A control-theoretic approach to brain-computer interface design*

Yin Zhang and Steven M. Chase

***Abstract*— Brain-computer interfaces (BCIs) have the potential to restore motor abilities to paralyzed individuals. These systems act by reading motor intent signals directly from the brain and using them to control, for example, the movement of a cursor on a computer screen or the motion of a robotic limb. To construct a BCI, a mapping must be specified that dictates how neural activity will actuate the device. How should these mappings be constructed to maximize user performance? Most approaches have focused on this problem from an estimation standpoint, i.e., mappings are designed to implement the best estimate of motor intent possible, under various sets of assumptions about how the recorded neural signals represent motor intent. Here we forward an alternate approach to the BCI design problem, using ideas from optimal control theory. We first argue that the brain can be considered as an optimal controller. We then introduce a mathematical definition of BCI usability, and formulate the BCI design problem as a constrained optimization problem that maximizes this usability.**

I. Introduction

Over the past decade, brain-computer interfaces (BCIs) have received a great deal of attention due to their promising clinical applications. These devices take recorded neural activity and use it to directly actuate some device, such as a robotic limb [1, 2] or a cursor on a computer screen [3-5]. One promising class of these devices is the intracortical BCI, which uses as its signal source the recorded activity of dozens to hundreds single neurons recorded with multielectrode recording arrays. Intracortical BCIs have recently shown impressive performance in early clinical trials [6-8].

These demonstrations of clinical feasibility have spawned renewed interest in the decoding problem: how should the mapping from neural activity to device movement be designed to maximize patient performance with the device? A variety of approaches have been suggested, including optimal linear estimators [9], Kalman filters [10, 11], particle filters [12], neural network decoders [13, 14], point-process filters [15, 16], kernel-based techniques [17], and others [18]. Differences in these approaches all stem from the different assumptions they make about how neural activity relates to desired movements. For example, if one assumes that neural activity encodes the desired movement velocity, the best approach might be either a velocity Kalman filter or a variant like the Laplace Gaussian filter [19], depending on whether one assumes that neural noise is best represented as Gaussian or Poisson. If one assumes instead that neural activity encodes intended joint torques, the best approach might be direct torque control or a hybrid thereof [20].

Nearly all existing decoders assume that neurons are driven by a "motor intent" signal, and recorded neural firing rates are treated as noisy observations of that underlying motor intent. Under this viewpoint, BCI design is properly treated as a signal estimation problem. An alternate view, however, is to consider the neural activity as a control signal that is sent via the spinal cord to the muscles. In this view, the neurons don't directly reflect an intended movement, although they may correlate with motor intent. Rather, the neurons represent the inputs that will drive the effector in the desired manner. While this may be a subtle distinction, it has powerful implications for BCI design. If neurons encode control signals, BCI design should not be approached as a signal estimation problem, but rather as a control-system design problem.

In this paper, we argue that the control-theoretic approach to BCI design may provide distinct advantages to the signal estimation approach, and present a general framework for designing BCI decoders as optimally-controllable systems. In section II we give a brief review of estimation-based BCI design, and list some of its shortcomings. In section III we argue that the brain may be considered as an optimal controller. In section IV, we present a mathematical definition of BCI usability, and reformulate the BCI decoding problem as a constrained optimization problem in control theory. In section V we present one solution to this problem in which we solve for the optimal dynamics of a second order linear decoder. We conclude in section VI.

II. Estimation-Based BCI Design

A. *Maximum Likelihood Approaches to BCI Design*

In this section we review the most commonly used BCI decoding algorithm, the velocity Kalman filter (VKF), taking care to specifically outline the assumptions it makes about the neural representation of movement. As the name implies, the VKF is among the class of BCI decoding algorithms that are based on signal estimation. Specifically, if one is recording the firing rates $\boldsymbol{y}_t \in \mathbb{R}^n$ from n neurons at time t, the VKF assumes that they will be related to the intended velocity of the end-point of the prosthetic effector $\boldsymbol{v}_t^* \in \mathbb{R}^d$ through the equation

$$\boldsymbol{y}_t = B\boldsymbol{v}_t^* + \boldsymbol{\varepsilon}_t. \tag{1}$$

Here the dimensionality of the velocity, d, is typically either 2 for movements constrained to a plane or 3 for movements in free space, and the noise in the fit is assumed to be zero mean Gaussian, $\boldsymbol{\varepsilon}_t \sim \mathcal{N}(\boldsymbol{0}, \Sigma)$.

---

*Research supported by the Craig H. Neilsen Foundation and PA Department of Health Research Formula Grant SAP #4100057653 under the Commonwealth Universal Research Enhancement program.

Y. Zhang is with the Robotics Institute of Carnegie Mellon University, Pittsburgh, PA, 15213 USA. (email: yinzhang@cs.cmu.edu).

S. M. Chase, to whom correspondence should be addressed, is with the Center for the Neural Basis of Cognition and the Department of Biomedical Engineering, Carnegie Mellon University, Pittsburgh, PA, 15213. (phone: 412-268-5512; e-mail: schase@cmu.edu).

In addition to the linear Gaussian assumption governing how neurons relate to intended velocity, the VKF also makes the assumption that the intended velocity evolves smoothly over time according to

$$\boldsymbol{v}_t^* = A\boldsymbol{v}_{t-1}^* + \boldsymbol{\zeta}_t, \tag{2}$$

where $\boldsymbol{\zeta}_t \sim \mathcal{N}(\boldsymbol{0}, \Pi)$ is also assumed to be Gaussian.

If the assumptions embodied in equations (1-2) hold true, the BCI design problem reduces to a Bayesian inference problem where the goal is to estimate the a posteriori probability of the intended kinematics when given the recorded spike counts. The solution in this case is, of course, the well-known Kalman filter [21], first introduced for BCI use by Wu et al. [10]. The Kalman filter provides an efficient recursive algorithm to compute the posterior probability $p\left(\boldsymbol{v}_t^* | \boldsymbol{y}_{1,\ldots,t}\right)$. The optimal estimate $\boldsymbol{v}_t$ of the intended kinematics $\boldsymbol{v}_t^*$ is the mean of this distribution. The position of the effector (either the computer cursor or the end-point of a robotic limb), $\boldsymbol{p}_t$, is taken as the integral of this estimated velocity. From these considerations, we can write down the state model that describes how the BCI cursor state evolves over time in response to inputs. We call this the BCI plant, and it takes the form

$$\begin{pmatrix}\boldsymbol{p}_{t+1}\\ \boldsymbol{v}_{t+1}\end{pmatrix} = \begin{pmatrix} I & \Delta I \\ \boldsymbol{0} & A - K_t BA\end{pmatrix}\begin{pmatrix}\boldsymbol{p}_t\\ \boldsymbol{v}_t\end{pmatrix} + \begin{pmatrix}\boldsymbol{0}\\ K_t\end{pmatrix}\boldsymbol{y}_t. \tag{3}$$

Here, Δ is the time interval for one step (often, ~30 ms or so), I is a $d \times d$ identity matrix, and K_t is the Kalman gain. Although technically the Kalman gain is a function of time, in practice it converges to a stable value within a few timesteps, and in fact in many BCI applications it is initialized to its stable value before decoding begins [22]. Typical converged values of $A - KBA$ are approximately 0.75 I.

B. Failures of the Estimation Framework for BCI Design

Estimation-based approaches to BCI design have performed quite well. In fact, by most metrics, estimation-based approaches have enabled cursor control that is within roughly a factor of two of the performance that can be achieved with the natural limb [3]. However, there are three observed features of BCI control that are hard to explain within the signal-estimation framework.

1) Certain algorithms perform worse than expected. In a survey of the performance achieved with different types of linear decoders, it turns out that control is superior with algorithms that correspond to simple physical systems relative to control with algorithms that do not [23].

2) The estimation-framework assumes static tuning. The motor system is capable of adapting to a wide variety of perturbations [for review, see 24, 25]. Learning also occurs under BCI control [26-28]. However, most BCI decoding algorithms assume static tuning. Thus, an algorithm designed to be optimal under static tuning may no longer prove to be optimal if the system is dynamic. Another consequence of learning is that algorithms that work well at predicting arm movements off-line may not be the algorithms that work well in a closed-loop setting [9], and further, algorithms that do not perform well off-line may turn out to be very good in closed-loop [29].

3) Biomimetic decoders are not guaranteed to be optimal. Finally, most decoders operate from a biomimetic standpoint, meaning they try to mimic natural limb control to the greatest extent possible. While this approach helps minimize training times so patients can develop control relatively quickly, it is not necessarily true that biomimetic algorithms are optimal. With learning, subjects can gain control of even non-intuitive BCI decoders [30-33].

To overcome these limitations, ideally, one needs an approach to BCI design that would take learning into account so that the decoder is optimal after learning has occurred. In the next two sections we present a method that should allow for provably-optimal BCI control, post-learning.

III. The Brain as an Optimal Controller

In 2002, Todorov and Jordan linked the field of control theory with the field of biological motor control, by showing how observed features of coordinated movements are predicted to arise from an optimal solution to a stochastic feedback control problem [34]. Subsequently, optimal feedback control theory has proven to be a useful framework for investigating behavioral motor control in a number of different domains [35-38]. Support for the idea that the brain should be considered as an optimal feedback controller includes the following. First, it has been found that task-deviations in kinematic trajectories tend to be left uncorrected if they are task irrelevant [39, 40], meaning that behavioral variance is larger along unconstrained task dimensions than it is along constrained task dimensions. This would make sense if the brain chose control signals to minimize overall effort (control signal magnitude), as it would require effort to compensate for errors, so it should not be done if those deviations do not directly impact performance. Second, movements have been shown to distribute optimally across redundant effectors, be they muscles of the wrist in a center-out movement task [41] or across the two arms when using both to control a single cursor [42]. Optimal here refers again to the idea that the brain would choose control signals to minimize effort, which would indicate that redundant control signals should each contribute some amount of push to the effector, as opposed to relying solely on a single control signal. Third, reflexive movements appear to be programmed to respond with gains that reflect the task geometry and goal [43-45]. This kind of flexible control over reflex arcs is predicted by optimal feedback control models.

Given these findings, here we assume the following. (1) The brain acts as an optimal controller, i.e., it emits control signals that are the minimum of a cost function. (2) Recorded neural activity in primary motor cortex is a direct reflection of this control signal, contaminated with noise (that may be signal dependent). (3) The end goal result of learning is to produce the optimal control signals for a given effector. With these assumptions, we can develop a rigorous framework for designing BCI interfaces. In the next two sections we will frame the BCI design problem as the result of a constrained optimization process, relate it to common problems in the field of optimal control, and solve for some optimal BCI parameters under a restricted set of assumptions.

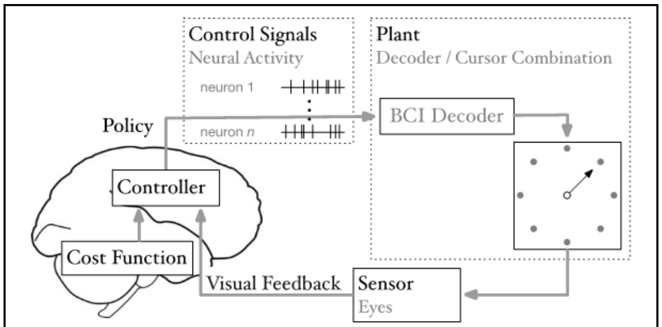

Figure 1: **Schematic of a BCI as a feedback control system.** The major components of a feedback control system are laid out on top of a typical BCI cursor control schematic.

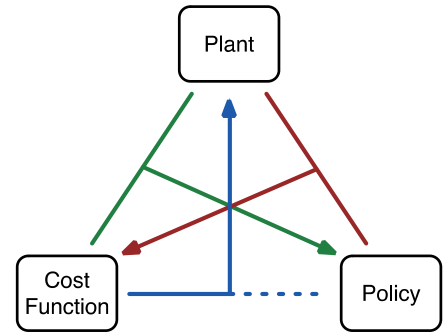

Figure 2: Cost function, plant, policy, with arrows connecting them to illustrate the various types of control theory problems. **Green**: optimal forward control. **Red**: inverse control. **Blue**: optimal plant design.

IV. Designing an Optimal BCI Mapping

In Fig. 1 we lay out a schematic of a typical BCI setup as a feedback control system. Target and cursor information are integrated by the brain, which chooses a control signal that minimizes some cost function. This control signal drives neurons in primary motor cortex. A subset of these neurons are recorded by a multi-electrode recording array and mapped through a decoder to control the state (position and velocity) of a cursor on a computer screen. Information about the new cursor state is then fed back to the brain by the visual system. Possible neuroanatomical sites of the controller and cost function operation are mentioned in [46].

A. Optimal Plant Design

Denoting x as the state of the cursor and u as the control signal, under the assumption that the brain acts as an optimal controller, it will output control signals, u^*, that are the minimum of some cost function $J(u)$. Formally, we say that

$$u^* = \operatorname*{argmin}_u J(u). \tag{4}$$

The cost function will take into account not only the relative energy or effort required by the control signal, but also the task goals, e.g., the desire to move the cursor to a target at a particular time. The optimal cost is defined as the cost on the optimal control signals,

$$J^* = J(u^*). \tag{5}$$

Naturally, because cursor state information is used in the computation of the cost, the optimal control signal will not only be a function of the task and state, but will also be a function of the plant: even if task goals and plant states are the same, plants with different parameters will result in different optimal control signals. We can explicitly represent the plant parameters, ϕ in the cost function to emphasize this point:

$$J^*(\phi) = \min_u J(u; \phi). \tag{6}$$

With this in mind, it becomes possible to define the usability of a plant: we say a plant with parameters ϕ_1 is more usable than a plant with parameters ϕ_2 if, for the same task, the expected cost under ϕ_1 is less than the expected cost under ϕ_2. Therefore we may write a formal definition of an optimal plant ϕ^* as

$$\begin{aligned}\phi^* &= \operatorname*{argmin}_\phi \mathbb{E}_{x_0,x^*}[J^*(\phi)] \\ &= \operatorname*{argmin}_\phi \mathbb{E}_{x_0,x^*}\left[\min_u J(u;\phi)\right],\end{aligned} \tag{7}$$

where the expectation here is taken over initial cursor states x_0 and cursor goal locations x^*. This would yield the expected cost over the range of movements that the device is likely to be used. Given that not all plant parameters may be possible due to physical or practical constraints, under the assumptions that the brain acts as an optimal controller, optimal BCI design amounts to solving the constrained optimization problem specified in (7).

B. Relationship to Other Problems in Control Theory

The optimal plant design problem is related to, but distinct from, typical classes of problems in control theory. In the forward control problem, one is given knowledge of the plant and the cost function, and the job is to find the control signals that achieve the task goals with minimal cost [47] (Fig. 2, green arrow). In the inverse control problem, one is given knowledge of the plant and a set of observed control signals that were issued for particular plant states, and the job is to infer the cost function for which those control signals would be optimal [48, 49] (Fig. 2, red arrow).

In the optimal plant design problem, one is given the cost function, and it is assumed that the ultimate plant will be operated in an optimal fashion. Thus, while the policy is not directly specified, it could be derived for any particular plant as a forward control problem. The goal of optimal plant design is to find the parameters of the plant that result in the least expected cost (Fig. 2, blue arrow).

V. Example: Optimizing BCI Dynamics

In this section we provide an example of how the control-theoretic approach might be used in BCI design. Specifically, we will derive the optimal plant dynamics of a second order linear decoder. We restrict ourselves to the class of static, second order linear plants because it provides a nice contrast to the standard VKF decoder presented in section II.

A general second order linear BCI decoder may be written as

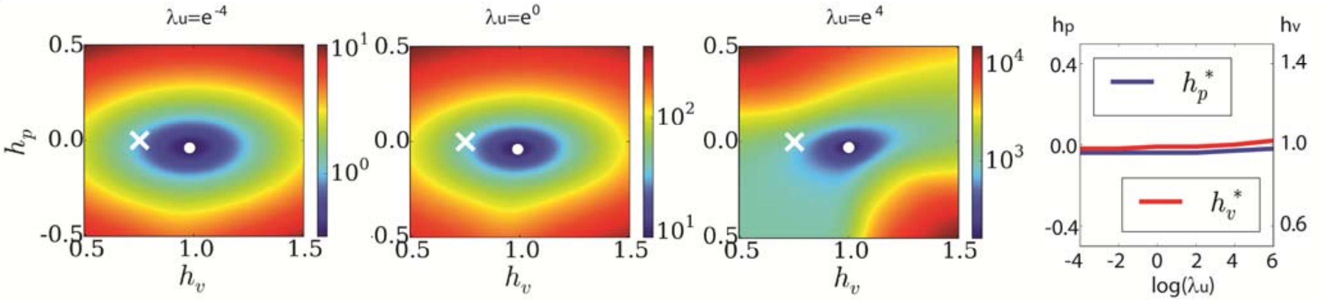

Figure 3: Simulation results. The leftmost three plots are heatmaps of the average optimal costs for BCI systems with different h_p and h_v values. Blue colors denote relatively small cost, red denotes relatively large cost. The different heatmaps show optimal costs under different λ_u values (shown at top). The white dot shows the optimal parameters for each value of λ_u, and the white cross shows the cost of the VKF parameters. The rightmost plot shows the optimal parameters as a function of λ_u.

$$\begin{pmatrix} \boldsymbol{p}_{t+1} \\ \boldsymbol{v}_{t+1} \end{pmatrix} = \begin{pmatrix} I & \Delta I \\ H_p & H_v \end{pmatrix} \begin{pmatrix} \boldsymbol{p}_t \\ \boldsymbol{v}_t \end{pmatrix} + \begin{pmatrix} \boldsymbol{0} \\ M \end{pmatrix} \boldsymbol{y}_t, \tag{8}$$

where the neural firing rates, $\boldsymbol{y}_t$, act as the control signals. We have constrained the decoder to be physical, meaning position is the integral of velocity, since decoders that correspond to simple physical systems seem to be easier to control than those that do not [23].

The free parameters of this decoder are the $d \times d$ matrix H_p which governs the spring-like properties of the plant, the $d \times d$ matrix H_v which governs the viscous properties of the plant, and M, the $d \times n$ matrix which determines how the neural activity affects the velocity of the cursor. By direct comparison to (3), we see that for the VKF, $H_p = \boldsymbol{0}$, $H_v = A - KBA$ and $M = K$.

Of course, (8) does not consider noise, which is inevitable in a real BCI system. For example, firing rates are known to have a Poisson-like distribution, where the variance of the noise depends on the overall mean rate. Other noise could come from the recording device, which is independent of the neural activity. Including these different kinds of noise in the BCI plant, we rewrite the system dynamics as

$$\begin{aligned} \begin{pmatrix} \boldsymbol{p}_{t+1} \\ \boldsymbol{v}_{t+1} \end{pmatrix} &= \begin{pmatrix} I & \Delta I \\ H_p & H_v \end{pmatrix} \begin{pmatrix} \boldsymbol{p}_t \\ \boldsymbol{v}_t \end{pmatrix} \\ &\quad + \begin{pmatrix} \boldsymbol{0} \\ M \end{pmatrix} \left(\boldsymbol{y}_t + \sum_{i=1}^{n} \sqrt{\kappa_i} \epsilon_{i,t} I_i \boldsymbol{y}_t + \boldsymbol{\omega}_t \right). \end{aligned} \tag{9}$$

Here, $\sum_{i=1}^{n} \sqrt{\kappa_i} \epsilon_{i,t} I_i \boldsymbol{y}_t$, represents the signal dependent noise, where I_i is a $n \times n$ matrix with 1 at location (i,i) and 0 elsewhere, $\epsilon_{i,t} \sim \mathcal{N}(0,1)$ is Gaussian noise, and $\kappa_i > 0$ controls the scaling of the signal dependent noise terms. This makes the firing rates approximately Poisson (but more tractable in a forward control setting). The second part, $\boldsymbol{\omega}_t \sim \mathcal{N}(\boldsymbol{0}, W)$, represents the signal independent noise.

Once the subject learns to control the BCI proficiently, he will generate control signals which minimize his internal cost function. In this paper, we consider the quadratic cost function, which has been widely used in motor learning studies [34, 42, 46], as

$$\begin{aligned} J(\boldsymbol{y}_{0,\ldots,T-1}) &= \sum_{t=0}^{T} \mathbb{E}_{\epsilon,\boldsymbol{\omega}} [\boldsymbol{x}_t^T Q_t \boldsymbol{x}_t] \\ &\quad + \sum_{t=0}^{T-1} \boldsymbol{y}_t^T R_t \boldsymbol{y}_t, \end{aligned} \tag{10}$$

where $Q_t \succcurlyeq 0$ and $R_t \succ 0$. Generally speaking, the first term corresponds to an accuracy cost (the penalty for not having the cursor at the goal), and the second term quantifies the total effort exerted during the task.

We are going to focus on optimizing the dynamics portion of this plant, H_p and H_v. A general algorithm that also optimizes M for a given set of recorded neurons is the subject of future work. To find the optimal dynamics H_p^* and H_v^*, we have

$$\begin{aligned} H_p^*, H_v^* &= \underset{H_p,H_v}{\operatorname{argmin}} \, \mathbb{E}_{\boldsymbol{x}_0,\boldsymbol{x}^*} [J^*(H_p, H_v)] \\ &= \underset{H_p,H_v}{\operatorname{argmin}} \, \mathbb{E}_{\boldsymbol{x}_0,\boldsymbol{x}^*} \left[\min_{\boldsymbol{y}_{0,\ldots,T-1}} J(\boldsymbol{y}_{0,\ldots,T-1}; H_p, H_v) \right]. \end{aligned} \tag{11}$$

To find the system with lowest optimal cost, we first need to compute the optimal cost for a particular set of parameter settings. This is a classical forward control problem. For a linear quadratic model, where the system dynamics is linear and the cost function is quadratic, the linear quadratic regulator (LQR) can give us the optimal solution via dynamic programming. The minimum cost under the optimal policy is

$$J^*(H_p, H_v) = \operatorname{tr}(P_0 X_0), \tag{12}$$

where $X_0 = \mathbb{E}(\boldsymbol{x}_0 \boldsymbol{x}_0^T)$ and P_0 is given by Riccati recursion (see [50] and the Appendix for details). The classic LQR has been extended for the signal-dependent noise case [51], for which the optimal cost becomes

$$J^*(H_p, H_v) = \operatorname{tr}(P_0 X_0) + \sum_{t=0}^{T-1} \operatorname{tr}(P_{t+1} W). \tag{13}$$

Details can be found in the Appendix and in [51].

Having dealt with the forward control problem, we now must optimize this cost over the system parameters H_p and H_v. For simplicity, we assume the device will be used symmetrically, which will lead to symmetry in the dynamics such that we can write $H_p = h_p I$ and $H_v = h_v I$, where h_p and h_v are scalars. We also know that when $h_p = 0$ there are no elastic effects and when $h_v = 1$ there are no damping effects, and likely solutions will not stray far from these points. Thus, for this simplified system we can brute-force search on (h_p, h_v) around $(0,1)$.

A. Task simulation

To demonstrate the performance, we can simulate the trajectories and costs that would arise with different plant parameters during the execution of a center-out / out-center point-to-point reaching task with the BCI. This is a fairly typical method of evaluating BCI performance [19]. In this task, the subject is required to move a cursor on a 2D computer screen from the center of the workspace to a target position p^* within $T_r = 20$ steps, and hold the cursor at p^* for another $T_h = 20$ steps. Then the subject needs to move the cursor back to the origin with T_r steps and hold there for another T_h steps. There are 8 targets uniformly distributed on a circle with radius equal to 10. The time interval Δ is set to be 0.1. We set the signal-dependent noise, κ_i, equal to 1 for all neurons to mimic Poisson spiking noise. The variance-covariance matrix of the signal-independent noise is set to be diagonal, i.e., $W = \sigma_\omega^2 I$ with $\sigma_\omega = 0.1$. We simulate 10 neurons with the preferred direction uniformly distributed on the circle and unit modulation depth, such that each column of M is the preferred direction of the corresponding neuron. The results do not depend strongly on the control-signal details.

To make this task solvable under linear-quadratic framework, the state is augmented with the target position p^* as $x_t = (p_t, v_t, p^*)^T$ and the cost function is

$$J = \sum_{t=T_r}^{T_r+T_h} \mathbb{E}\|p_t - p^*\|^2 + \lambda_u \sum_{t=0}^{T_r+T_h-1} \|My_t\|^2. \quad (14)$$

The first term measures the movement accuracy during the holding period, and the second term measures the total effort required to finish the task. λ_u controls the balance between those two terms within the cost function. We swept λ_u over a range from e^{-4} to e^{6}.

B. Simulation results

The expectation in (7) which determines that the optimal parameters will be optimal on average for the expected use of the device is handled in the simulation by the target distribution: the expectation is computed as the average cost over all movements. With this in mind, it becomes possible to compare costs for different parameter settings. In Fig. 3 we plot as a heatmap the cost for h_p and h_v ranging from -0.5 to 0.5 and from 0.5 to 1.5, respectively. The write dot indicates the optimal parameters. We can see that for a large range of λ_u, the optimal h_p^* is slight less than 0 and the optimal h_v^* is close to 1. The white cross indicates the typical parameters of the VKF, which is more than twice as costly as the optimal plant parameters.

Movement trajectories for three different system parameters are shown in Fig. 4. The distance here is the distance of the cursor to the start location, where the target location is indicated by the solid black line. The speed and the control signal are defined as $\boldsymbol{v}_t$ and $M\boldsymbol{y}_t$ projected onto the unit vector pointing from the starting position to the target position, respectively. In the top row, $h_p = 0$ and $h_v = 1$ are near the optimal parameters, and in the bottom row, $h_p = 0$ and $h_v = 0.75$ are near the typical parameters of the VKF. Comparing those two rows, we can see the distance and speed profiles are quite similar, though the VKF has slightly more variance in its speed profiles than the optimum plant. The major difference between the two algorithms comes from the control signals (right-hand column), which indicates that the total effort required by the VKF is higher than the effort required by the optimum plant. The VKF dynamics are equivalent to adding damping effects to the optimal system. For contrast, in the middle row we plot the optimal trajectories and control signals from an optimal system with some added elastic effects, where $h_p = 0.25$ and $h_v = 1$. We can see that although the control signals profiles for CO and OC are symmetric, the distance and speed profiles are not. To finish the task under such these parameters settings, not only must the total effort increase, but the movement accuracy also decreases.

Finally, it should be noted that the optimal plant parameters derived from the control-theoretic approach outlined in this paper cannot be achieved with a VKF. To achieve these settings with a VKF, A-KBA=I, meaning that either K=**0** or B=**0** (which would also force K to **0**). However, in a VKF, M=K (see (3)). To achieve zero viscous damping in the VKF, the control signal input must go to zero.

VI. Conclusion

Others have considered the dynamical control system properties of BCI and their impact on performance [16, 52, 53]. By treating the brain as an optimal controller, our work extends these approaches by introducing a rigorous definition of BCI usability, and recasting the BCI design problem as a constrained optimization problem over plant parameters. We used this approach to solve for the optimal dynamics of a 2D static linear decoder, and discovered that the optimal parameters are very close to a system with no spring or

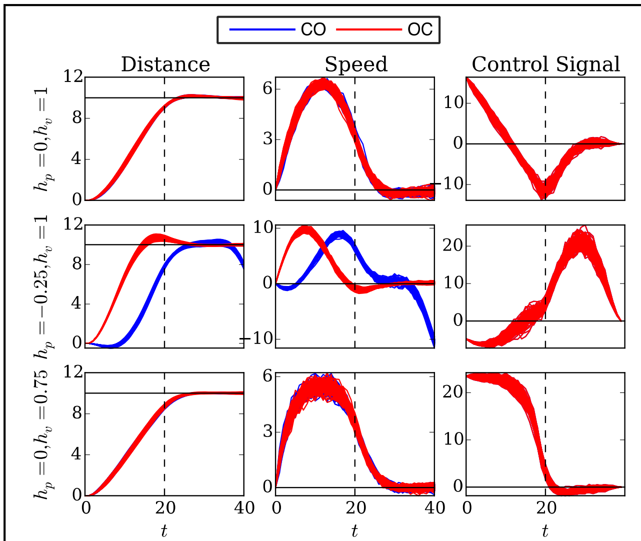

Figure 4: Examples of optimal BCI cursor control with three different plants. *Left column:* Distance from start location. Solid black line indicates the target distance. Vertical dashed line shows time target should be acquired. Red shows out-center (OC) movements, blue shows center-out (CO) movements. *Middle column:* Speed profiles as a function of time. *Right column:* Control signals as a function of time. *Top row:* Optimal plant parameters. *Middle row:* Plant with additional elastic terms. *Bottom row:* Plant with additional viscous terms, like the typical VKF.

damping qualities. These results are robust to a large range of noise values and effort vs accuracy cost weightings.

In one sense, these results might seem trivial: the optimal 2D linear BCI plant is an integrator that provides no resistance on the inputs. This is an intuitive solution given that we assumed no bias in the control signals and further assumed a minimal effort constraint. In another sense, the results are quite surprising: the optimal dynamics identified through this approach to BCI design are qualitatively different from those of the typical BCI decoder, the VKF. In fact, the parameters identified with our method fall outside of the allowable parameters of a VKF.

Although the BCI design problem can be recast as a constrained optimization problem, solving it is not necessarily trivial. Our simulation provided one solution in a situation that permitted brute-force search. For more complicated systems, gradient techniques can be employed, but are beyond the scope of this work. We also leave for future work the complicated problem of solving for the optimal mapping between neural activity and cursor movements. However, by considering the control-theoretic approach to BCI design, we hope to enable fundamental improvements in clinical BCI systems.

APPENDIX

In this appendix we present the modified Riccati recursion used to solve the linear quadratic forward control problem with signal dependent noise in our simulation. For further details, see [51].

Forward Control Algorithm

set $P_T = Q_T$.

for $t = T-1, \ldots, 0$ **do**

$$D_t = R_t + M^T P_{t+1} M + \sum_{i=1}^{n} \kappa_i I_i M^T P_{t+1} M\, I_i$$
$$L_t = D_t^{-1} M^T P_{t+1} H$$
$$P_t = Q_t + H^T P_{t+1}(H + ML_t)$$

The optimal policy: $y_t^*(x) = L_t x$

end for

REFERENCES

[1] J. K. Chapin, K. A. Moxon, R. S. Markowitz *et al.*, "Real-time control of a robot arm using simultaneously recorded neurons in the motor cortex," *Nat Neurosci*, vol. 2, no. 7, pp. 664-70, Jul, 1999.

[2] M. Velliste, S. Perel, M. C. Spalding *et al.*, "Cortical control of a prosthetic arm for self-feeding," *Nature*, vol. 453, no. 7198, pp. 1098-101, Jun 19, 2008.

[3] V. Gilja, P. Nuyujukian, C. A. Chestek *et al.*, "A high-performance neural prosthesis enabled by control algorithm design," *Nat Neurosci*, vol. 15, no. 12, pp. 1752-7, Dec, 2012.

[4] G. H. Mulliken, S. Musallam, and R. A. Andersen, "Decoding trajectories from posterior parietal cortex ensembles," *J Neurosci*, vol. 28, no. 48, pp. 12913-26, Nov 26, 2008.

[5] J. M. Carmena, M. A. Lebedev, R. E. Crist *et al.*, "Learning to control a brain-machine interface for reaching and grasping by primates," *PLoS Biol*, vol. 1, no. 2, pp. E42, Nov, 2003.

[6] J. L. Collinger, B. Wodlinger, J. E. Downey *et al.*, "High-performance neuroprosthetic control by an individual with tetraplegia," *Lancet*, vol. 381, no. 9866, pp. 557-64, Feb 16, 2013.

[7] L. R. Hochberg, D. Bacher, B. Jarosiewicz *et al.*, "Reach and grasp by people with tetraplegia using a neurally controlled robotic arm," *Nature*, vol. 485, no. 7398, pp. 372-5, May 17, 2012.

[8] T. Aflalo, S. Kellis, C. Klaes *et al.*, "Neurophysiology. Decoding motor imagery from the posterior parietal cortex of a tetraplegic human," *Science*, vol. 348, no. 6237, pp. 906-10, May 22, 2015.

[9] S. M. Chase, A. B. Schwartz, and R. E. Kass, "Bias, optimal linear estimation, and the differences between open-loop simulation and closed-loop performance of spiking-based brain-computer interface algorithms," *Neural Netw*, vol. 22, pp. 1203-1213, May 22, 2009.

[10] W. Wu, A. Shaikhouni, J. P. Donoghue *et al.*, "Closed-loop neural control of cursor motion using a Kalman filter," *Conf Proc IEEE Eng Med Biol Soc*, vol. 6, pp. 4126-9, 2004.

[11] Z. Li, J. E. O'Doherty, T. L. Hanson *et al.*, "Unscented Kalman filter for brain-machine interfaces," *PLoS One*, vol. 4, no. 7, pp. e6243, 2009.

[12] A. E. Brockwell, A. L. Rojas, and R. E. Kass, "Recursive bayesian decoding of motor cortical signals by particle filtering," *J Neurophysiol*, vol. 91, no. 4, pp. 1899-907, Apr, 2004.

[13] D. Sussillo, P. Nuyujukian, J. M. Fan *et al.*, "A recurrent neural network for closed-loop intracortical brain-machine interface decoders," *J Neural Eng*, vol. 9, no. 2, pp. 026027, Apr, 2012.

[14] J. Wessberg, C. R. Stambaugh, J. D. Kralik *et al.*, "Real-time prediction of hand trajectory by ensembles of cortical neurons in primates," *Nature*, vol. 408, no. 6810, pp. 361-5, Nov 16, 2000.

[15] X. Deng, D. F. Liu, K. Kay *et al.*, "Clusterless Decoding of Position from Multiunit Activity Using a Marked Point Process Filter," *Neural Comput*, vol. 27, no. 7, pp. 1438-60, Jul, 2015.

[16] M. M. Shanechi, A. Orsborn, H. Moorman *et al.*, "High-performance brain-machine interface enabled by an adaptive optimal feedback-controlled point process decoder," *Conf Proc IEEE Eng Med Biol Soc*, vol. 2014, pp. 6493-6, 2014.

[17] L. Shpigelman, Y. Singer, R. Paz *et al.*, "Spikernels: predicting arm movements by embedding population spike rate patterns in inner-product spaces," *Neural Comput*, vol. 17, no. 3, pp. 671-90, Mar, 2005.

[18] A. E. Brockwell, R. E. Kass, and A. B. Schwartz, "Statistical Signal Processing and the Motor Cortex," *Proc IEEE Inst Electr Electron Eng*, vol. 95, no. 5, pp. 881-898, May, 2007.

[19] S. Koyama, S. M. Chase, A. S. Whitford *et al.*, "Comparison of brain-computer interface decoding algorithms in open-loop and closed-loop control," *J Comput Neurosci*, vol. 29, no. 1-2, pp. 79-87, Aug, 2010.

[20] P. Y. Chhatbar, and J. T. Francis, "Towards a naturalistic brain-machine interface: hybrid torque and position control allows generalization to novel dynamics," *PLoS One*, vol. 8, no. 1, pp. e52286, 2013.

[21] R. E. Kalman, "A new approach to linear filtering and prediction problems," *J Fluids Eng*, vol. 82, no. 1, pp. 35-45, 1960.

[22] P. T. Sadtler, S. I. Ryu, E. C. Tyler-Kabara *et al.*, "Brain-computer interface control along instructed paths," *J Neural Eng*, vol. 12, no. 1, pp. 016015, Jan 21, 2015.

[23] Y. Zhang, and S. M. Chase, "Recasting brain-machine interface design from a physical control system perspective," *J Comput Neurosci*, Jul 5, 2015.

[24] J. W. Krakauer, and P. Mazzoni, "Human sensorimotor learning: adaptation, skill, and beyond," *Curr Opin Neurobiol*, vol. 21, no. 4, pp. 636-44, Aug, 2011.

[25] R. Shadmehr, M. A. Smith, and J. W. Krakauer, "Error Correction, Sensory Prediction, and Adaptation in Motor Control," *Annu Rev Neurosci*, vol. 33, pp. 89-108, Mar 18, 2010.

[26] S. M. Chase, R. E. Kass, and A. B. Schwartz, "Behavioral and neural correlates of visuomotor adaptation observed through a brain-computer interface in primary motor cortex," *J Neurophysiol*, vol. 108, no. 2, pp. 624-44, Jul, 2012.

[27] D. M. Taylor, S. I. Tillery, and A. B. Schwartz, "Direct cortical control of 3D neuroprosthetic devices," *Science*, vol. 296, no. 5574, pp. 1829-32, Jun 7, 2002.

[28] A. L. Orsborn, H. G. Moorman, S. A. Overduin *et al.*, "Closed-loop decoder adaptation shapes neural plasticity for skillful neuroprosthetic control," *Neuron*, vol. 82, no. 6, pp. 1380-93, Jun 18, 2014.

[29] M. D. Golub, B. M. Yu, A. B. Schwartz *et al.*, "Motor cortical control of movement speed with implications for brain-machine interface control," *J Neurophysiol*, vol. 112, no. 2, pp. 411-29, Jul 15, 2014.

[30] K. Ganguly, and J. M. Carmena, “Neural correlates of skill acquisition with a cortical brain-machine interface,” *J Mot Behav*, vol. 42, no. 6, pp. 355-60, Nov, 2010.
[31] K. Ganguly, and J. M. Carmena, “Emergence of a stable cortical map for neuroprosthetic control,” *PLoS Biol*, vol. 7, no. 7, pp. e1000153, Jul, 2009.
[32] P. T. Sadtler, K. M. Quick, M. D. Golub *et al.*, “Neural constraints on learning,” *Nature*, vol. 512, no. 7515, pp. 423-6, Aug 28, 2014.
[33] A. R. Marathe, and D. M. Taylor, “Decoding position, velocity, or goal: does it matter for brain-machine interfaces?,” *J Neural Eng*, vol. 8, no. 2, pp. 025016, Apr, 2011.
[34] E. Todorov, and M. I. Jordan, “Optimal feedback control as a theory of motor coordination,” *Nat Neurosci*, vol. 5, no. 11, pp. 1226-35, Nov, 2002.
[35] E. Todorov, “Optimality principles in sensorimotor control,” *Nat Neurosci*, vol. 7, no. 9, pp. 907-15, Sep, 2004.
[36] J. Diedrichsen, R. Shadmehr, and R. B. Ivry, “The coordination of movement: optimal feedback control and beyond,” *Trends Cogn Sci*, vol. 14, no. 1, pp. 31-9, Jan, 2010.
[37] T. P. Lillicrap, and S. H. Scott, “Preference distributions of primary motor cortex neurons reflect control solutions optimized for limb biomechanics,” *Neuron*, vol. 77, no. 1, pp. 168-79, Jan 9, 2013.
[38] F. Crevecoeur, and S. H. Scott, “Priors engaged in long-latency responses to mechanical perturbations suggest a rapid update in state estimation,” *PLoS Comput Biol*, vol. 9, no. 8, pp. e1003177, Aug, 2013.
[39] D. Liu, and E. Todorov, “Evidence for the flexible sensorimotor strategies predicted by optimal feedback control,” *J Neurosci*, vol. 27, no. 35, pp. 9354-68, Aug 29, 2007.
[40] R. J. van Beers, E. Brenner, and J. B. Smeets, “Random walk of motor planning in task-irrelevant dimensions,” *J Neurophysiol*, vol. 109, no. 4, pp. 969-77, Feb, 2013.
[41] A. H. Fagg, A. Shah, and A. G. Barto, “A computational model of muscle recruitment for wrist movements,” *J Neurophysiol*, vol. 88, no. 6, pp. 3348-58, Dec, 2002.
[42] J. Diedrichsen, “Optimal task-dependent changes of bimanual feedback control and adaptation,” *Curr Biol*, vol. 17, no. 19, pp. 1675-9, Oct 9, 2007.
[43] J. A. Pruszynski, and S. H. Scott, “Optimal feedback control and the long-latency stretch response,” *Exp Brain Res*, vol. 218, no. 3, pp. 341-59, May, 2012.
[44] M. Omrani, J. Diedrichsen, and S. H. Scott, “Rapid feedback corrections during a bimanual postural task,” *J Neurophysiol*, vol. 109, no. 1, pp. 147-61, Jan, 2013.
[45] J. Y. Nashed, F. Crevecoeur, and S. H. Scott, “Influence of the behavioral goal and environmental obstacles on rapid feedback responses,” *J Neurophysiol*, vol. 108, no. 4, pp. 999-1009, Aug, 2012.
[46] R. Shadmehr, and J. W. Krakauer, “A computational neuroanatomy for motor control,” *Exp Brain Res*, vol. 185, no. 3, pp. 359-81, Mar, 2008.
[47] B. D. O. Andersen, and J. B. Moore, *Optimal Control: Linear Quadratic Methods*: Prentice-Hall, 1989.
[48] P. Abbeel, and A. Ng, “Apprenticeship learning by inverse reinforcement learning,” *Proc. 21st International Conf on Machnie Learning*, 2004.
[49] K. Dvijotham, and E. Todorov, “Inverse optimal control with linearly solvable MDPs,” *Proc. 27th International Conf on Machnie Learning*, 2010.
[50] H. Kwakemaak, and R. Sivan, *Linear Optimal Control Systems*: Wiley-Interscience, 1972.
[51] E. Todorov, “Stochastic optimal control and estimation methods adapted to the noise characteristics of the sensorimotor system,” *Neural Comput*, vol. 17, no. 5, pp. 1084-108, May, 2005.
[52] M. M. Shanechi, G. W. Wornell, Z. M. Williams *et al.*, “Feedback-controlled parallel point process filter for estimation of goal-directed movements from neural signals,” *IEEE Trans Neural Syst Rehabil Eng*, vol. 21, no. 1, pp. 129-40, Jan, 2013.
[53] S. Gowda, A. L. Orsborn, S. A. Overduin *et al.*, “Designing dynamical properties of brain-machine interfaces to optimize task-specific performance,” *IEEE Trans Neural Syst Rehabil Eng*, vol. 22, no. 5, pp. 911-20, Sep, 2014.