

# Identifying Inverse Human Arm Dynamics Using a Robotic Testbed

Eric M. Schearer, Yu-Wei Liao, Eric J. Perreault, Matthew C. Tresch, William D. Memberg, Robert F. Kirsch,  
and Kevin M. Lynch

**Abstract**—We present a method to experimentally identify the inverse dynamics of a human arm. We drive a person's hand with a robot along smooth reaching trajectories while measuring the motion of the shoulder and elbow joints and the force required to move the hand. We fit a model that predicts the shoulder and elbow joint torques required to achieve a desired arm motion. This torque can be supplied by functional electrical stimulation of muscles to control the arm of a person paralyzed by spinal cord injury. Errors in predictions of the joint torques for a subject without spinal cord injury were less than 20% of the maximum torques observed in the identification experiments. In most cases a semiparametric Gaussian process model predicted joint torques with equal or less error than a nonparametric Gaussian process model or a parametric model.

## I. INTRODUCTION

People living with high spinal cord injuries do not have full voluntary control of their arms. They are limited in daily tasks which require reaching such as opening doors and feeding themselves. Restoration of reaching would allow people with high spinal cord injuries to live more independently. Approximately 273,000 people are living with spinal cord injuries in the United States. Since 2010, 52.2% of spinal cord injury patients discharged from hospitals have a high spinal cord injury, which limits the use of their arms [1].

Functional electrical stimulation (FES) is a promising technology for restoring reaching to people with paralysis. Functional electrical stimulation causes muscles to contract and induce joint movements. Although FES has had success in some applications [2], [3], [4], engineers have not yet exploited the full capability of the musculoskeletal system to perform a wide range of tasks using FES. Complex movements such as reaching require the coordination of

multiple muscles acting across multiple joints of the skeletal system.

FES applications requiring multiple muscles have generally used fixed muscle activation patterns. For instance, the Freehand System<sup>®</sup> [5] provides users control of their hand but does so by having only a few stereotyped stimulation patterns. FES controllers for walking [6] and cycling [7] also use stereotyped movements. While controllers for stereotyped movements have restored some function, there is clearly a need for flexible control strategies that can achieve arbitrary goals subject to the constraints of the musculoskeletal system.

We propose a two-part model-based strategy for flexible control of a paralyzed human arm. The first part is to identify the inverse dynamics of the arm, or more specifically to determine the shoulder and elbow torques needed to induce desired shoulder and elbow joint accelerations for any current set of shoulder and elbow joint positions and velocities. The second part is to determine how to electrically stimulate muscles to produce those torques. The focus of this paper is on the first part: identifying the inverse dynamics of the arm for any specific human subject.

While powerful computer models of the human arm exist [8], [9], these models rely on data from cadavers and even include data from more than one subject. They do not describe a specific FES-controlled system on a living person. Additionally, speed of computation of these computer models is an issue for real-time control. A model identified for a specific subject will be more useful in control than an idealized model developed for a computer simulation.

One challenge in developing an accurate subject-specific model is capturing the complexity of the human arm. Besides the rigid body dynamics that might describe a robot arm, the human arm dynamics include torques manifesting from the stiffness and damping in the muscles, tendons, and skin, among other factors. In addition, individual passive muscle models are difficult to identify because they cannot be isolated without removing the muscles from the skeleton. Finally, people with spinal cord injuries typically use some sort of passive arm support in unison with FES. The dynamics of the arm support must be included when identifying the arm inverse dynamics. Because of all of these factors the human arm dynamics cannot be accurately represented by a model linear in some set of parameters as is often the case with the inverse dynamics of a robot arm [10].

A second challenge in developing an accurate subject-specific model is to generalize over a large space of joint positions, velocities, and accelerations. Black-box function

This work was supported by NSF grant 0932263 and NSF Graduate Fellowship DGE-0824162.

E. M. Schearer and Y. Liao are with the Department of Mechanical Engineering, Northwestern University, Evanston, IL USA {eschearer@northwestern.edu, ywl@northwestern.edu}

E. J. Perreault and M. C. Tresch are with the Department of Biomedical Engineering and Department of Physical Medicine and Rehabilitation, Northwestern University, Evanston, IL USA {e-perreault@northwestern.edu, m-tresch@northwestern.edu}

W. D. Memberg is with the Department of Biomedical Engineering, Case Western Reserve University, Cleveland, OH USA {wdm@case.edu}

R. F. Kirsch is with the Department of Biomedical Engineering, Case Western Reserve University, Louis Stokes Veterans Affairs Medical Center, and the Department of Orthopedics, MetroHealth Medical Center Cleveland, OH USA {rfk3@case.edu}

K. M. Lynch is with the Department of Mechanical Engineering and the Northwestern Institute on Complex Systems, Northwestern University, Evanston, IL USA {kmlynch@northwestern.edu}

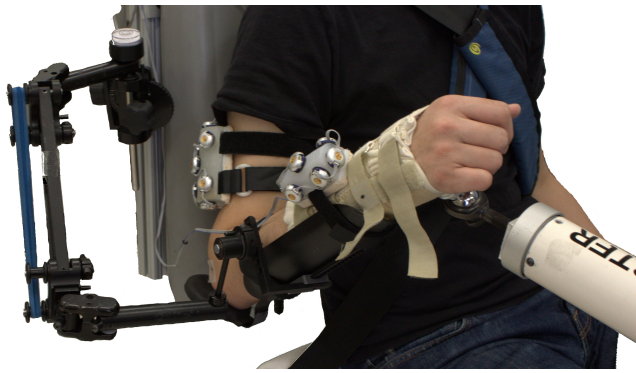


Fig. 1. Instrumentation for experiments. The picture shows placement of the rigid bodies with reflective markers for optical tracking. The mobile arm support is on the left of the picture with dark blue rubber bands on the far left providing supporting force. The subject's hand and distal forearm are in a soft cast that is attached via a magnet to a ball-and-socket joint at the end effector of the HapticMaster robot.

approximators may require a large amount of training data to produce an accurate model. Time is of the essence in identifying models for people with spinal cord injuries. Long training sessions discourage use of assistive devices and introduce system nonstationarity as muscles fatigue.

To address these two challenges we explore the use of Gaussian process regression with encoded global prior information. Gaussian processes have the flexibility to model complex nonlinearities, and they can easily incorporate prior knowledge that can aid in generalizing to unexplored regions of the workspace. Gaussian process regression has been shown to be superior to a parameterized rigid-body-dynamics model and a locally weighted projection regression model for predicting the inverse dynamics of a seven-degree-of-freedom SARCOS robot arm [11]. Incorporating global basis functions into a semiparametric Gaussian process model has been demonstrated to further decrease error in predicting the inverse dynamics of a seven-degree-of-freedom Barrett WAM robot arm [12].

This paper makes two primary contributions. The first is the presentation of an experimental method to identify the inverse dynamics of a human arm by moving a person's hand with a robot while measuring the human arm's shoulder and elbow joint positions, velocities, and accelerations along with the interaction force at the interface of the hand and the robot's end effector. The second is to show the advantages of using a semiparametric Gaussian process model over a purely nonparametric Gaussian process model or a purely parametric model. We quantify the ability of parametric, semiparametric, and nonparametric model structures to generalize in predicting shoulder and elbow torques.

## II. METHODS

To achieve the goals of this study we collected motion and force data from reaching movements of two subjects. One subject has a spinal cord injury, and the other subject does not. Briefly, a subject's hand is placed in a cast attached to the end of a robot. The robot drives the subject's hand along

various smooth reaching trajectories. During the execution of the reaching trajectories an optical system tracks the motion of the subject's arm, while the robot measures the force required to drive the hand along the trajectory. Based on the configuration of the arm, the forces required to drive the hand are transformed into equivalent torques about the shoulder and elbow. Subsection II-A describes the subjects and the test setup for these experiments.

We fit models mapping the motions of the arm, namely the shoulder and elbow joint positions, velocities, and accelerations, to the torques about the shoulder and elbow joints needed to drive the arm motions. Three different model structures were explored: (1) a parametric structure with stiffness and damping terms for each degree of freedom, (2) a nonparametric Gaussian process model, and (3) a semi-parametric model that uses the structure of the parametric model within the framework of a Gaussian process model. Subsection II-B describes these models in detail.

To quantify and compare the effectiveness of each of the three model structures in predicting torques, we computed the leave-one-out cross-validation error for the models in predicting the torque outputs given new inputs to the models. These analyses are described in Subsection II-C.

### A. Experiment Setup and Data Collection

This section describes the collection of data to compute shoulder and elbow joint positions, velocities, accelerations, and torques from experiments with two subjects. The first subject was a 56-year-old woman who sustained a hemisection of the spinal cord at the C1-C2 level from a gunshot wound in 1994. She cannot move her right arm, but she has some sensation and pain hypersensitivity. She experiences hypertonia in some of her arm muscles. Due to hypertonia the range of motion of her right arm is limited. This subject also has a surgically implanted FES device that was not used for this study. More details on this subject are included in [13] (Subject 1). The other subject was a 28-year old male with no history of musculoskeletal disorders. Protocols used for research were approved by the internal review boards at Northwestern University (IRB NO. STU00018382) and MetroHealth Medical Center (IRB NO. 04-00014).

The shoulder elevation plane, shoulder elevation, shoulder internal rotation, and elbow flexion as defined in [14] are referred to as joint positions, and their first and second derivatives are referred to as joint velocities and accelerations. We refer to torques corresponding to these four degrees of freedom as joint torques. The axes of rotation for the shoulder are seen in Fig. 2. Our model does not consider elbow pronation as it does not affect the position of the wrist, which is ultimately what we are trying to control.

A HapticMaster (Moog FCS) robot with three degrees of freedom moved the subject's hand along smooth reaching trajectories. The robot reports the 3D position of its end effector based on encoder data, and the robot has a three-axis force sensor at its end effector. The subject's hand was placed in a soft cast, and the cast was attached by a magnet to a ball-and-socket joint on a tie-rod end bolted to the robot's

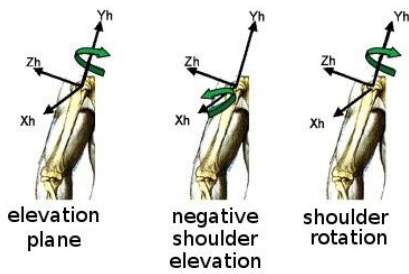


Fig. 2. The shoulder rotations are Euler angles with a YXY order. The shoulder elevation plane is the rotation about the Yh axis. Shoulder elevation is rotation about the resulting Xh axis. Shoulder rotation is rotation about the resulting Zh axis. This figure is adapted from [14] with the permission of the authors.

end effector as seen on the right in Fig. 1. The magnet acted as a safety mechanism, releasing the subject's hand if the robot applied more force than the magnet could hold.

Using the output from the robot's force sensor we compute the corresponding torques about each subject's shoulder and elbow joints. A force  $\mathbf{f}_m \in \mathbb{R}^3$  is measured at the robot's end effector and is transmitted through the ball-and-socket joint. The force at the ball-and-socket joint is transformed to a wrench  $\mathbf{f}_r \in \mathbb{R}^6$  at the subject's wrist. The moment is the cross-product of the vector  $\mathbf{r}_{wb} \in \mathbb{R}^3$  from the wrist to the ball-and-socket joint and the force at the ball-and-socket joint  $\mathbf{f}_m$ , so  $\mathbf{f}_r = [\mathbf{f}_m \quad \mathbf{r}_{wb} \times \mathbf{f}_m]^\top$ . The corresponding torques about the shoulder and elbow joints are  $\boldsymbol{\tau}_r = \mathbf{J}_r^\top(\mathbf{q})\mathbf{f}_r \in \mathbb{R}^4$  where  $\mathbf{J}_r \in \mathbb{R}^{6 \times 4}$  is the kinematic Jacobian of the subject's arm at the wrist.

To track the movement of the subject's arm we collected sensor measurements from the robot and from an Optotrak Certus Motion Capture System (Northern Digital, Inc.). The robot reported the position of its end effector. We fixed rigid bodies with reflective markers to the subject's chair, upper arm, and forearm (Fig. 1). The coordinate system of the chair rigid body was aligned with the subject's thorax. The motion capture system recorded the position and orientation of each rigid body with respect to a global coordinate system.

We estimated the shoulder and elbow joint positions, velocities, and accelerations offline using an extended Rauch-Tung-Striebel (RTS) smoother, which is essentially an extended Kalman filter run forward and backward. We implemented the RTS smoother by adding smoothing to the extended Kalman filter method and MATLAB<sup>®</sup> code from [15]. This method combines sensor information and a simple dynamic model of the arm's motion. It takes into account errors in marker placement, errors from assumptions of the arm dynamics, sensor noise, and marker movement during arm motion. Under these conditions the method performs the locally optimal estimate of the joint positions, velocities, and accelerations.

During a reaching trajectory the subject's hand moved in a straight line from a home position to a target position, stayed at the target position for one second, and then returned to the home position (Fig. 3). The position  $s$  along the

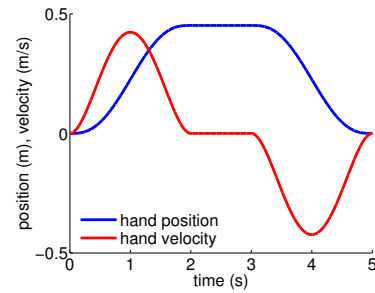


Fig. 3. Plot of hand position and hand velocity vs. time for the reaching trial with the longest distance travelled for the subject without SCI.

straight line was a fifth-order polynomial in time, so  $s = c_0 + c_1t + c_2t^2 + c_3t^3 + c_4t^4 + c_5t^5$ . A unique solution for the coefficients  $c_0, \dots, c_5$  can be found given the total time to complete the trajectory, the total spatial length of the trajectory, and zero initial and final velocity and acceleration. The polynomial trajectories ensured smooth positions, velocities, and accelerations. The two home positions were in front of the subject's mouth and in front of and just below the subject's shoulder. Target positions were determined by moving the subject's hand as high and to the right as possible, as high and to the left as possible, as low and to the right as possible, and as low and to the left as possible. For the subject without SCI these four extremes were limited by the workspace of the mobile arm support or of the robot. For the subject with SCI these four extremes were limited by the range of motion of her arm.

The robot moved the hand of the subject without spinal cord injury to nine target positions. The subject was told to relax his arm during the reaching movements. A planar three-by-three grid of target positions was constructed with the four extreme targets at the corners. The distance from each home position to each target position varied from 12 cm to 45 cm, and the time to reach from home to target was a constant two seconds, so the maximum speed of the hand during a single trajectory varied from 11 cm/s to 42 cm/s. Each home position and target position combination was run three times for a total of  $2 \times 9 \times 3 = 54$  reaching trajectories.

The robot moved the hand of the subject with spinal cord injury to a smaller target set at slower speeds and in a more limited workspace. Her hand was moved to four limit locations as with the other subject, and these locations were used as targets. Four randomly selected trajectories were run and repeated three times for a total of 12 trajectories. She completed trajectories starting at the mouth to high left, high right, and low right targets and a trajectory starting in front of the shoulder to the low right target. The distance from each home position to each target position varied from 16 cm to 36 cm, and the time to reach from home to target was a constant three seconds, so the maximum speed of the hand during a single trajectory varied from 10 cm/s to 23 cm/s.

A mobile arm support supported each subject's arm against gravity. The mobile arm support, pictured on the left of Fig. 1, applies force at the subject's forearm and at the end

of the humerus. The force is primarily a vertical force, but also has components in the horizontal plane. Arm supports are often needed by people with spinal cord injury because muscle atrophy leaves muscles too weak to lift the arm against gravity. FES is sufficient to move the arm given that the arm support compensates for gravity.

### B. Inverse Dynamics Models

The equations of motion for the arm when no muscles are activated are

$$B(\mathbf{q})\ddot{\mathbf{q}} + C(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) + \boldsymbol{\tau}(\mathbf{q}, \dot{\mathbf{q}}) = J_r^\top(\mathbf{q})\mathbf{f}_r + \boldsymbol{\tau}_s, \quad (1)$$

where  $\mathbf{q} \in \mathbb{R}^4$  is a vector of joint angles,  $\dot{\mathbf{q}} \in \mathbb{R}^4$  is a vector of joint velocities, and  $\ddot{\mathbf{q}} \in \mathbb{R}^4$  is a vector of joint accelerations. The four joint angles are the elevation plane, shoulder elevation, shoulder rotation, and elbow flexion described in Subsection II-A. The inertia matrix is  $B(\mathbf{q}) \in \mathbb{R}^{4 \times 4}$ , the matrix of Coriolis and centrifugal terms is  $C(\mathbf{q}, \dot{\mathbf{q}}) \in \mathbb{R}^{4 \times 4}$ , and the vector of gravity terms is  $\mathbf{g}(\mathbf{q}) \in \mathbb{R}^4$ . The vector of torques due to the passive stiffness and damping of the muscles is  $\boldsymbol{\tau}(\mathbf{q}, \dot{\mathbf{q}}) \in \mathbb{R}^4$ . The first term on the right-hand side of (1) is the vector of shoulder and elbow torques  $\boldsymbol{\tau}_r = J_r^\top(\mathbf{q})\mathbf{f}_r \in \mathbb{R}^4$  corresponding to the wrench applied to the subject's wrist by the robot. The second term on the right-hand side of (1) is the vector of shoulder and elbow torques  $\boldsymbol{\tau}_s \in \mathbb{R}^4$  corresponding to the forces applied by the mobile arm support.

We can compute the torques applied to the arm by the robot based on measurements of the shoulder and elbow joint angles and the wrench applied by the robot to the subject's wrist as described in Section II-A. The input to our models is  $\mathbf{x} = [\mathbf{q}^\top, \dot{\mathbf{q}}^\top, \ddot{\mathbf{q}}^\top]^\top$ . We rewrite (1) as

$$\boldsymbol{\tau}_r = \mathbf{p}(\mathbf{x}) = B(\mathbf{q})\ddot{\mathbf{q}} + C(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) + \boldsymbol{\tau}(\mathbf{q}, \dot{\mathbf{q}}) - \boldsymbol{\tau}_s, \quad (2)$$

where  $\mathbf{p}(\mathbf{x})$  represents the torques that must be applied to the shoulder and elbow joints to produce desired joint accelerations given the current joint positions and velocities. The goal is to identify a model of  $\mathbf{p}(\mathbf{x})$  given a series of joint positions, velocities, and accelerations and forces applied by the robot to move the subject's hand. We do not explicitly model any of the individual terms on the right-hand side of (2).

1) *Parametric Model*: A model that is linear in its parameters is preferred because we can use linear regression to identify the model and avoid solving a nonlinear optimization problem. A full parametric model might include a typical parameterization of  $B(\mathbf{q})\ddot{\mathbf{q}}$ ,  $C(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}}$ , and  $\mathbf{g}(\mathbf{q})$  in (2) based on the mass, center of mass, and moments of inertia of the humerus and forearm. The contribution of passive muscle forces  $\boldsymbol{\tau}(\mathbf{q}, \dot{\mathbf{q}})$  might be modeled by a function of polynomials of the joint positions and velocities [16] that is linear in its parameters. The contribution from the mobile arm support  $\boldsymbol{\tau}_s$  can be modeled as a linear function of the stiffness of the rubber bands of the arm support and their slack length.

After considering this full parametric model we found

more success with a simpler parametric model where the torque about a single degree of freedom  $p_p(\mathbf{x})$  is dominated by joint stiffness and damping as in

$$p_p(\mathbf{x}) = \mathbf{h}(\mathbf{x})^\top \boldsymbol{\beta} = k_s(\theta - \theta_0) + k_d\dot{\theta}, \quad (3)$$

where  $\theta$  and  $\dot{\theta}$  are the position and velocity of the given joint,  $\theta_0$  is the equilibrium position of the given joint, and  $k_s$  and  $k_d$  are the joint stiffness and damping. Here  $\boldsymbol{\beta} = [-k_s\theta_0 \quad k_s \quad k_d]^\top$  is a column vector of parameters, and  $\mathbf{h}(\mathbf{x}) = [1 \quad \theta \quad \dot{\theta}]^\top$  is a column vector of basis functions. The joint equilibrium position is the position where the joint rests when muscles are relaxed and no external forces are applied. It depends on the passive stiffness of the muscles and the stiffness of the arm support.

The transition from the full parametric model with coupled degrees of freedom in (2) to the simpler single-joint model in (3) is based on a number of observations. First because the reaching motions are generally slow, the inertial, Coriolis, and centrifugal terms in (2) are small compared to the gravity terms along with the passive stiffness and damping of the muscles and arm support. Second, the arm support essentially cancels out gravity, but still provides stiffness in the horizontal plane. This leaves only stiffness and damping as captured in (2).

We use linear regression to estimate the vector of parameters  $\boldsymbol{\beta}$  and predict the torque given a new test input  $\mathbf{x}_*$ . Gathering data from smooth reaching trajectories we construct the matrix  $H \in \mathbb{R}^{n \times 3}$  where each row is  $\mathbf{h}(\mathbf{x})^\top$  evaluated at each training input, and  $\mathbf{y}$  is a column vector of  $n$  training outputs. From linear regression the estimate of the parameters in (3) is  $\hat{\boldsymbol{\beta}} = (H^\top H)^{-1} H^\top \mathbf{y}$ . After constructing the vector of basis functions  $\mathbf{h}(\mathbf{x}_*)$  evaluated at a test input  $\mathbf{x}_*$  the prediction  $p_p(\mathbf{x}_*)$  of the parametric model given a test input  $\mathbf{x}_*$  is

$$p_p(\mathbf{x}_*) = \mathbf{h}(\mathbf{x}_*)^\top \hat{\boldsymbol{\beta}}. \quad (4)$$

The variance of the prediction of the parametric model is  $v_p(\mathbf{x}_*) = \sigma_n^2 (1 + \mathbf{h}(\mathbf{x}_*)^\top (H^\top H)^{-1} \mathbf{h}(\mathbf{x}_*))$ , where  $\sigma_n^2$  is the variance in the torque output.

2) *Nonparametric Model*: The second of the three models is a purely nonparametric Gaussian process model. A Gaussian process is a collection of random variables, any finite number of which have a joint Gaussian distribution [11]. An infinite collection of random variables amounts to a random function, so a Gaussian process can be viewed as a distribution of random functions.

A Gaussian process is defined by its mean function  $m(\mathbf{x})$  and its covariance function  $k(\mathbf{x}, \mathbf{x}')$ . The covariance function defines how much the output of a random function at input  $\mathbf{x}$  depends on the output of the random function at input  $\mathbf{x}'$ . A covariance function is defined by its vertical scale and a length scale, which are referred to as hyperparameters. The vertical scale defines how much a particular function drawn from the distribution can deviate from the mean function, and the length scale defines how much the function output can change with a change in the input. A small length scale

allows for large changes, and a large length scale only allows for small changes in the output given a change in the input.

The nonparametric model has a Gaussian process prior distribution with zero mean and a squared-exponential covariance function,

$$p_n(\mathbf{x}) \sim \mathcal{GP}(\mathbf{0}, k(\mathbf{x}, \mathbf{x}')). \quad (5)$$

The hyperparameters of the covariance function are found by maximizing the marginal likelihood or evidence of the training data, which is a standard technique for model selection in Gaussian process regression [11]. Maximizing the marginal likelihood balances training error and model complexity.

Inference with Gaussian processes amounts to assuming a prior distribution of random functions, collecting training samples, and then choosing those functions from the prior distribution that best agree with the training samples. The remaining functions make up the posterior distribution which is also a Gaussian process.

The prediction  $p_n(\mathbf{x}_*)$  of the nonparametric model given a test input  $\mathbf{x}_*$  is just the mean of the posterior Gaussian process evaluated at the test input

$$p_n(\mathbf{x}_*) = \mathbf{k}_*^\top (K + \sigma_n^2 I)^{-1} \mathbf{y} \quad (6)$$

where  $\mathbf{y}$  is a column vector of  $n$  training outputs,  $\mathbf{k}_* \in \mathbb{R}^n$  is a vector where the  $i^{\text{th}}$  element is the covariance function evaluated at the test input and the  $i^{\text{th}}$  training input, and  $K \in \mathbb{R}^{n \times n}$  is a matrix of the covariance function evaluated at all combinations of training inputs. Recall that the model inputs are the joint positions, velocities, and accelerations of the arm, and the outputs are the corresponding joint torques. The variance of the prediction of the nonparametric model is  $v_n(\mathbf{x}_*) = k(\mathbf{x}_*, \mathbf{x}_*) - \mathbf{k}_*^\top (K + \sigma_n^2 I)^{-1} \mathbf{k}_*$ .

3) *Semiparametric Model:* The last of the three models is a semiparametric Gaussian process model which incorporates into a Gaussian process model knowledge of the arm dynamics encoded in a parametric model. The semiparametric model uses its locally relevant Gaussian process component to predict complex dynamics near training data and its globally relevant parametric part to generalize to places in the input space that are far from the training data.

The semiparametric model is a Gaussian process where the mean function is the parametric model in (3) and the covariance function is the sum of a squared-exponential covariance function and a term that takes into account the uncertainty in the parameters of the linear model,

$$p_s(\mathbf{x}) \sim \mathcal{GP}(\mathbf{h}(\mathbf{x})^\top \mathbf{b}, k(\mathbf{x}, \mathbf{x}') + \mathbf{h}(\mathbf{x})^\top B \mathbf{h}(\mathbf{x})). \quad (7)$$

This is the semiparametric Gaussian process form proposed in [17]. It makes the prior assumption that the parameters  $\beta$  of the linear model are distributed normally with mean  $\mathbf{b} \in \mathbb{R}^k$  and covariance  $B \in \mathbb{R}^{k \times k}$ . The vector of basis functions  $\mathbf{h}(\mathbf{x})^\top$  is the same as in (3). We use the mean and covariance of the parameters found from fitting the parametric model to a data set to define the prior parameter

distribution for the semiparametric model.

The vertical scale hyperparameter of the squared-exponential covariance function  $k(\mathbf{x}, \mathbf{x}')$  was found by maximizing the marginal likelihood or evidence of the training data while the length scale parameters were chosen a priori to be small to allow the nonparametric part of the model to dominate in making predictions for test inputs close to the training inputs and the parametric part of the model to dominate for test inputs that are not close to the training inputs.

The prediction  $p_s(\mathbf{x}_*)$  of the semiparametric model given a test input  $\mathbf{x}_*$  is the mean of the posterior Gaussian process evaluated at the test input

$$p_s(\mathbf{x}_*) = \mathbf{h}(\mathbf{x}_*)^\top \bar{\beta} + \mathbf{k}_*^\top K_y^{-1} (\mathbf{y} - H^\top \bar{\beta}), \quad (8)$$

where  $K_y = K + \sigma_n^2 I$ . The posterior estimate of the parameters of the linear model is  $\bar{\beta} = (B^{-1} + H K_y^{-1} H^\top)^{-1} (H K_y^{-1} \mathbf{y} + B^{-1} \mathbf{b})$ . The first term on the right-hand side of (8) is the predicted output of the parameterized linear model. The second term is the prediction of the residuals of the parameterized linear model, and its magnitude depends on how close the test input is to the training inputs. Note that the terms  $K_y$  and  $\mathbf{k}_*$  for the semiparametric model will be different in general than those terms for the nonparametric model. The variance of the prediction of the semiparametric model is  $v_s(\mathbf{x}_*) = k(\mathbf{x}_*, \mathbf{x}_*) - \mathbf{k}_*^\top K_y^{-1} \mathbf{k}_* + (\mathbf{h}(\mathbf{x}_*) - H K_y^{-1} \mathbf{k}_*)^\top (B^{-1} + H K_y^{-1} H^\top)^{-1} (\mathbf{h}(\mathbf{x}_*) - H K_y^{-1} \mathbf{k}_*)$ .

### C. Analysis of Model Learning

To compare the ability of the three models to generalize to new test data, we computed the leave-one-out cross-validation (LOOCV) error for each model. Each model was identified on a training data set and then made predictions of torque outputs given inputs from a test data set, which the model did not use for training. The root mean square (RMS) error of the model's predictions of the outputs from the test data set is the generalization error.

For the subject without spinal cord injury each model was trained on each of 18 training sets and tested on each of 18 corresponding test sets. The subject completed 18 unique trajectories three times, so a training set included 17 of these trajectories, and a test set included the trajectory that was left out of the training set. Each of the 18 different training sets left out a different trajectory. For the subject with spinal cord injury there were four training sets each including three unique trajectories with each corresponding test set comprised of the left out trajectory.

## III. RESULTS

Joint angles and torques from one trial for the subject with spinal cord injury are shown in Fig. 4 along with the semiparametric model's predictions of the torques. The actual torques are almost always captured by the confidence intervals around the model's predictions. The confidence intervals also show the typical variation in the torque from



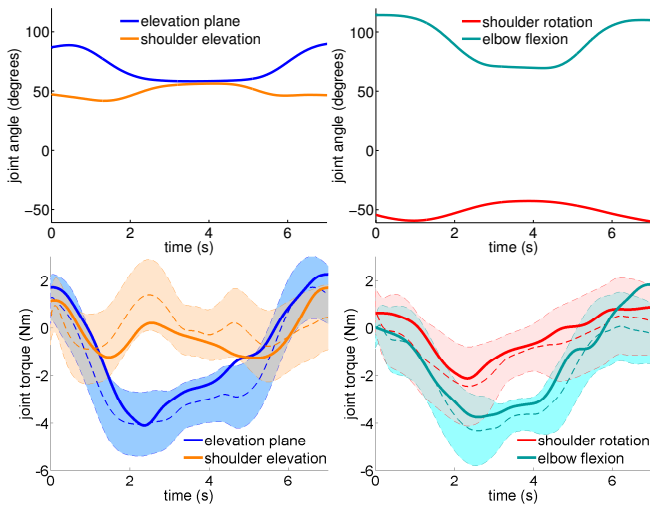


Fig. 4. Filtered joint angles and joint torques for one of the 12 arm trajectories for the subject with spinal cord injury. The thick solid lines represent the actual joint angle or torque from the experiment. The thick dashed lines represent the prediction of the semiparametric model that was trained with the nine trajectories that were different from the trajectory pictured. The shaded regions represent the 95% confidence intervals.

one of three repeated trials to the next given the same inputs. In general the range of joint angles was smaller and the torques required to move the arm were larger for the subject with spinal cord injury, as that subject has a smaller range of motion but more passive muscle stiffness due to hypertonia. Also note that the torques vary mainly with the corresponding joint angle, so the stiffness term in the parametric model (3) plays a significant role.

Stiffness was the dominant source of passive torque for both subjects (Table I). The subject with spinal cord injury had more passive stiffness and damping than did the subject without spinal cord injury. The larger stiffness coefficients for the subject with spinal cord injury were due both to larger passive stiffness in the muscles and more rubber bands on the arm support to support a heavier arm. There was almost no damping in the subject without spinal cord injury. Note that coefficients for shoulder rotation are not reported in Table I because they were not significantly greater than zero.

The semiparametric model predicted joint torques for a subject without spinal cord injury generally within 20% of the maximum torque required to drive the subject's hand (Fig. 5(a)). The exception is an outlier for shoulder rotation of just under 40%. The nonparametric model performed similarly, but with more outliers with large errors. The generalization error was found by training a model on all but one set of trajectories and testing that model on the set that was left out. Note that the training error (see solid circles in Fig. 5) was almost always smaller than the median generalization error for the nonparametric and semiparametric models although the training error was inside the 95% confidence region for the semiparametric model. This suggests that gathering more training data could improve the predictions.

The predictions of torques were not as good for the subject

TABLE I  
PARAMETRIC MODEL COEFFICIENTS

	$\theta_0$ (deg)	$k_s$ (N-m/deg)	$k_d$ (N-m/deg/s)
SCI elevation plane	82	0.13	0.070
non-SCI elevation plane	14	0.01	0.007
SCI shoulder elevation	49	0.12	0.064
non-SCI shoulder elevation	17	0.10	0.003
SCI elbow flexion	116	0.08	0.051
non-SCI elbow flexion	81	0.04	0.009

with spinal cord injury (Fig. 5(b)). Errors were between 10% and 40% for the semiparametric model. We attribute much of the additional error for the subject with spinal cord injury to the smaller identification data set which included 12 trajectories compared to 54 trajectories for the subject without spinal cord injury.

The superiority of the semiparametric model is illustrated in its performance on the more sparse data set of the subject with spinal cord injury. When all 18 trajectories were completed as with the subject without spinal cord injury the semiparametric model predicted torques as well as the nonparametric model and better than the parametric model (Fig. 5(a)). This shows that when presented with a set of test inputs similar to the training data the nonparametric and semiparametric models do very well because of the flexibility of the Gaussian process. When presented with a set of test inputs that are unlike the training data as was the case with the subject with spinal cord injury, the nonparametric model does poorly (see blue boxes in Fig. 5(b)) while the semiparametric model that has the parametric model encoded a priori does much better in predicting torques (see green boxes in Fig. 5(b)).

#### IV. DISCUSSION

We presented a method for identifying the inverse dynamics of any specific subject's arm and quantified the error in the method's predictions of joint torques. We also showed the advantage of using the semiparametric model structure over a pure nonparametric structure. The semiparametric structure includes prior information about the dynamics encoded in a linear model. This advantage is most evident when there are small data sets available and a model must generalize to other parts of the workspace of the arm. When working with people with spinal cord injuries gathering large amounts of system identification data is not feasible.

The method presented in this paper is critical in the design of flexible controllers for functional electrical stimulation. It allows us to use a computed-torque-type controller where we predict the joint torques required to produce desired joint accelerations given the current joint positions and velocities. Now we can specify a hand trajectory, compute the corresponding trajectory in joint space, and then determine the joint torques required to execute the trajectory. This work is a part of a larger study that also characterizes the ability of muscles controlled by functional electrical stimulation to

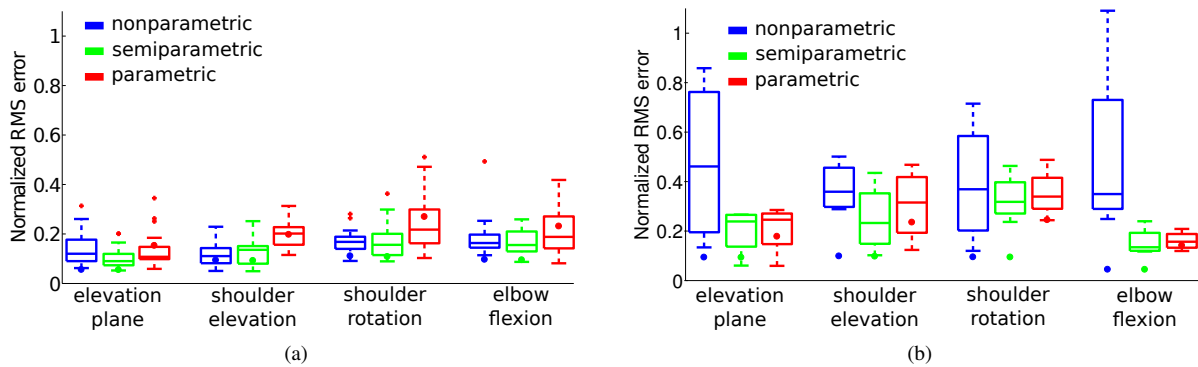


Fig. 5. Box plots for subject without spinal cord injury (a) and for subject with spinal cord injury (b). The horizontal lines inside the boxes represent the median RMS LOOCV errors of the predictions of the three models over the 18 unique test trajectories. The RMS torque error for each degree of freedom was normalized by the maximum torque magnitude for that degree of freedom over the entire data set for the subject. The boxes represent the 25%-75% confidence intervals, and the whiskers represent 5%-95% confidence intervals. The red crosses are the RMS LOOCV errors for outliers. The solid circles represent the mean training error for each model. Each set of three box plots is for an individual degree of freedom.

produce these desired torques and control the motion of the arm of a person with a high spinal cord injury.

One source of error in our torque predictions is the size of the data sets considered here. While this can be addressed with longer test sessions with people without spinal cord injuries, this is simply not possible when working with people with spinal cord injuries due to fatigue and other conditions related to their injuries. Small data sets are a fact of life, and the use of the semiparametric models to combine the flexibility of the Gaussian process with the generalization power of the parametric model is a direct response to this fact.

Another source of error is the limitation in accurate predictions of joint kinematics with optical tracking systems. Motion capture markers move relative to the skeleton in experiments especially as muscles are activated. Velocity and acceleration of joints are difficult to measure directly. Progress can be made in these tracking systems. We also plan in the future to incorporate uncertainties to the inputs of our model, which are the estimates of joint positions, velocities, and accelerations, into the Gaussian process framework.

## REFERENCES

- [1] "Spinal cord injury facts and figures at a glance," National Spinal Cord Injury Statistical Center, 2013.
- [2] 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.
- [3] P. H. Peckham and J. S. Knutson, "Functional electrical stimulation for neuromuscular applications," *Annual Reviews in Biomedical Engineering*, vol. 7, pp. 327–360, 2005.
- [4] L. R. Sheffler and J. Chae, "Neuromuscular electrical stimulation in neurorehabilitation," *Muscle Nerve*, vol. 35, no. 5, pp. 562–590, 2007.
- [5] 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.
- [6] D. Zhang and K. Zhu, "Modeling biological motor control for human locomotion with functional electrical stimulation," *Biological Cybernetics*, vol. 96, pp. 70–97, 2007.
- [7] P. Li, Z. Hou, F. Zhang, M. Tan, H. Wang, Y. Hong, and J. Zhang, "An FES cycling control system based on CPG," in *31st Annual International Conference of the IEEE EMBS*, 2009, pp. 1569–1572.
- [8] D. Blana, J. G. Hincapie, E. K. Chadwick, and R. F. Kirsch, "A musculoskeletal model of the upper extremity for use in the development of neuroprosthetic systems," *Journal of Biomechanics*, vol. 41, pp. 1714–1721, 2008.
- [9] K. R. S. Holzbaur, W. M. Murray, and S. L. Delp, "A model of the upper extremity for stimulating musculoskeletal surgery and analyzing neuromuscular control," *Annals of Biomedical Engineering*, vol. 33, no. 6, pp. 829–840, 2005.
- [10] C. G. Atkeson, C. H. An, and J. M. Hollerbach, "Estimation of inertial parameters of manipulator loads and links," *International Journal of Robotics Research*, vol. 5, no. 3, pp. 101–119, 1986.
- [11] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*. Cambridge, MA: The MIT Press, 2006.
- [12] D. Nguyen-Tuong and J. Peters, "Using model knowledge for learning inverse dynamics," in *International Conference on Robotics and Automation*, 2010.
- [13] 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.
- [14] 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.
- [15] E. Todorov, "Probabilistic inference of multijoint movements, skeletal parameters and marker attachments from diverse motion capture data," *IEEE Transactions on Biomedical Engineering*, vol. 54, no. 11, pp. 1927–1939, 2007.
- [16] C. T. Freeman, A. M. Hughes, J. H. Burridge, P. H. Chappell, P. L. Lewin, and E. Rogers, "A model of the upper extremity using FES for stroke rehabilitation," *Journal of Biomechanical Engineering*, vol. 131, no. 3, pp. 1–12, 2009.
- [17] A. O'Hagan, "Curve fitting and optimal design for prediction," *Journal of the Royal Statistical Society*, vol. 40, pp. 1–42, 1978.