Holding Static Arm Configurations with Functional Electrical Stimulation: A Case Study

Derek N. Wolf, *Student Member*, *IEEE* and Eric M. Schearer[1], *Member*, *IEEE*

***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.**

I. 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 parameters 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

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.

[1]D. N. Wolf and E. M. Schearer are with the Department of Mechanical Engineering, Cleveland State University and the Cleveland Functional Electrical Stimulation Center, Cleveland, OH USA d.n.wolf@vikes.csuohio.edu, e.schearer@csuohio.edu.

TABLE I
STIMULATION ELECTRODES USED

Electrode Placement	Muscles Targeted	Approximate Function	Type	Current Amplitude (mA)	Max Pulse Width (μs)
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 rotation	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

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.

II. MATERIALS AND 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.

A. 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, 19, 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 μs. 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).

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.

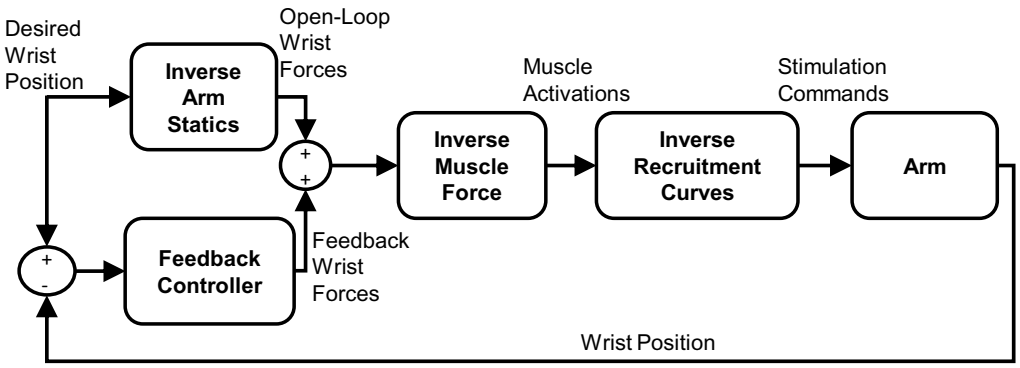

Fig. 1. Controller block diagram

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.

B. 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. 1. 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.

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,

$$\mathbf{f}_{\mathbf{r}_{\mathbf{static}}} = \mathbf{p}(\mathbf{q}) \tag{1}$$

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^{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,

$$\mathbf{f}_{\mathbf{r}_{\mathbf{stim}}} = \mathbf{p}(\mathbf{q}) - \mathbf{R}(\mathbf{q})\boldsymbol{\alpha}, \tag{2}$$

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^{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, $\mathbf{f}_{\mathbf{r}_{\mathbf{static}}}$,

$$\mathbf{R}(\mathbf{q})_{\mathbf{j}} = \mathbf{f}_{\mathbf{r}_{\mathbf{static}}} - \mathbf{f}_{\mathbf{r}_{\mathbf{stim}}\mathbf{j}}. \tag{3}$$

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. 1), 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].

C. 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 II-B which requires the wrist position and orientation as inputs. The controller (Fig. 1) 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. 1) is the desired wrist configuration (position and orientation), $\mathbf{q}_* \in \mathbf{SE(3)}$, 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 (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,

$$\begin{array}{ll} \underset{\boldsymbol{\alpha}}{\text{minimize:}} & ||\boldsymbol{\alpha}||_2^2 \\ \text{subject to:} & \mathbf{R}(\mathbf{q}_*)\boldsymbol{\alpha} = \mathbf{p}(\mathbf{q}_*) \\ & \alpha_i \in [0,1] \quad \forall i \in \{1,2,\ldots,9\} \end{array} . \tag{4}$$

For feedback control, (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,

$$\begin{array}{l} ||\boldsymbol{\alpha}||_2^2 + c_1\|\mathbf{R}(\mathbf{q}_*)\boldsymbol{\alpha} - \mathbf{p}(\mathbf{q}_*)\|_{\mathbf{2}}^{\mathbf{2}} + \mathbf{c_2 K} \\ K = \sum k_i \text{ where } k_i = \begin{cases} \boldsymbol{\alpha}_i^2 \text{ if } \boldsymbol{\alpha}_i < 0 \\ (\boldsymbol{\alpha} - 1)^2 \text{ if } \boldsymbol{\alpha}_i > 1 \\ 0 \text{ if } 0 \leq \boldsymbol{\alpha}_i \leq 1 \end{cases} \end{array} , \tag{5}$$

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 because they produced the same solution as the MATLAB function `quadprog` found for (4).

Equation (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) is found, the recruitment curves are inverted (inverse recruitment curves block of Fig. 1) 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.

D. 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 3×3×3 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) 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).

E. 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

$$r = \sqrt{\chi \lambda_{max}}, \tag{6}$$

where λ_{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, χ, 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.

III. 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. 2-3) along with the numerical results from all trials.

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

Figure 2 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 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. 2 (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 significant improvement in the accuracy of the feedback+

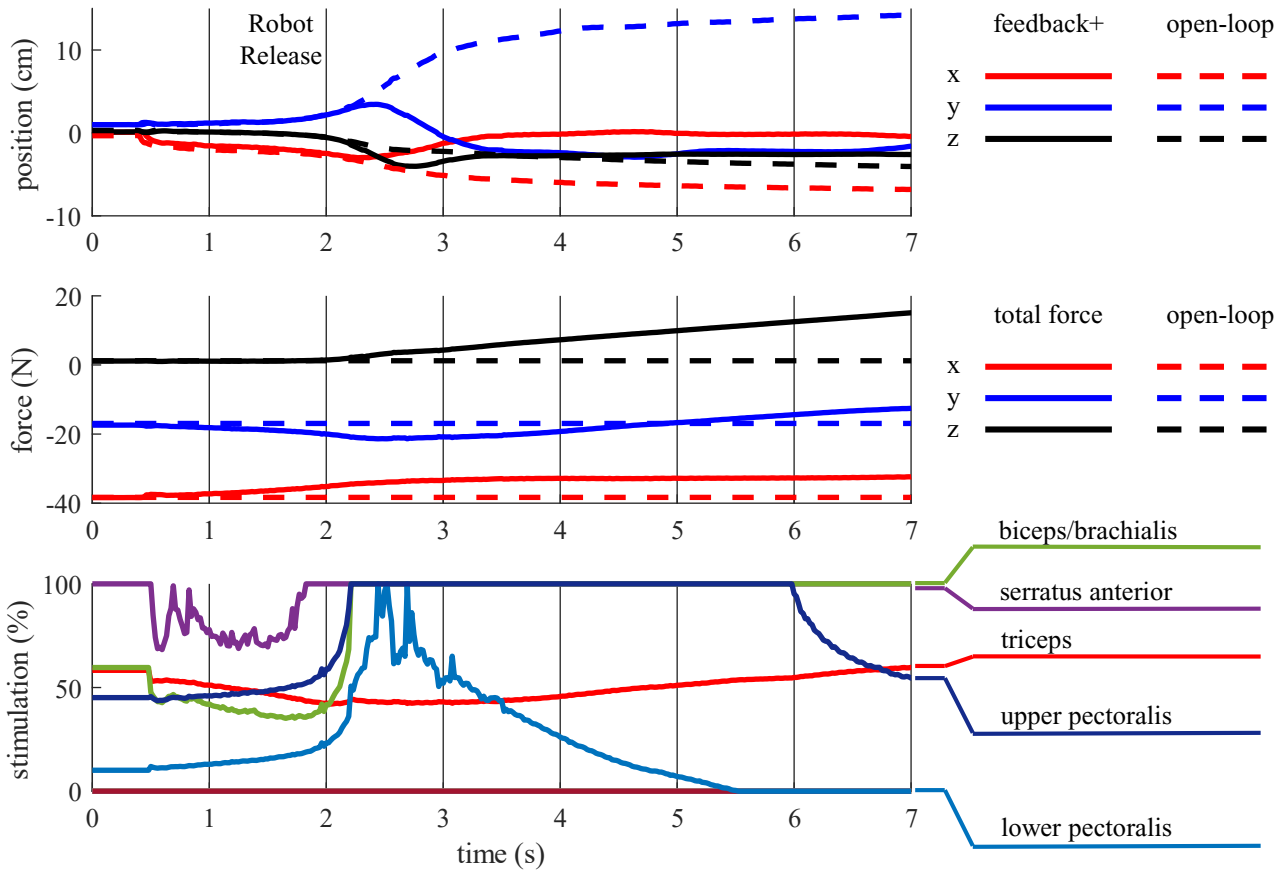

Fig. 2. This figure shows the controller performance during a trial of a representative target. The top plot shows the time response of the position of the wrist (adjusted so the target is at 0) for a single target with the open-loop (dashed lines) and the feedback+ (solid lines) controllers. The middle plot shows the total force input to the inverse muscle force block (see Fig. 1) 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.)

controller compared to the open-loop controller ($p < 0.001$). The example in Fig. 2 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. 3 with an average open-loop accuracy of $10.4\,\mathrm{cm}$ and feedback+ accuracy of $3.7\,\mathrm{cm}$.

The mean precision (standard deviation) for the open-loop controller was $7.7\,\mathrm{cm}$ ($8.7\,\mathrm{cm}$). 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 ($p = 0.13$). Figure 3 shows trials spatially with an open-loop precision of $7.1\,\mathrm{cm}$ and a feedback+ precision of $7.0\,\mathrm{cm}$.

Figure 2 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 ($p = 0.076$).

IV. 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.

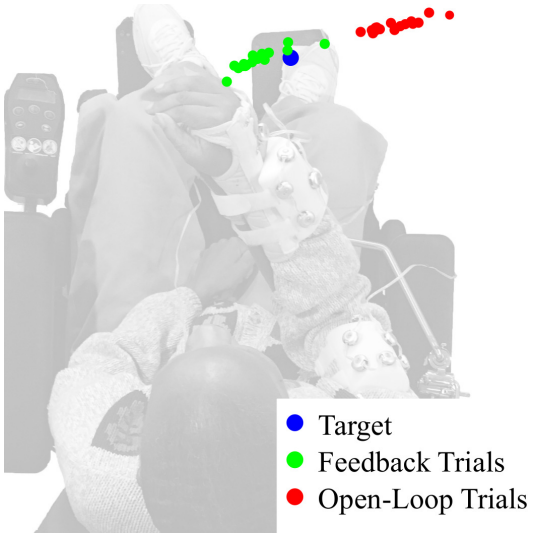

Fig. 3. Representative example showing the final positions for each trial for a single target (blue). The open-loop trials (red) had an average accuracy of 10.4 cm and precision of 7.1 cm. The feedback+ trials (green) had an average accuracy of 3.7 cm and precision of 7.0 cm

The accuracy of 2.9 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 cm, and translating the error to the shoulder joint, this error would result in a wrist position error of 9 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 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 cm was worse than the tracking accuracy of less than 2 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 control. 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.4 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. 2, 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

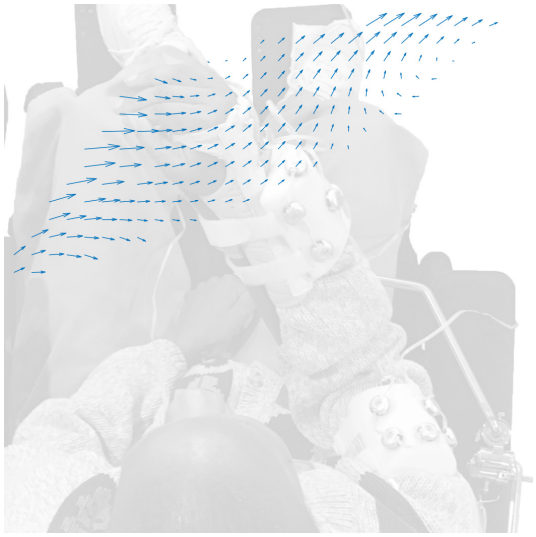

Fig. 4. Image showing the modeled direction of force produced at the wrist by the deltoids throughout the workpspace. The direction of force changes based on the position and orientation of the wrist.

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. 2). 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 para-

lyzed 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.

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*, vol. 21, no. 10, pp. 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*, vol. 34, no. 1, pp. 28–37, 2012.

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

[5] P. H. Peckham and J. S. Knutson, "Functional electrical stimulation for neuromuscular applications," *Annual Reviews in Biomedical Engineering*, vol. 7, pp. 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*, vol. 95, no. 6, pp. 1201–1211, 2014.

[7] K. M. Jagodnik and A. J. van den Bogert, "Optimization and evaluation of a proportional derivative controller for planar arm movement," *Journal of Biomechanics*, vol. 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*, vol. 47, pp. 533–542, 2009.

[9] K. Jagodnik, P. Thomas, A. van den Bogert, M. Branicky, and R. 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," vol. 9, no. 4, pp. 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*, vol. 10, no. 1, pp. 1–20, 2013.

[12] A. B. Ajiboye, F. R. Willett, D. R. Young, W. D. Memberg, B. A. Murphy, J. P. Miller, B. L. Walter, J. A. Sweet, H. A. Hoyen, M. 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*, vol. 389, no. 10081, pp. 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*, vol. PP, no. 99, p. 1, 2014.

[14] M. Kutlu, C. Freeman, A.-M. Hughes, and M. Spraggs, "A Home-based FES System for Upper-limb Stroke Rehabilitation with Iterative Learning Control," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 12 089–12 094, 2017.

[15] 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 *Rehabilitation Robotics (ICORR), 2017 International Conference on*. IEEE, 2017, pp. 789–794.

[16] R. S. Razavian, B. Ghannadi, and J. McPhee, "Feedback control of functional electrical stimulation for arbitrary upper extremity movements," in *Rehabilitation Robotics (ICORR), 2017 International Conference on*. IEEE, 2017, pp. 1451–1456.

[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*, vol. 17, no. 5, pp. 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*, vol. 34, no. 7, pp. 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*, vol. 45, no. 4, pp. 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*, vol. 19, no. 1, pp. 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*, vol. 2, no. 2, pp. 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*, vol. 35, no. 11, pp. 905–916, 1988.

[23] E. M. Schearer, Y.-W. Liao, E. J. Perreault, M. C. Tresch, W. D. Memberg, R. F. Kirsch, and K. 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*, vol. 24, no. 12, pp. 1405–1415, 2016.

[24] G. Wu, F. C. van der Helm, H. E. Veeger, M. Makhsous, P. V. Roy, C. Anglin, J. Nagels, A. R. Karduna, K. McQuade, X. Wang, F. W. Werner, and B. Buchholz, "ISB recommendation on definitions of joint coordinate systems of various joints for the reporting of human joint motion – part II: Shoulder, elbow, wrist and hand," *Journal of Biomechanics*, vol. 38, no. 5, pp. 981–992, 2005.

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

[26] M. Lang and S. Hirche, "Computationally Efficient Rigid-Body Gaussian Process for Motion Dynamics," *IEEE Robotics and Automation Letters*, vol. 2, no. 3, pp. 1601–1608, 2017.

[27] W. K. Durfee and K. E. MacLean, "Methods for estimating isometric recruitment curves of electrically stimulated muscle," *IEEE Transactions on Biomedical Engineering*, vol. 36, no. 7, pp. 654–666, 1989.

[28] A. P. L. Bó, L. O. da Fonseca, and A. C. C. de Sousa, "Fes-induced co-activation of antagonist muscles for upper limb control and disturbance rejection," *Medical Engineering & Physics*, vol. 38, no. 11, pp. 1176–1184, 2016.

[29] M. Ferrarin, F. Palazzo, R. Riener, and J. Quintern, "Model-Based Control of FES-Induced Single Joint Movements," vol. 9, no. 3, pp. 245–257, 2001.

[30] "Identification of electrically stimulated quadriceps muscles in paraplegic subjects," *IEEE Transactions on Biomedical Engineering*, vol. 46, no. 1, pp. 51–61, 1999.

[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*, vol. 23, no. 6, pp. 1095–1105, 2015.