

# A Model of Reaching Dynamics in Primary Motor Cortex

**Sohie Lee Moody**

National Institutes of Mental Health

**David Zipser**

University of California, San Diego

## Abstract

■ Features of virtually all voluntary movements are represented in the primary motor cortex. The movements can be ongoing, imminent, delayed, or imagined. Our goal was to investigate the dynamics of movement representation in the motor cortex. To do this we trained a fully recurrent neural network to continually output the direction and magnitude of movements required to reach randomly changing targets. Model neurons developed preferred directions and other properties similar to real motor cortical neurons. The key finding is that when the target for a reaching movement changes loca-

tion, the ensemble representation of the movement changes nearly monotonically, and the individual neurons comprising the representation exhibit strong, nonmonotonic transients. These transients serve as internal recurrent signals that force the ensemble representation to change more rapidly than if it were limited by the time constants of individual neurons. These transients, if they exist, could be observed in experiments that require only slight modifications of the standard paradigm used to investigate movement representation in the motor cortex. ■

## INTRODUCTION

The primary motor cortex, M1, plays a role in the representation of most voluntary movements. This is seen in movement-specific neural activity that occurs before and during movements, during learned delays preceding movements, and even during imagined movements (Decety et al., 1994; Georgopoulos, Kalaska, Caminiti, & Massey, 1982; Georgopoulos, Caminiti, & Kalaska, 1984; Smyrnis, Taira, Ashe, & Georgopoulos, 1992). Individual neuron responses contribute to the representation of a movement in a complex way. Many different features of the movement contribute to a cell's response in a time-dependent manner. The movement feature most extensively studied is direction. Many M1 neurons exhibit a movement direction to which they fire maximally, their "preferred direction" (Georgopoulos, Caminiti, Kalaska, & Massey, 1983a; Georgopoulos et al., 1982). These neuronal responses can be further modulated by movement magnitude, by movement starting position, by arm configuration, and by when the response is observed, relative to movement onset (Caminiti, Johnson, & Urbano, 1990; Fu, Suarez, & Ebner, 1993; Schwartz & Georgopoulos, 1987; Scott & Kalaska, 1995). An individual M1 neuron cannot uniquely specify all features of a movement. Rather, the representation is distributed over a

population. Some of this distributed information can be recovered using information recorded from many single neurons. For example, the direction of an actual reaching movement can be found by averaging the activity-weighted preferred directions of many neurons. This provides an ensemble representation of movement direction called the *population vector* (Georgopoulos, Kettner, & Schwartz, 1988; Georgopoulos et al., 1983a). It is still not clear just how much information about a movement, such as the direction, magnitude, trajectory, force, etc., is represented in M1, nor is it clear what the representation is (Mussa-Ivaldi, 1988).

Movement representation is a dynamic process with many interacting components such as target location and movement direction. The problem of understanding the dynamics of the motor cortex supporting this representation is potentially very complex. Both intrinsic effects arising from interactions between neurons in M1 and extrinsic effects due to feedback from other areas must ultimately be considered. Here we address only the question of the internal dynamics of M1 by considering tasks for which dynamic feedback from other areas can be largely ignored. The recurrent network model developed in this work has interesting dynamic features that are amenable to experimental verification. Many impor-

tant aspects of movement, such as translating direction and distance into instructions for moving an arm with redundant degrees of freedom, are not covered by our model. Some of these issues are addressed by other models (Bullock, Grossberg, & Guenther, 1993).

Some dynamic aspects of M1 have been investigated experimentally. One approach is to rapidly change the location of a visual target just after movement begins (Georgopoulos, Kalaska, Caminiti, & Massey, 1983b; Georgopoulos, Kalaska, & Massey, 1981). This leads to a continuous change in the direction of hand movement from pointing toward the first target to pointing to the second. Another experimental paradigm involves movements along complex visually specified trajectories such as ellipses and sinusoids (Schwartz, 1994a; Schwartz, 1993). The population vector rotates smoothly during these movements so that it points in the direction of hand motion. A particularly revealing experiment requires movement in a direction offset by a fixed angle from the direction of a visual target (Georgopoulos, Lurito, Petrides, Schwartz, & Massey, 1989). In this case the actual direction of motion must be computed internally by a *mental rotation*. The population vector first points in the direction of the visual target and then rotates in the appropriate direction until it reaches the direction of actual movement.

Possible mechanisms underlying M1 dynamics have been investigated using recurrent neural network models. Of particular interest are models trained to simulate the behavior of the population vector during movements along a complex path (Lukashin, Wilcox, & Georgopoulos, 1994; Lukashin & Georgopoulos, 1994; Lukashin, 1990). In these models it is assumed that significant aspects of the dynamics arise from interactions between M1 neurons. The neurons are represented by a set of coupled differential equations:

$$\tau \frac{\partial s_i}{\partial t} = -s_i + \sum_j w_{ij} b_j + \cos(\theta - P_i) \quad (1)$$

where  $b_j$  is the output of the  $j$ th neuron,  $b_i = \tanh(s_i)$ ,  $w_{ij}$  are the synaptic strengths coupling the neurons,  $\theta$  is the current direction of movement,  $P_i$  is the preferred direction, and  $\tau$  is a neuron time constant. The  $\cos(\theta - P_i)$  term serves to impose direction tuning. In some cases, information about the path to be followed was supplied as input, and in others cases, path information was trained as internal to M1. Networks were trained, using simulated annealing, to output a population vector that had the dynamics observed in experiments. The networks learned the task and performed with population vector dynamics similar to those observed experimentally. These modeling efforts confirm that recurrent connections within M1 can account for the observed

ensemble dynamics as represented by the population vector. They provide a basis for further analysis of the dynamics of M1.

## THE MODEL

Building on these experimental findings and modeling results, we set out to develop a simple recurrent neural network model of M1 that could be used to study both the transient behavior of individual neurons and the population dynamics of the network as a whole. Our goal was a dynamic model of how movement is represented in M1 that could deal with a wide variety of natural movements while introducing as few arbitrary initial constraints as possible. To this end we used an optimization, or "training," procedure to configure a network of fully interconnected units. The training set and input format were chosen to give a reasonable degree of realism, as will be described later. The network itself is described by the set of differential equations,

$$\tau \frac{\partial s_i}{\partial t} = -s_i + \sum_j w_{ij} b_j + \sum_k v_{ik} z_k \quad (2)$$

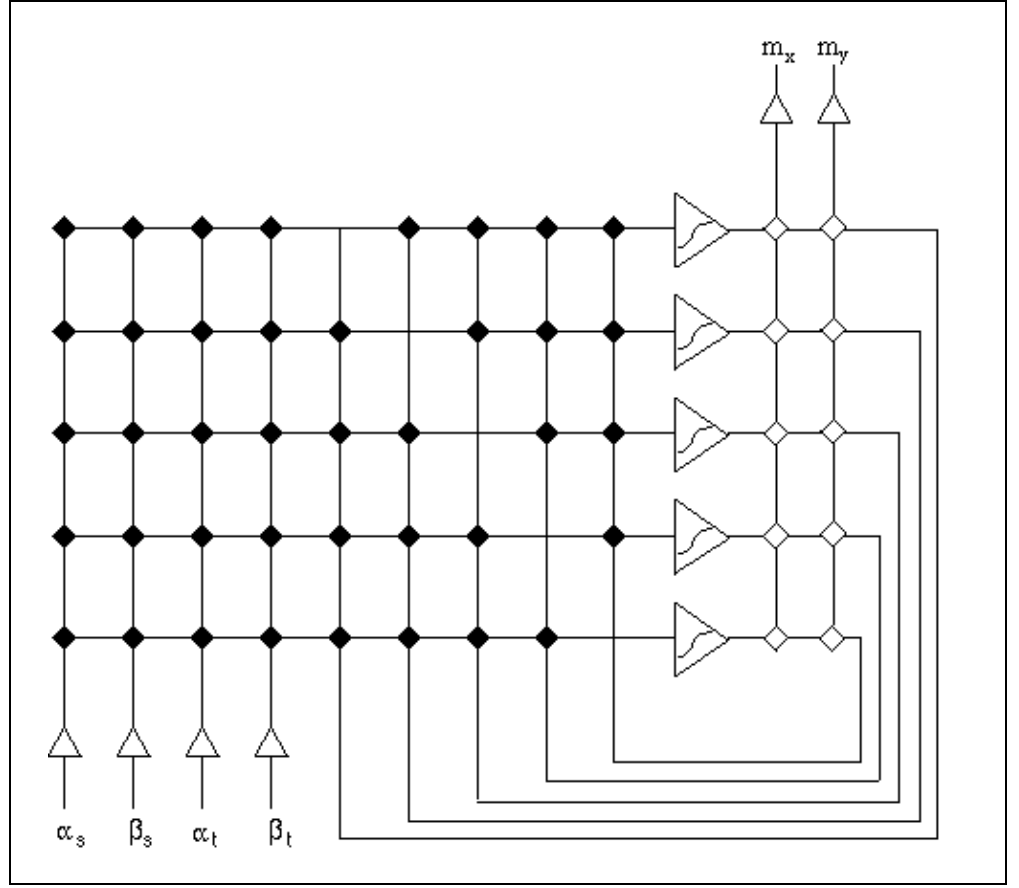
where all similar terms are defined as in Equation 1 except  $b_j = 1/(1 + e^{-s_j})$ . The rightmost term is the external input, where  $z_k$  are the input values and  $v_{ik}$  are the weights that connect each input line to every unit in the recurrent network. The ensemble movement vector represented by the network was determined using a weighted linear combination of all the unit activities,  $b$ . In two dimensions, the movement vector is given by

$$\left. \begin{aligned} m_x &= \sum_j x_j b_j \\ m_y &= \sum_j y_j b_j \end{aligned} \right\} \quad (3)$$

where  $m_x$  and  $m_y$  are the components of the current movement vector represented by the model, and  $x_j$  and  $y_j$  are constants. The  $m_x$  and  $m_y$  values are computed by a set of linear units serving as an output layer to the M1 model network. These linear units are not considered part of M1 but rather constitute a reporting mechanism that determines the ensemble representation and also serves to funnel in error signals during training. The model network is illustrated in Figure 1.

To achieve a veridical model it would be desirable to use an input format that approximated the real afferent signals to M1. Unfortunately, the actual afferent format used by M1 is unknown. However, it is known that parameters such as preferred direction, dynamic range, and baseline firing rate are affected by the starting position for movement (Caminiti et al., 1990; Kettner, Schwartz, & Georgopoulos, 1988; Georgopoulos, Kalaska,

**Figure 1.** Diagram of the recurrent network model. Each sigmoid-containing triangle represents a model unit, which receives external input and feedback from other network units but not from itself. A model unit's activation is determined by performing a non-linear operation on the weighted sum of its inputs. The solid black diamonds represent the  $w_{ij}$  and  $v_{ik}$ , and white diamonds represent output unit weights. An output unit's activation is determined by the linear weighted sum of the hidden unit activity (see Equation 3). A depth of five network copies were used in back propagation through time (BPTT) simulations. The inputs are explained in the text and caption to Figure 2.

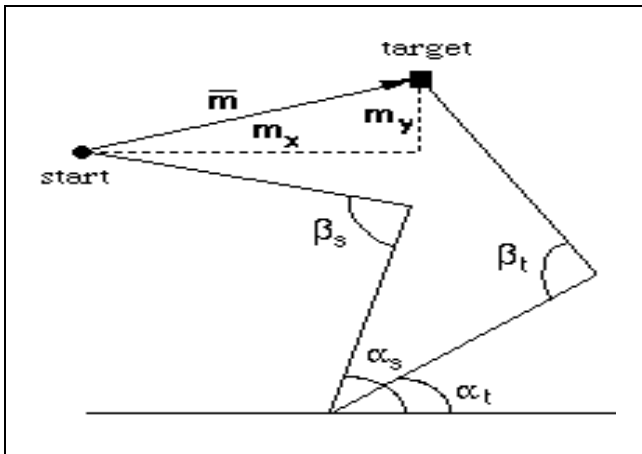


Crutcher, Caminiti, & Massey, 1984). This and the properties of the population vector imply that information about both starting location and target position are available to M1. It has been shown that as long as afferent information about the starting location and target position are represented in a reasonably smooth coordinate system, the M1 neurons will be tuned to preferred motion directions, if they are connected to all the input lines by arbitrary weights in an appropriate magnitude range (Sanger, 1994). This means that the choice of afferent representation should have a negligible effect on the general form of the tuning properties of model neurons as long as it is not redundant. Therefore, it is not necessary to impose preferred directions on the equations describing these neurons. We have empirically confirmed both of these conjectures with simulation experiments using randomly chosen weights and several different coordinate representations for model input (Lee, 1996).

The above considerations allow the use of any convenient input format to specify movements, so long as it codes information about current hand position and target location. In other studies inputs specifying this information in Cartesian, polar, and joint angle coordinates were investigated (Lee, 1996). All three gave very similar dynamics. We chose joint angle coordinates to represent

input to the M1 model described here because they interfaced naturally with the arm model used to generate the input data. Real arms have redundant degrees of freedom, allowing multiple postures to reach the same point in space. We used a two-dimensional arm to avoid this ill-posed problem because we believe it is orthogonal to the questions about the internal dynamics of M1 that we investigate here. The details of the two-dimensional arm angle system used are given in Figure 2.

To teach an M1 model to represent arbitrary 2-D movements, a training set containing movement segments sampling all directions and all changes in direction is needed. The task that motivated model input was picking up small food morsels scattered randomly on a surface. The hand moves from one morsel to the next, with the location of each new morsel representing the target for successive movements. A change in direction always occurs after a target is reached. To enrich this movement paradigm and provide for direction changes before a target is reached, we imagined that there was a second individual competing for the same morsels. Due to this competition, a target sometimes disappears before it is reached, leading to a change in direction to a new target. Direction changes may occur while the hand is in motion or while the hand is resting at a target. These considerations led to the following training regime for

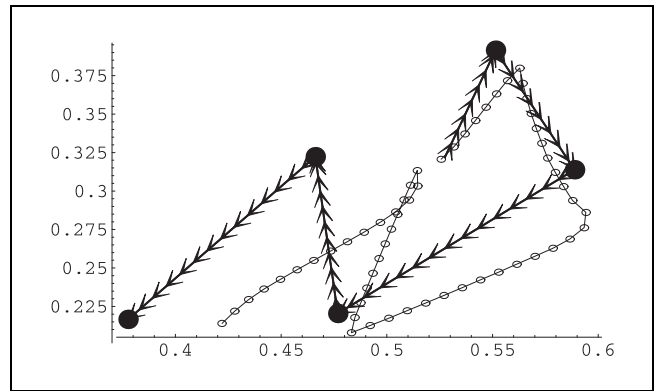


**Figure 2.** Model input and output. Input consists of two joint angle pairs,  $(\alpha_s, \beta_s)$  and  $(\alpha_t, \beta_t)$ , for start and target, respectively. The elbow angle  $\beta$  ranges from  $20^\circ$  to  $180^\circ$ ; the shoulder angle  $\alpha$ , from  $0^\circ$  to  $160^\circ$ , with 0 arbitrarily set at the positive  $x$ -axis. The model output consists of the movement vector components  $m_x$  and  $m_y$ , connecting the current hand position (the dot) to the target (the square).

generating a continuous stream of target- and hand-location input pairs.

The target-location input is chosen at random and is kept constant for a random length of time. The hand moves toward a target in fixed-length steps of 7% of the average movement vector length. If the hand reaches the target, it pauses at the target for a random interval and then pursues the next target. Hand stops at targets are indicated by setting the movement vector,  $(m_x, m_y)$  to zero (0,0). If the target changes location while the hand is moving (because the competitor took the target morsel), the hand either changes direction and moves toward a new target immediately or stops for a while. The rate of hand movement, and the probabilities of target location changes and hand stopping and starting are all parameters of the program that generates input-output data for the model during training. Figure 3 shows sample input training data, as well as a representation of the hand path generated by the trained model when supplied with the same sequence of input data, without feedback about actual hand position. There is some residual error, but the general shape of the two paths is quite similar.

The network was trained using a version of the Back Propagation Through Time learning algorithm, modified to approximate continuous time systems of equations, such as Equation (2) (Kehoe, 1995). Error was evaluated based on the difference between the  $m_x$  and  $m_y$  given by the model and the correct values of  $m_x$  and  $m_y$  required to connect the current hand position to the current target location. This error is minimized by a network that establishes an ensemble representation of the correct movement vector as rapidly as possible. It is



**Figure 3.** Sample training data. The arrowed line shows training data. The five large solid circles are targets. The input hand position, represented as arrowheads, moves in fixed steps toward the current target. (We also allow for the hand to stop as it moves toward a target as if the subject were distracted. The probability of such stops was set at values of 0.0 or 0.2, but these distractive stops had no discernable effect on the final model, so none are shown in this figure.) Each target appears for between 20 to 70 update cycles. New targets are produced in three ways: First, the current target exceeds its maximum target duration (70 update cycles), second, a target switch occurs as the hand is moving (probability = 0.20 per update cycle), and third, a target switch occurs as the hand is stopped (probability = 0.80 per update cycle). The model was run for 10 million cycles (about 6 hs) on a Silicon Graphics Iris Indigo workstation.

*Hand path.* The line with small circles shows a representation of hand path without feedback, generated by a trained model from the same hand and target input data. The hand path started at the same location as the first training location. Each time the training hand position changed, the hand path was advanced, for the same fixed distance, in the direction indicated by the trained network output  $m_x$  and  $m_y$ . See text for more detail.

the requirement for both speed and accuracy that leads to the steady-state and transient properties of the model.

## LINEAR APPROXIMATION TO THE TRAINED MODEL

A linearized version of the model was derived to aid in understanding the way the model works. The fact that the dynamic ranges of the model unit outputs  $b_i$  are confined to small, approximately linear, regions of the sigmoid function suggested that it would be possible to find an accurate linear approximation to the nonlinear Equation 2. Finding such an approximation was facilitated by the observation that the trained nonlinear model was stable. That is, given a fixed value of the input,  $z = (z_1, z_2, \dots, z_n)$ ,  $\partial s_i / \partial t \rightarrow 0$  as  $t \rightarrow \infty$  for all  $i$ , and for all  $z$  representing locations in the training region. Linearization is also simplified because the steady-state value of unit activations,  $b = (b_1, b_2, \dots, b_m)$ , are uniquely determined by the associated value of  $z$ .

The nonlinear Equation 2 can be written in matrix form:

$$\tau s' = -s + Wb + Vz \quad (4)$$

Linearizing this system around the stable state,  $s_z$ , corresponding to input  $z$ , gives the formal solution (Perko, 1991):

$$s = e^{(t/\tau)A} [s(0) + A^{-1}(b + Vz)] - A^{-1}(b + Vz) \quad (5)$$

where

$$A = WH'(s^z) - I$$

$$b = Wb(s^z) - WH'(s^z)s^z$$

In the solution above,  $I$  is the identity matrix,  $H'(s_z)$  is a matrix with diagonal  $\partial b(s_i)/\partial s_i$  and all nondiagonal elements are 0, and  $s(0)$  is the initial value of  $s$ . These equations were solved explicitly by the method described in the "Appendix" to obtain numerical values that can be compared to those generated by simulation of the nonlinear model.

One advantage of having a linear approximation to our model is that the issue of stability can be understood analytically. The linear system described by Equation 5 is stable if the real parts of the eigenvalues of  $A$  are negative. This was found to be the case in our trained model for all the  $A$  generated by any  $z$  in the training region. The negative eigenvalues of  $A$  guarantee that the exponential term in Equation 5 eventually goes to 0, giving a steady state value,  $s_z$ :

$$s^z = -A^{-1}(b + Vz) \quad (6)$$

The existence of stability is consistent with the experimental observations (Georgopoulos, Taira, & Lukashin, 1993).

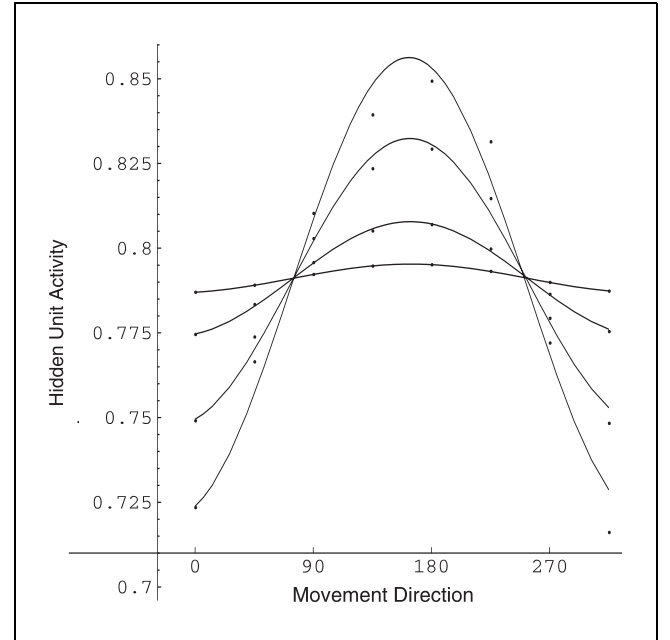
## PROPERTIES OF THE M1 MODEL

The properties of the model neurons were determined by simulating tasks similar to the ones used in experiments with monkeys. However, there is one important difference between how we test the model and how experiments are generally performed. During testing of the model, only the representation of movement changes; *no actual hand movement occurs*. This is accomplished by fixing the hand-position input to correspond to a point in the center of the training region and then measuring responses to targets at various locations. The reason we investigate only the no-movement case is that our model is limited to movement representation in M1 itself. It can provide no accurate information about the actual kinetics of hand movement. Without this we cannot simulate the feedback effect of movement on model kinetics.

Our main interest is in the transient properties of the model, but we also measured the steady-state responses of individual neurons and the population. The steady-state response properties were measured at many uniformly distributed locations. At each starting position eight equally spaced radial target locations were used. Measurements were made only after many simulation

cycles had elapsed. This allowed steady state to be reached. Since the hand position is kept constant at the center, no movement occurs in these tests. The population steady-state properties were determined using either the movement vector output by the network or by computing a population vector based on the outputs of individual recurrent neurons. The steady-state errors for direction and magnitude were small. For example, the average direction error of the population vector was  $14^\circ$  with a standard deviation of  $10^\circ$  and a maximum error of  $30^\circ$ . Direction errors in the movement vector, which was trained directly, were about half as great as the population vector error. Amplitude errors for both the population and movement vectors were around 5% of movement amplitude.

The steady-state response of individual neurons at any starting position is described by  $Kr\cos(\theta - P) + b$ . Figure 4 illustrates the steady-state response properties of a typical model neuron at a single starting position. The direction response of the model and real M1 neurons can be fit fairly well with a cosine, so they are similar. Monotonic amplitude modulation is seen in virtually all direction-tuned M1 neurons, so the model neurons are qualitatively like cortex neurons in this respect. However, the quantitative form of the real M1 neuron amplitude response is quite varied and only a small portion



**Figure 4.** Effect of amplitude and direction on unit activity (dots), shown relative to a cosine function (solid lines). Unit steady-state activity is plotted across eight directions in  $45^\circ$  intervals between  $0^\circ$  and  $360^\circ$ . The activity of one representative nonlinear network unit is shown for four different amplitudes (0.005, 0.02, 0.05, 0.08). Note that the larger amplitude produces a higher level of activity and that the unit exhibits direction as well as amplitude sensitivity. The model unit's fit (F-test) for the largest amplitude is  $R^2 > 0.96$ ;  $p < 0.01$ . All other fits have  $R^2 > 0.99$ .

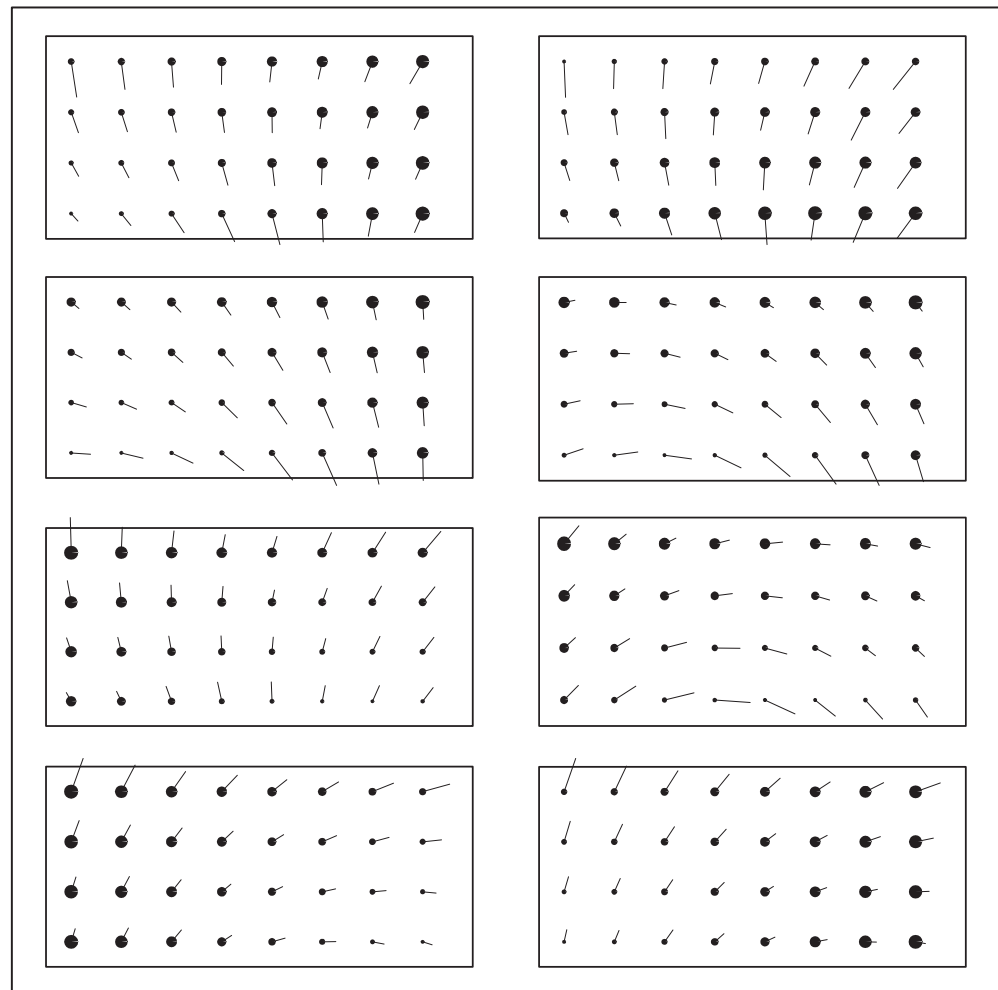
appears to respond in the simple way that model neurons do. The rest of the M1 neurons seem to have a more complex monotonic amplitude response that requires as many as six parameters to describe it (Fu et al., 1993; Schwartz & Georgopoulos, 1987). The relevance of this to the major findings of our model will be discussed below.

The response properties of M1 neurons vary with the starting location of a movement and other factors (Kalaska & Crammond, 1992; Hocherman & Wise, 1991; Lemon, 1988; Dum & Strick, 1990; Schwartz, 1994b). For example, preferred directions  $P$  were dependent upon the starting hand position. The median angular difference between a real neuron's preferred direction in the left space compared to the center space, center space compared to the right, and left space compared to right were  $28^\circ$ ,  $23^\circ$ , and  $43^\circ$ , respectively. The median angular difference between a model neuron's preferred direction to the same conditions were  $39^\circ$ ,  $38^\circ$ , and  $60^\circ$ , respectively. Baseline  $b$  and dynamic range  $K$  were also affected. Figure 5 illustrates the spatial variation of  $b$ ,  $P$  and  $K$  for eight hidden units. Note that the baseline variation with the starting hand location qualitatively resembles the

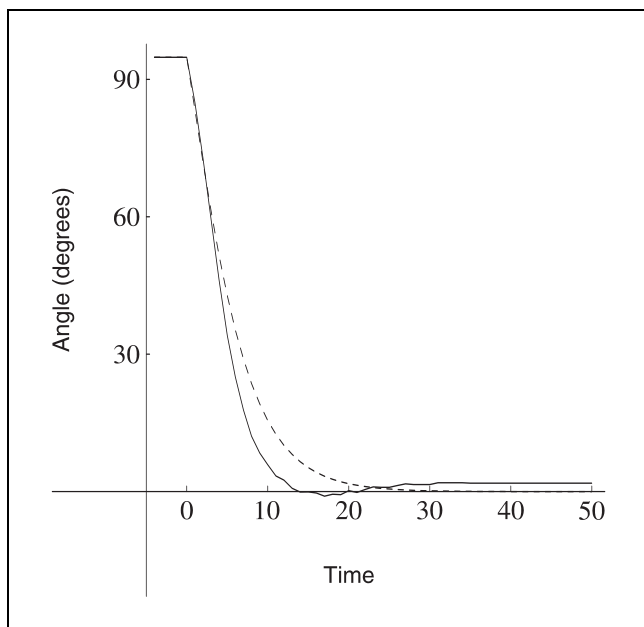
experimentally observed variation (Georgopoulos, Kalaska, et al., 1984). The overall pattern of the  $P$ ,  $K$ ,  $b$  variation is different for each hidden unit; moreover,  $P$ ,  $K$ , and  $b$  seem to vary independently of each other. The distribution of preferred directions is reasonably uniform even in this eight-unit model, as it is in the real cortex. The qualitative characteristics of the  $P$ ,  $K$ , and  $b$  variation in the model are similar to those found experimentally, but there are not enough experimental data available for detailed comparison. Ultimately it may be possible to work backward from detailed experimental data about the spatial variation of the tuning parameters  $P$ ,  $K$ , and  $b$  to find the coordinate system used to represent the target and hand position or the inputs to the motor cortex.

The transient behavior of the model was tested using a target-position switch paradigm. Targets in the switch task are on a circle surrounding the hand. The model is allowed to settle to steady state with the target at an initial position. Then the target-position input is changed to indicate a new location on the circle, and the responses of all model units are followed through time until a new steady state is reached. The hand-position

**Figure 5.** Effect of starting position on tuning parameters. The plots show the eight model unit responses at 32 ( $8 \times 4$ ) evenly spaced center points. A value for each reaching parameter (preferred direction, dynamic range, baseline) was calculated for each hidden unit for each location. At each point, the preferred direction is indicated by line orientation, dynamic range is indicated by line length, and baseline activity is indicated by dot size. The preferred direction line length was normalized so that each plot displays the same range; the same is true for baseline dot size.







**Figure 6.** The solid line graphs the direction of the actual model population vector following a change in target position from  $90^\circ$  to  $0^\circ$ . Note that the initial and final steady-state directions of the population vector are not exactly at  $90^\circ$  and  $0^\circ$  because there is some error in the model ensemble representation. The dashed line graphs an exponential decay at the rate of the neuron time constant used in the model. It illustrates the kinetics of population rotation expected if there were no recurrent interaction in the model network.

input is kept constant, so there is no actual movement taking place. This procedure is used to obtain information about the transient behavior of both the individual units and the ensemble representation.

The population vector was used as an indicator of ensemble transient behavior. The population vector of the trained model rotates smoothly in the correct direction for all changes in target position. Interestingly, it completes this rotation sooner than if there were no recurrent connections. This is illustrated in Figure 6, which compares the change in direction of the population vector following a switch in target location to the change expected if there were no recurrent connections (i.e., the rotation dynamic was determined only by neuron time constants). The observed speedup must be due to recurrent interactions because the input is held constant and there are no other external signal sources. As we shall see, this interaction involves strong, nonmonotonic transient responses in the individual model neurons.

The transient behavior of each individual model neuron was tested in two ways: first, by finding a pair of target positions at which the neuron has the same steady-state activity and second, by finding a pair of target positions that produce the maximum possible difference in steady-state activity. Different sets of target locations were required for each of the eight neurons in

the network. In all of these tests, the targets are arranged around a circle, so there are no amplitude changes, only changes in direction. Also, recall that no movement occurs in these tests. The response of an individual neuron changes from the steady-state value representing its contribution to a movement in one direction to its response representing movement in another direction. This differs from the typical monkey experiment, in which movement occurs and a neuron's response returns to baseline when the movement is complete.

The first test (i.e., no change in activity condition) is accomplished by choosing target locations corresponding to equal heights on opposite sides of the direction tuning curve (see the top panel of Figure 7). The large, nonmonotonic activity transients observed in this task are illustrated in the bottom panels of Figure 7. These transients start immediately after the change in target location and continue well past the time when the first 90% of change in the ensemble representation has occurred (indicated by the vertical dashed line). Note that the nonmonotonic transients have the same shape, but opposite signs, when we reverse the order in which the targets are presented. This sign reversal is an algebraic consequence of the target switch paradigm, at least for linear systems (for details, see the "Appendix").

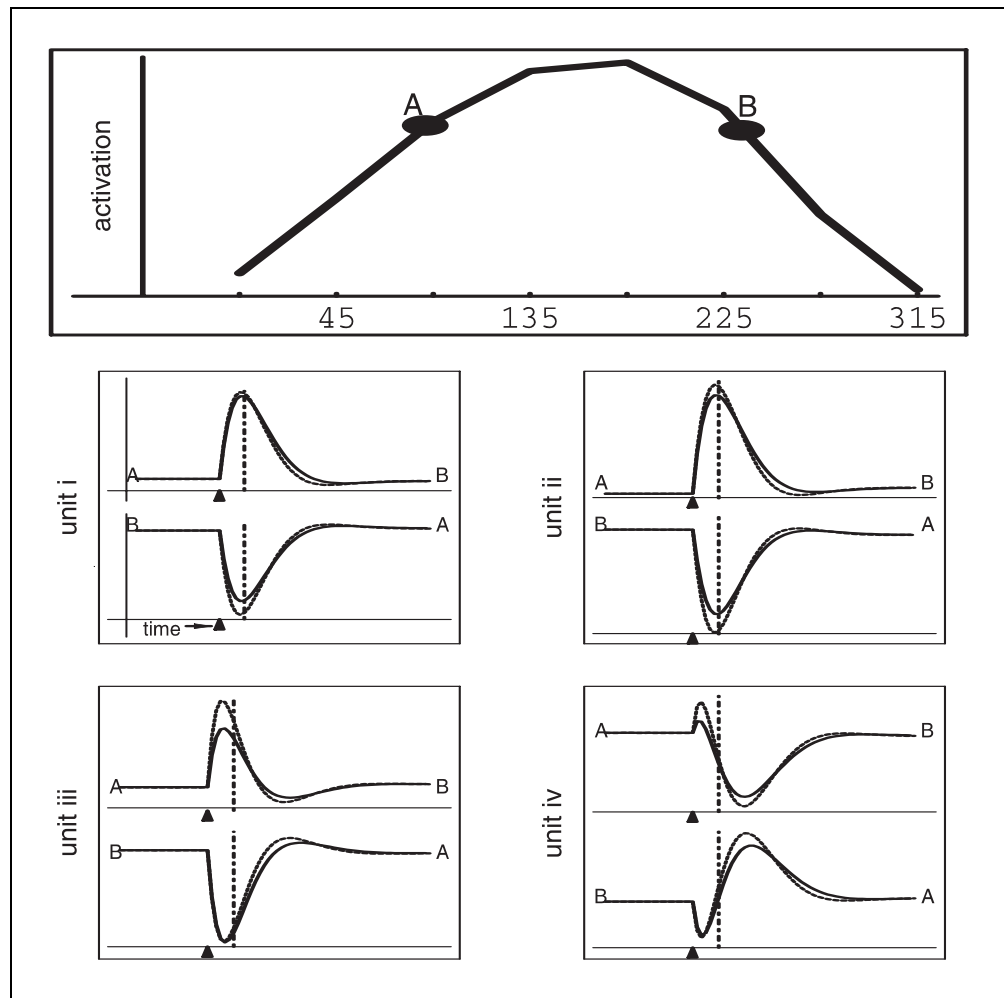
The units shown in Figure 8 exhibit *monotonic* changes in activity level between the initial and final target locations, when these two locations are at the extremes of the dynamic range (i.e., when the change in target location goes from a unit's preferred direction to its opposite). Figure 8 illustrates this case. For all changes intermediate between those illustrated in Figures 7 and 8, transient behavior is a combination of some nonmonotonic response with a net change in overall activity; see Figure 9.

## DISCUSSION

The accelerated change in population vector rotation can only arise from internal forcing signals generated by the nonmonotonic transients. However, for these transients not to disrupt the smooth rotation of the population vector, their direct contribution to the population vector must nearly cancel out. This is required because the strongest transients are from neurons with preferred directions orthogonal to the actual direction of movement. Cancellation is apparently accomplished in the model by having some units with positive transients balanced by others with negative transients. The optimization procedure used to train the model has managed to reap the benefit obtained from the forcing transients, while canceling out any adverse effect these transients might have on the direction of the population vector. This balancing could also occur statically in large populations of M1 neurons.

It is an open question whether anything like the dynamic effects seen in the model exist in M1. Experi-

**Figure 7.** Transient behavior with equal values for initial and final steady states. The top panel shows how the targets, A and B, selected at equal unit steady-state activations on the direction tuning curve. The four small panels show individual unit transient behavior during a target switch between equal steady-state values. The vertical axes are activities normalized, so all model neurons have the same size dynamic range on all graphs. The horizontal axis is time in units of network update cycles and is the same for all graphs. These same conventions are also used in the next two figures. The top half of each panel shows unit activity as the input switches from A to B; the bottom half shows the reverse switch. The arrow on the x-axis indicates the time of target switch. The solid trace shows nonlinear network model activation from simulation; the dashed trace shows the activation from linearized equations. The vertical dashed line indicates the point at which the ensemble representation has covered 90% of the distance to the target.



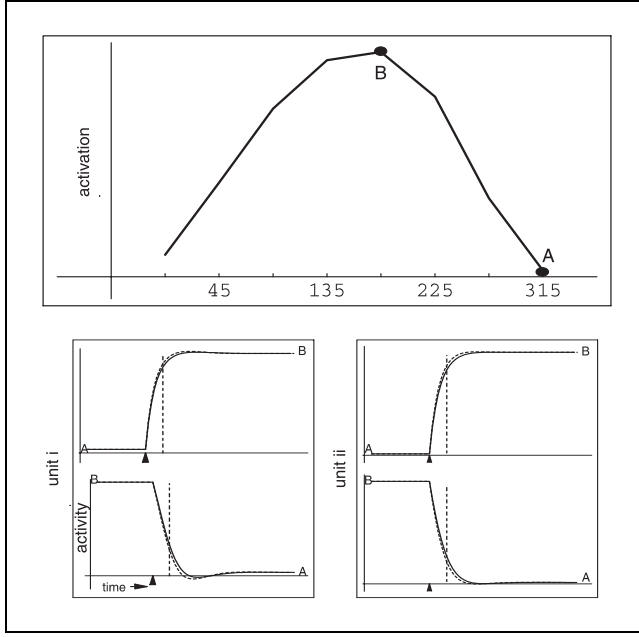
tal tests would be straightforward but have not, to our knowledge, been published. These experiments would require a modification of the standard paradigm so that they mimicked the no-movement task we use. This is important, because if movement is allowed to occur immediately following target presentation, there will be additional dynamic effects that can confuse the situation. Also the change of hand position with movement generates a change in background firing that would distort any transient activity. The ideal experiment would use a trained delay prior any movement during which a switch in target location could occur. In such experiments it should be possible to look for the existence of transients of the kind we predict. An experiment in which targets are switched without movement has been done that has some similarities to what is needed to test our model (Wise & Mauritz, 1985). The published data for the response of a cell to opposite directions of motion in the preferred and antipreferred directions give the result predicted in Figure 8.

Nonmonotonic transients of the kind we predict are detectable, as demonstrated by a kind of target switch

experiment involving recall (Pellizzer, Sargent, & Georgopoulos, 1995). In this experiment transients of the time scale and magnitude our model predicts are observed. Although consistent with our predictions, these observations are not significant for our model because they take place just preceding and during movement. Cells with activities intermediate between the stimulus and movement direction become active during the “mental rotation task” (Georgopoulos, 1995; Pellizzer et al., 1995). This is in general agreement with our model; however, very little complete time course data have been published, so direct comparison is not possible. Also this experiment involves the period just before and during movement. The need for such a stringent no-movement condition, although achievable, is perhaps a weakness of our model. However, to deal with actual movement we would have to be able to model the dynamics of cell recruitment and of feedback to M1 from other parts of the brain. There is not at present enough knowledge to do this with any hope of success.

Our model captures the form of direction tuning found in M1 quite well. The issue of amplitude repre-





**Figure 8.** Transient behavior of units with maximum change in steady-state values. The top panel shows the targets, A and B, selected at preferred and antipreferred directions on the direction tuning curve. The two small panels show representative individual unit transient behavior during a target switch. The top half of each panel shows unit activity as the input switches from A to B; the bottom half shows the reverse switch. Labeling conventions as in Figure 7.

sensation is more problematic. Many questions about how M1 neurons respond to the amplitude of a movement are not yet resolved. However, the neurons in our model represent amplitude in a way that probably corresponds to only a small subset of M1 neurons. This is not likely to affect the validity of the major dynamical predictions of the model because all the dynamic tests we report were at constant amplitudes, (i.e., the directions, but not the amplitudes of targets, were changed to investigate transient effects).

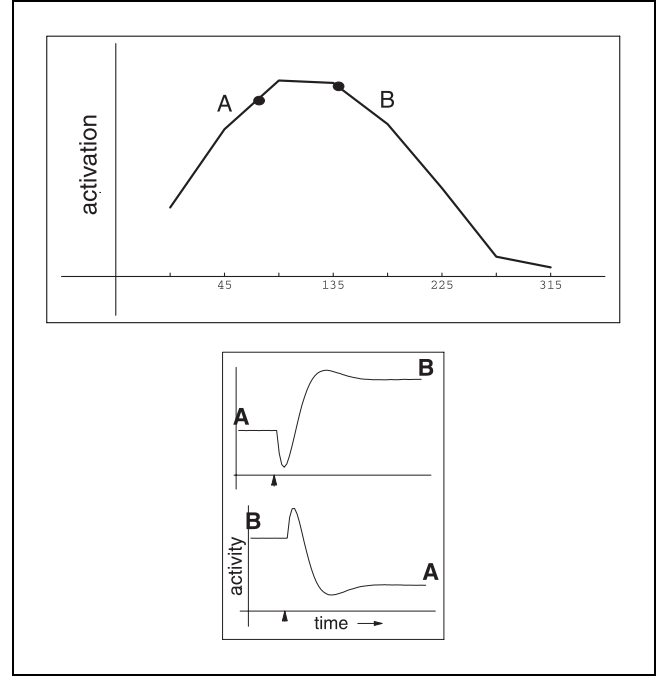
## Appendix

### Linearization of M1 Differential Equations

The method used to linearize Equation 2 is given here,

$$\left. \begin{aligned} \tau \frac{\partial s_i}{\partial t} &= -s_i + \sum_j w_{ij} b_j + \sum_k v_{ik} z_k \\ b_j &= \frac{1}{1 + e^{-s_j}} \end{aligned} \right\}$$

where  $w$  and  $v$  are fixed weights determined by training,  $z_k$  values are set to correspond to points A and B in figures 6 and 7, and  $\tau$  is a time constant.



**Figure 9.** Transient behavior with intermediate values. Same conventions as Figure 7.

The steady-state values  $s^z$  were found by simulation. Having obtained  $s^z$ ,  $b$  is expanded around  $s^z$ :  $b(s_i) = b_i + b'_i(s_i - s_i^z)$ ;  $h_i = h_i = b(s_i^z)$ ,  $b'_i = b_i(1 - b_i)$  and Equation 2 becomes

$$\begin{aligned} \tau \frac{\partial s_i}{\partial t} &\approx -s_i + \sum_j w_{ij} b'_j s_j + \sum_k v_{ik} z_k + \sum_j w_{ij} (b_j - b'_j s_j^z) \\ \tau \frac{\partial s_i}{\partial t} &\approx \sum_j A_{ij} s_j + \zeta_i \end{aligned} \quad (7)$$

where

$$\begin{aligned} A_{ij} &= -\delta_{ij} + \sum_j w_{ij} b'_j \\ \zeta_i &= \sum_k v_{ik} z_k + \sum_j w_{ij} (b_j - b'_j s_j^z) \end{aligned}$$

Let  $\alpha_i$  be the eigenvalues of  $A$  and  $\delta_{ij}$  be the Kronecker delta function. Define the matrix  $X$  such that each column of  $X$  is an eigenvector of  $A$ , and  $X^{-1}$  as its inverse.

After inserting  $XX^{-1}$ , each term of Equation 7 is multiplied by  $X^{-1}$  and summed over  $i$  to produce

$$\begin{aligned}
\tau \sum_i X_{ki}^{-1} \frac{\partial s_i}{\partial t} &= \sum_{ijkl} X_{ki}^{-1} A_{ij} X_{jl} X_{lm}^{-1} s_m + \sum_i X_{ki}^{-1} \zeta_i \\
&= \sum_{lm} \alpha_l \delta_{kl} X_{lm}^{-1} s_m + \sum_i X_{ki}^{-1} \zeta_i \quad (8) \\
&= \sum_m \alpha_k X_{km}^{-1} s_m + \sum_i X_{ki}^{-1} \zeta_i
\end{aligned}$$

If  $u_k$  and  $v_k$  are defined as the terms of Equation 8, respectively,

$$u_k = \sum_i X_{ki}^{-1} s_i \quad v_k = \sum_i X_{ki}^{-1} \zeta_i \quad (9)$$

the equations reduce to

$$\tau \dot{u}_k = \alpha_k u_k + v_k$$

with solutions

$$u_k(t) = (u_k(0) - u^z) e^{(\alpha_k t)/\tau} + u^z \quad (10)$$

$$u_k(0) = \sum_i X_{ki}^{-1} s_i(0) \quad (11)$$

The unit activity of  $s_i$  is then

$$s_i(t) = \sum_k X_{ik} u_k(t) \quad (12)$$

In Figures 6 and 7,  $s(0)$  is the steady state at the start point, and  $s^z$  is the steady state at the end point.

### Sign Inversion as a Result of Target-Switch Paradigm

We start with Equation 12 above:

$$s_i(t) = \sum_k X_{ik} u_k(t)$$

Substituting Equation 10 into Equation 12 yields

$$s_i(t) = \sum_k [X_{ik}(u_k(0) - u^z) e^{(\alpha_k t)/\tau} + X_{ik} u^z] \quad (13)$$

Substituting Equations 9 and 11 into Equation 13 yields

$$s_i(t) = \sum_k [X_{ik} e^{(\alpha_k t)/\tau} X_{kj}^{-1}] (s_j(0) - s_j^z) + X_{ik} X_{kj}^{-1} s_j^z \quad (14)$$

Simplifying Equation 14 yields

$$s_i(t) = \sum_j \left[ \sum_k (X_{ik} e^{(\alpha_k t)/\tau} X_{kj}^{-1}) (s_j(0) - s_j^z) \right] + s_j^z \quad (15)$$

The  $(s(0) - s^z)$  term in Equation 15 is the key to the sign reversal seen in Figures 6 and 7. When the  $s(0)$  and

$s^z$  are interchanged, the sign of this term changes, which causes the entire response to change sign as well.

### Acknowledgments

This research was supported by NSF grant SBR 92-56834 to Sohie Lee Moody and by NIMH grant 5R01MH45271 to David Zipser. We gratefully acknowledge Dr. David Wong for his contribution to the linear analysis of the network differential equations.

Requests for reprints should be sent to Sohie Lee Moody, Laboratory of Systems Neuroscience, NIMH, P.O. Box 608, Poolesville, MD 20837.

### REFERENCES

- Bullock, D., Grossberg, S., & Guenther, F. H. (1993). A self-organizing neural model of motor equivalent reaching and tool use by a multijoint arm. *Journal of Cognitive Neuroscience*, 5, 408-435.
- Caminiti, R., Johnson, P. B., & Urbano, A. (1990). Making arm movements within different parts of space: Dynamic aspects in the primate motor cortex. *Journal of Neuroscience*, 10, 2039-2058.
- Decety, J., Perani, D., Jeannerod, M., Bettinardi, V., Tadary, B., & Woods, R. (1994). Mapping motor representations with positron emission tomography. *Nature*, 371(6498), 600-606.
- Dum, R. P., & Strick, P. L. (1990). *Motor control: Concepts and issues*. London: Wiley.
- Fu, Q. G., Suarez, J., & Ebner, T. (1993). Neuronal specification of direction and distance during reaching movements in the superior precentral premotor area and primary motor cortex of monkeys. *Journal of Neurophysiology*, 70, 2097-2116.
- Georgopoulos, A. P. (1995). Current issues in directional motor control. *Trends in Neuroscience*, 18, 506-510.
- Georgopoulos, A. P., Caminiti, R., & Kalaska, J. F. (1984). Static spatial effects in motor cortex and area 5: Quantitative relations in a two-dimensional space. *Experimental Brain Research*, 54, 446-454.
- Georgopoulos, A. P., Caminiti, R., Kalaska, J. F., & Massey, J. T. (1983a). Spatial coding of movement: A hypothesis concerning the coding of movement direction by motor cortical populations. *Experimental Brain Research*, Supplement 7, 327-336.
- Georgopoulos, A. P., Kalaska, J. F., Caminiti, R., & Massey, J. T. (1982). On the relations between the direction of two-dimensional arm movements and cell discharge in primate motor cortex. *Journal of Neuroscience*, 2, 1527-1537.
- Georgopoulos, A. P., Kalaska, J. F., Caminiti, R., & Massey, J. T. (1983b). Interruption of motor cortical discharge subserving aimed arm movements. *Experimental Brain Research*, 49, 327-340.
- Georgopoulos, A. P., Kalaska, J. F., Crutcher, M. D., Caminiti, R., & Massey, J. T. (1984). *Dynamic aspects of neocortical function*. New York: Wiley.
- Georgopoulos, A. P., Kalaska, J. F., & Massey, J. T. (1981). Spatial trajectories and reaction times of aimed movements: Effects of practice, uncertainty, and change in target location. *Journal of Neurophysiology*, 46, 725-743.
- Georgopoulos, A. P., Kettner, R. E., & Schwartz, A. B. (1988). Primate motor cortex and free arm movements to visual targets in three-dimensional space II. Coding of the direc-

- tion of movement by a neuronal population. *Journal of Neuroscience*, 8, 2928–2937.
- Georgopoulos, A. P., Lurito, J. T., Petrides, M., Schwartz, A. B., & Massey, J. T. (1989). Mental rotation of the neuronal population vector. *Science*, 243, 234–236.
- Georgopoulos, A. P., Taira, M., & Lukashin, A. (1993). Cognitive neurophysiology of the motor cortex. *Science*, 260, 47–52.
- Hocherman, S., & Wise, S. P. (1991). Effects of hand movement path on motor cortical activity in awake, behaving rhesus monkeys. *Experimental Brain Research*, 83, 285–302.
- Kalaska, J. F., & Crammond, D. J. (1992). Cerebral cortical mechanisms of reaching movements. *Science*, 255(5051), 1517–1523.
- Kehoe, B. (1995). *Teaching recurrent networks of continuous units*. Technical report, University of California, San Diego.
- Kettner, R. E., Schwartz, A. B., & Georgopoulos, A. P. (1988). Primate motor cortex and free arm movements to visual targets in three-dimensional space III. Positional gradients and population coding of movement direction from various movement origins. *Journal of Neuroscience*, 8, 2938–2947.
- Lee, S. J. (1996). *The representation, storage and retrieval of reaching movement information in motor cortex*. Unpublished Ph.D. thesis, University of California, San Diego.
- Lemon, R. (1988). The output map of the primate motor cortex. *Trends in Neuroscience*, 11, 501–506.
- Lukashin, A. V. (1990). A learned neural network that simulates properties of the neuronal population vector. *Biological Cybernetics*, 63, 377–382.
- Lukashin, A. V., & Georgopoulos, A. P. (1994). A neural network for coding trajectories by time series of neuronal population vectors. *Neural Computation*, 6, 19–28.
- Lukashin, A. V., Wilcox, G. L., & Georgopoulos, A. P. (1994). Overlapping neural networks for multiple motor engrams. *Proceedings of National Academy of Science*, 91, 8651–8654.
- Mussa-Ivaldi, F. A. (1988). Do neurons in the motor cortex encode movement direction? An alternative hypothesis. *Neuroscience Letters*, 91, 106–111.
- Pellizzer, G., Sargent, P., & Georgopoulos, A. P. (1995). Motor cortical activity in a context-recall task. *Science*, 269, 702–705.
- Perko, L. (1991). *Differential equations and dynamical systems*. New York: Springer-Verlag.
- Sanger, T. D. (1994). Theoretical considerations for the analysis of population coding in motor cortex. *Neural Computation*, 6, 29–37.
- Schwartz, A. B. (1993). Motor cortical activity during drawing movements—population representation during sinusoid tracing. *Journal of Neurophysiology*, 70, 28–36.
- Schwartz, A. B. (1994a). Direct cortical representation of drawing. *Science*, 265(5171), 540–542.
- Schwartz, A. B. (1994b). Distributed motor processing in cerebral cortex. *Current Opinion in Neurobiology*, 4, 840–846.
- Schwartz, A. B., & Georgopoulos, A. P. (1987). Relations between the amplitude of 2-dimensional arm movements and single cell discharge in primate motor cortex. *Society for Neuroscience Abstracts*, 13, 244.
- Scott, S. H., & Kalaska, J. F. (1995). Changes in motor cortex activity during reaching movements with similar hand paths but different arm postures. *Journal of Neurophysiology*, 73, 2563–2567.
- Smyrnis, N., Taira, M., Ashe, J., & Georgopoulos, A. (1992). Motor cortical activity in a memorized delay task. *Experimental Brain Research*, 92, 139–151.
- Wise, S. P., & Mauritz, K.-H. (1985). Set-related neuronal activity in the premotor cortex of rhesus monkeys: effects of changes in motor set. *Proceedings of the Royal Society of London. B*, 223, 331–354.