Partial Cross-Correlation Analysis Resolves Ambiguity in the Encoding of Multiple Movement Features

Eran Stark,[1] Rotem Drori,[1,2] and Moshe Abeles[1,2,3]

[1]*Department of Physiology, Hadassah Medical School, and* [2]*The Interdisciplinary Center for Neural Computation, The Hebrew University of Jerusalem, Jerusalem; and* [3]*Gonda Brain Research Center, Bar-Ilan University, Ramat-Gan, Israel*

Stark, Eran, Rotem Drori, and Moshe Abeles. Partial cross-correlation analysis resolves ambiguity in the encoding of multiple movement features. *J Neurophysiol* 95: 1966–1975, 2006. First published November 30, 2005; doi:10.1152/jn.00981.2005. A classical question in neuroscience is which features of a stimulus or of an action are represented in brain activity. When several features are interdependent either at a given point in time or at distinct points in time, neural activity related to one feature appears to be correlated with other features. Thus techniques that simultaneously consider multiple features cannot account for delayed interdependencies between features. The result is an ambiguity with respect to the encoded features. Here, we resolve this ambiguity by applying a novel statistical method based on partial cross-correlations. The method yields estimates of linear correlations between neural activity and a given feature that are not affected by linear correlations with other features at multiple time delays. The method also provides a graphical output measured on a scale that allows for comparisons between different features, neurons, and experiments. We use real movement data and neural activity simulated according to a wide range of tuning models to illustrate the method. When applied to real neural activity, the procedure yields results that indicate which of the considered features the neural activity is related to and at what time delays.

INTRODUCTION

A general approach in experimental neurophysiology is to characterize single neuron activity in relation to stimulus or to action variables (Georgopoulos et al. 1982; Hubel and Wiesel 1962; Kakei et al. 1999; Merzenich and Brugge 1973; Mountcastle 1957; Scott and Kalaska 1997). A difficulty arises on attempting to relate neural activity to multiple variables that are interdependent. Neural activity may then appear to be related to one type of stimulus or action simply because it is related to another stimulus or action, which, in turn, is related to the first. For instance, during reaching movements made by a monkey (Georgopoulos et al. 1982), the firing rate of a neuron is estimated while the monkey moves its hand in various directions from a fixed initial position (Fig. 1*A*). Even when a clear directional response is observed, it is difficult to ascertain whether the response is related to the direction of the velocity vector, to the direction of the acceleration vector, or to both, because the two are highly correlated (Fig. 1*A*, *bottom*).

Ideally, experiments free from correlations between stimuli or between action dimensions should be constructed, but this is often impossible due to physical constraints. For instance, during continuous curved movements, correlations between directions of velocity and acceleration at zero delay are low.

Yet these features are correlated at other time delays (Fig. 1*B*). This is so because a change in the position of the arm is preceded by an elevation in movement velocity which, by necessity, is preceded by accelerating in the same direction.

The problem is quite general because the movement may involve either a visual stimulus such as a moving dot or a natural scene, or some output machinery such as an oculomotor apparatus or an arm. Neural activity that is related to one movement feature will appear to be related to other features that are correlated with the first at some temporal delay (Fig. 1*B*). Therefore inherent interdependencies between features must be controlled for statistically. Cross-correlation analysis does not account for such dependencies because it considers only two features at a time. When using other conventional techniques, such as regression analysis (Draper and Smith 1981), and considering multiple features at the same time, delayed dependencies between features are obviously not accounted for. This may bias the conclusions with respect to which features are encoded in neural activity (Todorov 2000). Despite its importance, up until now this issue was not attended to.

Here we present the partial cross-correlation matrices (PCCM) method. The method yields estimates of linear correlations between neural activity and movement features while taking into account interdependencies among features. The interdependencies are considered at various time delays. Consequently, the method renders it possible to determine to which movement features the neural activity is related and at what time delays. A graphical output, measured on a scale that can be readily compared between different features, neurons, and experiments, is also provided. The method thus provides a means of resolving ambiguity in parameter encoding.

In this paper, we present a detailed description of the method and demonstrate its use. We apply the method to two types of data: simulated neural activity with movement data from continuous movement experiments and real neural activity recorded from monkey motor cortex in conjunction with the same type of movement data.

METHODS

Experimental procedures

One monkey (*Macaca fascicularis*, female, 3.5 kg) was used in this study. Animal handling procedures were in accordance with the *National Institutes of Health Guide for the Care and Use of Laboratory Animals* (1996), complied with Israeli law, and were approved by
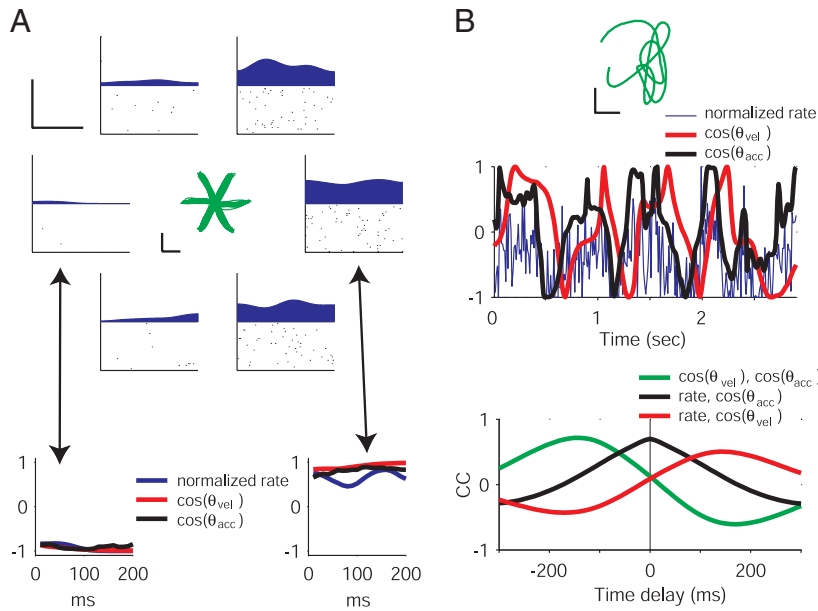

FIG. 1. Correlated movement features and simulated neural activity. *A*: center-out movements. Reaching movements were made from a central location outwards in different directions (*central panel;* calibration: 2 cm). Neural activity was simulated according to a model of tuning to θ_{acc}, applied to real movements (*Eq. 1;* $\tau_{acc} = 0$). Raster plots and peri-event time histograms are shown in the 6 peripheral panels (horizontal calibration: 100 ms; vertical calibration: 10 spikes/s). Rate was divided by the maximum and translated to the range −1 to 1, yielding a normalized rate. In this case, it is impossible to determine whether a directional selectivity of a neuron is related to θ_{vel} or θ_{acc} or both because the 2 have the same direction as shown by the examples in the *bottom panels. B*: continuous tracing paradigm. The problem is partially solved by using an alternative experimental paradigm. *Top*: path of 1 continuous tracing trial (calibration: 2 cm). Directions of velocity (*middle*, red) and acceleration (black) traces from that trial are shown together with simulated neural activity (blue). θ_{vel} and θ_{acc} are not the same as was the case in the center-out paradigm (*A*). This is reflected in low correlation coefficient (CC) between these 2 features at 0 delay (*bottom*, green; cross-correlation profiles were averaged over 387 trials). However, the 2 features are highly correlated at nonzero delays: acceleration precedes velocity by ~150 ms (peak of green curve). Ambiguity ensues: does the neuron encode acceleration at 0 delay (black) or velocity at a time delay of 150 ms (red)?

the Ethics Committee of the Hebrew University. The monkey was trained to perform a tracing task involving continuous curved movements. It was seated in a primate chair with the right hand restrained and operated a two-joint low-friction manipulandum with its left hand. A horizontal screen, mounted at chest level, blocked the view of the manipulandum and hand. A circular cursor (diameter: 0.4 cm) indicating hand endpoint was projected on the screen. The monkey traced given paths at its own pace. There were 40 different paths in each recording session, generated by cubic spline interpolation of 10 randomly placed points. This resulted in highly variable motion and modest correlations at zero delay (Fig. 1*B*). As soon as the monkey finished tracing a path, a juice reward was administered. Trials in which the monkey paused or deviated from the designated path for >800 ms were aborted.

During each recording session, up to eight glass-coated tungsten microelectrodes (impedance: 0.2–1.5 MΩ at 1 kHz), confined to a guide tube (internal diameter: 1.5 mm), were inserted into the contralateral primary motor or dorsal premotor area using a computer-controlled micro-drive (EPS, Alpha-Omega Engineering, Nazareth, Israel). The output signal of each electrode was amplified (10,000), band-pass filtered (0.3–6 kHz, MCP+, Alpha-Omega Eng.), and fed to a template matching device (MSD, Alpha-Omega Eng.) to isolate the activity of one to three units per electrode. Spikes and behavioral events were sampled at 1 kHz and logged on a custom data-acquisition system. Hand position was sampled at 100 Hz and was low-pass filtered (8 Hz). Velocity and acceleration were computed by taking the corresponding derivatives and converting the latter to polar coordinates. Spike trains were smoothed by convolution with a Gaussian kernel (SD = 50 ms) and down-sampled to 100 Hz. This produced an estimate of the rate profile at the same temporal resolution as the movement data. Only well-isolated units that were judged to exhibit stationary activity during more than 50 consecutive trials were used.

Simulation details

To test whether our method could indeed correct for the mixed effect of multiple movement features, we simulated neural activity based on models of tuning to various movement features. In this manner, we know which feature the neural activity is related to and can therefore compare our expectations with the output of the method. For simulations, we used real movement profiles obtained from the Macaque continuous tracing experiments. In the first model, firing rate

$R(t)$ was a function of the cosine of the direction of the acceleration vector

$$R(t) = B + G \cdot \cos\left[\theta_{acc}(t + \tau_{acc}) - \theta_0\right] + \varepsilon(t) \quad (1)$$

In the latter equation, B is the baseline firing rate (spikes/s), G is a dimensionless gain, θ_{acc} is the observed direction of the acceleration vector, θ_0 is the preferred direction (PD) of the putative neuron, and τ_{acc} is the delay between the neural activity and movement (ms). Positive delays correspond to neural activity preceding movement. ε is a zero-mean Gaussian noise with SD σ_N. To test the performance of the method on neural activity tuned to the direction of the velocity vector (θ_{vel}), we used the same model. Because previous studies have shown a relationship of neural activity to amplitudes of acceleration (A_{acc}) (Flament and Hore 1988) or of velocity (A_{vel}) (Moran and Schwartz 1999a) vectors, we considered models with amplitude components. We used a linear combination of a cosine relation with θ_{vel} and a linear relation with A_{vel}

$$R(t) = B + G_1 \cdot A_{vel}(t + \tau_{vel}) + G_2 \cdot \cos\left[\theta_{vel}(t + \tau_{vel}) - \theta_0\right] + \varepsilon(t) \quad (2)$$

which treats A_{vel} as an additive component. We also used a cosine relation with θ_{vel} scaled by amplitude (Paninski et al. 2004)

$$R(t) = B + G \cdot A_{vel}(t + \tau_{vel}) \cdot \cos\left[\theta_{vel}(t + \tau_{vel}) - \theta_0\right] + \varepsilon(t) \quad (3)$$

treating A_{vel} as a multiplicative factor. An additional composite model was a linear combination of two cosine relations, with θ_{vel} and with θ_{acc}

$$R(t) = B + G_1 \cdot \cos\left[\theta_{acc}(t + \tau_{acc}) - \theta_{0(acc)}\right] \\ + G_2 \cdot \cos\left[\theta_{vel}(t + \tau_{vel}) - \theta_{0(vel)}\right] + \varepsilon(t) \quad (4)$$

In all models $\theta_0 = 0°$ was employed. Numerical simulations were carried out at discrete time steps of 10 ms in the following manner. First, we generated a rate profile for each trial according to one of the models but without noise ϵ. The SD of this noiseless signal, σ_S, was estimated for each trial separately and averaged across trials. Noise SD was then computed according to $\sigma_N = \sigma_S/SNR$ (signal-to-noise ratio). We employed SNR = 1. Signal-dependent noise ϵ was then generated by random sampling from a Gaussian distribution with zero mean and SD of σ_N and added to the signal. Negative rates were clipped to zero. We verified that the mean and SD of simulation outputs matched those observed for real neurons (mean rate of 5

spikes/s and SD of rate 5 spikes/s; these values were based on a sample of 194 motor cortical neurons). Free model coefficients, B and G, were set to yield these outputs. We obtained the same results using a wide range of SNR values and arbitrary coefficient settings.

Partial correlation coefficients

The linear correlation between any two random variables A and B (for instance, firing rate and the cosine of θ_{acc}) can be estimated by ρ_{AB}—the product-moment correlation coefficient (CC)

$$\rho_{AB} = \frac{\langle (A - \langle A \rangle) \cdot (B - \langle B \rangle) \rangle}{\sigma_A \cdot \sigma_B} \quad (5)$$

where $\langle A \rangle$ is the arithmetic average of A, and σ_A is its SD and likewise for B. When an additional variable C (for instance, the cosine of θ_{vel}) is suspected to account for some of the correlation, its linear effect may be removed using the partial CC (Fisher 1925), defined as

$$\rho_{AB/C} = \frac{\rho_{AB} - \rho_{AC} \cdot \rho_{BC}}{\sqrt{(1 - \rho_{AC}^2) \cdot (1 - \rho_{BC}^2)}} \quad (6)$$

where ρ_{AB}, ρ_{AC}, and ρ_{BC} are product-moment CCs. Linear correlations—between A and C and between B and C—are removed from ρ_{AB} by subtraction, and the difference is normalized by the geometric average of the deviations of these correlations from one (partial CCs range from -1 to 1 as do product-moment CCs). This is conceptually similar to estimating the correlation between A and B, whereas C is held constant experimentally. For instance, if B and C are both noisy duplicates of A, with SNR of 1, then the correlation between each of these and A is about 0.71. It is clear however that the correlation between B and C is substantial as well ($\sim$0.5). Computing *partial* correlations between A and B (or C) gives 0.58 instead of 0.71, but the correlation between B and C goes down to $\sim$0. Thus the partial correlation clarifies which variable is related to which other variable directly and to which other variable indirectly.

If two additional variables are to be accounted for, say C_1 and C_2, the second-order partial CC can be computed by

$$\rho_{AB/(C_1 \cdot C_2)} = \frac{\rho_{AB/C_1} - \rho_{AC_1/C_1} \cdot \rho_{BC_2/C_1}}{\sqrt{(1 - \rho_{AC_2/C_1}^2) \cdot (1 - \rho_{BC_2/C_1}^2)}} \quad (7)$$

where ρ_{AB/C_1} and so on are partial CCs (*Eq. 6*). In the latter expression, we first take into account the effect of C_1 by estimating first-order partial CCs (ρ_{AB/C_1}, ρ_{AC_2/C_1}, and ρ_{BC_2/C_1}) and then the effect of C_2, by actually plugging these first-order partial CCs into the expression. Note that this is not the only possible expression for obtaining an estimate of $\rho_{AB/C_1 \cdot C_2}$; we can reverse the order, first taking into account the effect of C_2 and subsequently of C_1. By extension, when three additional variables are present there are six (3!) equivalent recursive expressions for estimating third-order partial correlations. In the same manner, when n additional variables are to be accounted for, the nth-order partial correlation may be obtained by recursion. Alternatively, partial CCs of any order may be obtained from multiple linear regression (MLR) coefficients and their SDs (see APPENDIX for more details regarding both computational procedures).

Accounting for time-dependent correlations between variables using partial cross-correlations

Applied to scalar variables, partial CCs will remove all linear correlations. When variables are time series, this application is equivalent to removal of zero delay correlations, whereas delayed correlations are not considered. Therefore if the value of variable B at time t is related to the value of variable C at time $t + \tau$ but not at time t, $\rho_{AB/C}$ will be about the same as ρ_{AB}. We want to correct for the delayed effect of variable C on the correlation between variables A and B. Because the two variables in question may themselves be correlated at nonzero delays, we write the partial *cross*-correlation between A and B, given C, as a function of two time delays

$$\rho_{AB(\tau_1)/C(\tau_2)} = \frac{\rho_{AB(\tau_1)} - \rho_{AC(\tau_2)} \cdot \rho_{B(\tau_1)C(\tau_2)}}{\sqrt{(1 - \rho_{AC(\tau_2)}^2) \cdot (1 - \rho_{B(\tau_1)C(\tau_2)}^2)}} \quad (8)$$

where τ_1 is the delay between variables A and B and τ_2 is the delay between variables A and C, so $\rho_{AB(\tau_1)}$ is the cross-correlation between variables A and B at a time delay of τ_1 and so on. *Eq. 8* yields a partial cross-correlation matrix (PCCM) in which each element is $\rho_{AB/C}$ as a function of two time delays, τ_1 and τ_2. Single-trial partial CCs are Fisher z-transformed, $\hat{\rho}(\tau_1, \tau_2) = \tanh^{-1}[\rho(\tau_1, \tau_2)]$, so they are distributed normally under the null hypothesis of no correlation between variables.

In this paper, we estimate PCCMs for movement features within the range of -300 to 300 ms, in 10-ms increments. This range was selected on the basis of previous studies (Moran and Schwartz 1999b). Specifically, our goal is to account for effects of θ_{vel} and A_{vel} at the same time delay relative to the firing rate, τ_1, and for the effects of θ_{acc} and of A_{acc} at another time delay, τ_2. Taking the sine and cosine of θ_{vel} and θ_{acc}, we obtain together with A_{vel} and A_{acc}, a total of six movement features: A_{vel}, $\sin(\theta_{vel})$, $\cos(\theta_{vel})$, A_{acc}, $\sin(\theta_{acc})$, and $\cos(\theta_{acc})$. Denoting the firing rate profile by R and using the indexing convention of *Eq. 8*, the fifth-order PCCMs can be written as

$$M_{A_{vel}}(\tau_1, \tau_2) = \rho_{RA_{vel}(\tau_1)}/$$
$$\{\sin[\theta_{vel}(\tau_1)], \cos[\theta_{vel}(\tau_1)], A_{acc}(\tau_2), \sin[\theta_{acc}(\tau_2)], \cos[\theta_{acc}(\tau_2)]\} \quad (9)$$

$$M_{\sin_{vel}}(\tau_1, \tau_2) = \rho_{R\sin[\theta_{vel}(\tau_1)]}/$$
$$\{A_{vel}(\tau_1), \cos[\theta_{vel}(\tau_1)], A_{acc}(\tau_2), \sin[\theta_{acc}(\tau_2)], \cos[\theta_{acc}(\tau_2)]\} \quad (10)$$

$$M_{\cos_{vel}}(\tau_1, \tau_2) = \rho_{R\cos[\theta_{vel}(\tau_1)]}/$$
$$\{A_{vel}(\tau_1), \sin[\theta_{vel}(\tau_1)], A_{acc}(\tau_2), \sin[\theta_{acc}(\tau_2)], \cos[\theta_{acc}(\tau_2)]\} \quad (11)$$

$$M_{A_{acc}}(\tau_1, \tau_2) = \rho_{RA_{acc}(\tau_2)}/$$
$$\{A_{vel}(\tau_1), \sin[\theta_{vel}(\tau_1)], \cos[\theta_{vel}(\theta_1)], \sin[\theta_{acc}(\tau_2)], \cos[\theta_{acc}(\tau_2)]\} \quad (12)$$

$$M_{\sin_{acc}}(\tau_1, \tau_2) = \rho_{R\sin[\theta_{acc}(\tau_2)]}/$$
$$\{A_{vel}(\tau_1), \sin[\theta_{vel}(\tau_1)], \cos[\theta_{vel}(\tau_1)], A_{acc}(\tau_2), \cos[\theta_{acc}(\tau_2)]\} \quad (13)$$

$$M_{\cos_{acc}}(\tau_1, \tau_2) = \rho_{R\cos[\theta_{acc}(\tau_2)]}/$$
$$\{A_{vel}(\tau_1), \sin[\theta_{vel}(\tau_1)], \cos[\theta_{vel}(\tau_1)], A_{acc}(\tau_2), \sin[\theta_{acc}(\tau_2)]\} \quad (14)$$

Note that when $\tau_1 = \tau_2 = 0$, the PCCM result is equivalent to the MLR solution (APPENDIX) when regressing the firing rate R on all six features, at zero delay. When $\tau_1 = \tau_2$, the PCCM result is equivalent to the MLR solutions obtained when regressing R on all six features lagged together (Ashe and Georgopoulos 1994).

Computational specifics

Partial cross-correlations may be estimated in either the time or the frequency domains. In the time domain, this is accomplished by lagging each of several variables on all relevant delays relative to another variable, such as firing rate, and trimming to get matrices of equal sizes. For instance, to compute a doubly-lagged PCCM for three time series—vectors A, B, and C—vectors B and C are replicated, zero-padded, and lagged relative to the reference vector A. Resulting matrices are trimmed such that the number of data points is equal for all time delays, and the partial CC at delays of (τ_1, τ_2) is then computed between the trimmed vector A and the appropriate column of the trimmed matrix B, given the appropriate column of the trimmed matrix C.

Such estimation carries a computational burden which is proportional to $N^2 M^k$, N being the length of the trimmed time series, M being the maximal lag of interest, and k being the number of time delays.

More efficient calculations are carried out in the frequency domain using the fast Fourier transform algorithm. When there are exactly three variables, the complex bispectrum (Nikias and Mendel 1993) can be used. The improved efficiency is due to the fact that multiplication in the frequency domain is equivalent to convolution in the temporal domain.

RESULTS

Disambiguating several variables

Interdependency between movement features may result in ambiguity with respect to the encoded features. This is illustrated by simulating neural activity with a cosine relation to the direction of the acceleration vector, θ_{acc}, at a zero delay (see *Eq. 1*). In this case, the activity is also highly correlated with the direction of the velocity vector, θ_{vel}. This is because θ_{vel} and θ_{acc} are strongly correlated with each other (Fig. 1*B*). The ambiguity can be resolved by estimating the partial correlation between the firing rate and θ_{acc}, while accounting for correlations between firing rate and θ_{vel}, and between θ_{acc} and θ_{vel} (*Eq. 6*). Because variables may be correlated at nonzero delays, such partial correlations change as a function of the delay between the firing rate and velocity τ_1, and the delay between firing and acceleration τ_2. Thus the partial correlation between firing rate and one movement feature, given another, is displayed as a function of τ_1 and τ_2 (*Eq. 8*). This is designated henceforth as a PCCM. A PCCM is obtained from the data of a single trial (such as in Fig. 1*B*, *middle*) and the resulting PCCMs are averaged across trials.

When applying the PCCM method to the simulation described above we obtain two PCCMs (Fig. 2*A*). The *top matrix* shows the partial cross-correlation between firing rate and the cosine of θ_{vel} given the cosine of θ_{acc}, and the *bottom matrix* shows the partial cross-correlation between firing rate and the cosine of θ_{acc} given the cosine of θ_{vel}. These PCCMs reveal a strong correlation between firing rate and θ_{acc} at zero delay. This correlation is robust to the delay between the firing rate and the cosine of θ_{vel} and is expressed as a vertical correlation stripe in the θ_{acc} PCCM. Moreover, the correlation between firing rate and θ_{vel} was effectively eliminated. This concurs with our expectations since the firing rate was simulated using

a model of tuning to θ_{acc}. Note that some patches are seen in the θ_{vel} PCCM, and that the vertical correlation stripe in the θ_{acc} PCCM is not homogenous. These patterns are caused by nonlinear relations between θ_{vel} and θ_{acc} and result from the fact that we compute only linear correlations. They are most prominent at the time delay where linear correlations between the two features are maximal (about ± 150 ms, green curve in Fig. 1*B*). Yet this does not invalidate the conclusion that the activity of this putative neuron is related to θ_{acc} at a zero delay.

Having obtained several such PCCMs we wish to determine to which movement features the firing rate is related. To this end, we use a *t*-statistic, which is a measure of correlation that does not depend on the number of trials, and can be compared between different variables. The *t*-statistic is computed based on the average value across trials (Fig. 2*A*) and the SD across trials (Fig. 2*B*)

$$t_s(\tau_1, \tau_2) = \left| \frac{\langle \hat{\rho}(\tau_1, \tau_2) \rangle}{SD[\hat{\rho}(\tau_1, \tau_2)]/\sqrt{n}} \right| \qquad (15)$$

where *n* is the number of trials, and $\hat{\rho}(\tau_1, \tau_2)$ is a Fisher *z*-transformed CC at a pair of time delays. For CCs that are normally distributed this statistic is distributed according to the *t*-distribution with $n-1$ df, hence *P* values can readily be computed. As is, the statistic is a useful visualization tool (Fig. 2*C*).

Application to amplitude and direction of the velocity and acceleration vectors

In the preceding example, we considered two movement features: the cosine of θ_{vel} and the cosine of θ_{acc}. We now apply the PCCM method to an extended set of features, by considering the sine components of θ_{vel} and θ_{acc}, in addition to their cosines. Furthermore, we consider the amplitudes of these vectors, A_{vel} and A_{acc}. Specifically, we consider θ_{vel} and A_{vel} at one point in time relative to the firing rate, and θ_{acc} and A_{acc} at another point in time. After taking the sine and cosine of θ_{vel} and θ_{acc}, we obtain, together with A_{vel} and A_{acc}, a total of six movement features, and estimate a PCCM for each (APPENDIX). Figure 3*A* shows the six PCCMs for the same movement data and simulated neural activity that were used previously (Fig.
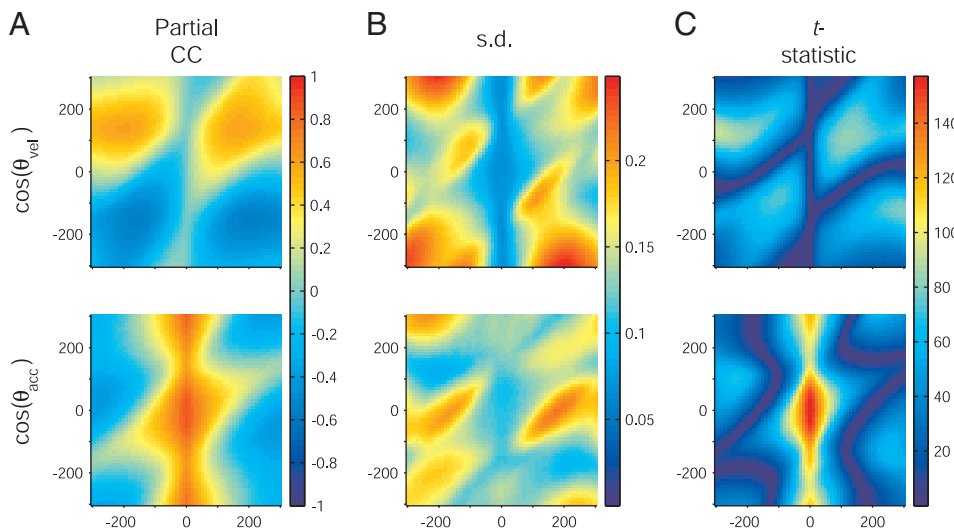

FIG. 2. Statistical analysis of partial cross-correlations. *A*: partial cross-correlation matrices (PCCMs) averaged across trials. The data used here are the same as in Fig. 1*B*. *Top*: PCCM for the firing rate and the cosine of θ_{vel}, given the cosine of θ_{acc}. *Bottom*: PCCM for firing rate and the cosine of θ_{acc}, given the cosine of θ_{vel}. As in all PCCMs displayed in this paper, horizontal axes measure the time delay between firing rate and acceleration whereas vertical axes show the delay between firing rate and velocity (ms). Firing rate precedes motion when delays are positive. Color scale indicates values of Fisher *z*-transformed partial CCs. Clearly, mean partial CCs are highest for the cosine of θ_{acc} at 0 delay, expressed as a vertical stripe in the *bottom panel*. *B*: SD matrices, computed based on the same data as in *A*. *C*: matrices expressed as *t*-statistic values can be compared across different features. In this example, the vertical stripe in the acceleration matrix is accentuated, underscoring the encoding of θ_{acc} by the putative neuron.
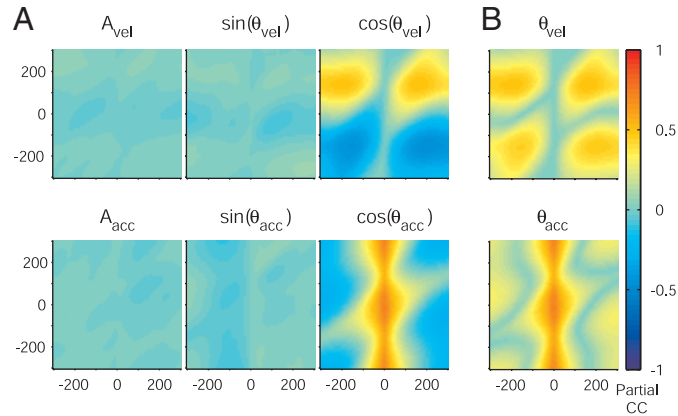
FIG. 3. PCCMs for multiple movement features. *A*: movement data and simulated neural activity used here are the same as in Figs. 1 and 2. In contrast to previous figures, here the analysis covers 6 movement features: A_{vel}, the sine of θ_{vel}, the cosine of θ_{vel}, A_{acc}, the sine of θ_{acc}, and the cosine of θ_{acc}. Thus each PCCM is computed between neural activity and 1 feature, given 5 other features. The name of the feature above each panel indicates that the partial correlation is between firing rate and this feature, given the other 5. Other conventions are the same as in Fig. 2. Obviously, the encoded feature is cosine of θ_{acc} at 0 delay. *B*: θ_{vel} sine and cosine PCCMs are combined (*Eq. 16*) yielding 1 PCCM for θ_{vel}. The same is done for acceleration, resulting in 1 PCCM for θ_{acc}.

2). Each of these PCCMs describes the correlation between firing rate and one of the six features, given the other five. To determine whether neural activity is related to θ_{vel} (or θ_{acc}) in general, the sine and cosine PCCMs, M_{sin} and M_{cos}, of θ_{vel} (or θ_{acc}) are combined, for each pair of time delays, by taking the square root of the sum of squares

$$M_\theta(\tau_1, \tau_2) = \sqrt{M_{sin}^2(\tau_1, \tau_2) + M_{cos}^2(\tau_1, \tau_2)} \qquad (16)$$

To illustrate this, we apply *Eq. 16* to the θ_{vel} sine and cosine PCCMs and then to the θ_{acc} sine and cosine PCCMs. Thus the four directional PCCMs are reduced to two (Fig. 3*B*). Each of these two matrices measures partial correlations between firing rate and θ_{vel} (or θ_{acc}), given A_{vel}, A_{acc}, and θ_{acc} (or θ_{vel}). Together with the two amplitude PCCMs (Fig. 3*A*, *leftmost panels*) we obtain a set of four matrices which gives a succinct characterization of the movement features encoded by the putative neuron.

Application of the PCCM method to neural activity simulated according to composite models

Up to this point, neural activity has been simulated according to a cosine model of directional tuning (*Eq. 1*). In this model, relationships between neural activity and the cosine of direction are fully captured by a product-moment CC. We now go one step further and test the capability of the method to resolve ambiguities in simulated neural activity based on real movement data and on more complex tuning models.

We use a composite model in which the firing rate is linearly related to movement features, A_{vel} and θ_{vel}, in an additive manner (*Eq. 2*). Based on the movement data and on the simulated firing rate profiles, we estimate four PCCMs (Fig. 4*A*). Indeed, these PCCMs show that the response of the putative neuron is related to A_{vel} and to θ_{vel} at a time delay of ~50 ms. As in the single feature simulations, some patches are seen in the acceleration PCCMs (Fig. 4*A*, *bottom*). These may result from nonlinear relations between velocity and accelera-
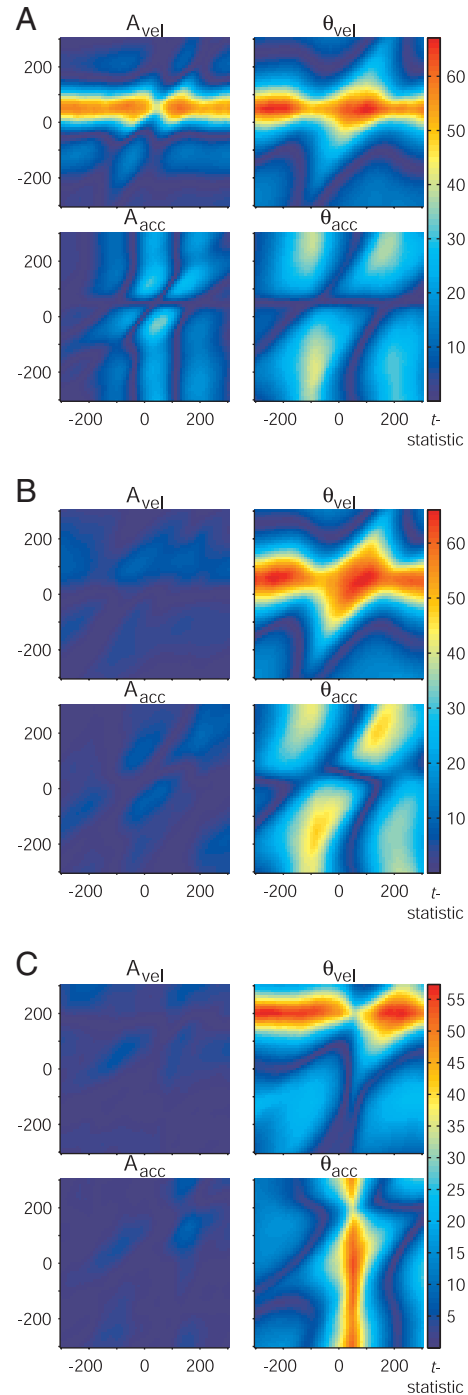

FIG. 4. Application of the PCCM method to neural data simulated according to composite models. *A*: cosine relation with θ_{vel} and linear relation with A_{vel} (*Eq. 2*; $\tau_{vel} = 50$ ms). The 4 PCCMs are organized as follows: *top*: velocity, *bottom*: acceleration; *left*: amplitude; *right*: direction. Colors depict *t*-statistic values. Each PCCM shows the correlation between firing rate and 1 movement feature, given the other features. Other conventions are the same as in Fig. 3. As expected from the model, horizontal stripes are evident in the A_{vel} and θ_{vel} PCCMs. *B*: cosine relation with θ_{vel} multiplied by A_{vel} (*Eq. 3*; $\tau_{vel} = 50$ ms). As expected, the amplitude relation is not noticeable in the PCCMs. The direction relation is obvious. *C*: cosine relation with θ_{vel} and cosine relation with θ_{acc} (*Eq. 4*). The delay between firing rate and velocity is 200 ms, and the delay between firing rate and acceleration is 50 ms. These features and delays are unmistakable in the PCCMs. All simulations in this figure were based on the same movement data (387 trials).

tion. Nonetheless, relationships of firing rate with acceleration are much smaller in magnitude than with velocity and indicate that the simulated neural activity is mostly related to A_{vel} and to θ_{vel} and not to acceleration.

The model of cosine tuning used in the preceding text for direction took into account the direction of the movement vector but not its amplitude. A more realistic model incorporating A_{vel} as a scaling factor for θ_{vel} (*Eq. 3*) is used next. In that model, rate depends on amplitude and on cosine of direction in a multiplicative manner. When directions are distributed homogenously between 0 and 360°, their cosine is distributed symmetrically around zero. In such a setting, the linear correlation between firing rate and amplitude is zero. In contrast, amplitudes are not symmetrically distributed, so the correlation between rate and direction cosine is not zero. Therefore we expect a clear relation with the direction and not with the amplitude. Indeed, when generating simulated firing rate profiles according to this model, the PCCM method reveals an exclusive relation with θ_{vel} (Fig. 4*B*).

Next, we use a composite model of tuning to θ_{vel} and θ_{acc} (*Eq. 4*). For both components, the preferred direction (PD) is the same, but the time delays are different: the simulated neural activity precedes velocity by 200 ms and acceleration by 50 ms. As in the previous cases, the PCCM method is successful in recovering the correct features and resolving ambiguity in their encoding (Fig. 4*C*). A noteworthy attribute in the resulting PCCMs, not seen in previous simulations, is that the two correlation stripes thin considerably at their graphical point of intersection, at delays of about {50, 200} ms. This arises from the fact that at this point in time, the simulated neural activity is an outcome of both features. Therefore computing partial correlations removes relations of the neural activity with both of them. It is clear, however, which features are encoded and at what time delays.

We tested the capability of the method to reveal encoded features by using models that include deviation from linear relations with direction cosine. By using a von Mises tuning function instead of a cosine, the concentration of a directional tuning curve was allowed to vary; this provides a better fit with firing rates of motor cortical neurons (Amirikian and Georgopoulos 2000). In addition, we used models of a nonlinear (sigmoidal) amplitude dependency, composite models of sigmoidal amplitude dependency and von-Mises directional rela-

tion, and non-Gaussian additive noise. The PCCM method exhibited remarkable robustness to the type of tuning function: the correct movement features were apparent in the resulting PCCMs for a wide range of simulation coefficients in linear and in nonlinear models alike.

Overall, these simulations demonstrate that the method is capable of revealing tuning to specific movement features despite ambiguities caused by delayed correlations between movement features.

Application of the PCCM method to real neural data

We applied the PCCM method to the activity of neurons recorded in Macaque motor cortex during continuous tracing movements. We present two typical examples. The first example (Fig. 5*A*) demonstrates a case of encoding a single movement feature, namely θ_{vel}. Only one matrix in the set of four PCCMs contains a high correlation stripe in striking similarity to the results obtained from the simulation of exclusive relationship to θ_{vel} (Fig. 4*B*). It is quite clear that this motor cortical neuron represents the direction of the velocity vector, regardless of other features.

A second example illustrates a case of simultaneous encoding of two movement features, A_{vel} and A_{acc} (Fig. 5*B*). There is a horizontal stripe in the A_{vel} PCCM at a time delay of approximately -25 ms, and there is a vertical stripe in the A_{acc} PCCM at the same time delay. This set of PCCMs resembles the two feature simulation results (Fig. 4*C*), the main difference being that instead of encoding directions, this neuron is related to the amplitudes of velocity and acceleration vectors.

Derivation of a preferred direction

Until now we have concentrated on the issues of which movement feature is encoded by neural activity, and at what time delay. However, there is often a need to determine the direction of the velocity—or the acceleration—vector preferred by a neuron; namely, the neuron's PD. Assume we have determined that a neuron is related to direction at a fixed time delay by observing a high correlation stripe. For instance, looking back at Fig. 5*A* we infer that the neuron is tuned to θ_{vel} due to the horizontal stripe in the θ_{vel} PCCM at a time delay of ~20 ms. To determine the velocity PD that is associated with this delay, we return to the separate sine and cosine PCCMs
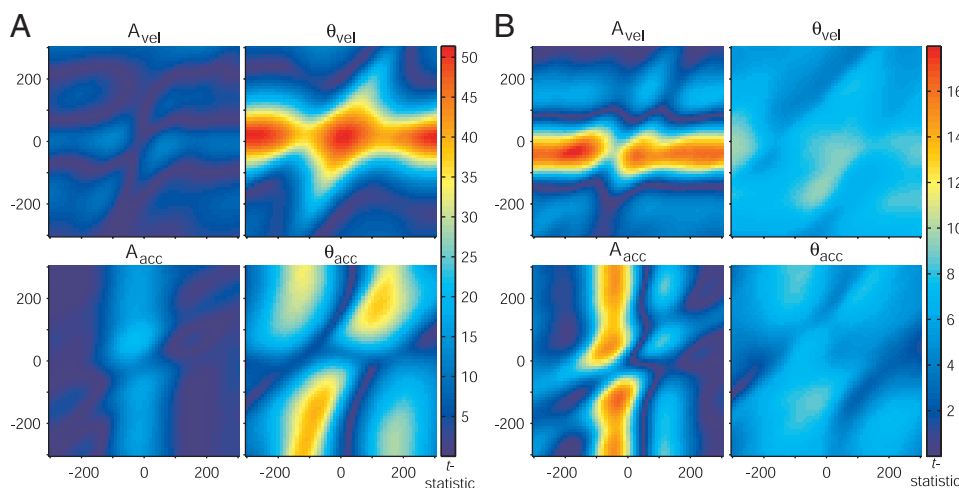

FIG. 5. Application of the PCCM method to real neural data. *A*: these 4 PCCMs were computed for a neuron recorded in the primary motor cortex during 250 continuous movement trials. The horizontal correlation stripe in the θ_{vel} PCCM shows the highest values in this set of PCCMs and is consistent with tuning to velocity direction at a time delay of 20 ms. Conventions are the same as in Fig. 4. *B*: PCCMs computed for another neuron, recorded in the primary motor cortex, based on 198 trials. The 2 correlation stripes in the A_{vel} and A_{acc} PCCMs have approximately the same magnitude and suggest a relation with both features. Time delays for both features are -25 ms, suggesting simultaneous representation of these features.

(Fig. 6*A*). A negative correlation stripe is evident in the sine PCCM and a positive correlation stripe is evident in the cosine PCCM, indicating that the PD of this neuron is between 270 and 360°. To better visualize this, we combine the sine and cosine PCCMs using the inverse tangent

$$M_{PD}(\tau_1, \tau_2) = \tan^{-1}\left(\frac{M_{\sin}(\tau_1, \tau_2)}{M_{\cos}(\tau_1, \tau_2)}\right) \qquad (17)$$

This equation gives the PD as a function of two time delays. Computed for multiple pairs of time delays, the latter equation yields a matrix of velocity PDs ranging from 0 to 360° (Fig. 6*B*). However, some of these values are meaningless because there is little correlation with θ_{vel} at the corresponding time delays. Thus we use the θ_{vel} PCCM expressed as *t*-statistic values (Fig. 5*A*, *top right*) to mask the irrelevant values in the velocity PD matrix (an arbitrary threshold of $1/\sqrt{2}$ of the maximal *t*-statistic was used for this example). At the relevant time delays (delineated between black lines in Fig. 6*B*), directionality values are quite homogenous with a mean ± SD of 303 ± 4°. We conclude that this neuron encodes θ_{vel} and that its preferred direction of the arm's movement is toward the monkey and slightly to the right. Note that because we estimated partial correlations, this PD is a θ_{vel} PD per se and does not result from a θ_{acc} PD and correlations between θ_{vel} and θ_{acc}. Moreover, the latter technique is not specific to a partial cross-correlation setting. It can be used to estimate PDs directly from any continuous movement data without discretization, assuming that the neural activity is related to a known feature at a known time delay.

DISCUSSION

The ideal experiment, designed with the purpose of relating neural activity to distinct stimulus or action variables, should contain no correlations—at any time delay—among the variables. For instance, pseudorandom binary sequences can be used to create visual stimuli in which individual pixels alternate between black and white in a memory-less and independent fashion (Reid et al. 1997). However, such stimuli tend to be far removed from natural settings, and their integration into experiments with the motor apparatus is difficult if at all possible. Here, we took a complementary approach and outlined a statistical method designed to resolve ambiguities in the encoding of multiple movement features, using a partial cross-correlation analysis. The method was successful in revealing not only which movement features were encoded in the neural activity but also at what time delays. It yielded consistent results under a wide range of simulation coefficients with linear and nonlinear models, simple and composite models, and simulated and real data.

An often-used method for quantifying interactions between neural activity and stimulus or action dimensions is multiple linear regression analysis (Draper and Smith 1981). This technique weights the impact that each of several variables has on the value of one variable such as firing rate. It has been used to estimate regression coefficients for several movement features lagged together relative to neural activity (Ashe and Georgopoulos 1994). The PCCM method extends the latter technique in two respects. First, it yields a graphical output measured on a scale that can be readily compared between different features, neurons, and experiments. Second, it directly accounts for effects that various variables may have on the relations between two other variables at different time delays.

Assumptions of the PCCM method

Several assumptions were made on developing the procedure outlined in this paper. First, when calculating the CC over all samples in a trial, we made the assumption of stationary relationships between the tested variables throughout the course of a trial. This assumption does not require each variable, for example firing rate, to be stationary by itself—a truly unrealistic requirement. This assumption can be overcome if the partial cross-correlation is estimated for many consecutive time windows within a given trial as was previously done in the framework of a linear regression (Fu et al. 1995). The time-resolved PCCMs computed in such time windows would take into account within-trial changes of delayed dependencies between features.

Second, in computing trial-averaged mean ± SD (*Eq. 15*), we actually assumed inter-trial stationarity. This assumption may be mitigated by employing a time-dependent mean and SD calculated, for instance, using a moving window over the sequence of trials.

Third, throughout this paper we estimated linear correlations between two variables measured on a linear scale. When variables have a monotonous, albeit a nonlinear relationship between them (for instance, a delta-shaped directional tuning function), the product-moment CC may estimate relations poorly. Thus when joint distributions of variables are far removed from joint normal distributions, a rank CC could be used instead. Additionally, in handling circular variables such as θ_{vel} and θ_{acc}, we used the sine and cosine components. A possible alternative is to use measures of circular-circular and/or linear-circular association directly for this type of data (Fisher 1993).

Finally, because we used correlations to measure dependencies, we focused on second-order, monotonous relationships between variables, ignoring higher-order and nonlinear relations. Mutual information may be used to estimate these kinds
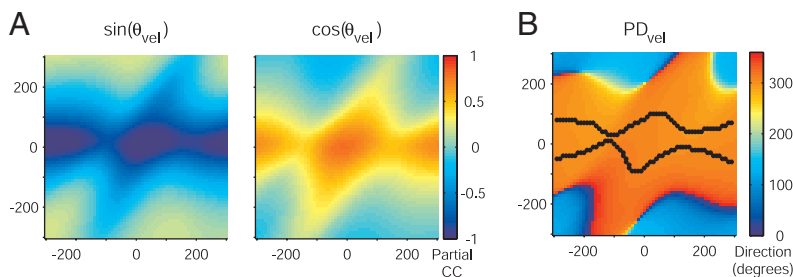
A

sin(θ_{vel})   cos(θ_{vel})

B

PD$_{vel}$

FIG. 6. Derivation of a preferred direction (PD) of a motor cortical neuron. *A*: PCCMs of θ_{vel} sine and cosine averaged across 250 trials (same neuron as in Fig. 5*A*). Color scale indicates partial CCs. There is a negative correlation with the sine of θ_{vel} at a time delay of ~20 ms. *B*: estimation of velocity PD from the sine and cosine PCCMs. Each pixel in the 2 PCCMs shown in *A* defines a planar vector: the angle of this vector is plotted for each pixel (*Eq. 17*). Based on the θ_{vel} *t*-statistic matrix (Fig. 5*A*, *top right*), a region of high correlation with θ_{vel} is delineated between the two black lines. The direction in this region is 303 ± 4° (mean ± SD).

of relationships. Such an approach was used to compute mutual information between position and velocity for a two-dimensional visual stimulus at zero delay (Adelman et al. 2003). This approach can be extended by estimating conditional mutual information (Cover and Thomas 1991) between neural activity and one lagged movement feature, given another lagged feature instead of partial CCs. Such an extension would potentially account for complex temporal dependencies between variables.

As shown, the above-discussed assumptions do not preclude an application of the method to experimental data. Moreover, useful insights are obtained from neural data that are inherently noisy.

Limitations of the PCCM method

There are several limitations of which we need to be aware. First, the method can only be used when the amount of data are sufficient. "Sufficiency" depends on various factors such as inter-trial variability and the SNR of the variables encoded in the neural activity, but some guidelines can be set. First, given experimental values, several dozen trials are required for the *t*-statistics to stabilize. The required number of trials does not increase exponentially with the number of variables used. This can be understood by considering a set of three variables: when estimating triplet-wise correlations among three variables, more data are required than when estimating first-order partial CCs because the latter depend only on three pair-wise CCs (*Eq. 6*). Thus instead of estimating the full three-way joint PDF, we need only estimate three marginal distributions of the full PDF, and this naturally requires less data than are required for estimating the full PDF. However, all variables must be collected at the same time as the neural data are. Second, trial length should be more than twice the maximal delay of interest to have some data left after the trimming process (METHODS).

Second, in the applications presented, we have restricted ourselves to three sets of variables: the firing rate, velocity, and acceleration. Different elements of each set, for instance, A_{vel}, the sine of θ_{vel}, and the cosine of θ_{vel}, were always lagged together. This facilitated a comprehensible graphic display, whereas accounting for a different time delay for many variables would have created a multi-dimensional hyper-cube. Thus a limitation of the graphic output of our method is that only two or three time delays can be easily displayed.

Third, the PCCM method will only disambiguate between the contributions of the specific features that were measured and included in the analysis. For instance, if the activity of some muscle is correlated with two of the measured features, correlations with two features will be observed. Moreover, the form of the tuning curve will affect the results. For instance, when using cosine and sine on simulated data where tuning was according to von Mises (Amirikian and Georgopoulos 2000), we got somewhat different *t*-statistic values, but the appropriate variables and delays were still clearly detected. It is possible that a completely different tuning curve (e.g., multiple directional peaks) would not be detected by the PCCM method. Thus the interpretational power of the method is limited by the nature and number of the included features.

Forth, when two variables are near duplicates one of another and both are related to a third variable *C*, the partial correlation between each of the two and *C*, given the other, will be about

zero, although the pair-wise cross-correlations are different from zero. This is largely a theoretical case since physiological data are noisy. Nonetheless, it is advisable to look at the distribution and the auto-correlation of each feature, as well as at joint distributions and pair-wise cross-correlations, together with PCCMs.

Finally, the linear version of the PCCM method, presented here, should not be used whenever a product-moment CC would not be used for characterizing pair-wise relations. Thus when the assumed generative model is nonadditive (e.g., a multiplicative relation between velocity and acceleration), when tuning is not monotonous (e.g., a Gaussian dependency on velocity amplitude), or when firing rates are highly non-Gaussian (e.g., very low), the method as is should not be used. Yet, a rank-based or an information-theoretic version of the method could be used instead as mentioned in the preceding text.

Interpretation of PCCMs

The PCCM graphs provide a rich way of visualizing partial correlations. In general, we look for graphical patterns (clusters or high valued pixels) rather than isolated high valued pixels. Several graphical patterns may be observed in a PCCM. If a single, prominent stripe is visible, then the neural activity is most likely to be related to the corresponding feature at a particular time delay, insensitive to another feature at all delays (for instance, Figs. 4 and 5). When one stripe is accompanied by another stripe, in another PCCM, then neural activity is probably related to two features, at distinct time delays (Figs. 4, *A* and *C*, and 5*B*). Other graphical characteristics are also possible. For instance, when neural activity is related to a specific temporal relation between two movement features, a "blob" may be observed, and when the neural activity is related to the co-occurrence of two features, at any time delay, a diagonal stripe may be seen.

In our experience, the most commonly observed graphical pattern in PCCM graphs is a horizontal or vertical stripe, which is compatible with encoding of some feature at a constant delay. It may thus be desired to reduce the dimensionality of a PCCM by searching for significant stripes. A simple procedure for doing so would be to define a stripe as a row (or column) of pixels in which most pixels have high values, both absolutely (indicating significant partial CCs) and relative to the maximal *t*-statistic value in a set of PCCMs (indicating a stripe of high partial CCs relative to its surroundings). A stripe corresponding to this definition would have a single time delay and indicate a relation to a single feature.

Applications and possible extensions

In this paper, we illustrated the PCCM method using movements from one type of experiments. The method is quite general and can be used in other applications. First, the procedure can be successfully applied to movements obtained from center-out experiments assuming minimal data size requirements are met (movement duration long enough to permit trimming and enough trials per direction; from our experience, 0.5 s/trial and 30 trials/direction are sufficient). Thus the method can be applied to a large corpus of existing data. Second, it is perhaps needless to point out that correlations

among any kinematical movement features (for example, position and its time derivatives), dynamical movement features (for example, muscular activity), nonmovement features (for example, visual stimuli, contrast and brightness, natural auditory sounds, natural visual scenes), or any combination thereof, can be handled in the manner presented here.

The PCCM method can be applied to other data types. For instance, the firing rate of a single neuron can be replaced with a series of spike times. This would generate a more detailed temporal relation between neural activity and movement features, akin to spike-triggered averages (De Ruyter van Steveninck and Bialek 1988). Alternatively, movement features can be replaced with local field potential signals recorded by two electrodes, whereas a third variable is the spike trains recorded by one of these electrodes (Halliday et al. 1995). As a third example, if one variable is the spike trains of one neuron and another variable is the spike trains of a second neuron, the PCCM method can be used to remove effects of multiple other variables on the functional connectivity between these two neurons (Aertsen et al. 1989; Ben-Shaul et al. 2001).

Finally, the PCCM method can be used to determine which movement features are encoded by neurons. The outcome may be used to guide selection of features for improved movement reconstruction and ultimately—for movement construction.

APPENDIX

Numerical evaluation of partial CC

High-order partial CCs may be evaluated directly by recursion (Dubois 1957) using the following MATLAB code

```
functiony = reduce(X)
n = size(X, 1);            % matrix size
d = ones(n − 1, 1);        % dummy
b = X(2: end, 1);          % before reduction
E = X(2: end, 2: end);     % before reduction
C = X(1, 2: end);          % variable to be partialed
y = E − (b*d′) · *(d*C);   % reduce
y(2: end, 1) = y(1, 2: end)′/y(1, 1);   % after reduction
```

For example, to evaluate *Eq. 9* at (τ_1, τ_2), X would be—at the first iteration—a symmetric 7×7 matrix of product-moment pair-wise CCs. The last column (row) in the correlation matrix X consists of CCs between R and each of the other variables, lagged appropriately (METHODS). The second to last column (row) consists of CCs between A_{vel} and all other variables. The routine is then called recursively five times, resulting in a 2×2 partial covariance matrix. From this matrix the fifth-order partial CC is computed directly.

Alternatively, partial CCs may be obtained from MLR. Denote the data matrix of n observations of m standardized features by X, and the firing rate by the $n \times 1$ vector y. The MLR solution is then an $m \times 1$ vector, $b = (X^TX)^{-1}X^Ty$, where X^T indicates matrix transpose and

X^{-1} indicates matrix inversion (Draper and Smith 1981). The asymptotic SD of each element of b, σ_i, $i = 1 \ldots m$, is obtained by multiplying the corresponding element of the variability vector [the square root of the main diagonal elements of $(X^TX)^{-1}$] by the root-mean-square error (the *L2* norm of the prediction error, $y - Xb$). Next, a *t*-statistic is computed for each regression coefficient (element of b), $t_1 = b_i/(\sigma_i \cdot \sqrt{\text{DOF}})$, where the regression degrees-of-freedom are DOF $= n - m$. Finally, the corresponding $(m-1)^{\text{th}}$-order partial CC is obtained by $t_i/\sqrt{t_i^2 + \text{DOF}}$. Thus partial CCs are monotonically related to the suitable MLR coefficients.

REFERENCES

Adelman TL, Bialek W, and Olberg RM. The information content of receptive fields. *Neuron* 40: 823–833, 2003.

Aertsen AM, Gerstein GL, Habib MK, and Palm G. Dynamics of neuronal firing correlation: modulation of "effective connectivity". *J Neurophysiol* 61: 900–917, 1989.

Amirikian B and Georgopoulos AP. Directional tuning profiles of motor cortical cells. *Neurosci Res* 36: 73–79, 2000.

Ashe J and Georgopoulos AP. Movement parameters and neural activity in motor cortex and area 5. *Cereb Cortex* 4: 590–600, 1994.

Ben-Shaul Y, Bergman H, Ritov Y, and Abeles M. Trial to trial variability in either stimulus or action causes apparent correlation and synchrony in neuronal activity. *J Neurosci Methods* 111: 99–110, 2001.

Cover TM and Thomas JA. *Elements of Information Theory.* New York: Wiley, 1991.

De Ruyter van Steveninck R and Bialek W. Real time performance of a movement-sensitive neuron in the blowfly visual system: coding and information transfer in short spike sequences. *Proc R Soc Lond B Biol Sci* 234: 379–414, 1988.

Draper NR and Smith H. *Applied Regression Analysis.* New York: Wiley, 1981.

Dubois PH. *Multivariate Correlational Analysis.* New York: Harper and Brothers, 1957.

Fisher NI. *Statistical Analysis of Circular Data.* Cambridge, UK: Cambridge Univ. Press, 1993.

Fisher RA. *Statistical Methods for Research Workers.* London: Oliver and Boyd, 1925.

Flament D and Hore J. Relations of motor cortex neural discharge to kinematics of passive and active elbow movements in the monkey. *J Neurophysiol* 60: 1268–1284, 1988.

Fu QG, Flament D, Coltz JD, and Ebner TJ. Temporal encoding of movement kinematics in the discharge of primate primary motor and premotor neurons. *J Neurophysiol* 73: 836–854, 1995.

Georgopoulos AP, Kalaska JF, Caminiti R, and Massey JT. On the relations between the direction of two-dimensional arm movements and cell discharge in primate motor cortex. *J Neurosci* 2: 1527–1537, 1982.

Halliday DM, Rosenberg JR, Amjad AM, Breeze P, Conway BA, and Farmer SF. A framework for the analysis of mixed time series/point process data–theory and application to the study of physiological tremor, single motor unit discharges and electromyograms. *Prog Biophys Mol Biol* 64: 237–278, 1995.

Hubel DH and Wiesel TN. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *J Physiol* 160: 106–154, 1962.

Kakei S, Hoffman DS, and Strick PL. Muscle and movement representations in the primary motor cortex. *Science* 285: 2136–2139, 1999.

Merzenich MM and Brugge JF. Representation of the cochlear partition of the superior temporal plane of the macaque monkey. *Brain Res* 50: 275–296, 1973.

Moran DW and Schwartz AB. Motor cortical representation of speed and direction during reaching. *J Neurophysiol* 82: 2676–2692, 1999a.

Moran DW and Schwartz AB. Motor cortical activity during drawing movements: population representation during spiral tracing. *J Neurophysiol* 82: 2693–2704, 1999b.

Mountcastle VB. Modality and topographic properties of single neurons of cat's somatic sensory cortex. *J Neurophysiol* 20: 408–434, 1957.

Nikias CL and Mendel JM. Signal processing with higher order spectra. *IEEE Signal Process Mag* 10: 10–36, 1993.

Paninski L, Fellows MR, Hatsopoulos NG, and Donoghue JP. Spatiotemporal tuning of motor cortical neurons for hand position and velocity. *J Neurophysiol* 91: 515–532, 2004.

Reid RC, Victor JD, and Shapley RM. The use of m-sequences in the analysis of visual neurons: linear receptive field properties. *Vis Neurosci* 14: 1015–1027, 1997.

Scott SH and Kalaska JF. Reaching movements with similar hand paths but different arm orientations. I. Activity of individual cells in motor cortex. *J Neurophysiol* 77: 826–852, 1997.

Todorov E. Direct cortical control of muscle activation in voluntary arm movements: a model. *Nat Neurosci* 3: 391–298, 2000.