Chinese Journal of Physiology 53(6): 417-422, 2010
DOI: 10.4077/CJP.2010.AMM033

Accuracy in Estimating Motor Commands Using Neuronal Population Coding

Shigefumi Hata

Department of Physics, Kyoto University, Kyoto 606-8502, Japan

Abstract

We consider the problem of estimating motor commands from an ensemble of neuronal activities. The population vector algorithm proposed by Georgopoulos provides largely biased estimations when preferred directions of neurons are non-uniformly distributed. To improve this, various decoding methods have been proposed. However, dependence of decoding accuracy on the motor command and other features of neural activities, such as baseline firing rates or amplitudes of tuning curves, have not been quantitatively discussed. In this study, we propose a new method to estimate the motor command in the maximum likelihood estimation framework, which is analytically tractable. We find that the estimation accuracy is independent of the motor command. Using our estimation method, we can estimate the motor command with equal accuracy in all directions.

Key Words: population coding, population vector, optimal linear estimator, probability density estimation

Introduction

Recent developments in experimental technology allow us to record neural activities from ensembles of motor cortical neurons simultaneously. The activities can be used as input signals to brain-machine interface (BMI) devices such as a robotic arm or a cursor on a computer screen (9, 10). For constructing a BMI, we need an appropriate decoding method.

In the pioneering work by Georgopoulos *et al*. (4), direction of the arm movement was estimated from neural activities recorded from monkey motor cortex with a method called as the population vector algorithm (PVA). However, it has been revealed that the PVA gives largely biased estimations when preferred directions of the neurons are non-uniformly distributed (6). To improve this, Salinas and Abbott proposed the optimal linear estimator (OLE) (7), and Sanger proposed the probability density estimation (PDE) (8). However, dependence of the decoding accuracy on the motor command and other features of neural activities, such as baseline firing rates or amplitudes of tuning curves, have not been quantitatively analyzed.

In this study, we use maximum likelihood estimation, one of the PDE algorithms, to estimate the motor command. We introduce an encoding model to analytically calculate the motor command and its confidence limit. Based on the analytical solution and numerical simulations, we examine the effects of the motor command on the estimation accuracy.

Methods

In this section, we introduce our method to estimate the motor command. The set includes (i) an encoding model that describes how neurons code the motor command into their spike counts, and (ii) a decoding algorithm to estimate the motor command from the observed spike counts. We introduce a set of encoding model and decoding algorithm that allows analytic calculation of the estimate of the motor command and its confidence limit.

Encoding Model

In this study, the encoding model consists of a

Corresponding author: Dr. Shigefumi Hata, Department of Physics, Kyoto University, Kyoto 606-8502, Japan. Tel: +81-75-753-3700, Fax: +81-75-753-3819, E-mail: hata@ton.scphys.kyoto-u.ac.jp
Received: December 11, 2009; Revised: February 8, 2010; Accepted: February 29, 2010.

tuning function and a spike count distribution, which represent the mean and the fluctuations of the spike counts, respectively.

1. Tuning Function

We assume that, given a motor command $\vec{v} = (\theta, v)$, mean spike count λ_i of neuron i is determined by

$$\lambda_i(\vec{v}) = \{a_i v\cos(\theta - \phi_i) + b_i\}^2 \quad [1]$$

$$= \{a_i(v_x x_i^{\mathrm{p}} + v_y y_i^{\mathrm{p}}) + b_i\}^2, \quad [2]$$

where ϕ_i is the preferred direction of neuron i, and we have defined $x^{\mathrm{p}}{}_i := \cos\phi_i$, $y^{\mathrm{p}}{}_i := \sin\phi_i$, $v_x = v\cos\theta$ and $v_y = v\sin\theta$. In this model, baseline firing rate and amplitude are given by $(-a_i + b_i)^2$ and $(a_i + b_i)^2 - (-a_i + b_i)^2 = 4a_i b_i$, respectively. We assume that the parameter a_i and b_i satisfy $a_i(v_x x_i^{\mathrm{p}} + v_y y_i^{\mathrm{p}}) + b_i \geq 0$. This condition makes the tuning function unimodal, which means that each neuron has only one preferred direction.

2. Spike Count Distribution

We assume that, given the motor command $\vec{v} = (v_x, v_y)$, the spike count n_i of neuron i is distributed around $\lambda_i(v_x, v_y)$ as follows:

$$p(n_i \,|v_x, v_y)$$

$$= \frac{1}{\sqrt{2\pi n_i}} \exp\left[-2\left(\sqrt{n_i} - \sqrt{\lambda_i(v_x, v_y)}\right)^2\right]. \quad [3]$$

We call this distribution a pseudo-Poisson distribution. As shown in Fig. 1, this distribution is similar to the Poisson distribution. The simplest spike-generating model gives spike counts that obey the Poisson distribution (3). Thus, the pseudo-Poisson distribution is plausible for representing the spike count distribution.

It is noteworthy that the pseudo-Poisson distribution was obtained by the square transformation, an inverse transformation of the square-root transformation (2), of a Gaussian distribution. In statistics, square-root transformations are used for Poisson-distributed data to obtain Gaussian-like distributions, which allows us to apply parametric tests to the Poisson-distributed data sets. Conversely, here we used the square transformation to obtain the Poisson-like distribution from the Gaussian-distribution.

Decoding Algorithm

To estimate the motor command $\vec{v}$, we use

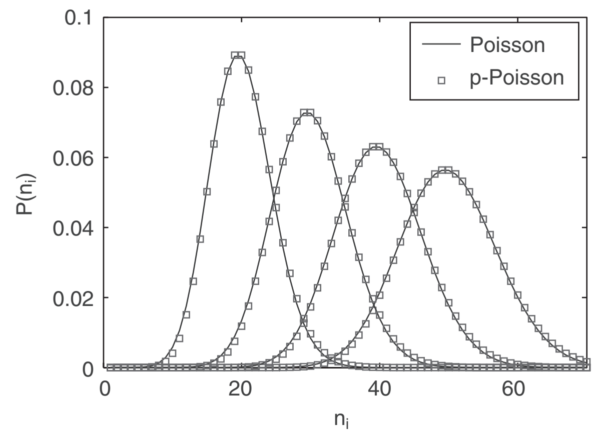

Fig. 1. Poisson (solid lines) and pseudo-Poisson (squares) distributions. Mean values of the four distributions are 20, 30, 40, and 50.

maximum likelihood estimation, a PDE technique, defined as

$$\hat{\vec{v}} = \underset{\vec{v}}{\mathrm{argmax}}\; p(\{n_i\}|\vec{v}). \quad [4]$$

Namely, the estimate $\hat{\vec{v}}$ is chosen to be the most probable $\vec{v}$ for causing a set of spike counts $\{n_i\}$.

Calculation Results

In this section, we calculate the estimate of the motor command and its confidence limit in our method.

Estimate

We assume that the spike counts $\{n_i\}$ of neurons are independent of each other:

$$p(\{n_i\}|\vec{v}) = \prod_{i=1}^{N} p(n_i|\vec{v}). \quad [5]$$

Under this assumption, we can calculate the maximum likelihood estimate $\hat{\vec{v}}$ of the motor command $\vec{v}$ as

$$\hat{\vec{v}} = \underset{\vec{v}}{\mathrm{argmax}} \prod_{i=1}^{N} p(n_i|\vec{v}) \quad [6]$$

$$\Rightarrow \begin{cases} \left| \dfrac{\partial}{\partial v_x} \displaystyle\prod_{i=1}^{N} p(n_i|\vec{v}) \right|_{v_x = \hat{v}_x} = 0 & [7] \\ \left| \dfrac{\partial}{\partial v_y} \displaystyle\prod_{i=1}^{N} p(n_i|\vec{v}) \right|_{v_y = \hat{v}_y} = 0 & [8] \end{cases}$$

$$\Rightarrow \begin{cases} \hat{v}_x = \dfrac{\Sigma_i a_i(\sqrt{n_i} - b_i)(S_{yy}x^{\mathrm{p}}{}_i - S_{xy}y^{\mathrm{p}}{}_i)}{S_{xx}S_{yy} - S_{xy}{}^2} & [9] \\ \hat{v}_y = \dfrac{\Sigma_i a_i(\sqrt{n_i} - b_i)(S_{xx}y^{\mathrm{p}}{}_i - S_{xy}x^{\mathrm{p}}{}_i)}{S_{xx}S_{yy} - S_{xy}{}^2}, & [10] \end{cases}$$

where we have introduced $S_{xx} := \Sigma_i a_i{}^2 x^{\mathrm{p}}{}_i{}^2$, $S_{yy} := \Sigma_i a_i{}^2 x^{\mathrm{p}}{}_i{}^2$ and $S_{xy} := \Sigma_i a_i{}^2 x^{\mathrm{p}}{}_i y^{\mathrm{p}}{}_i$.

Confidence Limit

Next we calculate the confidence limit of the estimate. We can show that the confidence limit has an ellipsoidal shape as follows:

$$\prod_{i=1}^{N} p(n_i|\vec{v}) = (2\pi)^{-\frac{N}{2}} \exp\left[-2\sum_{i=1}^{N}\{\sqrt{n_i} - a_i(v_x x_i^{\mathrm{p}} + v_y y_i^{\mathrm{p}}) - b_i\}^2\right] \prod_{i=1}^{N} n_i^{-\frac{1}{2}} \quad [11]$$

$$\propto \frac{1}{2\pi\sigma_1\sigma_2}\exp\left[-\frac{v_1{}^2}{2\sigma_1{}^2} - \frac{v_2{}^2}{2\sigma_2{}^2}\right], \quad [12]$$

where

$$\begin{cases} v_1 = \dfrac{\left\{S_{xx} - S_{yy} - \sqrt{(S_{xx} - S_{yy})^2 + 4S_{xy}{}^2}\right\}(v_x - \hat{v}_x) + 2S_{xy}(v_y - \hat{v}_y)}{2\left\{(S_{xx} - S_{yy})^2 + 8S_{xy}{}^2 - 2(S_{xx} - S_{yy})\sqrt{(S_{xx} - S_{yy})^2 + 4S_{xy}{}^2}\right\}^{1/2}} & [13] \\ v_2 = \dfrac{2S_{xy}(v_x - \hat{v}_x) - \left\{S_{xx} - S_{yy} - \sqrt{(S_{xx} - S_{yy})^2 + 4S_{xy}{}^2}\right\}(v_y - \hat{v}_y)}{2\left\{(S_{xx} - S_{yy})^2 + 8S_{xy}{}^2 - 2(S_{xx} - S_{yy})\sqrt{(S_{xx} - S_{yy})^2 + 4S_{xy}{}^2}\right\}^{1/2}}, & [14] \end{cases}$$

$$\begin{cases} \sigma_1{}^2 = 2\left(\sum_i a_i{}^2 - \sqrt{(S_{xx} - S_{yy})^2 + 4S_{xy}{}^2}\right)^{-1} & [15] \\ \sigma_2{}^2 = 2\left(\sum_i a_i{}^2 + \sqrt{(S_{xx} - S_{yy})^2 + 4S_{xy}{}^2}\right)^{-1}. & [16] \end{cases}$$

Considering that $p(\{n_i\}|\vec{v})$ takes the same value on the confidence limit, the confidence limit is an ellipse whose center is $(\hat{v}_x, \hat{v}_y)$, ratio of the lengths of major and minor axes is

$$\frac{\sigma_1}{\sigma_2} = \sqrt{\frac{\Sigma_i a_i^2 + \sqrt{(S_{xx} - S_{yy})^2 + 4S_{xy}{}^2}}{\Sigma_i a_i^2 - \sqrt{(S_{xx} - S_{yy})^2 + 4S_{xy}{}^2}}}, \quad [17]$$

and the slope of the long axis is

$$l = -\frac{S_{xx} - S_{yy} - \sqrt{(S_{xx} - S_{yy})^2 + 4S_{xy}{}^2}}{2S_{xy}}. \quad [18]$$

If the shape of the $100 \times (1 - \alpha)\%$ confidence limit is an ellipse with the length of its major and minor axes equal to $R\sigma_1$ and $R\sigma_2$, respectively, then we obtain

$$\int_{\left(\frac{v_1}{\sigma_1}\right)^2 + \left(\frac{v_2}{\sigma_2}\right)^2 \le R^2} dv_1 dv_2 \frac{1}{2\pi\sigma_1\sigma_2}\exp\left[-\frac{v_1{}^2}{2\sigma_1{}^2} - \frac{v_2{}^2}{2\sigma_2{}^2}\right]$$

$$= 1 - \exp\left[-\frac{R^2}{2}\right] \quad [19]$$

$$= 1 - \alpha \quad [20]$$

$$\Rightarrow R = \sqrt{-2\ln\alpha}. \quad [21]$$

Equations [17], [18], and [21] indicate that the shape of the confidence limit, or the set of parameters σ_1, σ_2 and R, depends only on $\{\phi_i\}$, $\{a_i\}$ and α. This means that estimation accuracy is independent of the spike count $\{n_i\}$ and of the motor command $\vec{v}$. Thus, using our estimation method, we can estimate any motor command with equal accuracy.

Comparison to Other Estimation Methods

In this section, we numerically simulate the estimation process of motor commands from randomly generated spike counts. We compare the estimation accuracy of our method to those of other estimation methods which we introduced in introduction. The simulation scheme is as follows: We first generate a set of spike counts n_i of each neuron i with the tuning curve $\tilde{\lambda}_i(\vec{v}) = \exp[e_i\cos(\theta - \phi_i) + f_i] + g_i$, which is called "von Mises tuning function" in (1), and Poisson spike count distribution $p(n_i|\vec{v}) = \tilde{\lambda}_i(\vec{v})\exp[-\tilde{\lambda}_i(\vec{v})](n_i!)^{-1}$. Then, using each estimation method, we calculate the estimation error $|\vec{v} - \hat{\vec{v}}|$ of the motor command esti-

Table 1. Summary of the estimation methods used in the simulation

Method	Tuning model	Distribution	Decoding algorithm
Method A (Our method)	$\{a_i v\cos(\theta - \phi_i) + b_i\}^2$	pseudo-Poisson	maximum likelihood estimation
Method B (MLE)	$\{a_i v\cos(\theta - \phi_i) + b_i\}^2$	Poisson	maximum likelihood estimation
Method C (OLE)	$\{a_i v\cos(\theta - \phi_i) + b_i\}^2$	Poisson	optimal linear estimation
Method D (PVA)	$c_i v\cos(\theta - \phi_i) + d_i$		population vector algorithm

mated from the set of spike counts, where $\vec{v}$ is the true motor command and $\hat{\vec{v}}$ is the estimated command.

To estimate the motor command, we use four types of estimation method: our method, MLE, OLE and PVA, summarized in Table 1. In MLE and OLE, we use the same tuning function $\lambda_i(\vec{v}) = \{a_i v\cos(\theta - \phi_i) + b_i\}^2$ as that in our method. The parameters a_i, b_i, c_i and d_i are determined to fit each tuning model to the tuning curve $\tilde{\lambda}_i(\vec{v})$ that we use in generating spikes.

Note that when we estimate motor commands from real neuronal population codes, we cannot know the tuning curves of real neurons. Thus, we generally have to use tuning models that differ from the tuning curves of real neurons. Therefore, we assume the tuning model $\lambda_i(\vec{v})$ of each method to be different from the tuning curve $\tilde{\lambda}_i(\vec{v})$ that used to generate spikes.

Mean Estimation Error

Fig. 2 shows dependence of the mean estimation error on the number of neurons N. We assume uniform [Fig. 2(a)] and nonuniform [Fig. 2(b)] distributions of the preferred directions $\{\phi_i\}$ in the numerical simulation. In both cases, PVA gives the poorest estimation. Our method and MLE have almost the same accuracy. OLE has low accuracy for small number of neurons N. However, with increasing N its estimation error decreases and for large N it gives better estimation than our method.

Dependence of Estimation Accuracy on the Motor Command

In section 3, we obtained the estimation accuracy of our method which was determined by the parameters $\{a_i\}$ and $\{\phi_i\}$ and was independent of the motor command $\vec{v}$. To confirm this, we calculate estimation errors for eight-directional motor commands with fixed $\{a_i\}$ and $\{\phi_i\}$. In this analysis, we use nonuniformly distributed $\{\phi_i\}$ to clarify the dependence on the direction of the motor command.

Fig. 3(a) shows the set of preferred directions $\{\phi_i\}$ that we used in the simulation and (b) shows dependence of the estimation error on the direction θ of the motor command. We can see that the errors in

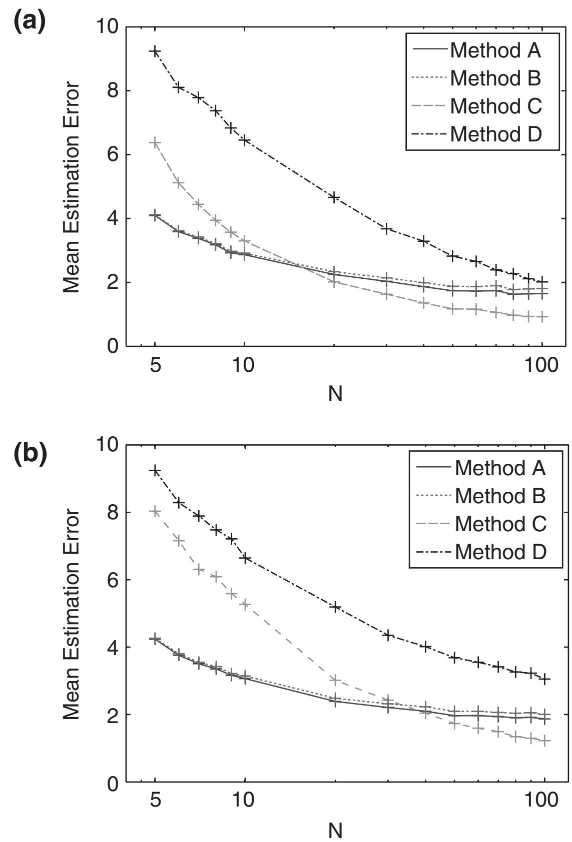

Fig. 2. Dependence of the estimation error $|\hat{\vec{v}} - \vec{v}|$ on the number of neurons N, where $\vec{v}$ is the true motor command and $\hat{\vec{v}}$ is the estimated motor command, for methods A (solid line), B (dotted line), C (dashed line), and D (dot-dashed line). Each data point is calculated from 1000 trials. The set of preferred directions $\{\phi_i\}$ is common to each data point. The true motor command $\vec{v}$ is changed trial by trial. (a) Mean estimation error. Preferred directions $\{\phi_i\}$ are uniformly distributed. (b) Mean estimation error. Preferred directions $\{\phi_i\}$ obey the Gaussian distribution with mean $3\pi/4$ and variance $\pi/2$.

PVA depends on the direction θ more strongly than other three methods. To quantify this θ-dependence, we use one-way ANOVA (see details in Ref. No. 5) to calculate the F value shown in the top right corner of each panel. Among the four methods, our method gives the smallest F value and OLE gives nearly the

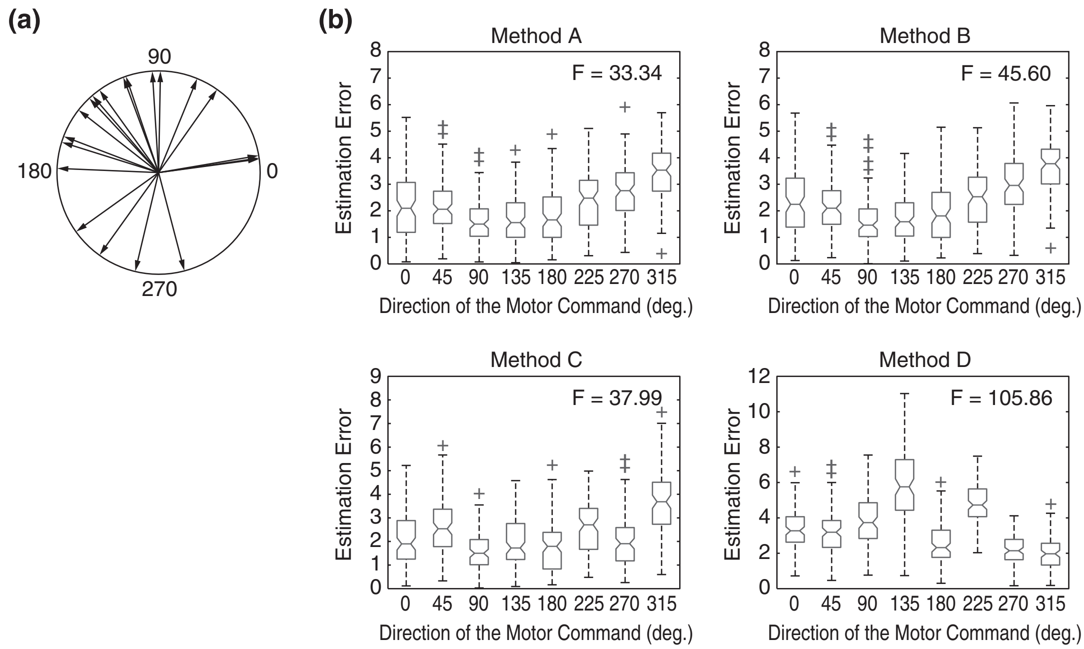

Fig. 3. Dependence of the estimation error $|\vec{\hat{v}} - \vec{v}|$ on the direction θ of the motor command $\{\vec{v}\}$ in the case that the set of preferred directions are biased. (a) Preferred directions $\{\phi_i\}$ which obey the Gaussian distribution with mean $3\pi/4$ and variance $\pi/2$. (b) Estimation error $|\vec{\hat{v}} - \vec{v}|$ depending on the direction θ. Each column is calculated from 100 trials. In each column, the box represents lower quartile, median, and upper quartile values. Dashed lines show the extent of the rest of the data. Notches on each box represent the uncertainty about the medians for box-to-box comparison. Boxes whose notches do not overlap indicate that the medians of the two groups differ at 5% significance level.

same value. This indicates that the estimation error in our method and in OLE have weak dependences on the direction θ of the motor command $\vec{v}$.

Discussion

Relation to PVA

Our estimation method is closely related to PVA. When the set of preferred directions $\{\phi_i\}$ is uniformly distributed and all the parameters $\{a_i\}$ take the same value a, we can simplify our estimation result for the motor command as follows:

$$S_{xx} + S_{yy} = {a_i}^2 \sum_{i=1}^{N} ({x^{\mathrm{p}}_i}^2 + {y^{\mathrm{p}}_i}^2) = {a_i}^2 N \qquad [22]$$

$$S_{xx} - S_{yy} = {a_i}^2 \sum_{i=1}^{N} ({x^{\mathrm{p}}_i}^2 + {y^{\mathrm{p}}_i}^2) = \sum_{i=1}^{N} \cos 2\phi_i \sim 0 \qquad [23]$$

$$\Rightarrow S_{xx} = S_{yy} = {a_i}^2 \frac{N}{2}\,. \qquad [24]$$

$$S_{xy} = a^2 \sum_i \sin 2\phi_i \sim 0\,. \qquad [25]$$

Thus, we obtain

$$\vec{\hat{v}} = \left(\frac{1}{a} \frac{\Sigma_i(\sqrt{n_i} - b_i)x^{\mathrm{p}}_i}{\Sigma_i {x^{\mathrm{p}}_i}^2},\ \frac{1}{a} \frac{\Sigma_i(\sqrt{n_i} - b_i)y^{\mathrm{p}}_i}{\Sigma_i {y^{\mathrm{p}}_i}^2} \right) \qquad [26]$$

$$= \frac{2}{N} \sum_{i=1}^{N} \frac{\sqrt{n_i} - b_i}{a} (\cos\phi_i, \sin\phi_i)\,. \qquad [27]$$

In Georgopoulos's PVA, it is assumed that the mean firing rate of neuron i obeys $\lambda_i^{\mathrm{PV}}(\vec{v}) = c_i v \cos(\theta - \phi_i) + d_i$, which leads to the estimate

$$\vec{\hat{v}}_{PV} = \frac{2}{N} \sum_{i=1}^{N} \frac{n_i - d_i}{c_i} (\cos\phi_i, \sin\phi_i)\,. \qquad [28]$$

Therefore, our estimate has the same form as that of PVA except for the root sign over n_i. Thus, our method is considered as a generalization of PVA that allows us to estimate the motor command more correctly even if the preferred directions are non-uniformly distributed and the amplitudes differ for each neuron.

Strength and Limitation

PVA gives good estimation when a set of preferred directions is uniformly distributed and the number of neurons is large. However, in accord with the previous studies, PVA gives poor estimation when the preferred directions are non-uniformly distributed. In such cases, our method, MLE or OLE gives better estimate than PVA. For large number of neurons N, OLE is better than our method and MLE. On the other hand, for small N, we can use our method or MLE to obtain better estimation than OLE. In particular, with our method, we can obtain the estimate whose accuracy is less affected by the direction of the motor command. In addition, it is also notable that calculation cost of our method is low. If we use the maximum likelihood algorithm as a decoding method, we generally need numerical calculations to obtain the estimate, and therefore its cost is much higher than other methods. In our method, however, we have an encoding model [2] and [3], which allows us to calculate the estimate analytically in MLE framework.

However, our method also has some limitations. For the purpose of calculating the estimate analytically, we introduced the tuning function [2] and the spike count distribution [3]. Poisson-like spike count distribution will be a plausible assumption in most cases, however, the tuning function [2] that we assumed may not always be plausible. In such a case, our method may give poor estimation and we need to introduce more plausible tuning functions. Thus, our method can be applied to the case that the tuning function [2] fits to real neurons well.

Summary

We addressed Georgopoulos's problem regarding estimation of the motor command. In the maximum likelihood estimation framework, we introduced Poisson-like spike count distributions and a specific tuning function model to analytically calculate the estimate and its accuracy. According to the analytical solution, the estimation accuracy of our method is independent of the motor command. This result is also supported by numerical simulations. Thus, using our estimation method, we will be able to estimate the motor command with equal accuracy in all directions.

Acknowledgments

I am grateful to Prof. Shinomoto for the intensive discussion. I also appreciate illuminating comments of Dr. Kobayashi and Dr. Shimokawa.

References

1. Amirikan, B. and Georgopoulos, A.P. Directional tuning profiles of motor cortical cells. *Neurosci. Res.* 36: 73-79, 2000.
2. Bartlett, M.S. Square root transformation in analysis of variance. *J. Roy. Stati. Soc. Suppl.* 3: 68-78, 1936.
3. Dayan, P. and Abbott, L.F. Theoretical Neuroscience: computational and mathematical modeling of neural systems, The MIT Press, 2001.
4. Georgopoulos, A.P., Caminiti, R., Kalaska, J.F. and Massey, J.T. Spatial Coding of Movement. A hypothesis concerning the coding of movement direction by motor cortical populations. *Exp. Brain Res. Suppl.* 7: 327-336, 1983.
5. Howell, D.C. Statistical Methods for Psychology, Wadsworth Publishing, 2006.
6. Mussa-Ivaldi, F.A. Do neurons in the motor cortex encode movement direction? An alternative hypothesis. *Neurosci. Lett.* 91: 106-111, 1998.
7. Salinas, E. and Abbott, L.F. Vector reconstruction from firing rates. *J. Comput. Neurosci.* 1: 89-107, 1994.
8. Sanger, T.D. Probability density estimation for the interpretation of neural population codes. *J. Neurophysiol.* 76: 2790-2793, 1996.
9. Serruya, M.D., Hatsopoulos, N.G., Paninski, L., Fellows, M.R. and Donoghue, J.P. Instant neural control of a movement signal. *Nature* 416: 141-142, 2002.
10. Wessberg, J., Stambaugh, C.R., Kralik, J.D., Beck, P.D., Laubach, M., Chapin, J.K., Kim, J., Biggs, S.J., Srinivasan, M.A. and Nicolelis, M.A.L. Real-time prediction of hand trajectory by ensembles of cortical neurons in primates. *Nature* 408: 361-365, 2000.