise times were similar for all reach distances, we used the overall average value of 1.3 s as the position switching time, $t_{switch}$ for our quasi-static controller. To select a distance between points, the goal was to select the largest distance possible while maintaining accuracy. With that goal in mind, we selected 6 cm spacing for our paths. (12 cm spacing also produced low error with the point to point moves, however a few test trials of paths with 12 cm spacing demonstrated that this accuracy was not maintained over several steps.)

The final parameters selected for the controller were a distance between positions in the path of $d_* = 6$ cm, the switching time of $t_{switch} = 1.3$ s, and the mechanical damping of 120 N-s/m in all directions.

### 5.3.3 Discussion

We have presented a quasi-static control architecture which is capable of accurately controlling the position of the wrist of a simulated human arm during reaching movements. Using a model-based, quasi-static method proved more accurate than a simple model-based PI controller. The accuracy of the final controller is worse than the accuracy found in [11] for holding static positions but similar. While this is a simulation and so does not have the issues of a practical experiment, maintaining a similar accuracy over the course of an entire reach is encouraging.

This paper determined the key parameters necessary which can be used as a starting point to achieve reaching motions with a practically implemented quasi-static controller. Further tuning of the parameters will be necessary when used with a human subject. A relatively simple improvement prescribed by the results was the addition of a damper to the passive arm support. Oscillation within a reaching motion can be discomforting to an individual with a paralyzed limb controlled with FES. The inability to eliminate oscillations using derivative control due to the delays in the system limited the ability to improve the performance of the controller. More complex methods have been developed to compensate

for the delays inherent in an electrically stimulated neuromuscular system, but these are generally developed only for single joints or often require some knowledge (with some uncertainty allowed) of the parameters of the system [13][14]. Our modeling method avoids parametric modeling due to the difficulty in defining the parameters and therefore does not fit these compensation methods well. The physical addition of the damper to the system is a simple way to improve the system performance and achieve similar results.

The average speed of the reaching motions prescribed by this controller (4.6 cm/s) is very slow compared to other controllers. A planar arm controlled by FES has been driven with maximum velocities of 25 m/s [9]. Even the simple controller without intermediate points much more quickly achieved its final position. Also, the simple controller produced smoother motions without stopping and starting. However, for many every day motions, slow arm movements are acceptable as accuracy is more important (for example, when eating off a plate). However, it is still necessary to improve the speed and smoothness of the movements to make the reaching similar to pre-injury abilities. Since the rise times for all single point-to-point trials were similar, the best way to improve the speed of the controller is to increase the distance between positions in the quasi-static path. However, this change must be done carefully because the increase in distance between points in this paper generally led to a decrease in accuracy.

There were many trials with larger distances that still had very good accuracy (and some small distance trials with bad accuracy). These good accuracy trials often occurred when the path was well within the feasible region. For many of the trials with bad accuracy, points along the path were near the border of the feasible region. This can be seen in Fig. 16 where the controller is accurate until the portion of the path near the boundary of the feasible space where errors in the model are not well compensated for. Model errors on the border of the feasible space could mean the point is actually infeasible or could cause the arm to move to infeasible configuration which cannot be recovered from. Additionally, the ability of the controller to compensate for errors is lower in this region as the muscles are able to achieve

less compensatory forces. Also, because the feasible space is not convex, large distances between points can lead to straight line paths which cross through infeasible regions. This is most likely one reason that the smaller distances between points have better accuracy. The current method of selecting a path is a simple nearest neighbor search. A more intelligent path selection method which selected feasible points away from the boundary could allow for larger distances between points and, in turn, higher speed of movement.

Overall, this work presents a quasi-static control architecture capable of achieving FES-driven reaching motions.

**Acknowledgments**

This work was supported by NIH NINDS grant N01-NS-5-2365, Veterans' Affairs grant B2359-C, NSF grant 1751821, the Cleveland State University Faculty Research and Development Program, and the Cleveland State Graduate Student Research Award. The authors would like to thank Dr. Ton van den Bogert at Cleveland State University for continuous support and advice in working with the Dynamic Arm Simulator.

## 5.4 REFERENCES

[1] Hamid-Reza Kobravi and Abbas Erfanian. Decentralized adaptive robust control based on sliding mode and nonlinear compensator for the control of ankle movement using functional electrical stimulation of agonist–antagonist muscles. *Journal of Neural Engineering*, 6(4):046007, 2009. ISSN 1741-2560. doi: 10.1088/1741-2560/6/4/046007.

[2] P. Hunter Peckham and Jayme S. Knutson. Functional Electrical Stimulation for Neuromuscular Applications. *Annual Review of Biomedical Engineering*, 7(1):327–360, 2005. ISSN 1523-9829. doi: 10.1146/annurev.bioeng.6.040803.140103.

[3] A. Bolu Ajiboye, Francis R. Willett, Daniel R. Young, William D. Memberg, Brian A.

Murphy, Jonathan P. Miller, Benjamin L. Walter, Jennifer A. Sweet, Harry A. Hoyen, Michael W. Keith, P. Hunter Peckham, John D. Simeral, John P. Donoghue, Leigh R. Hochberg, and Robert F. Kirsch. Restoration of reaching and grasping movements through brain-controlled muscle stimulation in a person with tetraplegia: a proof-of-concept demonstration. *The Lancet*, 389(10081):1821–1830, May 2017. ISSN 01406736. doi: 10.1016/S0140-6736(17)30601-3.

[4] Kathleen M. Jagodnik and Antonie J. van den Bogert. Optimization and evaluation of a proportional derivative controller for planar arm movement. *Journal of Biomechanics*, 43(6):1086–1091, Apr 2010. ISSN 00219290. doi: 10.1016/j.jbiomech.2009.12.017.

[5] Dimitra Blana, Robert F. Rf Kirsch, and Edward K. Ek Chadwick. Combined feedforward and feedback control of a redundant, nonlinear, dynamic musculoskeletal system. *Medical and Biological Engineering and Computing*, 47(5 SPEC. ISS.):533–542, 2009. ISSN 01400118. doi: 10.1007/s11517-009-0479-3.Combined.

[6] Kathleen M. Jagodnik, Philip S. Thomas, Antonie J. Van Den Bogert, Michael S. Branicky, and Robert F. Kirsch. Training an Actor-Critic Reinforcement Learning Controller for Arm Movement Using Human-Generated Rewards. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 25(10):1892–1905, Oct 2017. ISSN 15344320. doi: 10.1109/TNSRE.2017.2700395.

[7] L. Lan, K. Y. Zhu, and C. Y. Wen. Design of a Threshold FES Control System for Arm Movement. *Journal of Mechanics in Medicine and Biology*, 09(04):449–479, Dec 2009. ISSN 0219-5194. doi: 10.1142/S0219519409003127.

[8] C T Freeman. Upper Limb Electrical Stimulation Using Input-Output Linearization and Iterative Learning Control. *Control Systems Technology, IEEE Transactions on*, PP(99):1, 2014. ISSN 1063-6536. doi: 10.1109/TCST.2014.2363412.

[9] Reza Sharif Razavian, Borna Ghannadi, Naser Mehrabi, Mark Charlet, and John McPhee. Feedback Control of Functional Electrical Stimulation for 2-D Arm Reaching Movements. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 26(10):2033–2043, Oct 2018. ISSN 1534-4320. doi: 10.1109/TNSRE.2018.2853573.

[10] Derek N. Wolf and Eric M. Schearer. Evaluating an open-loop functional electrical stimulation controller for holding the shoulder and elbow configuration of a paralyzed arm. In *2017 International Conference on Rehabilitation Robotics (ICORR)*, pages 789–794. IEEE, Jul 2017. ISBN 978-1-5386-2296-4. doi: 10.1109/ICORR.2017.8009344.

[11] Derek N. Wolf and Eric M. Schearer. Holding Static Arm Configurations With Functional Electrical Stimulation: A Case Study. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 26(10):2044–2052, Oct 2018. ISSN 1534-4320. doi: 10.1109/TNSRE.2018.2866226.

[12] Eric M. Schearer, Yu-Wei Liao, Eric J. Perreault, Matthew C. Tresch, William D. Memberg, Robert F. Kirsch, and Kevin M. Lynch. Semiparametric Identification of Human Arm Dynamics for Flexible Control of a Functional Electrical Stimulation Neuroprosthesis. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 24(12):1405–1415, Dec 2016. ISSN 1534-4320. doi: 10.1109/TNSRE.2016.2535348.

[13] Naji Alibeji, Nicholas Kirsch, Shawn Farrokhi, and Nitin Sharma. Further Results on Predictor-Based Control of Neuromuscular Electrical Stimulation. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 23(6):1095–1105, 2015. ISSN 15344320. doi: 10.1109/TNSRE.2015.2418735.

[14] P. Cooman and R. F. Kirsch. Control of a time-delayed 5 degrees of freedom arm

model for use in upper extremity functional electrical stimulation. In *2012 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pages 322–324. IEEE, Aug 2012. ISBN 978-1-4577-1787-1. doi: 10.1109/EMBC.2012.6345934.

[15] Katharine H. Polasek, Harry A. Hoyen, Michael W. Keith, Robert F. Kirsch, and Dustin J. Tyler. Stimulation stability and selectivity of chronically implanted multicontact nerve cuff electrodes in the human upper extremity. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 17(5):428–437, Oct 2009. ISSN 15344320. doi: 10.1109/TNSRE.2009.2032603.

[16] Brian Smith, P Hunter Peckham, Michael W Keith, and Dennis D Roscoe. Stimulator for Versatile Control of Paralyzed Muscle. *IEEE Transactions on Biomedical Engineering*, BME-34(7):499–508, 1987. ISSN 0018-9294. doi: 10.1109/TBME.1987.325979.

[17] B. Smith, Zhengnian Tang, M.W. Johnson, S. Pourmehdi, M.M. Gazdik, J.R. Buckett, and P.H. Peckham. An externally powered, multichannel, implantable stimulator-telemeter for control of paralyzed muscle. *IEEE Transactions on Biomedical Engineering*, 45(4):463–475, Apr 1998. ISSN 00189294. doi: 10.1109/10.664202.

[18] Ronald L. Hart, Niloy Bhadra, Fred W. Montague, Kevin L. Kilgore, and P. Hunter Peckham. Design and testing of an advanced implantable neuroprosthesis with myoelectric control. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 19(1):45–53, 2011. ISSN 15344320. doi: 10.1109/TNSRE.2010.2079952.

[19] William D. Memberg, P. Hunter Peckham, and Michael W. Keith. A Surgically-Implanted Intramuscular Electrode for an Implantable Neuromuscular Stimulation System. *IEEE Transactions on Rehabilitation Engineering*, 2(2):80–91, 1994. ISSN 10636528. doi: 10.1109/86.313149.

[20] Gregory G. Naples, J. Thomas Mortimer, Avram Scheiner, and James D. Sweeney. A Spiral Nerve Cuff Electrode for Peripheral Nerve Stimulation. *IEEE Transactions on Biomedical Engineering*, 35(11):905–916, 1988. ISSN 15582531. doi: 10.1109/10.8670.

[21] Reza Sharif Razavian, Borna Ghannadi, Naser Mehrabi, Mark Charlet, and John McPhee. Feedback Control of Functional Electrical Stimulation for 2D Arm Reaching Movements. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, PP(c):1, 2018. ISSN 15344320. doi: 10.1109/TNSRE.2018.2853573.

[22] Edward K. Chadwick, Dimitra Blana, Robert F. Kirsch, and Antonie J. Van Den Bogert. Real-Time Simulation of Three-Dimensional Shoulder Girdle and Arm Dynamics. *IEEE Transactions on Biomedical Engineering*, 61(7):1–10, 2014. ISSN 15582531. doi: 10.1109/TBME.2014.2309727.

[23] Scott L. Delp, Frank C. Anderson, Allison S. Arnold, Peter Loan, Ayman Habib, Chand T. John, Eran Guendelman, and Darryl G. Thelen. OpenSim: Open-source software to create and analyze dynamic simulations of movement. *IEEE Transactions on Biomedical Engineering*, 54(11):1940–1950, 2007. ISSN 00189294. doi: 10.1109/TBME.2007.901024.

[24] Ajay Seth, Jennifer L. Hicks, Thomas K. Uchida, Ayman Habib, Christopher L. Dembia, James J. Dunne, Carmichael F. Ong, Matthew S. DeMers, Apoorva Rajagopal, Matthew Millard, Samuel R. Hamner, Edith M. Arnold, Jennifer R. Yong, Shrinidhi K. Lakshmikanth, Michael A. Sherman, Joy P. Ku, and Scott L. Delp. OpenSim: Simulating musculoskeletal dynamics and neuromuscular control to study human and animal movement. *PLOS Computational Biology*, 14(7):e1006223, Jul 2018. ISSN 1553-7358. doi: 10.1371/journal.pcbi.1006223.

[25] Ge Wu, Frans C.T. Van Der Helm, H. E.J. Veeger, Mohsen Makhsous, Peter Van Roy,

Carolyn Anglin, Jochem Nagels, Andrew R. Karduna, Kevin McQuade, Xuguang Wang, Frederick W. Werner, and Bryan Buchholz. ISB recommendation on definitions of joint coordinate systems of various joints for the reporting of human joint motion - Part II: Shoulder, elbow, wrist and hand. *Journal of Biomechanics*, 38(5): 981–992, 2005. ISSN 00219290. doi: 10.1016/j.jbiomech.2004.05.042.

[26] CE Rasmussen and CKI Williams. *Gaussian processes for machine learning*. 2006.

# CHAPTER VI

# MODEL LEARNING FOR CONTROL OF A PARALYZED HUMAN ARM WITH FUNCTIONAL ELECTRICAL STIMULATION

*Our overall aim is to restore reaching motions to individuals with paralyzed limbs due to spinal cord injuries. In previous chapters, I have presented our modelling method and demonstrated the ability to use the model as the basis of a controller for holding static wrist positions. With the improvements developed in Chapter V, we used this controller to move the wrist along straight line reaching paths. This was the first demonstration of a 3-dimensional, full-arm reaching motions driven by FES. The controller demonstrated success in these motions, but it struggled to reach all targets. These results drove the development of our final controller in Chapter VII.*

**Conference publication:**

1. D. N. Wolf, Z. A. Hall and E. M. Schearer. Model Learning for Control of a Paralyzed Human Arm with Functional Electrical Stimulation. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 10148-10154, 2020.

**ABSTRACT**

*Functional electrical stimulation (FES) is a promising technique for restoring reaching ability to individuals with tetraplegia. To this point, the complexities of goal-directed reaching motions and the shoulder-arm complex have prevented the realization of this potential in full-arm 3D reaching tasks. We trained a Gaussian process regression model to form the basis of a feedforward-feedback control structure capable of achieving reaching motions with a paralyzed upper limb. Over a series of 95 reaches of at least 10 cm in length, the controller achieved an average accuracy (measured by the Euclidean distance of the wrist to the final target position) of 3.8 cm and an average error along the path of 3.5 cm. This controller is the first demonstration of an accurate, complete-arm, FES-driven 3D reaching controller to be implemented with an individual with tetraplegia.*

## 6.1 Introduction

For the approximately 170,000 individuals with some level of tetraplegia due to spinal cord injuries [1], the restoration of hand and arm function is their greatest priority to improving their quality of life [2]. Functional electrical stimulation (FES) is a promising technique for helping these individuals complete the reaching motions necessary for daily living.

Many approaches have been attempted for achieving arm function with FES by first reducing the complexity of the arm-control problem. For repetitive tasks such as grasping, the complexity of the system was reduced by using preprogrammed, repeated stimulation patterns [3]. The extension of the repeated stimulation pattern method to full-arm reaching [4] cannot achieve all daily reaching tasks because the ever changing, goal-directed nature of reaching motions would require an infeasible amount of predetermined stimulation patterns. Any everyday reaching controller must be able to automatically select the stimulation commands necessary to achieve any novel, feasible reach.

Another common approach to reducing the complexity of controlling reaching motions

is to control each joint independently. FES controllers have demonstrated success in controlling individual joints such as elbow extension [5]. Extending this success to controlling multiple joints separately, the MUNDUS program [6] used a lockable exoskeleton to lock all uncontrolled joints while a single joint was driven with FES. The current state-of-the-art FES-reaching system, the BrainGate2 clinical trials, controls each joint independently but simultaneously [7]. Based on user intent, which was read by an intracortical brain computer interface, the controller selected a position along a predefined stimulation pattern for each of the joints. The controller had difficulties with multi-joint motions because the independent joint control could not account for joint interactions. The independent joint control method also does not allow for using the kinematic redundancy of the arm to complete tasks in different ways. In order to successfully control reaching motions, it is necessary to treat the arm as a complete system and not as independent joints.

Various methods of controlling the complete arm have been attempted. In computer simulations, optimized proportional-derivative control [8], feedforward-feedback control [9], reinforcement learning [10], and threshold control [11] have all successfully controlled reaching motions. Practically implementing these methods is difficult due to the real-world arm dynamics differing from the simulation.

Model learning, which uses data-driven machine learning models rather than parameterized physics-based models to predict the behavior of physical systems, has been used extensively to control robots (see [12] for a review) and is especially suitable for using FES to control the human arm. We intend to control multiple joints with multiple muscles – a problem that grows significantly in complexity as more joints and muscles are added. Using a physics-based model for FES control as in [13] may be effective for single joint systems with one or two muscles. However, as system complexity increases, the number of parameters needed to accurately model the arm increases. Further, guaranteeing parameter identifiability (e.g. of joint inertias) is extremely difficult given the limitations on range of motion and acceptable movement speeds for people with spinal cord injuries. Although

model learning (with artificial neural networks) has been used for FES control of planar reaching in healthy persons [14], it has not been demonstrated for 3D motions in people with spinal cord injuries.

In our own prior research, we have used model learning approaches to complete steps towards full-arm reaching. We have used semiparametric Gaussian process regression (GPR) to predict joint torques produced by muscles [15]. We built on this success by using nonparametric GPR models of the arm to form the basis of a feedforward-feedback controller to hold static wrist positions [16]. With this controller, we attempted quasi-static reaches with no intermediate points (used the model of the final position as the model for the entire reach) with some success, but there was significant oscillation and large overshoot in the reaching error [17]. In simulation, we showed that adding external damping and quasi-static intermediate points improved the controller performance [18].

The purpose of the current study is to build upon our previous work and develop a control structure capable of achieving full-arm, 3D reaching motions driven by FES. This is an important step towards the use of FES in the home to restore the full-arm reaching motions critical to completing many activities of daily living. We present a method of developing a subject-specific model of an individual with tetraplegia's arm and its response to electrical stimulation. We use this model as the basis of a combined feedforward-feedback controller capable of automatically determining the stimulation commands necessary to achieve desired reaching motions within the subject's workspace.

We evaluated the performance of the controller for completing reaching tasks. In particular we quantified the accuracy of the controller for moving the wrist to a desired final position and determined if there was a difference in accuracy based on target location. These results will guide the future developments of FES-driven reaching controllers.

## 6.2 Methods

In this study, we used a model learning based control strategy to complete reaching motions with an individual with high tetraplegia and an implanted FES neuroprosthesis. During the experiment, we 1) developed a Gaussian process regression model for the force the muscles produce as a function of the wrist position, and 2) used the model as the basis of an FES controller to move the wrist along desired paths.

The experiment took place during a four-hour time block. Experimental set up and identifying the model of the arm required approximately 1.5 hours. The participant took a half-hour break for lunch. The remaining time of the session was used to attempt randomly selected reaching motions. The participant was allowed breaks whenever requested.

### 6.2.1 Experimental Setup

We completed the experiments with a single human participant who has high tetraplegia and lacks voluntary control of her right arm. The participant's abdomen is implanted with a stimulator-telemeter [19][20][21] that can deliver current to activate nine independent muscle groups: triceps, deltoids, latissimus dorsi, serratus anterior, biceps/brachialis, supra/infraspinatus, rhomboids, lower pectoralis, and upper pectoralis. Muscle stimulation is delivered via bi-phasic, charge balanced pulses delivered at 13 Hz. The amplitude of the pulses is constant for each muscle group. The activation of each muscle group is controlled by varying the pulse-width (referred to as the stimulation input) from 0-250 $\mu$s. The participant's wheelchair is equipped with a passive arm support that assists against the force of gravity to create a comfortable and achievable workspace. More details can be found in [22] (Subject 1) and [16]. Protocols used for this research were approved by the institutional review boards at Cleveland State University (IRB NO. 30213-SCH-HS) and MetroHealth Medical Center (IRB NO. 04-00014).

We gathered training data for using a HapticMaster (Moog FCS) robot with three degrees of freedom. The robot was used to record the 3D forces of its end-effector. An

Optotrak Certus Motion Capture System (Northern Digital, Inc.) was used to measure the position of the wrist for data gathering for modeling and feedback during reaching.

Our previous research (and that of others) has demonstrated that significant oscillations occur with feedback FES control due to the delays in the FES system (low frequency of stimulation and electrical-muscular activation delays). In simulation, we were able to improve the controller performance by adding physical damping to the arm support [18]. Due to this finding, we used the robot to create a damped environment (20 N-s/m in each direction) during the reaching experiments.

The control and data collection occurred at 52 Hz, but stimulation inputs were updated at 13 Hz.

### 6.2.2 Model Learning

Our model learning procedure was previously presented in [16]. We present a detailed summary here for completeness. The basis of our controller (shown in Fig. 19) was a model consisting of three parts: 1. arm statics (predicts the forces necessary to hold a desired wrist position), 2. muscle force production (the mapping from wrist position to the maximum forces produced at the wrist by each muscle), and 3. recruitment curves (the mapping from muscle group electrical stimulation to muscle activation). Our controller inverts each part of the model to determine the muscle group stimulation commands necessary to achieve a desired wrist position.

To gather the model training data, a robot held the participant's wrist at a series of static positions within her comfortably reachable workspace. The connection of the participant's wrist to the robot was via a ball-in-socket joint that does not transmit torque. The robot was equipped with a three-dimensional force sensor at its end-effector, and the force needed to hold the wrist stationary, $\mathbf{f_r} \in \mathbb{R}^3$, was recorded.

To determine the arm statics, the robot held the arm in a position with zero muscle

stimulation, and, thus, all muscle activations, $\boldsymbol{\alpha} \in \mathbb{R}^9$, were zero. Therefore,

$$\mathbf{f}_{\mathbf{r}_{\mathbf{static}}} = \mathbf{p}(\mathbf{x}) \tag{6.1}$$

where $\mathbf{p}(\mathbf{x}) \in \mathbb{R}^3$ are the forces necessary to hold the wrist in the static position, $\mathbf{x} \in \mathbb{R}^3$.

To determine the force production of the $j^{\text{th}}$ muscle group, the muscle group was stimulated at its maximum stimulation command so that $\boldsymbol{\alpha}(j) = 1$. The forces required for the robot to hold the wrist stationary, $\mathbf{f}_{\mathbf{r}_{\mathbf{stim}}\mathbf{j}}$, are then defined by the difference of the forces with zero stimulation (i.e. the required static forces) and the forces produced by the muscle group,

$$\mathbf{f}_{\mathbf{r}_{\mathbf{stim}}} = \mathbf{p}(\mathbf{x}) - \mathbf{M}(\mathbf{x})\boldsymbol{\alpha}, \tag{6.2}$$

where $\mathbf{M}(\mathbf{x}) \in \mathbb{R}^{3\times 9}$ is the linear mapping of muscle activation to forces at the wrist and $\mathbf{p}(\mathbf{x})$ are the forces when stimulating no muscles. Each column of $\mathbf{M}(\mathbf{x})$ represents the forces produced in each Cartesian direction by 100% activation of the corresponding muscle group. The $j^{\text{th}}$ column of $\mathbf{M}(\mathbf{x})$ is determined by subtracting $\mathbf{f}_{\mathbf{r}_{\mathbf{stim}}\mathbf{j}}$, the recorded force during stimulation of muscle group $j$, from the previously identified static forces, $\mathbf{f}_{\mathbf{r}_{\mathbf{static}}}$,

$$\mathbf{M}(\mathbf{x})_{\mathbf{j}} = \mathbf{f}_{\mathbf{r}_{\mathbf{static}}} - \mathbf{f}_{\mathbf{r}_{\mathbf{stim}}\mathbf{j}}. \tag{6.3}$$

At a previous experimental session, we gathered training data by measuring $\mathbf{f}_{\mathbf{r}_{\mathbf{static}}}$ and $\mathbf{f}_{\mathbf{r}_{\mathbf{stim}}}$ for all muscle groups at 27 wrist positions, $\mathbf{x}$, within the participant's workspace. The positions were selected to be spaced throughout the subject's workspace at 3 levels (low, medium, and high). The boundaries of the workspace were defined by the subject's physical comfort. We repeated the set of measurements three times with a random order of wrist positions and muscle group activations. The data was used to train a set of GPR models with the input being the wrist position and the output being the force in one direction measured by the robot. A model was trained for the force produced in each degree of freedom for each of the nine muscle groups as well as for the static arm (zero muscle activation) resulting

in 30 total models. The squared exponential covariance function with automatic relevance detection was used, and the optimal hyperparameters were identified [23]. Using the GPR models, we can calculate $\mathbf{f}_{\mathbf{r}_{\mathbf{static}}}$ and $\mathbf{f}_{\mathbf{r}_{\mathbf{stim}}}$ and thus determine $\mathbf{p}(\mathbf{x})$ and $\mathbf{M}(\mathbf{x})$, via (6.3), at any position in the subject's workspace. In the controller (Fig. 19), the GPR models form the basis of the "Inverse Arm Statics" and "Inverse Muscle Force."

To account for changes in the arm and muscles from the time of modeling, we gathered new training data from a single set of 27 wrist positions at the start of the reaching experiments session. We reused the previously found hyperparameters to train a new model with less data and thus in less time. This method allows for identifying an accurate model the day of the experiment which is critical for accurate control.

The recruitment curves, the mapping from stimulation input to muscle group activation, for each muscle group were identified using the deconvolved ramp method [24].

### 6.2.3 Controller

The controller (Fig. 19) is a modified version of the controller used in [16] to hold static wrist positions. We present the complete controller for thoroughness. The controller uses the model presented in section 6.2.2 along with a feedback controller to determine the forces and corresponding muscle group stimulations necessary to move the wrist along a straight line path to a desired final position.

The controller first calculates the open-loop forces in each Cartesian direction, $\mathbf{p}(\mathbf{x}_*)$, necessary to hold a desired wrist position, $\mathbf{x}_* \in \mathbb{R}^3$, by using the GPR models of the inverse arm statics. A positional proportional-integral feedback controller produces corrective forces to adjust the open-loop forces to get the desired forces, $\mathbf{f}_{\mathbf{des}}$. Next, the GPR models of muscle force production and (6.3) are used to identify the elements of the mapping from muscle group activations to wrist forces at the desired wrist position, $\mathbf{M}(\mathbf{x}_*)$. After determining the desired forces and the muscle-force mapping, $\mathbf{M}(\mathbf{x}_*)$, we calculate the muscle activations, $\boldsymbol{\alpha}$, that will produce the desired forces.

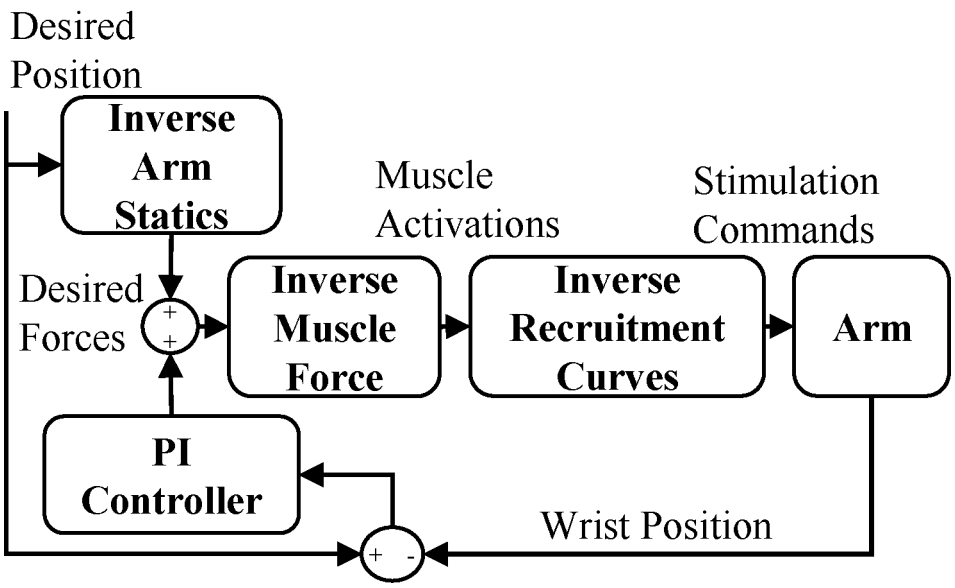



Figure 19: Controller block diagram: The controller automatically determines and applies the muscle stimulation commands to achieve a desired wrist position.

Determining the desired muscle activations during real-time feedback control requires overcoming two main problems at this point: 1) the arm is a redundant system in that there are more muscle groups than degrees of freedom (i.e. $\mathbf{M}(\mathbf{x}_*)$ is not square), and 2) the use of feedback means we have no control over the forces that the controller calls for and thus feedback overcompensation, the calling for forces above the greatest possible force, and muscle activation saturation can occur. Solving the system redundancy can traditionally be completed using a constrained optimization routine. However, in the case of feedback overcompensation, which can happen with little feedback compensation in an individual with tetraplegia due to muscle weakness from atrophy, constrained optimization routines are unable to find a feasible solution, one where the muscle forces are between zero and one, because one does not exist.

Our solution to these problems is to use the quasi-Newton method to find the set of

activations, $\boldsymbol{\alpha}$, that minimizes the penalty function,

$$
\begin{aligned}
&||\boldsymbol{\alpha}||_2^2 + c_1\|\mathbf{M}(\mathbf{x}_*)\boldsymbol{\alpha} - \mathbf{f_{des}}\|_{\mathbf{2}}^{\mathbf{2}} + \mathbf{c_2K} + \mathbf{c_3T} \\
&K = \sum k_i \text{ where } k_i = \begin{cases} \boldsymbol{\alpha}_i^2 \text{ if } \boldsymbol{\alpha}_i < 0 \\ (\boldsymbol{\alpha} - 1)^2 \text{ if } \boldsymbol{\alpha}_i > 1 \\ 0 \text{ if } 0 \leq \boldsymbol{\alpha}_i \leq 1 \end{cases}, \\
&T = \mathbf{f_{des}} \times \mathbf{M}(\mathbf{x}_*)\boldsymbol{\alpha}
\end{aligned}
\tag{6.4}
$$

where $||\boldsymbol{\alpha}||_2^2$ minimizes the muscle activations, $\|\mathbf{M}(\mathbf{x}_*)\boldsymbol{\alpha} - \mathbf{f_{des}}\|_{\mathbf{2}}^{\mathbf{2}}$ penalizes activations that do not produce the desired force, $K$ penalizes activations which do not belong to $\alpha_i \in [0, 1]$, and $T$ penalizes activations that produce forces in an incorrect direction. The penalty weights were chosen to be $c_1 = 100$, $c_2 = 10,000$ and $c_3 = 1,000$ because they produced feasible muscle activations with the forces in the right direction during offline testing.

To account for feedback overcompensation, we modified the controller in [16] with the aim of producing the largest possible force in the direction of the desired force. The addition of the $T$ term to the objective function penalizes forces not in the desired direction. When the forces become significantly larger than the maximum possible forces in a given wrist position, the penalty function solution breaks down and can lead to solutions which are infeasible (activations greater than one or less than zero) or to an inability to find a solution.

To improve these solutions, we developed a method to restrict the forces to within a rectangular prism of the maximum forces that can be produced. The maximum force that can be produced in each of the Cartesian directions (positive and negative directions) is determined and recorded offline. In the force space, a rectangular prism is drawn with faces at each of the maximum forces. If the desired force is greater than the maximum force that can be produced in any of the Cartesian directions, the intersection of the force vector and the rectangular prism is found, and this point becomes the new desired force.

This scales the desired force back to a position closer to the feasible force space while maintaining the desired direction. While this point is not guaranteed to be feasible, this scaling resulted in more reasonable activations as a solution to (6.4). The combination of this force scaling method with the $T$ term in (6.4) leads to selecting significantly improved activations over the controller in [16] when feedback overcompensation occurs.

Once the activations are found, the inverse recruitment curves block calculates the stimulation inputs.

### 6.2.4 Reaching Experiments and Data Analysis

To evaluate our controller's ability to control reaching motions, we quantified the accuracy of the controller over a series of reaches throughout the participant's workspace. Each reaching trial lasted for five seconds and consisted of a one-second hold at the starting position, a two-second ramp from the starting position to the target position, and a two-second hold at the target position. The straight-line ramp between positions was selected because previous quasi-static experiments showed that planning a defined path of closely spaced points between the target and goal would improve the performance (speed, smoothness, and accuracy of movement) [17][18], and a ramp is the limit of lowest spacing and time between each quasi-static point. We use the term quasi-static because the controller is based on a static model but is used to create reaching movement. An example of a desired trajectory can be seen in Fig. 20.

Prior to completing the reaching experiments, the proportional-integral controller was tuned using a series of 3 random reaching motions. The gains were manually tuned to improve the final accuracy without increasing oscillations. The values of the proportional and integral gains were selected to be 10 N/mm and 0.3 N/mm-s respectively.

During the reaching trials, the subject's wrist was connected to a robot which moved the subject's wrist to the starting position for each trial and created a damped environment during the reach. At the start of each trial, to limit the effects of the transient muscle dy-

namics and guarantee the controller starts at the correct point, the wrist was held stationary for the first 0.5 seconds. For the rest of the reach, the wrist was allowed to move as driven by the muscle stimulation.

To select the target reaching motions, we created a grid of wrist positions with 1 cm spacing within the convex hull of the 27 positions visited during the gathering of the model training data. On the day of the experiments, the subject's workspace spread 14 cm in the x direction, 23 cm in the y direction, and 11 cm in the z direction (see Fig. 20(a) for the coordinate frame). Start and target positions were randomly selected from this wrist position grid to create reaches of at least 10 cm in length. The average reach length was 13 cm. Once the start and target positions were selected, the complete desired reaching path was determined. For each wrist position along the path, $\mathbf{x}_*$, the open-loop muscle forces, $\mathbf{p}(\mathbf{x}_*)$, and the muscle force production matrix, $\mathbf{M}(\mathbf{x}_*)$ were determined offline before the trials. At each time step, the controller used $\mathbf{p}(\mathbf{x}_*)$ and $\mathbf{M}(\mathbf{x}_*)$ for the current desired position.

The final accuracy of the reach was determined by the Euclidean distance between the average final wrist position over the final 0.5 seconds and the desired target position. We also measured the accuracy over a complete reaching motion which we refer to as the path accuracy. The path accuracy for a single trial is defined as the average Euclidean distance from the wrist position to the desired target position over all time steps. We analyzed the effects of the position of the target and whether the selected path had a feasible target position on the controller performance. Feasible target positions are defined as positions where the model can select muscle activations capable of achieving the predicted open-loop forces. In an attempt to complete more possible reaches, non-feasible and feasible target positions were tested. We completed as many unique reaches as possible in the allotted time (95 total reaches). A 2-sample t-test was used to determine if these factors had an effect on the controller.

## 6.3 Results

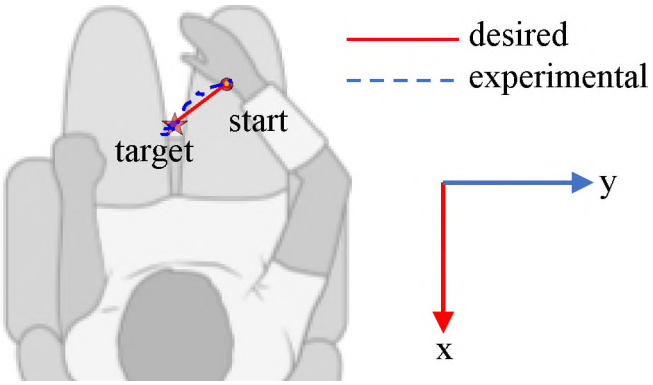
Over 95 trials, our controller achieved reaching motions with an average final accuracy of 3.8 cm (standard deviation of 2.2 cm) and an average path accuracy of 3.5 cm (standard deviation of 1.5 cm). Fig. 20 shows a representative reach with a final accuracy of 2.1 cm and a path accuracy of 1.9 cm. Fig. 3 shows the average accuracy at each point of the trajectory for all trials normalized by the length of each trial. As seen in both figures, the wrist position was generally able to track the desired reaching path and finish near the desired position. These examples and the overall accuracy results demonstrate that the controller successfully calculated the muscle stimulation commands required to achieve the desired reaching motions.

Fig. 20 also shows that though there were some oscillations, the amount was limited and generally low frequency which could be tolerated by the subject and would still be useful for functional tasks. This experimentally demonstrates the efficacy of using a proportional-integral controller to produce reaching motions within a damped environment.

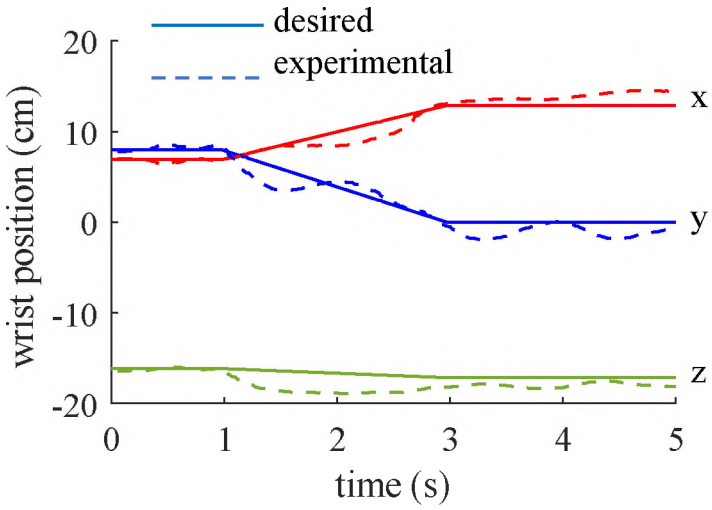
There was a significant difference ($p < 0.001$) in the final accuracy of reaching motions with feasible target positions ($\mu = 2.1$ cm, N=22) and the accuracy of reaches with infeasible target positions ($\mu = 4.3$ cm, N=73). There was also a significant difference ($p < 0.01$) in the final accuracy of reaching motions to the extreme right of the subject's workspace, defined by the target being greater than 5 cm to the right of the center of the subject's thorax ($\mu$ =4.9 cm, N=25) and the rest of the workspace ($\mu$ =3.4 cm, N=70). Fig. 22 shows this difference in accuracy based on target position.

## 6.4 Discussion

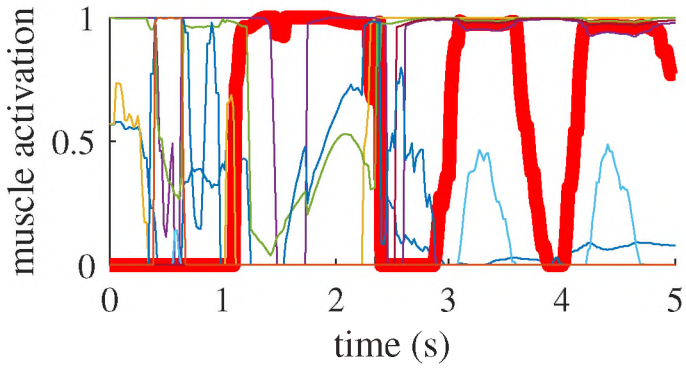
We have used model learning to develop a controller capable of achieving arbitrary reaching motions with an FES-controlled paralyzed arm. Our controller accurately moves the wrist along a desired path while accounting for the issues of a redundant system and feedback




(a) Overhead view of the reach in the x-y plane




(b) Time response of the reach in each direction




(c)

Figure 20: This figure shows the details of an example 10 cm reach with a final accuracy of 2.1 cm and path accuracy of 1.9 cm. (a) shows the overhead view in the x-y plane. (b) shows the time history of the reach in each Cartesian direction (dashed lines). As seen, the wrist was able to track the desired wrist position (solid lines). (c) shows the muscle group activations for the reach and demonstrates that the the controller is able to automatically select muscle activations to compensate for error. To move in the positive-y direction, the deltoids muscle group (highlighted) is activated. This is expected because the deltoids abducts the shoulder and moves the hand away from the subject's midline.

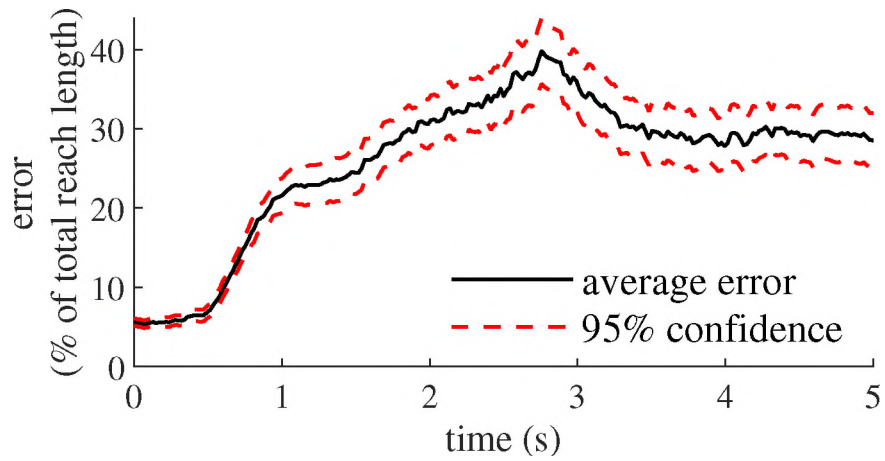



Figure 21: A plot of the average error and confidence intervals for all trials. The error for a single point is defined as the Euclidean distance from the current desired point in the trajectory and is displayed as a percentage of the total trajectory length. The path error for a given trial is the mean of the errors at each point of the trajectory.


Figure 22: This figure shows the target position and relative accuracy (represented by the size and color of each point) for all completed reaches. As seen, for targets to the right of the subject's workspace the accuracy is on average, worse than for the other target positions.

overcompensation/muscle activation saturation due to muscle weakness in individuals with spinal cord injuries. To our knowledge, this is the first demonstration of autonomously-selected electrical muscle stimulation for integrated shoulder and elbow control to produce 3D reaching movements in an individual with tetraplegia.

The controller accurately completed reaches throughout the participant's workspace. The final accuracy of 3.8 cm is sufficient to complete many daily reaching tasks such as grabbing a cup to drink. For points that our model predicted were feasible, the accuracy of 2.1 cm is, to our knowledge, the best reported 3D reaching accuracy achieved by FES. The improvement in accuracy for feasible points over the accuracy of infeasible points is promising moving forward as the predicted feasible points can be used to select reaching paths that will have better overall accuracy (i.e. only traveling through/to predicted feasible points). The good overall accuracy for all points is also important as it is difficult to choose paths that cross only feasible positions because of the limited workspace for individuals with tetraplegia.

The achieved path accuracy of 3.5 cm shows that the controller is able to accurately track a desired wrist path. This is important because when completing reaching tasks it is necessary to be able to reach a desired final hand position via differing paths. For example, when reaching out to pick up a fork off a table, the person may need to avoid bumping a cup of water with the hand. With the achieved path accuracy, our controller has the potential to achieve desired target positions while traveling along different paths.

Our accuracy was similar to the accuracy found in [14] of approximately 2 cm for planar reaching motions with healthy subjects as well as to our previous work with holding static wrist positions with an accuracy of 2.9 cm [16]. This is encouraging because maintaining a similar accuracy while expanding to 3D reaching motions, controlling more muscles, and working with an individual with tetraplegia is critical if our controller is to restore everyday reaching motions outside of a laboratory environment.

Reaches to the extreme right hand side of the subject's workspace were less accurate

than those towards the middle of the workspace. It was observed during trials to the right positions that the wrist would seem to reach a "sticking point" when trying to move out to the right. Reaching these targets often involved moving near the boundary of the workspace and sometimes a straight line to the target, as would be called for by the feedback controller, could pass through an infeasible or unreachable space. Therefore, more advanced trajectory selection that guides the wrist through only reachable points may improve the performance to these extreme targets.

One major issue with any model-based FES controller is the changing muscle dynamics due to fatigue and atrophy. It is difficult to ensure that the model remains accurate over time. The performance of our controller demonstrates that our new, faster modeling procedure of developing new training data while maintaining the hyperparameters from a previously trained model is a way to update the model to maintain accuracy. The complete modelling procedure takes approximately three hours to gather the data and train the models. The day-of experiment update only requires 35 minutes. This increase in model learning efficiency will allow the model to be updated more frequently and the controller to maintain it's performance over time. Additionally, the performance seen during this experiment validated our previously simulated result that a damped environment can improve the performance of an FES controller [18]. It is relatively simple to create a damped environment by adding physical damping to the arm support that individuals with tetraplegia often require to assist against the force of gravity and create a functional workspace.

The performance of this controller is a positive step to using an FES-controlled arm to restore everyday reaching tasks to individuals with high tetraplegia. To complete the goal of completing all possible reaching tasks, the accuracy must be improved throughout the workspace. This could be done through better path planning or through robotic assistance. Robotic exoskeletons have been shown to work cooperatively with FES to improve the accuracy of control for walking [25] and elbow flexion movements [26]. The robotic exoskeleton could be also used to replace the subject's arm support and produce the necessary

damped environment. Additionally, with an accurate low-level FES controller, a brain control interface (or other input device), such as that used in the BrainGate2 study [7], could be used to determine the desired reaching target. Our controller could then automatically complete the desired reach. The subject's intent during the BrainGate2 study was able to be decoded, but controlling each joint independently made the reaching motions difficult. Our controller could replace this low level independent joint control with a complete arm controller. Combined with these possible solutions, an accurate FES-reaching controller is a critical step to restoring the reaching ability to individuals with tetraplegia in the home.

**Acknowledgments**

This work was supported by NIH NINDS grant N01-NS-5-2365, Veteran's Affairs grant B2359-C, NSF grant 1751821, NSF grant 1659559, and the Cleveland State University Graduate Student Research Award.

## 6.5 REFERENCES

[1] NSCISC. Spinal Cord Injury Facts and Figures at a Glance. 2018.

[2] Kim D. Anderson. Targeting Recovery: Priorities of the Spinal Cord-Injured Population. *Journal of Neurotrauma*, 21(10):1371–1383, Oct 2004. ISSN 0897-7151. doi: 10.1089/neu.2004.21.1371.

[3] P. Hunter Peckham and Jayme S. Knutson. Functional Electrical Stimulation for Neuromuscular Applications. *Annual Review of Biomedical Engineering*, 7(1):327–360, 2005. ISSN 1523-9829. doi: 10.1146/annurev.bioeng.6.040803.140103.

[4] William D. Memberg, Katharine H. Polasek, Ronald L. Hart, Anne M. Bryden, Kevin L. Kilgore, Gregory A. Nemunaitis, Harry A. Hoyen, Michael W. Keith, and Robert F. Kirsch. Implanted neuroprosthesis for restoring arm and hand function in

people with high level tetraplegia. *Archives of Physical Medicine and Rehabilitation*, 95(6):1201–1211.e1, Jun 2014. ISSN 1532821X. doi: 10.1016/j.apmr.2014.01.028.

[5] William D. Memberg, Patrick E. Crago, and Michael W. Keith. Restoration of elbow extension via functional electrical stimulation in individuals with tetraplegia. *Journal of rehabilitation research and development*, 40(6):477–86, 2003. ISSN 0748-7711.

[6] Alessandra Pedrocchi, Simona Ferrante, Emilia Ambrosini, Marta Gandolla, Claudia Casellato, Thomas Schauer, Christian Klauer, Javier Pascual, Carmen Vidaurre, Margit Gföhler, Werner Reichenfelser, Jakob Karner, Silvestro Micera, Andrea Crema, Franco Molteni, Mauro Rossini, Giovanna Palumbo, Eleonora Guanziroli, Andreas Jedlitschka, Marco Hack, Maria Bulgheroni, Enrico D'Amico, Peter Schenk, Sven Zwicker, Alexander Duschau-Wicke, Justinas Miseikis, Lina Graber, and Giancarlo Ferrigno. MUNDUS project: MUltimodal Neuroprosthesis for daily Upper limb Support. *Journal of NeuroEngineering and Rehabilitation*, 10(1):66, 2013. ISSN 1743-0003. doi: 10.1186/1743-0003-10-66.

[7] A. Bolu Ajiboye, Francis R. Willett, Daniel R. Young, William D. Memberg, Brian A. Murphy, Jonathan P. Miller, Benjamin L. Walter, Jennifer A. Sweet, Harry A. Hoyen, Michael W. Keith, P. Hunter Peckham, John D. Simeral, John P. Donoghue, Leigh R. Hochberg, and Robert F. Kirsch. Restoration of reaching and grasping movements through brain-controlled muscle stimulation in a person with tetraplegia: a proof-of-concept demonstration. *The Lancet*, 389(10081):1821–1830, May 2017. ISSN 01406736. doi: 10.1016/S0140-6736(17)30601-3.

[8] Kathleen M. Jagodnik and Antonie J. van den Bogert. Optimization and evaluation of a proportional derivative controller for planar arm movement. *Journal of Biomechanics*, 43(6):1086–1091, Apr 2010. ISSN 00219290. doi: 10.1016/j.jbiomech.2009.12.017.

[9] Dimitra Blana, Robert F. Rf Kirsch, and Edward K. Ek Chadwick. Combined feed-forward and feedback control of a redundant, nonlinear, dynamic musculoskeletal system. *Medical and Biological Engineering and Computing*, 47(5 SPEC. ISS.):533–542, 2009. ISSN 01400118. doi: 10.1007/s11517-009-0479-3.Combined.

[10] Kathleen M. Jagodnik, Philip S. Thomas, Antonie J. Van Den Bogert, Michael S. Branicky, and Robert F. Kirsch. Training an Actor-Critic Reinforcement Learning Controller for Arm Movement Using Human-Generated Rewards. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 25(10):1892–1905, Oct 2017. ISSN 15344320. doi: 10.1109/TNSRE.2017.2700395.

[11] L. Lan, K. Y. Zhu, and C. Y. Wen. Design of a Threshold FES Control System for Arm Movement. *Journal of Mechanics in Medicine and Biology*, 09(04):449–479, Dec 2009. ISSN 0219-5194. doi: 10.1142/S0219519409003127.

[12] Duy Nguyen-Tuong and Jan Peters. Model learning for robot control: a survey. *Cognitive processing*, 12(4):319–340, 2011.

[13] Chris T. Freeman, Ann-Marie Hughes, Jane H. Burridge, Paul H. Chappell, Paul L. Lewin, and Eric Rogers. A Model of the Upper Extremity Using FES for Stroke Rehabilitation. *Journal of Biomechanical Engineering*, 131(3):031011, 2009. ISSN 01480731. doi: 10.1115/1.3005332.

[14] Reza Sharif Razavian, Borna Ghannadi, Naser Mehrabi, Mark Charlet, and John McPhee. Feedback Control of Functional Electrical Stimulation for 2-D Arm Reaching Movements. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 26(10):2033–2043, Oct 2018. ISSN 1534-4320. doi: 10.1109/TNSRE.2018.2853573.

[15] Eric M. Schearer, Yu-Wei Liao, Eric J. Perreault, Matthew C. Tresch, William D. Memberg, Robert F. Kirsch, and Kevin M. Lynch. Semiparametric Identification of

Human Arm Dynamics for Flexible Control of a Functional Electrical Stimulation Neuroprosthesis. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 24(12):1405–1415, Dec 2016. ISSN 1534-4320. doi: 10.1109/TNSRE.2016.2535348.

[16] Derek N. Wolf and Eric M. Schearer. Holding Static Arm Configurations With Functional Electrical Stimulation: A Case Study. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 26(10):2044–2052, Oct 2018. ISSN 1534-4320. doi: 10.1109/TNSRE.2018.2866226.

[17] Derek N Wolf and Eric M Schearer. Simple quasi-static control of functional electrical stimulation-driven reaching motions. In *2019 9th International IEEE/EMBS Conference on Neural Engineering (NER)*, pages 211–214. IEEE, 2019.

[18] Derek N Wolf and Eric M Schearer. Developing a quasi-static controller for a paralyzed human arm: A simulation study. In *2019 IEEE 16th International Conference on Rehabilitation Robotics (ICORR)*, pages 1153–1158. IEEE, 2019.

[19] Brian Smith, P Hunter Peckham, Michael W Keith, and Dennis D Roscoe. Stimulator for Versatile Control of Paralyzed Muscle. *IEEE Transactions on Biomedical Engineering*, BME-34(7):499–508, 1987. ISSN 0018-9294. doi: 10.1109/TBME.1987.325979.

[20] B. Smith, Zhengnian Tang, M.W. Johnson, S. Pourmehdi, M.M. Gazdik, J.R. Buckett, and P.H. Peckham. An externally powered, multichannel, implantable stimulator-telemeter for control of paralyzed muscle. *IEEE Transactions on Biomedical Engineering*, 45(4):463–475, Apr 1998. ISSN 00189294. doi: 10.1109/10.664202.

[21] Ronald L. Hart, Niloy Bhadra, Fred W. Montague, Kevin L. Kilgore, and P. Hunter Peckham. Design and testing of an advanced implantable neuroprosthesis with myo-

electric control. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 19(1):45–53, 2011. ISSN 15344320. doi: 10.1109/TNSRE.2010.2079952.

[22] Katharine H. Polasek, Harry A. Hoyen, Michael W. Keith, Robert F. Kirsch, and Dustin J. Tyler. Stimulation stability and selectivity of chronically implanted multicontact nerve cuff electrodes in the human upper extremity. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 17(5):428–437, Oct 2009. ISSN 15344320. doi: 10.1109/TNSRE.2009.2032603.

[23] CE Rasmussen and CKI Williams. *Gaussian processes for machine learning*. 2006.

[24] William K. Durfee and Karon E. MaClean. Methods for Estimating Isometric Recruitment Curves of Electrically Stimulated Muscle. *IEEE Transactions on Biomedical Engineering*, 36(7):654–667, 1989. ISSN 15582531. doi: 10.1109/10.32097.

[25] Kevin H. Ha, Spencer A. Murray, and Michael Goldfarb. An Approach for the Cooperative Control of FES with a Powered Exoskeleton during Level Walking for Persons with Paraplegia. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 24(4):455–466, 2016. ISSN 15344320. doi: 10.1109/TNSRE.2015.2421052.

[26] Derek Wolf, Nathan Dunkelberger, Craig G. McDonald, Kyra Rudy, Christopher Beck, Marcia K. O'Malley, and Eric Schearer. Combining functional electrical stimulation and a powered exoskeleton to control elbow flexion. In *2017 International Symposium on Wearable Robotics and Rehabilitation (WeRob)*, pages 1–2. IEEE, Nov 2017. ISBN 978-1-5386-4377-8. doi: 10.1109/WEROB.2017.8383860.

# CHAPTER VII

# TRAJECTORY OPTIMIZATION AND MODEL PREDICTIVE CONTROL FOR PRACTICAL IMPLEMENTATION OF FES-CONTROLLED REACHING MOTIONS IN INDIVIDUALS WITH SPINAL CORD INJURIES

*In this chapter we present a novel control scheme for achieving FES-driven 3D reaching motions in individuals with tetraplegia. We first complete a simulation study which demonstrates the importance of trajectory planning to account for the subject-specific muscle capabilities of an individual with a spinal cord injury. We develop a trajectory optimization method to find feasible reaching trajectories. We then use a model predictive control controller to drive the subject's arm along the desired trajectory. This control scheme is practically implemented in a subject with tetraplegia, and presents a major step towards achieving FES-driven full-arm reaching motions.*

**Publication:**

1. Model Predictive Control for Achieving Functional Electrical Stimulation-Controlled Reaching in an Individual with Tetraplegia. *in preparation*
2. Trajectory optimization for FES-Controlled Reaching Motions with Spinal Cord Injuries: A Simulation Study. *in preparation*

## 7.1 Introduction

Functional electrical stimulation (FES) neuroprostheses are a promising technology for restoring reaching functions to individuals with upper-limb paralysis caused by spinal cord injury (SCI). In the United States, there are nearly 175,000 individuals who suffer from some form of tetraplegia because of SCI alone [1], and for most of these individuals, regaining hand and arm function would most greatly improve their quality of life [2]. Unfortunately, to date, FES has yet to achieve reaching at a level necessary for every day use because of the complexity of the arm and the variability of every day reaching motions.

FES activates paralyzed muscles by applying electrical stimulation to the nerves or muscles. It has demonstrated success in restoring trunk control [3], some limited walking [4], and grasping [5] mainly through the use of repeated stimulation patterns or repeated/cyclic motions with learning and feedback. Additionally, many different techniques including reinforcement learning [6, 7], optimized PID control [8], and iterative learning control [9] have shown success in controlling a single, or even a couple degrees of freedom in the arm. Extending these successes to control the entire arm through reaching movements has proven difficult.

There have been successes in developing controllers for novel FES-driven reaching motions; however, due to the unique actuation issues seen in the arms of individuals with SCI, these techniques have found limited success in restoring reaching to the SCI population. Reaching controllers have been implemented in simulation [8], in a rehabilitation setting with subjects who have suffered a stroke [9], or with healthy subjects [10]. When working to restore reaching to individuals who have suffered paralysis due to an SCI, a unique set of characteristics arises due to the properties of their muscles. Individuals with SCI suffer from rapid muscle atrophy [11] which combines with increased muscle fatigue when electrically stimulated [12]. Additionally, some muscles suffer a complete loss of function even in the presence of stimulation because of lower motor neuron damage [13]. These subject-specific actuation issues are especially important and must be accounted when con-

trolling full-arm reaching motions because the muscles of the arm cross multiple degrees of freedom which must be actively controlled to accurately place the hand in space.

A common method of compensating for the issues in muscle actuation driven by FES is to use robots to support the desired motions and simplify the complexity of controlling multiple degrees of freedom [14]. In [15], the authors used a robotic exoskeleton to assist with elbow extension when the electrically stimulated triceps was fatigued. For full-arm reaching motions with FES, two seminal approaches are the MUNDUS [16] [17] and the BrainGate2 studies [18] which both also compensated for these muscular actuation difficulties by using robots. The MUNDUS project controlled each joint one at a time with FES while an exoskeleton locked the other joints in place. The BrainGate2 clinical study used input from an intracortical brain computer interface to select the desired muscle FES patterns for the elbow and wrist/hand, but it used a robotic arm support to control shoulder elevation. While both the MUNDUS and BrainGate2 studies demonstrated some success, each of these projects saw failures arise from the coupling of the robotic system with the FES-actuated arm. The MUNDUS project saw major hand position errors arise from slipping in the locked joints, mainly shoulder rotation. For the BrainGate2 system, the major errors in control arose due to the uncontrolled coupled motions produced in other degrees of freedom by the robotic arm support. To better control the arm through reaching motions, it is necessary to develop a controller that directly accounts for the actuation limitations in an individual with SCI as well as the dynamics of the system.

To our knowledge, there have been two main attempts to control full-arm reaching motions without robots actively controlling degrees of freedom, the Razavian controller [10] and our own previous work [19]. Both controllers used model-learning methods to determine configuration dependent models of forces produced by the muscles along with a feedback controller to move the wrist along a straight-line path to a desired wrist position. Razavian achieved 2D reaching motions using FES in a healthy individual. Our own previous work has achieved 3D reaching motions using straight line paths in a participant with

a spinal cord injury with reasonable accuracy, but there were areas of the workspace where the accuracy was limited [20].

Due to the limited muscular actuation of an individual with SCI, it is possible, and more likely probable, that straight line paths are not feasible for all possible reaches in the workspace. In [21], the authors found that there were many configurations in the workspace which are not controllable due to the muscle weakness and lack of activation associated with an individual with SCI. It is important to note that when we discuss controllability in this dissertation, we are not discussing the mathematical, control theory definition of controllability. Instead, we are using it to discuss the related idea that the controller is unable to move directly from the current state in the direction of the desired next state. When trying to follow a path that includes a configuration that is not controllable, the muscle activations quickly become saturated, the arm is unable to move closer to the target, and the accuracy plummets. Muscle activation saturation is a frequent occurrence in feedback control of a paralyzed arm due to the weakness and loss of function of the muscles. When using a simple feedback controller with no knowledge of the dynamics of the system, there is not a clear answer as to what muscle activation to apply when this saturation occurs and the arm can fail to move closer to the target.

In this chapter, we present a simulation study that demonstrates the issues that arise from the significant actuation weakness of the muscles of the arm of an individual with SCI. We hypothesize that planning paths using both the knowledge of the subject's muscles and a model of the dynamics of the system will result in more accurate reaches throughout the subject's workspace. We present two new control structures, a traditional feedback controller with a pre-planned feedforward component and a model predictive control (MPC) controller. These controllers attempt to better use the knowledge of the arm's actuation and dynamics to control 3D reaching motions in individuals with SCI.

After demonstrating the feasibility of using trajectory optimization to find feasible reaching motions and the validity of the control schemes in simulation, we practically im-

plement the MPC control scheme in an individual with high tetraplegia due to SCI. MPC control strategies have been used in FES control for lower limb movements [22, 23]. In this study, however, we present the first implementation of an MPC control scheme for FES-driven reaching movements in an individual with tetraplegia.

## 7.2 Trajectory optimization for FES-Controlled Reaching Motions with Spinal Cord Injuries: A Simulation Study

### 7.2.1 Methods

In this paper, we develop a control strategy for 3D FES-driven reaching motions that accounts for subject-specific muscle weakness and loss of function. An illustration of our control framework is seen in Fig. 23. We first identified a subject specific mathematical model of a subject with high tetraplegia due to spinal cord injury's arm and its response to electrical stimulation. Using this muscle capability model, we developed a dynamic simulation of the arm. We used the simulation to complete a trajectory optimization routine to find feasible trajectories that account for the dynamics and subject-specific muscle capabilities. Using these optimized trajectories, we then compared a feedback controller, feedforward-feedback controller, and model predictive control (MPC) controller to drive the arm along the desired trajectories and straight line paths.

#### Subject-specific muscle model

Details of our model identification procedures can be found in [21, 24]. We present a brief summary of the procedure and resulting model here.

To identify the muscle capabilities of a specific individual with high tetraplegia due to SCI, we completed a system identification experiment with a single human participant with high tetraplegia. The subject sustained a hemisection of the spinal cord at the C1-C2 level. We worked with her right arm which she is unable to voluntarily move except for limited shrugging of the shoulder. She exhibits normal to hypersensitive sensation on her

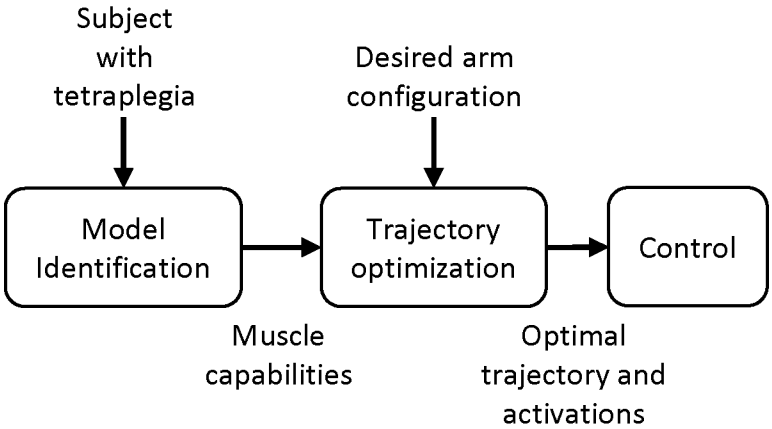



Figure 23: Framework for our control structure presented in this paper. We identify a subject-specific model of an arm and its response to electrical stimulation. We then use this model of the muscular capabilities of the subject to use a simulation of the arm to find optimal trajectories to achieve a desired arm configuration. Our controller then drives the arm along the desired trajectory to the target configuration.

right side and does exhibit hypertonia in some of her muscles. More details on the subject can be found in [25] (Subject 1). Protocols used for this research were approved by the institutional review boards at Cleveland State University (IRB NO. 30213-SCH-HS) and MetroHealth Medical Center (IRB NO. 04-00014).

The subject is implanted with the IST-12 stimulator telemeter in her abdomen [26–28]. The device uses intramuscular electrodes [29] and nerve cuff electrodes [30] to activate paralyzed muscles. Control signals are sent from the computer to the device via a radio frequency link. We controlled nine muscle groups with the device: 1. triceps, 2. deltoids, 3. latissimus dorsi, 4. serratus anterior, 5. biceps and brachialis, 6. supraspinatus and infraspinatus, 7. rhomboids, 8. lower pectoralis, and 9. upper pectoralis. Muscle stimulation uses bi-phasic, charge balanced pulses delivered at 13 Hz. The amplitude and maximum pulse-width of the muscles were determined as the point when no additional muscle force was achieved or when the participant reported discomfort. Safety limits were in place to assure the safety of the stimulation.

To gather data for model-learning, we used a HapticMaster (Moog FCS) robot with three degrees of freedom. The robot records the 3D forces at its end-effector. The sub-

ject's wrist was attached to the robot via the ADL gimbal attachment (Moog FCS) which transmits force but not torque to the robot. An Optotrak Certus Motion Capture System (Northern Digital, Inc.) captured data used to calculate the arm's configuration defined as three rotations at the shoulder - shoulder plane of elevation, shoulder elevation, and shoulder rotation - and two rotations at the elbow - flexion and pronation - as defined in [31]. At 27 positions spaced throughout the subject's workspace, we measured the amount of force produced by each muscle group when stimulated at their maximum pulse-width as well as with no muscle groups stimulated with the wrist held statically by the robot. When multiplied by the transpose of the Jacobian of the arm, the torques about each of four degrees of freedom - shoulder elevation plane, shoulder elevation, shoulder rotation, and elbow flexion (pronation does not create force at the wrist) - can be calculated. The process was repeated three times, and the data was used to train a semiparametric Gaussian process regression (GPR) model [32] for each muscle group. The input to the model is the configuration of the arm and the output is the joint torque predicted to be measured by the robot when a muscle group is stimulated. The difference between the predicted torque with no muscles stimulated and with a muscle group stimulated is the predicted amount of torque produced by the muscle group.

It is assumed that the torques produced by the muscle groups combine linearly, an assumption that is supported by [21, 33]. Therefore, the torque, $\boldsymbol{\tau} \in \mathbb{R}^4$ produced by a set of muscle activations, $\boldsymbol{\alpha} \in \mathbb{R}^9$ where $\boldsymbol{\alpha} \in [0, 1]$ for each muscle group, is determined by

$$\boldsymbol{\tau} = \boldsymbol{M}(\boldsymbol{q})\boldsymbol{\alpha} \tag{7.1}$$

where $\boldsymbol{M}(\boldsymbol{q}) \in \mathbb{R}^4 \times 9$ is the configuration dependent ($(\boldsymbol{q})$ muscle torque production matrix. The $i^{\text{th}}$ column of $\boldsymbol{M}$ corresponds to the torques produced by the $i^{\text{th}}$ muscle group when stimulated at 100% activation.

Using the semiparametric GPR model, the capabilities of the subject's muscles at any

arm configuration in the subject's workspace can be predicted by calculating the muscle torque production matrix.

### Dynamic arm simulation

For use in both the trajectory optimization and simulation of control of a paralyzed arm, we developed a dynamic simulation of the subject's arm that uses our previously described muscle torque production models to simulate the true capabilities of an individual with high tetraplegia due to SCI. The simulation consisted of two links, a humerus and a forearm, and had four degrees of freedom. There were three rotations at the shoulder corresponding to the shoulder plane of elevation, shoulder elevation, and shoulder rotation. There was an additional degree of freedom at the elbow corresponding to elbow flexion. All rotations are defined in [31]. Pronation was not included as a degree of freedom in the model as the muscle groups controlled in the subject do not include the groups which naturally control the pronation angle which made it nearly impossible to find trajectories which included controlling elbow pronation. Additionally, the subject's arm support greatly limits the freedom of the arm along that degree of freedom and so removing pronation from the system does not result in a loss of generality of our simulation. As the semiparametric GPR model requires pronation angle as an input, the system assumes a constant pronation angle of $163°$, the pronation angle at the equilibrium resting position of the subject.

The lengths of the segments were measured on our subject to be 0.315 m for the humerus and 0.253 m for the forearm. The mass, moments of inertia, and position of the center of masses for each link were estimated for our subject using the properties from [34]. The equations of motions and the derivatives of the equations of motion were found using Autolev [35]. The model was actuated using torques across each of the degrees of freedom. The control inputs to the model were the set of nine muscle activations, and (7.1) was used to determine the torque across each joint. These torques were then applied to the arm.

The simulation makes several assumptions regarding the dynamics of the real-life system. The simulation did not include gravity. The subject's shoulder muscles are not strong enough to support against the force of gravity. Due to this, the subject uses a mobile arm support to support against the force of gravity. We made the assumption that the arm support perfectly compensates for the force of gravity on the arm. The simulation also included passive stiffness of 1 Nm/rad and damping of 1 Nms/rad on each degree of freedom. The equilibrium configuration for the joint stiffness was the passive equilibrium configuration measured with the research subject. The purpose of these parameters is to produce a unique passive equilibrium point which is critical for the numerical simulation stability. The stiffness also adds some portion of the dynamics produced by the elasticity in the subject's mobile arm support. It is important to note, however, that the parameters and dynamics of the arm support were not explicitly measured or determined, and the arm support was not explicitly simulated by our model. When expanding the methods to practical implementation in a human subject, the dynamics of the arm support may need to be more explicitly modeled.

To simulate reaching motions driven by FES, the system was simulated using the backwards Euler method with a time step of 0.02 seconds. Newton's method was used to find the next state of the system at the end of the time step. For each time step, the control inputs were discretized and held constant across the entire time step which is realistic to how the real stimulation systems work where the frequency of stimulation determines the rate at which control inputs can change. The dynamics of the system and the muscle torque production models were modeled as continuous systems which varied with the state of the system, defined by the joint angles and joint velocities of the arm. This requires the derivatives of the equations of motions, which were obtained from Autolev, and the derivative of the muscle torque production matrix with respect to the state to be calculated. To calculate the derivative of the muscle torque production matrix, the derivative of the semiparametric GPR models were calculated as presented in [36].

The system was simulated using MATLAB.

### Trajectory optimization

Using the dynamic arm simulation presented in 7.2.1, we used trajectory optimization methods to determine feasible reaching trajectories that accounted for the subject-specific muscle capabilities of our research subject with high tetraplegia due to SCI. We created a grid of arm configurations with 20° spacing between the maximum and minimum joint angles measured in the training data in section 7.2.1. This resulted in a grid of 81 target configurations (see Fig. 24). The desired starting configuration for each reaching motion was defined as the resting equilibrium configuration as measured while identifying the model. This configuration placed the subject's wrist near the center of the reachable workspace.

For each target configuration, we attempted to find a feasible trajectory from the starting configuration to the target. To do so, we use the trajectory optimization techniques described for optimizing human gait in [37]. In this method, we use the direct collocation method which transforms the optimal control problem of calculating the optimal muscle activations to achieve the desired motion to a constrained nonlinear program. IPOPT [38] was used to solve this nonlinear program.

With a known nonlinear model of the dynamics of the system, $\boldsymbol{q}(k+1) = f(q(k), \boldsymbol{\alpha}(k))$ for each time-step, $k$, and for $n$ nodes, the trajectory optimization problem can be written as

$$\begin{aligned}
\underset{\boldsymbol{\alpha}, \boldsymbol{q}_{traj}}{\text{minimize:}} \quad & \text{mean}(\boldsymbol{\alpha}_{traj}^2) + \gamma \, \text{mean}((\boldsymbol{q}_{traj} - \boldsymbol{q}_{targ})^2) \\
\text{subject to:} \quad & \alpha_i(k) \in [0,1] \quad \forall i \in \{1,2,\ldots,9\}, \quad \forall k \in \{1,2,\ldots,n\} \\
& \boldsymbol{q}_{traj}(k+1) = f(\boldsymbol{q}_{traj}(k), \alpha(k)), \quad \forall k \in \{1,2,\ldots,n-1\} \\
& \boldsymbol{q}_{traj}(1) = \boldsymbol{q}_0 \\
& \boldsymbol{q}_{traj}(n) = \boldsymbol{q}_{targ} \\
& \boldsymbol{q}_{min} \leq \boldsymbol{q}_{traj}(k) \leq \boldsymbol{q}_{max}, \quad \forall k \in \{1,2,\ldots,n\}
\end{aligned} \quad . \tag{7.2}$$

The first term of the objective function minimizes the average of the squared muscle activations for all $n$ nodes of the trajectory, $\boldsymbol{\alpha}_{traj}$, and the second term attempts to minimize the distance from each configuration across all $n$ nodes of the trajectory, $\boldsymbol{q}_{traj}$ to the target position, $\boldsymbol{q}_{targ}$. Finding trajectories that minimize muscle activations is a desirable goal because it limits fatigue in the subject and allows for greater control bandwidth for a feedback controller to adjust activation before muscle saturation. The second term was added to produce more physiologically realistic reaches. With only minimizing muscle activations, the optimization found trajectories which involved swinging the arm the opposite direction of the target configuration and allowing the stiffness in the joints to swing the arm to the desired position. While straight line paths are not always reasonable, for most reaching tasks, a subject will want to reach in the most direct path possible to achieve the desired motion. $\gamma$ is a weighting factor which was selected to be $\gamma = 1$ rad$^{-2}$ to achieve the overall goal of the objective function to balance the goals of minimal activations and direct path reaches.

The optimization problem includes constraints on the state (joint angles and joint velocities), muscle activations, dynamics, and task constraints. To guarantee the controller found trajectories within the subject's comfortable workspace, the joint angles were constrained to be between the minimum and maximum joint angles seen during the system model identification in 7.2.1 with an additional $11°$ of rotation in each direction to allow for trajectories along the edge of the workspace. The range of motion was $89°$ to $143°$ for the shoulder plane of elevation, $21°$ to $77°$ for shoulder elevation, $-124°$ to $-78°$ for shoulder rotation, and $31°$ to $82°$ for elbow flexion. The joint velocities could have a maximum magnitude of 10 rad/s. The combined state constraints are represented by $\boldsymbol{q}_{min}$ and $\boldsymbol{q}_{max}$. The muscle activations were required to remain between 0 and 1. Lastly, the dynamics constraints ensured that the dynamics from the simulation developed in 7.2.1 were satisfied throughout the trajectory. This dynamics constraint ensures that the trajectories include the muscle capabilities specific to our subject with high tetraplegia. The dynamics

are estimated using the semi-implicit Euler method with a time-step of 0.025 s (80 nodes). The task constraints ensured that the first node began at the start configuration with zero velocity, $\boldsymbol{q}_0$, and the final node ended at the target configuration with zero velocity, $\boldsymbol{q}_{targ}$.

For each target position, we completed the trajectory optimization with 80 nodes. Increasing the number of nodes increases the computational load but improves the estimation of the system dynamics. To select a number of nodes, an optimization was completed with 200 nodes, and this trajectory was accepted as the ground truth. Starting with ten nodes, for a trajectory for which the optimization found a feasible solution, the rms error of the predicted trajectory to the ground truth trajectory was calculated. The number of nodes was increased until the trajectory was determined with acceptable error from the ground truth optimal trajectory.

The duration of each trajectory was 2 seconds. For the first attempt at finding a trajectory for a given target position, we used an initial guess of a straight line trajectory with zero activation. If IPOPT was unable to find an acceptable solution in 1500 iterations, we would try to find a feasible trajectory for the target position one additional time with a random initial guess. If a feasible trajectory was still not found, the target position was abandoned and the next target configuration was attempted. It is important to note that not finding a feasible trajectory does not guarantee that one does not exist as the optimization may converge to an infeasible local minimum. For all targets, the amount of time to complete the optimization ranged from 9 seconds to 1,486 seconds with an average time of 221 seconds. For targets which found a feasible trajectory, the average amount of time to complete the optimization routine was 113 seconds.

### Controlling simulated reaching motions

To test our trajectories found with the optimization methods in 7.2.1 and to prepare for practically implementing these methods in controlling FES-driven reaching motions in an individual with high tetraplegia, we compared three controllers: 1. a feedback controller, 2.

a combined feedforward-feedback controller (referred to as "FF+FB" in figures and tables), and 3. a model predictive control (MPC) controller.

The feedback controller in the first two controllers is similar to the controller presented in [20] used for straight line reaching and is used to provide a baseline comparison to the new controllers developed in this paper. The controller uses a PID controller to transform errors in joint-position and velocity to desired control torques, $\boldsymbol{\tau}_{des} \in \mathbb{R}^4$, across each degree of freedom. For the current configuration of the arm, $\boldsymbol{q}$, the muscle force production matrix, $\boldsymbol{M}(\boldsymbol{q})$, is predicted using the subject specific model developed in 7.2.1. For the feedback only controller, to solve for the desired muscle activations, $\boldsymbol{\alpha}$, we then solve the following quadratic programming problem using the `quadprog` function in MATLAB,

$$\begin{aligned} \underset{\boldsymbol{\alpha}}{\text{minimize:}} \quad & ||\boldsymbol{\alpha}||_2^2 \\ \text{subject to:} \quad & \mathrm{M(q)}\boldsymbol{\alpha} = \boldsymbol{\tau}_{\mathbf{des}} \\ & \alpha_i \in [0,1] \quad \forall i \in \{1,2,\ldots,9\} \end{aligned} \quad . \tag{7.3}$$

For the feedforward-feedback controller, we have to add the optimal activations found via the trajectory optimization to the commands produced by the feedback control, $\boldsymbol{\alpha}_{ff}$ and ensure that the total activation for each muscle is between 0 and 1. The new optimization problem becomes

$$\begin{aligned} \underset{\boldsymbol{\alpha}}{\text{minimize:}} \quad & ||\boldsymbol{\alpha}||_2^2 \\ \text{subject to:} \quad & \mathrm{M(q)}\boldsymbol{\alpha} = \boldsymbol{\tau}_{\mathbf{des}} \\ & \alpha_i + \alpha_{ff,i} \in [0,1] \quad \forall i \in \{1,2,\ldots,9\} \end{aligned} \quad . \tag{7.4}$$

If overcompensation occurs and the feedback controller calls for torques which are infeasible due to the muscle capabilities of our subject, the controller attempts to find the muscle activations which produce the maximum torque in the desired direction of $\boldsymbol{\tau}_{des}$. We achieve this by asking for 70% of the requested torque. If no feasible solution is found with the new requested torque, we continue to scale down the requested torqued to 70% of the pre-

vious requested torque for up to 10 iterations to attempt to find a set of muscle activations which produce torque in the desired direction. If after 10 iterations no solution is found, the controller outputs zero muscle activation.

The parameters of the PID controller were manually tuned on several trajectories with the goal of producing accurate reaches with smooth activation profiles. For the feedforward-feedback controller, the proportional gain was 4 N/mm, derivative gain was 1 N-s/mm, and integral gain was 1 N/mm-s.

We also developed an MPC control scheme with the hypothesis that including knowledge of the system dynamics more explicitly in the controller would produce more accurate reaches. Additionally, MPC controllers are able to explicitly account for the constraints of the system and thus eliminate the issue of overcompensation. The MPC control scheme we developed is based on the incremental MPC formulation presented in [39] which incorporates the benefits of integral control to the MPC control scheme. To develop the discrete state-space matrices, the autolev equations of motion were used to linearize the system about the current state. The MATLAB function `c2d` was used to create a discretized state-space system. The state of the system included the joint angles and joint velocities. The output of the system and the reference trajectory included only the joint angles.

For a given time-step, $k$, the discretized state-space model of the system can be written as

$$x(k+1) = Ax(k) + Bu(k) \tag{7.5}$$

$$y(k) = Cx(k) + Du(k) \tag{7.6}$$

, which can be used to predict the next state of the system, $x(k+1)$, and the current system output, $y(k)$. For the controller developed in this dissertation, it is assumed that $D = 0$. To add integral action to the controller, the state is augmented with the current control input, and the new control input is defined as the change in control input, $\Delta u$. The state-space

system becomes

$$\begin{bmatrix} x(k+1) \\ u(k) \end{bmatrix} = \begin{bmatrix} A & B \\ 0 & I \end{bmatrix} \begin{bmatrix} x(k) \\ u(k-1) \end{bmatrix} + \begin{bmatrix} B \\ I \end{bmatrix} \Delta u(k) \tag{7.7}$$

$$y(k) = [C\ D] \begin{bmatrix} x(k) \\ u(k-1) \end{bmatrix} + D\Delta u(k). \tag{7.8}$$

These state-space matrices are assumed constant for the control calculations. The controller aims to select the input commands which minimize the objective function

$$J = \sum_{i=1}^{n_y} e(k+i)^T e(k+i) + \lambda \sum_{i=0}^{n_u-1} \Delta u(k+i)^T \Delta u(k+i). \tag{7.9}$$

The first term of the equation minimizes the error, $e(k+i)$, for a given time-step which is defined as the estimated output as calculated by equations (7.8) subtracted from the reference trajectory. The prediction horizon, $n_y$, determines for how many time steps forward the model predicts states and system error. The control horizon, $n_u$, determines the number of time steps forward that the controller optimizes control inputs. For time steps $n_u < i < n_y$, $\Delta u = 0$. The lumped scalar weighting $\lambda$ acts as a muscle group activation smoothness parameter by weighting the amount that the activation commands change during a time-step.

The parameters of the MPC controller were tuned on several trajectories with the goal of producing accurate reaches along with smooth activation profiles. The time step of the simulation was 0.02 seconds. The prediction horizon was selected to be 4 time steps, and the control horizon was 2 time steps. This weighting on the change in muscle activations, $\lambda$, was selected to be 0.001 which was the highest value that did not see a large drop in accurately achieving desired arm configurations. We selected this highest value of the control input weighting to create smoother activation profiles which are more comfortable for subjects.

The feedback and MPC controllers were tested while trying to follow straight line trajectories (in joint space), and all three controllers were tested while trying to follow the optimized trajectories found in 7.2.1 to determine the effect path planning has on the accuracy of the system. The straight line paths were determined by fitting a fifth-order polynomial between the starting and target arm configurations. The controllers were first tested using the "perfect model" condition where the muscle capability model used by the controller and the dynamics simulation were identical. Additionally, each controller was also tested with an "uncertain model" condition to determine their robustness to uncertainty in the system. In this condition, the muscle force capability model of the dynamics simulation was different than the model used by the controller. The uncertain muscle models were created by developing a new set of training data for the models produced in 7.2.1. The training data were randomly pulled from the predicted distribution (mean and variance) calculated by the semiparametric GPR models. We repeated the control with uncertain models for all trajectories 10 times to a wide selection of uncertain muscle capability matrices. This uncertain model would produce changes in both the magnitude of the torque production as well as the torque-space direction of each muscle. Another realistic scenario is the presence of fatigue in the subject which would affect the magnitude of torque created but not the direction. To test the controller's response to fatigue, a "fatigued model" was created by limiting the muscle force production matrix to 90% of the predicted value.

### 7.2.2 Results

Planning optimal trajectories which account for the capabilities of the muscles of a subject with tetraplegia due to SCI produced improved accuracy and the ability to reach more positions throughout the subject's workspace. Table V presents the average accuracy, defined as the Euclidean distance from the final position of the simulation's hand (calculated by forward kinematics) to the target wrist position. While the controllers drove the arm in joint-space, for a subject, the most important measure for completing functional tasks is

| controller | trajectory | perfect model mean accuracy (std) cm | uncertain model mean accuracy (std) cm | fatigued model mean accuracy (std) cm |
|---|---|---|---|---|
| FF+FB | planned | 1.4 (1.6) | 8.5 (4.4) | 2.3 (2.5) |
| MPC | planned | 3.0 (2.2) | 9.8 (4.0) | 6.8 (2.9) |
| | straight | 7.3 (5.6) | 10.9 (4.9) | 7.8 (4.3) |
| Feedback | planned | 12.5 (9.7) | 15.5 (8.5) | 13.7 (10.6) |
| | straight | 16 (10.1) | 18.2 (7.9) | 16.2 (9.9) |

Table V: Average final wrist position error from the target position for each controller

the ability to place the subject's hand at a target; therefore, we use hand position as the measurement of success.

On average, the feedforward-feedback and MPC control schemes with planned trajectories performed with improved accuracy over the other three methods when using a perfect model. Additionally, these two methods demonstrated greater ability in reaching more positions throughout the workspace. Fig. 24 shows the positions that were achieved by each controller with better than 5 cm accuracy. Fig. 25(a) shows the number of targets that were reached with at least the benchmark level of accuracy for benchmark accuracies ranging from 2 to 20 cm. There is not a large improvement in the accuracy nor the number of points reached with straight line paths or planned paths with the feedback controller. These results demonstrate that both planning trajectories and incorporating knowledge of the capabilities of the subject's muscles more directly into the control scheme, either via feedforward activation commands or through a model in MPC control, are critical to producing accurate reaching motions in an individual with SCI. When adding uncertainty to the model, the average accuracy across all controllers drops significantly (see Table V). However, the feedforward-feedback and MPC control schemes are still able to achieve more target wrist positions than the control methods without planning or without feedforward activations for the Feedback controller (see Fig.25(b). With the fatigued muscle model, the accuracy is again worse than with the perfect model, but the feedforward-feedback and MPC control schemes still perform the best overall at achieving target positions.

Fig. 26 and Fig. 27 show an example of a trial with a perfect model for which planning

and including knowledge of the muscle capabilities produced a significant improvement accuracy and provides a good comparison of the different control strategies. For straight trajectories, the feedback controller produced an accuracy of 15.0 cm, and the MPC controller achieved an accuracy of 5.1 cm. For the planned trajectories, the feedback controller produced an accuracy of 7.6 cm, the feedforward-feedback controller achieved 0.1 cm of accuracy, and the MPC controller achieved an accuracy of 0.7 cm. At around t=1 s, the feedback controller in both the straight line and planned trajectories reached uncontrollable configurations and asked for torques in a direction which is not feasible. Due to this situation of overcompensation, the controller, unable to find activations to produce torque in the desired direction, requests zero muscle activation, and the arm does not move. The MPC controller, on the other hand, is able to avoid this situation by using its knowledge of the dynamics and muscle capabilities of the system to determine the best feasible torque-space direction to apply torque. For example, due to the muscle capability matrix, it may not be possible in a given configuration to apply positive torque to both the shoulder elevation and shoulder rotation degrees of freedom. Though not seen in this example, this issue of overcompensation and a feedback controller applying zero torque also occurs in the feedforward-feedback controller. With a perfect model, we would expect to see the feedforward-feedback controller achieve perfect tracking. However, for several trials that were not achieved with perfect accuracy (see Fig. 25(a)), the sampling time difference and difference in dynamics approximation methods (semi-implicit Euler method vs. backward Euler method) between the trajectory optimization and the control simulation produced small differences in motion while in an uncontrollable configuration, and the feedback was unable to compensate for it. The feedback controller in this situation will ask for zero torque. However, the MPC controller can predict if applying positive torque to one of the degrees of freedom will improve the accuracy of the controller along the desired trajectory. Errors which do develop in the MPC controller are due to the linearization of a nonlinear dynamical system.

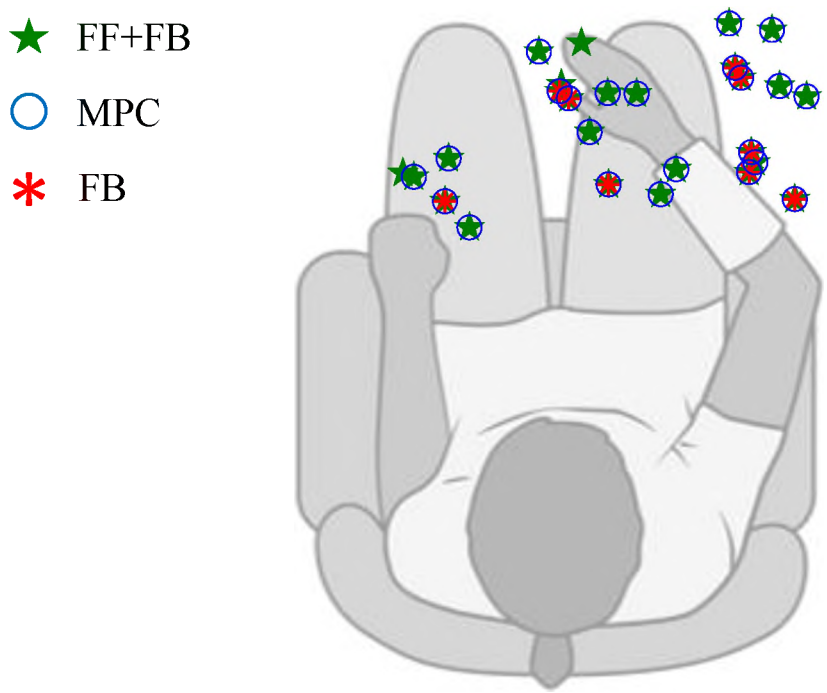
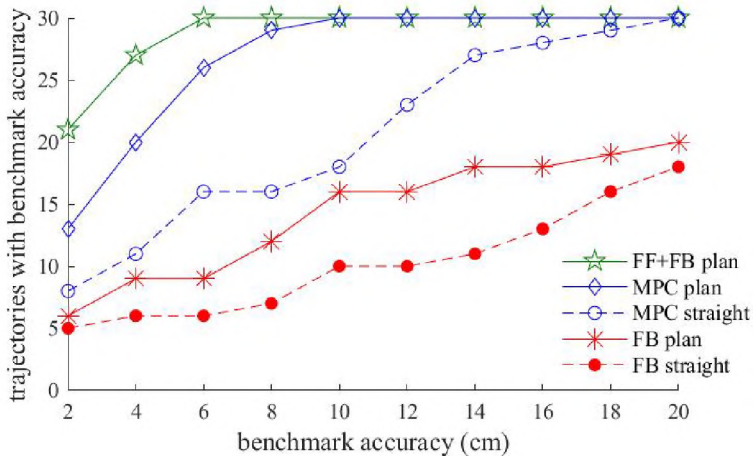
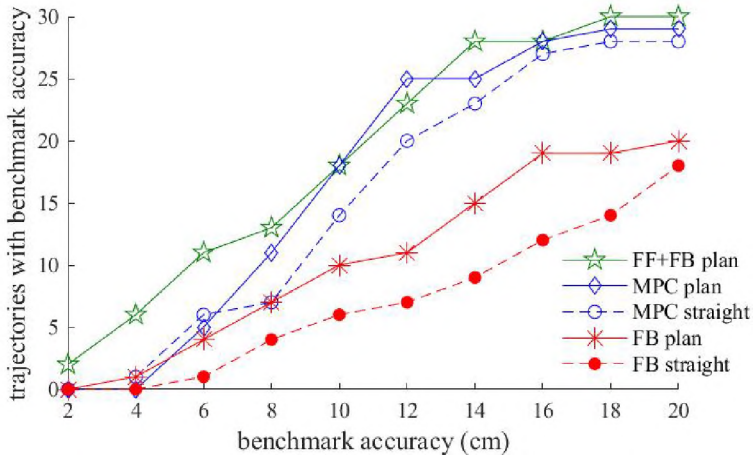


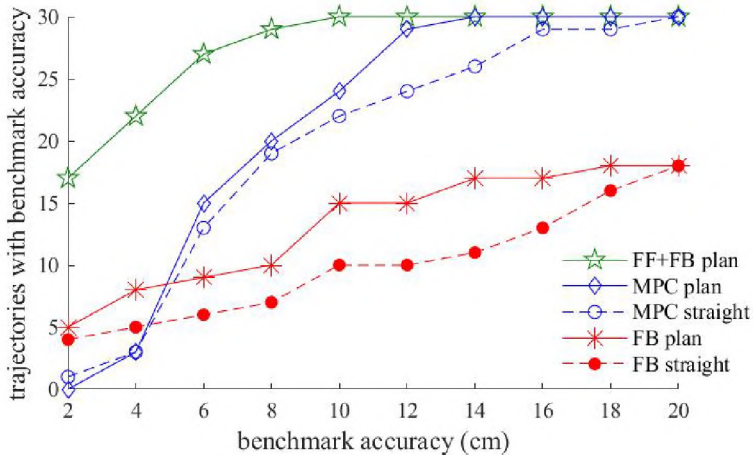

Figure 24: This figures shows the target hand positions of targets which are achieved with at least 5 cm accuracy using the planned trajectories and each of the three control strategies. The FF+FB controller achieves all 30 targets, the MPC controller achieves 24 targets, and the FB controller achieves nine target hand positions.




(a) perfect model




(b) uncertain model




(c) fatigued model

Figure 25: This plot shows the number of trajectories for each controller which achieved at least the benchmark accuracy on the horizontal axis when using a (a) perfect model, (b) uncertain model (average of 10 trials is shown), and (c) fatigued model.

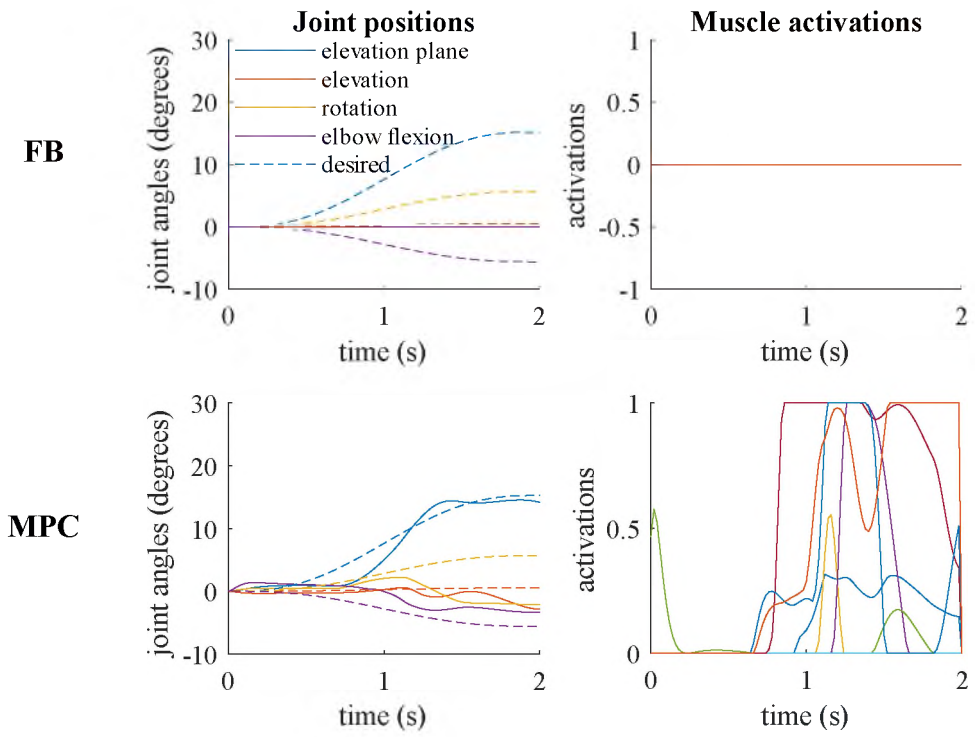



Figure 26: This figure shows the movement and activation patterns for controlling a simulated arm along a desired straight trajectory (dashed line) for all three control strategies. To better show the movement of all four joints, the joint angles relative to the starting configuration are plotted.

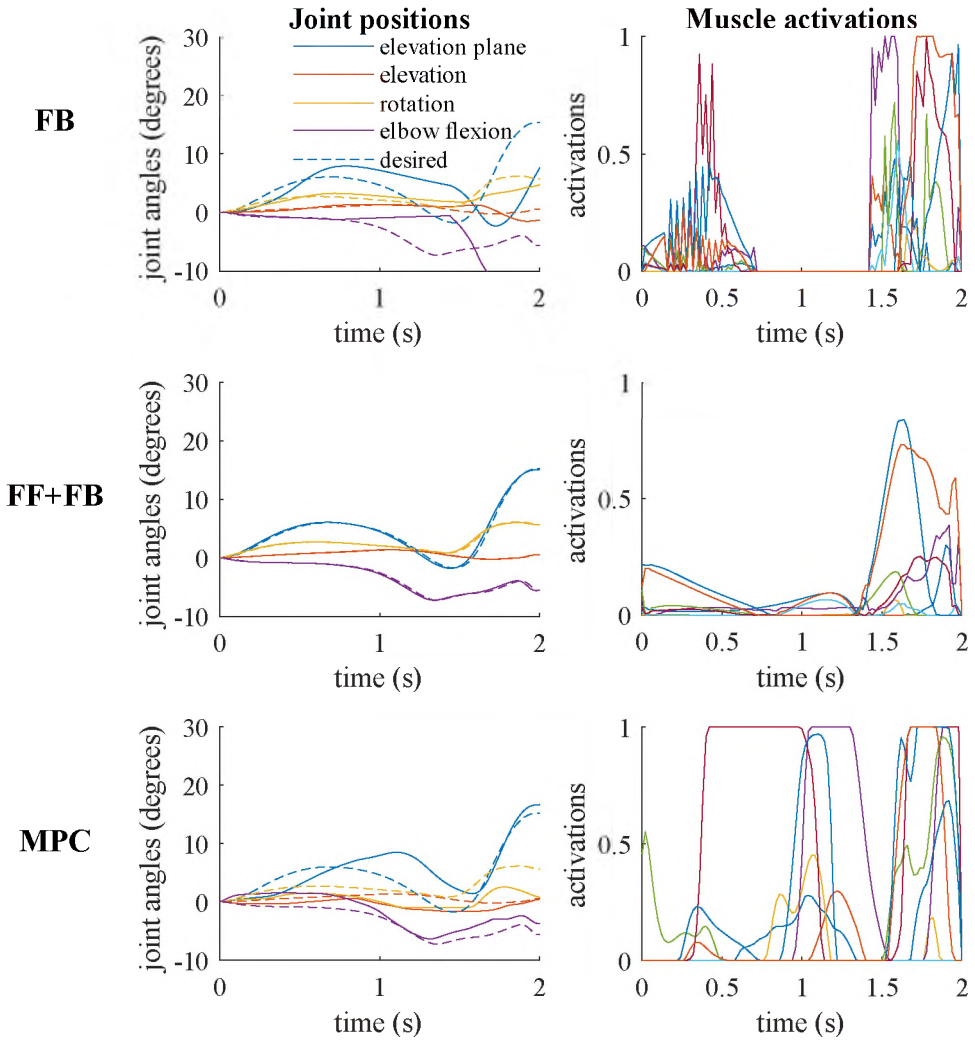



Figure 27: This figure shows the movement and activation patterns for controlling a simulated arm along a desired optimized trajectory (dashed line) for all three control strategies. To better show the movement of all four joints, the joint angles relative to the starting configuration are plotted.

### 7.2.3 Discussion

We have presented a framework for controlling an FES-driven human arm through desired reaching trajectories. We use trajectory optimization techniques to determine feasible trajectories which account for the subject-specific muscle capabilities as well as the dynamics of the arm. With the feasible trajectories, we demonstrate that using the knowledge of the arm's muscles and dynamics, either by using feedforward activations or in an MPC control strategy, produces improved accuracy and the ability to reach target positions throughout the workspace.

Reaching all portions of the subject's workspace was found to be difficult with straight line paths in [40], and we confirmed that result with the straight line trajectories in this study including demonstrating situations where simple PID feedback controllers will fail to produce activations. As had been observed in [21], due to the unique muscle capability characteristics of individuals with tetraplegia due to SCI, including muscle atrophy and lower motor neuron damage, the workspace of the subject will include configurations that are not controllable in that the muscles are unable to drive the arm in the direction of the next desired state. The need to account for these unique, subject-specific capabilities prevents straight line feedback controllers such as the one presented in [10] from being successfully implemented in individuals with SCI. Even in the presence of no uncertainty in the controller, if these configurations are not avoided and planned for, as seen in this paper, the reaching motion will not be successful. This point bears repeating, even with a perfect model of the subject's muscle capabilities and the dynamics of the system, trajectory optimization is necessary to avoid paths which include uncontrollable configurations and produce accurate reaches.

While we attempted to model some level of uncertainty, the dynamic simulation presented in this study is very basic and does not include many nonlinearities and sources of uncertainty which exist in individuals with SCI including electro-mechanical delays, muscle activation dynamics, rapid fatigue, and the nonlinear elasticity of the arm-support.

When practically implementing the control framework in an individual with SCI, the increased uncertainty makes it even more critical to avoid uncontrollable configurations. While the feedfoward-feedback controller produced the best accuracy in this study, there were still situations where the arm would get stuck in an uncontrollable configuration and zero muscle activation would be requested for all muscle groups. The use of an MPC controller offers a solution to this specific issue, but it did not perform as well overall in the presence of model errors. One improvement to the system could be to combine the feedforward command with an MPC feedback controller [41].

Other methods of trajectory optimization could also be used to better avoid uncontrollable locations to improve the performance of all controllers. Some of the found trajectories were on the edge of controllability and even small deviations would lead to large errors. This can be seen by the small sampling differences in activation leading to several trials with perfect feedforward control to end up not near the target. One possible solution to this would be to map the controllability of the configuration workspace. Our own previous work has attempted to do a form of mapping the configuration dependent capabilities of the workspace for rehabilitation purposes [42]. With a similar mapping, additional terms could be added to the trajectory optimization or a trajectory optimization algorithm such as CHOMP [43] could be used to bias the trajectories away from uncontrollable locations.

To our knowledge, the framework developed in this paper is the first attempt to search for and control optimized, feasible trajectories using a subject-specific model of the muscle capabilities for FES-driven reaching motions.

## 7.3 Model Predictive Control for Achieving Functional Electrical Stimulation-Controlled Reaching in an Individual with Tetraplegia

### 7.3.1 Methods

We develop a control framework to drive the arm of an individual with high tetraplegia due to SCI along desired reaching trajectories within her workspace (see Fig. 23). In

these experiments we largely follow the procedures developed in section 7.2.1 with some adjustment due to practical implementation differences. We first identify a subject-specific model of our subject's arm and its response to stimulation. We then use this model and our trajectory optimization scheme to find feasible reaching motions throughout the target workspace. Using an MPC control scheme, we then attempt to move the subject's arm along desired trajectories within the workspace.

The experimental session took place over a single 3.5 hour experiment. Initial set-up (attaching motion capture markers and setting up coordinate frames and finding comfortable arm configurations) took 40 minutes. Data gathering for the day of model identification and training the new model required 30 minutes. Finding feasible trajectories with trajectory optimization required an additional 20 minutes. We spent 15 minutes tuning the MPC controller, and the remainder of the time was used for completing reaches. The subject was allowed breaks whenever requested.

### Experimental setup

We worked with a single subject with high tetraplegia due to SCI. The subject, as previously described in section 7.2.1, sustained a hemisection of the spinal cord at the C1-C2 level. We worked with her right arm which is unable to voluntarily move her right arm except for limited shrugging of the shoulder. She exhibits normal to hypersensitive sensation on her right side and does exhibit hypertonia in some of her muscles. Her wheelchair is equipped with a passive arm support which uses elastic bands to assist against the force of gravity and maintain a comfortable arm configuration for achieving reaching motions. The arm support produces a resting equilibrium position at about the center of the subject's torso. More details on the subject can be found in [25] (Subject 1). Protocols used for this research were approved by the institutional review boards at Cleveland State University (IRB NO. 30213-SCH-HS) and MetroHealth Medical Center (IRB NO. 04-00014).

The subject is implanted with the IST-12 stimulator telemeter in her abdomen [26–28].

The device uses intramuscular electrodes [29] and nerve cuff electrodes [30] to activate paralyzed muscles. Control signals are sent from the computer to the device via a radio frequency link. We controlled nine muscle groups with the device: 1. triceps, 2. deltoids, 3. latissimus dorsi, 4. serratus anterior, 5. biceps and brachialis, 6. supraspinatus and infraspinatus, 7. rhomboids, 8. lower pectoralis, and 9. upper pectoralis. Muscle stimulation uses a bi-phasic, charge balanced pulse delivered at 13 Hz. The amplitude and maximum pulse-width of the muscles were determined as the point when no additional muscle force was achieved or when the participant reported discomfort. To change the amount of muscle activation achieved, we adjust the pulse-width of the stimulation. We refer to this as the stimulation input. Safety limits were in place to assure the safety of the stimulation.

We gathered training data for using a HapticMaster (Moog FCS) robot with three degrees of freedom. The robot was used to record the 3D forces of its end-effector. The subject's wrist was attached the robot via a ball and socket joint. The subject's muscles are often not strong enough to move the arm against the elasticity of the arm support. Therefore, the robot was used to provide a supporting force which countered the arm support and the force of gravity to allow the subject's arm to move more freely. The robot was also used to create a haptic bounding box around the edge of the workspace to ensure patient comfort and safety. An Optotrak Certus Motion Capture System (Northern Digital, Inc.) was used to measure the arm configuration for data gathering for modeling and feedback during reaching.

The control and data collection occurred at 52 Hz, but stimulation inputs were updated at the stimulation frequency of 13 Hz. The experiment was controlled using MATLAB xPC target on a Dell Dimension 8400 PC with a Pentium 4 3.20 GHz processor. Trajectory optimization was completed using MATLAB 2019b and IPOPT [38].

### Day of model identification

Our initial subject-specific, muscle capability model was developed on October 13, 2020 and is described in section 7.2.1. Our reaching experiments took place one month later, on November 12, 2020. To account for changes in the subject's muscle force capability due to atrophy/hypertrophy, changes in muscle tone, or other day-to-day fluctuation in muscle strength, we developed a "day of" model update.

To create the day of model, we followed the procedure described in section 7.2.1 to gather model training data. We first moved the subject's arm to 13 targets spread throughout the subject's comfortable 3D workspace. The first target was selected to be a central target near the equilibrium resting point produced by the arm support. For each target, we recorded the configuration of arm - defined as the shoulder elevation plane, shoulder elevation, shoulder rotation, elbow flexion - and the torque produced about each degree of freedom. Pronation torque was not measured as pronation does not affect the position of the subject's hand and so we did not aim to directly control it. Additionally, the subject's arm support limits the freedom of the pronation angle. Due to this, pronation was assumed constant in our models and our arm simulation for trajectory optimization. With the new training data, defined by the arm configuration and torque production pair, we developed new semiparametric Gaussian process regression models [32] by updating the training information but maintaining the hyperparameters from the previous model. This allowed us to update our model efficiently without the need to repeat each target multiple times nor the need to use significant computation time recomputing the hyperparameters.

Day of models allow us to achieve better accuracy by updating our model for changes in the subject's muscle capabilities due to day-to-day fatigue, atrophy, muscle-tone, etc. Additionally, there will always be some bias error in the measurement of the joint angles using rigid bodies because of day-to-day identification differences of bony landmarks on the subject to which the coordinate frames are referenced. A day of model will incorporate these errors and allow for better predictions than if a model from a previous day was used.

However, there are some trade-offs in order to achieve this new model. The main negative of completing a day of model update is the time needed to complete the day-of modeling process which is nearly 30 minutes. Additionally, by maintaining the hyperparameters from a previous day, there may be some errors because these parameters will not be based on the joint measurements of the current day.

The system model was used to produce muscle torque capability matrices (see (7.3)) as well as predict the passive force needed to hold the wrist at a static position. This predicted force was then applied by the robot to allow the arm to move more freely against the elastic force of the arm support. This force did not actively control the arm and only provided support against the forces of gravity and the arm support.

An additional requirement of the model when practically implementing the controller is to map the stimulation input to the muscle activations. This mapping is known as the recruitment curves. The mapping was identified using the deconvolved ramp method [44].

### Trajectory optimization

Using the day of model, we used the trajectory optimization scheme and simulation of arm dynamics developed in section 7.2.1 to find feasible trajectories which accounted to the subject's muscle capabilities and the simulated dynamics of the system. The starting configuration for all trajectories was selected to be the central resting position as defined by the first target of the day of model identification. The remaining 12 training configurations were used as target configurations. This allowed us to have a known, comfortable configuration at the start and end of the trajectory.

As described in section 7.2.1, for each target configuration, we attempted to find a feasible two-second trajectory using direct collocation to formulate the trajectory optimization

as the the following nonlinear optimization problem,

$$
\begin{aligned}
\underset{\boldsymbol{\alpha},\boldsymbol{q}_{traj}}{\text{minimize:}} \quad & \text{mean}(\boldsymbol{\alpha}_{traj}^2) + \gamma\,\text{mean}((\boldsymbol{q}_{traj} - \boldsymbol{q}_{targ})^2) \\
\text{subject to:} \quad & \alpha_i(k) \in [0,1] \quad \forall i \in \{1,2,\ldots,9\}\,, \quad \forall k \in \{1,2,\ldots,n\} \\
& \boldsymbol{q}_{traj}(k+1) = f(\boldsymbol{q}_{traj}(k), \alpha(k)), \quad \forall k \in \{1,2,\ldots,n-1\} \\
& \boldsymbol{q}_{traj}(1) = \boldsymbol{q}_0 \\
& \boldsymbol{q}_{traj}(n) = \boldsymbol{q}_{targ} \\
& \boldsymbol{q}_{min} \leq \boldsymbol{q}_{traj}(k) \leq \boldsymbol{q}_{max}, \quad \forall k \in \{1,2,\ldots,n\}
\end{aligned}
\quad . \tag{7.10}
$$

The first term in the objective function minimizes the average of the squared muscle activations throughout the entire trajectory, $\boldsymbol{\alpha}_{traj}$, and the second term attempts to minimize the distance from each configuration in the trajectory, $\boldsymbol{q}_{traj}$ to the target position, $\boldsymbol{q}_{targ}$. To ensure subject comfort, the trajectories were constrained to remain inside the configurations seen during model training. This range was expanded by 11.5 degrees in each direction to ensure the target positions were within the range. The second term of the objective function was also introduced with subject comfort in mind. Due to time constraints, each trajectory optimization was only completed one time. $\gamma$ was selected to be $\gamma = 1$ rad$^{-2}$ to achieve the overall goal of balancing the objectives of minimal activations and direct path reaches. There were also constraints ensuring feasible muscle activations and the trajectory would start and end at the desired configurations with zero velocity. If a trajectory was not found, the target was removed, and the next target was attempted. For the 12 possible reaches, we were able to find 11 feasible reaching trajectories. The process of finding the 11 trajectories required approximately 15 minutes.

Once a trajectory was found, a full-reach trajectory was created. The full trajectory was five seconds long. First, the starting position was held for one second. By using the robot, the subject's hand would always start in the same position. However, the configuration was not guaranteed to be the correct starting position. By holding the starting position for one second, the controller had a chance to correct for initial errors and start its motion

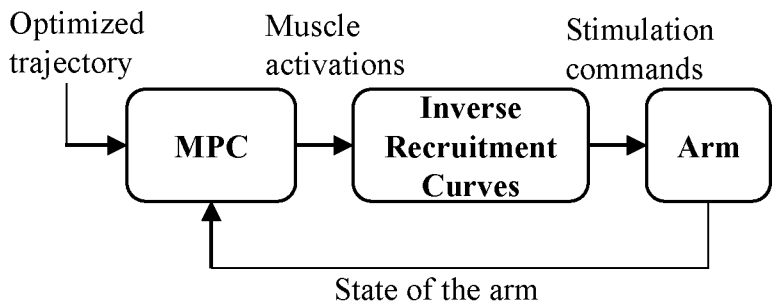



Figure 28: MPC controller block diagram

near the correct configuration. The next two seconds of the trajectory were the optimized trajectory found above. Finally, the final two seconds of the trajectory were to hold the target configuration to allow for the controller time to correct for errors.

### Controller

We use our subject-specific model as the basis of an MPC controller to drive the arm of a subject with high tetraplegia along a desired trajectory to a target configuration (see Fig. 28). The input to the controller is the optimized desired trajectory and the current state of the arm. The MPC controller then calculates the desired muscle activations to best achieve the desired trajectory. The inverse recruitment curves block then determines the stimulation commands that correspond to the desired muscle activations, and the stimulation is applied to the arm.

The state of the arm is defined by the arm configuration - shoulder elevation plane, shoulder elevation, shoulder rotation, elbow flexion - and the joint velocities. The state of the arm was measured with motion capture and a $5^{th}$ order moving average filter was applied to limit the noise. The velocities were calculated using numerical differentiation.

The MPC controller was developed in the same method as presented in 7.2.1, however, adjustments were made for practical implementation on the stimulation hardware and

subject comfort. We developed an MPC control scheme based on the incremental MPC formulation presented in [39] which incorporates the benefits of integral control to the MPC control scheme. The parameters of the MPC controller were tuned on several trajectories with the goal of producing accurate reaches along with comfortable stimulation profiles and motions. Based on the limitations of the system hardware, the prediction horizon was selected to be $n_y = 3$ and the control horizon was $n_u = 2$. This allowed the MPC optimization problem to be solved within the 52 Hz control loop using the active-set method.

The key features for subject comfort were limiting oscillation of the arm and smooth stimulation profiles. To create a smooth stimulation pattern, we selected a scalar weighting on the change in input, $\lambda = 1$. During our initial tuning, it became clear that oscillations would be an issue with the controller. Our previous research (and that of others) has demonstrated that significant oscillations occur with feedback FES control due to the delays in the FES system (low frequency of stimulation and electrical-muscular activation delays). In simulation, we were able to improve the controller performance by adding physical damping to the arm support [40]. Due to this finding, we used the robot to create a damped environment during the reaching experiments. We tuned the damping force to 70 Nm/s in all directions.

Due to the time constraints of the control loop, we linearized the system offline along the desired trajectory. To develop the state-space matrices, the Autolev equations of motion were used to linearize the system about the desired state at the current time. The MATLAB function `c2d` was used to discretize the state-space systems. The output of the state-space models were the joint angles and the reference trajectory included only the discretized desired joint angles as a function of time-step. At each point during the reaching experiment, the controller would use the linearized system matrices from the desired reference state of the arm at the next time step. The reference signal for the controller was the desired arm configuration. Joint velocities were not including in the reference signal due to the aforementioned issues with system delays leading to derivative control instability.

### Experiments and data analysis

For each target reach, the subject's arm was moved by the robot to the desired starting position. By only having a robot connected to the subject's wrist to move to the starting position, we could not ensure that the starting configuration was correct. For the first second of each reach the desired configuration was the starting configuration to allow the controller to attempt to obtain the correct starting configuration. For the duration of the five second reach, the arm was allowed move freely as driven by the muscle stimulations. The robot provided only a damped environment and support against the predicted passive forces at the wrist during the movement. The set of 11 reaching motions was repeated nine times based on the amount of time defined by the subject's schedule. A total of 99 reaches were completed.

To analyze the effectiveness of the controller, we calculated the accuracy as defined as the Euclidean distance away from the desired position/configuration. To ensure that oscillations around the target position are accounted for, the accuracy was defined as the mean Euclidean distance error over the final second of each trial. During the final second the controller is trying to maintain the desired target configuration. The accuracy of the controller was recorded based on the wrist position as placing the hand at a desired location in space is the most important goal of a reaching controller. The wrist position of both the desired position and experimental measurements were calculated using forward kinematics.

To determine the controller's effectiveness throughout the workspace, a two-sample t-test was completed for each grouping of targets to determine if there was improved accuracy to the left or right side of the workspace, up or down, and forward or backwards in the subject's workspace. The targets were grouped based on their position relative to the average position of all the targets.

### 7.3.2 Results

Overall, over 99 reaching motions to 11 different targets, the controller achieved an average wrist position accuracy of 8.5 cm (standard deviation of 2.8 cm). Table VI shows the accuracy results for all trials including a breakdown of the accuracy based on the position of the targets. The controller was able to reach targets throughout the workspace, but it was more accurate to targets on the right side ($p < 0.001$), targets forward($p < 0.001$), and targets down in the workspace($p = 0.002$). These differences in accuracy based on target position are illustrated in Fig. 29. The image shows the average accuracy over the nine sets of reaches for each target position. The size and color of each point represents the relative accuracy for each target position.

A representative reaching trial with an accuracy of 8.5 cm is shown in Fig. 30. The target position is denoted by the red arrow in Fig. 24. The reach is able to move in the correct direction, but there is significant amounts of oscillation near the target position. Shoulder rotation is the joint with the largest error that the controller is unable to correct. As this joint moves away from the desired target, the wrist position also moves away from the desired target.

Fig. 30(c) shows the muscle activation commands for the triceps, biceps/brachialis, and the upper pectoralis muscle groups. These activations demonstrate the ability of the MPC controller built on our muscle capability models to select muscle activations which make sense physiologically. For the elbow flexion angle, the reach first requires elbow extension so the triceps, the main elbow extensor muscle, is activated. As the position overshoots, the biceps/brachialis turn on to stop the elbow extension. These two muscle groups vary in activation to control the elbow flexion as it oscillates around the desired position. Additionally, as the shoulder rotation moves away from the desired target, the integral action of the MPC controller is noticeable as the upper pectoralis, an internal rotator of the arm, increases in activation. While this is a simplified explanation of the reach (the biceps/brachialis and triceps also produce torques about the shoulder), this example reach

| Target location | All | Left | Right | Forward | Back | Up | Down |
|---|---|---|---|---|---|---|---|
| mean accuracy (cm) | 8.5 | 10.0 | 6.7 | 7.2 | 10.0 | 9.4 | 7.7 |
| standard deviation (cm) | 2.8 | 2.2 | 2.2 | 2.4 | 2.3 | 2.3 | 2.9 |
| $p$-value | | $< 0.001$ | | $< 0.001$ | | 0.002 | |

Table VI: Wrist position error for all targets and broken down by the position of the targets

demonstrates the potential of the control strategy to achieve reaching motions throughout the workspace.

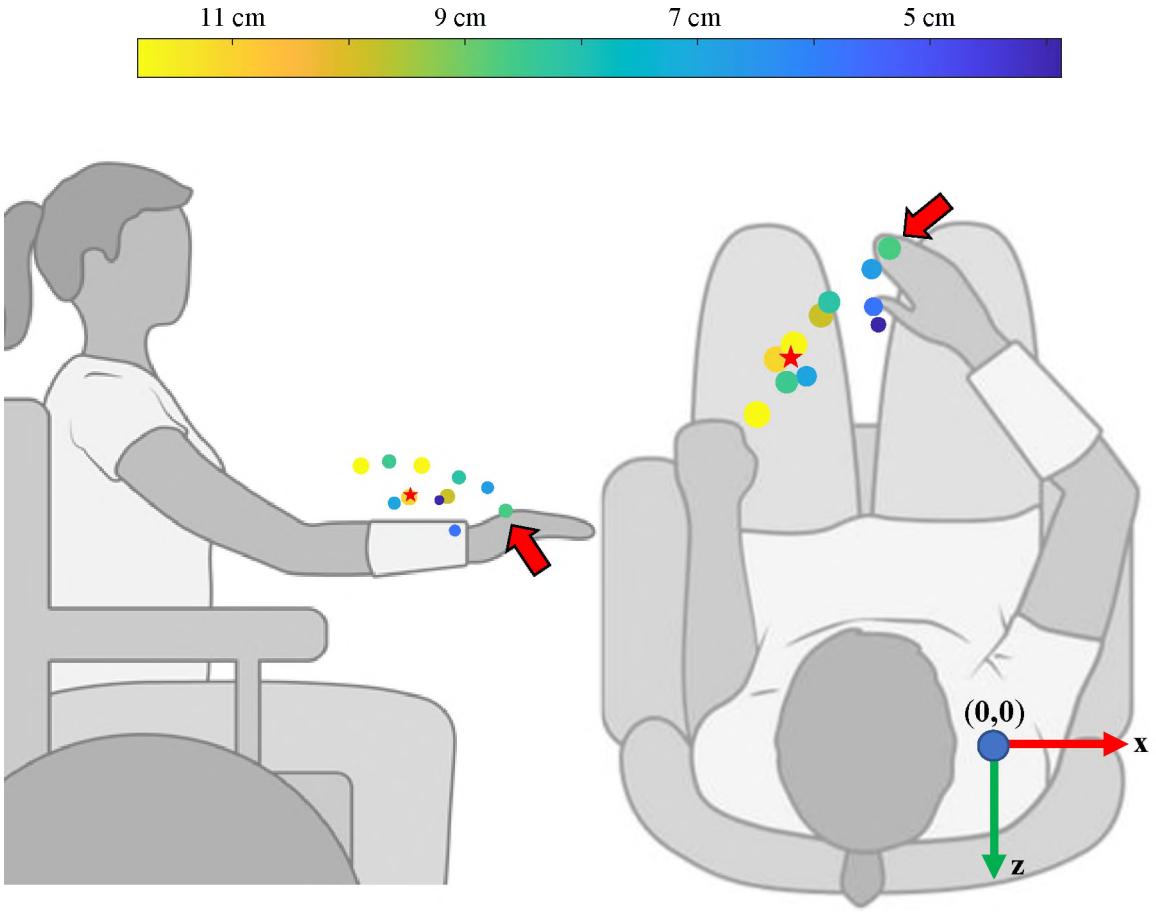



Figure 29: This figures shows the average accuracy of each target position over nine sets of reaching trials. The desired starting position for every reach is denoted by the red star. The accuracy of the target is denoted by both the size and color of the circle. The accuracy of the reaches improved for targets to the right, to the front, and down in the workspace. The red arrow denotes the representative target position which is shown in Fig. 30. The coordinate frame and origin of the reaching Cartesian workspace is also shown at the subject's right shoulder.

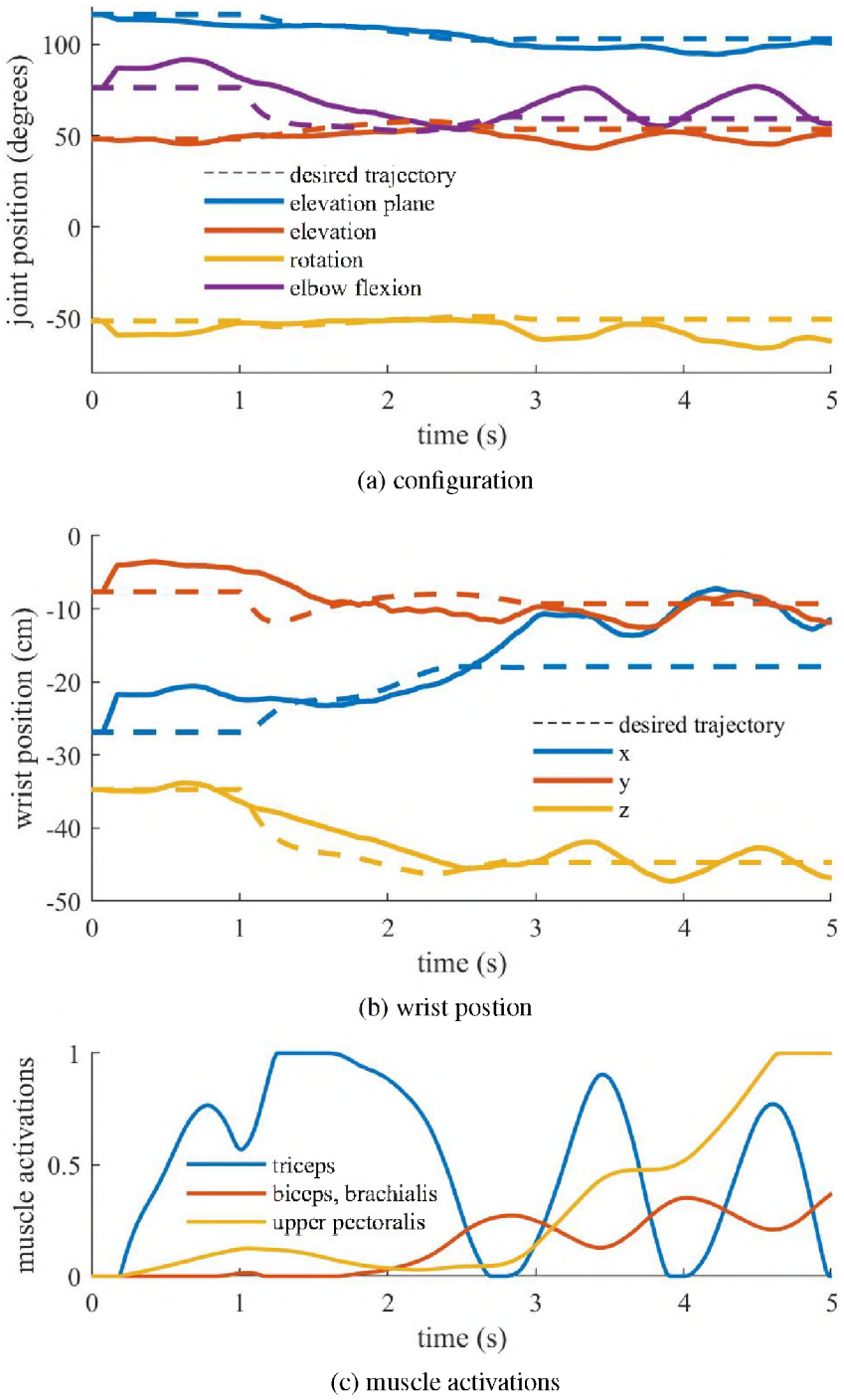



Figure 30: This figure shows a representative example of a FES-driven reaching motion controlled by our MPC scheme in both (a) configuration space and (b) Cartesian wrist position space. (c) shows the muscle activation commands for three muscle groups, the triceps, biceps/brachialis, and the upper pectoralis. These muscle group activations demonstrate the ability of the MPC control scheme to select muscle activations which make sense physiologically and are able to control the motion of the arm.

### 7.3.3 Discussion

We present a novel control structure that is capable of achieving FES-driven reaching motions throughout the workspace of a subject with high tetraplegia due to SCI. The use of trajectory optimization and an MPC controller to directly account for the subject's muscle capabilities resulted in significant improvements to the performance compared to past FES reaching controllers. First, though there were differences in accuracy based on the target position in the workspace, the control strategy was able to move the arm throughout the space. The MPC controller was able to avoid getting stuck in uncontrollable configurations as seen by the straight-line feedback controller in [20]. Also, we achieved similar accuracy to the accuracy achieved by our simulation study with model uncertainty presented in section 7.2 (10.9 cm accuracy for the simulated MPC controller and 8.5 cm accuracy for the implemented MPC controller). This is encouraging for using the simulation to guide further development of the control structure including using the simulation study to better understand the types of uncertainty which the MPC controller is robust to. Additionally, by controlling the whole arm, we are able to produce more natural motions than the reaches achieved by MUNDUS [17] once a trajectory is found. However, trajectory optimization can take a significant amount of time (from about ten seconds at a minimum and up to five minutes if 1,500 iterations occur). To complete daily reaching tasks using trajectory optimization for each reach, the optimization needs to occur significantly faster to be practical.

While the positives are significant, there were important limitations to the study. The overall accuracy of 8.5 cm is worse than the accuracy produced by our own previous work in [20] and other controllers [10, 17, 18]. This accuracy is not good enough to complete many activities of daily living including eating off a plate though some compensatory torso movements could assist these errors. A much greater issue is the oscillation which makes completing daily tasks difficult (consider eating off a fork that is shaking) and can become uncomfortable for the user. To improve both the oscillation and the accuracy of the controller, the model at the basis of the MPC control strategy needs to be improved. Our current

simulation of the arm only accounts for basic multi-body dynamics of the arm with estimated mass properties and the capabilities of the muscles. A more advanced model which explicitly included muscle activation dynamics, the stiffness introduced by the arm support, and delays in control inputs because of the stimulation frequency could significantly improve the model. Our previous simulation study [40] has shown that these nonlinearities produce instability and oscillation that derivative gain cannot account for. By improving the modeling of these sources of error, the reference trajectory could include velocity and the controller could better use that information to eliminate oscillation.

Identifying the parameters of the subject's arm dynamics is a difficult process which requires gathering a significant amount of data. One potential option would be to use semiparametric models of the muscle capabilities, similar to those presented in this paper for isometric positions, that include the arm dynamics [21]. With this model of the arm dynamics and muscle force production matrices, we could better predict the dynamics of the arm and achieve more accurate trajectories. A similar method has been shown to work in robotic simulations using a receding horizon LQR controller and a GPR model of the robot dynamics [36].

This study represents the first implementation of an MPC controller for FES-controlled reaching motions. The use of trajectory optimization and MPC control creates a control scheme which can account for the unique muscle capabilities of individuals with SCI including muscle weakness due to fatigue and atrophy as well as complete loss of muscle function due to lower motor neuron damage. With an improved model, this control scheme has the potential to unlock many activities of daily living that require reaching motions for individuals with SCI.

**Acknowledgements**

We sincerely thank Dr. Ton van den Bogert for his extensive support and guidance in developing the arm simulation and the trajectory optimization methods presented in this paper.

We would also like to thank Dr. Hanz Richter for his input in developing the MPC controller.

## 7.4 REFERENCES

[1] National Spinal Cord Injury Statistical Center, Facts and Figures at a Glance. *Birmingham, AL: University of Alabama at Birmingham*, 2020.

[2] Kim D. Anderson. Targeting Recovery: Priorities of the Spinal Cord-Injured Population. *Journal of Neurotrauma*, 21(10):1371–1383, oct 2004. ISSN 0897-7151. doi: 10.1089/neu.2004.21.1371.

[3] K. T. Ragnarsson. Functional electrical stimulation after spinal cord injury: Current use, therapeutic effects and future directions. *Spinal Cord*, 46(4):255–274, 2008. ISSN 13624393. doi: 10.1038/sj.sc.3102091.

[4] Naji A. Alibeji, Nicholas Andrew Kirsch, and Nitin Sharma. A Muscle Synergy-Inspired Adaptive Control Scheme for a Hybrid Walking Neuroprosthesis. *Frontiers in Bioengineering and Biotechnology*, 3(December):1–13, 2015. ISSN 2296-4185. doi: 10.3389/fbioe.2015.00203.

[5] Kevin L. Kilgore, Harry A. Hoyen, Anne M. Bryden, Ronald L. Hart, Michael W. Keith, and P. Hunter Peckham. An Implanted Upper-Extremity Neuroprosthesis Using Myoelectric Control. *The Journal of Hand Surgery*, 33(4):539–550, Apr 2008. ISSN 03635023. doi: 10.1016/j.jhsa.2008.01.007.

[6] E. Ambrosini, S. Ferrante, J. Zajc, M. Bulgheroni, W. Baccinelli, E. D'Amico, T. Schauer, C. Wiesener, M. Russold, M. Gfoehler, M. Puchinger, M. Weber, S. Becker, K. Krakow, M. Rossini, D. Proserpio, G. Gasperini, F. Molteni, G. Ferrigno, and A. Pedrocchi. The combined action of a passive exoskeleton and an EMG-controlled neuroprosthesis for upper limb stroke rehabilitation: First results

of the RETRAINER project. In *2017 International Conference on Rehabilitation Robotics (ICORR)*, pages 56–61. IEEE, Jul 2017. ISBN 978-1-5386-2296-4. doi: 10.1109/ICORR.2017.8009221.

[7] Kathleen M. Jagodnik, Philip S. Thomas, Antonie J. Van Den Bogert, Michael S. Branicky, and Robert F. Kirsch. Training an Actor-Critic Reinforcement Learning Controller for Arm Movement Using Human-Generated Rewards. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 25(10):1892–1905, oct 2017. ISSN 15344320. doi: 10.1109/TNSRE.2017.2700395.

[8] Kathleen M. Jagodnik and Antonie J. van den Bogert. Optimization and evaluation of a proportional derivative controller for planar arm movement. *Journal of Biomechanics*, 43(6):1086–1091, Apr 2010. ISSN 00219290. doi: 10.1016/j.jbiomech.2009.12.017.

[9] Christopher T Freeman. Upper Limb Electrical Stimulation Using Input-Output Linearization and Iterative Learning Control. *IEEE Transactions on Control Systems Technology*, 23(4):1546–1554, Jul 2015. ISSN 1063-6536. doi: 10.1109/TCST.2014.2363412.

[10] Reza Sharif Razavian, Borna Ghannadi, Naser Mehrabi, Mark Charlet, and John McPhee. Feedback Control of Functional Electrical Stimulation for 2-D Arm Reaching Movements. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 26(10):2033–2043, oct 2018. ISSN 1534-4320. doi: 10.1109/TNSRE.2018.2853573.

[11] Lynne R. Sheffler and John Chae. Neuromuscular electrical stimulation in neurorehabilitation. *Muscle & Nerve*, 35(5):562–590, 2007. ISSN 0148639X. doi: 10.1002/mus.20758.

[12] Ryan J. Downey, Matthew J. Bellman, Hiroyuki Kawai, Chris M. Gregory, and War-

ren E. Dixon. Comparing the Induced Muscle Fatigue Between Asynchronous and Synchronous Electrical Stimulation in Able-Bodied and Spinal Cord Injured Populations. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 23(6): 964–972, 2015. ISSN 15344320. doi: 10.1109/TNSRE.2014.2364735.

[13] P. Hunter Peckham and Jayme S. Knutson. Functional Electrical Stimulation for Neuromuscular Applications. *Annual Review of Biomedical Engineering*, 7(1):327–360, 2005. ISSN 1523-9829. doi: 10.1146/annurev.bioeng.6.040803.140103.

[14] Nathan Dunkelberger, Eric M. Schearer, and Marcia K. O'Malley. A review of methods for achieving upper limb movement following spinal cord injury through hybrid muscle stimulation and robotic assistance. *Experimental Neurology*, 328(February): 113274, Jun 2020. ISSN 00144886. doi: 10.1016/j.expneurol.2020.113274.

[15] Martin Rohm, Matthias Schneiders, Constantin Müller, Alex Kreilinger, Vera Kaiser, Gernot R Müller-Putz, and Rüdiger Rupp. Hybrid brain–computer interfaces and hybrid neuroprostheses for restoration of upper limb functions in individuals with high-level spinal cord injury. *Artificial Intelligence in Medicine*, 59(2):133–142, 2013. ISSN 0933-3657. doi: https://doi.org/10.1016/j.artmed.2013.07.004.

[16] Christian Klauer, Thomas Schauer, Werner Reichenfelser, Jakob Karner, Sven Zwicker, Marta Gandolla, Emilia Ambrosini, Simona Ferrante, Marco Hack, Andreas Jedlitschka, Alexander Duschau-Wicke, Margit Gföhler, and Alessandra Pedrocchi. Feedback control of arm movements using Neuro-Muscular Electrical Stimulation (NMES) combined with a lockable, passive exoskeleton for gravity compensation. *Frontiers in Neuroscience*, 8(Sep):1–16, 2014. ISSN 1662453X. doi: 10.3389/fnins.2014.00262.

[17] Alessandra Pedrocchi, Simona Ferrante, Emilia Ambrosini, Marta Gandolla, Claudia Casellato, Thomas Schauer, Christian Klauer, Javier Pascual, Carmen Vidaurre, Mar-

git Gföhler, Werner Reichenfelser, Jakob Karner, Silvestro Micera, Andrea Crema, Franco Molteni, Mauro Rossini, Giovanna Palumbo, Eleonora Guanziroli, Andreas Jedlitschka, Marco Hack, Maria Bulgheroni, Enrico D'Amico, Peter Schenk, Sven Zwicker, Alexander Duschau-Wicke, Justinas Miseikis, Lina Graber, and Giancarlo Ferrigno. MUNDUS project: MUltimodal Neuroprosthesis for daily Upper limb Support. *Journal of NeuroEngineering and Rehabilitation*, 10(1):66, 2013. ISSN 1743-0003. doi: 10.1186/1743-0003-10-66.

[18] A. Bolu Ajiboye, Francis R. Willett, Daniel R. Young, William D. Memberg, Brian A. Murphy, Jonathan P. Miller, Benjamin L. Walter, Jennifer A. Sweet, Harry A. Hoyen, Michael W. Keith, P. Hunter Peckham, John D. Simeral, John P. Donoghue, Leigh R. Hochberg, and Robert F. Kirsch. Restoration of reaching and grasping movements through brain-controlled muscle stimulation in a person with tetraplegia: a proof-of-concept demonstration. *The Lancet*, 389(10081):1821–1830, May 2017. ISSN 01406736. doi: 10.1016/S0140-6736(17)30601-3.

[19] Derek N. Wolf and Eric M. Schearer. Holding Static Arm Configurations With Functional Electrical Stimulation: A Case Study. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 26(10):2044–2052, oct 2018. ISSN 1534-4320. doi: 10.1109/TNSRE.2018.2866226.

[20] D N Wolf, Z A Hall, and E M Schearer. Model Learning for Control of a Paralyzed Human Arm with Functional Electrical Stimulation. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 10148–10154, 2020. ISBN 2577-087X VO -. doi: 10.1109/ICRA40945.2020.9196992.

[21] Eric M. Schearer, Yu-Wei Liao, Eric J. Perreault, Matthew C. Tresch, William D. Memberg, Robert F. Kirsch, and Kevin M. Lynch. Semiparametric Identification of Human Arm Dynamics for Flexible Control of a Functional Electrical Stimulation Neuroprosthesis. *IEEE Transactions on Neural Systems and Rehabilitation Engi-*

*neering*, 24(12):1405–1415, Dec 2016. ISSN 1534-4320. doi: 10.1109/TNSRE.2016.2535348.

[22] Nicholas Kirsch, Naji Alibeji, and Nitin Sharma. Nonlinear model predictive control of functional electrical stimulation. *Control Engineering Practice*, 58:319–331, Jan 2017. ISSN 09670661. doi: 10.1016/j.conengprac.2016.03.005.

[23] Xuefeng Bao, Nicholas Kirsch, and Nitin Sharma. Dynamic control allocation of a feedback linearized hybrid neuroprosthetic system. In *2016 American Control Conference (ACC)*, volume 2016-July, pages 3976–3981. IEEE, Jul 2016. ISBN 978-1-4673-8682-1. doi: 10.1109/ACC.2016.7525534.

[24] D N Wolf and E M Schearer. Evaluating an open-loop functional electrical stimulation controller for holding the shoulder and elbow configuration of a paralyzed arm. In *2017 International Conference on Rehabilitation Robotics (ICORR)*, pages 789–794, 2017. ISBN 1945-7901 VO -. doi: 10.1109/ICORR.2017.8009344.

[25] Katharine H. Polasek, Harry A. Hoyen, Michael W. Keith, Robert F. Kirsch, and Dustin J. Tyler. Stimulation stability and selectivity of chronically implanted multicontact nerve cuff electrodes in the human upper extremity. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 17(5):428–437, oct 2009. ISSN 15344320. doi: 10.1109/TNSRE.2009.2032603.

[26] Brian Smith, P Hunter Peckham, Michael W Keith, and Dennis D Roscoe. Stimulator for Versatile Control of Paralyzed Muscle. *IEEE Transactions on Biomedical Engineering*, BME-34(7):499–508, 1987. ISSN 0018-9294. doi: 10.1109/TBME.1987.325979.

[27] B. Smith, Zhengnian Tang, M.W. Johnson, S. Pourmehdi, M.M. Gazdik, J.R. Buckett, and P.H. Peckham. An externally powered, multichannel, implantable stimulator-

telemeter for control of paralyzed muscle. *IEEE Transactions on Biomedical Engineering*, 45(4):463–475, Apr 1998. ISSN 00189294. doi: 10.1109/10.664202.

[28] Ronald L. Hart, Niloy Bhadra, Fred W. Montague, Kevin L. Kilgore, and P. Hunter Peckham. Design and testing of an advanced implantable neuroprosthesis with myoelectric control. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 19(1):45–53, 2011. ISSN 15344320. doi: 10.1109/TNSRE.2010.2079952.

[29] W.D. Memberg, P.H. Peckham, and M.W. Keith. A surgically-implanted intramuscular electrode for an implantable neuromuscular stimulation system. *IEEE Transactions on Rehabilitation Engineering*, 2(2):80–91, Jun 1994. ISSN 10636528. doi: 10.1109/86.313149.

[30] Gregory G. Naples, J. Thomas Mortimer, Avram Scheiner, and James D. Sweeney. A Spiral Nerve Cuff Electrode for Peripheral Nerve Stimulation. *IEEE Transactions on Biomedical Engineering*, 35(11):905–916, 1988. ISSN 15582531. doi: 10.1109/10.8670.

[31] Ge Wu, Frans C.T. Van Der Helm, H. E.J. Veeger, Mohsen Makhsous, Peter Van Roy, Carolyn Anglin, Jochem Nagels, Andrew R. Karduna, Kevin McQuade, Xuguang Wang, Frederick W. Werner, and Bryan Buchholz. ISB recommendation on definitions of joint coordinate systems of various joints for the reporting of human joint motion - Part II: Shoulder, elbow, wrist and hand. *Journal of Biomechanics*, 38(5):981–992, 2005. ISSN 00219290. doi: 10.1016/j.jbiomech.2004.05.042.

[32] CE Rasmussen and CKI Williams. *Gaussian processes for machine learning*. 2006.

[33] Eric M. Schearer, Yu Wei Liao, Eric J. Perreault, Matthew C. Tresch, William D. Memberg, Robert F. Kirsch, and Kevin M. Lynch. Multi-muscle FES force control of the human arm for arbitrary goals. *IEEE Transactions on Neural Systems and*

*Rehabilitation Engineering*, 22(3):654–663, 2014. ISSN 15344320. doi: 10.1109/TNSRE.2013.2282903.

[34] E.K. Chadwick, D Blana, A.J. van den Bogert, and R.F. Kirsch. A Real-Time, 3-D Musculoskeletal Model for Dynamic Simulation of Arm Movements. *IEEE Transactions on Biomedical Engineering*, 56(4):941–948, Apr 2009. ISSN 0018-9294. doi: 10.1109/TBME.2008.2005946.

[35] David A Levinson and Thomas R Kane. AUTOLEV — A New Approach to Multibody Dynamics BT - Multibody Systems Handbook. pages 81–102. Springer Berlin Heidelberg, Berlin, Heidelberg, 1990. ISBN 978-3-642-50995-7. doi: 10.1007/978-3-642-50995-7_7.

[36] J Boedecker, J T Springenberg, J Wülfing, and M Riedmiller. Approximate real-time optimal control based on sparse Gaussian process models. In *2014 IEEE Symposium on Adaptive Dynamic Programming and Reinforcement Learning (ADPRL)*, pages 1–8, 2014. ISBN 2325-1867 VO -. doi: 10.1109/ADPRL.2014.7010608.

[37] Marko Ackermann and Antonie J van den Bogert. Optimality principles for model-based prediction of human gait. *Journal of Biomechanics*, 43(6):1055–1060, 2010. ISSN 0021-9290. doi: https://doi.org/10.1016/j.jbiomech.2009.12.012.

[38] Andreas Wächter. An interior point algorithm for large-scale nonlinear optimization with applications in process engineering, 2002.

[39] Hanz Richter. *Advanced control of turbofan engines*. Springer Science & Business Media, 2011. ISBN 1461411718.

[40] Derek N. Wolf and Eric M. Schearer. Developing a quasi-static controller for a paralyzed human arm: A simulation study. In *IEEE International Conference on Rehabilitation Robotics*, volume 2019-June, pages 1153–1158. IEEE, Jun 2019. ISBN 9781728127552. doi: 10.1109/ICORR.2019.8779381.

[41] Diego S Carrasco and Graham C Goodwin. Feedforward model predictive control. *Annual Reviews in Control*, 35(2):199–206, 2011. ISSN 1367-5788. doi: https://doi.org/10.1016/j.arcontrol.2011.10.007.

[42] Eric M Schearer and Derek N Wolf. Predicting functional force production capabilities of upper extremity functional electrical stimulation neuroprostheses: a proof of concept study. *Journal of Neural Engineering*, 17(1):16051, 2020. ISSN 1741-2552.

[43] Matt Zucker, Nathan Ratliff, Anca D. Dragan, Mihail Pivtoraiko, Matthew Klingensmith, Christopher M. Dellin, J. Andrew Bagnell, and Siddhartha S. Srinivasa. CHOMP: Covariant Hamiltonian optimization for motion planning. *International Journal of Robotics Research*, 32(9-10):1164–1193, 2013. ISSN 02783649. doi: 10.1177/0278364913488805.

[44] William K. Durfee and Karon E. MaClean. Methods for Estimating Isometric Recruitment Curves of Electrically Stimulated Muscle. *IEEE Transactions on Biomedical Engineering*, 36(7):654–667, 1989. ISSN 15582531. doi: 10.1109/10.32097.

# CHAPTER VIII

# CONCLUSION

## 8.1 Contributions

Functional electrical stimulation (FES) is a promising technology for restoring reaching motions to individuals with tetraplegia due to spinal cord injury (SCI). While FES has been successful at restoring some function to individuals with SCI including grasping [1] and walking [2], these successes have yet to be fully transmitted to full-arm reaching motions. The state-of-the-art systems either do not control the arm as a complete system [3, 4] which leads to unnatural movements and errors in the system, or work with healthy subjects in planar motions [5] which cannot be directly translated to individuals with SCI. This dissertation developed a control scheme which included learning subject-specific muscle capability models, using trajectory optimization to determine feasible trajectories, and applying a model predictive control (MPC) controller to achieve full-arm reaching motions in individuals with high tetraplegia due to SCI. I defined two main aims:

**Aim 1:** Develop a complete-arm FES-driven reaching controller that is capable of holding static hand positions for an individual with high tetraplegia due to SCI.

**Aim 2:** Develop a subject-specific model-based control strategy to use FES to drive the arm of an individual with high tetraplegia due to SCI along a desired path in the subject's workspace.

In Chapter III, I developed a subject-specific model of the arm of an individual with

tetraplegia and its static response to stimulation. The model was validated by using it for open-loop control of static wrist positions. This subject-specific model is the key foundation to the controllers developed in this dissertation. In Chapter IV, we built on the open-loop control structure by adding feedback control which used the subject-specific muscle capability models. With this addition, we completed Aim 1 and were able to control static wrist positions throughout the subject's workspace. This work demonstrated the potential for a subject-specific model to be used as the basis of a controller for FES-controlled reaching.

In Chapter V, we developed a series of improvements to the controller to allow for reaching motions to be completed. First, reaching motions with no planned trajectory were attempted. The oscillation and errors from this process demonstrated the need for some level of path planning. Next, in a simulation study, straight line, quasi-static paths were simulated to determine the conditions and environment in which successful FES-controlled reaching could occur. It became clear from this study that the delays in the system due to the ability to switch control inputs only at the stimulation frequency of 13 Hz lead to instability which cannot be fixed with simple derivative control. The solution was to create a damped environment using a mobile arm support. Building on these results, in Chapter VI, we attempted to control the arm along straight line reaching paths using a subject-specific, model-based feedback controller. The controller was able to achieve fairly good accuracy, but it would get "stuck" due to ending up in uncontrollable locations in the workspace which resulted in feedback overcompensation, the asking of more torque than is possible.

To try to avoid the uncontrollable arm configurations, in Chapter VII, we developed a trajectory optimization scheme to determine feasible trajectories to achieve target wrist positions. We then developed an MPC controller that capable of controlling the arm in simulation and practically implemented the control scheme in an individual with tetraplegia. This Chapter completed Aim 2. The main contribution of this Chapter is the clear demonstration of the need for both trajectory planning and for the controller to explicitly take

into account the muscle capabilities of an individual with tetraplegia in order to achieve accurate FES-controlled reaching motions.

## 8.2 Future Work

While the controller developed in this dissertation presents a significant step towards the goal of restoring reaching motions to individuals with tetraplegia, there are still several key areas of improvement that are needed. The framework for FES control presented by Lynch and Popovic [6] is a good source to identify these needed improvements, "The FES system must:

1. compensate for the nonlinear, time-varying, and coupled nature of the muscle being controlled, including the effects of fatigue and training.

2. be stable in the presence of the time delays and perturbations (reflex contractions) that are inherenet to the system.

3. be implemented in portable, battery powered electronics, and should be designed for at least 16 hours of operation each day...

4. be compatible with efficient setup and calibration procedures that are simple enough to be performed by a therapist or patient..."

Working with a subject with SCI (as is recommended by Lynch and Popovic because of the unique muscle characteristics of individuals with SCI), the work in this dissertation has attempted to meet the first two goals. The model based control strategy offers a framework that is capable of controlling the full-arm and accounting for the coupled dynamics of the system. However, the accuracy of our controller was not sufficient for every day use and there were oscillations in the motion. To fully meet the first two requirements of the framework using our MPC control scheme, a more sophisticated model which better includes the nonlinear dynamics and time delays inherent to the system needs to be developed.

Additionally, our model identification procedures and trajectory optimization methods take too long to be used for everyday reaching tasks. Even our current day of model update procedure requires nearly 30 minutes to gather data and develop the new model. There is a critical need to develop modeling methods which ideally can update in real-time as reaches occur compensating for the rapid fatigue seen in individuals with SCI. At the very least, there needs to be a model update procedure that can be quickly and easily performed by a caregiver.

Our trajectory optimization routine is also too slow for everyday use. While some trajectories are found in less than 10 seconds, others take several minutes. Since there is no way to know in advance how long a single optimization will take, or if a feasible solution even exists, it is impractical to implement this method as it is currently developed for everyday reaching control. A subject will be unwilling to use the system if every time they want to complete a reach, there is a chance that it will take three minutes before the reach can begin. Additionally, some optimized trajectories were near the edge of the controllability and errors in the model resulted in feedback overcompensation. While the MPC controller is better able to deal with this situation than the original feedback controller, there is room to improve the controller by better avoiding uncontrollable configurations. Further research should be completed to determine if additional terms should be added to our optimization objective function to bias the system towards controllable configurations, or if an entirely new trajectory planning method should be used.

While the major reaching goal of this paper was to place the subject's wrist at the desired location. For achieving everyday reaching tasks, the subject may have additional task-specific goals including the orientation of the hand or stiffness of the arm. Adding these target outcomes to the control structure developed here will result in a dramatic increase in complexity. The additional task constraints will make it more difficult to find feasible trajectories, and the additional degrees of freedom will substantially increase the complexity of control.

Lastly, it is important to gain a clinical understanding of the capabilities of the controller. The best way to do this would be to work with a clinician to implement testing such as the Capabilities of Upper Extremities Test (CUE-T) [7]. Doing so will provide a better understanding of the feasibility of implementing the current controller and what key areas of improvement are needed for everyday use of the system.

Improvements in modeling, trajectory planning, and a greater range of target outcomes would be significant developments towards the implementation of the control scheme developed in this dissertation to restore everyday reaching motions in individuals with tetraplegia due to spinal cord injuries.

## 8.3 REFERENCES


[1] Ronald L. Hart, Niloy Bhadra, Fred W. Montague, Kevin L. Kilgore, and P. Hunter Peckham. Design and testing of an advanced implantable neuroprosthesis with myoelectric control. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 19(1):45–53, 2011. ISSN 15344320. doi: 10.1109/TNSRE.2010.2079952.

[2] Naji A. Alibeji, Nicholas Andrew Kirsch, and Nitin Sharma. A Muscle Synergy-Inspired Adaptive Control Scheme for a Hybrid Walking Neuroprosthesis. *Frontiers in Bioengineering and Biotechnology*, 3(December):1–13, 2015. ISSN 2296-4185. doi: 10.3389/fbioe.2015.00203.

[3] A. Bolu Ajiboye, Francis R. Willett, Daniel R. Young, William D. Memberg, Brian A. Murphy, Jonathan P. Miller, Benjamin L. Walter, Jennifer A. Sweet, Harry A. Hoyen, Michael W. Keith, P. Hunter Peckham, John D. Simeral, John P. Donoghue, Leigh R. Hochberg, and Robert F. Kirsch. Restoration of reaching and grasping movements through brain-controlled muscle stimulation in a person with tetraplegia: a proof-of-concept demonstration. *The Lancet*, 389(10081):1821–1830, May 2017. ISSN 01406736. doi: 10.1016/S0140-6736(17)30601-3.

[4] Alessandra Pedrocchi, Simona Ferrante, Emilia Ambrosini, Marta Gandolla, Claudia Casellato, Thomas Schauer, Christian Klauer, Javier Pascual, Carmen Vidaurre, Margit Gföhler, Werner Reichenfelser, Jakob Karner, Silvestro Micera, Andrea Crema, Franco Molteni, Mauro Rossini, Giovanna Palumbo, Eleonora Guanziroli, Andreas Jedlitschka, Marco Hack, Maria Bulgheroni, Enrico D'Amico, Peter Schenk, Sven Zwicker, Alexander Duschau-Wicke, Justinas Miseikis, Lina Graber, and Giancarlo Ferrigno. MUNDUS project: MUltimodal Neuroprosthesis for daily Upper limb Support. *Journal of NeuroEngineering and Rehabilitation*, 10(1):66, 2013. ISSN 1743-0003. doi: 10.1186/1743-0003-10-66.

[5] Reza Sharif Razavian, Borna Ghannadi, Naser Mehrabi, Mark Charlet, and John McPhee. Feedback Control of Functional Electrical Stimulation for 2-D Arm Reaching Movements. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 26(10):2033–2043, oct 2018. ISSN 1534-4320. doi: 10.1109/TNSRE.2018.2853573.

[6] C L Lynch and M R Popovic. Closed-loop control for FES: Past work and future directions. In *10th Annual Conference of the International FES Society*, pages 2–4, 2005.

[7] Ralph J. Marino, Mary Patrick, Whitney Albright, Benjamin E. Leiby, MJ Mulcahey, Mary Schmidt-Read, and Stephen B. Kern. Development of an objective test of upper-limb function in tetraplegia: The capabilities of upper extremity test. *American Journal of Physical Medicine and Rehabilitation*, 91(6):478–486, Jun 2012. ISSN 08949115. doi: 10.1097/PHM.0b013e31824fa6cc.

# APPENDIX

## Copyrights and Permissions

### Personal published work permissions

All papers that I have personally published that are presented in this dissertation have been through IEEE conferences and journals.

Thesis / Dissertation Reuse

**The IEEE does not require individuals working on a thesis to obtain a formal reuse license, however, you may print out this statement to be used as a permission grant:**

*Requirements to be followed when using any portion (e.g., figure, graph, table, or textual material) of an IEEE copyrighted paper in a thesis:*

1) In the case of textual material (e.g., using short quotes or referring to the work within these papers) users must give full credit to the original source (author, paper, publication) followed by the IEEE copyright line © 2011 IEEE.
2) In the case of illustrations or tabular material, we require that the copyright line © [Year of original publication] IEEE appear prominently with each reprinted figure and/or table.
3) If a substantial portion of the original paper is to be used, and if you are not the senior author, also obtain the senior author's approval.

*Requirements to be followed when using an entire IEEE copyrighted paper in a thesis:*

1) The following IEEE copyright/ credit notice should be placed prominently in the references: © [year of original publication] IEEE. Reprinted, with permission, from [author names, paper title, IEEE publication title, and month/year of publication]
2) Only the accepted version of an IEEE copyrighted paper can be used when posting the paper or your thesis on-line.
3) In placing the thesis on the author's university website, please display the following message in a prominent place on the website: In reference to IEEE copyrighted material which is used with permission in this thesis, the IEEE does not endorse any of [university/educational entity's name goes here]'s products or services. Internal or personal use of this material is permitted. If interested in reprinting/republishing IEEE copyrighted material for advertising or promotional purposes or for creating new collective works for resale or redistribution, please go to http://www.ieee.org/publications_standards/publications/rights/rights_link.html to learn how to obtain a License from RightsLink.

If applicable, University Microfilms and/or ProQuest Library, or the Archives of Canada may supply single copies of the dissertation.

### Figure reuse permissions

SPRINGER NATURE

**MUNDUS project: MUltimodal Neuroprosthesis for daily Upper limb Support**

**Author:** Alessandra Pedrocchi et al

**Publication:** Journal of NeuroEngineering and Rehabilitation

**Publisher:** Springer Nature

**Date:** Jul 3, 2013

*Copyright © 2013, Pedrocchi et al., licensee BioMed Central Ltd.*

**Creative Commons**

This is an open access article distributed under the terms of the Creative Commons CC BY license, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

You are not required to obtain permission to reuse this article.

To request permission for a type of use not listed, please contact Springer Nature

# APPENDIX

## Copyrights and Permissions



ELSEVIER LICENSE
TERMS AND CONDITIONS

Nov 19, 2020

---

This Agreement between Derek Wolf ("You") and Elsevier ("Elsevier") consists of your license details and the terms and conditions provided by Elsevier and Copyright Clearance Center.

| | |
|---|---|
| License Number | 4952641142549 |
| License date | Nov 19, 2020 |
| Licensed Content Publisher | Elsevier |
| Licensed Content Publication | The Lancet |
| Licensed Content Title | Restoration of reaching and grasping movements through brain-controlled muscle stimulation in a person with tetraplegia: a proof-of-concept demonstration |
| Licensed Content Author | A Bolu Ajiboye,Francis R Willett,Daniel R Young,William D Memberg,Brian A Murphy,Jonathan P Miller,Benjamin L Walter,Jennifer A Sweet,Harry A Hoyen,Michael W Keith,P Hunter Peckham,John D Simeral,John P Donoghue,Leigh R Hochberg,Robert F Kirsch |
| Licensed Content Date | 6–12 May 2017 |
| Licensed Content Volume | 389 |
| Licensed Content Issue | 10081 |
| Licensed Content Pages | 10 |

# APPENDIX

## Copyrights and Permissions



| | |
|---|---|
| Start Page | 1821 |
| End Page | 1830 |
| Type of Use | reuse in a thesis/dissertation |
| Portion | figures/tables/illustrations |
| Number of figures/tables/illustrations | 1 |
| Format | both print and electronic |
| Are you the author of this Elsevier article? | No |
| Will you be translating? | No |
| Title | Achieving practical functional electrical stimulation-driven reaching motions in an individual with tetraplegia |
| Institution name | Cleveland State University |
| Expected presentation date | Dec 2020 |
| Portions | Figure 1 |
| Requestor Location | Derek Wolf<br>1674 Victoria Ave<br><br>LAKEWOOD, OH 44107<br>United States<br>Attn: Derek Wolf |
| Publisher Tax ID | 98-0397604 |
| Total | 0.00 USD |