High-Performance Brain-Machine Interface Enabled by an Adaptive Optimal Feedback-Controlled Point Process Decoder

Maryam M. Shanechi[1], *Member, IEEE,* Amy Orsborn[2], Helene Moorman[3], Suraj Gowda[4], and Jose M. Carmena[5], *Senior Member, IEEE*

***Abstract*— Brain-machine interface (BMI) performance has been improved using Kalman filters (KF) combined with closed-loop decoder adaptation (CLDA). CLDA fits the decoder parameters during closed-loop BMI operation based on the neural activity and inferred user velocity intention. These advances have resulted in the recent ReFIT-KF and SmoothBatch-KF decoders. Here we demonstrate high-performance and robust BMI control using a novel closed-loop BMI architecture termed adaptive optimal feedback-controlled (OFC) point process filter (PPF). Adaptive OFC-PPF allows subjects to issue neural commands and receive feedback with every spike event and hence at a faster rate than the KF. Moreover, it adapts the decoder parameters with every spike event in contrast to current CLDA techniques that do so on the time-scale of minutes. Finally, unlike current methods that rotate the decoded velocity vector, adaptive OFC-PPF constructs an infinite-horizon OFC model of the brain to infer velocity intention during adaptation. Preliminary data collected in a monkey suggests that adaptive OFC-PPF improves BMI control. OFC-PPF outperformed SmoothBatch-KF in a self-paced center-out movement task with 8 targets. This improvement was due to both the PPF's increased rate of control and feedback compared with the KF, and to the OFC model suggesting that the OFC better approximates the user's strategy. Also, the spike-by-spike adaptation resulted in faster performance convergence compared to current techniques. Thus adaptive OFC-PPF enabled proficient BMI control in this monkey.**

I. Introduction

Brain-machine interfaces (BMI) have demonstrated that human and non-human primates can use their motor cortical activity to control computer cursors or robotic arms (e.g., [1], [2], [3], [4], [5], [6], [7], [8], [9]). BMIs record a subject's neural activity, use a decoder to infer from this activity the subject's motor intent and control a device, and provide visual feedback to the subject. Various decoders such as linear regression, population vector, and Kalman filters (KF) have been used in real-time BMIs. In addition to selecting a decoding model, the model's parameters need to be found for each subject. This parameter fitting is often performed in open loop by recording the neural activity in a training session while subjects execute or imagine movements, and then solving for optimal parameter values based on the observed neural activity in response to movement. However, recent studies have shown that fitting the decoder parameters in closed-loop BMI operation can improve its performance [5], [6]. This is due to the change in neural representations when subjects control a BMI compared to when they control their arm [1], [2], [10].

Fitting the decoder parameters in closed-loop is referred to as closed-loop decoder adaptation (CLDA) [1], [3], [5], [6], [7], [11]. Recent work has combined CLDA with KF to develop the ReFIT-KF decoder and achieve proficient BMI control [5], [6]. In ReFIT-KF, parameters are first initialized based on arm reaching movements [5], visual feedback of cursor movements [6], or even arbitrarily [6]. The initialized KF is then used by the subject to make brain-controlled movements towards targets on the computer screen. In the process, KF parameters are refit by collecting batches of neural activity and inferring the subject's intended velocity in these batches. ReFIT-KF finds the intended velocity at each time by rotating the cursor's decoded velocity vector towards the target while keeping its magnitude unchanged, and by equating it to zero when at the target. This intention estimation method is termed CursorGoal [5]. Recently, SmoothBatch-KF [6] that uses ReFIT-KF and CursorGoal was proposed. In SmoothBatch-KF parameters are adapted smoothly once every 90 sec and converge to a good solution even when not initialized using arm movements.

Here we develop a novel closed-loop BMI architecture termed adaptive optimal feedback-controlled (OFC) point process filter (PPF) and show that it enables robust and high-performance spike-by-spike BMI control. The architecture is developed based on a point process model of spikes and an optimal feedback control model of brain that we have proposed previously [12], [13], [14], [8] and used for joint decoding of target and trajectory in the specific case of target-directed movements and when decoder parameters were found in open loop. Here we extend our decoding framework to develop adaptive OFC-PPF that fits the decoder parameters in closed-loop and is not specific to target-directed movements. Adaptive OFC-PPF updates the decoder parameters with every quasi-spike event (every 5ms) unlike ReFIT-KF and SmoothBatch-KF that do so on the time-

[1]M. M. Shanechi is with the School of Electrical and Computer Engineering, Cornell University, Ithaca, NY 14853 USA. She was previously with the Department of Electrical Engineering and Computer Science, University of California, Berkeley, CA 94720 USA. shanechi@cornell.edu.

[2]Amy Orsborn is with the UCB-UCSF Graduate Group in Bioengineering, University of California, Berkeley, CA 94720 USA amyorsborn@berkeley.edu.

[3]Helene Moorman is with the Helen Wills Neuroscience Institute, University of California, Berkeley, CA 94720 USA helenem@berkeley.edu.

[4]Suraj Gowda is with the Department of Electrical Engineering and Computer Science, University of California, Berkeley, CA 94720 USA surajgowda@berkeley.edu.

[5]J. M. Carmena is with the Department of Electrical Engineering and Computer Science, the Helen Wills Neuroscience Institute, and the UCB-UCSF Graduate Group in Bioengineering, University of California, Berkeley, CA 94720 USA carmena@eecs.berkeley.edu.

978-1-4244-7929-0/14/$26.00 ©2014 IEEE

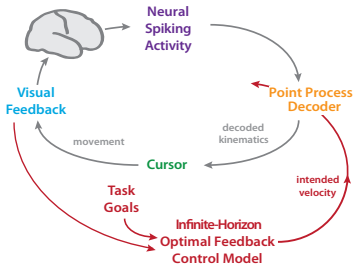

Fig. 1. Adaptive OFC-PPF architecture.

scale of minutes. Moreover, it allows subjects to issue neural commands and receive feedback at a faster rate (every 5ms) compared to KF (typically every 50–100 ms). Finally, unlike CursorGoal, it models the brain in closed-loop BMI control as an infinite-horizon optimal feedback-controller to infer velocity intention during adaptation and to devise an assisted training paradigm for consistent parameter convergence. We compare performance across different decoders within the same monkey in a self-paced center-out movement task with 8 targets. We show that spike-by-spike adaptation results in faster performance convergence compared to current batch-based methods. Our preliminary data from this monkey also suggests that adaptive OFC-PPF results in higher performance compared to SmoothBatch-KF. We show that this improvement is due to both the PPF's fast rate of control and feedback and to the OFC intention estimation suggesting that it better approximates user's strategy.

II. Methods

We develop a new closed-loop BMI architecture, adaptive OFC-PPF (Fig. 1), based on an optimal feedback control model of the brain and a point process model of the spikes. We first build the OFC and the PPF models and then show how to combine them to develop the architecture.

A. Infinite-horizon OFC model for CLDA

A BMI system can be modeled as an optimal feedback control system [12], [13], [14], [8]. When controlling the BMI, the brain (controller) decides on the next neural command based on the current state of the cursor and the task goal to reach the desired target. Hence we can build an optimal feedback control model of the brain to predict its control commands. This model can be constructed by defining an approximate forward dynamics model, quantifying the task goals as cost functions, and modeling the visual feedback. Our BMI architecture uses this OFC design to infer subject's intended kinematics in the presence of poor parameter estimates and to adaptively update these estimates.

We denote the sequence of kinematic states by $\mathbf{x}_0, \cdots, \mathbf{x}_t$ and assume that they evolve in the linear dynamical system

$$\mathbf{x}_t = \mathbf{A}\mathbf{x}_{t-1} + \mathbf{B}\mathbf{u}_{t-1} + \mathbf{w}_{t-1}. \quad (1)$$

This is the forward dynamics model with parameters $\mathbf{A}$ and $\mathbf{B}$ that we fit based on manual trajectories. Here $\mathbf{u}_t$ is the control command at time t that the brain (i.e., controller) issues, and $\mathbf{w}_t$ is a zero-mean white Gaussian state noise with covariance matrix $\mathbf{W}$. We denote the decoded cursor kinematics, which is rendered on the screen at time t, by $\mathbf{x}_{t|t}$. We assume that the subject can perfectly observe $\mathbf{x}_{t|t}$ (i.e., noiseless visual feedback). We also implicitly assume that the brain has formed an internal forward model of the dynamics of movement in response to control commands $\mathbf{u}_t$ in the task [8]. Studies using motor control tasks [15] and more recently using BMI tasks [16] have suggested the existence of such an internal model.

To predict the brain's intended control command and consequently the intended kinematics, we form a cost function that quantifies the task goal, and minimize it over $\mathbf{u}_t$. Target-directed trajectories during proficient control are dependent on the desired movement duration. Hence in our work for decoding of target-directed movements when decoder parameters were known [12], [14], [8], we formed a finite-horizon cost function and decoded the trajectory by jointly estimating the movement kinematics and its horizon (i.e., duration) from neural activity. In CLDA, however, parameter estimates are poor initially and hence control is not proficient. Hence, we cannot estimate the intended horizon from neural activity. Moreover, targets may not be reached during the initial trials because of poor control. Thus instead of forming a finite-horizon cost function and decoding the horizon, we formulate an infinite-horizon cost function to develop adaptive OFC-PPF as a CLDA method. Recent motor control studies suggest that an infinite-horizon formulation could be an alternative to finite-horizon models [17].

We define the state as $\mathbf{x}_t = [\mathbf{d}_t, \mathbf{v}_t]'$ where the components represent position and velocity in the two dimensions and denote the target position by $\mathbf{d}^*$. We form the cost function

$$J = \sum_{t=1}^{\infty} \| \mathbf{d}_t - \mathbf{d}^* \|^2 + w_v \| \mathbf{v}_t \|^2 + w_r \| \mathbf{u}_t \|^2 \quad (2)$$

where the three terms in the sum enforce positional accuracy, stopping condition, and energetic efficiency, respectively, and the weights w_r and w_v are selected appropriately based on experimental movements. Given the linear Gaussian state-space model (1) and the quadratic cost function (2), the optimal $\mathbf{u}_t$ is given by the standard linear-quadratic-regulator (LQR) solution. Specifically, $\mathbf{u}_t$ at each time is a linear function of the controller's (brain's) estimate of the state at that time [18]. Given the assumption of noiseless visual feedback, the controller's (brain's) estimate of the state at each time is equal to the displayed state on the screen, $\mathbf{x}_{t|t}$, and hence OFC-PPF finds the intended control as

$$\mathbf{u}_t = -\mathbf{L}(\mathbf{x}_{t|t} - \mathbf{x}^*), \quad (3)$$

where $\mathbf{x}^* = [\mathbf{d}^*, \mathbf{0}]'$ is the target state for position and velocity, and $\mathbf{L}$ is the steady-state solution to the algebraic Riccati equation found recursively and offline [18].

B. Point process model of the spikes

Adaptive OFC-PPF enables subjects to send control commands and receive feedback with every spike event and at a much faster rate compared to KF. Moreover, it enables the BMI to adapt the parameters with every spike event in contrast to the time-scale of minutes in current methods. To achieve these goals, OFC-PPF incorporates a point process observation model of the spiking activity in closed loop.

We denote the neural observations of the ensemble of C neurons by $\mathbf{N}_1, \cdots, \mathbf{N}_t$ where $\mathbf{N}_t = (N_t^1, \cdots, N_t^C)$ is the binary spike events of the C neurons at time t. Assuming conditional independence among neurons given the state, the point process observation model is given by [19], [20]

$$p(\mathbf{N}_t|\mathbf{x}_t) = \prod_c (\lambda_c(t|\mathbf{x}_t)\Delta)^{N_t^c} e^{-\lambda_c(t|\mathbf{x}_t)\Delta} \quad (4)$$

where Δ is the time bin taken to be small enough to contain at most one spike (5 ms here) and $\lambda_c(t|\mathbf{x}_t)$ is the instantaneous firing rate of neuron c. We use a modified cosine tuning model of the motor cortex [21] to write $\lambda_c(t|\mathbf{x}_t)$ for each neuron as a log-linear function of velocity in the two dimensions

$$\lambda_c(t|\mathbf{x}_t) = \exp(\beta_c + \boldsymbol{\alpha}_c' \mathbf{v}_t) \quad (5)$$

where $\boldsymbol{\phi}_c = [\beta_c, \boldsymbol{\alpha}_c]$ are the decoder parameters for neuron c that need to be estimated in the CLDA (see Section II-C).

C. Spike-by-spike control and adaptation using OFC-PPF

We combine the OFC and PPF models to develop adaptive OFC-PPF. During the process of adaptation, we use the infinite-horizon OFC model to perform intention estimation as in (3). We develop one recursive Bayesian decoder for the kinematics and one decoder for each neuron's parameters. A recursive Bayesian decoder consists of a prior model on the states and an observation model relating the neural activity to these states. The observation model for both the kinematics and the parameters is the same and given by (4).

We construct the prior model of the parameters for each neuron using a random-walk state-space model given by

$$\boldsymbol{\phi}_c(t) = \boldsymbol{\phi}_c(t-1) + \mathbf{q}_t, \quad (6)$$

where $\mathbf{q}_t$ is white Gaussian noise with covariance matrix $\mathbf{Q}$.

To enable consistent parameter convergence despite poor initial performance, we devise a combined CLDA and assisted training paradigm by using the OFC state-space model (see (1) and (3)) as the prior in the kinematics decoder. This allows the kinematics decoder to explore the space even with poor initial parameters and moreover keeps the subject engaged during the adaptation process. Given the prior and the observation models for the kinematics and the parameters, we can find the recursions of the decoders.

For the kinematics decoder, let's denote the one step prediction mean by $\mathbf{x}_{t|t-1} = E(\mathbf{x}_t|\mathbf{N}_{1:t-1})$, the prediction covariance by $\mathbf{W}_{t|t-1}$, the minimum mean-squared error (MMSE) estimate that is displayed to the subject by $\mathbf{x}_{t|t}$, and its covariance by $\mathbf{W}_{t|t}$. Combining (3) and (1), the prediction step of OFC-PPF for the kinematics is found as

$$\mathbf{x}_{t|t-1} = (\mathbf{A} - \mathbf{BL})\mathbf{x}_{t-1|t-1} + \mathbf{BL}\mathbf{x}^* \quad (7)$$

$$\mathbf{W}_{t|t-1} = (\mathbf{A} - \mathbf{BL})\mathbf{W}_{t-1|t-1}(\mathbf{A} - \mathbf{BL})' + \mathbf{W} \quad (8)$$

The decoder update step as we have derived previously [8] is given by (see also [20] for the general case)

$$\mathbf{W}_{t|t}^{-1} = \mathbf{W}_{t|t-1}^{-1} + \sum_{c=1}^{C} \tilde{\boldsymbol{\alpha}}_c \tilde{\boldsymbol{\alpha}}_c{}' \lambda_c(t|\mathbf{x}_{t|t-1})\Delta \quad (9)$$

$$\mathbf{x}_{t|t} = \mathbf{x}_{t|t-1} + \mathbf{W}_{t|t} \sum_{c=1}^{C} \tilde{\boldsymbol{\alpha}}_c (N_t^c - \lambda_c(t|\mathbf{x}_{t|t-1})\Delta) \quad (10)$$

where $\tilde{\boldsymbol{\alpha}}_c = [\mathbf{0}, \boldsymbol{\alpha}_c]'$ (since the observation model assumes no position tuning). Hence adaptive OFC-PPF decodes the kinematics using (7)–(10).

For the parameters of each neuron, we similarly build an additional point process decoder with the prior model in (6). Hence the prediction step for each neuron's parameter decoder can be obtained as in (7), (8) but by setting $\mathbf{B} = \mathbf{0}$, $\mathbf{A} = \mathbf{I}$, $\mathbf{W} = \mathbf{Q}$, and the update step can be obtained as in (9), (10) but by reversing the role of $\boldsymbol{\phi}_c$ and $\mathbf{x}_t$ (see (5)) and setting $C = 1$ (i.e., estimating the parameters of each neuron only based on its own neural activity). Note that we do not perform joint estimation of parameters and kinematics [20]. Instead, given the poor initial parameters, we rely on the OFC model to provide the intended kinematics to each neuron's parameter decoder using (3) and (7) (Fig. 1).

Adaptive OFC-PPF stops assisted training once non-assisted performance exceeds a desired level (Fig. 2). This condition is evaluated by testing the subject's non-assisted performance in short intervals using a random-walk PPF for kinematics in which $\mathbf{B}$ is set to $\mathbf{0}$. Once assistance stops, the subject keeps using this random-walk PPF as the decoder.

D. Closed-loop BMI experiments

Variability in recordings and task designs make across-study comparisons difficult, so here we compare performance across different decoders within the same subject with a self-paced center-out movement task consisting of 8 targets (detailed in [6]). We recorded from 17–20 multiunits in the primary motor cortex of one rhesus monkey. The subject initiated a trial by moving the cursor to the center target. Once the cursor entered the center, one of 8 peripheral targets appeared followed by a go cue (that changed the center target's color). After the go cue, the subject moved the cursor towards the displayed target. To be successful, subjects had to reach the correct target and hold it for 250 ms. If the cursor left the target before the 250 ms, the trial was considered unsuccessful and the subject was not given the chance to correct this hold error unlike some prior tasks (e.g. [5]). Subjects had to return the cursor to the center and hold there to initiate the next trial. We use success rate, i.e., the number of successful trials per minute, as the performance measure and also calculate the movement error and the reach time.

III. Results

Here we present the performance of adaptive OFC-PPF and SmoothBatch-KF on the self-paced center-out task.

A. Spike-by-spike adaptation vs. batch-based adaptation

We compared adaptive OFC-PPF that adapts the parameters with every spike event (every 5ms) with a version of OFC-PPF that did so every 90 sec using SmoothBatch adaptation [6]. In the latter case, we still used the OFC to infer velocity intention, however, we collected 90 sec batches of neural activity and the corresponding intentions, and refit the decoder parameters within each batch using generalized-linear-model (GLM) maximum-likelihood techniques. Fig. 2 shows the monkey's performance in two consecutive days,

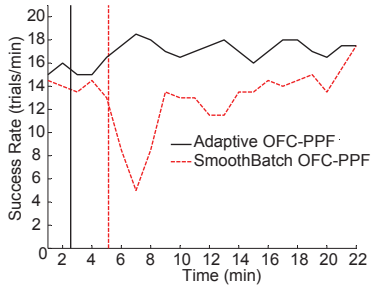

Fig. 2. Performance convergence over time for adaptive OFC-PPF and SmoothBatch OFC-PPF run on two consecutive days and starting from the same initial parameters. Vertical lines show the time point where assistance stops as the subject's non-assisted success rate exceeds the desired level of 5 after that point. Success rate is calculated in sliding 2 min windows.

one with adaptive OFC-PPF and one with SmoothBatch OFC-PPF. Both decoders started from the same initial parameters. Adaptive OFC-PPF resulted in faster convergence of performance to its steady-state value. This result held on average across days. Looking at 6 days of convergence with SmoothBatch OFC-PPF and 6 days with adaptive OFC-PPF, performance converged in the former case within 18.1 ± 11.1 minutes and in the latter case within 7.4 ± 2.9 minutes.

B. Steady-state: OFC-PPF vs. SmoothBatch-KF

We compared adaptive OFC-PPF with SmoothBatch-KF in closed-loop experiments across days. OFC-PPF outperformed SmoothBatch-KF, improving success rate by 32% in this monkey. Movement error and reach time were also improved by 14% and 18%, respectively. We also examined the contribution of the OFC and the PPF components to the observed improvement. To dissociate the two effects, we first implemented a SmoothBatch-KF that, instead of using CursorGoal, used the OFC intention estimation. We compared the steady-state performance of this OFC SmoothBatch-KF with OFC-PPF and found that success rate in the latter was 30% higher than the former, demonstrating that PPF's faster control and feedback rate was necessary for performance improvement. We then implemented a PPF that used CursorGoal intention estimation during adaptation and compared that to OFC-PPF. We found that OFC-PPF improved the success rate by 27% compared to this CursorGoal-PPF, indicating that OFC intention estimation was essential for performance improvement and better approximated the user's strategy.

IV. Conclusions

We have developed a new closed-loop BMI architecture, adaptive OFC-PPF. Adaptive OFC-PPF allows subjects to control the BMI with every spike event, enables the BMI to adapt the parameters with every spike event, and uses an infinite-horizon OFC model of the brain to infer user's intention during adaptation and to design a new assisted training technique. Preliminary data from one monkey suggests that spike-by-spike adaptation results in faster performance convergence compared with current techniques, that OFC-PPF outperforms SmoothBatch-KF, and that both the OFC and the PPF models are necessary for this performance improvement.

References

[1] D. M. Taylor, S. I. H. Tillery, and A. B. Schwartz, "Direct cortical control of 3D neuroprosthetic devices," *Science*, vol. 296, June 2002.

[2] J. M. Carmena, M. A. Lebedev, R. E. Crist, J. E. O'Doherty, D. M. Santucci, D. F. Dimitrov, P. G. Patil, C. S. Henriquez, and M. A. L. Nicolelis, "Learning to control a brain-machine interface for reaching and grasping by primates," *PLoS Biol.*, vol. 1, no. 2, 2003.

[3] M. Velliste, S. Perel, M. C. Spalding, A. S. Whitford, and A. B. Schwartz, "Cortical control of a prosthetic arm for self-feeding," *Nature*, vol. 453, pp. 1098–1101, June 2008.

[4] L. R. Hochberg, D. Bacher, B. Jarosiewicz, N. Y. Masse, J. D. Simeral, J. Vogel, S. Haddadin, J. Liu, S. S. Cash, P. van der Smagt, and J. P. Donoghue, "Reach and grasp by people with tetraplegia using a neurally controlled robotic arm," *Nature*, vol. 485, pp. 372–377, May 2012.

[5] V. Gilja, P. Nuyujukian, C. A. Chestek, J. P. Cunningham, B. M. Yu, J. M. Fan, M. M. Churchland, M. T. Kaufman, J. C. Kao, S. I. Ryu, and K. V. Shenoy, "A high-performance neural prosthesis enabled by control algorithm design," *Nat. Neurosci.*, vol. 15, pp. 1752–1757, Dec. 2012.

[6] A. L. Orsborn, S. Dangi, H. G. Moorman, and J. M. Carmena, "Closed-loop decoder adaptation on intermediate time-scales facilitates rapid bmi performance improvements independent of decoder initialization conditions," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 20, no. 4, pp. 468–477, July 2012.

[7] J. L. Collinger, B. Wodlinger, J. E. Downey, W. Wang, E. C. Tyler-Kabara, D. J. Weber, A. J. McMorland, M. Velliste, M. L. Boninger, and A. B. Schwartz, "High-performance neuroprosthetic control by an individual with tetraplegia," *The Lancet*, vol. 381, no. 9866, pp. 557–564, Feb. 2013.

[8] M. M. Shanechi, Z. M. Williams, G. W. Wornell, R. Hu, M. Powers, and E. N. Brown, "A real-time brain-machine interface combining motor target and trajectory intent using an optimal feedback control design," *PLOS ONE*, vol. 8, no. 4, p. e59049, Apr. 2013.

[9] M. M. Shanechi, R. C. Hu, M. Powers, G. W. Wornell, E. N. Brown, and Z. M. Williams, "Neural population partitioning and a concurrent brain-machine interface for sequential motor function," *Nat. Neurosci.*, vol. 15, no. 12, pp. 1715–1722, Dec. 2012.

[10] K. Ganguly and J. M. Carmena, "Emergence of a stable cortical map for neuroprosthetic control," *PLoS Biol.*, vol. 7, no. 7, July 2009.

[11] B. Mahmoudi and J. C. Sanchez, "A symbiotic brain-machine interface through value-based decision making," *PLOS ONE*, vol. 6, no. 3, p. e14760, Apr. 2011.

[12] M. M. Shanechi, G. W. Wornell, Z. M. Williams, and E. N. Brown, "A parallel point-process filter for estimation of goal-directed movements from neural signals." in *Proc. IEEE conference on acoustics, speech, and signal processing (ICASSP)*, Dallas, TX, Mar. 2010.

[13] M. M. Shanechi, Z. M. Williams, G. W. Wornell, and E. N. Brown, "A brain-machine interface combining target and trajectory information using optimal feedback control," in *Computational and Systems Neuroscience (COSYNE) Meeting*, Salt Lake City, USA, Feb. 2011.

[14] M. M. Shanechi, G. W. Wornell, Z. M. Williams, and E. N. Brown, "Feedback-controlled parallel point process filter for estimation of goal-directed movements from neural signals," *IEEE Trans. Neural Syst. Rehabil. Eng.*, doi: 10.1109/TNSRE.2012.2221743, Oct. 2012.

[15] R. Shadmehr and J. W. Krakauer, "A computational neuroanatomy for motor control," *Exp. Brain Res.*, vol. 185, pp. 359–381, 2008.

[16] M. D. Golub, B. M. Yu, and S. M. Chase, "Internal models engaged by brain-computer interface control," in *Proc. IEEE EMBS*, San Diego, USA, Aug. 2012, pp. 1327–1330.

[17] D. Huh, E. Todorov, and T. J. Sejnowski, "Infinite horizon optimal control framework for goal directed movements," in *Society for Neuroscience (SFN) Meeting*, San Diego, USA, Nov. 2010.

[18] D. Bertsekas, *Dynamic Programming and Optimal Control*. Athena Scientific, 2005.

[19] E. N. Brown, L. M. Frank, D. Tang, M. C. Quirk, and M. A. Wilson, "A statistical paradigm for neural spike train decoding applied to position prediction from ensemble firing patterns of rat hippocampal place cells," *J. Neurosci.*, vol. 18, no. 18, pp. 7411–7425, Sept. 1998.

[20] U. T. Eden, L. M. Frank, R. Barbieri, V. Solo, and E. N. Brown, "Dynamic analysis of neural encoding by point process adaptive filtering," *Neural Comput.*, vol. 16, pp. 971–998, 2004.

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