

A Real-Time Brain-Machine Interface Combining Motor Target and Trajectory Intent Using an Optimal Feedback Control Design

Maryam M. Shanechi^{1,2,3,4,5*}, Ziv M. Williams^{4,5}, Gregory W. Wornell³, Rollin C. Hu^{4,5}, Marissa Powers⁴, Emery N. Brown^{5,6,7,8}

1 School of Electrical and Computer Engineering, Cornell University, Ithaca, New York, United States of America, **2** Department of Electrical Engineering and Computer Science, University of California, Berkeley, California, United States of America, **3** Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, Massachusetts, United States of America, **4** Department of Neurosurgery, Massachusetts General Hospital, Boston, Massachusetts, United States of America, **5** Harvard Medical School, Boston, Massachusetts, United States of America, **6** Department of Anesthesia, Critical Care and Pain Medicine, Massachusetts General Hospital, Boston, Massachusetts, United States of America, **7** Department of Brain and Cognitive Sciences, Massachusetts Institute of Technology, Cambridge, Massachusetts, United States of America, **8** Institute for Medicine, Engineering and Science, Massachusetts Institute of Technology, Cambridge, Massachusetts, United States of America

Abstract

Real-time brain-machine interfaces (BMI) have focused on either estimating the continuous movement trajectory or target intent. However, natural movement often incorporates both. Additionally, BMIs can be modeled as a feedback control system in which the subject modulates the neural activity to move the prosthetic device towards a desired target while receiving real-time sensory feedback of the state of the movement. We develop a novel real-time BMI using an optimal feedback control design that jointly estimates the movement target and trajectory of monkeys in two stages. First, the target is decoded from neural spiking activity before movement initiation. Second, the trajectory is decoded by combining the decoded target with the peri-movement spiking activity using an optimal feedback control design. This design exploits a recursive Bayesian decoder that uses an optimal feedback control model of the sensorimotor system to take into account the intended target location and the sensory feedback in its trajectory estimation from spiking activity. The real-time BMI processes the spiking activity directly using point process modeling. We implement the BMI in experiments consisting of an instructed-delay center-out task in which monkeys are presented with a target location on the screen during a delay period and then have to move a cursor to it without touching the incorrect targets. We show that the two-stage BMI performs more accurately than either stage alone. Correct target prediction can compensate for inaccurate trajectory estimation and vice versa. The optimal feedback control design also results in trajectories that are smoother and have lower estimation error. The two-stage decoder also performs better than linear regression approaches in offline cross-validation analyses. Our results demonstrate the advantage of a BMI design that jointly estimates the target and trajectory of movement and more closely mimics the sensorimotor control system.

Citation: Shanechi MM, Williams ZM, Wornell GW, Hu RC, Powers M, et al. (2013) A Real-Time Brain-Machine Interface Combining Motor Target and Trajectory Intent Using an Optimal Feedback Control Design. *PLoS ONE* 8(4): e59049. doi:10.1371/journal.pone.0059049

Editor: Wang Zhan, University of Maryland, College Park, United States of America

Received: August 30, 2012; **Accepted:** February 11, 2013; **Published:** April 10, 2013

Copyright: © 2013 Shanechi et al. This is an open-access article distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Funding: This work was supported by National Institutes of Health (NIH) Grant Nos. DP1-0D003646-01 and R01-EB006385. The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing Interests: The authors have declared that no competing interests exist.

* E-mail: shanechi@cornell.edu

Introduction

There has been a large body of work in the past decade on real-time brain-machine interfaces (BMI) demonstrating that neural signals from the motor cortical areas can be used to control computer cursors or robotic arms in human and non-human primates [1–22]. One type of such BMIs, which comprises most of this work, aims to estimate a continuous trajectory—for example the position of a computer cursor on the screen moving towards a visual target [1–14,19]. Recent efforts with this type of BMIs have demonstrated the ability to estimate continuous movement from motor cortical activity. The other type of BMIs aim to predict a desired discrete target without estimating the corresponding desired trajectory towards it [15,16] and are valuable for purposes such as typing on a keyboard. Recently, we designed another type

of BMI for sequential motor function that can concurrently decode the full motor sequence before movement initiation [20].

The successful real-time attempts at individual decoding of the continuous trajectory or the target of movement motivate the development of a new type of real-time BMIs that aim to estimate jointly both the trajectory of the movement and the intended target. This approach is justified by two main reasons. First, the activity in the motor cortical areas has been shown to be related to both target and kinematics of movement [2,15,23–43]. Peri-movement activity, i.e., the activity around the time of movement, in the primary motor cortex, posterior parietal cortex (PPC), and dorsal premotor cortex (PMd) is related to the movement kinematics such as direction, velocity, position, and acceleration [2,23–32]. In addition to perimovement activity, neural activity in

the PPC has been shown to encode the intended target [15,33–36] prior to movement initiation. Similar activity has also been observed in the premotor cortex including PMd [28,30,37–43]. Second, this approach more closely mirrors the natural way in which the sensorimotor system decides on a plan of action and executes its movement. In other words, the several components of the musculo-skeletal system are coordinated in order to reach a target and hence the target of a movement and the desired trajectory to reach it are strongly correlated [44,45]. Indeed, there has been a body of offline work demonstrating the advantage of combining both target and trajectory related information in the decoder using either simulated neural data in [46–48] and in our work [49,50], or previously recorded neural data in [11,51]. We also presented promising results of a real-time BMI that jointly decodes the target and trajectory in [52–54].

In addition to modeling both the target and trajectory information, similar to the natural sensorimotor system, a BMI system can be modeled as a feedback control system. When using a BMI, the subject (controller) decides on the control commands and consequently modulates the neural activity to move the prosthetic device to a desired target (i.e., achieve the task goal) while receiving real-time visual feedback of the state of the movement. Based on these considerations, a more principled BMI design that aims to mirror the sensorimotor control system and jointly decode the movement target and the corresponding trajectory would allow for a potentially more accurate movement execution.

Here we develop a real-time BMI that uses a novel optimal feedback control design and combines information about target and trajectory intent, and demonstrate its implementation in sensorimotor tasks performed by two rhesus monkeys. This BMI employs a novel two-stage approach. In the first stage, it uses the neural spiking activity prior to movement initiation to predict the intended target of the movement. In the second stage, it combines this prediction with the peri-movement spiking activity to estimate the movement trajectory. To decode the trajectory, inspired by the optimal feedback control theory of the sensorimotor system [44,45,55–57], we build an optimal feedback-controlled state-space model for goal-directed movements, which we use in a recursive Bayesian decoder. We have derived and presented the algorithmic details of this decoder, used in the second stage of the BMI, in [49,50] in a simulation study and assuming knowledge of the intended target, and in [58]. We implement the combined two-stage BMI for decoding movements in an instructed-delay center-out task. The two-stage BMI processes the spikes directly in real time, i.e., at the millisecond time-scale on which the neural spiking activity is recorded. Here, we show that the two-stage BMI performs better than either stage alone, demonstrating the advantage of combining both target and trajectory related information in real time. The optimal feedback control design results in trajectories that are smoother, have lower estimation error, and acquire the target more accurately. As a baseline, we also make offline comparisons to a linear ridge regression decoder [11,19], a regularized variant of the commonly used linear least squares regression decoder [2–5,7], on the training sessions (manual control) data and show that the two-stage decoder outperforms the regression decoder.

Results

We measured the performance of our BMI in an instructed-delay center-out directional task in which two monkeys used a joystick to move a cursor from the center of the screen to one of four targets displayed at its periphery (see Materials and Methods). Unlike many BMI motor tasks in which the subject can freely

move until reaching a target, this task required the monkey to reach the correct target *without* touching any of the incorrect targets under a limited time constraint. Hence only trajectories that reached the correct target and at no point touched an incorrect target placed at the other three sides of the screen were rewarded (Figure 1A). A performance measure used in these experiments was the acquisition accuracy, which is the percentage of trials on which the task is successfully completed. Multi-electrode spiking activity was recorded from PMd and the supplementary motor area (SMA) from which 20 ± 2 neurons (mean \pm s.d.) were isolated and used. At the beginning of each day, the monkey first performed the standard task using a joystick (training session) during which target and kinematic neural models were constructed. The monkey then performed the same task as before but this time cursor position was controlled by the neural activity recorded from the monkey (BMI sessions; Figure 1A) and the monkey received visual feedback of the cursor on the screen.

Jointly Decoding the Target and Trajectory Using an Optimal Feedback Control Design

The BMI processing consisted of two stages and decoded two aspects of movement. During the first stage, it used a maximum-likelihood (ML) decoder based on a point process model of the neural spiking activity to predict the monkey's intended target of movement during the delay period after target presentation but before movement initiation (see Materials and Methods). In the second stage, the BMI combined this decoded target with the peri-movement activity to estimate the trajectory. In this stage, the spiking activity of each neuron was modeled as a point process fitted to position and velocity. The BMI estimated the trajectory using an optimal feedback control design that combined the decoded target with the peri-movement activity. This design is inspired by the optimal feedback control theory of the sensorimotor control system used to explain its function [44,45,55–57,59], and is cognizant of the fact that in the BMI context the system to be controlled is the BMI instead of the musculo-skeletal system (Figure 1B; see Materials and Methods for details). The result is a recursive Bayesian decoder that we term the feedback-controlled parallel point process filter (FC-P-PPF). We introduced this decoder in [49,50] assuming knowledge of the target and using simulated neural spiking activity. Here we implement this decoder as the second stage in the two-stage real-time BMI and show its performance in combination with target prediction and using neural recordings in instructed-delay center-out tasks both offline and in real time. In the BMI experiments, the decoder updated the estimated position of the cursor in fine-scaled steps of 5 ms in real time, which was also used to bin the spikes.

Model Training and Validation

Models for the BMI were trained on the neural spiking activity during the training session at the beginning of each day. Neural recordings were made during target presentation prior to the presentation of the “go” cue, which signaled that the monkey could move the joystick, as well as during movement itself after the “go” cue (Figure 1A).

Each training session consisted of an average of 89 ± 2 trials. Point process models relating the spiking activity of each recorded neuron to target location and movement kinematics were constructed based on the known target location and cursor position for each trial and the recorded multiple-neuronal activity using the generalized linear models (GLM) framework [60]. Models were then cross-validated (leave-one-out) on the same data by finding the corresponding target predictions and kinematic estimates. Target location was predicted using the ML decoder

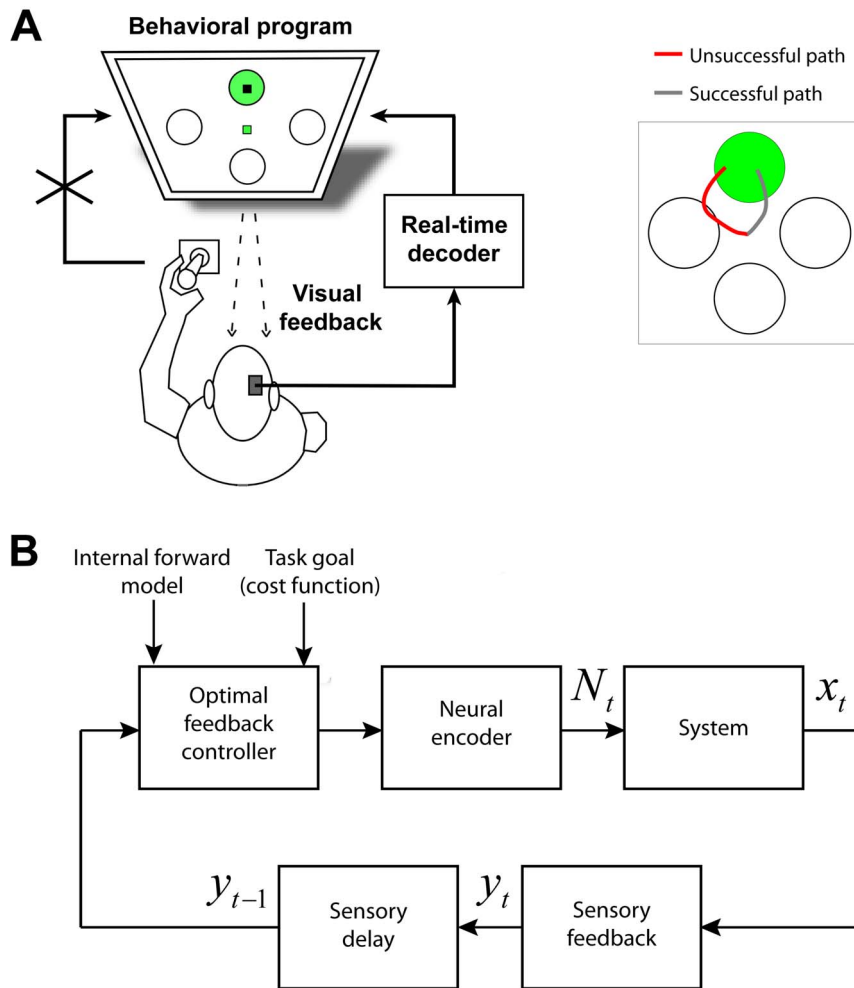


Figure 1. Experimental task and the optimal feedback control model. (A) Experimental task. The experiment consisted of an instructed-delay center-out task with four targets (*left*). To be rewarded, the monkey not only had to acquire the correct target, but also had to avoid touching any of the incorrect targets first (example successful and unsuccessful paths are shown on the *right*). After target presentation, there was 1 s of delay before the “go” cue, signaling that the monkey could begin moving the joystick. During the training sessions, the monkey controlled the position of the cursor using a joystick. During the BMI sessions, the joystick was disconnected and the real-time decoder controlled the cursor. (B) Optimal feedback control framework to model the BMI. An optimal feedback control framework is used to model the BMI motor task. In this framework, each task is performed to accomplish a goal during which there is real-time sensory feedback (e.g., visual feedback), y_t , about the state of the system to be controlled (e.g., BMI), x_t . Based on the intended goal, the internal forward model about the dynamics of the system, and the sensory feedback about the state of the system, the brain (controller) decides on a control command, which is reflected in its neural activity, N_t , and controls the system (see Materials and Methods). Here we have assumed that the brain receives perfect sensory feedback of the state, i.e., $y_t = x_t$.
doi:10.1371/journal.pone.0059049.g001

from the neural spiking activity in the 800 ms delay period prior to the “go” cue. Kinematics were estimated using either the two-stage decoder that combined the target predictions from the first stage with the peri-movement activity, or its second stage but not taking into account the target predicted from the first stage.

During the delay period, the ensemble spiking activity (20 ± 2 neurons) predicted the correct target with high accuracy in the training sessions (leave-one-out cross-validation). The prediction accuracy of the trained point process target models across sessions (see Materials and Methods), measured as the percentage of trials on which the models correctly predicted the target using the delay neural activity, was $81 \pm 3\%$ (mean \pm s.d.). To examine further the contribution of the individual neurons to the target prediction accuracy, we performed a neuron dropping analysis in which the spiking activity of a single neuron during the delay period was used to decode the target (Figure S1A–D). We found that across sessions, $48 \pm 12\%$ of the neurons had a target prediction accuracy

significantly greater than chance ($P < 0.05$). We further found that relatively few neurons (on average 3.3 ± 1.0 across sessions) were sufficient to obtain a target prediction accuracy that was higher than 90% that of the ensemble (Figure S1E).

During movement, the premotor neurons were tuned to position and velocity. Fitting the point process models for the kinematics using the GLM framework [60] (see Materials and Methods), we found that across sessions $47 \pm 16\%$ of the premotor neurons were significantly tuned to either position or velocity at least in one dimension ($P < 0.05$; Bonferroni correction for multiple comparisons). Of these neurons, 57% were tuned to position only, 15% were tuned to velocity only, and 28% were tuned to both position and velocity. In agreement with previous studies, these findings suggested that the recorded premotor neurons held significant information about both the target and kinematics of the movement.

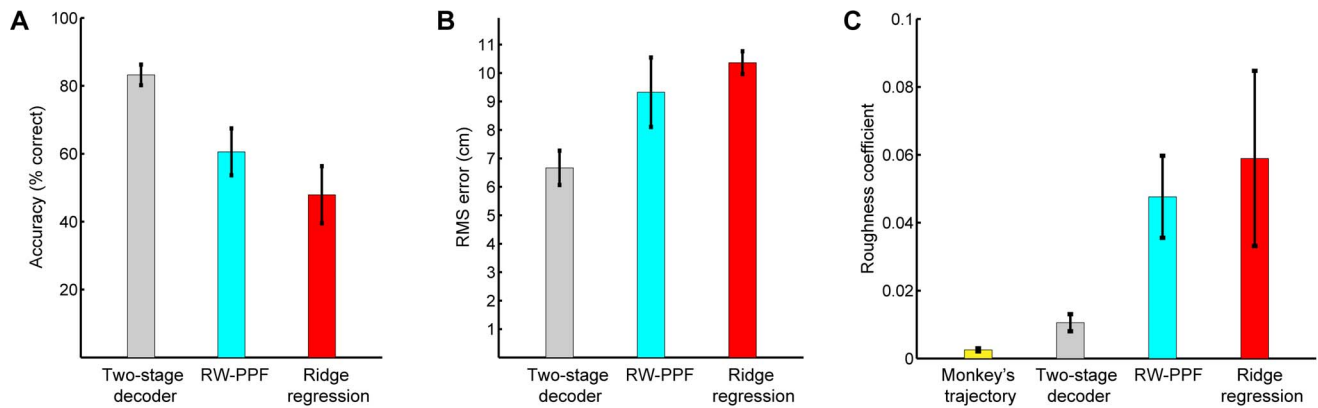


Figure 2. Offline model comparisons. The bars show mean quantities and the error bars show the standard deviation (s.d.) around the mean across sessions. All quantities are obtained from the training sessions using leave-one-out cross-validation. (A) Accuracy of the different models. (B) RMS error of the different models. (C) Roughness coefficient of the different models. The two-stage decoder (used in the real-time BMI) outperforms all other models in terms of accuracy, RMS error, and smoothness. doi:10.1371/journal.pone.0059049.g002

Offline Model Comparisons

We tested the performance of the two-stage decoder in an offline analysis of the training sessions data using leave-one-out cross-validation. We also compared to the performance of the second stage of the decoder alone by replacing the feedback-controlled state-space model in the FC-P-PPF (see Materials and Methods) with a random-walk (RW) model, which uses no prior target information and only enforces smoothness in the trajectory. The resulting filter is the RW-PPF (see Materials and Methods). We also compared the performance of the two-stage decoder to that of the linear ridge regression decoder [11,19,61] that is a regularized variant of the commonly used least-squares linear regression decoder [2–5,7] (see Materials and Methods). For each decoder we updated the position estimate every 5 ms, which was also the bin width for the spiking activity. Note that the chance level acquisition accuracy in our task using any decoder is at most 25% since there are four targets on the screen and hitting the wrong one results in an error. We also confirmed that estimating the trajectory from shuffled neural activity using a regression decoder results in approximately this chance level accuracy (see Materials and Methods).

We found that the acquisition accuracy of the two-stage decoder across sessions was $83 \pm 3\%$, higher compared to $61 \pm 7\%$ for the RW-PPF that used only the peri-movement activity (one-sided McNemar test, $P < 10^{-15}$; Figure 2A). Hence target predictions from the first stage of the decoder resulted in a correction rate of $(83 - 61)/(100 - 61) \approx 56\%$ for the inaccurate trajectory estimation in the second stage (Figure 3B). More specifically, while 15% of the trials with correct RW-PPF performance were not acquired correctly by the two-stage decoder, 80% of the trials with incorrect RW-PPF performance were corrected in the two-stage decoder, resulting in an overall improvement. Also, the trajectories estimated by the two-stage decoder were closer to the monkey's trajectory (Figure 3A,B and Figure 4). To quantify this, we measured the average root mean-square (RMS) error across all trials (Figure 2B). We found that the RMS error of the RW-PPF on average was 40% higher than the two-stage decoder (one-sided t -test, $P < 10^{-15}$). As an alternative measure of error between the estimated and the true trajectories, we also computed the signal-to-noise ratio (SNR) that is calculated as the ratio of the desired signal (joystick position) variance and the mean-squared estimation error, $SNR = 10 \log_{10} \frac{\text{signal variance}}{\text{mean-square error}}$, and is used in a number of

BMI studies [13,62,63]. We found, consistently, that the SNR of RW-PPF was lower than the two-stage decoder (one-sided t -test, $P < 10^{-15}$; Figure S2). While first-stage target prediction alone does not estimate a continuous trajectory, we can compare it to the two-stage decoder in terms of target acquisition accuracy. Making this comparison, we found that the second stage resulted in a correction rate of $(83 - 81)/(100 - 81) \approx 11\%$ for the target prediction errors (one-sided McNemar test, $P < 10^{-3}$; Figure 3C; Figure S3). Here, only 0.2% of the trials with correct target prediction were not acquired correctly by the two-stage decoder, while 12% of the trials with incorrect target prediction were corrected in the two-stage decoder. However, as these were offline estimation of fast joystick movements, the more appropriate test for the correction of the incorrect target predictions by the peri-movement activity in the second stage is in real-time BMI sessions (see below).

We also compared the performance of the two-stage decoder to that of a linear ridge regression decoder (Figure 3). The ridge regression decoder reconstructed the position at each time as a linear function of the history of the ensemble firing rates, which were calculated every 5 ms in sliding bins of 100 ms (see Materials and Methods for more detail). We included up to 800 ms—same duration as the delay period used for target prediction—of history coefficients in the regression decoder. Specifically, we found the performance of the ridge regression decoder using 200 ms, 400 ms, 600 ms, and 800 ms of history coefficients and selected the number of history coefficients that minimized the mean-square error using leave-one-out cross-validation. The average accuracy of the ridge regression decoder across sessions was $48 \pm 8\%$, which was lower than the two-stage decoder (one-sided McNemar test, $P < 10^{-15}$; Figure 2A). Also the average RMS error of the ridge regression decoder was 55% higher than the two-stage decoder (one-sided t -test, $P < 10^{-15}$; Figure 2B). Consistently, the SNR of the ridge regression decoder was lower than the two-stage decoder (one-sided t -test, $P < 10^{-15}$; Figure S2). Note that the ridge regression decoder uses the delay period activity due to its history coefficients.

Finally the trajectory estimations in the two-stage decoder were smoother than either the RW-PPF or the ridge regression decoder (Figure 3). To quantify this, we calculated the average roughness coefficient [64] (see Materials and Methods) for each of the decoders (Figure 2C). The roughness coefficient measures the

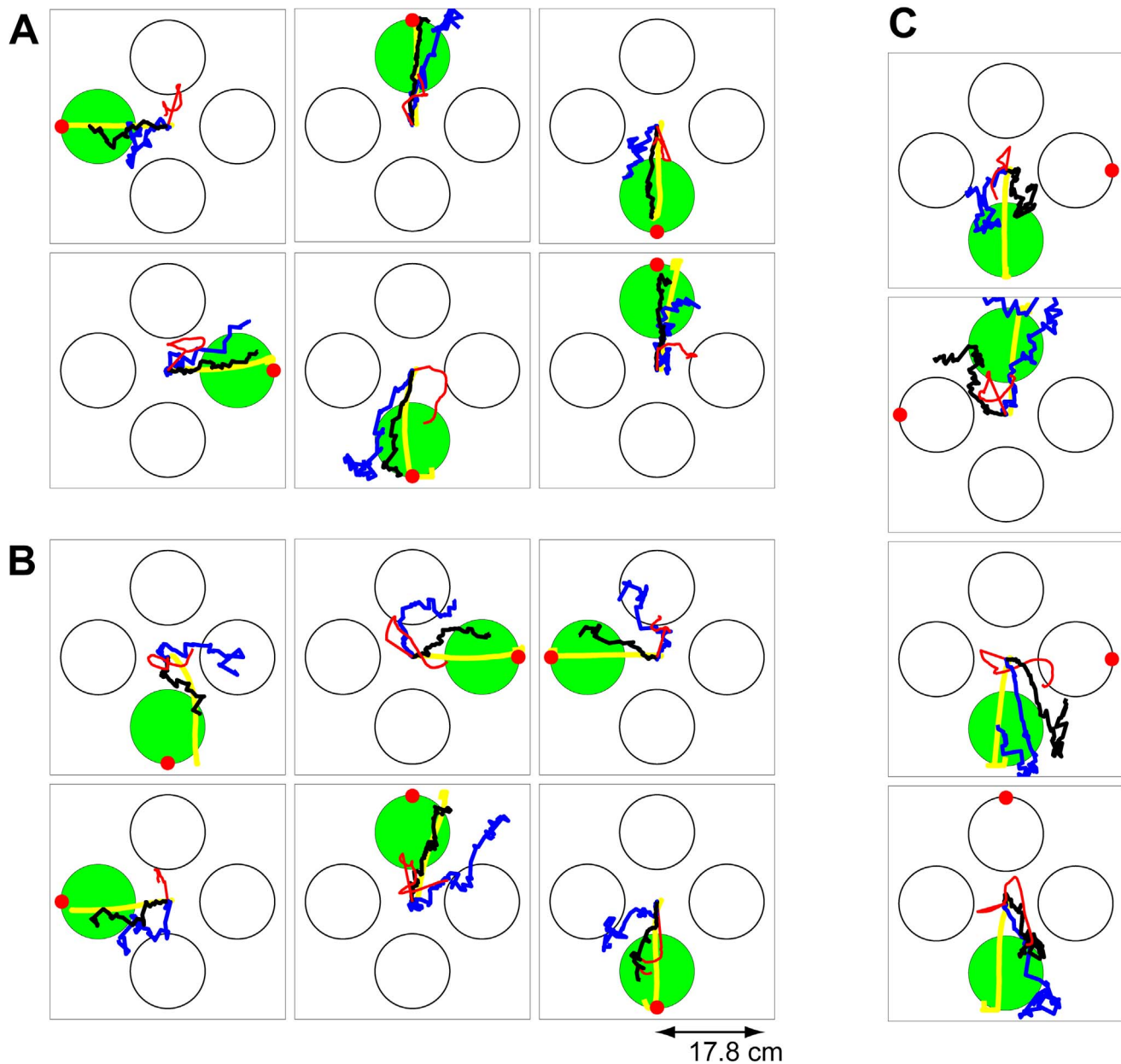


Figure 3. Comparison of the offline trajectory estimates. The green circle shows the instructed target and the yellow line shows the monkey's trajectory. The black line shows the trajectory estimate using the two-stage decoder, the red circle shows the predicted target from the first stage, the blue line shows the trajectory estimate of RW-PPF (i.e., the second stage of the decoder without using the target prediction), and the red line shows that of the linear ridge regression decoder. (A) Sample trials in which both the two-stage decoder and RW-PPF acquire the target correctly. (B) Sample trials in which the two-stage decoder acquires the target correctly but RW-PPF does not. (C) Sample trials in which the two-stage decoder acquires the target correctly but the target is predicted incorrectly from the first stage.
doi:10.1371/journal.pone.0059049.g003

degree of smoothness in the estimated trajectory and is smaller for smoother estimates. We found that the average roughness coefficients of the RW-PPF and the linear ridge regression decoder were 4.5 and 5.6 times larger than that of the two-stage decoder, respectively (one-sided *t*-test, $P < 10^{-14}$ in both cases).

Combined Target and Trajectory Decoding in a Real-Time BMI

To investigate whether kinematic and target related activity can be jointly used to obtain accurate motor performance in real time,

monkeys performed the same task as before but using a BMI. The task was again a center-out task requiring the monkey to move the cursor using the BMI to the correct target without touching any of the incorrect targets. The real-time BMI used the two-stage decoder. During the 800 ms delay period prior to the “go” cue, the BMI predicted the target and after the “go” cue it combined this target information with the peri-movement activity using the optimal feedback control design (FC-P-PPF) to decode the trajectory. Note that the cursor was held at the center during the delay period. We found that using the two-stage BMI, the monkeys could perform the task with an average accuracy of

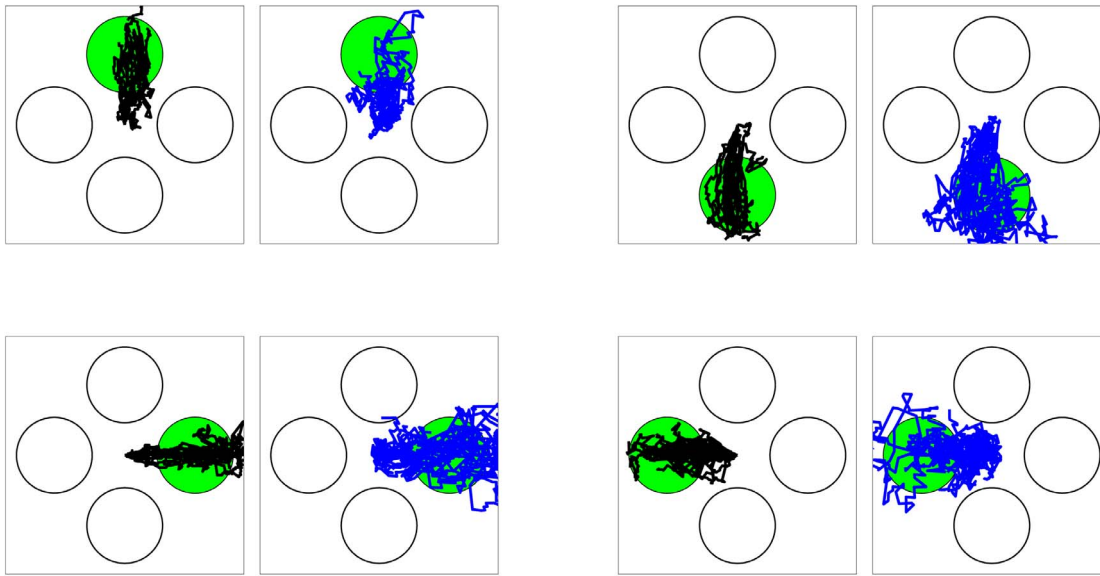


Figure 4. Trajectory estimates and their variations. The green circle shows the instructed target. The black lines show the trajectories estimated using the two-stage decoder in multiple trials in which the decoder is correct. The blue lines show the trajectories estimated by RW-PPF (i.e., the second stage of the decoder without using the target prediction) in multiple trials in which RW-PPF is correct. doi:10.1371/journal.pone.0059049.g004

77±9% (Figure 5A). To assess the stability of the performance throughout the recordings per day, we compared it in the first and second half of sessions. We found that accuracy did not change significantly (Wilcoxon rank-sum test, $P>0.5$) and remained stable.

We found that the two stages of the BMI performed in a complementary manner to achieve its overall accuracy; the correct target predictions could compensate for the inaccurate performance of the kinematic decoder and the ongoing trajectory estimation could correct the incorrect target predictions. To illustrate this complementary property of the BMI and how its two stages contributed to its overall performance, we compared with the performance of each of its stages alone using either the target-related delay or the kinematic-related peri-movement activities. To compare the accuracy of the BMI with that of using only the peri-movement activity, i.e., using the second stage of the BMI alone without target information, we decoded the trajectory offline using RW-PPF on the same real-time data set (Figure 5B–D). The average acquisition accuracy of the RW-PPF was $61 \pm 31\%$ (Figure 5A). This suggests that the first stage of the BMI resulted in a correction rate of $(77 - 61)/(100 - 61) \approx 41\%$ for the second stage errors (Figure 5A,C). More specifically, while 14% of the trials with correct RW-PPF performance were not acquired correctly by the BMI, 63% of the trials with incorrect RW-PPF performance were acquired correctly by the BMI, resulting in an overall improvement. While the first stage of the BMI alone (i.e., real-time target prediction) cannot estimate the continuous trajectory, we can still compare it to the two-stage BMI in terms of target acquisition accuracy. The average real-time target prediction accuracy of the first stage of the BMI was $72 \pm 3\%$. This indicates that the second stage of the BMI (FC-P-PPF) resulted in a correction rate of $(77 - 72)/(100 - 72) \approx 18\%$ for the target prediction errors of the first stage (Figure 5A,D). More specifically, while 4% of the trials with correct target prediction were not acquired correctly by the BMI, 27% of the trials with incorrect target prediction were acquired correctly by the BMI, resulting in an overall improvement. Hence the joint performance

of the BMI was higher than either stage alone using either kinematic or target related activity. Also the trajectories estimated by the BMI were smoother than those of the RW-PPF and had a significantly lower roughness coefficient.

The acquisition time in the BMI sessions, i.e., time until the trial ended by rewarding the monkey, was close to the natural acquisition time in the training sessions. In our experiments we used a short 3 s time-out condition to make the task more challenging and the required acquisition time closer to that of monkey's own movement. The median acquisition time for the natural movement was 0.6 ± 0.3 s and for the two-stage BMI was 0.9 ± 0.5 s.

Control Comparisons

To determine differences in BMI performance across monkeys and therefore the robustness of the two-stage BMI to individual variability, we examined differences in performances between the two monkeys, A and B. For monkey A, the real-time BMI accuracy (percentage of trials in which the correct target was acquired without the cursor touching the incorrect targets) was $67 \pm 4\%$. Comparing to the second stage alone using only the peri-movement activity, we found that the accuracy of the RW-PPF on the real-time data set for this monkey was $29 \pm 1\%$. Hence, even though the kinematic tuning in this monkey was weak, the real-time BMI had a relative high accuracy. This showed that the first stage of the BMI resulted in a correction rate of $(67 - 29)/(100 - 29) \approx 54\%$ for the inaccurate performance of the trajectory decoder. The real-time target prediction accuracy of the first stage using only the delay activity in this monkey was $71 \pm 1\%$. Note that because of the weak kinematic tuning in the recorded neurons in this monkey, the correction happened only by the first stage. For monkey B, the real-time BMI accuracy was $82 \pm 5\%$. Comparing to the second stage alone using only the peri-movement activity, we found that the accuracy of the RW-PPF in this monkey was $76 \pm 24\%$ and hence the first stage of the BMI resulted in a correction rate of $(82 - 76)/(100 - 76) \approx 25\%$ for the inaccurate trajectory estimation of the second stage. Comparing to

the average real-time target prediction accuracy of the first stage, which was $73 \pm 4\%$, suggested that the second stage of the BMI in monkey B resulted in a correction rate of $(82 - 73)/(100 - 73) \approx 33\%$ for the target prediction errors in the first stage. Therefore, in this monkey both stages exhibited the corrective behavior.

Together these findings suggest that although the performance of one stage in the two-stage BMI may not be equal across monkeys or recording areas, by combining both target and trajectory related information the two-stage approach provides a robust computational system that maintains good accuracy under variable experimental conditions (see Figure S4 for a comment on ridge regression).

Discussion

Based on our understanding of the sensorimotor system [44,45,55–57,59], natural movement incorporates information about the intended target as well as the trajectory of the movement. We implemented a novel real-time BMI designed to mimic the sensorimotor control system by a two-stage approach: First the activity prior to movement initiation is used to predict the intended target, and second this prediction is combined with the peri-movement spiking activity to estimate the trajectory using an optimal feedback control design. This is to our knowledge the first time that sensorimotor optimal feedback control principles have been used to decode the intended movement from neural recordings. It is also the first time that combined estimation of target and trajectory has been done in real time.

To decode the movement, we used an optimal feedback control design inspired by the optimal feedback control theory of the sensorimotor system [44,45]. In this view, each task is performed to accomplish a goal during which there is sensory feedback about the external state of the system. Specifying an approximate forward dynamics model, modeling the real-time sensory feedback about the state of the system, and quantifying the task goals as cost functions and the desired time to accomplish them, we can predict the next plan of action or control command by finding the one that minimizes the cost function. For a BMI, the same framework can be applied to predict the next plan of action. The difference is that the system being controlled is the BMI instead of an individual's own musculo-skeletal system. Hence the individual's next plan of action is reflected in the neural activity, this activity controls the BMI, and the individual in turn receives real-time visual feedback of the resulting movement (Figure 1B). Also, the BMI does not have knowledge of the desired time to accomplish the goal, which is decided by the controller (the individual). The present BMI hence resolved this movement duration uncertainty based on the neural spiking activity in real time (see Materials and Methods; [49,50]). The BMI also processed the spiking activity directly and hence operated at the millisecond time-scale of the spikes. In addition to its application to interpreting the sensorimotor function, optimal feedback control has also been deemed valuable for interpreting the neural basis of movement in the motor cortical areas [65]. This further motivates the use of optimal feedback control principles for the design of real-time BMIs.

In addition to modeling the BMI as an optimal feedback control system, we also decoded both target-related delay and kinematic-related peri-movement activities using a two-stage design. We

demonstrated that the two stages in the BMI functioned in a complementary manner. When the spiking activity for one stage was less informative, the other stage often provided sufficient information for the BMI to reach the correct target. As a result, the two-stage BMI performed better than either stage alone. Overall, the estimated trajectories using the two-stage optimal feedback control design were more accurate, had lower RMS error, and were smoother than the linear ridge regression decoder or a random-walk point-process decoder.

Unlike “free-roaming” motor tasks in which subjects could move freely until reaching a target, the present task was demanding in that at no point the trajectories could touch the incorrect targets and then proceed to the correct target. This was considered an incorrect response (Figure 1A). In addition the response time was constrained. Despite this, the monkeys were able to achieve a relatively high accuracy using the two-stage BMI ($77 \pm 9\%$). This accuracy was obtained by using relatively few neurons (11 on average) that were tuned to either target or trajectory.

Our BMI used direct point process modeling of the spiking activity. It hence processed the spikes directly in real time as opposed to a rate function calculated from these spikes as is done in previous real-time BMI work. Recent work [66] has demonstrated that reducing the bin width used to calculate the firing rates of the spiking activity, which are in turn used as input in a BMI, improves its performance. An interesting question for future investigation in real-time experiments is therefore whether moving to the time-scale on which the spiking activity is recorded, i.e., processing the spikes directly, could improve the performance of real-time BMIs.

The two-stage decoder combined both the target-related activity prior to movement and the trajectory-related peri-movement activity in real time. This combination makes the two-stage decoder robust to variations across recording conditions. Hence in scenarios where one type of activity is not strongly tuned in a recording area, it could still be possible for the decoder to result in acceptable performance. For example, if the delay activity is not strongly tuned to the target in an experiment, the second stage, i.e., the feedback-controlled parallel point process filter, can still be used to model the BMI target-directed movement and decode it from the peri-movement activity. In this case the decoder would need to additionally put a prior distribution on the target locations since no target information can be obtained prior to movement (see Materials and Methods and [51]). It is also important to note that in certain applications, such as key selection, there may be no need for estimating the continuous trajectory, and faster performance may take precedence over more accurate performance. In such scenarios, first-stage target prediction alone using the delay activity may be a valuable approach [15,16].

In the two-stage BMI, one stage compensated for the inaccuracies of the other and vice versa. Since our model relies on neural activity to estimate the movement, we cannot test directly to what extent the brain uses these two aspects of motor control to execute a movement. It is interesting to speculate, however, that similar to findings made in these experiments, the premotor cortex may use information on intended target location to correct for discrepancies in ongoing movement. Similarly, it may use information about ongoing movement to fine-tune

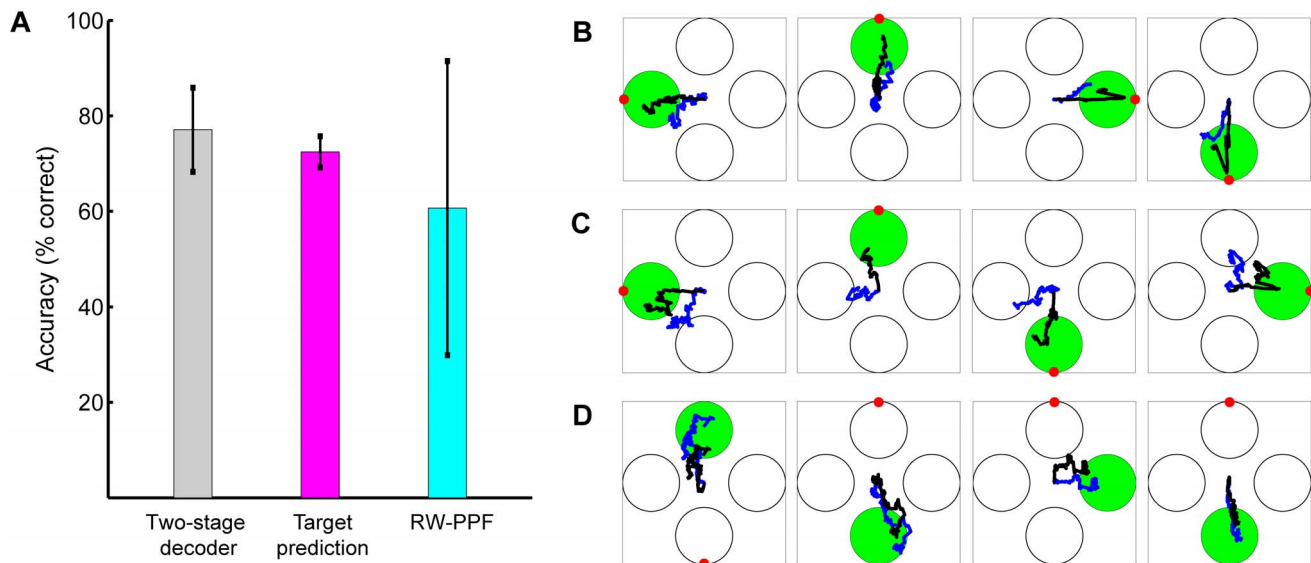


Figure 5. Real-time trajectories and the complementary property of the two-stage BMI. (A) Comparison of the real-time BMI accuracy with the real-time target prediction accuracy from the first stage and also with the accuracy of RW-PPF (i.e., the second stage without using the target prediction) obtained offline using the same real-time data set. The bars show mean quantities and the error bars show the standard deviation (s.d.) around the mean across sessions. The two-stage BMI outperforms either stage alone. (B–C) Sample decoded trajectories. In all panels, the green circle shows the instructed target, the red circle shows the predicted target from the first stage, the black line shows the trajectory estimate of the two-stage BMI, and the blue line shows that of RW-PPF using only the peri-movement activity. Sample trials where both the two-stage BMI and RW-PPF correctly acquire the target are shown in B. The complementary property of the two-stage BMI is illustrated in C and D. C shows sample trials in which the correct target prediction of the first stage compensates for the inaccurate estimation of the kinematic decoder in the second stage. D shows sample trials in which the kinematic decoder in the second stage, using ongoing peri-movement activity, compensates for the incorrect target prediction of the first stage.

doi:10.1371/journal.pone.0059049.g005

differences between initially intended target location and target location during movement itself. Hence, in addition to enabling the design of more accurate decoding algorithms, the present optimal feedback control design may provide insight into the control aspects of the motor function in natural settings. Therefore the two-stage optimal feedback control based BMI could provide an important and unique new step in developing neuroprosthetics that take advantage of the multiple types of movement information.

Materials and Methods

Ethics Statement

This study was performed in strict accordance with the recommendations in the Guide for the Care and Use of Laboratory Animals of the National Institutes of Health, and under regulation of the Subcommittee on Research Animal Care at Harvard Medical School. The protocol was approved by the Institutional Animal Care and Use Committee for Massachusetts General Hospital (animal welfare assurance number: A3596-01). All surgery was performed under inhalational anesthesia in combination with Ketamine and opiate analgesia, and every effort was made to minimize suffering, in accordance with the recommendations of the Weatherall report, “The use of non-human primates in research”. Animals were housed in their cages in the primate animal facility. Animals were fed regularly with regulated fluid administration and were given environmental enrichment. Animals were completely unrestrained within their home cages. At the conclusion of experiments one animal was euthanized to allow for histological verification of recording sites with Ketamine and Pentobarbital, in conformance with recom-

mendations of the American Veterinary Medical Association (AVMA) Guidelines on Euthanasia.

Behavioral Task

We used two adult male rhesus monkeys (*macaca mulatta*) in the study. During the tasks, the animals were seated in a primate chair (Crist Instrument Co Ltd, Damascus, MD). The primates’ head were restrained using a head post, and a spout was placed in front of their mouth to deliver juice using an automated solenoid. A spring-loaded, two-degrees of freedom manipulandum was mounted anterior to chair on the side contralateral to recording. A computer monitor was placed in front of the animals at eye level which displayed the task. A NI DAQ card (National Instruments, TX) was used for the I/O behavioral interface, and the behavioral program was run in Matlab (MathWorks, MA) using custom made software (www.monkeylogic.net).

Primates performed a center-out visually-instructed motor directional task that penalized touching the incorrect targets. The monkeys held the joystick contralateral to the site of recordings and could move their limb freely in the horizontal and vertical dimensions during the task. A computer monitor displayed the target locations and a cursor was used to represent the position of the joystick handle. Each individual trial began with the presentation of a central fixation point surrounded by four gray circular targets. Once the animals held the cursor within a central radius for a delay of 500 ms, one of the four randomly selected targets would turn green. After another 1000 ms, the fixation point would change color (“go” cue), at which time the monkeys could use the joystick to move the cursor from the center of the screen to the instructed target. Once the cursor reached the target, the animal received a drop of juice following a 320 ms delay if the correct target was reached and no incorrect targets

were touched by the cursor. If during a trial, the animals moved prematurely, failed to reach any target during the allowed time or touched an incorrect target before reaching the correct target, the trial aborted. Note that all targets had the same size as illustrated in Figures 3, 4, and 5. Once reward was delivered, another 1000 ms would lapse, the targets would erase, and the sequence would repeat again. The animals were required to return the spring-loaded joystick to the center fixation point before a new trial began.

Neurophysiologic Recordings and BMI Setup

A titanium head post and recording electrodes were surgically implanted in each monkey contralateral to the side of joystick use. All procedures were performed in an IACUC-approved aseptic primate surgical facility. Prior to electrode implantation, craniotomies were performed over the sites of interest using standard stereotactic coordinates. Once the cortex was exposed and the sulcal anatomy identified, several silicone microelectrode arrays were placed in the cortex (Neuronexus technologies, MI) into the PMd and supplementary motor areas. The electrodes were secured into place using fibrin glue, silicone sealant, and methylmethacrylate. The distal leads were then attached to a female connector and secured to the skull with titanium miniplates and dental acrylic. Anatomic post-mortem confirmation of electrode positioning was performed in one monkey. The second monkey is still performing behavioral tasks.

Recordings began at two weeks following surgical recovery. A Plexon multichannel acquisition processor was used to amplify and band-pass filter the neuronal signals (150 Hz–8 kHz; Plexon Inc., TX). Shielded cabling carried the signals from the electrode array to a set of six 16-channel amplifiers. Signals were then digitized at 40 kHz and processed to extract action potentials in real time by the Plexon workstation. Classification of the action potential waveforms were accomplished using dual-window discrimination and principle component analysis. Units with stable, identifiable waveform shapes and adequate refractory periods determined by autocorrelation were then used for the real-time experiments. Joystick position was sampled and recorded at 1 kHz. Neuronal data obtained from the Plexon workstation, in the form of action potential time stamps and channel, were then transmitted to a second PC computer running Matlab (Mathworks, MA) in real time. On decoder training sessions, the primates would use the joystick to move a cursor on the screen to one of four randomly selected targets over multiple trials. On decoder BMI sessions, the monkeys would still be allowed to use the joystick but the cursor image displayed on the screen would be supplied by the Matlab real-time decoder (Figure 1A). The cursor was initially placed at the center fixation point at the beginning of each trial. Here, estimated cursor movements would be relayed through a DAQ I/O (National Instruments, TX) to a third PC computer running the behavioral task. The computer would then display the estimated cursor position.

Chance Level Accuracy

Since the task requires reaching the correct target and since *first* touching *any* of the other (incorrect) targets results in an error, at best any chance-based decoder would *first* reach the correct target with equal probability among the four possible. Hence the performance of such a chance-based decoder is at most 25%. We also examined the performance of a linear regression decoder on shuffled neural activity and confirmed that it approximately resulted in 25% accuracy. To do so we shuffled the calculated firing rates for each neuron across time and trials, keeping its average firing rate the same.

Target Decoding

The BMI decodes the monkeys' intended target of movement by recording the ensemble spiking activity during the 800 ms delay interval prior to the “go” cue. Note that the delay between the start of target presentation and “go” cue is 1000 ms. We do not use the activity in the first 200 ms in the BMI in order to allow sufficient time for the visual target information to reach the PMd and SMA [16]. Using offline cross-validation analyses we observed that discarding this activity improves the prediction accuracy. Spiking activity of each neuron during this delay interval is modeled as a homogeneous Poisson process (a point process with constant rate) whose firing rate is a function of the intended target, fitted using the GLM framework. A maximum-likelihood (ML) decoder first calculates the likelihood probability of this ensemble activity for each possible target, G , and then selects the target with the highest likelihood as its prediction. Denoting the neural point process observations of the ensemble of C neurons by $\mathbf{N}_1, \dots, \mathbf{N}_t$ where $\mathbf{N}_t = (N_t^1, \dots, N_t^C)$ is the binary spike events of the C neurons at time t , and assuming that the neurons are conditionally independent given the target, the point process likelihood model for the ensemble is given by [60]

$$p(\mathbf{N}_t|G) = \prod_{c=1}^C (\lambda_c(G)\Delta)^{N_t^c} e^{-\lambda_c(G)\Delta} \quad (1)$$

where $\Delta = 5$ ms is the time increment used for binning the spikes and $\lambda_c(G)$ is the modeled firing rate of the c th neuron during the delay period for target G . The ML decoder then predicts the target as the one maximizing the ensemble likelihood,

$$\begin{aligned} \hat{G} &= \arg \max_{G=1,\dots,4} p(\mathbf{N}_1, \dots, \mathbf{N}_{T_d/\Delta}|G) \\ &= \arg \max_G \prod_{c=1}^C \prod_{t=1}^{T_d/\Delta} (\lambda_c(G)\Delta)^{N_t^c} e^{-\lambda_c(G)\Delta} \end{aligned} \quad (2)$$

Here $T_d = 800$ ms is the delay period. During the delay period the cursor was held at the center of the screen.

Kinematic Decoding

In the second stage, the BMI combines the decoded target with the peri-movement ensemble spiking activity using a recursive Bayesian decoder with an optimal feedback control design. A recursive Bayesian decoder in general consists of two probabilistic models: the prior model on the time sequence of kinematic states, and the observation model relating the neural signal to these states. The prior model should in general incorporate any prior information available about the kinematic states, which for a goal-directed movement includes the intended target. We build the prior movement model of the decoder using an optimal feedback control design, which takes into account the decoded target location and the sensory feedback. To develop the observation model, we use a point process model of the spiking activity whose instantaneous rate is a log-linear function of kinematics. The resulting decoder hence processes the spikes directly in real time and operates at the millisecond time scale of the spiking activity. In the next few sections we present the prior movement model and the neural observation model used in the decoder for estimation of goal-directed movements, and the decoder recursions.

Goal-Directed Movement Model: an Optimal Feedback Control Design

Previous offline work have built goal-directed prior models for reaching movements by conditioning a linear Gaussian state-space model, also known as a random-walk model, on being at the target at a *known* arrival time [46,67] or using a linear feedforward controlled (i.e., not taking into account the sensory feedback) model again assuming a known arrival time [47]. Alternatively, goal-directed prior models have been built by using a training data set, for example fitting a linear Gaussian state-space model for a given target to empirical reaches to its location [51] or fitting a single model for arbitrary targets based on a data set of reaches to their locations [11].

Since the goal of the decoder is to estimate the intended movement of the subject, a prior movement model that aims to emulate the sensorimotor processing underlying motor control could result in more accurate estimation of movement. Hence we build the prior goal-directed state-space model for the movement kinematics based on the optimal feedback control theory of the sensorimotor system [44,45]. This theory has been successfully used to interpret the sensorimotor function. For example, it has been shown that this theory can predict the bell-shaped velocity profiles and straight line trajectories observed in reaching movements [56,57]. In this optimal feedback control framework, each task is performed to accomplish a goal during which there is sensory feedback about the state of the system. Based on the intended goal, the system's forward dynamics model, the sensory feedback (for example vision and proprioception) about the current state of the system, and the desired time to accomplish the goal, the subject (controller) decides on the next plan of action or control command (for example muscle activation) and can hence make real-time adjustments based on the feedback to improve behavior. Specifying an approximate forward dynamics model and quantifying the task goals as cost functions and also the sensory feedback, this framework can predict the next plan of action in the presence of model uncertainty and sensory noise.

The difference in applying this framework to natural arm movement [44,45] and movement using a BMI is that in the latter case the system to be controlled is the BMI (Figure 1B). While performing the BMI task, the monkey decides on the next control command based on the visual feedback of the cursor position and the intended target. Similar to natural movement, in movement using a BMI, the next plan of action is in turn reflected in the neural activity but this time controls the system or BMI through the decoder (as we will develop) as opposed to directly controlling the arm (Figure 1B). Hence the BMI can be modeled in this optimal feedback control framework.

Motivated by this view, we develop a prior feedback-controlled state-space model for the kinematics that exploits the information about the target location and takes into account the sensory feedback. We introduced this model in [49,50] in a simulation study. Based on this model, the decoder can predict the monkey's next plan of action or control command and consequently the next kinematic state. Note that our prior model does not rely on a training data set, as is the case in [11,51], and can therefore easily extend to different target locations without requiring a set of empirical reaches to these locations. Finally, by using the optimal feedback control formulation, it could generalize to tasks other than reaching movements, if desired, by simply quantifying the goals of such tasks as the cost function in this formulation. We now present the construction of this model.

Denoting the sequence of kinematic states by $\mathbf{x}_0, \dots, \mathbf{x}_t$, we assume they are generated according to the linear dynamical system,

$$\mathbf{x}_{t+1} = \mathbf{A}\mathbf{x}_t + \mathbf{B}\mathbf{u}_t + \mathbf{w}_t. \quad (3)$$

This is the forward dynamics model. Here, \mathbf{u}_t is the control signal at time t , which is decided by the controller (the primate in the BMI context), \mathbf{w}_t is the zero-mean white Gaussian state noise with covariance matrix \mathbf{W} , and \mathbf{A} and \mathbf{B} are parameters of the forward model. Here we assume that the sensory feedback \mathbf{y}_t , is noiseless and $\mathbf{y}_t = \mathbf{x}_t$. This means that we assume the monkey has perfect sensory feedback of the cursor position on the screen. We also implicitly assume that the brain has acquired an internal representation of this forward model, i.e., has formed an internal forward model of the dynamics of movement in response to control commands \mathbf{u}_t in the task [59]. To find \mathbf{u}_t in the control framework, we need to specify a cost function that will then be minimized by optimizing over \mathbf{u}_t . The cost function in a given task should quantify its goal. For the above linear Gaussian dynamics, if we pick the cost function as a quadratic function of the state and control variables, i.e.,

$$J = \sum_{t=1}^{T-1} (\mathbf{x}_t' \mathbf{Q}_t \mathbf{x}_t + \mathbf{u}_t' \mathbf{R} \mathbf{u}_t) + \mathbf{x}'_T \mathbf{Q}_T \mathbf{x}_T, \quad (4)$$

where T is the movement duration, \mathbf{Q}_t is positive semidefinite and \mathbf{R} is positive definite, then the optimal control signal at any time, \mathbf{u}_t , is simply a linear *feedback* of the state at that time [68], i.e.,

$$\mathbf{u}_t = -\mathbf{L}_t(T)\mathbf{x}_t, \quad (5)$$

where \mathbf{L}_t can be found recursively and offline [68]. This is the well-known linear quadratic Gaussian (LQG) solution. Note that we assumed the monkey has perfect visual feedback of the cursor state, \mathbf{x}_t , in BMI control. Hence this optimal feedback control policy can be interpreted as the monkey's corrective control command in response to the visual feedback of the BMI cursor state that may deviate from the intended trajectory.

Substituting (5) in (3) reduces this state-space model to the optimal feedback-controlled state-space model

$$\mathbf{x}_{t+1} = (\mathbf{A} - \mathbf{B}\mathbf{L}_t(T))\mathbf{x}_t + \mathbf{w}_t, \quad (6)$$

which can now be used as the prior model to make prediction on the kinematic states. Note that \mathbf{Q}_t and \mathbf{R} should be appropriately designed for an application of interest and $\mathbf{L}_t(T)$ is time-varying and a function of the duration T . Note also that the sensory feedback is incorporated in the control term since the control term is simply a linear function of the current kinematic state (see (5)), assumed to be known through the feedback. This is in turn reflected in the prior model in (6).

We can now specialize these to the reaching movements used in our experiments. For a reaching movement the cost function should enforce end-point positional accuracy, stopping condition, and energetic efficiency [56,57]. Denoting the desired target position by \mathbf{d}^* and taking the state to be $\mathbf{x}_t = [\mathbf{d}_t, \mathbf{v}_t, \mathbf{a}_t]$ where the components represent position, velocity and force in the two dimensions respectively, similar to previous studies [56,57] we take this cost function to be the weighted sum

$$J = \|\mathbf{d}_T - \mathbf{d}^*\|^2 + w_v \|\mathbf{v}_T\|^2 + w_a \|\mathbf{a}_T\|^2 + w_r \sum_{t=1}^{T-1} \|\mathbf{u}_t\|^2 \quad (7)$$

where the weights are chosen to penalize the terms in the cost function approximately equally on average [56,57]. We adapt the following first order lowpass muscle-like system [57] for the dynamical system in (3) in each dimension,

$$\begin{bmatrix} \mathbf{d}_{t+1} \\ \mathbf{v}_{t+1} \\ \mathbf{a}_{t+1} \end{bmatrix} = \begin{bmatrix} 1 & \Delta & 0 \\ 0 & 1 - \frac{b\Delta}{m} & \frac{\Delta}{m} \\ 0 & 0 & 1 - \frac{\Delta}{\tau} \end{bmatrix} \begin{bmatrix} \mathbf{d}_t \\ \mathbf{v}_t \\ \mathbf{a}_t \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \frac{\Delta}{\tau} \end{bmatrix} \mathbf{u}_t + \begin{bmatrix} 0 \\ 0 \\ \mathbf{w}_t \end{bmatrix} \quad (8)$$

where the parameters $b = 10$ Ns/m, $\tau = 0.05$ s, and $m = 1$ kg come from biomechanics [57]. Note that we again assume that the monkey has acquired an internal forward model of the dynamics of movement in the task consistent with the above forward model [59] and uses this internal forward model to decide on the next control command after switching from manual control to biomimetic BMI control. The noise term in the forward model \mathbf{w}_t in turn captures the uncertainty in the forward model. In general, we can allow for this internal forward model to change once in BMI mode especially if a non-biomimetic decoder is used in combination with learning. Such considerations, however, are outside the scope of this paper.

Having specified the forward dynamics model and the cost function for the reaching movement, the feedback matrices $\mathbf{L}_t(T)$ can now be easily precomputed offline from the recursive solution of LQG [68] and stored for real-time use. Having these matrices, we can predict the monkey's next plan of action reflected in the control signal, \mathbf{u}_t , using (5) and assuming perfect sensory feedback about the current state of the cursor on the screen, \mathbf{x}_t . Note that in our two-stage decoder \mathbf{d}^* is determined from the decoded target location in the first stage (for more algorithmic details see [49,50,58]).

Observation Model

We build the observation model for each neuron as a point process whose instantaneous firing rate is a function of kinematics [60], i.e.,

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

We used a modified version of the cosine tuning model [23,26] for the instantaneous firing rate, modeling it as a log-linear function of position and velocity in the two dimensions [60], i.e.,

$$\lambda_c(t | \mathbf{x}_t) = \exp(\alpha_0^c + \alpha_c' \mathbf{x}_t), \quad (9)$$

where \mathbf{x}_t denotes these kinematic states at time t and α_0^c and α_c are fitted using the GLM framework [60] on the peri-movement spiking activity. More specifically, denoting the model parameters for neuron c by $\phi_c = [\alpha_0^c, \alpha_c]$, the GLM framework finds the maximum likelihood estimate

$$\hat{\phi}_c = \arg \max_{\phi_c} p(N_{1:T_t}^c | \mathbf{x}_{1:T_t}; \phi_c)$$

where $N_{1:T_t}^c$ is the peri-movement spiking activity of the neuron during training and $\mathbf{x}_{1:T_t}$ are the corresponding kinematic states. Using the GLM framework, P -values can also be obtained for all the model parameters [60] (for example using the glmfit function

in Matlab) and hence the tuning properties of the neurons can be examined.

We assumed that the spiking activity of the neurons are conditionally independent given the kinematic states and hence the observation model for the ensemble is given by

$$p(\mathbf{N}_t | \mathbf{x}_t) = \prod_{c=1}^C (\lambda_c(t | \mathbf{x}_t) \Delta)^{N_t^c} e^{-\lambda_c(t | \mathbf{x}_t) \Delta} \quad (10)$$

Uncertainty in the Movement Duration

Having the prior and the observation models we can now develop the recursions for the Bayesian decoder. However, the prior model built in (6) is dependent on the movement duration, T , which is not known to the decoder. In other words, unlike natural movement in which the monkey (controller) decides on the movement duration, in movement using a BMI the decoder does not have a priori knowledge of this duration. This is typically the case for goal-directed state-space models as there is much more constraint on the movement kinematics close to the arrival time at the target compared to far from it since in the former case the trajectory soon needs to reach the intended target [49,50,58]. Hence we develop the BMI decoder to jointly resolve this duration uncertainty and estimate the trajectory purely based on the neural spiking activity. We first present the recursions of a feedback-controlled point-process filter assuming a known movement duration and then show how we can resolve the duration uncertainty inherent to the prior model.

Feedback-Controlled Point Process Filter (FC-PPF) for a Known Movement Duration

For now we assume that the movement duration is known. The minimum mean-square error (MMSE) estimator is given by the mean of the posterior density that is $p(\mathbf{x}_t | \mathbf{N}_{1:t}, T)$ for a given duration T . Denoting the one step prediction mean by $\mathbf{x}_{t|t-1, T} = E(\mathbf{x}_t | \mathbf{N}_{1:t-1}, T)$, its covariance matrix by $\mathbf{W}_{t|t-1, T}$, the MMSE estimate by $\hat{\mathbf{x}}_{t|t, T}$, and finally its covariance matrix by $\mathbf{W}_{t|t, T}$, $\hat{\mathbf{x}}_{t|t, T}$ is found from the following recursions

$$\mathbf{x}_{t|t-1, T} = (\mathbf{A} - \mathbf{B}\mathbf{L}_t(T))\mathbf{x}_{t-1|t-1, T}. \quad (11)$$

$$\mathbf{W}_{t|t-1, T} = (\mathbf{A} - \mathbf{B}\mathbf{L}_t(T))\mathbf{W}_{t-1|t-1, T}(\mathbf{A} - \mathbf{B}\mathbf{L}_t(T))' + \mathbf{W}_t \quad (12)$$

$$\begin{aligned} \mathbf{W}_{t|t, T}^{-1} &= \mathbf{W}_{t|t-1, T}^{-1} + \\ &\sum_{c=1}^C \left[\left(\frac{\partial \log \lambda_c(t | \mathbf{x}_t)}{\partial \mathbf{x}_t} \right)' \left(\frac{\partial \log \lambda_c(t | \mathbf{x}_t)}{\partial \mathbf{x}_t} \right) \lambda_c(t | \mathbf{x}_t) \Delta \right. \\ &\left. - (N_t^c - \lambda_c(t | \mathbf{x}_t) \Delta) \frac{\partial^2 \log \lambda_c(t | \mathbf{x}_t)}{\partial \mathbf{x}_t \partial \mathbf{x}_t'} \right]_{\mathbf{x}_{t|t-1, T}} \end{aligned} \quad (13)$$

$$\begin{aligned} \hat{\mathbf{x}}_{t|t, T} &= \mathbf{x}_{t|t-1, T} \\ &+ \mathbf{W}_{t|t, T} \sum_{c=1}^C \left[\left(\frac{\partial \log \lambda_c(t | \mathbf{x}_t)}{\partial \mathbf{x}_t} \right)' (N_t^c - \lambda_c(t | \mathbf{x}_t) \Delta) \right]_{\mathbf{x}_{t|t-1, T}} \end{aligned} \quad (14)$$

where $[\cdot]_{\mathbf{x}_{t|t-1,T}}$ denotes the evaluation of the expression at $\mathbf{x}_{t|t-1,T}$. These recursions are obtained using (6) for the prediction step and a Gaussian approximation for the update step as previously derived [49,50,69]. For the rate model in (9) since $\frac{\partial \log \lambda_c}{\partial \mathbf{x}_t} = \alpha'_c$ and $\frac{\partial^2 \log \lambda_c}{\partial \mathbf{x}_t \partial \mathbf{x}_t'} = \mathbf{0}$ these recursions simplify to

$$\mathbf{x}_{t|t-1,T} = (\mathbf{A} - \mathbf{B}\mathbf{L}_t(T))\mathbf{x}_{t-1|t-1,T} \quad (15)$$

$$\mathbf{W}_{t|t-1,T} = (\mathbf{A} - \mathbf{B}\mathbf{L}_t(T))\mathbf{W}_{t-1|t-1,T}(\mathbf{A} - \mathbf{B}\mathbf{L}_t(T))' + \mathbf{W}_t \quad (16)$$

$$\mathbf{W}_{t|t,T}^{-1} = \mathbf{W}_{t|t-1,T}^{-1} + \sum_{c=1}^C \alpha_c \alpha'_c \lambda_c(t|\mathbf{x}_{t|t-1,T}) \Delta \quad (17)$$

$$\mathbf{x}_{t|t,T} = \mathbf{x}_{t|t-1,T} + \mathbf{W}_{t|t,T} \sum_{c=1}^C \alpha_c (N_t^c - \lambda_c(t|\mathbf{x}_{t|t-1,T}) \Delta) \quad (18)$$

To provide some insight into these recursions, note that in the prediction step given in (15) the feedback-controlled prior model is used to move the estimate forward. In the update step given in (18) the estimate is found by making a correction or update to this prediction. Here, $\lambda_c(t|\mathbf{x}_{t|t-1,T})\Delta$ is the predicted probability of having a spike in the time interval Δ and hence the correction is (1-predicted probability of a spike) if a spike occurs and (0- predicted probability of a spike) if no spike occurs. Hence if a spike occurs and the predicted probability of a spike is high this correction is small and vice versa. Therefore the estimate is a combination of the prediction and the correction terms. The more informative the spiking activity is about the state (determined through α_c), the more weight is placed on the correction term and vice versa. If the spiking activity is not informative at all or is not used, then the estimate will just be the prediction, which is obtained only using the feedback-controlled state model and ignoring the observation model. In this case and given a movement duration, the prediction step will generate a straight line from the center to the predicted target location according to the prior model.

Resolving the Duration Uncertainty: Feedback-Controlled Parallel Point Process Filter (FC-P-PPF)

The feedback-controlled state-space model in (6) (and many other goal-directed state-space models) is a function of movement duration, T , not known a priori to the real-time BMI [49,50,58]. Hence any goal-directed real-time decoder needs to resolve this duration uncertainty. We introduced a framework to resolve this duration uncertainty in [49] by discretizing the movement duration, finding the kinematic estimate for each discretized duration, and then optimally combining these kinematic estimates based on the neural data. A similar approach using a discretized set of durations was subsequently used in a simulation study in [70] to resolve the duration uncertainty of the prior model developed in [46] for estimation of simulated trajectories. Our framework for resolving the duration uncertainty is based on mixture modeling, a common approach in statistical inference that is used to estimate a desired density in different applications. For example, mixture modeling combined with sequential state estimation in dynamical systems, when the system is operating under different or changing

regimes of operation, has been used as early as in [71]. See also the mixture Kalman filtering work in [72] and references therein. For decoding the kinematics from neural activity, mixture modeling was first used in [51] and successfully applied to combine empirically fitted and time-invariant state models for reaching movements to different targets in an offline study. Here, we use mixture modeling to combine feedback-controlled prior models of different durations and hence resolve the duration uncertainty inherent to this prior model.

Our framework works by discretizing the duration, finding the kinematic estimate for each discretized duration using an FC-PPF, running these FC-PPF filters in parallel, and finding the likelihood of each of the discretized duration points jointly with the corresponding trajectory estimate, all purely based on the neural spiking activity and in real time. The decoder then optimally combines the trajectory estimates corresponding to the discretized durations to get the overall trajectory estimate. The result is the feedback-controlled parallel point process filter (FC-P-PPF) that we have derived in detail in [49,50,58]. Denoting the overall MMSE estimator by $\mathbf{x}_{t|t}$, it is given by the mean of the posterior density, which using the law of total probability is expanded as,

$$\mathbf{x}_{t|t} = E(\mathbf{x}_t | \mathbf{N}_{1:t}) = \sum_{j=1}^J p(T_j | \mathbf{N}_{1:t}) \mathbf{x}_{t|t,T_j} \quad (19)$$

where T_1, \dots, T_J are the J discretization points for T , $\mathbf{x}_{t|t,T_j}$ is the estimate given that the duration is T_j and found from the recursions in (15) to (18), $p(T_j | \mathbf{N}_{1:t})$ is the likelihood of the corresponding duration given the peri-movement neural activity, and finally the summation is over all j for which $T_j > t$. The likelihood of the corresponding duration, $p(T_j | \mathbf{N}_{1:t})$, can be computed as derived in our work in [49,50,58] and is only a function of the prediction and posterior means and covariances found in (15) to (18) and the parameters of the observation model. Here we provide the final expressions for readers' convenience. These are given by

$$p(T_j | \mathbf{N}_{1:t}) \propto p(\mathbf{N}_{1:t} | T_j) \quad (20)$$

with

$$p(\mathbf{N}_{1:t} | T_j) = \prod_{i=1}^t p(\mathbf{N}_i | \mathbf{N}_{1:i-1}, T_j) = \prod_{i=1}^t g(\mathbf{N}_i | T_j) \quad (21)$$

Using a Gaussian approximation to the posterior, the term $g(\mathbf{N}_i | T_j)$ as we derive in [49,50,58] is given by

$$\begin{aligned} g(\mathbf{N}_i | T_j) &= \sqrt{\frac{|\mathbf{W}_{i|T_j}|}{|\mathbf{W}_{i|i-1,T_j}|}} p(\mathbf{N}_i | \mathbf{x}_{i|T_j}, \mathbf{N}_{1:i-1}) \\ &\times \exp \left[-\frac{1}{2} (\mathbf{x}_{i|T_j} - \mathbf{x}_{i|i-1,T_j})' \mathbf{W}_{i|i-1,T_j}^{-1} (\mathbf{x}_{i|T_j} - \mathbf{x}_{i|i-1,T_j}) \right] \\ &= \sqrt{\frac{|\mathbf{W}_{i|T_j}|}{|\mathbf{W}_{i|i-1,T_j}|}} \prod_c \left(\lambda_c(i|\mathbf{x}_{i|T_j}) \Delta \right)^{N_i^c} e^{-\lambda_c(i|\mathbf{x}_{i|T_j}) \Delta} \\ &\times \exp \left[-\frac{1}{2} (\mathbf{x}_{i|T_j} - \mathbf{x}_{i|i-1,T_j})' \mathbf{W}_{i|i-1,T_j}^{-1} (\mathbf{x}_{i|T_j} - \mathbf{x}_{i|i-1,T_j}) \right] \end{aligned} \quad (22)$$

for $j=1, \dots, J$ where all the quantities are known. Note that

$\sum_{j=1}^J p(T_j | \mathbf{N}_{1:t}) = 1$ and hence we can compute all the duration likelihoods. Finally, combining in (19), we get the final estimate. Here, we coarsely discretized the duration of a trial.

Random-Walk Point Process Filter (RW-PPF)

Setting $\mathbf{B} = \mathbf{0}$ reduces the state-space model in (3) to the random-walk model and consequently the recursions of FC-PPF with this choice recover the RW-PPF.

Ridge Regression

We also find the performance of the linear ridge regression decoder on the neural data. The commonly used regression decoder in the BMI literature is the ordinary linear least squares regression decoder that is fit by minimizing the residual squared error. The linear least squares regression, however, can result in high mean-square estimation error due to a high error variance, for example in cases with correlated regressors [61]. An alternative to the ordinary linear least squares regression is the linear ridge regression that can result in lower mean-square error by reducing the error variance at the price of some increase in bias [61]. Hence ridge regression has been used in a number of BMI studies [11,19]. In ridge regression, the position variable at a given time is reconstructed as a linear combination of the history of the standardized firing rates of the ensemble of C neurons over a selected number of time bins prior to and including that time. In ridge regression, the regression coefficients are found by minimizing the residual squared error plus a regularization term that penalizes large coefficient estimates [11,61]. Denoting the total number of time samples in the training data set by U , the number of history coefficients in the regression model by H , the $U \times HC$ dimensional standardized firing rate matrix with \mathbf{R} , the mean subtracted position variable by $\tilde{\mathbf{d}}$, and the regression coefficients by γ , the ridge regression coefficients are given by [11,61].

$$\hat{\gamma} = (\mathbf{R}'\mathbf{R} + \delta\mathbf{I})^{-1}\mathbf{R}'\tilde{\mathbf{d}} \quad (23)$$

where δ is the regularization parameter. The special case of $\delta = 0$ gives the ordinary least squares regression solution. We selected δ by finding the mean-square error for a wide range of regularization parameters and selecting the regularization parameter that minimized the mean-square error using leave-one-out cross-validation. We found the performance using multiple choices for the length of the history window used by the ridge regression. Specifically, we found the performance of the ridge regression decoder using 200 ms, 400 ms, 600 ms, and 800 ms of history coefficients and selected the number of history coefficients that minimized the mean-square error using leave-one-out cross-validation.

Possible Extensions

In our implementation of the BMI, the first stage of the decoder makes a decision about the intended target. However, instead, the BMI can easily include all targets in the FC-P-PPF but weight them properly by their corresponding likelihood calculated from the first stage similar to a previous offline study [51]. In our case, this means including more parallel filters in the FC-P-PPF for the different targets (and their discretized durations). This will consequently increase the complexity of the decoder. We chose not to implement this extension since, using offline analysis, we observed that it resulted in little improvement at the price of four times the complexity. The absence of a significant improvement in this case was likely a result of the fact that in our experiments the

target-related activity during the delay period was strongly tuned to the targets and that overall the peri-movement activity was not as strongly tuned to the task. However, in cases where such target-related activity is not strong, this extension will potentially result in further improvement in the second stage as it allows for its higher weighting compared to the first stage. Also applying such an extension could potentially allow the FC-P-PPF decoding algorithm to be applied in situations where no delay period activity and hence no target information from the first stage is available. In such a case, equal weights would be initially assigned to the filters corresponding to each possible target in the FC-P-PPF, and these weights would then be updated during movement based on the peri-movement activity.

Even though in the present experiments we used four targets, the decoder can generalize to the case with more targets. This is because the prior model in (6) can be generalized to arbitrary target locations by just replacing \mathbf{d}^* accordingly in the cost function in (7). Given the present results, it is conceivable that a similar complementary performance could be observed for the two-stage BMI in the case where more targets are present, especially given that in this case it will be harder for either stage alone to result in accurate performance (for example the first stage needs to decode one out of more targets). Investigating the behavior of the decoder in experimental setups with more target locations will be a valuable future research direction.

Finally, in our work we use the target onset and “go” cues to indicate the boundaries between baseline, plan, and movement epochs as is typically done in BMI experiments [73]. For a BMI to be truly autonomous, however, such epochs should also be detected based on the neural activity (see, e.g., [73]). Hence extending our approach to also detect the movement epoch based on the neural activity will be a future research direction.

Number of Neurons Required for Accurate Target Prediction

To find the number of neurons that were sufficient to obtain an accurate target prediction during the delay period, we performed a single neuron analysis in which the spiking activity of a single neuron was used to decode the target (Figure S1). We then sorted the neurons based on their single-neuron accuracies. From the sorted set, we selected different number of neurons and performed the decoding analysis for them. For example the decoding analysis for two neurons was done for the two neurons with the highest single neuron accuracies. Doing so, we found the target prediction accuracy as a function of the number of top cells included in the decoding. We found that on average across sessions only $17 \pm 6\%$ of the neural ensemble or 3.3 ± 1.0 neurons were sufficient to obtain a prediction accuracy higher than 90% of the ensemble accuracy.

Roughness Coefficient

The roughness coefficient for a sequence is an indicator of how smooth it is [64]. For a sequence d_1, \dots, d_T , it is defined as

$$r = \frac{\sum_{t=2}^T (d_t - d_{t-1})^2}{\sum_{t=2}^T (d_t - \bar{d})^2}$$

where \bar{d} denotes the mean of the sequence. It can also be generalized to the vector case by writing

$$r = \frac{\sum_{t=2}^T \|\mathbf{d}_t - \mathbf{d}_{t-1}\|^2}{\sum_{t=2}^T \|\mathbf{d}_t - \bar{\mathbf{d}}\|^2}$$

For a trajectory of duration T this provides a measure of smoothness where smaller coefficients correspond to smoother trajectories. In this case \mathbf{d}_t corresponds to the position vector at time t .

Supporting Information

Figure S1 Neuron dropping analysis. (A–D) Activity of a single neuron under the four targets in one training session. In the top figure, each row corresponds to a different trial and the black dots indicate the spike times. The bottom figure shows the corresponding firing rate. Activity is aligned to the target presentation time and vertical dashed lines indicate the target presentation and “go” cue times. The target prediction accuracy (leave-one-out cross-validation) of this neuron is 65%. (E) Target prediction accuracy as a function of the number of top neurons. The solid lines show the target prediction accuracy of six training sessions as a function of the number of neurons included in the prediction (The curve for each session is shown in a different color for clarity). Neurons were sorted based on their single neuron accuracy. Chance level accuracy for target classification is 25%. (TIF)

Figure S2 Offline SNR comparisons. The bars show mean quantities and the error bars show the standard deviation (s.d.) around the mean across sessions. SNR is obtained from the training sessions using leave-one-out cross-validation. (TIF)

References

- Chapin JK, Moxon KA, Markowitz RS, Nicolelis MAL (1999) Real-time control of a robot arm using simultaneously recorded neurons in the motor cortex. *Nat Neurosci* 2: 664–670.
- Wessberg J, Stambaugh CR, Kralik JD, Beck PD, Laubach M, et al. (2000) Real-time prediction of hand trajectory by ensembles of cortical neurons in primates. *Nature* 408: 361–365.
- Serruya MD, Hatsopoulos NG, Paninski L, Fellows MR, Donoghue JP (2002) Instant neural control of a movement signal. *Nature* 416: 141–142.
- Hochberg LR, Serruya MD, Friehs GM, Mukand JA, Saleh M, et al. (2006) Neuronal ensemble control of prosthetic devices by a human with tetraplegia. *Nature* 442: 164–171.
- Carmena JM, Lebedev MA, Crist RE, O’Doherty JE, Santucci DM, et al. (2003) Learning to control a brain-machine interface for reaching and grasping by primates. *PLoS Biol* 1: 193–208.
- Taylor DM, Tillery SIH, Schwartz AB (2002) Direct cortical control of 3D neuroprosthetic devices. *Science* 296: 1829–1832.
- Ganguly K, Carmena JM (2009) Emergence of a stable cortical map for neuroprosthetic control. *PLoS Biol* 7: 1–13.
- Wolpaw JR, McFarland DJ (2004) Control of a two-dimensional movement signal by a non-invasive brain-computer interface in humans. *Proc Natl Acad Sci USA* 101: 17849–17854.
- Velliste M, Perel S, Spalding MC, Whitford AS, Schwartz AB (2008) Cortical control of a prosthetic arm for self-feeding. *Nature* 453: 1098–1101.
- Moritz CT, Perlmutter SI, Fetz EE (2008) Direct control of paralysed muscles by cortical neurons. *Nature* 456: 639–643.
- Mulliken GH, Musallam S, Andersen RA (2008) Decoding trajectories from posterior parietal cortex ensembles. *J Neurosci* 28: 12913–12926.
- Kim SP, Simeral JD, Hochberg LR, Donoghue JP, Black MJ (2008) Neural control of computer cursor velocity by decoding motor cortical spiking activity in humans with tetraplegia. *J Neural Eng* 5: 455–476.
- Li Z, O’Doherty JE, Hanson TL, Lebedev MA, Henriquez CS, et al. (2009) Unscented Kalman filter for brain-machine interfaces. *PLoS ONE* 4: 1–18.
- Chase SM, Schwartz AB, Kass RE (2009) Bias, optimal linear estimation, and the differences between open-loop simulation and closed-loop performance of spiking-based brain-computer interface algorithms. *Neural Networks* 22: 1203–1213.
- Musallam S, Corneil BD, Greger B, Scherberger H, Andersen RA (2004) Cognitive control signals for neural prosthetics. *Science* 305: 258–262.
- Santhanam G, Ryu SI, Yu BM, Afshar A, Shenoy KV (2006) A high-performance brain-computer interface. *Nature* 442: 195–198.
- O’Doherty JE, Lebedev MA, Ifft PJ, Zhuang KZ, Shokur S, et al. (2011) Active tactile exploration using a brain-machine-brain interface. *Nature* 479: 228–231.
- DiGiovanna J, Mahmoudi B, Fortes J, Principe JC, Sanchez JC (2009) Coadaptive brain-machine interface via reinforcement learning. *IEEE Trans Biomed Eng* 56: 54–64.
- Hauschild M, Mulliken GH, Fineman I, Loeb GE, Andersen RA (2012) Cognitive signals for brain-machine interfaces in posterior parietal cortex include continuous 3D trajectory commands. *Proc Natl Acad Sci USA* 109: 17075–17080.
- Shanechi MM, Hu RC, Powers M, Wornell GW, Brown EN, et al. (2012) Neural population partitioning and a concurrent brain-machine interface for sequential motor function. *Nat Neurosci* 15: 1715–1722.
- Mussa-Ivaldi FA, Alford ST, Chiappalone M, Fadiga L, Karniel A, et al. (2010) New perspectives on the dialogue between brains and machines. *Front Neurosci* 4: 44–52.
- Pohlmeier EA, Miller LE, Mussa-Ivaldi FA, Perreault EJ, Solla SA (2003) Prediction of EMG from multiple electrode recordings in primary motor cortex. In: *Proc. IEEE EMBS*. 17–21.
- Georgopoulos A, Kalaska JF, Caminiti R, Massey JT (1982) On the relations between the direction of two-dimensional arm movement and cell discharge in primate motor cortex. *J Neurosci* 2: 1527–1537.
- Georgopoulos A, Schwartz AB, Kettner RE (1986) Neural population coding of movement direction. *Science* 233: 1416–1419.
- Ashe J, Georgopoulos AP (1994) Movement parameters and neural activity in motor cortex and area 5. *Cereb Cortex* 6: 590–600.
- Moran DW, Schwartz AB (1999) Motor cortical representation of speed and direction during reaching. *J Neurophysiol* 82: 2676–2692.
- Paninski L, Fellows MR, Hatsopoulos NG, Donoghue JP (2004) Spatiotemporal tuning of motor cortical neurons for hand position and velocity. *J Neurophysiol* 91: 515–532.

Figure S3 Offline accuracy of the two-stage decoder on the training sessions data. Accuracies of the two-stage decoder, the first stage target prediction, and the RW-PPF (i.e., the second stage without using the target prediction) on the training sessions joystick movements are shown. The bars show mean quantities and the error bars show the standard deviation around the mean across sessions. Comparisons of the RMS error and the smoothness of the decoded trajectories for the two-stage decoder and RW-PPF are given in Figure 2. Note that target prediction, unlike the two-stage, RW-PPF, and the linear regression decoders, does not generate an estimated trajectory. (TIF)

Figure S4 Comparison of the trajectory estimates. For completeness, we also found the accuracy of the ridge regression decoder on the same real-time BMI data set. It is important to note, however, that since the performance of the ridge regression here is found offline, it is likely lower from its performance if used in real time and practiced by the monkey. The black line shows the trajectory estimate of the real-time BMI. The blue line shows that of RW-PPF using only the peri-movement activity, and the red line shows that of a linear ridge regression decoder, both run offline but using the same real-time data set as the BMI. (A) Sample trials in which the ridge regression decoder is correct. (B) Sample trials in which the ridge regression decoder is incorrect. The average accuracy of the linear ridge regression decoder on the BMI data sets was $50 \pm 11\%$ (mean \pm s.d.; c.f. Figure 5A). (TIF)

Author Contributions

Developed the algorithms: MMS EB GW. Developed the BMI: MMS ZW. Conceived and designed the experiments: ZW. Performed the experiments: MMS ZW RH MP. Analyzed the data: MMS. Contributed reagents/materials/analysis tools: MMS ZW. Wrote the paper: MMS ZW EB GW.

28. Crammond DJ, Kalaska JF (1996) Differential relation of discharge in primary motor cortex and premotor cortex to movements versus actively maintained postures during a reaching task. *Exp Brain Res* 108: 45–61.
29. Caminiti R, Johnson PB, Galli C, Ferraina S, Burnod Y (1991) Making arm movements within different parts of space: The premotor and motor cortical representation of a coordinate system for reaching to visual targets. *J Neurosci* 11: 1182–1197.
30. Kurata K (1993) Premotor cortex of monkeys: Set- and movement-related activity recting amplitude and direction of wrist movements. *J Neurophysiol* 69: 187–200.
31. Mulliken GH, Musallam S, Andersen RA (2008) Forward estimation of movement state in posterior parietal cortex. *Proc Natl Acad Sci USA* 105: 8170–8177.
32. Averbeck BB, Chafee MV, Crowe DA, Georgopoulos AP (2005) Parietal representation of hand velocity in a copy task. *J Neurophysiol* 93: 508–518.
33. Snyder LH, Batista AP, Andersen RA (1997) Coding of intention in the posterior parietal cortex. *Nature* 386: 167–170.
34. Batista AP, Buneo CA, Snyder LH, Andersen RA (1999) Reach plans in eye-centered coordinates. *Science* 285: 257–260.
35. Buneo CA, Jarvis MR, Batista AP, Andersen RA (2002) Direct visuomotor transformations for reaching. *Nature* 416: 632–636.
36. Andersen RA, Burdick JW, Musallam S, Pesaran B, Cham JG (2004) Cognitive neural prosthetics. *Trends in Cognitive Sciences* 8: 486–493.
37. Messier J, Kalaska JF (2000) Covariation of primate dorsal premotor cell activity with direction and amplitude during a memorized-delay reaching task. *J Neurophysiol* 84: 152–165.
38. Crammond DJ, Kalaska JF (1994) Modulation of preparatory neuronal activity in dorsal premotor cortex due to stimulus-response compatibility. *J Neurophysiol* 71: 1281–1284.
39. Boussaoud D, Bremmer F (1999) Gaze effects in the cerebral cortex: reference frames for space coding and action. *Exp Brain Res* 128: 170–180.
40. Kakei S, Hoffman DS, Strick PL (2003) Sensorimotor transformations in cortical motor areas. *Neurosci Res* 46: 1–10.
41. Churchland MM, Santhanam G, Shenoy KV (2006) Preparatory activity in premotor and motor cortex reflects the speed of the upcoming reach. *J Neurophysiol* 96: 3130–3146.
42. Crammond DJ, Kalaska JF (2000) Prior information in motor and premotor cortex: Activity during the delay period and effect on pre-movement activity. *J Neurophysiol* 84: 986–1005.
43. Crutcher MD, Russo GS, Ye S, Backus DA (2004) Target-, limb-, and context-dependent neural activity in the cingulate and supplementary motor areas of the monkey. *Exp Brain Res* 158: 278–288.
44. Todorov E, Jordan MI (2002) Optimal feedback control as a theory of motor coordination. *Nat Neurosci* 5: 1226–1235.
45. Todorov E (2004) Optimality principles in sensorimotor control. *Nat Neurosci* : 907–915.
46. Srinivasan L, Eden UT, Willisky AS, Brown EN (2006) A state-space analysis for reconstruction of goal-directed movements using neural signals. *Neural Comput* 18: 2465–2494.
47. Kemere C, Meng T (2005) Optimal estimation of feed-forward-controlled linear systems. In: *Proc. IEEE international conference on acoustics, speech, and signal processing (ICASSP)*. 353–356.
48. Srinivasan L, Eden UT, Mitter SK, Brown EN (2007) General-purpose filter design for neural prosthetic devices. *J Neurophysiol* 98: 2456–2475.
49. Shanechi MM, Wornell GW, Williams ZM, Brown EN (2010) A parallel point-process filter for estimation of goal-directed movements from neural signals. In: *Proc. IEEE international conference on acoustics, speech, and signal processing (ICASSP)*. Dallas, TX, 521–524.
50. Shanechi MM, Wornell GW, Williams ZM, Brown EN (2012) Feedback-controlled parallel point process filter for estimation of goal-directed movements from neural signals. *IEEE Trans Neural Syst Rehabil Eng*, published online, doi: 10.1109/TNSRE.2012.2221743.
51. Yu BM, Kemere C, Santhanam G, Afshar A, Ryu SI, et al. (2007) Mixture of trajectory models for neural decoding of goal-directed movements. *J Neurophysiol* 97: 3763–3780.
52. Shanechi MM, Williams ZM, Wornell GW, Brown EN (2010) A real-time brain-machine interface combining plan and peri-movement activities. In: *Research in Encoding And Decoding of Neural Ensembles (AREADNE) Meeting*. Santorini, Greece.
53. Shanechi MM, Williams ZM, Wornell GW, Brown EN (2010) Combining plan and peri-movement activities improves the performance of brain-machine interfaces. In: *Society for Neuroscience (SFN) Meeting*. San Diego, USA.
54. Shanechi MM, Williams ZM, Wornell GW, Brown EN (2011) A brain-machine interface combining target and trajectory information using optimal feedback control. In: *Computational and Systems Neuroscience (COSYNE) Meeting*. Salt Lake City, USA.
55. Loeb GE, Levine WS, He J (1990) Understanding sensorimotor feedback through optimal control. *Cold Spring Harb Symp Quant Biol* 55: 791–803.
56. Todorov E (2005) Stochastic optimal control and estimation methods adapted to the noise characteristics of the sensorimotor system. *Neural Comput* 17: 1084–1108.
57. Liu D, Todorov E (2007) Evidence for the exible sensorimotor strategies predicted by optimal feedback control. *J Neurosci*: 9354–9368.
58. Shanechi MM (2011) *Real-Time Brain-Machine Interface Architectures: Neural Decoding from Plan to Movement*. Ph.D. thesis, Massachusetts Institute of Technology.
59. Shadmehr R, Krakauer JW (2008) A computational neuroanatomy for motor control. *Exp Brain Res*: 359–381.
60. Truccolo W, Eden UT, Fellows MR, Donoghue JP, Brown EN (2005) A point process framework for relating neural spiking activity to spiking history, neural ensemble, and extrinsic covariate effects. *J Neurophysiol* 93: 1074–1089.
61. Marquardt DW, Snee RD (1975) Ridge regression in practice. *Am Stat* 29: 3–20.
62. Kim SP, Sanchez JC, Rao YN, Erdogmus D, Carmena JM, et al. (2006) A comparison of optimal MIMO linear and nonlinear models for brain-machine interfaces. *J Neural Eng* 3: 145–161.
63. Sanchez J, Kim SP, Erdogmus D, Rao Y, Principe J, et al. (2002) Input-output mapping performance of linear and nonlinear models for estimating hand trajectories from cortical neuronal firing patterns. In: *Proceedings of the 12th IEEE Workshop on Neural Networks for Signal Processing*. 139–148.
64. Bloomfield P (2000) *Fourier analysis of time series*. Hoboken, NJ: Wiley.
65. Scott SH (2004) Optimal feedback control and the neural basis of volitional motor control. *Nat Rev Neurosci* 5: 532–546.
66. Cunningham JP, Nuyujukian P, Gilja V, Chestek CA, Ryu SI, et al. (2011) A closed-loop human simulator for investigating the role of feedback control in brain-machine interfaces. *J Neurophysiol* 105: 1932–1949.
67. Kulkarni JE, Paninski L (2008) State-space decoding of goal-directed movements. *IEEE Signal Process Mag* 25: 78–86.
68. Bertsekas D (2005) *Dynamic Programming and Optimal Control*. Belmont, MA: Athena Scientific.
69. Eden UT, Frank LM, Barbieri R, Solo V, Brown EN (2004) Dynamic analysis of neural encoding by point process adaptive filtering. *Neural Comput* 16: 971–998.
70. Srinivasan L, da Silva M (2011) Breaking the fixed-arrival-time restriction in reaching movements of neural prosthetic devices. *IEEE Trans Biomed Eng* 58: 1555–1564.
71. Ackerson GA, Fu KS (1970) On state estimation in switching environments. *IEEE Trans Autom Control* AC-15: 10–17.
72. Chen R, Liu JS (2000) Mixture Kalman filters. *J Roy Statist Soc B* 62: 493–508.
73. Kemere C, Santhanam G, Yu BM, Afshar A, Ryu SI, et al. (2008) Detecting neural-state transitions using hidden Markov models for motor cortical prostheses. *J Neurophysiol* 100: 2441–2452.