Neural Shaping with Joint Optimization of Controller and Plant under Restricted Dynamics

Bryan D. He
Department of Computer Science
California Institute of Technology
Pasadena, CA 91126, USA
Email: bryanhe@nsplab.org

Lakshminarayan Srinivasan
Neural Signal Processing Laboratory
University of California, Los Angeles
Los Angeles, CA 90095, USA
Email: ls2@nsplab.org

***Abstract*—The prototypical brain-computer interface (BCI) algorithm translates brain activity into changes in the states of a computer program, for typing or cursor movement. Most approaches use *neural decoding* which learns how the user has encoded their intent in their noisy neural signals. Recent adaptive decoders for cursor movement improved BCI performance by modeling the user as a feedback controller; when this model accounts for adaptive control, the neural decoder is termed co-adaptive. This recent collection of control-inspired neural decoding strategies disregards a major antecedent conceptual realization, whereby the user could be induced to adopt an encoding strategy (control policy) such that the encoder-decoder pair (or equivalently, controller-plant pair) is optimal under a desired cost function.**

We call this alternate conceptual approach *neural shaping*, in contradistinction to neural decoding. Previous work illuminated the general form of optimal controller-plant pair under a cost representing information gain. For BCI applications requiring the user to issue discrete-valued commands, the information-gain-optimal pair, based on the posterior matching scheme, can be user-friendly.

In this paper, we discuss the application of neural shaping to cursor control with continuous-valued states based on continuous-valued user commands. We examine the problem of jointly optimizing controller and plant under quadratic expected cost and restricted linear plant dynamics. This simplification reduces joint controller-plant selection to a static optimization problem, similar to approaches in structural engineering and other areas. This perspective suggests that recent BCI approaches that alternate between adaptive neural decoders and static neural decoders could be local Pareto-optimal, representing a sub-optimal iterative-type solution to the optimal joint controller-plant problem.

I. Introduction

Brain-computer interfaces (BCI) allow humans to control states of a computer by modulating their brain activity. For example, a recent demonstration allowed a paralyzed patient to control detailed movements of an anthropomorphic robotic arm through neural activity recorded from surgically-implanted electrodes in motor cortex [1]. BCI algorithms govern how computer states change with neural activity and what feedback is sent to the user. Although BCI can be used for typing, drawing, and other activities based on discrete-valued user commands [2], we focus on user control of movement based on continuous-valued user commands.

Neural decoding is the dominant conceptual approach to designing BCI algorithms. In this perspective, neural activity encodes the user's intent, corrupted by noise. The BCI algorithm is designed as a decoder that learns this encoding in order to estimate user intent from neural signals. This perspective encompasses most existing BCI methods, including the population vector algorithm [3], Kalman filter [4], adaptive point process filter [5]–[7], ReFIT [8], Joint RSE [9], and co-adaptive estimation [10].

Some empirical work suggests that users can improve closed-loop performance even during periods where the neural decoder is fixed [11], [12]. This raises the possibility that neural activity could be purposefully shaped to implement encoding schemes that optimally match the decoder. Neural decoding entirely disregards this opportunity in its design. For example, adaptive neural decoders attempt to match the decoder to the user's current encoding scheme. This leads to rapid early improvements, but possibly sacrifices asymptotic performance.

Separate work antecedant to the newer adaptive decoders proposed a comprehensive framework for the BCI algorithm as the joint selection of encoder and decoder [2], [13], [14]. In one demonstration of this approach, the posterior matching scheme [15] was employed in the context of discrete-valued (binary) user commands to specify a two-dimensional path of movement [2]. In this case, adopting the encoding scheme did not likely require significant user training. In other work, an anthropomorphic robotic hand was designed by joint optimization of controller (a simulated user) and plant (the robotic hand) [16].

Neural decoding methods called 'turntaking' alternate between user learning and decoder adaptation [17], [18], but this iterative approach likely settles on a Pareto-optimal controller-plant which is not globally optimal [19]. For example, the closed-loop decoder adaptation (CLDA) method [17], [18] converges to decoder parameters that can induce pathological behavior in cursor movement, like tremor and curling force fields [20]. Importantly, the authors of that work corrected this pathological behavior by adjusting the related CLDA parameters based on analysis of closed-loop dynamics assuming some control model of the user [20]. In relation to this approach, neural shaping would have formally optimized these parameters based on cost of closed-loop dynamics, assuming optimal control by the user.

To help develop neural shaping for BCI cursor control, we examine the problem of jointly optimizing controller and plant under multi-dimensional continuous-valued state space and quadratic cost. Prior work solved the one-dimensional case [13], building on related results [21], [22]. To the best of our knowledge, the analytical solution for arbitrary dimensions and quadratic cost is not known and may be difficult [19]. To simplify the multi-dimensional case, we parametrize the plant to restrict the types of linear-Gaussian dynamics. This simplified problem may be sufficient to break the local Pareto-optimality of turntaking neural decoders, and possibly unlock substantial performance gains for patients that will rely on this technology.

II. Problem Statement

A discrete-time stochastic control system is defined by a plant, controller, and cost function. The plant is a difference equation in discrete time steps k ranging from 0 to N, describing the evolution of a state vector x_k in relation to control inputs u_k and noise n_k:

$$x_{k+1} = g(x_k, u_k, n_k) \tag{1}$$

A controller receives information about the plant state x_k as well as the objective state x^* and determines a control input, according to a control policy:

$$u_k = f(x_k, x^*) \tag{2}$$

The quality of a control policy can be evaluated with a cost function which penalizes both states and control inputs with relation to a known objective state x^*, and a distribution on the initial state x_0 and noise n_k.

$$C(f, g, x^*) = \underset{\substack{x_0, n_k \\ k=0,1,\ldots,N-1}}{E} \{C(u_0, \ldots, u_{N-1}, x_0, \ldots, x_N, x^*)\} \tag{3}$$

In contrast, the plant does not have knowledge of the true objective state x^*. Instead, it knows only a distribution on the objective state. As such, the plant can be evaluated with a cost function averaged over the distribution on the initial state x_0, objective state x^*, and noise n_k,

$$C(f, g) = \underset{\substack{x_0, x^*, n_k \\ k=0,1,\ldots,N-1}}{E} \{C(u_0, \ldots, u_{N-1}, x_0, \ldots, x_N, x^*)\} \tag{4}$$

Because it determines the length of time over which cost is accrued, N is also called the horizon, which can be finite or infinite. For infinite horizon, the cost functions (3),(4) are evaluated in the limit as $N \to \infty$ after multiplying by $1/N$ to represent the average cost per time step.

A. Optimal Control Problem

Given plant dynamics g and cost function $C(f, g)$, the optimal control problem seeks the control policy f^* that minimizes cost:

$$f^* = \underset{f}{\operatorname{argmin}}\, C(f, g) \tag{5}$$

B. Inverse Optimal Control Problem

A related problem is the inverse optimal control problem [23]. Given plant dynamics g and control policy f^*, this problem seeks to find all cost functions $C^*(f, g)$ under which the controller-plant pair is optimum.

$$f^* = \underset{f}{\operatorname{argmin}}\, C^*(f, g) \tag{6}$$

This problem is of particular interest to mathematical biologists and computational neuroscientists that seek to uncover the costs to which various natural systems are optimized [24].

C. Optimal Controller-Plant Problem

Given cost function $C(f, g)$, distribution on initial state x_0, and distribution on objective state x^*, the optimal controller-plant problem seeks control policy f^* and plant dynamics g^* that are optimal under the cost function,

$$(f^*, g^*) = \underset{f,g}{\operatorname{argmin}}\, C(f, g) \tag{7}$$

The problem can be simplified (as with our case below) by parametrizing the plant dynamics g. Following existing nomenclature on combined plant/controller optimization [19], this the simultaneous formulation of the neural shaping problem. The nested (bi-level) formulation of this problem [19] is,

$$f^*(g) = \underset{f}{\operatorname{argmin}}\, C(f, g) \tag{8}$$

$$g^* = \underset{g}{\operatorname{argmin}}\, C(f = f^*(g), g) \tag{9}$$

The nested formulation determines the optimal control policy f^* for a generically parameterized plant specification g, denoted $f^*(g)$ to emphasize that the optimal control policy is determined by the plant. The plant is optimized with knowledge that the corresponding optimal control policy is in effect. In the special case discussed below, we use this approach to determine an analytical solution to f^* and g^*. In practice, an analytical solution may not be possible, so f^* is determined numerically for any given choice of g. The approach runs in nested loops, where the inner loop generates the optimal controller for the plant generated by the outer loop [19].

The simultaneous and nested formulations provide global optimal plant/controller pairs [19]. This is not guaranteed for sequential and iterative formulations [19]. In the sequential formulation, the plant is optimized, followed by the controller. The iterative formulation alternates between optimizing plant and controller.

III. Example BCI-related Controller-Plant Solution

We now solve a specific case of the optimal plant problem, where the plant dynamics (1) are linear-Gaussian,

$$x_{k+1} = Ax_k + \beta B(u_k + n_k) \tag{10}$$

$$= Ax_k + \beta Bu_k + w_k \tag{11}$$

with the plant dynamics defined as

$$A = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} \tag{12}$$

$$B = \begin{bmatrix} \Delta \\ 1 \end{bmatrix} \tag{13}$$

$$n_k \sim \left(\begin{bmatrix} 0 \end{bmatrix}, \Sigma_n = \begin{bmatrix} \sigma \end{bmatrix} \right) \tag{14}$$

$$w_k \sim \left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \Sigma_w = (\beta B)\Sigma_n(\beta B)' = \sigma\beta^2 \begin{bmatrix} \Delta^2 & \Delta \\ \Delta & 1 \end{bmatrix} \right) \tag{15}$$

This system represents a computer cursor restricted to move along a line, with discrete time steps of Δ. The state x_k is a 2×1 vector consisting of the cursor position and velocity. The initial state is x_0, and the objective state is x^*. The control u_k is a 1×1 vector giving the intended velocity. The zero-mean gaussian noise with variance σ is directly added to the control input. The plant is parameterized by a single control scaling parameter β. The cost (3) considered by the controller is quadratic and infinite-horizon with known objective state x^*,

$$\lim_{N\to\infty} \frac{1}{N} \underset{\substack{w_k, x_0 \\ k=0,1,\ldots,N-1}}{E} \Bigg((x_N - x^*)'Q_N(x_N - x^*) + \sum_{k=0}^{N-1} \left((x_k - x^*)'Q_k(x_k - x^*) + u_k'R_k u_k\right) \Bigg) \tag{16}$$

The costs matrices are defined as

$$Q_k = \begin{bmatrix} q_1 & 0 \\ 0 & q_2 \end{bmatrix} \tag{17}$$

$$R_k = \begin{bmatrix} r \end{bmatrix} \tag{18}$$

where q_1 and q_2 are parameters specifying the costs on kinematics and r is the parameter specifying the cost on control. In particular, q_1 penalizes cursor distance from the origin and q_2 penalizes cursor velocity. Notation is summarized in Table I.

We now find the optimal controller for any given choice of the control scaling parameter β. A known result is that the optimal controller for linear-Gaussian plant dynamics with infinite-horizon quadratic cost is linear and time-invariant [25], written in terms of the solution G to the algebraic Ricatti equation,

$$G = A'(G - GB(B'GB + R)^{-1}B'G)A + Q \tag{19}$$

In this case, the solution is analytic,

$$G = \begin{bmatrix} c & 0 \\ 0 & q_2 \end{bmatrix} \tag{20}$$

where

$$c = \frac{\beta\Delta q_1 + \sqrt{q_1}\sqrt{\beta^2(\Delta^2 q_1 + 4q_2) + 4r}}{2\beta\Delta} \tag{21}$$

The optimal controller [25] is a time-invariant control policy

$$f(x_k, x^*) = L(x_k - x^*) \tag{22}$$

Variables

Name	Type	Description
k	Scalar	Discrete time step index
N	Scalar	Number of time steps (cost horizon)
Δ	Scalar	Time step of the system
β	Scalar	Control scaling parameter
σ	Scalar	Variance of the noise added to the control input
x_k	2×1	State of the closed-loop system (position and velocity)
x^*	2×1	Objective state
u_k	1×1	Control input (intended velocity)
n_k	1×1	Gaussian noise added to control
w_k	2×1	Gaussian noise added to state
A_k	2×2	Matrices specifying the plant dynamics in response to the previous state
B_k	2×1	Matrices specifying the plant dynamics in response to control input
Q_k	2×2	Matrices specifying kinematic cost
R_k	1×1	Matrices specifying control cost
L	1×2	Matrix specifying the optimal controller
J	Scalar	Optimal cost

Functions

Name	Description
$f(x_k, x^*)$	Control policy mapping current state and objective state to a control input
$g(x_k, u_k, n_k)$	Plant dynamics mapping current state, control input, and noise to a new state
$C(f, g, x^*)$	Cost function from the perspective of the controller mapping the control policy, plant dynamics, and objective state to the expected cost
$C(f, g)$	Cost function from the perspective of the controller mapping the control policy, plant dynamics, and objective state to the expected cost

TABLE I
DEFINITION OF NOTATION

where

$$L = -(B'GB + R)^{-1}B'GA \tag{23}$$

$$= \begin{bmatrix} -\frac{\beta c\Delta}{\beta^2(c\Delta^2 + q_2) + r} & 0 \end{bmatrix} \tag{24}$$

The optimal cost [25] as determined by the plant , is

$$J = \underset{x_0, x^*}{E}\{(x_0 - x^*)'G(x_0 - x^*)\} + \sum_{k=0}^{N-1} \underset{w_k}{E}\{w_k'Gw_k\} \tag{25}$$

$$= \underset{x_0, x^*}{E}\{(x_0 - x^*)'G(x_0 - x^*)\} + \sum_{k=0}^{N-1} \text{tr}(\Sigma_w G) \tag{26}$$

$$= \underset{x_0, x^*}{E}\{(x_0 - x^*)'G(x_0 - x^*)\} + \sum_{k=0}^{N-1} \beta^2\sigma(\Delta^2 c + \beta) \tag{27}$$

$$= \underset{x_0, x^*}{E}\{(x_0 - x^*)'G(x_0 - x^*)\} + N\beta^2\sigma(\Delta^2 c + \beta) \tag{28}$$

Denote the initial state as $x_0 = \begin{bmatrix} p_0 \\ v_0 \end{bmatrix}$, and the objective state as $x^* = \begin{bmatrix} p^* \\ v^* \end{bmatrix}$. The plant only knows the distribution of the initial state and objective state. As a result, it must take an expected value over the possible initial states and objective states. The optimal cost from the perspective of the plant for an N-step horizon is

$$J = \underset{x_0, x^*}{E} \left\{ (p_0 - p^*)^2 c + (v_0 - v^*)^2 q_2 \right\} + N\beta^2\sigma(\Delta^2 c + \beta) \tag{29}$$

Writing

$$p^2 = \underset{x_0, x^*}{E} \left\{ (p_0 - p^*)^2 \right\} \tag{30}$$

$$v^2 = \underset{x_0, x^*}{E} \left\{ (v_0 - v^*)^2 \right\} \tag{31}$$

the cost (29) simplifies to

$$J = p^2 c + v^2 q_2 + N\beta^2\sigma(\Delta^2 c + \beta) \tag{32}$$

where c is given by (21).

Note that we have purposefully chosen the cost to be finite horizon from the plant's perspective, despite having optimized the controller based on an infinite-horizon cost. This is to avoid a degenerate case, as will be discussed below with relation to Figure 1.

The optimal control scaling parameter β^* is defined as

$$\beta^* = \underset{\beta}{\operatorname{argmin}}\, J \tag{33}$$

To find this value, the derivative of J in terms of β is set to zero.

$$\frac{\partial}{\partial \beta} J = 0 \tag{34}$$

$$\begin{aligned} &\frac{1}{\beta^2\Delta\sqrt{\beta^2(\Delta^2 q_1 + 4q_2) + 4r}} \Big[-2p^2\sqrt{q_1}r + \\ &\beta^2\Delta N(\beta^2\Delta\sqrt{q_1}(\Delta^2 q_1 + 4q_2) + 2\Delta\sqrt{q_1}r + \\ &\beta(\Delta^2 q_1 + 2q_2)\sqrt{\beta^2(\Delta^2 q_1 + 4q_2) + 4r})\sigma \Big] = 0 \end{aligned} \tag{35}$$

The optimal control scaling parameter β^* is analytically solved in Mathematica. The resulting solution is voluminous and omitted due to space constraints. By varying β in the optimal cost function of (32), we can develop intuition on how this plant control scaling factor influences closed-loop performance.

Figure 1 shows that control scaling parameters that are either too small or too large both erode closed-loop performance. The optimal scaling parameter elicits large amplitude control signal from the user, enhancing signal-to-noise, without demanding so much control signal that cost on control becomes unmanageable or kinematics deteriorates. The pattern of an optimal scaling parameter persists across different cost horizons. Longer cost horizons are more sensitive to control scaling parameter, perhaps because deteriorated performance is accumulated over longer periods of time.

We also see in Figure 1 that the optimum β decreases as N grows. In the infinite horizon limit as $N \to \infty$, the optimum β for average cost per step goes to zero. In this degenerate scenario, the plant is incentivized to minimize the influence of control noise on the state once the goal state has been achieved. To do this, the plant must decrease the value of β. This in turn causes the plant state to more slowly arrive at x^*, but since the horizon is infinite, slow convergence to x^* does not ultimately affect cost per step.

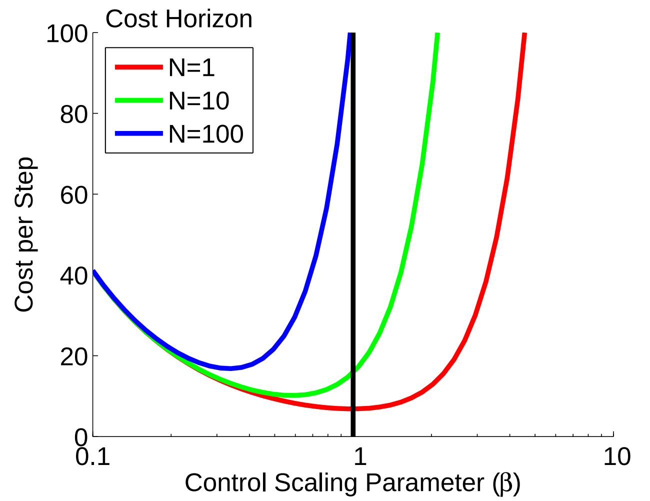

Fig. 1. Cost per step versus control scaling parameter for different cost horizons. The parameters for the cost function are $q_1 = 0.18$, $q_2 = 0.1$, and $r = 0.1$. The time step of the system is $\Delta = 0.033$. The noise added to the control has a variance of $\sigma = 1$.

Figure 2 shows that smaller control scaling parameters are needed as noise on control input grows. The smaller control scaling parameter elicits larger control signals from the user, improving control signal-to-noise in the face of growing noise on control. This improved signal-to-noise comes at the expense of increased cost on control. Interestingly, when noise on control is sufficiently small, the optimal scaling parameter can afford to be greater than 1, meaning that control signals are magnified, allowing the user to conserve cost on control.

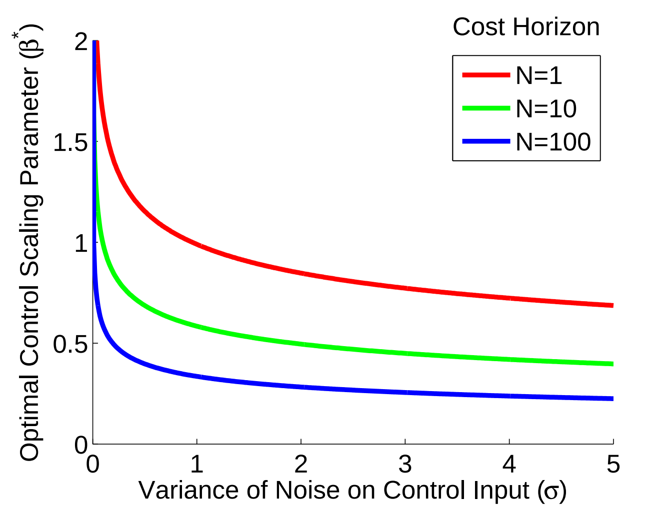

Fig. 2. Optimal control scaling parameter versus increment noise for different cost horizons. The parameters for the cost function are $q_1 = 0.18$, $q_2 = 0.1$, and $r = 0.1$. The time step of the system is $\Delta = 0.033$.

Figure 3 displays the complete phase diagram relating optimal scaling parameter to noise on control and cost on control. The phase diagram recapitulates that increasing control cost promotes an optimal plant that scales up the control signal in an attempt to conserve the user's control cost. The noise on control input is a countervailing force, promoting an optimal plant that scales down control input, inducing the user to generate higher signal-to-noise control inputs, thereby incurring increased cost on control.

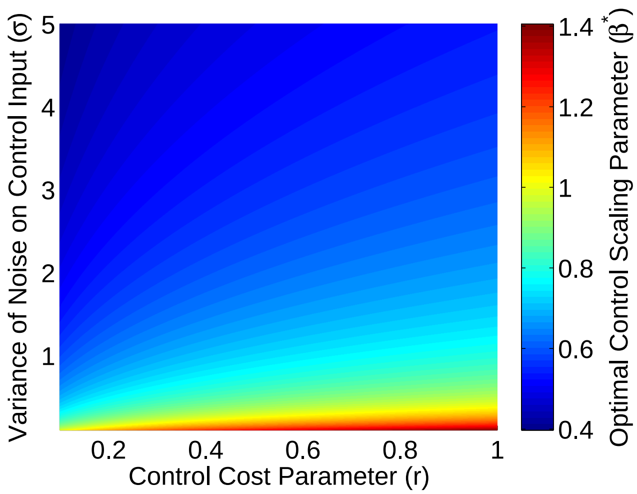

Fig. 3. Phase diagram of the optimal control scaling parameter as it depends on control cost and noise on control, using a 10-step cost horizon. When cost on control is low and the noise level is high, the optimal control scaling parameter is high to scale down the noise and encourage larger control inputs. When the control cost is high and the noise level is low, it is hard to elicit larger control inputs and the noise is less disruptive.

IV. Conclusion

The relevance of brain-computer interfaces (BCI) to medical and consumer applications hinges on the extent to which BCI algorithms will induce reliable, high-performance closed-loop behavior in the interaction between users and their computers. Neural shaping aims to acheive substantial gains in performance by recasting the design of BCI algorithms from neural decoding to methods that induce optimal controller-plant policies. From the neuroscientific standpoint, this approach can be viewed as directing the process of operant conditioning (reward-based learning, [26]) towards neural signal characteristics that favor high quality control. Building on initial work in this area [2], [13], [14], [16], with connections to an expansive related multi-disciplinary literature, the approach to neural shaping outlined in this paper suggests one path in the near term for experiments to demonstrate and break local Pareto-optimal solutions [19] resulting from existing BCI algorithms based on neural decoders.

Acknowledgment

This work was suppoted by the 2013 Caltech Summer Undergraduate Research Fellowship (to B.D.H.) and the American Heart Association Scientist Development Grant (11SDG7550015 to L.S.).

References

[1] 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, 2013.

[2] C. Omar, A. Akce, M. Johnson, T. Bretl, R. Ma, E. Maclin, M. McCormick, and T. P. Coleman, "A feedback information-theoretic approach to the design of brain–computer interfaces," *Intl. Journal of Human–Computer Interaction*, vol. 27, no. 1, pp. 5–23, 2010.

[3] A. P. Georgopoulos, A. B. Schwartz, and R. E. Kettner, "Neuronal population coding of movement direction," *Science*, vol. 233, no. 4771, pp. 1416–1419, 1986.

[4] 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 computation*, vol. 18, no. 1, pp. 80–118, 2006.

[5] 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 Computation*, vol. 16, no. 5, pp. 971–998, 2004.

[6] L. Srinivasan, U. T. Eden, S. K. Mitter, and E. N. Brown, "General-purpose filter design for neural prosthetic devices," *Journal of Neurophysiology*, vol. 98, no. 4, pp. 2456–2475, 2007.

[7] M. M. Shanechi, Z. M. Williams, G. W. Wornell, R. C. 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, 2013.

[8] V. Gilja, P. Nuyujukian, C. A. Chestek, J. P. Cunningham, M. Y. Byron, J. M. Fan, M. M. Churchland, M. T. Kaufman, J. C. Kao, S. I. Ryu, *et al.*, "A high-performance neural prosthesis enabled by control algorithm design," *Nature neuroscience*, vol. 15, no. 12, pp. 1752–1757, 2012.

[9] K. C. Kowalski, B. D. He, and L. Srinivasan, "Dynamic analysis of naive adaptive brain-machine interfaces," *Neural Computation*, vol. 25, no. 9, pp. 2373–2420, 2013.

[10] J. S. Merel, R. Fox, T. Jebara, and L. Paninski, "A multi-agent control framework for co-adaptation in brain-computer interfaces," in *Advances in Neural Information Processing Systems*, pp. 2841–2849, 2013.

[11] K. Ganguly and J. M. Carmena, "Emergence of a stable cortical map for neuroprosthetic control," *PLoS biology*, vol. 7, no. 7, p. e1000153, 2009.

[12] W. Wang, J. L. Collinger, A. D. Degenhart, E. C. Tyler-Kabara, A. B. Schwartz, D. W. Moran, D. J. Weber, B. Wodlinger, R. K. Vinjamuri, R. C. Ashmore, *et al.*, "An electrocorticographic brain interface in an individual with tetraplegia," *PloS one*, vol. 8, no. 2, p. e55344, 2013.

[13] S. Gorantla and T. Coleman, "Information-theoretic viewpoints on optimal causal coding-decoding problems," *International Journal on Human-Computer Interaction*, vol. 27, no. 1, pp. 5–23, 2011.

[14] S. Kim and T. P. Coleman, "Team decision theory and brain-machine interfaces," in *Neural Engineering (NER), 2011 5th International IEEE/EMBS Conference on*, pp. 605–608, IEEE, 2011.

[15] O. Shayevitz and M. Feder, "Optimal feedback communication via posterior matching," *Information Theory, IEEE Transactions on*, vol. 57, no. 3, pp. 1186–1222, 2011.

[16] L. Zollo, S. Roccella, E. Guglielmelli, M. C. Carrozza, and P. Dario, "Biomechatronic design and control of an anthropomorphic artificial hand for prosthetic and robotic applications," *Mechatronics, IEEE/ASME Transactions on*, vol. 12, no. 4, pp. 418–429, 2007.

[17] A. Orsborn, S. Dangi, H. Moorman, and J. Carmena, "Closed-loop decoder adaptation on intermediate time-scales facilitates rapid bmi performance improvements independent of decoder initialization conditions," *Neural Systems and Rehabilitation Engineering, IEEE Transactions on*, vol. 20, pp. 468–477, July 2012.

[18] S. Dangi, A. L. Orsborn, H. G. Moorman, and J. M. Carmena, "Design and analysis of closed-loop decoder adaptation algorithms for brain-machine interfaces," *Neural computation*, vol. 25, no. 7, pp. 1693–1731, 2013.

[19] H. K. Fathy, J. A. Reyer, P. Y. Papalambros, and A. Ulsov, "On the coupling between the plant and controller optimization problems," in *American Control Conference, 2001. Proceedings of the 2001*, vol. 3, pp. 1864–1869, IEEE, 2001.

[20] S. Gowda, A. L. Orsborn, and J. M. Carmena, "Parameter estimation for maximizing controllability of linear brain-machine interfaces," in *Engineering in Medicine and Biology Society (EMBC), 2012 Annual International Conference of the IEEE*, pp. 1314–1317, IEEE, 2012.
[21] R. Bansal and T. Basar, "Simultaneous design of communication and control strategies for stochastic systems with feedback," in *Analysis and optimization of systems*, vol. 46, pp. 603–616, Springer, 1988.
[22] Tatikonda, "Control under communication constraints," in *Ph.D. dissertation, Massachusetts Institute of Technology*, 2000.
[23] R. E. Kalman, "When is a linear control system optimal?," *Journal of Basic Engineering*, vol. 86, no. 1, pp. 51–60, 1964.
[24] E. Todorov, "Optimality principles in sensorimotor control," *Nature neuroscience*, vol. 7, no. 9, pp. 907–915, 2004.
[25] D. P. Bertsekas, D. P. Bertsekas, D. P. Bertsekas, and D. P. Bertsekas, *Dynamic programming and optimal control*, vol. 1. Athena Scientific Belmont, 1995.
[26] E. E. Fetz and D. V. Finocchio, "Operant conditioning of specific patterns of neural and muscular activity," *Science*, vol. 174, no. 4007, pp. 431–435, 1971.