Pergamon

Neural Networks, Vol. 9, No. 3, pp. 397–410, 1996

Printed in Great Britain
0893-6080/96 $15.00 + .00

0893-6080(95)00138-7

CONTRIBUTED ARTICLE

Modeling of Directional Operations in the Motor Cortex: a Noisy Network of Spiking Neurons is Trained to Generate a Neural-Vector Trajectory

ALEXANDER V. LUKASHIN,[1, 2] GEORGE L. WILCOX[2] AND APOSTOLOS P. GEORGOPOULOS[1, 2]

[1] Department of Veterans Affairs Medical Center, Minneapolis and [2] University of Minnesota Medical School

(*Received* 5 *July* 1994; *revised and accepted* 14 *November* 1995)

Abstract—*A fully connected network of spiking neurons modeling motor cortical directional operations is presented and analyzed. The model allows for the basic biological requirements stemming from the results of experimental studies. The dynamical evolution of the network's output is interpreted as the sequential generation of neuronal population vectors representing the combined directional tendency of the ensemble. Adding these population vectors tip-to-tail yields the neural-vector trajectory that describes the upcoming movement trajectory. The key point of the model is that the intra-network interactions provide sustained dynamics, whereas external inputs are only required to initiate the population. The network is trained to generate neural-vector trajectories corresponding to basic types of two-dimensional movements (the network with specified connections can store one trajectory). A simple modification of the simulated annealing algorithm enables training of the network in the presence of noise. Training in the presence of noise yields robustness of the learned dynamical behaviors. Another key point of the model is that the directional preference of a single neuron is determined by the synaptic connections. Accordingly, individual preferred directions as well as tuning curves are not assigned, but emerge as the result of interactions inside the population. For trained networks, the spiking behavior of single neurons and correlations between different neurons as well as the global activity of the population are discussed in the light of experimental findings. Copyright © 1996 Elsevier Science Ltd*

Keywords—Simulated annealing, Gaussian noise, Synaptic weight.

1. INTRODUCTION

The recording of the activity of directionally tuned motor cortical cells in the brain of behaving animals provides a powerful tool for studying the cortical processing that underlies cognitive directional operations (see Georgopoulos et al., 1993 for a review). The activity of a directionally tuned cell is highest for a movement in a particular direction (the cell's preferred direction) and decreases progressively with movements farther away from this direction; the preferred directions observed range throughout the directional continuum (Georgopoulos et al., 1982, 1988). The global directional tendency of the motor cortex is represented by the neuronal population vector, which is a sum of preferred direction vectors, weighted according to the cell's activity (Georgopoulos et al., 1983, 1986). The population vector has proven to be a good predictor for the direction of limb movement under a variety of conditions (Georgopoulos et al., 1993). Moreover, the population vector can be used as a temporal probe of the changing directional tendency of the neuronal ensemble. In order to obtain the time evolution of the population vector, the entire time course of an experiment is divided into short time bins, and the neuronal population vector is calculated at successive intervals during the period of interest. The trajectory of an arm movement during a reaching or drawing task can be accurately predicted by the neural-vector trajectory reconstructed from the time series of neuronal population vectors attached to each other tip-to-tail (Georgopoulos et al., 1988; Schwartz, 1993).

Understanding the neural mechanisms underlying the observed dynamics of motor cortical activity

Acknowledgements: This work was supported by United States Public Health Service grant PSMH48185 to A. P. Georgopoulos, a contract from Office of Naval Research to A. P. Georgopoulos and A. V. Lukashin, and grants of supercomputer resources from the Minnesota Supercomputer Institute to G. L. Wilcox.

Requests for reprints should be sent to A. V. Lukashin, Brain Sciences Center, VA Medical Center 11B, One Veterans Drive, Minneapolis, MN 55417, USA.

requires biologically relevant models for describing the collective dynamical behavior of neuronal ensembles. In this respect, it is useful to consider separately two types of movement control. The first type comprises visually guided reaching and tracing movements. In the framework of corresponding neural network models, at each step of the dynamical evolution, the information concerning future changes is fed into the network by means of time varying external inputs (Bullock & Grossberg, 1988; Burnod et al., 1992; Bullock et al., 1993; Lukashin & Georgopoulos, 1993, 1994b). The second type of movement control is the generation and performance from memory of explicitly defined, learned movement trajectories, such as drawing simple curves. It is reasonable to hypothesize that the dynamical pattern of neural activity of a learned movement is permanently stored in a subset of fixed connections between motor cortical cells so that once it is initiated it unfolds in time, generating the specific motor act (Houk, 1989). In this paper we concentrate on the modeling of this type of motor control. A network of simplified units was proposed to model this kind of directional operation (Lukashin & Georgopoulos, 1994a). The network was trained to generate different neural-vector trajectories, and resulting sets of synaptic connection strengths were analyzed in the light of available experimental data (Georgopoulos et al., 1993). However, a number of important biological requirements were not taken into account. For example, individual neurons were represented by simple input–output devices with sigmoidal activation function, the model did not include noise, the directional preference of cells was imposed externally, and the tuning properties of individual neurons were not compared with those experimentally observed.

The purpose of the present work is to further analyze the dynamical characteristics of populations of directionally tuned neurons in the framework of a substantially extended model. We focus on the following basic requirements derived from experimental data. First, since the activity of neural populations consists of nerve impulses, a relevant model should yield not only global activities and time-averaged firing rates but also spike trains of single neurons and their correlations: for this purpose we use the model of integrate-and-fire neurons. Although the dynamical behavior of a single such neuron is straightforward, the population dynamics of a network of them can be highly complex. Second, the time evolution of excitation and inhibition of cortical cells is longer than the synaptic delays of the circuits involved (Douglas et al., 1989). This means that cortical processing probably does not rely on precise timing between individual synaptic inputs. Indeed, real motor cortical cells do not repeat precisely the temporal spike patterns even though their activities relate to one and the same behavioral task (Georgopoulos et al., 1982). Therefore, a biologically plausible model should include noise that changes the precise timing of synaptic transmission. Third, the external inputs probably do not provide the major signal arriving at a given motor cortical neuron. Instead intra-cortical connections likely provide most of the interactions (Douglas et al., 1989; Keller, 1993). This means that a model describing cortical processing should exhibit sustained dynamical activity due to intrinsic interactions, whereas external inputs would provide only initial activation. Fourth, since changes in the population activity are probably driven by the interactions between cells, it is reasonable to hypothesize that the directional preference of single cells can also emerge as the result of specifically organized connections inside the motor cortex. In other words, inputs from other neurons influence a neuron's preferred direction and tuning function as well as its output connections to motor effector systems. The experimentally observed correlation between the synaptic weight and the difference between preferred directions of the connected neurons (Georgopoulos et al., 1993, 1994; Fetz & Shupe, 1994) and previous modeling studies (Lukashin & Georgopoulos, 1994a) supports this hypothesis.

In this paper we propose a neural network model incorporating the above requirements. A population of fully interconnected spiking neurons in the presence of noise and a small activating external input is trained to generate neural-vector trajectories corresponding to three basic types of two-dimensional movements: left turn, right turn and straight-line motion (after one training trial the network with fixed connections is able to store and generate a one neural-vector trajectory; to generate another trajectory another set of connections should be found by training). We suggest a special training procedure (modified simulated annealing algorithm) that allows for the presence of noise and ensures the robustness of a learned dynamical behavior. The dynamical and structural features of the trained networks are discussed and compared with the results of experimental findings.

2. MODEL

2.1. Integrate-and-Fire Neurons

We use a standard network of integrate-and-fire neurons (see, for example, Atiya & Baldi, 1989; Usher et al., 1993). Consider a fully connected population of N neurons, each one characterized by a time-dependent variable $U_i(t)$ that represents the cell's

potential ($i = 1, \ldots, N$) at the instant of time t. Let $t_i(n)$ be the time when the ith neuron fires the nth spike. During the following refractory period, that is, during the time interval when the potential $U_i(t)$ is below a threshold value U_{thresh}, the neuron integrates over its inputs according to the resistance–capacitance equation:

$$\tau \frac{dU_i}{dt} = -U_i(t) + S_i(t) + E_i(t); \quad U_i(t) < U_{\text{thresh}} \tag{1}$$

under initial condition:

$$U_i[t = t_i(n)] = U_{\text{rest}} \tag{2}$$

where τ is a characteristic time constant, $S_i(t)$ is the sum of the synaptic inputs from all other neurons in the network (see below), $E_i(t)$ is some external signal that models thalamic input, and U_{rest} is the resting value of the potential. Once the potential $U_i(t)$ reaches the threshold U_{thresh}, the neuron fires, sends its output to the other neurons in the population, and resets its potential to the resting value U_{rest}. The instant t the potential $U_i(t)$ reaches the threshold corresponds to the time of the next spike $t_i(n+1)$, after which the integration process (1) resumes. The array of the times $t_i(n)$ with $n = 1, 2, \ldots, n_{\max}$ is the spike train of neuron i.

2.2. Synaptic Inputs

The neurons in the network communicate via exchange of spikes. These interactions are described by the synaptic input term $S_i(t)$ on the right-hand side of eqn (1). Let a postsynaptic neuron i be connected to neuron j through a synapse of efficacy w_{ij} which is positive for an excitatory synapse and negative for an inhibitory synapse. The amplitude of the signal coming to neuron i from neuron j is equal to the efficacy w_{ij}, whereas the time course is described by a response function $r_j(t)$. We approximate the response function by a simple exponential decay (McCormick, 1990) assuming that the signal arrives without a transmission delay. The total synaptic input is calculated by a linear summation over contributions from all neurons in the population. The variability of the spike patterns can be modeled as a noise term added to the right-hand side of eqn (1). Below we assume the effect of noise is proportional to the instantaneous value of the total noiseless synaptic input. Thus, the total synaptic signal $S_i(t)$ received by neuron i is given by the equation:

$$S_i(t) = [1 + \xi(t;\sigma)] \sum_{j=1}^{N} w_{ij} r_j(t) \tag{3}$$

where $\xi(t;\sigma)$ is the Gaussian noise of zero mean and standard deviation σ. To include saturation effects at strong excitation/inhibition levels, we assume, if another presynaptic spike arrives while a residual effect of the previous one still remains, the signal is not allowed to increase above the level assigned for a single spike (Ekeberg et al., 1991). Therefore, the response function $r_j(t)$ is given by the equation

$$r_j(t) = \lambda \exp\left[-\frac{t - t_j(n_{\text{last}})}{\tau_r}\right] \tag{4}$$

where λ is a parameter (we took $\lambda = 1$ in routine calculations), τ_r is a characteristic response time, and $t_j(n_{\text{last}})$ is the instant of the most recent spike in neuron j.

The external signal E_i (eqn (1)) serves to initiate the dynamics of the population. In routine calculations we used the expression $E_i = a + b \cos(\theta - 2\pi i/N)$, where a and b are parameters. This particular cosine distribution of input activations provides initiation of a two-dimensional trajectory in the direction $(\cos\theta, \sin\theta)$. In fact, in routine calculations the angle θ was set at zero degrees; all the simulated movements started out in the same direction. After the appearance of several initial spikes, the external input becomes negligibly small in comparison with the intra-network synaptic signal S_i (see below).

2.3. Neuronal Population Vector and Neural-Vector Trajectories

Following the experimental approach (Georgopoulos et al., 1982, 1983, 1986, 1988, 1993), we use the neuronal population vector and the neural-vector trajectories as the output of the network. Let $\mathbf{C}_i$ be the unit preferred direction vector for the ith cell (calculation of the preferred direction of a cell is described below). The entire time course of a movement trajectory is divided into time bins Δt, and the neuronal population vector is calculated at successive intervals. The neuronal population vector $\mathbf{P}$ taken at the kth time bin is defined by the sum:

$$\mathbf{P}(k) = \sum_{i=1}^{N} V_i(k)\mathbf{C}_i \tag{5}$$

where $V_i(k)$ is the activity of neuron i at the kth time bin:

$$V_i(k) = \frac{\text{number of spikes during } (k-1)\Delta t < t < k\Delta t}{\Delta t}. \tag{6}$$

Since the preferred directions are assumed not to vary

in time after training stops, the changing direction of the neuronal population vector is completely defined by the evolution of activities $V_i(k)$. Adding the population vectors $\mathbf{P}(k)$ tip-to-tail yields the radius vector:

$$\mathbf{R}(k) = \mathbf{P}(1) + \mathbf{P}(2) + \cdots + \mathbf{P}(k-1) + \mathbf{P}(k) \qquad (7)$$

which defines the kth point at the neural-vector trajectory. Since a particular neural-vector trajectory is the result of the network dynamics, and the dynamics are driven by the interactions among members of the population, the set of connection strengths $\{w_{ij}\}$ determines the shape of the emergent trajectory.

2.4. Preferred Directions and Synaptic Weights

In the framework of the model with the firing activity dominantly driven by the intra-network interactions, synaptic connection strengths should be correlated with directional preferences of interconnected neurons. The problem we address is to find a relation between synaptic weights and preferred directions such that the network could learn a given trajectory and obey the rules relating the population vectors to individual neuron directional preferences. To this end we use the following approach. In addition to the set of preferred direction vectors $\mathbf{C}_i (i = 1, \ldots, N)$, we introduce another set of unit vectors $\mathbf{D}_j (j = 1, \ldots, N)$ and assume that the synaptic weights w_{ij} are proportional to the scalar product of vectors $\mathbf{C}_i$ and $\mathbf{D}_j$:

$$w_{ij} = \epsilon(\mathbf{C}_i \mathbf{D}_j) \qquad (8)$$

where ϵ is a constant which determines the range of the variability of synaptic weights. In fact, there is no direct biological motivation underlying expression (8). The following reasons have led us to this conclusion. First, it is important to emphasize that eqn (8) does not suppose a symmetry of the synaptic weights, for the scalar product $(\mathbf{C}_i \mathbf{D}_j)$ is not necessarily equal to $(\mathbf{C}_j \mathbf{D}_i)$, although it might be in some instances; it is the asymmetry of connections that makes it possible to train the network to generate complex trajectories. Second, one may hope that the difference between directions of vectors $\mathbf{C}_i$ and $\mathbf{D}_i$ needed to provide successful training of the network will be quite small; this would result in desired tuning properties of individual neurons (see below). Finally, the representation of connection strength by the scalar product of two vectors essentially reduces the number of parameters to be adjusted during the training procedure ($2N$ directions instead of N^2 synaptic weights in general case).

Substituting eqn (8) into eqn (3) yields the synaptic signal in the form:

$$S_i(t) = [1 + \xi(t; \sigma)]\epsilon(\mathbf{C}_i \mathbf{Q}(t)) \qquad (9)$$

where $\mathbf{Q}(t)$ is a weighted sum of vectors $\mathbf{D}_j$:

$$\mathbf{Q}(t) = \sum_{j=1}^{N} r_j(t) \mathbf{D}_j \qquad (10)$$

and the response function $r_j(t)$ is defined by eqn (4).

The neuronal population vector $\mathbf{P}(k)$ defined by eqn (5) is calculated using the spiking activity of the ensemble; as such, it is a measure of the network's *outcome* calculated at the time bin k. On the other hand, the vector $\mathbf{Q}(t)$ defined by eqn (10) determines the *input* signal received by the population at the instant of time t; it always appears as a component of the scalar product $(\mathbf{C}_i \mathbf{Q})$. If the activity V_i were a continuous linear function of the input S_i, the vector $\mathbf{Q}$ would be equal (in the absence of noise) to the vector $\mathbf{P}$ at the previous time step. However, in the present model, the activity of a neuron is related to synaptic input through integrate-and-fire equations in the presence of noise. The correspondence between the vectors $\mathbf{P}$ and $\mathbf{Q}$ in this case will be analyzed below.

3. TRAINING PROCEDURE

3.1. Error Function

The network of spiking neurons described in the preceding sections was trained to generate a given dynamical behavior in terms of neural-vector trajectories (eqn (7)). To this end, the set of synaptic weights $\{w_{ij}\}$ was adjusted to minimize the root-mean-square error between the desired trajectory shape $\mathbf{R}_d(k)$ and that generated by the network (eqns (5), (6) and (7)):

$$F = \left(\frac{1}{K} \sum_{k}^{K} |\mathbf{R}_d(k) - \mathbf{R}(k)|^2 \right)^{1/2} \qquad (11)$$

with $\mathbf{R}(0) = \mathbf{R}_d(0)$. Equation (11) implies that the total time for tracing a trajectory is equal to $K\Delta t$, where Δt is the time bin. In the framework of our model, the synaptic weight w_{ij} is expressed through the scalar product of vectors $\mathbf{C}_i$ and $\mathbf{D}_j$ (eqn (8)). Therefore, the training is equivalent to the searching of two appropriate sets of vectors $\mathbf{C}_i (i = 1, \ldots, N)$ and $\mathbf{D}_j (j = 1, \ldots, N)$. Since the error function (11) has local minima, we minimized the difference

between the desired and actual trajectories by means of the simulated annealing algorithm (Kirkpatrick et al., 1983). It should be noted that it is not being hypothesized that simulated annealing is what is occurring in the real motor cortex, but instead this is only a convenient means for finding appropriate values of adjustable parameters.

3.2. Simulated Annealing

Specifically, the directions of vectors $\mathbf{C}_i = (\cos\alpha_i, \sin\alpha_i)$ and $\mathbf{D}_j = (\cos\gamma_j, \sin\gamma_j)$ were initialized uniformly in two-dimensional space, that is, $\alpha_i = 2\pi i/N$; $\gamma_j = 2\pi j/N$. The cosine expression used for input activations (the term $E_i = a + b\cos(\theta - 2\pi i/N)$ in eqn (1)) is equivalent to the assignment of the initial trajectory based upon pre-annealing values of variable parameters in the direction $(\cos\theta, \sin\theta)$; in routine calculations $\theta = 0°$. Then the angles α_i $(i = 1, \ldots, N)$ and γ_j $(j = 1, \ldots, N)$ were adjusted to make the error function (11) as small as possible. The optimization scheme was based on the standard Monte Carlo procedure (Aart & van Laarhoven, 1987). New probe values for adjustable angles α_i and γ_j were obtained by small variations in the previous ones. In routine calculations we used: $\alpha_i^{\text{new}} = \alpha_i + (\pi/30)\delta_i$; $\gamma_j^{\text{new}} = \gamma_j + (\pi/30)\delta_j$, where δ_i and δ_j are random numbers uniformly distributed on the interval $[-1, 1]$. The new values of angles were accepted not only for changes that lowered the error function, but also for changes that raised it. The probability of the latter event was chosen such that the system eventually obeyed the Boltzmann distribution at a given "temperature", if the error function (11) is treated as the "energy" of the system. The simulated annealing was initialized at a sufficiently high temperature, at which a relatively large number of state changes were accepted. The temperature was then decreased according to the cooling schedule $T_{n+1} = \beta T_n$, where T_n was the temperature at the nth step and the value β was adjusted to avoid local minima during the annealing. Generally, if the cooling is sufficiently slow $(1 - \beta \ll 1)$ for equilibrium to be established at each temperature, the global minimum, i.e., $F = 0$, can be reached in the limit of zero temperature. Though this cannot be guaranteed in practice when the optimal cooling rate is unknown, we demonstrate below that an approach based on the simulated annealing results in a reasonable algorithm for the training of spiking networks.

3.3. Influence of Noise

It was important to include noise not only to make the model closer to reality, but also to improve the robustness of the learned dynamical behavior. Suppose that a noiseless network of spiking neurons has been trained to generate a trajectory. It is conceivable that during the corresponding dynamics the potential $U_i(t)$ for one of the neurons will come close to but not over the threshold U_{thresh}. Suppose that after the instant of time t the potential will decline due to the inhibition from other neurons. Clearly, the addition of even a small positive element of noise could raise the potential to the threshold so that the neuron fires a spike at the instant of time t. The appearance of an unexpected new spike might cause a perturbation of network behavior; the accumulation of many such perturbations could completely destroy the initially learned trajectory. Indeed, our simulations showed that the network trained to generate a trajectory in the absence of noise could not reproduce the learned task even if a very small amount of noise (the standard deviation of noise $\sigma \sim 10^{-4} - 10^{-3}$) was added after the training. To avoid this instability, the resulting set of connections among neurons should provide dynamics which are insensitive to the local appearance or disappearance of spikes. Such a set of connections may be found if the training procedure allows for the presence of noise.

3.4. Annealing in the Presence of Noise

Consider the zero-temperature limit in which the simulated annealing and gradient descent algorithms become the same. Let new probe values for the adjustable parameters (preferred direction angles) be chosen, and a corresponding new trajectory be calculated in the presence of noise. Also suppose that the new trajectory gives a decrease in the error function (11). In this case, in accordance with the standard algorithm, the probe parameters should be unconditionally accepted. However, the decrease in the error function is not necessarily due to a successful choice of parameters. In the presence of noise, the decrease in the error function might be the result of generation of the trajectory due to an accidentally "succcessful" trial. Another trial using the same set of connections could give a large increase in the error function, and, in this case, the acceptance of new parameters would be a wrong step. This situation would represent inconsistent learning. We suggest a modified version of simulated annealing that works much better than the standard algorithm.

To obtain robustness in the learned dynamical behavior, we use the following approach. If probe values of adjustable parameters are unsuccessful using the standard algorithm, they are rejected, as previously discussed. Consider the opposite case, when a generated trajectory gives an appropriate value of the error function, $F = F^0$, for the new parameters to be accepted. In this case, the change in

parameters is not accepted at once. Instead, the trajectory is generated additionally m times, and corresponding values of the error function $F^1, \ldots, F^m$ are calculated. The new parameters are accepted only if all absolute values of differences $|F^l - F^0|$, with $l = 1, \ldots, m$, do not exceed 3σ, where σ is the standard deviation of Gaussian noise. The idea behind this approach is that, for a large enough number m of repetitions, only those changes in parameters would be accepted that are successful for a "typical" trajectory at a given value of noise. In spite of its simplicity, the modified algorithm is surprisingly efficient. Our simulations showed that only a few repetitions (m = 4–6) are needed to obtain a robust dynamical behavior.

4. RESULTS OF SIMULATIONS AND DISCUSSION

In the present paper we concentrate on the study of dynamical behavior of the noisy network of spiking neurons which is trained to generate one of the three basic types of two-dimensional movements: left turn, straight-line motion and right turn. The network models the movements in terms of neural-vector trajectories (eqn (7)). Table 1 lists the unchangeable parameters of the model that we used in routine simulations. In particular, this set of parameters makes it possible to obtain firing frequencies in a realistic range for motor cortical cells between 10 and 70 impulses per second. The parameters to be found during the training are the sets of vectors $\mathbf{C}_i$ and $\mathbf{D}_j$ $(i,j = 1, \ldots, N)$.

4.1. Neural-Vector Trajectories and Spike Trains

Training sessions for each type of trajectory were repeated 10 times with different sequences of random numbers used in the simulated annealing procedure and for modeling of the noise (in routine calculations the number of steps in the simulated annealing varied from 10^4 to 10^6). Typical results are presented in Figure 1. The left-turn trajectories are shown in the left column. Curve A is the desired trajectory $\mathbf{R}_d(k)$ that must be approximated by the neural-vector trajectory $R(k)$. The two curves labeled B demonstrate the final result for one training session. They correspond to two different noisy realizations of the trajectory with the same set of obtained directions $\{\mathbf{C}_i\}$ and $\{\mathbf{D}_j\}$. The difference between the curves reflects a typical variation in the trajectory shape attributable to the presence of noise. Another two left-turn curves (C) are two noisy realizations of the trajectory that were obtained after a different training session. The difference between C and B reflects the fact that different training sessions give different sets of preferred direction vectors and correspondingly

TABLE 1
Parameters used for the Simulations

Description of parameter	Notation	Value
Number of neurons	N	50
Time step during integration (eqn (1))	—	0.1 ms
Characteristic time constant (eqn (1))	τ	30 ms
Synaptic response time (eqn (4))	τ_r	10 ms
Time bin (eqn (6))	Δt	25 ms
Total time of dynamics (eqn (11))	$K\Delta t$	1000 ms
Threshold potential (eqn (1))	U_{thresh}	0
Resting potential (eqn (2))	U_{rest}	−1
Range of synaptic variability (eqn (7))	ϵ	0.2
External signal: $E_i = a + b\cos(2\pi i/N)$	a and b	0.1
Standard deviation of noise (eqn (8))	σ	0.1
Rate of cooling (simulated annealing)	$1 - \beta$	10^{-5}
Repetition of noisy trajectory	m	5

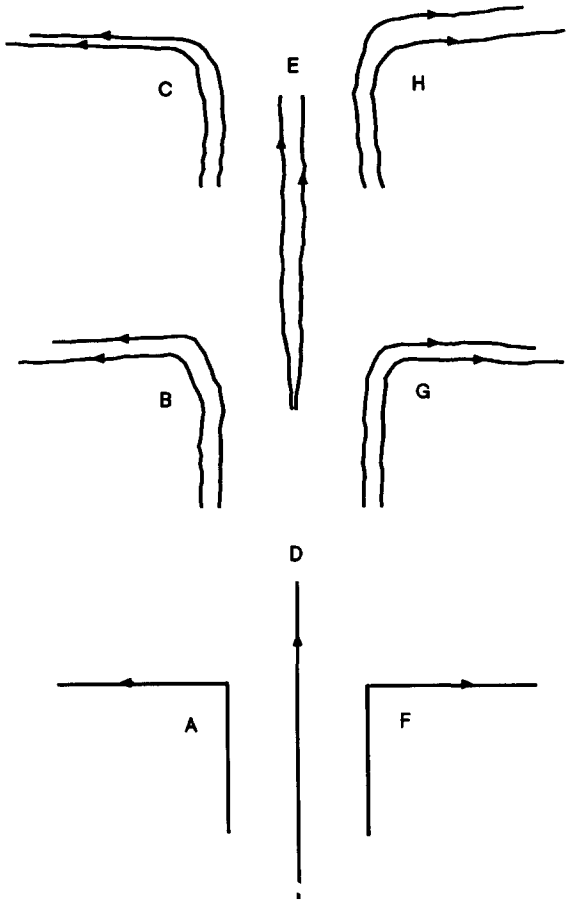

FIGURE 1. The desired trajectories and the neural-vector trajectories generated by the network after training. (Left turn) A is the shape of the desired left-turn trajectory. Two curves in B are two noisy realizations of the left-turn trajectory obtained after one training session. The two curves in C are the result of another training session. (Straight-line motion) D is the desired trajectory. The two curves in E are two noisy realizations of one trajectory. (Right turn) F is the desired trajectory. The two pairs of curves G and H are noisy realizations of the trajectories obtained after two sessions of training the network to generate the right-turn trajectory.

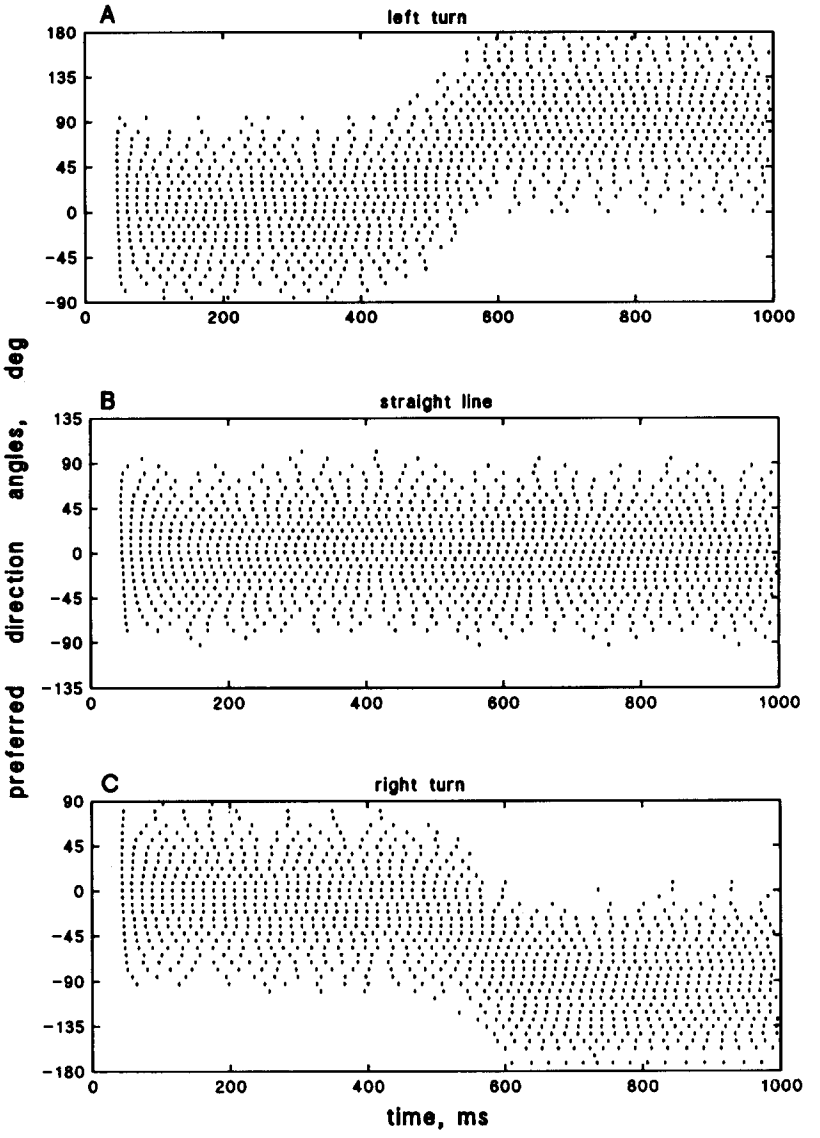

FIGURE 2. The spike trains generated by the network. Rasters A, B and C are the spike trains that underlie left-turn, straight-line and right-turn trajectories, respectively. Time is plotted on the abscissa. Each horizontal row of bars represents the spike train for one neuron. Each bar marks the spiking time in the neuron. The neurons are ordered along the vertical axis by their preferred direction angles. The angle equal to zero corresponds to vertical direction in Figure 1. Only those neurons are represented that reveal spiking activity during the performance of a trajectory by the network. Note that different rasters have different ranges of the engaged preferred directions.

different sets of synaptic connection strengths (eqn (8)). Curve D is the desired trajectory for the straight-line motion. Correspondingly, the two noisy curves (E) are the result of one training session. The right column shows the desired curve (F) and the neural-vector trajectories (G and H) that were obtained after two different training sessions for the right turn. It is seen that qualitatively the trajectories generated by the network reproduce the desired curves well, although they do not correspond to the global minima of the error function F (11). Note that the experimentally measured neural-vector trajectories exhibit the same ability to reproduce the desired curves (Georgopoulos et al., 1988; Schwartz, 1993).

Whereas Figure 1 demonstrates the generated trajectories graphically, Figure 2 shows examples of the spike trains used to calculate the trajectories. In contrast to the raster B that represents the network activity during the straight-line motion, rasters A and C demonstrate essential changes in the activity during the left (A) and right (C) turns. In both cases, those neurons whose preferred directions differ from the direction of the upcoming movement (90° for the left turn and −90° for the right turn) by more than 90° are gradually inhibited after the first 400 ms of the dynamics. At the same time, some of the neurons whose activities are inhibited during the first 400 ms, begin to fire. The type of dynamics generated by the network for left and right turns resembles qualitatively a transition between two quasi-stable states of

the system. The time of stability is about 400 ms, and the transition (the turn itself) lasts about 200 ms. The transition apparently depends on specially organized asymmetries of the connections between neurons (see below).

4.2. Population Clusters

Another useful way to analyze the dynamics of the ensemble of spiking neurons is to construct a time sequence of population clusters. These data are represented in Figure 3 for a left-turn trajectory. The set of preferred directions obtained during training is shown at the top cluster in Figure 3. The directions of the lines correspond to preferred directions of 50 neurons. Note that the resulting quasi-uniform distribution of the preferred directions in space is in agreement with the experimental data (Georgopoulos et al., 1988). To construct the sequence of population clusters, the whole dynamics is divided into time bins equal to 25 ms, and for each time bin, the activity of each neuron is calculated by eqn (6). These activities are arranged in the cluster, so that the direction of each line corresponds to the preferred direction of the neuron, and the length of each line is proportional to the neuronal firing frequency, taken at the time bin. Figure 3 shows 16 such clusters with time bins indicated under each cluster (in milliseconds). The discrete distribution of line lengths reflects the fact that, during the time bins lasting 25 ms, each neuron is able to generate only 0, 1 or, at most, 2 spikes. The thick arrows represent neuronal population vectors $\mathbf{P}(k)$ calculated for the corresponding clusters by eqn (5). Only about half of all neurons fire spikes at a given time bin: namely, those neurons whose preferred directions $\mathbf{C}_i$ do not differ from the direction of vector $\mathbf{P}(k)$ by more than 90°. When the global directional tendency changes,

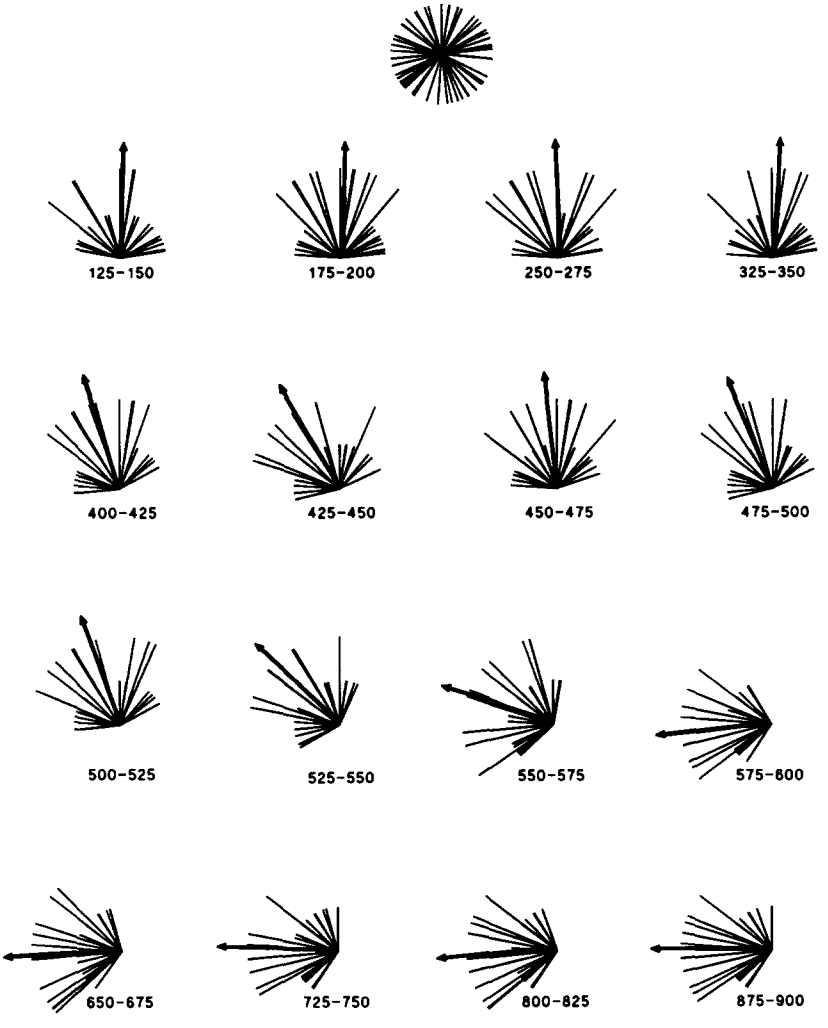

FIGURE 3. The time sequence of the population clusters for the left-turn trajectory. The very top cluster of lines shows the set of preferred directions $\{C_i\}$ obtained as the result of training. The direction of each line is the preferred direction of one of 50 neurons. The other 16 clusters represent the evolution of activity of neurons while the network generates the trajectory. The length of each line corresponds to the activity of a single neuron. The thick arrows show the directions of the postsynaptic neuronal population vector $\mathbf{P}(k)$ at each point in time. Lengths of population vectors are normalized to the same value.

some of these neurons are inhibited, whereas other neurons begin to fire (compare with Figure 2). The transition between two directions (0° during the first 400 ms and 90° during the last 400 ms) takes about 200 ms (clusters from 400 to 600 ms).

Qualitatively, the dynamics of the population clusters is much like those demonstrated by Georgopoulos et al. (1989) and by Schwartz (1993) for the ensembles of motor cortical cells. It should be noted that the network was trained to generate a trajectory of given *shape*. The amount of time really needed for the network to perform the rotation is an intrinsic feature of the model. Though the velocity of rotation is determined by particular values of parameters used in simulations (Table 1), there is an interesting correspondence between the transition lines predicted by the model and those reported experimentally that deserves further study. In the model, the average angular velocity of the rotation of the neuronal population vector during the turn is about 450 deg/s (Figure 3). This value is quite close to experimental values that range from 400 and 700 deg/s (Georgopoulos et al., 1989; Lurito et al., 1991).

4.3. Synaptic Inputs

The dependence of the total synaptic signal $S_i(t) + E_i$ (eqn (1)) on time is shown in Figure 4 for two selected neurons. Both curves correspond to the same left-turn trajectory. The preferred direction angles of the neurons are −47.4° and 128.9°. Since there are no spikes in the network during the first 30–50 ms of the dynamics, the synaptic inputs at this interval are equal to constant external signals E_i depicted as horizontal lines parallel to the x-axis in the beginning of the dynamics. External signals remain constant throughout but diminish in importance once neurons start firing; the value of the synaptic signal then changes dramatically. Essential fluctuations of the synaptic input are due to the arrival of new spikes as well as to the presence of noise. The neuron with a preferred direction angle of −47.4° receives a relatively large positive synaptic signal during the first 350 ms. This input supports the firing activity of the neuron at the level of about 45 imp/s (compare with Figure 2). The value of the synaptic input then gradually decreases due to inhibition from other cells. This period (400–600 ms) corresponds to the time period when the network's global directional tendency turns left. After the input becomes negative, the neuron stops firing (Figure 2). The opposite case, consisting of early inhibition followed by excitation, is demonstrated in Figure 4 for the neuron whose preferred direction angle is 128.9°. It is important to emphasize that, as seen in Figure 4, when the first spiking signals arrive, the neuronal activity is driven by the intra-network connections, whereas the external signal is only required to initialize the dynamics.

4.4. Directional Preference and Population Dynamics

The neuronal population vector was introduced as a formal measure of directional tendency in the motor cortex. In addition, it was shown experimentally (Georgopoulos et al., 1982, 1983, 1986, 1988, 1993) that the activity of a motor cortical cell can be approximated by a linear function of the scalar

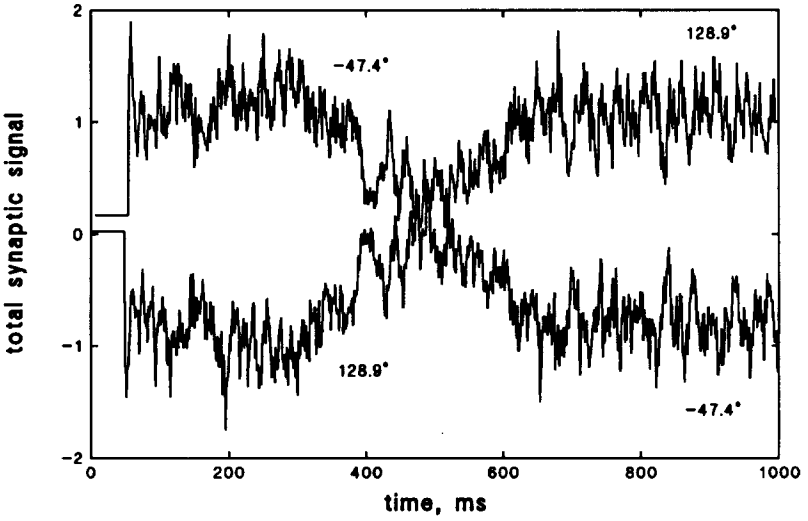

FIGURE 4. The total synaptic signal for two neurons as a function of time during the left-turn trajectory. The changes in the synaptic inputs are shown for two neurons, whose preferred directions are indicated near the curves. The synaptic input is calculated as the sum of the intra-network synaptic signal $S_i(t)$, which includes spiking signals from the rest of the network and the noise, and the constant external signal E_i. The latter values correspond to horizontal lines in the beginning of the dynamics. The total signal is plotted in 1 ms steps (the basic time step of the model is 0.1 ms).

product ($\mathbf{C}_i\mathbf{P}$) of the neuron's preferred direction $\mathbf{C}_i$ and the neuronal population vector $\mathbf{P}$. In other words, the tuning function of a motor cortical cell is represented by the relation $A + B\cos(\varphi - \alpha_i)$, where φ is the angle that characterizes the direction of the neuronal population vector. Our purpose in the present study was to develop a training algorithm that would make it possible to obtain a desired neural-vector trajectory together with desired tuning properties of individual neurons. To this end we use a special choice of connectivity matrix (eqn (8)), where each element w_{ij} is represented as the scalar product between the vectors $\mathbf{C}_i$ and $\mathbf{D}_j$. After training using this form of the connectivity matrix, a correlation emerges between the direction of population vector $\mathbf{P}$ and activities of individual units.

The data presented in Figure 5 by solid circles demonstrate a typical example of how the mean firing rate of directionally tuned neurons correlates, in the framework of the model, with the combined directional tendency of the whole network. Qualitatively, these results are in accordance with the experimental findings mentioned above (dashed line). The activity of individual neurons is highest when the neuron's preferred direction $\mathbf{C}_i$ is close to the direction of the population vector $\mathbf{P}$ and gradually decreases with $\mathbf{C}_i$ farther away from the direction of vector $\mathbf{P}$. The same dependence was observed if the direction $\mathbf{C}_i$ was fixed whereas the direction $\mathbf{P}$ was varied (data not shown). Quantitatively, the difference between experimental and modeling dependencies is substantial only at the tails of the tuning curves. In the model, a neuron is completely inhibited if its preferred direction differs from the direction $\mathbf{P}$ by more than 90°. The deviation of tuning curves obtained by simulations from those experimentally observed is apparently significant from the standpoint of how well the model describes biology. However, this deviation is not important to model performance. Indeed, the performance of network models using the population vector as the outcome of the ensemble operations is only weakly sensitive to the shape of the tuning curve tail (Seung & Sompolinsky, 1993). We have checked (data not shown) that, within the integrate-and-fire model used in the present study, a variation of parameters determining the dynamics of individual neurons including spontaneous firing rate only weakly affected the shape of the tuning curves. We assume an extension of the model might improve the tuning curve.

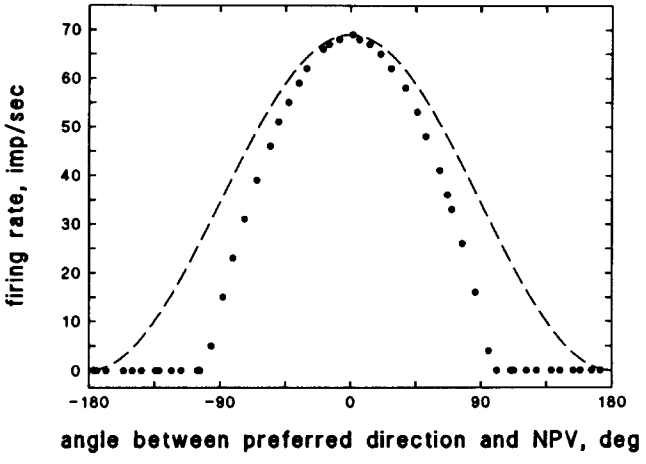

FIGURE 5. Mean firing rate of neurons in the population is plotted against the difference between their preferred directions $\mathbf{C}_i$ and the direction of the postsynaptic neuronal population vector P. The dependence (solid circles) presented here was calculated on the basis of the spike trains shown in Figure 2(B). During the dynamics, the direction of vector P is permanently close to 0°. The abscissa value for each solid circle corresponds to the neuron's preferred direction. The mean firing rate plotted on the ordinate for a neuron was calculated as the total number of spikes during the whole dynamics lasting 1 s. The dashed line represents a dependence that fits most of experimental data: $A + B\cos(\varphi - \alpha_i)$, where α_i is the preferred direction angle, φ is the angle that characterizes the direction of neuronal population vector. Specifically, here we take $\varphi = 0°$, $A = B = 34.5$ imp/s.

A mechanism that underlies the tuning properties of individual neurons could be clarified in the framework of the suggested model. The relation between the preferred directions and the synaptic weights (eqn (8)) yields a special form of synaptic input (eqns (9) and (10)). In accordance with these equations, the synaptic input is a function of the scalar product ($\mathbf{C}_i\mathbf{Q}$) (eqn (9)). The synaptic input, in turn, determines the spiking activity of each neuron (eqn (1)). Therefore, the activity depends on the direction of vector $\mathbf{Q}$. On the other hand, it turns out that the direction of vector $\mathbf{Q}$, which determines the input into a neuron, is close to the direction of population vector $\mathbf{P}$, which describes the network's outcome. This is demonstrated in Figure 6 where the directions of vectors $\mathbf{Q}$ and $\mathbf{P}$ are shown as functions of time for left- and right-turn trajectories. Although the directions do not coincide exactly, they are close to each other throughout the movement. These results explain the fact that the individual neuron's spiking activity can be represented as a function of the population vector $\mathbf{P}$.

4.5. Role of Asymmetric Connections

If the synaptic weights w_{ij} (eqn (8)) are symmetric, that is $(\mathbf{C}_i\mathbf{D}_j) = (\mathbf{C}_j\mathbf{D}_i)$, only macroscopically stationary states of the network are possible (e.g., straight-line neural-vector trajectories). This result can be easily understood by an analogy with networks of simplified, non-spiking units which exhibit stable behavior for symmetric connections (Hopfield, 1984; Atiya & Baldi, 1989). Indeed, Amit and Tsodiks (1992) have shown that on a relatively

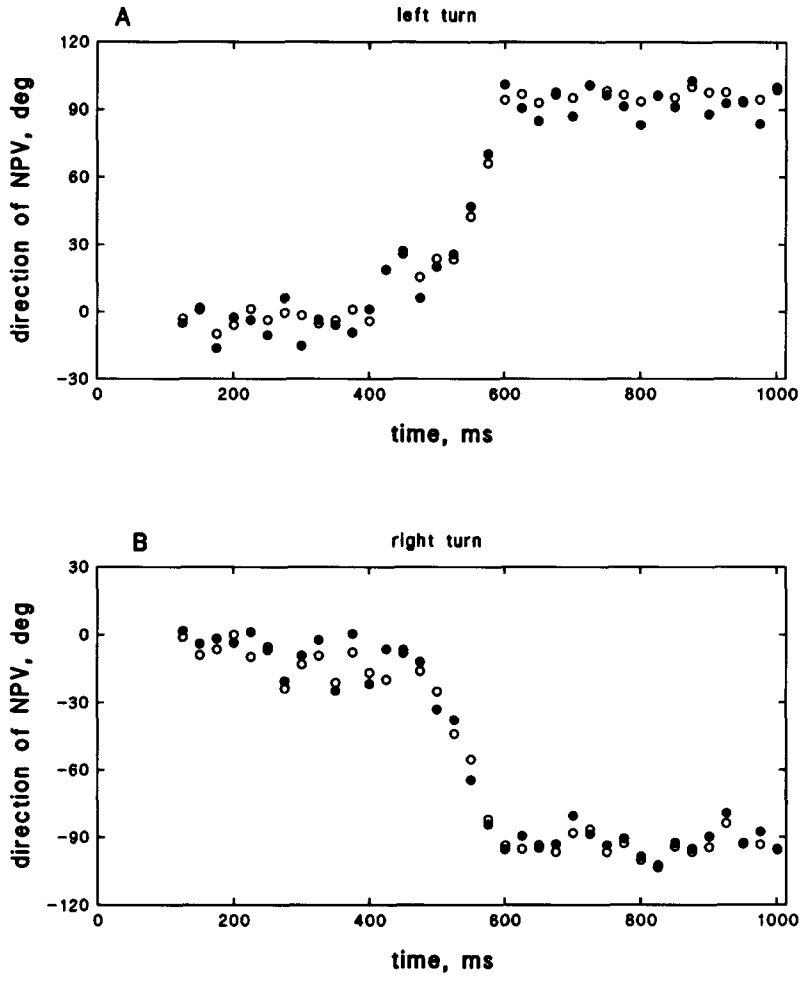

FIGURE 6. Directions of the vectors P and Q as functions of time during the dynamical evolution of networks generating left-turn (A) and right-turn (B) trajectories. The direction of the presynaptic population vector Q (eqn (10)) was initially calculated as a continuous function of time. Then averaged directions were calculated for each 25 ms time bin. These data are plotted as open circles. Solid circles show directions of the postsynaptic population vector P, which is a discrete function of time bins by definition (eqn (5)). The data represented in A for the left-turn correspond to the same trajectory that is analyzed in Figure 3 in terms of population clusters.

long time scale the dynamical behavior of integrate-and-fire neurons can be averaged out and characterized by continuous variables like those used for simplified units.

It is the asymmetry of the synaptic weights that makes possible the generation of a nontrivial dynamical behavior such as a trajectory with an orthogonal turn. In our simulations, the synaptic weights were initialized symmetrically, yielding a straight-line trajectory. During the adjustment of the weights, the asymmetry increased progressively with the decrease in the error function (11). An example of the asymmetry needed for generation of a left-turn trajectory is shown in Figure 7 where the angle between vectors $\mathbf{C}_j$ and $\mathbf{D}_j$ (ranging from 0 to 180°) is plotted against the angle between $\mathbf{C}_i$ and $\mathbf{D}_j$. This presentation of the data was chosen to provide a uniform distribution of points along both abscissa and ordinate. The deviation of points from the main diagonals reflects the level of the asymmetry.

In fact, the role of asymmetry can also be clarified by the analogy with networks of simplified units. A trajectory with a turn could be interpreted as a transition (the turn itself) from one stationary state to another (two quasi-straight-line sections of the path). For the networks of simple units, Sompolinsky and Kanter (1986) and Kleinfeld (1986) have found ways of controlling transitions between stable states. Two types of connections have been introduced; usual symmetric short-time connections that stabilize each state, and long-time (delayed) asymmetric connections that tend to cause transitions. The relationship between the two types of connection determines how long each state stabilizes before changing to the next. Although our model does not include two separate types of connections, the above mechanism is close to that emerged in our model after training. Indeed, as can be seen from the data presented in Figure 7, the asymmetry of connections appears nonrandom, because points are clustered near the main diagonals. This constraint from the asymmetry (i.e., a correlation between weights) tends to stabilize the

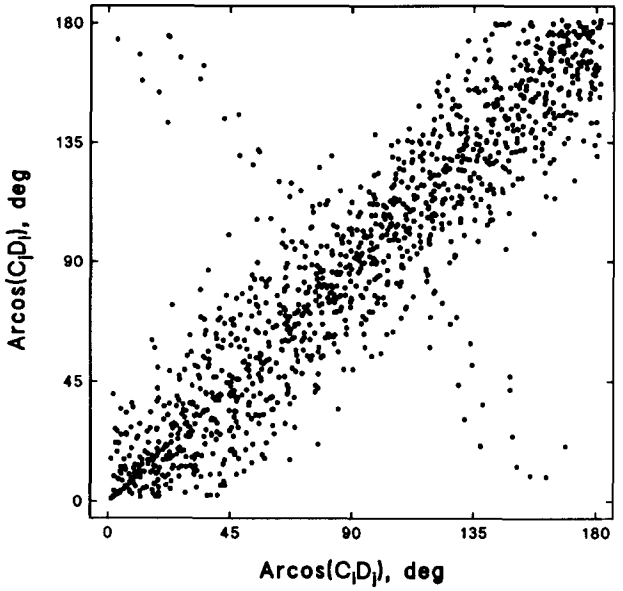

FIGURE 7. Asymmetry of the synaptic weights ensuring the generation of the left-turn trajectory. The abscissa of each point is the angle between vectors C_i and D_i calculated as an inverse cosine function of the corresponding scalar product of the vectors. The ordinate of the point is the angle between C_i and D_i calculated in the same way.

current macroscopic state of the network. In addition, the existence of asymmetry tends to change the dynamics. The relationship between these two tendencies appears to determine the global dynamical features of the model.

4.6. Robustness of the Network's Performance

In routine calculations, the standard deviation σ of the Gaussian noise $\xi(t;\sigma)$ (eqn (9)) presented during training was set at 0.1. This level of noise, on the one hand, makes it possible to avoid an instability in the network's performance, and, on the other hand, does not unreasonably increase the number of steps needed to train the network. (At most, a three-fold increase in training steps was observed when noise was added to the simulations.) A typical value of the error function (11) for trajectories shown in Figure 1 is $F \simeq 1$ whereas different noisy realizations of the same learned trajectory yield differences between error functions of the order of $3\sigma = 0.3$. Nevertheless, our training algorithm does not guarantee 100% stability of the network's behavior, especially if the level of noise is higher than that used during the training. The results of quantitative analysis of this problem are presented in Figure 8.

We used the following approach to check the sensitivity of the model to noise. The synaptic weights obtained as the result of a training session with $\sigma = 0.1$ were fixed and the corresponding trajectory was generated using different amounts of noise, σ ranging from 0 to 0.5. For each σ value the trajectory

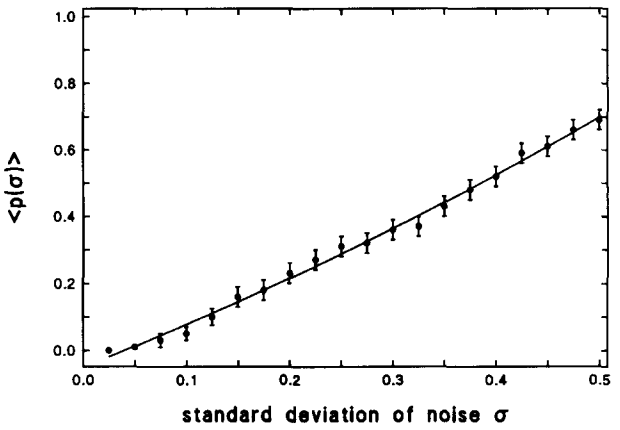

FIGURE 8. The averaged probability of destroying a left- or right-turn trajectory as a function of the noise value. The straight-line trajectories are much less sensitive to the noise and were not taken into account. Points represent the results of simulations; error bars show the standard deviations of mean values; the solid curve is the best fit.

generation was repeated 100 times, and the probability $p(\sigma)$ of destroying the trajectory was evaluated as the number of trials giving the error function $F \geqslant 2$ over the total number of trajectory generations. (Note that the error function values for the trajectories shown in Figure 1 are within the interval $F = 1 \pm 0.3$.) The probability $p(\sigma)$ was then averaged over 20 sets of synaptic weights obtained in training sessions for trajectories with left and right turns. The resulting averaged probability $\langle p(\sigma) \rangle$ is plotted in Figure 8 as a function of σ. For the noise up to the value used in training ($\sigma = 0.1$), the probability of destroying a trajectory is small (less than 5%). As was expected, the probability $\langle p(\sigma) \rangle$ increases with further increases in noise. It is seen, however, that about 50% of trajectories still remain stable even for $\sigma = 0.4$, that is, for a four-fold increase in noise over the value used in training. It is important to note that the network's performance degrades linearly with increasing noise; this property represents the robustness of the learned dynamical behavior.

5. CONCLUSIONS

We have incorporated some previously neglected biological requirements into a formal network that models directional operations in the motor cortex and illustrates their relevance. In particular, we have included the noise that changes the precise timing of synaptic transmission. We demonstrated the feasibility of a modified simulated annealing algorithm to train a noisy network of spiking neurons. Our approach makes it possible to model network behavior in more detail. Examination of this detail reveals the spiking activity of individual neurons, correlations between different neurons and correlations between individual neurons and global activity.

The key point of the model is that the intra-network interactions provide sustained behavior of the population, whereas external inputs are only required to initialize the dynamics. The model reveals the features that are noteworthy if compared with the motor cortex: for example, firing rates are within a reasonable range; the connection strengths are asymmetric, there is an interesting quantitative correspondence between the velocity of the rotation of the neuronal population vector and the same value measured experimentally; the spike patterns vary from trial to trial due to noise; a learned dynamical behavior is robust with respect to the presence of increasing noise; preferred directions adjusted during the training are quasi-uniformly distributed in space; and directional tuning curves are in agreement with experimental data.

Of course, there are biological details that are still neglected. For example, we ignore all events on the level of ion channels as well as all effects of adaptation. Our previous studies (Lukashin & Georgopoulos, 1994b) suggested that these phenomena would not change the ability of the network to perform directional operations. At the same time, the omission of synaptic transmission delays, the lack of inhibitory interneurons, and, more generally, the connection topology of our model may be criticized. The incorporation of these factors is the next obvious step in the modeling of directional operations.

Finally, the suggested model implies that each learned movement is represented by a specific network with fixed connection strengths. In other words, multiple trajectories cannot be stored by the model as presented. The natural question of how these networks should be arranged to control a very large number of possible movements has been discussed in detail by Lukashin et al. (1994) in the framework of a simplified model. It was found that learned movement trajectories of different shapes can be stored in, and generated by, largely overlapping neural networks. The overlapping architecture includes a general-purpose core part, common to all networks, and special-purpose parts, specific for particular trajectories. It was shown (Lukashin et al., 1994) that the percentage of trajectory-specific units needed to generate a certain trajectory may be negligibly small if compared with the common core part. Though these results suggest an efficient mechanism for storing learned motor patterns, the problem of generation of multiple trajectories by a noisy network of spiking neurons requires further investigation.

REFERENCES

Aart, E. H. L., & van Laarhoven, P. J. M. (1987). *Simulated annealing: a review of the theory and applications*. Dordrecht: Kluwer Academic Publisher.

Amit, D. J., & Tsodiks, M. V. (1992). Quantitative study of attractor neural network retrieving at low spike rates. *Network*, **2**(3), 259–273.

Atiya, A., & Baldi, P. (1989). Oscillations and synchronizations in neural networks: an exploration of the labeling hypothesis. *International Journal of Neural Systems*, **1**, 103–124.

Bullock, D., & Grossberg, S. (1988). Neural dynamics of planned arm movements: Emergent invariants and speed-accuracy properties during trajectory formation. *Psychological Review*, **95**, 49–90.

Bullock, D., Grossberg, S., & Guenther, F. H. (1993). A self-organizing neural model of motor equivalent reaching and tool used by a multijoint arm. *Journal of Cognitive Neuroscience*, **5**, 408–435.

Burnod, Y., Grandguillaume, P., Otto, I., Ferraina, S., Johnson, P. B., & Caminiti, R. (1992). Visuomotor transformation underlying arm movement toward visual targets: A neural network model of cerebral cortical operations. *Journal of Neuroscience*, **12**, 1435–1453.

Douglas, R. J., Martin, K. A. C., & Whitteridge, D. (1989). A canonical microcircuit for neocortex. *Neural Computation*, **1**, 480–488.

Ekeberg, O., Wallen, P., Lansner, A., Traven, H., Brodin, L., & Grillner, S. (1991). A computer based model for realistic simulations of neural networks. I. The single neuron and synaptic interaction. *Biological Cybernetics*, **65**, 81–90.

Fetz, E. E., & Shupe, L. E. (1994). Measuring synaptic interactions (technical comments). *Science*, **263**, 1295–1296.

Georgopoulos, A. P., Kalaska, J. F., Caminiti, R., & Massey, J. T. (1982). On the relation between the direction of two-dimensional arm movements and cell discharge in primate motor cortex. *Journal of Neuroscience*, **2**, 1527–1537.

Georgopoulos, A. P., Caminiti, R., Kalaska, J. F., & Massey, J. T. (1983). Spatial coding of movement: A hypothesis concerning the coding of movement by motor cortical population. *Experimental Brain Research Supplement*, **7**, 327–336.

Georgopoulos, A. P., Schwartz, A. B., & Kettner, R. E. (1986). Neuronal population coding of movement direction. *Science*, **233**, 1416–1419.

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 direction 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. V. (1993). Cognitive neurophysiology of the motor cortex. *Science*, **260**, 47–52.

Georgopoulos, A. P., Taira, M., & Lukashin, A. V. (1994). Measuring synaptic interactions (response to technical comments). *Science*, **263**, 1296–1297.

Hopfield, J. J. (1984). Neurons with graded response have collective computational properties like those of two-state neurons. *Proceedings of the National Academy of Sciences USA*, **81**, 3088–3092.

Houk, J. C. (1989) In R. M. J. Cotterill (Ed.), *Models of brain function*. (pp. 309–325). Cambridge: Cambridge University Press.

Keller, A. (1993). Intrinsic synaptic organization of the motor cortex. *Cerebral Cortex*, **3**, 430–441.

Kirkpatrick, S., Gelatt, C. D., & Vecchi, M. P. (1983). Optimization by simulated annealing. *Science*, **220**, 671–680.

Kleinfeld, D. (1986). Sequential state generation by model neural networks. *Proceedings of the National Academy of Sciences USA*, **83**, 9469–9473.

Lukashin, A. V., & Georgopoulos, A. P. (1993). A dynamical

neural network model for motor cortical activity during movement: population coding of movement trajectories. *Biological Cybernetics*, **69**, 517–524.

Lukashin, A. V., & Georgopoulos, A. P. (1994a). A neural network for coding of trajectories by time series of neuronal population vectors. *Neural Computation*, **6**, 19–28.

Lukashin, A. V., & Georgopoulos, A. P. (1994b). Directional operations in the motor cortex modeled by a neural network of spiking neurons. *Biological Cybernetics*, **71**, 79–85.

Lukashin, A. V., Wilcox, G. L., & Georgopoulos, A. P. (1994). Overlapping neural networks for multiple motor engrams. *Proceedings of the National Academy of Sciences USA*, **91**, 8651–8654.

Lurito, J. T., Georgakopoulos, T., & Georgopoulos, A. P. (1991). Cognitive spatial-motor processes. 7. The making of movements at an angle from stimulus direction: studies of motor cortical activity at the single cell and population level. *Experimental Brain Research*, **87**, 562–580.

McCormick, D. A. (1990). Membrane properties and neurotransmitter actions. In G. M. Sheperd (Ed.), *The synaptic organization of the brain*. (3rd ed.). Oxford: Oxford University Press.

Schwartz, A. B. (1993). Motor cortical activity during drawing movement: population representation during sinusoid tracing. *Journal of Neurophysiology*, **70**, 28–36.

Seung, H. S., & Sompolinsky, H. (1993). Simple models for reading neuronal population codes. *Proceedings of the National Academy of Sciences USA*, **90**, 10749–10753.

Sompolinsky, H., & Kanter, I. (1986). Temporal association in asymmetric neural networks. *Physical Review Letters*, **57**, 2861–2864.

Usher, M., Schuster, H. G., & Niebur, E. (1993). Dynamics of populations of integrate-and-fire neurons, partial synchronization and memory. *Neural Computation*, **5**, 570–586.