Computer cursor control by motor cortical signals in humans with tetraplegia

Sung-Phil Kim, John D. Simeral, Leigh R. Hochberg, John P. Donoghue and Michael J. Black

Abstract—**A direct neural interface system (NIS) promises to provide communication and independence to persons with paralysis by harnessing intact motor cortical signals to enable controlling prosthetic devices. An intracortical NIS aims to achieve this by sensing extracellular neuronal signals through chronically implanted microelectrodes and by decoding the spiking activity of neurons into prosthetic control signals. In non-human primate studies, decoding has been performed by finding a relationship between neuronal signals and actual limb movements. However, such decoding approaches face challenges in the case of paralyzed persons where there is no true movement information. Specifically, we have focused on dealing with several key questions in decoding of neural activity in humans with paralysis: what movement parameters should be decoded?; which decoding algorithms lead to more accurate estimation of movement parameters?; how do we train decoding algorithms without observing actual movement parameters?; and how many control parameters can be decoded from a single neural ensemble? In this paper, we summarize our recent studies to address these questions to improve decoding performance, which enables a human with tetraplegia to drive a 2D computer cursor to an arbitrary position and execute a "click" on the area of interest.**

I. Introduction

A neural interface system (NIS) promises to restore some lost function to persons with paralysis who suffer from "locked-in" syndrome, aiming to provide communication and independence. The fundamental concept in an NIS is to connect the intact cortical signals of a person with paralysis directly to external prosthetic devices, bypassing damaged, diseased or missing structures. An intracortical NIS aims to achieve this by sensing extracellular neural signals in the motor cortex using chronically implanted microelectrodes and translating those signals into motor control parameters to actuate prosthetic devices. A pilot clinical study for feasibility of a human NIS began in 2004 (Investigational Device Exemption). The first report from this study by Hochberg *et al.* [1] demonstrated that an intracortical NIS could detect movement-related signals from the motor cortex of a person with spinal cord injury years after injury and that these signals could be used for voluntary control of external devices, including a robotic hand, a computer cursor and other physical and virtual devices. In particular, computer cursor control has been a primary application since it provides a foundation for a variety of direct communication tools.

Central to this study and underlying the effective functioning of the NIS, is the understanding of the coding of movement-related information in the activity of cortical neurons. Spikes (or action potentials) are known to be an important information signal of neurons and spike activity can be closely related to movement parameters, such as movement speed or direction. A number of studies in non-human primates have explored how neural spiking activity is modulated when planning, executing and adjusting limb control. However, neural control for prosthetic devices without knowing true limb movements raises a new question: how can we *decode* the observed neural spiking activity only with the unobservable intention of movement? This question leads to several issues in the development of an NIS for paralyzed persons: Which kinematic parameters in the imagined movement are most naturally represented in neuronal ensemble activity? Which mathematical algorithms should we use to decode neural spiking activity? How can we generate training samples only with imagined movements to optimize the parameters of decoding algorithms? How many control variables can we extract from a single neuronal ensemble? In this paper, we summarize how we addressed some of these issues and what we have found from the study of decoding neural activity in humans with tetraplegia for prosthetic control.

Previous non-human primate studies have shown that motor cortical neurons encode various movement parameters,

Manuscript received January 31, 2009. This work was supported in part by MGH-Deane Institute, NIH/NINDS-Javits (NS-25074)/NICHD-NCMRR (N01-HD-53403)/NIBIB, Dept. of Veterans Affairs, Rehabilitation R&D Service, ONR (N00014-07-1-0803), Cyberkinetics Neurotechnology Systems, Inc., Doris Duke Charitable Foundation, Katie Samson Foundation, National Center for Research Resources (NCRR C06-16549-01A1). Disclosures: LRH: Clinical trial support, Cyberkinetics Neurotechnology Systems, Inc. (CYKN); JDS: Consultant, CYKN; JPD: prior Chief Scientific Officer and director, stock holdings, CYKN.

S.-P. Kim and M. J. Black are with the Dept. of Computer Science, Brown Univ., Providence, RI, USA (phone: 401-863-7600; fax: 401-863-7657; e-mail: (spkim@ cs.brown.edu).

J. D. Simeral is with the Center for Restorative and Regenerative Medicine, Rehabilitation R&D Service, Dept. of Veterans Affairs Medical Center, Providence RI, USA; and the Dept. of Neuroscience, Brown Univ., Providence, RI, USA (e-mail: John_Simeral@brown.edu).

L. R. Hochberg is with the Center for Restorative and Regenerative Medicine, Rehabilitation R&D Service, Dept. of Veterans Affairs Medical Center, Providence RI, USA; Dept. of Neuroscience, Brown Univ., Providence, RI, USA; and Massachusetts General Hospital, Harvard Medical School, Boston, MA, USA (e-mail: Leigh_Hochberg@brown.edu).

J. P. Donoghue is with the Dept. of Neuroscience, Brown Univ., Providence, RI, USA and Cyberkinetics Neurotechnology Systems, Inc., Foxborough, MA, USA (e-mail: John_Donoghue@brown.edu).

including hand position, hand velocity, limb force and joint torques [2-9]. We have studied how some of these movement parameters were correlated with neural activity when a person imagined cursor movement. We have found that cursor velocity was more correlated with neural activity than other parameters such as position and acceleration in two participants [10].

A number of mathematical algorithms have been developed to decode movement parameters from neural activity (see [11-12] for review). In particular, some studies suggested that the decoding algorithms based on probabilistic Bayesian inference estimated arm/hand kinematic parameters better than simple direct estimators such as linear filters [13-16]. Accordingly, we used the Kalman filter [13-14], to decode the movement of a computer cursor from neural activity in humans with tetraplegia. We have demonstrated that the Kalman filter improved decoding of cursor velocity when compared with a linear filtering method [10].

In non-human able-bodied primate studies, decoding algorithms can be trained to associate neural activity with limb movements using training samples generated by simultaneously recording both movement and neural signals. In the study of humans with paralysis, however, training of a decoding algorithm must be achieved in the absence of physical movement. In Hochberg *et al.* [1], a combination of open-loop and closed-loop training methods was used that enabled subjects to gain control of a "neural cursor". In this training approach, a subject viewed a cursor moving on a monitor and was instructed to *imagine* moving the cursor. Then the cursor movement kinematics and synchronized neural signals were used to train a decoding algorithm. We modified this procedure by adding a new training task for velocity-based decoding algorithms, emphasizing velocity in the design of cursor motion [10]. We also extended it to generate new training data that were used for training a multi-state decoder [19].

A practical neural cursor interface should provide the utility of a computer mouse. This combines the ability to move a cursor to an arbitrary screen location, hold the cursor still, and execute a click action to make a selection at that location. A central question to point-and-click cursor control is whether multi-state signals (i.e. both a *continuous* state for pointing and a *discrete* state for clicking) can be simultaneously decoded from a single neuronal ensemble. Based on non-human primate studies that demonstrated multi-state signals could be decoded from the motor cortical activity of monkeys [17-18], we have modified a previous probabilistic decoding model [18] to make it feasible for real-time performance. Our recent study has shown that we could successfully decode both continuous and discrete states from a single neuronal ensemble in a human [19]. We have demonstrated in this study that a human with tetraplegia could use the NIS with the multi-state decoder to point to and click on targets.

In the remainder of this paper, we present more details of the issues and findings in decoding neural activity of humans with paralysis. In the section III, we illustrate training methods for the decoding algorithms. In section IV, we describe the correlation between neural activity and imagined cursor movements. In the section V and VI, we review the decoding algorithms and demonstrate closed-loop neural cursor control performance using each of the decoding algorithms. In the section VII, we describe how we enabled point-and-click cursor control. In the final section, we summarize on-going challenges and how we will address them.

II. Neural Interface System Pilot Study

A. Pilot Study

A pilot clinical study of the BrainGate Neural Interface System was initiated by Cyberkinetics Neurotechnology Systems, Inc. under a Food and Drug Administration (FDA) Investigational Device Exemption (IDE) and with Institutional Review Board (IRB) approvals; the studies began in May, 2004. "Caution: Investigational Device. Limited by Federal Law to Investigational Use"

B. Participants

Clinical trial sessions of the BrainGate NIS (Cyberkinetics Neurotechnology Systems, Inc.) were conducted by Cyberkinetics technicians with two participants with tetraplegia (paralysis of both arms and both legs). Participant S3 was a 54 year old woman who had thrombosis of the basilar artery and extensive pontine infarction nine years prior to trial recruitment. Participant A1 was a 37 year old man with amyotrophic lateral sclerosis (ALS, motor neuron disease), recruited to the trial six years after being diagnosed with ALS. Both participants were right hand dominant, and the intracortical array was placed in the left precentral gyrus in the region of the arm representation [1].

C. Recording Sessions

During each recording session, neural signals were recorded from the motor cortex of the participants using a chronically-implanted 96-channel Cyberkinetics microelectrode array and the BrainGate NIS. After digitization (30 kHz per channel), real-time, amplitude-thresholding software was used to identify different waveshapes on each channel [20]. Single neurons and multi-neuron activity with consistent waveforms [20] (both referred to here as 'units') were accepted or rejected for inclusion in the study at the beginning of each session based on visual inspection of the isolated waveforms. No further criteria were applied to identify single neurons. During neural recording, participants viewed a computer monitor that displayed task information related to various cursor control tasks as described below.

III. Training Decoding Algorithms

The overall training procedure was composed of a series of short periods, called "blocks," each of which lasted 1-1.5 min. We devised two types of training blocks: open-loop (OL) and closed-loop (CL). In OL blocks, a training cursor (TC) was displayed on a computer monitor and moved to reach a target, generating cursor movement trajectories. The training cursor was moved either manually by a technician or automatically by a computer program. During the presentation of the TC movement on the monitor, the participants were instructed to imagine moving their dominant arm or hand as if they were moving the TC. Following the execution of multiple OL blocks, we trained a decoding algorithm using neural signals together with the TC kinematic data. Afterwards, in CL blocks, we presented not only the TC but also a feedback cursor (FC) whose movement was decoded from the participant's neural activity using the previously trained decoding algorithm. We hypothesized that showing the FC would prompt the participant to adjust their neural activity to improve cursor control by sensing the error between the TC and the FC. The decoding algorithm was iteratively trained after every other CL block to incorporate the potential adjustment of neural activity into the parameter estimation.

We utilized two different cursor movement tasks depending on the decoded kinematic parameter: a random pursuit-tracking task, which can generate a wide range of cursor position data, was used for cursor position decoding and a center-out-and-back task, which can provide well-defined cursor velocity data, was used for cursor velocity decoding. In the latter case, the cursor speed followed a bell-shaped profile. Figure 1 illustrates the training procedure and tasks.

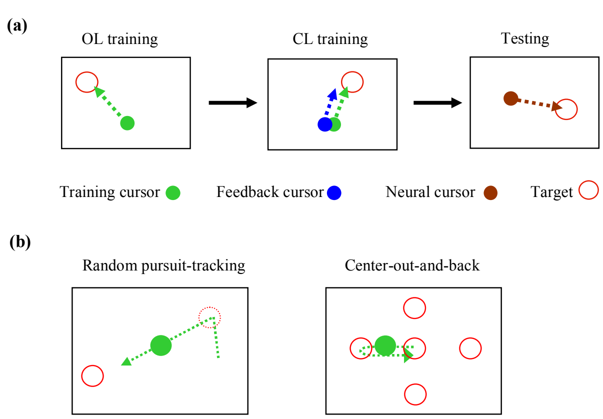

Fig. 1. (a) Training and testing procedure for human neural interface systems (NISs). A neural cursor control trial session is composed of training and testing phases. The training phase is further divided into open-loop (OL) and closed-loop (CL) training. In OL training, the training cursor (TC) is moved on the monitor by a technician or a computer program to reach a target. In CL training, a feedback cursor (FC) is shown together with the TC to provide feedback of how well cursor movement is decoded from neural activity. In the testing phase, only a neurally controlled cursor (NC) is shown and moved by the participant to acquire a target. (b) Cursor movement tasks. A random pursuit-tracking task is used to generate training data for position decoding algorithms and a center-out-and-back task is used for velocity decoding algorithms. (Kim *et al.* [10]).

IV. Neural Correlation with Cursor Kinematics

The correlation between neural firing rates and cursor kinematic parameters, including position and velocity, was measured for every neuronal unit recorded during the open-loop (OL) training blocks. The OL blocks were chosen because neural activity was directly related to visualized cursor kinematics and not associated with any particular decoding algorithm. We evaluated correlation using the Pearson correlation coefficient (CC) which provides a direct measure without employing an explicit model to find a mapping between firing rates and kinematic parameters. In computation of the CC, we searched over multiple time lags between each unit's firing rate and the kinematic parameter and found the optimal lag yielding the maximum CC value.

From the CC measures across multiple recording sessions in S3 and A1 (17 sessions, >1000 units), we found that neuronal correlation with cursor velocity was stronger than with cursor position, more units showed correlation with velocity, and >90% of all the recorded units were statistically correlated with at least one of velocity or position [10]. Figure 2 shows some results of this correlation analysis. These results suggest that decoding cursor velocity, which is more correlated with neural activity, may yield better neural cursor control than decoding cursor position.

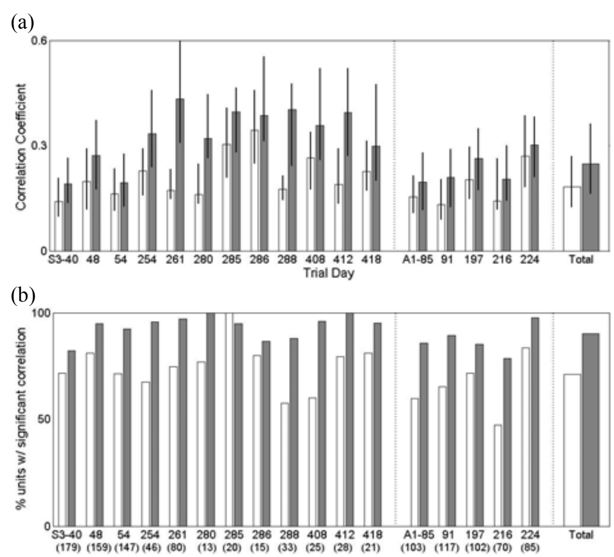

Fig. 2. (a) Correlation coefficients between individual units' firing rates and cursor kinematic parameters – position (white) or velocity (gray). The median with 25% and 75% percentiles are presented for each recording session. (b) The number of units showing significant correlation ($p < 0.01$, t-test) with cursor velocity or position (Kim et al. 2008). Trial day indicates the day of recording session after implantation. (Kim *et al.* [10])

V. Decoding Algorithms

A decoding algorithm built for an NIS extracts the movement intention of a subject from their neural activity and converts it to the kinematic parameters required to control a prosthetic device. Such decoding algorithms draw on mathematical models describing the (causal) relationship between neural activity and kinematics. Most decoding algorithms have their own parameters which need to be determined through a training procedure. Here, training refers to a process through which, a decoding algorithm, or more precisely the parameters of the algorithm, are trained to model a neural motor mapping. In this paper, we compare two decoding algorithms, the linear filter and the Kalman filter, both of which have been widely used in many human and non-human brain-computer interface studies. We derive these two algorithms from a probabilistic framework and illustrate main differences between them.

Suppose $\mathbf{x}_t$ is a D×1 vector of cursor kinematics such as position $[p_x, p_y]^T$ or velocity $[v_x, v_y]^T$ in the x (horizontal) and y (vertical) coordinates, sampled at a discrete time instant t. Let $\mathbf{z}_t$ be an N×1 vector of firing rates of N units. The firing rate of a unit was estimated as the number of spikes in a fixed-size time window (e.g. 100ms) in our study. A goal in decoding is to find the most probable value of $\mathbf{x}_t$ when we observe the entire history of firing rates, $\mathbf{z}_{1:t-j}$ where j denotes a time lag between $\mathbf{x}_t$ and $\mathbf{z}_t$. This can be represented as modeling a conditional probability, $p(\mathbf{x}_t \mid \mathbf{z}_{1:t-j})$.

The linear filter achieves this goal based on the maximum likelihood estimation with the assumptions of a linear mapping and white noise:

$$\mathbf{x}_t = \mathbf{a}_0 + \sum_{i=1}^{t-j} \mathbf{a}_i^T \mathbf{z}_i + \boldsymbol{\varepsilon}_t = \mathbf{a}_0 + \sum_{i=t-j-L}^{t-j} \mathbf{a}_i^T \mathbf{z}_i + \boldsymbol{\varepsilon}_t . \qquad (1)$$

Here we assume that $\mathbf{x}_t$ is correlated only with L past values of $\mathbf{z}_{t-j}$. Since $\boldsymbol{\varepsilon}_t$ is regarded as a Gaussian random variable with zero mean and variance of σ^2, (1) is rewritten as,

$$p\left(\mathbf{x}_t \mid \mathbf{z}_{t-j-L:t-j}\right) = G\left(\mathbf{x}_t - \mathbf{a}_0 - \sum_{i=t-j-L}^{t-j} \mathbf{a}_j^T \mathbf{z}_i ; 0, \sigma^2\right), \qquad (2)$$

where $G(\tau; \mu, \sigma^2)$ represents a Gaussian distribution of τ with mean μ and variance σ^2. The parameters, $\{\mathbf{a}_i\}$ are learned using maximum likelihood estimation.

The Kalman filter models $p(\mathbf{x}_t \mid \mathbf{z}_{1:t-j})$ using a Bayesian formulation,

$$p(\mathbf{x}_t \mid \mathbf{z}_{1:t-j}) = \frac{1}{\kappa} p(\mathbf{z}_{t-j} \mid \mathbf{x}_t) \int p(\mathbf{x}_t \mid \mathbf{x}_{t-1}) p(\mathbf{x}_{t-1} \mid \mathbf{z}_{t-1-j}) d\mathbf{x}_{t-1} \qquad (3)$$

where κ is a normalization constant. $p(\mathbf{x}_t \mid \mathbf{z}_{1:t-j})$ is inferred by finding the conditional mean of $p(\mathbf{x}_t \mid \mathbf{x}_{t-1})$ in terms of the previously estimated $p(\mathbf{x}_{t-1} \mid \mathbf{z}_{t-1-j})$ and updating it by $p(\mathbf{z}_{t-j} \mid \mathbf{x}_t)$ which carries information of newly observed $\mathbf{z}_{t-j}$. In the Kalman filtering algorithm, $p(\mathbf{x}_t \mid \mathbf{x}_{t-1})$ and $p(\mathbf{z}_{t-j} \mid \mathbf{x}_t)$ are approximated as linear Gaussian models. The parameters in these models are learned from training data using least squares.

A key difference between these two filters is that the Kalman filter models the prior probability of $\mathbf{x}_t$ in $p(\mathbf{x}_t \mid \mathbf{x}_{t-1})$ whereas the linear filter does not. This prior model may be advantageous to decoding for NISs especially when we need to devise movement models without the knowledge of true movements.

VI. Cursor Control Improvement by Kalman Velocity Decoding

In this section, we demonstrate how neural cursor control was improved by choosing to decode cursor velocity ($\mathbf{x}_t = [v_x, v_y]^T$) over position and choosing to use the Kalman filter over the linear filter. We evaluated cursor control performance in a four-target acquisition task across 14 recording sessions in S3 and A1. The participants were asked to move the neural cursor (NC) from center to one of four radially located targets and to hold the NC on the target for 500ms. The participants performed this target acquisition trial approximately 80 times each session. In each trial, the participants had to acquire a target within 7s or the trial was deemed a failure.

Figure 3 illustrates neural cursor control performance using the linear position filter versus the Kalman velocity filter. Cursor movement decoded with the Kalman velocity filter was much smoother and straighter than that decoded with the linear position filter. We quantified cursor control performance by a number of measures, including the number of directional changes and deviation of the cursor path from a straight line trajectory. These measures confirmed that the Kalman velocity filter exhibited fewer directional changes and smaller deviations than the linear position filter [10]. In addition, the participants demonstrated more accurate target acquisition performance with the Kalman velocity filter, increasing the acquisition rate by ~10%. The time taken to reach the target was similar for both filters. In fact, the cursor

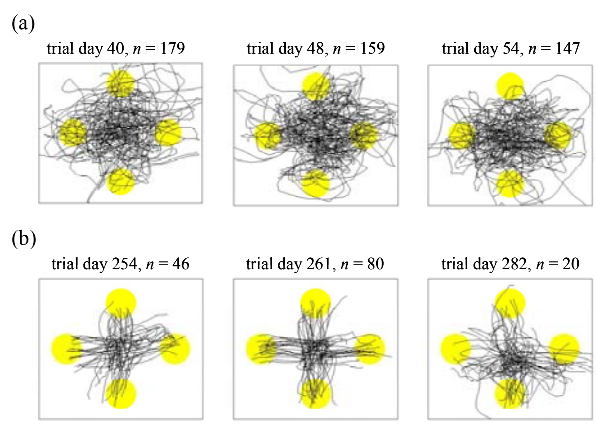

Fig. 3. Neural cursor trajectories during four-target center-out task. Targets were acquired when the cursor dwelled on it for > 0.5s. Each line shows a cursor trajectory starting from the center of the monitor to each of four targets (yellow squares). n denotes the number of units used in the decoding filter. (a) Examples of three sessions where the neural cursor **position** was decoded by the **linear** filter (b) Examples of three sessions where the neural cursor **velocity** was decoded by the **Kalman** filter. (Kim *et al.* [10])

decoded by the linear position filter generally moved faster but its erratic and curved motion meant it take long to reach and hold on the target.

We further examined whether the performance improvement of the Kalman velocity filter over the linear position filter was achieved due to the choice of kinematic parameter or the choice of decoding algorithm. The participant S3 performed two center-out tasks, one with the linear velocity filter and the other with the Kalman velocity filter, in a single session (performed three times over multiple days). The results demonstrated that the linear velocity filter generated smoother and more stable cursor movements than the linear position filter, whereas the Kalman velocity filter yielded slightly better performance than the linear velocity filter. This suggests that choosing to decode cursor velocity may contribute more to cursor control improvement than choosing to use the Kalman filter.

VII. Point and Click Cursor Control

To enable "point-and-click" cursor control from neural activity, we have addressed a key question of whether it is possible to decode multiple states (i.e. a continuous pointing state and a discrete click state) simultaneously from a single neural population.

We first revised the above training method by adding a new training phase for click states [19]. In this phase, the participant was asked to imagine a specific arm/hand movement (e.g. squeezing the hand) whenever a click cue was shown on the monitor.

Next, we developed a new multi-state decoding algorithm based on the previous work by Wood *et al.* [18]. We simplified the probabilistic multi-state model proposed in [18] to be suitable for real-time applications. We represented the continuous cursor state as cursor velocity and the discrete state as one of two classes, {*movement*, *click*}. We used the Kalman filter to decode cursor velocity and a linear classifier to decode the discrete state. The Kalman filter was trained using the center-out task training samples (see Section III). The linear classifier was trained using the data collected during the click training phase described above. After training, multi-state decoding worked as follows: when the movement class was decoded from the classifier, the velocity signal from the Kalman filter was used to drive the cursor; when the click class was decoded, the cursor was forced to stop with zero velocity and a click signal was generated [19].

We tested whether the multi-state decoding algorithm and the revised training method could lead to point-and-click operation of a computer cursor in humans with tetraplegia [19]. Over multiple NIS study sessions, the participant (S3) performed a closed-loop target acquisition task using the click function generated by multi-state decoding. In this task, one of eight radially located targets was highlighted and the participant moved the neural cursor toward the target and clicked on the target to acquire it within a specified maximum time (e.g. 9s). Over three recording sessions, S3 successfully acquired ~97% of the targets by clicking on them, with errors occurring only due to time limit. There were no false clicks on incorrect targets. False clicks on non-target screen regions were generated on average less than once per trial; many of these occurred close to the target. Note that it was not possible to determine whether these false clicks were generated as a result of inaccurate discrete state decoding or because that the participant intended to click but the cursor was not correctly positioned on the target. Figure 4 illustrates neural cursor trajectories across three point-and-click sessions [19].

VIII. Future Work

We have summarized progress in neural decoding for intracortical NISs, enabling humans with tetraplegia to control arbitrary point-to-point movements of a computer cursor and click on specified target areas. There remain a number of scientific and engineering issues that need to be resolved in order to develop a more reliable and useful NIS. Here we discuss several on-going issues particularly related to neural decoding and our approaches to address them.

First, only a few simple decoding models have been adopted so far for NISs, largely due to the fact that they are easily implemented in real time. These models assume a linear relationship between neural activity and kinematic parameters which does not generally hold when representing complex and nonlinear neural ensemble activity. There have been several studies showing that nonlinear decoding models produced more accurate estimates of kinematics from motor cortical signals [21-23]. Therefore, we are developing a nonlinear decoding model that better describes the relationship between neural and kinematic data and also runs effectively in a real-time computing system. Among many nonlinear approaches, we are particularly exploring the nonlinear representation and decoding of neural activity using Gaussian processes [24], nonlinear dynamic system models [25] and particle filters [15].

Second, the current decoding models and training paradigms have not explicitly focused on decoding cursor speed. This may account for the somewhat slow cursor movement seen in our experiments. This problem is partially associated with the aforementioned linearity assumption. For

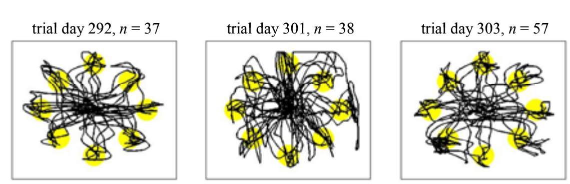

Fig. 4. Neural cursor trajectories with point-and-click cursor control. Targets were acquired when the participant clicked on the target with their neural activity. Black lines are cursor trajectory from center to each of eight targets and yellow circles are targets. n denotes the number of units (Kim *et al.* [19])

the Kalman filter case, the linear Gaussian model used to describe how movement changes in time appears to be too simple to represent cursor dynamics. Consequently, the velocity profile of the neurally controlled cursor movement is different from the idealized ballistic speed profile which is generally observed in real arm movements. To address this issue, we are seeking a new probabilistic movement prior model that may effectively represent cursor dynamics, resulting in cursor motion more responsive to the intention of a user.

Finally, the current paradigm of training and using decoding models freezes the model parameters after training is finished. This paradigm assumes that the statistical and encoding properties of neural signals do not change significantly before and after the end of training. However, decoding performance could decrease if any of those properties significantly change after training. Our group has shown in an off-line non-human primate data analysis that the parameters of a linear function relating neural firing rates to kinematic parameters significantly changed over short-time periods [26]. We seek to address this non-stationarity by developing an adaptive decoding filter that constantly adjusts its parameters according to the change in the statistical and encoding properties of the neural signals. We believe that this adaptive decoding filter will enable NIS cursor control that is more robust to the non-stationarity of neural signals.

Acknowledgment

We thank Gregory Shakhnarovich, Frank Wood and Wilson Truccolo for comments and assistance. We also thank Abe Caplan, Eric Peterson, Michael Fritz and Cyberkinetics Neurotechnology Systems, Inc. for running the recording sessions described here.

References

[1] L.R. Hochberg *et al.*, "Neuronal ensemble control of prosthetic devices by a human with tetraplegia," *Nature,* vol. 442, pp. 164-171, 2006.

[2] D.R Humphrey, D.M. Schmidt and W.D. Thompson, "Predicting measures of motor performance from multiple cortical spike trains," *Science* vol. 170, pp. 758-762, 1970.

[3] A.P. Georgopoulos, J.F. Kalaska, R. Caminiti and J.T. Massey, "On the relations between the direction of two-dimensional arm movements and cell discharge in primate motor cortex," *J. Neurosci.* vol. 2, pp. 1527-1537, 1982.

[4] J. Ashe and A.P. Georgopoulos, "Movement parameters and neural activity in motor cortex and area 5," *Cereb. Cortex* vol. 4, pp. 590-600, 1994.

[5] D.W. Moran and A.B. Schwartz, "Motor cortical representation of speed and direction during reaching," *J. Neurophysiol.* vol. 82, pp. 2676-2692, 1999.

[6] L.E. Sergio and J.F. Kalaska, "Changes in the temporal pattern of primary motor cortex activity in a directional isometric force versus limb movement task," *J. Neurophysiol.* vol. 80, pp. 1577-1583, 1998.

[7] E. Todorov, "Direct cortical control of muscle activation in voluntary arm movements: a model," *Nat. Neurosci.* vol. 3, pp. 391-398, 2000.

[8] J.P. Donoghue, J.N. Sanes, N.G. Hatsopolous and G. Gaal, "Neural discharge and local field potential oscillations in primate motor cortex during voluntary movements" *J. Neurophyiol.* vol. 79, pp. 159-173, 1997.

[9] L. Paninski, M.R. Fellows, N.G. Hotsopoulos and J.P. Donoghue, "Spatiotemporal tuning of motor cortical neurons for hand position and velocity," *J. Neurophysiol.* vol. 91, pp. 515-532, 2004.

[10] S.-P. Kim, J.D. Simeral, L.R. Hochberg, J.P. Donoghue and M.J. Black, "Neural control of computer cursor velocity by decoding motor cortical spiking activity in humans with tetraplegia," *J Neural Engineering* vol. 5, pp. 455-476, 2008.

[11] M.M. Churchland, B.M. Yu, M Sahani and K.V. Shenoy, "Techniques for extracting single-trial activity patterns from large-scale neural recordings," *Curr. Opinion in Neurobiol.* vol. 17, pp. 609-618, 2007.

[12] A.B. Schwartz, D.M. Taylor and S.I. Helms Tillery, "Extraction algorithms for cortical control of arm prosthetics," *Curr. Opinion in Neurobiol.* Vol. 11, pp. 701-707, 2001.

[13] W. Wu, Y. Gao, E. Bienenstock, J.P. Donoghue and M.J. Black, "Bayesian population decoding of motor cortical activity using a Kalman filter," *Neural Comput.* Vol. 18, pp. 80-118, 2006.

[14] W. Wu, M.J. Black, Y. Gao, E. Bienenstock, M. Serruya, A. Shaikhouni and J.P. Donoghue, "Neural decoding of cursor motion using a Kalman filter," *Adv. in Neural Info. Processing Sys. 15*, S. Becker, S. Thrun and K. Obermayer (Eds.), MIT Press, pp. 133-140, 2003.

[15] A.E. Brockwell, A.L. Rojas and R.E. Kass, "Recursive Bayesian decoding of motor cortical signals by particle filtering." *J. Neurophysiol.* vol. 91, pp.1899-1907, 2004.

[16] Y. Gao, M.J. Black, E. Bienenstock, S. Shoham and J.P. Donoghue, "Probabilistic inference of arm motion from neural activity in motor cortex," *Adv. in Neural Info. Processing Sys. 14*, T.G. Dietterich, S. Becker and Z. Ghahramani (Eds.), MIT Press, pp. 221-228, 2002.

[17] S. Darmanjian *et al.*, "Bimodal brain-machine interfaces for motor control of robotic prosthetics," in *Proc. 2003 IEEE/RSJ Int'l Conf. on Intelligent Robots and Systems*, vol. 4, Las Vegas, NV, pp. 3612-17.

[18] F. Wood, Probhat, J.P. Donoghue and M.J. Black, "Inferring attentional state and kinematics from motor cortical firing rates," in *Proc. 2005 Conf. of IEEE EMBS*, Shanghai, China, pp. 1544-1547.

[19] S.-P. Kim, J.D. Simeral, L.R. Hochberg, J.P. Donoghue, G.M. Friehs and M.J. Black, "Multi-state decoding of point-and-click control signals from motor cortical activity in a human with tetraplegia," *Proc. IEEE EMBS Conf on Neural Engineering*, Hawaii, USA, pp. 486-9, 2007.

[20] S. Suner, M.R. Fellows, C. Vargas-Irwin, G.K. Nakata and J.P. Donoghue, "Reliability of signals from a chronically implanted, silicon-based electrode array in non-human primate primary motor cortex," *IEEE Trans. Neural Systems and Rehab. Eng.* vol. 13, pp. 524-541, 2005.

[21] J.C. Sanchez, D. Erdogmus, J.C. Principe, J. Wessberg and M.A.L. Nicolelis, "Interpreting spatial and temporal neural activity through a recurrent neural networks brain machine interface" *IEEE Trans. Neural Systems and Rehab. Eng.* vol. 13 pp. 213-219, 2005.

[22] L. Shpigelman, Y. Singer, R. Paz and E. Vaadia, "Spikernels: Predicting arm movements by embedding population spike rate patterns in inner-product spaces." *Neural Computation* vol. 17, pp. 671-690, 2005.

[23] S.P. Kim, J.C. Sanchez, Y.N. Rao, D. Ergodmus, J.C. Principe, J.M. Carmena, M.A. Levedev and M.A.L. Nicolelis, "A comparison of optimal MIMO linear and nonlinear models for brain-machine interfaces," *J Neural Engineering* vol. 3, pp. 145-161, 2006.

[24] J.P. Cunninghan, M. Sahani and K.V. Shenoy, "Fast Gaussian process methods for point process intensity estimation," *Proc. 25th Ann. Int'l Conf. Machine Learning (ICML 2008)*, Helsinki, Finland, pp. 192-199, 2008.

[25] Y.N. Rao, S.-P. Kim, J.C. Sanchez, D. Erdogmus, J.C. Principe, J.C. Carmena, M.A. Lebedev and M.A.L. Nicolelis, "Learning mappings in brain-machine interfaces with echo state networks," *IEEE Int'l Conf. Acoustic, Speech and Sig. Processing*, Philadelphia, PA, 2005.

[26] S-.P. Kim, F. Wood, M.R. Fellows, J.P. Donoghue and M.J. Black, "Statistical analysis of the non-stationarity of neural population codes," *Proc. IEEE / RAS-EMBS Int'l Conf. on Biomedical Robotics and Biomechatronics,* pp. 811-816, Pisa, Italy, 2006.