

Improving Neural Prosthetic System Performance by Combining Plan and Peri-Movement Activity

Byron M. Yu¹, Stephen I. Ryu^{1,2}, Gopal Santhanam¹, Mark M. Churchland¹, Krishna V. Shenoy^{1,3}

¹Department of Electrical Engineering, ²Department of Neurosurgery, ³Neurosciences Program, Stanford University, Stanford, CA 94305, USA

Abstract—While most neural prosthetic systems to date estimate arm movements based solely on the activity prior to reaching movements during a delay period (plan activity) or solely on the activity during reaching movements (peri-movement activity), we show that decode classification can be improved by 56% and 71% respectively by using both types of activity together. We recorded from the pre-motor cortex of a rhesus monkey performing a delayed-reach task to one of seven targets. We found that taking into account the time-varying structure in peri-movement activity further improved performance by 15%, while doing the same for plan activity did not improve performance. We also found low correlations in activity between pairs of simultaneously-recorded units and across time periods within a given trial condition. These results show that decode performance can be significantly improved by combining information from the plan and peri-movement periods, and that there is nearly no loss in performance when assuming independence between units and across time periods within a given trial condition.

Keywords—Brain-machine interface, neural coding and decoding, motor control, pre-motor cortex.

I. INTRODUCTION

Neural prosthetic systems aim to assist disabled patients by translating neural activity into control signals for prosthetic devices. It is now possible for monkeys to move computer icons solely by activating neural populations that participate in natural arm movements [1]–[3] and there is intense interest in increasing system performance.

Current neural prosthetic decode algorithms are based either on *plan* activity using maximum-likelihood (ML) techniques [4], or on *peri-movement* activity using linear filters or population vectors [1]–[3]. Plan activity is neural activity present before or even without natural arm movements, while peri-movement activity is neural activity present during natural arm movements. We recently proposed an estimation algorithm that decoded jointly using plan and peri-movement activity [5], [6]. In a simulation framework, we showed that decoding using **both** types of activity led to more accurate reconstructions of movement trajectories than when decoding based on plan or peri-movement activity alone.

This work was supported by NDSEG Fellowships (B.M.Y. and G.S.), the Christopher Reeve Paralysis Foundation (S.I.R. and K.V.S.), the Helen Hay Whitney Foundation Postdoctoral Research Fellowship (M.M.C.) and the following awards to K.V.S.: NSF Center for Neuromorphic Systems Engineering at Caltech, ONR, Whitaker Foundation, Center for Integrated Systems at Stanford, Sloan Foundation, and Burroughs Wellcome Fund Career Award in the Biomedical Sciences. Please address correspondence to byronyu@stanford.edu.

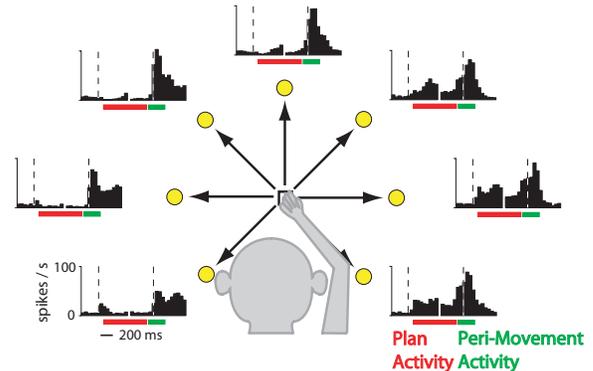


Fig. 1. Spike histograms showing plan and peri-movement activity to each of the seven targets (Unit G20040203.21.3). Dotted lines denote peripheral target presentation and movement onset.

Here we apply this idea to a delayed reach task using neural data recorded from the dorsal pre-motor (PMd) cortex. We compare the decode performance of models that use both plan and peri-movement activity with those that either use only one type of activity or ignore the difference between the two types of activity. We also consider the correlation in activity between pairs of simultaneously-recorded units and across time periods within a single unit.

II. BEHAVIORAL TASK AND RECORDINGS

Animal protocols were approved by the Stanford University Institutional Animal Care and Use Committee. We trained a rhesus monkey (*Macaca Mulatta*) to perform delayed center-out reaches to visual targets presented on a fronto-parallel screen (Fig. 1). The monkey touched a central target and fixated his eyes on a crosshair at the upper right corner of the central target. After 700 ms, a pseudo-randomly chosen peripheral target located at one of seven possible radial locations appeared (0, 45, 90, 135, 180, 225, 315° at a radius of 10 cm). At the same time, the crosshair moved out to the upper-right corner of the peripheral target and the monkey saccaded there. After a pseudo-randomly chosen delay period of 200, 750, or 1000 ms, the peripheral target increased in size as the GO cue and the monkey reached to the peripheral target. After a peripheral target hold time of 400 ms, the monkey received a liquid reward.

Trials with a delay period of 200 ms were inserted to encourage the monkey to plan throughout the delay period and were not used in the subsequent analyses. We recorded 3D arm position (60 Hz) and eye position (240 Hz).

A commercially available 96-channel silicon electrode array (Neuroport 128, Cyberkinetics Inc., Foxborough, MA,

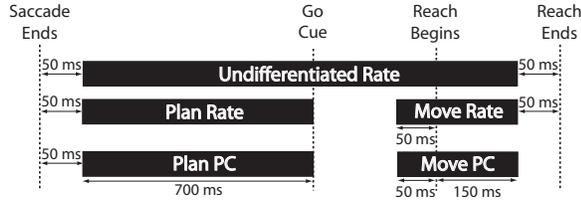


Fig. 2. Definition of behavioral periods. Reaction times ranged from 214 to 392 ms (mean 270, s.d. 27). Reach durations ranged from 191 to 363 ms (mean 256, s.d. 29).

USA) was microsurgically implanted into the right premotor cortex of a rhesus macaque monkey with standard neurosurgical techniques. The array is connected to a data acquisition system (Cerebus, Cyberkinetics Inc., Foxborough, MA, USA) that provides on-line recording and processing of neural signals. Online manual spike sorting was performed by setting a voltage threshold trigger to obtain waveforms and time-amplitude hoops to isolate sets of waveforms defined as units. 125 total units, including single-neuron and multi-neuron units, were isolated.

III. MODELING AND DECODING

We considered the behavioral periods defined in Fig. 2. From each of these windows in a given trial, we extracted either the firing rate averaged across the window or the principal component (PC) score of the windowed spike train.

While the firing rate captures the overall level of spiking activity in a given window, the PC score takes into account the time-varying structure of activity in the window. The PC scores were computed by first convolving each spike train with a Gaussian kernel of length 50 ms. Then, the smoothed spike trains were grouped by reach direction and averaged across trials. The PC directions were determined based on these seven average responses. The PC scores of the both training and test trials were computed using these PC directions.

We fit the trial-by-trial firing rates and PC scores using a multivariate Gaussian distribution

$$f(\mathbf{r} | d) = \frac{1}{(2\pi)^{n/2} |\Sigma_d|^{1/2}} e^{-\frac{1}{2}(\mathbf{r} - \boldsymbol{\mu}_d)^T \Sigma_d^{-1} (\mathbf{r} - \boldsymbol{\mu}_d)} \quad (1)$$

where $\mathbf{r} \in \mathbb{R}^n$ is a vector of firing rates and/or PC scores across units for a single trial, and $\boldsymbol{\mu}_d \in \mathbb{R}^n$ and $\Sigma_d \in \mathbb{R}^{n \times n}$ are the mean vector and the covariance matrix fitted to the training data for reach direction $d \in \{1, \dots, 7\}$. This is illustrated in Fig. 3 for $n = 2$.

We decoded reach direction using maximum likelihood

$$\hat{d} = \underset{d}{\operatorname{argmax}} P(d | \mathbf{r}) \quad (2)$$

$$= \underset{d}{\operatorname{argmax}} \frac{f(\mathbf{r} | d) P(d)}{f(\mathbf{r})} \quad (3)$$

$$= \underset{d}{\operatorname{argmax}} f(\mathbf{r} | d) \quad (4)$$

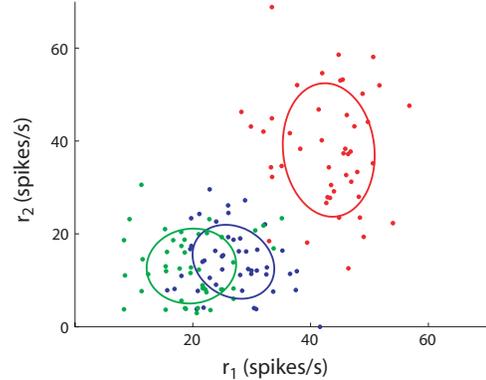


Fig. 3. Multivariate Gaussian data-fitting. Each point corresponds to a single trial and its color corresponds to the actual reach direction on that trial. A covariance ellipsoid is fit to the set of data points for each reach direction. In this example, r_1 represents Plan Rate and r_2 represents Move Rate (Unit G20040204.92.3). Only three reach directions (0° blue, 90° green, 180° red) and their 50% confidence ellipsoids are shown.

where \hat{d} is the estimated reach direction. (3) is obtained using Bayes' rule and (4) is a result of all reach directions being equally likely and $f(\mathbf{r})$ not being dependent on d .

All decode algorithms considered in this work use (1) for data fitting and (4) for decoding. The decode algorithms differ only in the quantities represented by \mathbf{r} , as detailed below. Let N be the number of simultaneously-recorded units used for decoding.

A. Undifferentiated Rate

For each unit on a given trial, we extracted the average firing rate across a large window encompassing the plan and peri-movement periods. By not distinguishing between plan and peri-movement activity, this algorithm served as a baseline for comparison with other algorithms in this work. Here, \mathbf{r} contains one firing rate value for each unit ($n = N$).

B. Plan Rate / Move Rate

Since a unit can have distinctly different activation patterns during the plan and peri-movement periods, we reasoned that it would be advantageous to compute the average firing rate separately for each period. Here, \mathbf{r} contains two firing rate values for each unit – one for the plan period and the other for the peri-movement period ($n = 2N$).

C. Plan Rate / Move PC

Although plan activity generally maintains a relatively constant firing rate across time, peri-movement activity usually exhibits a marked time-varying structure. To take this time-varying firing rate into account, we characterized the peri-movement activity using its first PC score. This is in contrast to computing the firing rate averaged across the peri-movement period, which washes out any time-varying activity structure. Here, \mathbf{r} contains a plan period firing rate and a peri-movement PC score for each unit ($n = 2N$).

IV. RESULTS

Decode performance was measured using an absolute angular error metric. On a given trial,

$$e = |\theta_d - \theta_{\hat{d}}| \quad (5)$$

where e is the angular error, θ_d is the actual reach angle, $\theta_{\hat{d}}$ is the decoded reach angle. Although reach direction was classified to one of seven directions, we used angular error rather than percent correct because angular error takes into account how far away the decoded reach direction is from the actual reach direction.

A. Independent Model

The primary comparison in this work was between Undifferentiated Rate and Plan Rate / Move Rate. We assumed independence between all pairs of elements in \mathbf{r} , conditioned on the reach direction d . This was equivalent to zeroing all off-diagonal elements of Σ_d in (1). If we ignored the distinction between the plan period and the peri-movement period (Undifferentiated Rate), we obtained an average error of 4.9° , as shown in Fig. 4. However, if we computed the firing rates in the plan and peri-movement periods separately and brought their estimates of reach direction together (Plan Rate / Move Rate), the average error dropped to 3.8° . Performance could be further improved by taking the time-structure of peri-movement activity into account (Plan Rate / Move PC), which produced an average error of 3.2° . There was a significant difference in performance for all pairwise comparisons of these three decode algorithms (Wilcoxon paired-sample test, $p < 0.05$). Fig. 5 shows how the performance of these three algorithms varies with unit count and training set size.

Plan PC / Move PC gave an average error of 3.4° , which did not differ from the performance of Plan Rate / Move PC (Wilcoxon paired-sample test, $p > 0.05$). This result is consistent with the observation that the firing rate stays relatively constant during the plan period, since trying to take into account any time-varying structure did not improve decode performance.

We also compared the decode performance of using both plan and peri-movement activity (Plan Rate / Move Rate) with using only one of the two types of activity. We found that the average error for Plan Rate alone was 8.6° , while that for Move Rate alone was 13.2° . There was a significant difference in performance between each of these two decode algorithms and Plan Rate / Move Rate (Wilcoxon paired-sample test, $p < 0.05$).

B. Second-Order Model

We then asked whether a second-order model using the full covariance matrix Σ_d in (1) would perform better than the independent model. To avoid overfitting, we compared the performance of the independent and second-order models at $n = 2$ with 45 training trials per reach condition. We

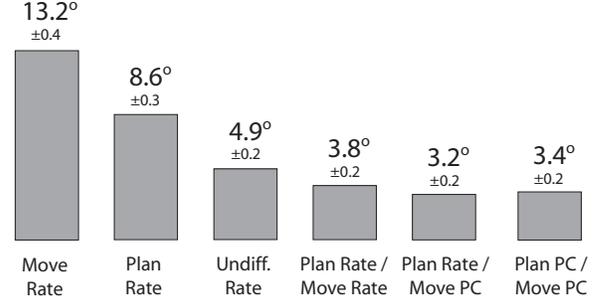


Fig. 4. Comparison of cross-validated decode performance. Heights of bars indicate absolute angular error (mean ± sem) over 5000 decodes. Each decode was based on 20 units and 45 training trials per reach direction.

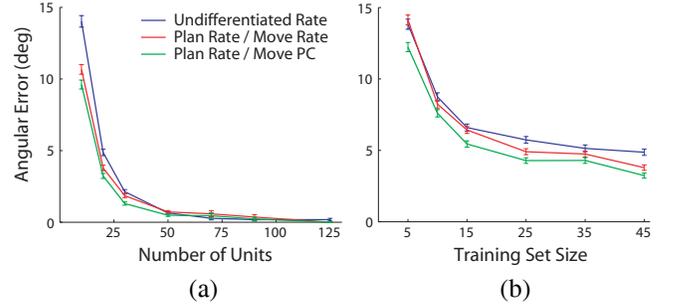


Fig. 5. Angular error (mean ± sem) as a function of (a) unit count with 45 training trials per reach direction, and (b) training set size with 20 units.

considered three cases: Plan Rate alone with two units, Move Rate alone with two units, and Plan Rate / Move Rate with one unit. In all three cases, there was no difference in performance between the second-order and independent models (Wilcoxon paired-sample test, $p > 0.05$), as shown in Table I.

TABLE I
ANGULAR ERROR OF INDEPENDENT AND SECOND-ORDER MODELS
(MEAN ± SEM OVER 5000 DECODES)

	Independent	Second-Order
Plan Rate (2 units)	60.0 ± 0.8°	59.9 ± 0.8°
Move Rate (2 units)	64.3 ± 0.8°	64.3 ± 0.8°
Plan Rate / Move Rate (1 unit)	62.9 ± 0.8°	63.4 ± 0.8°

Fig. 6(a) and (b) show the histograms of correlation coefficients (black) for all pairwise combinations of the 125 simultaneously-recorded units for Plan Rate alone and Move Rate alone, respectively. Fig. 6(c) shows the histogram of correlation coefficients (black) between Plan Rate and Move Rate for each unit. We found that 13.1% of unit pairs for Plan Rate, 6.1% of units pairs for Move Rate, and 5.0% of units for Plan Rate / Move Rate had significant correlations (t test, $p < 0.05$).

To determine whether these correlations could have arisen by chance, we shuffled trials corresponding to the same reach direction and recomputed the histograms of correlation coefficients (green). We found a significant difference between the the trial-shuffled and unshuffled distributions in the first two cases (Wilcoxon paired-sample test, $p < 0.05$),

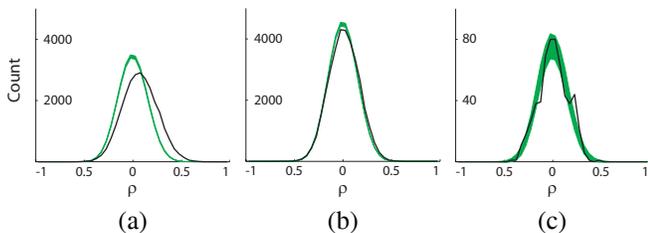


Fig. 6. Histograms of trial-shuffled (green, mean \pm s.d. over 1000 shuffles) versus unshuffled (black) correlation coefficients. (a) Plan Rate of unit pairs (unshuffled mean=0.073), (b) Move Rate of unit pairs (unshuffled mean=0.010), (c) Plan Rate and Move Rate of a single unit (unshuffled mean=0.005). We did not include correlation coefficients for which either one or both of the sets firing rates being correlated had an average of less than one spike per second.

but no difference in the third case (Wilcoxon paired-sample test, $p > 0.05$).

V. DISCUSSION

Current neural prosthetic decode algorithms are based either on plan activity [4] or on peri-movement activity [1]–[3]. We asked whether we could increase decode performance by using **both** types of activity. To investigate, we trained a rhesus monkey to perform a delayed reach task in which plan and peri-movement activity were temporally dissociated.

We used a multivariate Gaussian distribution to separately model plan and peri-movement activity and a maximum likelihood decoder to bring their estimates together. We found that Plan Rate / Move Rate had 56% lower error than Plan Rate alone and 71% lower error than Move Rate alone. Thus, neural prosthetic decode performance can be increased by utilizing both plan and peri-movement activity.

However, this result is hardly surprising given that Plan Rate / Move Rate takes into account more spike data than Plan Rate alone or Move Rate alone. A more appropriate comparison is to equalize the amount of spike data used by comparing Plan Rate / Move Rate with Undifferentiated Rate. While Plan Rate / Move Rate allows the plan period to have a different firing rate from the peri-movement period, Undifferentiated Rate requires both periods to have the same firing rate by averaging the firing rate across the two periods. We found that Plan Rate / Move Rate had 22% lower error than Undifferentiated Rate, even though Plan Rate / Move Rate took into account slightly less spike data than Undifferentiated Rate (as can be seen in Fig. 2).

The decode performance using both plan and peri-movement activity could be further improved by taking into account the time-varying structure in peri-movement activity. Overall, we obtained an improvement in performance of 62% over Plan Rate alone and 75% over Move Rate alone by using Plan Rate / Move PC.

Previous studies indicate varying levels of firing rate correlation between pairs of units within a single trial condition. While low correlations were reported in M1 [7], [8] and SMA [9], a different study found comparatively high correlations in M1 [10]. Here, we report relatively low

correlations between pairs of units and across time periods in PMd.

The comparison between trial-shuffled and unshuffled correlation coefficients shows that, while there are real correlations in the unshuffled data, most of the unshuffled correlations could have arisen by chance. Thus, it is not surprising that using these spurious correlations in the second-order model did not improve performance.

VI. CONCLUSION

Neural prosthetic system performance can be improved by utilizing both plan and peri-movement activity. We showed that combining the two types of activity (Plan Rate / Move Rate) led to greater classification accuracy than either using only one type of activity (Plan Rate alone, Move Rate alone) or ignoring the difference between them (Undifferentiated Rate). We also found no performance difference between using the full covariance structure (second-order model) and assuming independence between unit pairs and across time periods (independent model). We are currently exploring different techniques to model and combine plan and peri-movement activity to further improve decode performance.

ACKNOWLEDGMENTS

We thank Caleb Kemere for valuable discussions and Missy Howard for expert surgical assistance and veterinary care.

REFERENCES

- [1] D. M. Taylor, S. I. Helms Tillery, and A. B. Schwartz, "Direct cortical control of 3D neuroprosthetic devices," *Science*, vol. 296, pp. 1829–1832, 2002.
- [2] M. D. Serruya, N. G. Hatsopoulos, L. Paninski, M. R. Fellows, and J.P. Donoghue, "Instant neural control of a movement signal," *Nature*, vol. 416, pp. 141–142, 2002.
- [3] J. M. Carmena, M. A. Lebedev, R. E. Crist, J. E. O'Doherty, D. M. Santucci, D. F. Dimitrov, P. G. Patil, C. S. Henriquez, and M. A. L. Nicolelis, "Learning to control a brain-machine interface for reaching and grasping by primates," *PLoS Biology*, vol. 1, no. 2, 2003.
- [4] K. V. Shenoy, D. Meeker, S. Cao, S. A. Kureshi, B. Pesaran, P. Mitra, C. A. Buneo, A. P. Batista, J. W. Burdick, and R. A. Andersen, "Neural prosthetic control signals from plan activity," *NeuroReport*, vol. 14, pp. 591–596, 2003.
- [5] C.T. Kemere, G. Santhanam, B.M. Yu, K.V. Shenoy, and T.H. Meng, "Decoding of plan and peri-movement neural signals in prosthetic systems," *IEEE Workshop on Signal Processing Systems*, pp. 276–283, Oct. 2002.
- [6] C.T. Kemere, K.V. Shenoy, and T.H. Meng, "Model-based neural decoding of reaching movements: a maximum likelihood approach," *IEEE Transactions on Biomedical Engineering*, 2004, in press.
- [7] D. Lee, N.L. Port, W. Kruse, and A.P. Georgopoulos, "Variability and correlated noise in the discharge of neurons in motor and parietal areas of the primate cortex," *J. Neuroscience*, vol. 18, pp. 1161–1170, 1998.
- [8] D. Lee, N. P. Port, W. Kruse, and A. P. Georgopoulos, "Neuronal population coding: Multielectrode recordings in primate cerebral cortex," in *Neuronal ensembles: strategies for recording and decoding*, H. Eichenbaum and J. L. Davis, Eds. Wiley-Liss, New York, 1998.
- [9] B.B. Averbeck and D. Lee, "Neural noise and movement-related codes in the macaque supplementary motor area," *J. Neuroscience*, vol. 23, pp. 7630–7641, 2003.
- [10] E. M. Maynard, N. G. Hatsopoulos, C. L. Ojakangas, B. D. Acuna, J. N. Sanes, R. A. Normann, and J. P. Donoghue, "Neuronal interactions improve cortical population coding of movement direction," *J. Neuroscience*, vol. 19, pp. 8083–8093, 1999.