SUPPLEMENTAL TEXT

Primary Motor Cortex Tuning to Intended Movement Kinematics in Humans with Tetraplegia. ***Journal of Neuroscience, 2008.***

Wilson Truccolo[1], Gerhard M. Friehs[4,5], John P. Donoghue[1,6] and Leigh R. Hochberg[2,3,1]

[1]Department of Neuroscience and Brain Science Program, 60 Olive St., Providence, RI, 02912, USA. [2]Center for Restorative and Regenerative Medicine, Rehabilitation R&D Service, Department of Veterans Affairs, Veterans Health Administration, 830 Chalkstone Avenue, Providence, Rhode Island 02908, USA. [3]Department of Neurology, Massachusetts General Hospital, Brigham and Women's Hospital, and Spaulding Rehabilitation Hospital, Harvard Medical School, 55 Fruit Street, Boston, Massachusetts 02114, USA. [4]Department of Clinical Neurosciences (Neurosurgery), Brown University. [5]Department of Neurosurgery, Rhode Island Hospital, 120 Dudley Street, Suite 103, Providence, Rhode Island 02905, USA. [6]Cyberkinetics Neurotechnology Systems, Inc., 100 Foxborough Boulevard–Suite 240, Foxborough, Massachusetts 02035, USA.

Corresponding author: Wilson_Truccolo@Brown.edu

A. Off-line decoding of instantaneous velocity during pursuit tracking task

We represented the spiking activity (recorded spike times) of each sorted unit as a discrete time neural point process (Truccolo et al., 2005), i.e., as a binary sequence (spike train) obtained by determining whether a spike occurred or not in each consecutive time bin of width $\Delta t = 1ms$. The value of this sequence for the c^{th} neuron, among C recorded neurons, at a particular discrete time t is here denoted by $\Delta N_t^c \in \{0,1\}$.

The observed spike trains from all of the tuned neurons together with the corresponding velocity model (Eq. 2, main text; fit to training data sets) were used to decode the velocity of the TC cursor (in test data sets). A stochastic state space point process filter was used. See Eden et al. (2004) and Truccolo et al. (2005) for detailed derivations and applications to motor neurophysiology. Following Bayes' rule, the posterior probability distribution for velocity at time $t+\tau$ can be expressed as

$$p(\dot{\mathbf{x}}_{t+\tau} \mid \Delta N_{0:t}) = \frac{\Pr(\Delta N_t \mid \Delta N_{0:t-\Delta t}, \dot{\mathbf{x}}_{t+\tau})\, p(\dot{\mathbf{x}}_{t+\tau} \mid \Delta N_{0:t-\Delta t})}{\Pr(\Delta N_t \mid \Delta N_{0:t-\Delta t})}, \quad \text{(S1)}$$

where $\Delta N_{0:t} = \{\Delta N_{0:t}^{1}, \Delta N_{0:t}^{2}, \ldots, \Delta N_{0:t}^{C}\}$ corresponds to the collection of spike trains from C observed neurons during an interval from time zero up to time *t*, and $\dot{\mathbf{x}}_{t+\tau}$ is the velocity vector at time $t+\tau$. The following state space model was adopted

$$\begin{aligned} &\dot{\mathbf{x}}_{t+\tau} = \mathbf{F}\dot{\mathbf{x}}_{t+\tau-\Delta t} + \varepsilon_{t+\tau} \\ &\Delta N_t^c \sim \Pr\{\Delta N_t^c \mid \lambda_t^c(\dot{\mathbf{x}}_{t+\tau})\} \end{aligned}, \quad \text{(S2)}$$

where the time evolution of the velocity vector $\dot{\mathbf{x}}_{t+\tau}$ was modeled as a Gaussian autoregressive process of order 1, the dimension of the state space is $m = 2$ in our case, $\mathbf{F}$ is a $m \times m$ state matrix, and ε_t is the noise term given by a zero mean *m*-dimensional white noise Gaussian vector with $m \times m$ covariance matrix $\mathbf{W}_\varepsilon$. The matrices **F** and $\mathbf{W}_\varepsilon$ were fit via maximum likelihood. The point process observation equation was expressed in terms of the conditional probability

$$\Pr\{\Delta N_t^c \mid \lambda_t^c(\dot{\mathbf{x}}_{t+\tau})\} \approx [\lambda_t^c(\dot{\mathbf{x}}_{t+\tau})\Delta t]^{\Delta N_t^c}[1-\lambda_t^c(\dot{\mathbf{x}}_{t+\tau})\Delta t]^{1-\Delta N_t^c}, \quad \text{(S3)}$$

where $\lambda_t^c(\dot{\mathbf{x}}_{t+\tau})$ is given by the velocity model (Eq. 2, main text).

Based on this state space model, a predicted or decoded velocity $\hat{\dot{\mathbf{x}}}_{t+\tau}$ given the observed ensemble spiking activity at time t can be obtained via the following recursive point process filter:

$$\hat{\dot{\mathbf{x}}}_{t+\tau} = \hat{\dot{\mathbf{x}}}_{t+\tau}^{-} + \mathbf{W}_{t+\tau} \times \sum_{c=1}^{C} \nabla \log \lambda_t^c (\hat{\dot{\mathbf{x}}}_{t+\tau}^{-})[\Delta N_t^c - \lambda_t^c (\hat{\dot{\mathbf{x}}}_{t+\tau}^{-})\Delta t], \qquad \text{(S4)}$$

where

$$\hat{\dot{\mathbf{x}}}_{t+\tau}^{-} = \mu_{\mathbf{x}} + \mathbf{F}\hat{\dot{\mathbf{x}}}_{t+\tau-1} \qquad \text{(S5)}$$

is the one step prediction;

$$\begin{aligned} \mathbf{W}_{t+\tau} = inv(\ \ & inv(\mathbf{W}_{t+\tau}^{-}) + \\ & \sum_{c=1}^{C} [\nabla \log \lambda_t^c (\dot{\mathbf{x}}_{t+\tau}^{-})]\ \lambda_t^c (\hat{\dot{\mathbf{x}}}_{t+\tau}^{-})\ \Delta t\ [\nabla \log \lambda_t^c (\hat{\dot{\mathbf{x}}}_{t+\tau}^{-})]^T \\ & - \sum_{c=1}^{C} \nabla^2 \lambda_t^c (\dot{\mathbf{x}}_{t+\tau}^{-})\ [\Delta N_t^c - \lambda_t^c (\dot{\mathbf{x}}_{t+\tau}^{-})\ \Delta t]\ ) \end{aligned} \qquad \text{(S6)}$$

is the estimated posterior covariance matrix of $\dot{\mathbf{x}}_{t+\tau}$;

$$\mathbf{W}_{t+\tau}^{-} = \mathbf{F}\mathbf{W}_{t+\tau-1}\mathbf{F}^T + \mathbf{W}_{\varepsilon} \qquad \text{I} \qquad \text{(S7)}$$

is the one-step prediction covariance matrix;

and *inv*( ) and *T* denote the matrix inverse and transpose, respectively. The term $\nabla(\nabla^2)$ denotes the *m*-dimensional column vector ($m\times m$ matrix) of first (second) partial derivatives with respect to $\hat{\mathbf{x}}_{k+\tau}$. The initial value for the decoded velocity at time $t=0$ was sampled from a zero mean Gaussian distribution with variance equal to the variance of the observed TC velocity. Note that the true observed velocities never entered the recursive algorithm; the input to the decoding algorithm consisted only of the observed spike trains. For visualization purposes, we plotted direction and speed of the decoded velocity vector in the Results section. The decoded output was smoothed ad-hoc with a Hanning window.

B. Off-line decoding of instantaneous position during pursuit tracking task

For position decoding, we employed the same algorithm describe above in (A), with the exception that the state space model was written in terms of position and that the position model, i.e. $\lambda_t(\mathbf{x}_{t+\tau})$ from Eq. 1 (main text), was used.

C. Phase randomization confidence intervals for cross-correlation functions during pursuit tracking task

We computed cross-correlation functions (at multiple time lags, -4 to 4 sec) between the 1-minute long spike trains (transformed into spike counts in 50ms time bins) and the corresponding time series of a selected kinematic covariate. These functions were then averaged across the four pursuit tracking blocks in each session. Confidence intervals (95%) for the null hypothesis (zero cross-correlation) were obtained as follows. For each neuron and covariate time series pair, an empirical distribution of cross-correlation values was generated based on 400 surrogated data sets. Each of these surrogated data sets was generated via a phase randomization procedure applied to the covariate time series. This procedure consisted of first computing the Fourier transform of the time series, followed by a random permutation of the phase of the Fourier transform components, and finally applying the inverse Fourier transform. In this way, the surrogated data sets randomized the spiking-covariate temporal relationship, while preserving the same power spectrum structure of the covariate.

References

Eden U.T., Frank, L.M., Barbieri, R., Solo, V., Brown, E.N. (2004) Dynamic analysis of neural encoding by point process adaptive filtering. *Neural Computation* 16: 971–998.

Truccolo, W., Eden, U. T., Fellows, M. R., Donoghue, J. P., Brown, E. N. (2005). A point process framework for relating neural spiking activity to spiking history, neural ensemble, and extrinsic covariate effects. *Journal of Neurophysiology* 93(2): 1074-89.