An Active Learning Algorithm for Control of Epidural Electrostimulation

Thomas A. Desautels, Jaehoon Choe*, Parag Gad*, Mandheerej S. Nandra, Roland R. Roy, Hui Zhong, Yu-Chong Tai, V. Reggie Edgerton, Joel W. Burdick

***Abstract*—Epidural electrostimulation has shown promise for spinal cord injury therapy. However, finding effective stimuli on the multi-electrode stimulating arrays employed requires a laborious manual search of a vast space for each patient. Widespread clinical application of these techniques would be greatly facilitated by an autonomous, algorithmic system which choses stimuli to simultaneously deliver effective therapy and explore this space. We propose a method based on GP-BUCB, a Gaussian process bandit algorithm. In $n = 4$ spinally transected rats, we implant epidural electrode arrays and examine the algorithm's performance in selecting bipolar stimuli to elicit specified muscle responses. These responses are compared with temporally interleaved, intra-animal stimulus selections by a human expert. GP-BUCB successfully controlled the spinal electrostimulation preparation in 37 testing sessions, selecting 670 stimuli. These sessions included sustained, autonomous operations (10 session duration). Delivered performance with respect to the specified metric was as good as or better than that of the human expert. Despite receiving no information as to anatomically likely locations of effective stimuli, GP-BUCB also consistently discovered such a pattern. Further, GP-BUCB was able to extrapolate from previous sessions' results to make predictions about performance in new testing sessions, while remaining sufficiently flexible to capture temporal variability. These results provide validation for applying automated stimulus selection methods to the problem of spinal cord injury therapy.**

***Index Terms*—Implants, Learning Automata, Neural Engineering, Neuromuscular Stimulation, Spinal Cord Injury.**

I. Introduction

Animal [1] and human [2], [3] studies have shown that multi-electrode epidural spinal stimulation can provide significant recovery of motor and autonomic function in subjects suffering from severe spinal cord injury (SCI). Fig. 1 shows a 27-electrode stimulating array we employ in rat SCI models. When implanted in the epidural space, the application of appropriate spatio-temporal patterns of electrical stimulation to these arrays can facilitate the operation of spinal circuits caudal to the spinal injury, enabling beneficial motor function, and the recovery of some voluntary control and autonomic function [2], [3].

Multi-electrode stimulating arrays provide many benefits over simple wire electrodes, such as patient-customized stimulation parameters, compensation for errors in surgical placement of the array, and potential adaptation to spinal cord plasticity. Moreover, this adaptivity is needed in the clinic: optimal stimulus patterns vary substantially across patients [3] and over time for each patient due to spinal cord plasticity. Hence, the clinical process of deploying multi-electrode epidural stimulation requires a burdensome search for the optimal stimulation paradigm for each patient.

The space of possible stimuli which must be searched to find high-performing electrode combinations can be vast; e.g., the 19 parameters (which describe active electrode selection, electrode polarity, voltage, stimulus frequency, and pulse width) which can be varied on the 16-electrode array used in [2], [3] allow for 10^8 different stimulus combinations. When restricted to pulse-like stimuli (parametrized by amplitude, frequency, phase, and pulse width), the array shown in Fig. 1 presents a 108-dimensional space that must be searched. Even though the range of permissible values is greatly reduced by safety and design factors, the necessary search for the small set of high-performing or optimal parameters can be tedious and of limited direct therapeutic value.

This burden suggests the need for automated algorithms to facilitate, simplify, and accelerate the search process. Other factors motivate the need for automated search techniques: (1) different stimulation patterns are needed to facilitate different motor behaviors (e.g., standing vs. stepping); (2) the current labor-intensive process of finding optimal stimulation patterns is expensive and may not result in optimal parameter choices; (3) no method yet exists to predictively account for plasticity in the nervous system during locomotor recovery after severe paralysis; and (4) a systematic method can facilitate widespread, cost-effective distribution of this therapy, as there are currently few therapists trained to optimize epidural stimulation for recovery after an SCI.

This paper introduces a novel algorithm which can automatically optimize multi-electrode array stimulation parameters.[1]

Submitted December 15, 2014. This work was supported by NIH U01EB15521, R01EB007615, the Leona M. and Harry B. Helmsley Charitable Trust, and the Christopher and Dana Reeve, Broccoli, Walkabout, and F. M. Kirby Foundations. J. Choe and P. Gad contributed equally.

T. A. Desautels was with the California Institute of Technology, Pasadena, CA 91125, USA. He is now with the Gatsby Computational Neuroscience Unit, University College London, London WC1N 3AR UK (email: tdesautels@gatsby.ucl.ac.uk).

J. Choe was and P. Gad, R. R. Roy, H. Zhong, and V. R. Edgerton are with the University of California, Los Angeles, Los Angeles, CA, USA. J. Choe is now with HRL Laboratories, LLC, Malibu, CA, USA.

M. S. Nandra was and Y.-C. Tai and J. W. Burdick are with the California Institute of Technology, Pasadena, CA, USA.

VRE, RRR, and JWB hold shareholder interest in NeuroRecovery Technologies (NRT). VRE is also the President and Chairman of the Board. VRE, RRR, and JWB hold certain inventorship rights on intellectual property licensed by The Regents of the University of California to NRT and its subsidiaries. A patent has been submitted covering concepts described here.

[1] A preliminary version of this work has been reported [4].

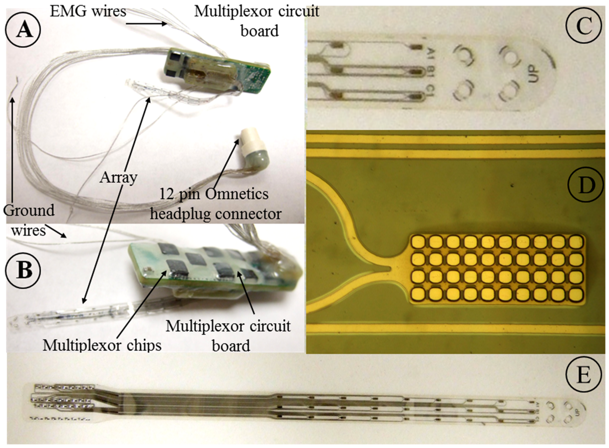

A
EMG wires
Multiplexor circuit board
Array
12 pin Omnetics headplug connector
Ground wires
B
Multiplexor chips
Multiplexor circuit board
C
UP
D
E

Fig. 1. Parylene Array Device. (A) and (B): The complete implant, including main circuit board, parylene electrode array, headplug connector, EMG wires, and ground wires. (C): Tip of the array, showing suture sites. (D): Detail of a single electrode. The bright surface regions are electrical contacts and the darker material is a layer of parylene to prevent delamination. (E): View of the entire parylene and platinum microfabricated portion of the implant. Figure reproduced under Open Access from [1].

The methodology is based on a novel *Gaussian Process Batch Upper Confidence Bound* (GP-BUCB) active machine learning technique developed by some of the authors [5]. GP-BUCB uses a batch processing format that meshes with clinical practice: data from a completed training session is processed to select a batch of stimuli to be tested in the next training session. This approach decouples the data processing step from the clinical process. When the data processing step is sufficiently rapid, as in the experiments here, this approach can also be used within a session. The basic theory underlying GP-BUCB is developed and analyzed elsewhere [5]. This paper focuses on animal experiments designed to evaluate the method. These studies used a block strategy in which an experienced neurophysiologist and the GP-BUCB algorithm each selected and applied batches of stimuli in an interleaving sequence. Each was blind to the competitor's actions and the responses elicited, allowing comparison of the exploratory/exploitative strategies, while largely compensating for spinal plasticity. Our experiments showed that an automated algorithm can find high-performing stimulus parameters efficiently, competitively with a human expert. This should lead to a better therapeutic recovery experience, and shows that an automated algorithm can be a competitive option for determining effective therapeutic strategies.

Relation to Prior Work. This paper lies at the intersection of multiple fields: machine learning, neuromotor rehabilitation, and multi-electrode array technology. Herein, we address spinal cord injury, a medical condition with broad impact, which has been widely studied in both human [2], [3] and animal models [6]. Reviews of SCI research include [7]–[9]. Although regenerative approaches [10] hold long-term promise, they have not yet had substantial success in patients with a complete SCI [7].

Since there is no present cure for SCI, current practice focuses on therapy for the injured neuromuscular system. Locomotor therapy [11] produces gains in some patients through a variety of putative mechanisms [12], [13]. Several therapies involve electrical stimulation. Functional electrostimulation (FES) [14] aims to create muscle activation patterns associated with a desired movement. Since FES is an open-loop method, the stimulation must be carefully designed and/or user-controlled if complex behaviors are desired. FES also can engender rapid fatigue [15]. Other approaches focus on spinal cord stimulation to rehabilitate the cord's motor control ability, without addressing the injury itself [16]. This philosophy relies on the spinal cord's intrinsic circuitry caudal to the injury site, which remains viable and adaptable [8]. Specific targets include interneuronal networks responsible for reflexes and the central pattern generators that coordinate muscle activity in walking [17]. A variety of methods for delivering the electrical stimulus have been suggested, including penetrating microelectrodes [18].

The present paper focuses on *epidural spinal electrostimulation (EES)*, which relies upon an electrode array implanted in the epidural space over the dorsal aspect of the lumbosacral spinal cord. Although originally developed for chronic pain therapy [19], epidural electrostimulation can produce complex motor patterns [17] in SCI subjects. EES has been successfully applied to spinal and decerebrate cats, spinalized rats, and humans with an SCI [9]. Properly configured EES can produce walking motions [9]. The combination of EES with partial body weight support exercise training can produce substantial functional and metabolic gains in locomotion in an incomplete quadriplegic patient [20]. More recently, some of the present authors have used EES coupled with motor training to demonstrate substantial gains in a motor-complete patients [2], [3]. One mechanism believed to be involved with in this type of stimulation is the activation of afferent fibers as they enter the spinal cord through the dorsal nerve roots [21]. A detailed attempt to examine spinal circuitry via EES-evoked potentials in human participants is presented in [22]. Even in this light, it is difficult to choose stimuli *a priori*, and so a search is necessary.

Since the search for optimal stimulating parameters consumes valuable patient and clinician time, any automated algorithm must quickly find high-performing stimulus parameters (*explore* the stimulus space) and then apply them (*exploit* its knowledge) to enable useful therapy. The unavoidable tension between these two requirements is classically called the *exploration-exploitation tradeoff*. Well-designed *active learning* algorithms strike an effective balance. The companion paper [5] reviews the active learning literature. Here we focus on the use of active learning for neurostimulation.

Our algorithm is not the first to "tune" the electrical stimuli applied by a medical device. Machine learning algorithms have tuned cochlear implants [23]. Computational models have been proposed to adjust deep brain stimulation (DBS) parameters [24], and automated DBS motor evaluation is in development [25]. In comparison to our approach, the ad-hoc techniques used in the cochlear tuning have no formal convergence guarantees. Moreover, EES invokes a more complex response (involving sensorimotor spinal networks) than does cochlear stimulation. Although the need for automated DBS tuning has been recognized [26], it is not yet a reality.

Whereas this paper focuses on neural stimulation, active learning has been applied to Brain Computer Interfaces (BCIs), which classify recorded neural activity. Active learning has been used to find the stimuli which produce good discrimination between volitional and resting states [27], [28], to calibrate BCI decoders [29] and BCI systems [30]. In contrast, this paper uses an approach with more structure (a Gaussian process class of reward functions) over the space of actions in order to enable very large decision sets. Further, we optimize a reward function *while simultaneously applying the therapy*, rather than in a separate, potentially non-therapeutic session.

Finally, we note that our automated stimulus optimization approach can likely be adapted to other multi-electrode therapies, such as transcranial direct current [31], transcutaneous spinal cord [32], or deep brain stimulation.

Paper Organization. Section II describes the procedures and the electrode arrays used in our experiments. The theory and structure of the GP-BUCB algorithm [5] is briefly reviewed in Section III and Section IV describes adaptations of GP-BUCB to the clinical and experimental setting. Section V summarizes our experimental results, with Section VI providing a detailed analysis.

II. Experimental Methods

This section describes the preparation of the animal subjects (Section II-A), the micro-fabricated (Section II-B) and wire electrode arrays (Section II-C) used to stimulate their spinal cords, and basic testing procedures (Section II-D).

A. Injury, Implantation, and Animal Care

Our surgical and care procedures are largely derived from [1], and are similar to those used for cats [33]. All animals are adult female Sprague Dawley rats, approximately 300 g in mass at time of implantation. The following procedures are performed on each animal, usually in a single surgery:

1) Partial laminectomy at the T8-T9 vertebral level and complete spinal cord transection at approximately the T8 spinal segment, including the dura, using microscissors;
2) Placement of gel foam at the transection site as a coagulant and separator of the cut spinal cord ends;
3) Partial or full laminectomies of some vertebrae (T11, T12, L3, and L4 for animals receiving parylene arrays, T12, T13, L1, and L2 for those receiving wired arrays);
4) Implantation of an epidural stimulating array, inserted using the T11, L4 (T12, L2) laminectomies for parylene (wire) arrays, placing the most rostral electrodes in the middle of the T12 vertebral level, and sutured to the dura at the rostral end using 8-0 Ethilon sutures;
5) Implantation of 2 ground wires (each made of 5 braided 0.003 cm gold wires, A-M Systems, Sequim, WA) for parylene arrays, or one teflon coated stainless steel wire for the wired arrays (0.304 mm, AS 632, Cooner Wire, Chatsworth, CA). The grounds are placed in the muscle mass dorsal to the spinal column;
6) Implantation of a pair of multi-stranded, Teflon-coated stainless steel EMG wires (AS 632, Cooner Wire) into the bellies of left and right tibialis anterior (LTA & RTA) and left and right soleus (LSol & RSol) muscles; and
7) Attachment of two headplug connectors (Amphenol, Wallingford, CT), that are screwed to the skull and secured with dental cement.

All surgeries are performed under aseptic conditions with general anesthesia (Isoflurane) delivered via face mask. Analgesia is provided with buprenex (0.5-1.0 mg/kg, 3 times per day subcutaneously), begun before the end of surgery and continued for 3-5 days post-surgery. The animals were administered Baytril sub-cutaneously at the end of surgery and at 12-hr intervals thereafter for at least 3 days. The animals are allowed to recover from anesthesia within an incubator and are individually housed both pre- and postoperatively, with free access to food and water. After recovering for one week (day 7 post-surgery, denoted P7), the experiments begin. Experiments continue as long as the animal and array both remain viable, or until 6 weeks post-operative (P42). Due to their spinal cord injuries, the animals' bladders are manually expressed 2-3 times per day and their hind limbs are manually moved through a complete range of motion once per day to retain joint mobility.

All procedures were carried out in accordance with the NIH Guide for the Care and Use of Laboratory Animals and were approved by the UCLA Animal Research Committee.

B. Parylene Arrays

Two animals were implanted with *parylene* stimulating arrays, which were built using MEMS fabrication and traditional microelectronics, described in detail in [1]. The 26 mm x 3mm array is sized to the rat spinal cord. The array consists of 27 platinum electrodes (each 0.2 mm x 0.5 mm, partially covered by an anti-delamination waffle pattern of parylene) and platinum wire traces deposited on a parylene C substrate (see Fig. 1). The flexibility of the 20μm thick array conforms to the cord's movements, allowing delivery of roughly the same stimulus in various body positions. The electrodes are in three rostro-caudal columns (see Fig. 2) denoted "A," "B," and "C" from the animal's left to its right, and are numbered by rows from 1 (rostral, L2 spinal cord level, T12 vertebral level) to 9 (caudal, S2 spinal cord level, L2 vertebral level). The rows are equally spaced.

The array is coupled to a biocompatible circuit board (33.2 mm x 10.3 mm) mounted dorsally on the spinal column. This board's circuits control the stimuli, record responses, and communicate with external circuitry through a headplug connector. A stimulus consists of a single pair of electrodes (cathode and anode) chosen from 29 candidates, the 27 on the array and 2 grounds located within the body. Considering only bipolar configurations (i.e., those without grounds), there are 702 possible stimulus pairs. Due to the design of the driving circuitry, certain pairs (36 in total) cannot be stimulated, leaving 666 pairs over which the stimulus can be optimized.

C. Wire-Based Spinal Stimulating Arrays

The *wire* electrode arrays consist of 7 Teflon-coated, multi-stranded, stainless steel wires (5×30 guage, A-M Systems; 2×AS 632 wires, Cooner Wire) laid parallel. Openings cut into the wire insulation create exposed electrodes on the

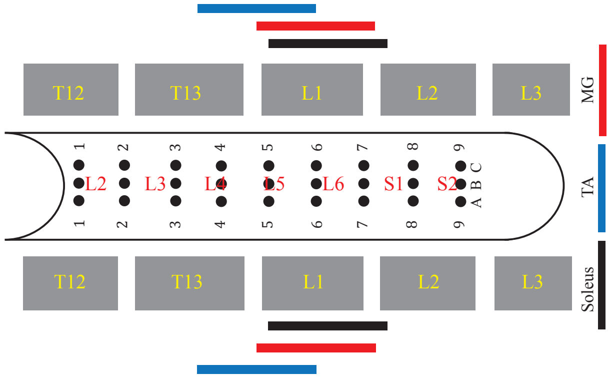


Fig. 2. Placement of a 9x3 array relative to the spinal cord (segmental level in red) and vertebrae (gray). The colored bars at the top and bottom represent the rostro-caudal spinal locations of the soleus, tibialis anterior, and medial gastrocnemius motor pools. Adapted under Open Access from [1].

surface facing the dura. The wires are sutured to the dura above and below the electrode, ensuring consistent stimulus location. The other end is then connected to a headplug connector, to which the implanted EMG wires are also routed. The 7 stimulating electrodes are placed over the same spinal locations as electrodes A1, A4, A9, B2, C1, C4, and C9 on the parylene array (Fig. 2).

D. Animal Testing Procedures

During testing, the animals are suspended in a vest that supports their body weight. The device is positioned so that the animal is in a bipedal position, with both hindpaws in contact with a custom surface offering good traction. In a typical experiment, the algorithm chooses a batch of five bipolar electrode pairs, possibly with repetition. The stimulus associated to each selection consists of a train of 10 (or 20 in some animals) constant voltage pulses (1 Hz repetition, monophasic, 0.5 ms pulse width) applied across the selected electrodes.

The EMG signals corresponding to evoked potentials are amplified (A-M Systems model 1700) and recorded by custom LabVIEW (National Instruments, Austin, TX) software, with a 10 kHz sampling rate. The digitized signals are processed with custom MATLAB (The MathWorks, Inc., Natick, MA) code to calculate the peak-to-peak amplitude of each evoked potential within a time window defined relative to the onset of the stimulus pulse. Along with previously recorded data, these new observations are used to select the five actions comprising the next batch. On a typical testing day, five batches are selected. A *run* of the algorithm typically lasts several days, and several runs may be performed on the same animal, with the algorithm's memory wiped between runs.

Human-selected batches (5 choices of electrode pairs and voltage) are interleaved between tests of the algorithm's batch selections in an alternating manner, constituting an intra-animal baseline. The algorithm and human were blind to one another's actions and the animal's resulting responses. The two data sources were not combined for decision-making purposes, but they were combined for meta-analysis and tuning of the algorithm's hyperparameters.

III. Review of the GP-BUCB algorithm

The GP-BUCB algorithm is extensively developed elsewhere [5], [34]; here, we briefly sketch its construction. GP-BUCB is a Gaussian process (GP) bandit algorithm. The bandit setting [35] describes an agent which seeks to obtain high reward in its interaction with an unknown reward function f. In round t, the agent selects action $\boldsymbol{x}_t$ from a decision set D, and then receives the single observation, $y_t = f(\boldsymbol{x}_t) + \epsilon_t$, of the corresponding reward, $f(\boldsymbol{x}_t)$, corrupted by noise ϵ_t. Since only the reward corresponding to the action is observed, actions must be selected in view of *both* exploration and exploitation.

GP bandits impose the structural assumption that the reward function is drawn from a GP, a probability distribution over functions. This assumption is useful mathematically and because it allows a relatively intuitive encoding of prior (e.g., anatomical) knowledge about the problem. A GP is characterized by a mean function $\mu_0(\boldsymbol{x})$ and a kernel function $k(\boldsymbol{x}, \boldsymbol{x}')$, which describes the covariance between $f(\boldsymbol{x})$ and $f(\boldsymbol{x}')$. With their *hyperparameters*, θ, the kernel and mean functions encode assumptions about likely reward functions f, such as smoothness with respect to actions $\boldsymbol{x}$, or locations of regions which are likely to yield high reward.

At time t, GP-BUCB updates the posterior distribution over $f(\boldsymbol{x})$ for each $\boldsymbol{x} \in D$ using the actions $\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{t-1}\}$ and observations $\boldsymbol{y}_{1:t-1} = \{y_1, \ldots, y_{t-1}\}$: the posterior over $f(\boldsymbol{x})$ is $f(\boldsymbol{x}) \sim \mathcal{N}(\mu_{t-1}(\boldsymbol{x}), \sigma^2_{t-1}(\boldsymbol{x}))$, where

$$\mu_{t-1}(\boldsymbol{x}) = \mu_0(\boldsymbol{x}) + \vec{k}^T(\boldsymbol{x})(\mathbf{K} + \sigma_n^2 I)^{-1}(\boldsymbol{y}_{t-1} - \vec{\mu_0}) \text{ and}$$
$$\sigma^2_{t-1}(\boldsymbol{x}) = k(\boldsymbol{x}, \boldsymbol{x}) - \vec{k}^T(\boldsymbol{x})(\mathbf{K} + \sigma_n^2 I)^{-1}\vec{k}(\boldsymbol{x}), \tag{1}$$

where $\vec{k}(\boldsymbol{x})$ is the column vector of $k(\boldsymbol{x}, \boldsymbol{x}_i)$ for $i = 1, \ldots, t-1$, $\vec{\mu_0}$ is the corresponding vector of prior means $\mu_0(\boldsymbol{x}_i)$, and $\mathbf{K}$ is the Gram matrix, i.e., $[\mathbf{K}]_{i,j} = k(\boldsymbol{x}_i, \boldsymbol{x}_j)$. This posterior assumes that f is drawn from the GP and that the noise is i.i.d., with $\epsilon_t \sim \mathcal{N}(0, \sigma_n^2)$. The noise accounts for all factors which influence the response, but are unobserved by the algorithm. The GP-BUCB algorithm addresses the problem of making decisions in "batches," where not all feedback is immediately available; only measurements $\boldsymbol{y}_{\text{fb}[t]}$, where $\text{fb}[t] < t - 1$ are available. Conventionally, all observations $\boldsymbol{y}_{t-1}$ are required to select $\boldsymbol{x}_t$, but by using delayed measurements, the expenditure of valuable clinical time waiting for data processing is avoided. Mathematically, if the pending actions $\boldsymbol{x}_{\text{fb}[t]+1}, \ldots, \boldsymbol{x}_{t-1}$ are known to the algorithm, the posterior variance *available after those observations arrive* can be calculated from (1), since this equation does not depend on the observation values. GP-BUCB uses this pseudo-updated posterior to select $\boldsymbol{x}_t$,

$$\boldsymbol{x}_t = \underset{\boldsymbol{x} \in D}{\operatorname{argmax}}[\mu_{\text{fb}[t]}(\boldsymbol{x}) + \beta_t^{1/2}\sigma_{t-1}(\boldsymbol{x})], \tag{2}$$

where β_t is a time-varying parameter (in the following, chosen to increase during a testing session). This lets GP-BUCB select observations in non-redundant batches (or despite a delay). A number of performance guarantees for GP-BUCB with particular choices of β_t are obtained by [5].

IV. Clinical Adaptations of GP-BUCB

Noteworthy modifications of the GP-BUCB algorithm which are needed for clinical application are described below.

TABLE I
GP priors used to model responses to spinal stimuli. The GP prior mean is $\mu_0(t) = c_0$ or $c_0 + tc_1$.

Animal	Run	Kernel		Mean		Repetitions
		Function	Hyperparameters	Function	Hyperparameters	Allowed
P1	Run 1a	SE	$l = [0.2740, 0.4659, 0.3297, 0.6300, 1.2018]$, $\sigma = 0.1244, \sigma_n = 0.0268$	constant	$c_0 = 0.0$ mV	2
	Run 1b				$c_0 = 0.1$ mV	1
W1	Run 1a	SE	$l = [0.5480, 0.9317, 0.6593, 1.2599, 2.4034]$, $\sigma = 0.1244, \sigma_n = 0.0268$	constant	$c_0 = 0.1$ mV	3
	Run 1b			constant	$c_0 = 0.9$ mV	2
	Run 1c	SE	$l = [0.1147, 46.9592, 0.1949, 5.0130, 0.9114]$, $\sigma = 0.5288, \sigma_n = 0.1708$	constant	$c_0 = 0.9$ mV	2
	Run 2	SE	$l = [2.9453, 6.3777, 3.4291, 2.3453, 2.4034]$, $\sigma = 0.7903, \sigma_n = 0.1364$	constant	$c_0 = 0.9$ mV	2
	Run 3			constant	$c_0 = 1.4$ mV	2
W2	Run 1a	Hybrid	$l_1 = [1.0000, 2.7183, 1.0000, 2.7183, 2.7183]$, $\sigma_1 = 0.2865, \sigma_2 = 0.2231, \sigma_n = 0.1353$	Linear	$c_1 = 0.08$ mV/day, $c_0 = -0.2492$ mV	2
	Run 1b			Linear	$c_1 = 0.08$ mV/day, $c_0 = 1.7$ mV	2
	Run 2			Linear	$c_1 = 0.08$ mV/day, $c_0 = -0.1348$ mV	2
P2	Run 1	Hybrid	$l_1 = [1.0000, 2.7183, 1.0000, 2.7183, 2.7183]$, $\sigma_1 = 0.2865, \sigma_2 = 0.2231, \sigma_n = 0.1353$	Linear	$c_1 = 0.08$ mV/day, $c_0 = 0.2513$ mV	2

A. Objective Function

To optimize clinical reward, GP-BUCB must observe the actual reward or its faithful surrogate. Many methods exist to quantify standing performance [36], [37], which we eventually aim to optimize by EES. Many motor skill grading schemes rely on human observation [38], [39] or automated post-hoc analysis [40], [41]. Although we do not use human grading due to the variability in human judgement, our method can work with human-based ordinal ratings and is mathematically agnostic to the particular choice of scalar reward function.

We use measured EMG activity as the reward function in our experiments. We chose the peak-to-peak amplitude of the left tibialis anterior (LTA, a foot dorsiflexor) muscle response to individual stimulus pulses (fixed amplitude of 5V, with 7V used in animal P2), in the "middle response" (MR, 4.5 - 7.5 ms post-stimulus) latency period, corresponding to motor responses which likely involve a single synaptic delay in the motor path. Since the evoked potentials represent a low-level function of spinal interneuron networks, they may be less sensitive to stimulus parameter variations than higher-level motor functions. At the 1 Hz stimulus frequency, the pulse responses can be dissociated from their predecessors and successors. Although this performance metric does not involve a complex motor behavior, it provides a short-term, measurable surrogate for therapeutic effectiveness, as it is an indicator of interneuronal function. Importantly, although our end-goal is to improve therapeutic outcomes, focusing on a short-term surrogate avoids the credit assignment problem, i.e., determining which actions are more or less responsible for that therapeutic outcome. This choice also allows us to use an intra-animal control design, thus avoiding a requirement for infeasibly large cohorts of animals.

Showing that the algorithm can successfully control this activity and learn the structure of spinal cord's responses demonstrates a step toward a therapeutic implementation. Further, since the prior means specified for this function in Table I do not vary with respect to stimulus location, the entirety of the function's spatial structure must be discovered online; thus, successful learning of spatial structure indicates that the algorithm may be able to discover previously unknown or patient-specific patterns via exploration.

Since the performance of this algorithm was compared to a human experimenter, we describe the experimenter's search and exploitation strategy. For a typical day, involving five batches (with five selections of stimulus parameters per batch):

- During batches 1-3, the human's first 3 actions were chosen using prior knowledge of electrode placement relative to the target muscle's motor pool, and observations from earlier tests. The 4^{th} and 5^{th} actions were typically small variations on successes in the first 3 choices.
- In the 4^{th} and 5^{th} batches, the human selected actions to explore portions of the spinal cord and array.

The human and the algorithm acted under different rules, confounding their competitive analysis. First, the human did not repeat an action within any day. Further, the human received immediate feedback, i.e., could observe the responses to an action and immediately use this information. Despite these factors, we maintain that the human expert's performance is an important baseline, due to the human's reasonable effectiveness at finding and utilizing good stimulus parameters.

B. Time Variation of the Reward Function

The GP-BUCB algorithm explores and exploits the reward function by selecting actions which individually are expected to yield high reward, substantial new information on the reward function, or both. Assuming a stationary reward function, well chosen algorithm parameters, and some technical conditions, GP-BUCB is guaranteed to converge over time to a subset of actions which yield maximal reward [34]. In practice, the response function is temporally non-stationary across sessions due to factors including spinal cord plasticity due to training and the natural timecourse of post-injury response. Other effects include array degradation or small physical array displacements along the cord. Since spinal cord plasticity is an essential phenomenon of this therapeutic approach, the algorithm's model of the spinal response must capture temporal variation.

To do so, a variable representing time, T, (in units of days post-injury) is added to the GP model. Hence, GP-BUCB must regress on a function of both applied stimulus $\boldsymbol{x}$ and time T. This approach requires the covariance function to be a map $k : (D \times T) \times (D \times T) \to \mathbb{R}$. Further, actions are selected from the subset of $D \times T$ corresponding to the present time t, effectively a time-varying decision set D_t. The modeling problem becomes one of extrapolating from the past to the present, both on the order of days (between sessions) or minutes (between batches). In our setting, the posterior uncertainty at t grows as t increases and previous observations recede into the past. Even with optimal action selection, the posterior uncertainty cannot decrease below a level set by the measurement noise, rate of observation, and rate of change in the underlying responses, and effect analogous to non-zero steady state uncertainty limits for Kalman filters [42] and Gaussian Markov processes [43]. Thus, the algorithm will never know the best action with vanishingly small uncertainty. Fortunately, the spinal cord's current state appears to vary slowly; given observations from previous days, the algorithm's internal model should make reasonable predictions about the spinal cord's responses. Further, it is not necessary to select the optimal action at time t, only one which provides useful therapy. Indeed, maintaining some variety in the stimuli may be appropriate, as training via an overly limited set of stimuli may result in inferior therapy [41].

C. Redundancy/Repetition Control

Another deviation from the classical GP bandit setting concerns the repetitions of stimuli within short periods. For most experiments (see Table I), the algorithm was not allowed to request the same action more than twice per day in order to ensure adequate exploration. Since 5 batches of 5 stimuli are typically requested daily, 13-25 distinct electrode pairs are tested. Some of the consequences of this restriction are dicsussed in Section VI-A.

D. Kernel and Mean Functions

The kernel and mean functions describe the algorithm's prior assumptions about spinal cord response. To ensure reasonable action selection, their chosen form must carefully encode crucial problem information. Most importantly, $k(\boldsymbol{x}, \boldsymbol{x}')$ encodes the dependence between stimuli $\boldsymbol{x}$ and $\boldsymbol{x}'$. For our experiments, the 5-dimensional stimulus description vector consisted of the anode and cathode spatial coordinates on the array (4-dimensions) and the time, in days post-injury. For the first two animals, we used a squared-exponential (SE) kernel $k_{SE}(\boldsymbol{x}_i, \boldsymbol{x}_j)$ (see [43], Section 4.2)

$$k_{SE}(\boldsymbol{x}_i, \boldsymbol{x}_j) = \sigma^2 \exp[-r^2/2], \tag{3}$$

$$r = \sqrt{(\boldsymbol{x}_i - \boldsymbol{x}_j)^T \Sigma^{-1} (\boldsymbol{x}_i - \boldsymbol{x}_j)}, \tag{4}$$

where $\Sigma = \operatorname{diag}(l_1^2, \ldots, l_5^2)$ and l_n is the *length scale* corresponding to dimension n of $\boldsymbol{x}$. The SE kernel implies that the GP is infinitely mean-square differentiable (see [43], Section 4.1.1). For the third and fourth animals, a *hybrid* kernel $k_h(\boldsymbol{x}_i, \boldsymbol{x}_j)$ was used:

$$k_h(\boldsymbol{x}_i, \boldsymbol{x}_j) = \sigma_1^2 k_m(\boldsymbol{x}_i, \boldsymbol{x}_j) + \sigma_2^2 \delta(i, j), \tag{5}$$

where k_m is a 3rd order Matérn kernel, $k_m(\boldsymbol{x}_i, \boldsymbol{x}_j) = (1 + \sqrt{3}r)\exp(-\sqrt{3}r)$, which allows considerably rougher functions than the SE kernel [43], and $\delta(i, j)$ is the Dirac delta function on stimulus indices i, j. This formulation tries to infer the noisy function $f(\boldsymbol{x}_i, i) = g(\boldsymbol{x}_i) + \eta_i$, $\eta_i \sim \mathcal{N}(0, \sigma_2^2)$ instead of the underlying function $g(\boldsymbol{x})$, which has kernel k_m. Since k_h implies a minimum uncertainty about $f(\boldsymbol{x})$, the response to any candidate action, short-time variations in the reward are subsumed into the noise term, leaving the overall function shape to be captured by the Matérn kernel. The hybrid kernel proved to be more successful. Both kernels were implemented using the GPML toolbox [44]. Table I shows all kernel hyperparameters used in the experiments.

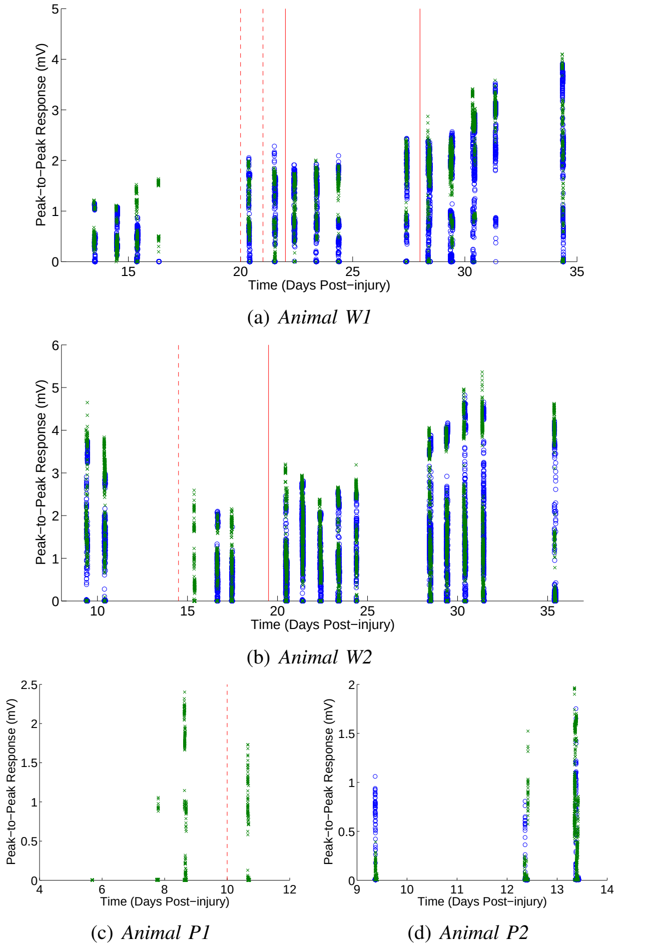
Following conventional practice, the hyperparameters used with the first two animals were estimated from an experimental data set selected by the human expert, which evenly sampled the decision set, and thus could be expected to faithfully estimate the spatial lengthscales. The fitting was performed using a conjugate gradient method[2] on the data likelihood under the GP model. This fitting procedure was reasonably successful for the spatial lengthscales, but poorly captured the multi-day temporal lengthscales. This poor fit likely results from the strong influence of short-timescale effects (e.g., noise) in the value of the data likelihood, dominating the effects of fitting the underlying spinal plasticity. Consequently, the time lengthscale in the first two animals was hand-selected. Similarly, the hyperparameters were hand-selected for the third and fourth animals' kernel in such a way as to largely ignore in-session variation and produce model predictions capturing the long-term changes in the spinal cord's responses.

V. Results

Experiments in four rats prepared as in Section II were carried out, resulting in 37 testing sessions and 1200 actions (670 actions selected by the algorithm, with the rest selected by the competing human). Fig. 3 shows a retrospective of GP-BUCB performance across all experiments. Each graph plots the reward (peak-to-peak evoked potential amplitudes) resulting from all stimulus pulses, color-coded according to the human or GP-BUCB origin of the stimulus selection.

These plots show substantial variation in the evoked potentials over the time frame of the experiments. In all animals, the per-day range of rewards is similar for both human- and GP-BUCB-selected stimuli, indicating that response variations do not strongly depend upon which agent selects the actions and thus that the interleaved block structure provides a reasonable, intra-animal baseline for examining GP-BUCB. The temporal variation in reward appears to be composed of both day-to-day fluctuations and a long-term, upward trend. While some of this upward trend is due to the algorithm's growing knowledge of effective stimulus patterns, much of it is a general, plastic increase in spinal cord responses to the stimuli. Table II summarizes the experiments, the number of actions initiated by the algorithm, and the number of unique stimuli chosen.

[2] Using the minimize.m function in the GPML toolbox [44].



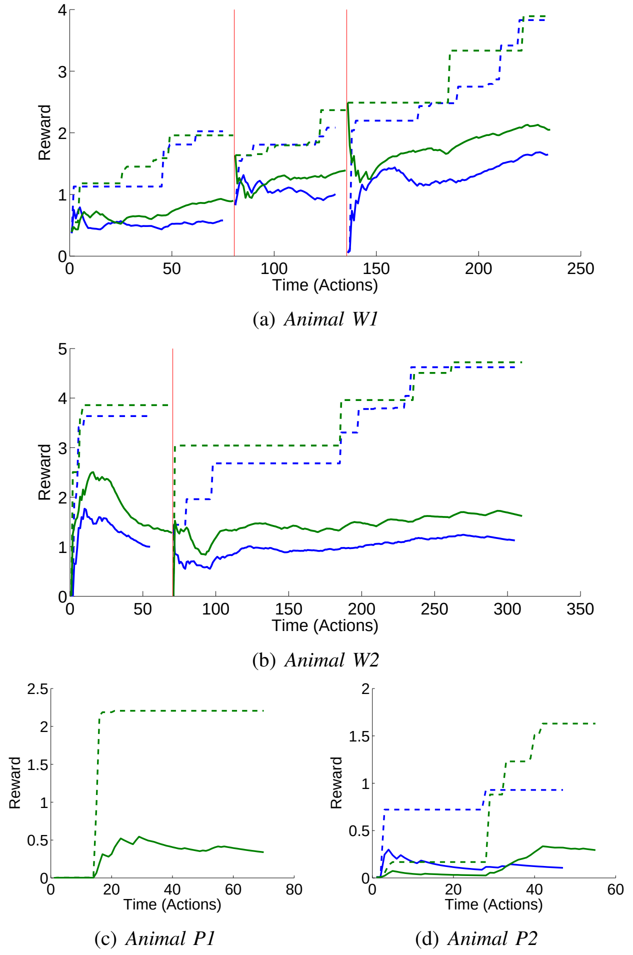
(a) *Animal W1*

(b) *Animal W2*

(c) *Animal P1*

(d) *Animal P2*

Fig. 3. Peak-to-peak amplitude (i.e., reward in mV) of individual stimulus pulses for each animal. Blue circles: potentials evoked by the human expert's choices; Green 'x': potentials due to algorithm choices. Solid red lines denote a wipe of the algorithm's memory. Dashed lines denote hyperparameter changes without a memory wipe. The human experimenter and algorithm were blind to each other's actions and the resulting rewards, and their actions were interleaved in batches. The decreased evoked potential magnitude from days 9 and 10 to 15-17 in animal W2 is a typical feature of the preparation.



(a) *Animal W1*

(b) *Animal W2*

(c) *Animal P1*

(d) *Animal P2*

Fig. 4. Human expert's (blue) and GP-BUCB's (green) reward, defined as the peak-to-peak amplitude (mV) of MR evoked by an epidural electrical stimulus of 5 V (animals P1, W1, and W2) or 7 V (animal P2), delivered in 1 Hz pulse trains. The *average* reward (solid lines) measures the rewards generated by *every* action, while the *maximum* reward (dashed) is the best reward observed up to a given time. GP-BUCB typically has a superior average reward to the human, while maintaining a superior or competitive maximum reward. In animals W1 (a) and W2 (b), runs (i.e., periods between memory wipes) are separated by red vertical lines.

A. Evaluaton of Reward

Reward is calculated from the response to an action, i.e., a continuous sequence of pulses. Let y_τ denote the peak-to-peak response to the τ^{th} stimulus pulse. The reward resulting from an action started at time t is calculated as:

$$w_t = \frac{1}{\tau_{max}(t) - \tau_{min}(t) + 1} \sum_{\tau=\tau_{min}(t)}^{\tau_{max}(t)} y_\tau, \qquad (6)$$

where $\tau_{min}(t)$ and $\tau_{max}(t)$ respectively index the first and last pulses for action t.

The *maximum reward*, W_{max}, may be examined in terms of the maximum reward value observed so far,

$$W_{max}(T) = \max_{t \le T}(w_t).$$

The maximum reward describes the thoroughness of the search over the stimulus space, since a large maximum reward implies that high-performing stimuli have been found, and thus could conceivably be exploited. However, this quantity ignores the number of applications of high-performing stimuli, so strategies like random search may perform well in this metric, without delivering effective therapy. Alternatively, the *average reward* so far observed

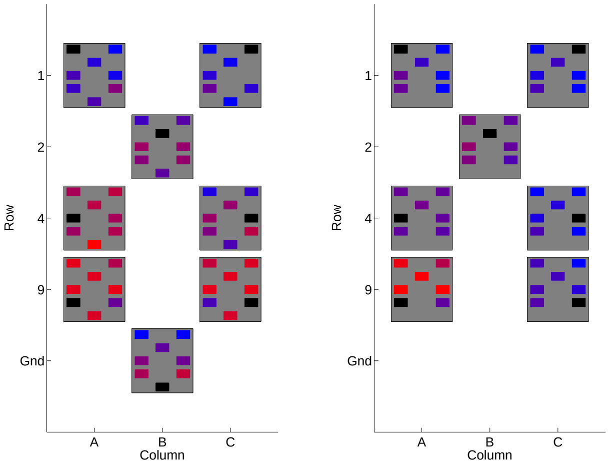
$$W_{avg}(T) = \frac{1}{T}\sum_{t=1}^{T} w_t,$$

measures how well exploration and exploitation are traded off; high average reward implies that a search process spent most of its search time choosing effective actions, yet also explored thoroughly enough to find high-performing stimuli. Superior average performance of one approach relative to another at time T is indicated by a larger average reward value.

Wire Arrays: Results: Fig. 3(a) and (b) show the search responses obtained by GP-BUCB and the human expert in animals implanted with wire arrays, while Fig. 4(a) and (b) show the associated maximum and average rewards. The algorithm chose a total of 235 and 310 actions respectively for animals W1 and W2. In both animals, left, caudal anode locations were associated with strong responses to the stimulus. Although the

TABLE II
NUMBER OF ACTIONS CHOSEN BY GP-BUCB (UNIQUE), AND SPEARMAN'S ρ BETWEEN EACH SELECTED STIMULUS'S REWARD AND THE NUMBER OF STIMULUS PULSES APPLIED BY GP-BUCB OR HUMAN.

ANIMAL & RUN	MACHINE ACTIONS (UNIQUE)	$\rho_{Machine}(p)$	$\rho_{Human}(p)$
P1, RUN 1	70 (43)	0.264 (0.0906)	N/A
W1, RUN 1	80 (23)	0.105 (0.6348)	-0.073 (0.6829)
RUN 2	55 (23)	0.339 (0.1132)	-0.045 (0.8091)
RUN 3	100 (29)	0.480 (0.0083)	0.325 (0.0500)
W2, RUN 1	70 (28)	0.582 (0.0012)	-0.034 (0.8682)
RUN 2	240 (33)	0.824 (< 0.0001)	0.569 (0.0001)
P2, RUN 1	55 (40)	0.581 (0.0001)	-0.477 (0.0021)



(a) *Animal W1* (b) *Animal W2*

Fig. 5. Time-averaged, intra-day-normalized response for cathode-anode electrode pairs, animal W1 (a) and animal W2 (b), shown with respect to array location as viewed dorsally (rostral at top). Each large box corresponds to a spatial location of a wire array electrode. Within the boxes, each small rectangle denotes the spatial location of one cathode which could be paired with an anode located at the large box's position, giving every stimulus a unique address. Each small rectangle's color corresponds to the time-averaged normalized response strength (1: red; 0: blue) of that anode-cathode pair. Black rectangles represent invalid pairs, e.g., A1_A1. Data are shown only from days where a total of at least 15 actions were taken between the human and algorithm. The set of highly excitable configurations for animal W1 is broader than that for animal W2. Note that monopolar stimuli (using a ground wire) were tested by the human experimenter in animal W1; these results are shown in the bottom-most boxes.

sizes of the strongly responding regions of the stimulus space were different in the two animals (see Fig. 5), the algorithm learned the response function well in both cases. Both animals exhibited increased responsiveness over the course of the experiments.

Parylene Microarray Animals: Results: Fig. 3(c) and 4(c) (animal P1) and Fig. 3(d) and 4(d) (animal P2) show the maximum and average reward obtained with the parylene arrays, which logged 4 days and 70 actions in animal P1, and 3 days and 55 actions in animal P2. GP-BUCB quickly found and exploited high-performing stimuli in this much larger space of possible stimuli. Note that high-performing stimuli observed in these two animals had caudal, left-side anode locations, as seen in the wire array experiments, though having such an anode was not itself sufficient for good responsiveness.

B. Computational Performance

When implemented in the MATLAB programming language[3], the EMG data processing needed to prepare one new batch typically required 2-3 min, whereas GP-BUCB's selection of new stimuli required approximately 5 s of processor time. In our alternating double blind experiment format, the total time to construct a new stimulus batch was less than the time required for the human expert to perform a batch of tests. These results indicate that, given a sufficiently efficient and fluid data acquisition and processing system, the algorithm could perform closed-loop active learning much more rapidly than was the case in our experiments. Under the most severe condition, with $n = 4432$ previous evoked potential observations, a new batch of five actions was constructed in 142.0 s, with 50 s devoted to Cholesky decomposition, and 50 s allocated to evaluation the kernel matrix. Computations are typically dominated by the Cholesky decompositions (which require an order n^3 calculation) and memory usage by storage of the $n \times n$ kernel matrices (order n^2 memory). As n grows with the increasing number of observations, the computational load can be managed by "forgetting" older observations, using approximate Cholesky decomposition, or sparse approximate GP regression [45]. Aggregating the individual, pulse-by-pulse responses to an action also saves substantial effort.

VI. DISCUSSION

Our animal experiments show that GP-BUCB can compete with a human expert in optimizing stimuli. This section also explores additional ramifications of our experiments on the prospects for using machine learning in multi-electrode stimulation therapy. Section VI-A analyzes the wire array results, including a discussion of inter-animal comparisons. Section VI-B addresses the analogous parylene array results. Section VI-C examines another perspective on therapeutic utility. Lessons learned regarding kernel function and hyperparameter selection are the focus of Section VI-D. Section VI-E examines the effectiveness of the algorithm's search from the perspective of information gain. Finally, effects on hindlimb muscles other than the LTA are the subject of Section VI-F .

A. Wire Array Animals

The wire array experiments tested how GP-BUCB copes with a time-varying reward function. The algorithm generally succeeded in future performance prediction. For example, in animal W2 (run 2) GP-BUCB showed strong queueing behavior (it first selected what were predicted and proven to be the best stimuli) before being forced by the repetition limit to consider other stimuli. This behavior results as the signature saw-tooth shape[4] in the average reward plots (Fig. 4(b)). Moreover, GP-BUCB autonomously maintained good performance (showing the same good queueing behavior) over long periods, e.g., in animal W2's two-week-long second run, which included two multi-day gaps in testing. In this respect,

[3] Running on either a 2.2 GHz quad-core i7 processor machine, with 8 GB RAM, and Mac OS 10.6, or an AMD Athlon 64 X2 Dual Core 3800+, 3 GB RAM machine running Ubuntu 12.04.

[4] The upward edge is produced by the superior reward resulting from the first stimuli in the session and the flatter or downward-sloped portion of the average reward curve results from the poorer stimuli which are queued later in the session.

these experiments validated that good forward prediction is possible over such intervals due to the utility of GP-BUCB's internal model, and reached a standard of performance which could be clinically relevant.

Algorithm Analysis. Several possible criticisms might follow from the fact that GP-BUCB can daily request a number of actions (typically 25, at least 13 unique, in 5 batches of 5 actions) which is of the same order as 42, the size of the decision set, D.

- *Measures of success:* Simply finding a high reward stimulus may be an inadequate measure of algorithm success.
- *Differentiation between algorithms:* Given the small size of D and the repetition limit, other learning algorithms would likely select similar actions to GP-BUCB.
- *Generalization to unconstrained settings:* Our use of a repetition limit can cloud an assessment of how effectively GP-BUCB can track f in an unconstrained setting.

The experimental data support additional several arguments for GP-BUCB's success. First, and unsurprisingly[5], GP-BUCB found high-performing stimuli in the first day (within 2 batches of 5 stimuli, in each case) in all wire array runs. This indicates that GP-BUCB's search is free from gross deficiencies. Second, GP-BUCB demonstrates success in the sense of avoiding ineffective stimuli. The number of unique electrode pairs selected (see Table II) was always substantially smaller than the size of the decision set. GP-BUCB thus avoids exhaustive testing, but appears to effectively model the reward for untested stimuli. Finally, since GP-BUCB was found to be competitive with the performance of a human expert in terms of the maximum reward so far observed, while maintaining a better average reward, it follows that GP-BUCB thoroughly searches for high-reward regions while simultaneously exploiting its knowledge in a therapeutically more effective fashion. These observations argue that GP-BUCB makes a successful exploration-exploitation tradeoff.

The second criticism argues that the decision set size and repetition limit mean that all algorithms will have to explore D somewhat, and that further, convergence cannot be observed, making algorithms indistinguishable. GP-BUCB's ability to discriminate against (likely bad) stimuli, discussed in relation to the first criticism, is a crucial feature in this setting, and compares favorably with simple methods such as random search or bandits without structured payoffs. With respect to convergence, the strong queueing behavior indicates that GP-BUCB has converged to the extent allowed by the repetition restriction, since high-reward actions are selected first (i.e., in preference to other actions), despite the low exploratory value of doing so under (2), until disallowed by the repetition limit. GP-BUCB's queueing behavior also requires a reasonably correct posterior over the rewards which will be received in the day's *first* batch of stimulation, implying that its internal model is adequate to the inter-day prediction task.

Fatigue offers an alternate explanation for the observed queueing behavior. If the animal rapidly tires during a session, and this fatigue appears as a decrease in evoked peak-to-peak amplitude, the responses elicited by the first stimuli would appear stronger than those from later stimuli. These "better" stimuli would then be selected early in future sessions, a feedback loop which could produce the observed queueing effect. However, the human expert's choices would also show a strong within-session fatigue pattern, which they do not. Also, the difference, ΔV, between average responses to different applications of the same stimulus in the same session (by human, algorithm, or both) can be measured. A large and negative ΔV is required if queueing is explained by fatigue. However, for actions leading to substantial responses (within-session average $\geq$ 0.2 mV), ΔV is small in magnitude ($\Delta V = -0.0871 \pm 0.4897$ mV, $n = 142$ pairs from animal W2), even over prolonged periods (-0.0336 ± 0.4228 mV for 34 intervals of $\geq$ 1 hour). Further, the correlation between the interval and ΔV is very weak ($r = 0.0090$). Fatigue, therefore, is not a convincing explanation for the observed queueing behavior.

The third criticism is more difficult to directly refute since no experiments without the repetition constraint were performed, and the repetition limit forces the algorithm to track stimuli which are high-performing, but suboptimal. However, for stationary reward functions, the performance of GP-BUCB is supported by theoretical guarantees and validation in simulation [5]. Although [5] does not provide guarantees in the case of time-varying rewards, such guarantees have been derived for other methods, e.g., [46], suggesting that similar guarantees could be derived for GP-BUCB. Together, the relatively benign time variation in the reward function in our experiments and the stationary case guarantees suggest that unconstrained GP-BUCB could also maintain a useful posterior over responses to previously high-performing stimuli.

Cross-animal Comparisons. Fig. 5 compares the performance of electrode pair selections across animals W1 and W2. The data demonstrate substantial repeatability between animals with regard to the evoked potential strength elicited by the same stimulus, particularly for the highest-performing stimuli (typically those with electrode A9 as the anode).

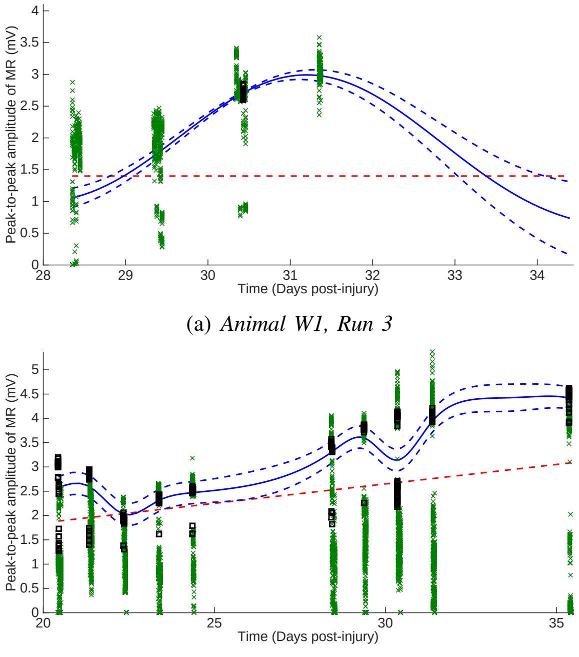
B. Parylene Array Animals

The data from the animals implanted with parylene arrays allow us to assesses how well GP-BUCB can search a large decision space (666 possible electrode pairs, of which no more than 25 can be tested in a typical day), using a structured reward function. In both animals, after finding a high-performing configuration (C8_A9 in animal P1, and C6_A9 in animal P2), the algorithm moved sharply to exploit this knowledge, allocating repeated queries to neighboring actions. Both discovered configurations including an anode at the left, caudal corner of the array, similar to those found in the wire array animals. Note that GP-BUCB overcame its spatially flat prior on f to find this anatomical pattern. The key, high-performing stimulus was found in the 3^{rd} batch for animal P1, and the 6^{th} batch for animal P2. The efficiency of the search and the rapid exploration of high-performing stimuli argue that the structured GP model provides a benefit over conventional bandit algorithms for this application.

[5] If 6 electrode pairs are effective out of 42, e.g., all those pairs including a particular anode, 10 stimuli drawn randomly without replacement will include one of these 6 with probability $> 82\%$.

TABLE III
PROPORTION OF APPLIED STIMULI WHICH ELICITED A PEAK-TO-PEAK RESPONSES EXCEEDING A THRESHOLD. BOLD INDICATES MORE SUPRA-THRESHOLD RESPONSES.

ANIMAL	AGENT	THRESHOLD Π (MV)			
		0.5	1.0	1.5	2.0
P1	ALGORITHM	0.229	0.157	0.100	0.043
W1, RUN 1	ALGORITHM	**0.662**	**0.450**	**0.263**	0.000
	HUMAN	0.440	0.253	0.067	**0.013**
W1, RUN 2	ALGORITHM	**0.945**	**0.727**	**0.545**	**0.091**
	HUMAN	0.720	0.520	0.280	0.040
W1, RUN 3	ALGORITHM	**0.910**	**0.850**	**0.790**	**0.560**
	HUMAN	0.828	0.697	0.626	0.424
W2, RUN 1	ALGORITHM	**0.657**	**0.543**	**0.357**	**0.214**
	HUMAN	0.564	0.418	0.273	0.127
W2, RUN 2	ALGORITHM	**0.779**	**0.642**	**0.442**	**0.308**
	HUMAN	0.681	0.472	0.285	0.153
P2	ALGORITHM	**0.255**	**0.091**	**0.036**	0.000
	HUMAN	0.085	0.000	0.000	0.000



(a) *Animal W1, Run 3*

(b) *Animal W2, Run 2*

Fig. 6. GP posterior over the response function $f(\text{A1, A9}, t)$. Points show responses evoked by electrode pair A1_A9 (black squares) and other stimuli (green 'x's). The red dashed line is the prior mean of the entire GP with respect to time. The posterior mean (solid) and ± 1 standard deviation confidence intervals (dashed) for $f(\text{A1, A9}, t)$ are shown in blue. (a): Animal W1, run 3, shows the effects of poor kernel and hyperparameter choices. The posterior prediction of the responses at the beginning of P34 (at right, using data acquired on P31 and earlier) was extrapolated poorly. This resulted from the squared-exponential kernel's inherent smoothness, the time lengthscale used, and the low assumed noise level. Actions selected on day P34, using this badly inaccurate model, were consequently unproductive. (b): The altered kernel, mean, and hyperparameter choices employed in animal W2, run 2, avoid this pathology, while still providing useful prediction.

Although no comparison to a human expert was made in animal P1, the P2 experiment followed the interleaving human-machine batch paradigm. Possibly due to prior anatomical knowledge, the human found strongly responding stimuli on the first testing day. However, soon after, the algorithm found high-performing stimuli in both animals. In the final day of animal P2 testing, GP-BUCB found stimuli of comparable reward to the best found by the human, as seen in Fig. 4(d).

C. Therapeutic Relevance

The reward in (6) makes the heuristic assumption that strength of response and contribution to long-term therapeutic outcomes are equivalent (i.e., linearly related). An alternate reward heuristic is

$$R_t = I(w_t > \Pi),$$

where w_t is defined in (6), Π is a threshold dividing stimuli into therapeutically effective and ineffective classes by average twitch strength, and $I(\cdot)$ is the indicator function. This alternate reward function is reasonable if fine differences between strongly responding stimuli (important to the previous reward measures) are irrelevant. Table III presents a comparison of human and machine reward under this heuristic for different values of Π. For nearly all runs and values of Π, proportionally more algorithm actions met or exceeded the specification. Thus, if therapeutic effectiveness is a function of the proportion of stimuli which elicit a strong response, the algorithm is more effective than the human expert's chosen strategy.

D. Kernels and Hyperparameters

For the structured GP model to faithfully capture muscle responses, yet allow rapid learning, the kernel function, mean function, and their hyperparameters must be chosen well. Poor choices can lead to undesired behavior, as seen in Fig. 6(a). However, the comparatively good performance obtained in later animals with the hybrid Matérn kernel, (5), shows that appropriate kernel choice can resolve this problem (Fig. 6(b)).

Although the hybrid kernel was successful, our experiments suggest other possible kernels. For example, the hybrid kernel's noise term could be replaced by a term which models the covariance between measurements of the same action on the same day, but otherwise treats observations as independent. Such a kernel includes a hidden additive variable for each configuration on each day. A linear temporal term in the kernel may also be useful, provided that old observations can be forgotten, as it models the fact that non-responsive configurations typically remain non-responsive.

The kernel, mean, and hyperparameters were periodically re-examined. If appropriate, hyperparameters were re-initialized after an animal's first run, accompanied by a wipe of the algorithm's memory; Table I and Fig. 3 detail these refitting and memory wipe events. The choice to re-fit or wipe algorithm memory was made by human inspection. This approach protected the algorithm from making prolonged, catastrophic errors, introducing some bias. However, this manual oversight constitutes a meta-algorithm which can be compared to the oversight required to detect device failure and is thus not unrealistic. Automated execution of such re-examinations (in some sense, a form of hyperprior fitting) and also heuristics for detection of various failure modes should naturally be built into future systems, but are beyond the scope of this work.

E. Information Gain

GP-BUCB can also be asssessed by the amount of information it gains about the spinal cord's responses. Using the GP model, the information gain with respect to the entire

TABLE IV
INFORMATION GAINS (GP-BUCB, I_m; RANDOM SEARCH, I_r; HUMAN, I_h), COMPUTED USING HYBRID KERNEL. I_h CANNOT BE CALCULATED IN ANIMAL W1 DUE TO USE OF MONOPOLAR STIMULI, FOR WHICH $k_h(\boldsymbol{x}, \boldsymbol{x}')$ IS UNDEFINED. *: REWARD RATIO OUTLIER DISCARDED.

ANIMAL	MEAN I_m/I_r	MEAN REWARD	MEAN I_m/I_h	DAYS
& RUN	(± STD.)	RATIO (± STD.)	(± STD.)	USED
P1, RUN 1	0.730 ± 0.145	N/A	N/A	0
W1, RUN 1	0.549 ± 0.165	1.368 ± 0.519	N/A	5*
RUN 2	0.791 ± 0.132	1.474 ± 0.490	N/A	4
RUN 3	0.697 ± 0.105	1.297 ± 0.317	N/A	5
W2, RUN 1	0.811 ± 0.145	1.540 ± 0.560	0.798 ± 0.093	4
RUN 2	0.802 ± 0.119	1.600 ± 0.749	0.895 ± 0.152	10
P2, RUN 1	0.924 ± 0.214	2.520 ± 2.415	1.119 ± 0.323	3

reward function f can be quantified directly. Table IV shows the averaged ratio of the algorithm's daily gain in conditional information to that of the human and to that of a random sampling policy. Interestingly, the algorithm gains a comparable (but smaller) amount of information to both the human and random searches. The algorithm's repeated, exploitative actions, which produce higher mean daily reward, may account for most of this difference. However, the algorithm's high conditional information gain, despite allocating many actions to stimuli which are well-understood, suggests that the actions allocated to exploration are well-chosen.

F. Effect of Learning on Other Hindlimb Muscles

In addition to LTA responses, the peak-to-peak twitch amplitudes were recorded in the left soleus, right TA, and right soleus muscles. Often, several muscles responded strongly together to a given stimulus. Although the LTA twitch amplitude accounted for a large portion of the response amplitudes in the other muscles ($r > 0.25, p \ll 1e^{-3}$ in all animals and muscles, except the RTA in animal P1), these responses in other muscles also demonstrated substantial independence. Even in this simple paradigm, many combinations of muscle responses can be elicited. The creation of an efficiently computable reward metric which faithfully matches therapeutic utility in multi-muscle behaviors is a highly non-trivial problem worthy of future investigation.

VII. CONCLUSIONS

Our experiments show that an automated algorithm can handle the difficult task of maximizing the performance of a relatively complicated neural interface. These results provide a strong indication that a structured GP model can effectively capture spinal responses over large sets of possible stimuli (tens to hundreds) and substantial periods of time (weeks). Further, the GP model allowed effective extrapolation forward in time from previous sessions and observations, a requirement to avoid extensive re-exploration. GP-BUCB also found task-relevant anatomical structure in the responses it encountered, despite starting with no information as to which regions of the cord were expected to be most responsive. This indicates that these structures, and potentially novel or patient-specific ones, can be found by online, automatic experimentation. Given the extensive differences among individual injuries and injured spinal cords, this capability is essential. Finally, our results show that GP-BUCB can effectively trade off exploration and exploitation in a simple therapeutic application. In particular, the algorithm allocated more actions to high-performing stimuli while still matching the human experimenter's effectiveness in finding the best stimuli. Such a demonstration implies that automated algorithms have the potential to efficiently use the stimuli available to them, both inside and outside of the clinic, to simultaneously learn about the spinal cord's responses and deliver effective therapy. Our successful handling of this problem constitutes the successful completion of a necessary prerequisite for a well-founded attempt to solve the more complex clinical problems ahead, maximizing standing or stepping performance in paralyzed subjects and tuning stimuli to individual patients' needs.

ACKNOWLEDGMENT

We thank Yanan Sui and Mrinal Rath for their assistance in executing the experiments.

REFERENCES

[1] P. Gad *et al.*, "Development of a multi-electrode array for spinal cord epidural stimulation to facilitate stepping and standing after a complete spinal cord injury in adult rats," *J. Neuroeng. & Rehab.*, vol. 10, no. 1, p. 2, 2013.

[2] S. Harkema *et al.*, "Effect of epidural stimulation of the lumbosacral spinal cord on voluntary movement, standing, and assisted stepping after motor complete paraplegia: A case study," *The Lancet*, vol. 377, no. 9781, pp. 1938–1947, 2011.

[3] C. A. Angeli *et al.*, "Altering spinal cord excitability enables voluntary movements after chronic complete paralysis in humans," *Brain*, vol. 137, no. 5, pp. 1394–1409, 2014.

[4] T. Desautels *et al.*, "Application of a batch active learning algorithm to spinal cord injury therapy (program no. 466.01)," in *Neuroscience 2013 Abstracts*. San Diego, CA: Soc. for Neuroscience, November 2013.

[5] ——, "Parallelizing exploration–exploitation tradeoffs in Gaussian process bandit optimization," *J. Mach. Learning Res.*, vol. 15, pp. 3873–3923, December 2014.

[6] R. van den Brand *et al.*, "Restoring voluntary control of locomotion after paralyzing spinal cord injury," *Science*, vol. 336, no. 6085, pp. 1182–1185, 2012.

[7] S. Thuret *et al.*, "Therapeutic interventions after spinal cord injury," *Nat. Rev. Neurosci.*, vol. 7, no. 8, pp. 628–643, 2006.

[8] V. R. Edgerton *et al.*, "Rehabilitative therapies after spinal cord injury," *J. of Neurotrauma*, vol. 23, no. 3-4, pp. 560–570, 2006.

[9] Y. Gerasimenko *et al.*, "Epidural stimulation: Comparison of the spinal circuits that generate and control locomotion in rats, cats and humans," *Exper. Neur.*, vol. 209, no. 2, pp. 417–425, 2008.

[10] A. P. Pêgo *et al.*, "Regenerative medicine for the treatment of spinal cord injury: More than just promises?" *J. of Cellular and Molecular Medicine*, vol. 16, no. 11, pp. 2564–2582, 2012.

[11] A. Wernig and S. Müller, "Laufband locomotion with body weight support improved walking in persons with severe spinal cord injuries," *Spinal Cord*, vol. 30, no. 4, pp. 229–238, 1992.

[12] G. Courtine *et al.*, "Spinal cord injury: Time to move," *The Lancet*, vol. 377, pp. 1896–1898, June 2011.

[13] ——, "Recovery of supraspinal control of stepping via indirect propriospinal relay connections after spinal cord injury," *Nature Medicine*, vol. 14, no. 1, pp. 69 – 74, 2008.

[14] W. T. Liberson *et al.*, "Functional electrotherapy: Stimulation of the peroneal nerve synchronized with the swing phase of the gait of hemiplegic patients," *Archives of Physical Medicine and Rehabilitation*, vol. 42, pp. 101–105, 1961.

[15] A. Thrasher *et al.*, "Reducing muscle fatigue due to functional electrical stimulation using random modulation of stimulation parameters," *Artificial Organs*, vol. 29, no. 6, pp. 453–458, 2005.

[16] E. J. Bradbury and S. B. McMahon, "Spinal cord repair strategies: Why do they work?" *Nature Rev. Neurosci.*, vol. 7, pp. 644–653, August 2006.

[17] M. R. Dimitrijevic *et al.*, "Evidence for a spinal central pattern generator in humans." *Ann. of the New York Academy of Sciences*, vol. 860, p. 360, 1998.

[18] A. Prochazka *et al.*, "Neural prostheses," *J. of Physiology*, vol. 533, no. 1, pp. 99–109, 2001.

[19] C. N. Shealy *et al.*, "Electrical inhibition of pain by stimulation of the dorsal columns: Preliminary clinical report," *Anesthesia and Analgesia*, vol. 46, pp. 489–491, July-August 1967.

[20] R. Herman *et al.*, "Spinal cord stimulation facilitates functional walking in a chronic, incomplete spinal cord injured," *Spinal Cord*, vol. 40, pp. 65–68, 2002.

[21] K. Minassian *et al.*, "Human lumbar cord circuitried can be activated by extrinsic tonic input to generate locomotor-like activity," *Human Movement Sci.*, vol. 26, pp. 275–295, 2007.

[22] D. G. Sayenko *et al.*, "Neuromodulation of evoked muscle potentials induced by epidural spinal cord stimuation in paralyzed individuals," *J. of Neurophysiology*, vol. 111, no. 5, pp. 1088–1099, March 2014.

[23] D. Baskent *et al.*, "Using genetic algorithms with subjective input from human subjects: Implications for fitting hearing aids and cochlear implants," *Ear and Hearing*, vol. 28, no. 3, pp. 370–380, 2007.

[24] A. Frankemolle, "Reversing cognitive-motor impairment in parkinson's disease patients using computational modelling approach to deep brain stimulation programming," *Brain*, vol. 133, no. 3, pp. 746–761, 2010.

[25] T. Mera, "Kinematic optimization of deep brain stimulation across multiple systems in parkinson's disease," *J. Neurosci. Methods*, vol. 198-286, no. 2, p. 280, 2011.

[26] J. Giuffrida, "Automated optimization of deep brain stimulation settings multiple parkinson's disease motor symptoms," *Neurology*, vol. 74, no. 9, p. Suppl. 2:A479, 2011.

[27] J. Fruitet *et al.*, "Bandit algorithms boost brain computer interfaces for motor-task selection of a brain-controlled button," in *Adv. Neural Info. Proc. Syst.*, 2012, pp. 458–466.

[28] ——, "Automatic motor task selection via a bandit algorithm for a brain-controlled button," *J. Neural Eng.*, vol. 10, no. 1, p. 016012, 2013.

[29] T. Gürel and C. Mehring, "Unsupervised adaptation of brain-machine interface decoders," *Frontiers in neuroscience*, vol. 6, 2012.

[30] C. Vidaurre *et al.*, "Co-adaptive calibration to improve BCI efficiency," *J. Neural Eng.*, vol. 8, no. 2, p. 025009, 2011.

[31] A. Brunoni *et al.*, "Clinical research with transcranial direct current stimulation (tdcs): Challenges and future directions," *Brain Stimulation*, vol. 5, pp. 175–95, 2012.

[32] R. M. Gorodnichev *et al.*, "Transcutaneous electrical stimulation of the spinal cord: A noninvasive tool for the activation of stepping pattern generators in humans," *Human Physiology*, vol. 38, no. 2, pp. 158–167, April 2012.

[33] R. R. Roy *et al.*, "Chronic spinal cord-injured cats: Surgical procedures and management," *Lab. Animal Sci.*, vol. 42, no. 4, pp. 335– 343, 1992.

[34] T. Desautels *et al.*, "Parallelizing exploration–exploitation tradeoffs with Gaussian process bandit optimization," in *Proc. 29th Int. Conf. Mach. Learning*, 2012.

[35] H. Robbins, "Some aspects of the sequential design of experiments," *Bul. Am. Math. Soc.*, vol. 55, 1952.

[36] T. E. Prieto *et al.*, "Measures of postural steadiness: Differences between healthy young and elderly adults," *IEEE Trans. Biomed. Eng.*, vol. 43, no. 9, pp. 956–966, 1996.

[37] B. R. Santos *et al.*, "Reliability of centre of pressure summary measures of postural steadiness in healthy young adults," *Gait & posture*, vol. 27, no. 3, pp. 408–415, 2008.

[38] D. M. Basso *et al.*, "A sensitive and reliable locomotor rating scale for open field testing in rats," *J. of Neurotrauma*, vol. 12, no. 1, pp. 1–21, 1995.

[39] M. Antri *et al.*, "Locomotor recovery in the chronic spinal rat: Effects of long-term treatment with a 5-HT2 agonist," *Eur. J. of Neuroscience*, vol. 16, no. 3, pp. 467–476, 2002.

[40] A. J. Fong *et al.*, "Spinal cord-transected mice learn to step in response to quipazine treatment and robotic training," *J. Neurosci.*, vol. 25, no. 50, pp. 11 738–11 747, 2005.

[41] L. L. Cai *et al.*, "Implications of assist-as-needed robotic step training after a complete spinal cord injury on intrinsic strategies of motor learning," *J. Neurosci.*, vol. 26, no. 41, pp. 10 564–10 568, 2006.

[42] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Trans. ASME–J. Basic Eng.*, vol. 82, no. D, pp. 35–45, 1960.

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

[44] C. E. Rasmussen and H. Nickisch, "Gaussian processes for machine learning (GPML) toolbox," *J. Mach. Learning Res.*, vol. 11, pp. 3011–3015, 2010.

[45] J. Quiñonero-Candela and C. Rasmussen, "A unifying view of sparse approximate gaussian process regression," *J. of Mach. Learning Res.*, vol. 6, pp. 1939–1959, 2005.

[46] A. Slivkins, "Contextual bandits with similarity information," *J. Mach. Learning Res.*, vol. 15, pp. 2533–2568, July 2014.

Thomas A. Desautels received the B.S. degree in biomedical engineering from the University of California, Davis, CA, USA, and the M.S. and Ph.D. degrees from the California Institute of Technology, Pasadena, CA, USA.

He is now a visiting research associate at the Gatsby Computational Neuroscience Unit of University College London, London, UK, where his work is supported by the Whitaker International Scholarship. His research interests include neural decoding and interfaces, learning algorithms, and neuroprosthetics.

Jaehoon Choe received the B.A. degree in biological sciences from the University of Chicago, Chicago, IL, USA, and the Ph.D. degree in neuroscience from the University of California, Los Angeles, CA, USA.

He is now a postdoctoral research associate in the Information Systems Sciences Laboratory at HRL Laboratories LLC, Malibu, CA, USA. His research interests include brain-machine interfaces, neuroprosthetics, and neural modeling.

Parag Gad received the B.Eng. degree in biomedical engineering from the University of Mumbai, Mumbai, India, and the M.S. and Ph.D. degrees in Biomedical Engineering from the University of California, Los Angeles.

He is currently employed as an assistant researcher in the department of Integrative Biology and Physiology at University of California, Los Angeles. His research includes developing hardware, software tools, and stimulation strategies to facilitate locomotion and bladder control after paralysis.

Manheerej S. Nandra, biography not available at the time of publication.

Roland R. Roy received the B.S. degree in physical education from the University of New Hampshire, Durham, NH, USA, and the MS and PhD degrees in exercise physiology from Michigan Sate University, East Lansing, MI, USA.

He has been at the University of California, Los Angeles, since 1977 where he is currently a Distinguished Researcher in the Department of Integrative Biology and Physiology and the Brain Research Institute. He is the primary or co-author of over 400 research articles and book chapters.

Dr. Roy is a member of the Society for Neuroscience, American Physiological Society, and the American College of Sports Medicine.

Hui Zhong, biography not available at the time of publication.

Yu-Chong Tai (M'96-SM'03-F'06) received his B.S. degree in electrical engineering from the National Taiwan University, Taiwan, and the M.S. and Ph.D. degrees in electrical engineering from the University of California, Berkeley, Berkeley, CA, USA.

He joined the California Institute of Technology as an Assistant Professor in electrical engineering in 1989. He is the Anna L. Rosen Professor of Electrical Engineering and Mechanical Engineering and the Executive Officer of Medical Engineering at the California Institute of Technology. His research focuses on Micro-Electro-Mechanical Systems (MEMS) and implantable micro biomedical devices.

Dr. Tai was the co-chairman of the 2002 IEEE MEMS Conference. He received the 2015 IEEE Robert Bosch MEMS/NEMS Award. He is also a senior member of the American Society of Mechanical Engineers (ASME).

V. Reggie Edgerton, biography not available at the time of publication.

Joel W. Burdick received his B.S. degree in mechanical engineering from Duke University and M.S. and Ph.D. degrees in mechanical engineering from Stanford University.

He has been with the department of Mechanical Engineering at the California Institute of Technology since May 1988, where he is the Richard and Dorothy Hayman Professor of Mechanical Engineering and BioEngineering. He was appointed an IEEE Robotics Society Distinguished Lecturer in 2003. His current research interests include sensor-based robot motion planning, multi-fingered robotic hand manipulation, and spinal cord injury rehabilitation.