



Contributed article

Optimization of olfactory model in software to give $1/f$ power spectra reveals numerical instabilities in solutions governed by aperiodic (chaotic) attractors

Hung-Jen Chang*, Walter J. Freeman, Brian C. Burke

Division of Neurobiology, LSA 129 and Department of Molecular and Cell Biology, University of California, Berkeley, USA

Received 17 August 1995; accepted 12 September 1997

Abstract

We present a general connectionist model for an olfactory system. The dynamical behavior of each node (neural ensemble) of the model is governed by a second-order ordinary differential equation (ODE) followed by an asymmetric sigmoidal function, relating the aggregate activity of neurons to system parameters and stimuli from an outside environment. A digital implementation of the general connectionist model simulates the characteristics of a mammalian olfactory system having modifiable synaptic connections and spatio-temporal interactions among neural ensembles. Each of four distributed delay terms is represented by a second-order ODE. A parameter optimization algorithm is an integral component of the model. The parameter optimization discussed in this paper results in aperiodic oscillations having a near $1/f$ -type power spectrum with a peak in the gamma range, simulating the electro-encephalographic (EEG) potentials from the neural olfactory system. Random optimization is used for a rough search in a global parameter domain and the parameter self-adaptation rule serves for fine tuning within a local domain after the global search. By design the model is started from an unstable zero point by an external impulse input. It requires 650–850 ms to pass through an initializing transient before settling into a strange attractor. The attractor persists for at least 1500 ms, which is 7.5–20 times longer than the duration of the maximal stationary states observed in the EEGs and it is stable under perturbation by simulated sensory inputs giving response amplitudes less than 2 times the basal aperiodic activity. However, the optimization to $1/f$ -type activity reveals an inherent limit on digital simulation of chaotic states, owing to attractor crowding such that the size of basins decreases with increasing size of the model, until it approaches the size of digitizing step in computation, here a 64-bit word ($\sim 10^{-16}$). Outputs that are not optimized to approach $1/f$ -type power spectra transit earlier to limit cycle activity, usually well before 2000 ms. The duration of stationarity is increased by randomizing the terminal bit of the 64-bit words representing the state variables. The $1/f$ -type solutions are also exquisitely sensitive to parameter truncation; parameter values must be saved in their full binary form for re-starting. The implications in terms of numerical instability, chaos, attractor crowding and the shadowing theorem are discussed. © 1998 Elsevier Science Ltd. All rights reserved.

Keywords: $1/f$ Power spectrum; Attractor crowding; Chaos; EEG wave; Neural ensemble; Olfactory system; Shadowing; Pseudo-trajectory

Nomenclature

n	number of channels of an olfactory system, i.e. a distributed OB layer with n units
$x_i(t)$	wave amplitude activity of cell ensemble $i \in \{1, 2, \dots, N\}$ at time $t \in \mathcal{R}$
$y_i(t)$	pulse density activity of cell ensemble $i \in \{1, 2, \dots, N\}$ at time $t \in \mathcal{R}$
$r_i(t)$	external stimulus (forcing term) of x_i at time t
w_{ij}	system parameter, which is the connection strength from y_j to $y_i \neq j$
k_i	system parameter, which is the gain for r_i to y_i

LR	set of cell ensemble nodes in one of the layers of the olfactory model, i.e. $LR \in \{PG, OB, AON, PC\}$
$Q(x_i, q^{(LR)})$	sigmoidal I/O transformation function for x_i with a system parameter, $q^{(LR)}$, defining its least upper bound
DL	set of four delay feedback nodes
a	0.220/ms, which reflects the slower real rate constant of x_i
b	0.720/ms, which reflects the faster real rate constant of x_i

1. Introduction

There are two steps to model a very large-scale neural system. The first is to construct a set of parametric equations, which incorporate the anatomical synaptic connections and physiological interactions among neural

* Requests for reprints should be sent to Huang-Jen Chang, Department of Molecular and Cell Biology, University of California, Berkeley, CA 94720, U.S.A.; E-mail: hjchang@icsib28.icsi.berkeley.edu.

ensembles that are perturbed by external inputs. The second is to derive quantitative rules, which direct self-adjustment of the values of parameters of the model from its searching history. The automation of parameter optimization in large-scale biologically plausible cortical networks opens a new approach to investigate sensorimotor operations in brains (Freeman and Shimoide, 1994; Chang and Freeman, 1996). By using biological data as the criteria, we optimize parameters in a neural ensemble model used to simulate output of the olfactory system for specified input and for basal activity with no input.

An early model simulating a biological neural system was a relatively lower dimensional chaotic system (Freeman, 1987) in which slight variation of initial conditions of

variables or system parameters led to diverse output patterns. The results from computer simulations are thus highly dependent on the numerical precision and differences in machines, ODE solvers, programming languages, etc. Instead of showing the values of parameters used to simulate experimental data, it makes more sense to report the optimization algorithm, including specification of performance criteria, for researchers to generate the data by using their own software and hardware. Both the parametric mathematical model and its parameter optimization algorithm, based on the attendant activity perturbation dynamics, should be regarded as integral components of a realistic neural model.

The parameter optimization algorithm, which leads the

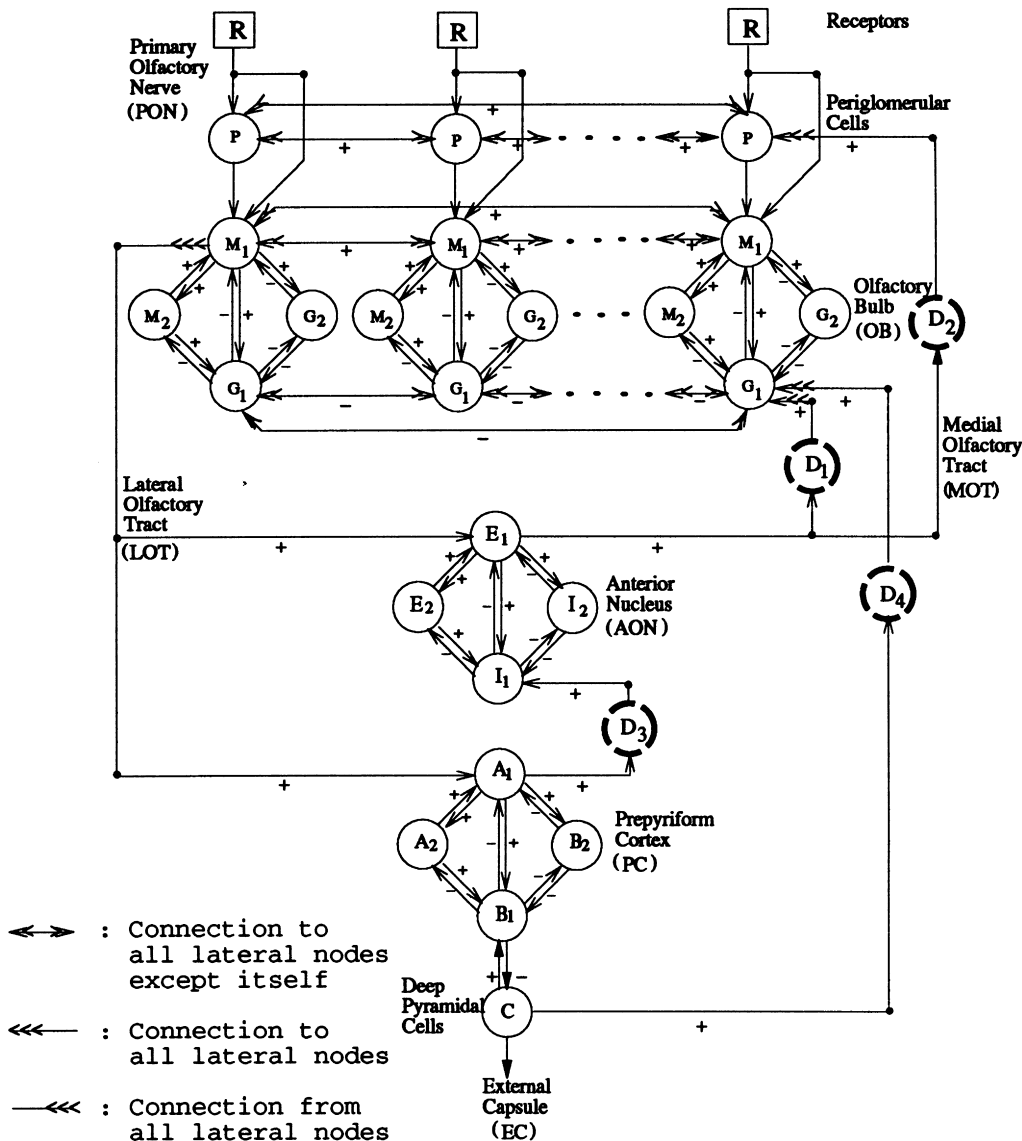


Fig. 1. This topological diagram specifies the connections among neural ensembles of the olfactory system. The olfactory bulb (OB) and prepyriform cortex (PC) are distributed layers of coupled oscillators. The anterior olfactory nucleus (AON) is a control nucleus relaying centrifugal modulatory commands. These three oscillators have incommensurate characteristic frequencies. Only the OB is at present modeled as an array. Each node stands for a second-order ODE for the single ensemble dynamics. An input from receptors (R) goes to periglomerular (P) and mitral cells (M). The mitral cells transmit to granule cells (G) and to AON and PC, from which the final output is sent to the external capsule (EC) from deep pyramidal cells (C), as well as back to the OB and AON.

model to operate in a domain of aperiodic solutions simulating the olfactory EEG, includes two techniques: random optimization (Matyas, 1965) and error propagation (Chang and Freeman, 1996). The first serves for a rough searching in a global parameter domain and the second serves for fine tuning within the local domain from the global search. This optimization algorithm, including specification of performance criteria should be regarded as an integral component of a realistic neural model, although it is not unique. The results reported here reveal a significant barrier to implementation of a digital embodiment of the model, owing to the instability of numerical solutions of the equations.

1.1. Architecture of a model of an olfactory system

The central olfactory system consists of the periglomerular layer (PG), the bulb (OB), anterior olfactory nucleus (AON) and prepyriform cortex (PC). Each part is composed of cell ensembles, which are populations of excitatory or inhibitory neurons. Fig. 1 shows a schematic diagram of principal types of neurons, pathways and synaptic connections in the olfactory mucosa, bulb and cortex. Each node represents an ensemble in a local neighborhood. The dynamics is represented by a linear second-order equations to model cable delay and passive membrane delay. The linear part is followed by a static sigmoid nonlinear function.

The ensemble rate constants, a and b , of each node were fixed throughout at physiologically determined values (Freeman, 1975). Length constants were obviated by using a globally connected network. Gain constants, $q^{(LR)}$, for the asymptotic maximum of the sigmoidal I/O relationship (Fig. 2) were fixed at a common value (Eeckman and Freeman, 1991) within the periglomerular (P) array in the PG and at other values in the mitral (M) and granule (G) arrays in the OB, the excitatory (E,A) and inhibitory (I,B) neurons in the AON and PC and the deep pyramidal cells (C) in the PC. The connection strengths between nodes

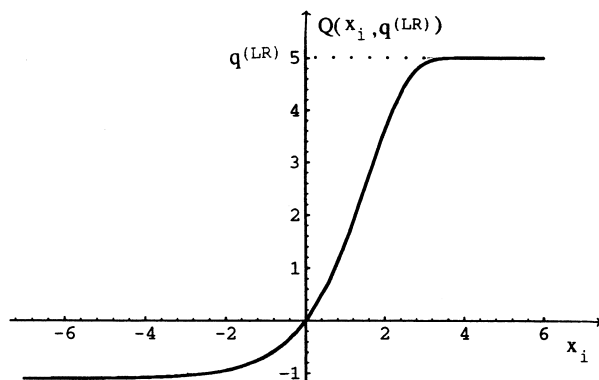


Fig. 2. Asymmetric sigmoidal I/O transformation function [$q^{(LR)} = 5.000$]. This function was derived from the Hodgkin–Huxley system (Freeman, 1979) and holds for all parts of the central olfactory system (Eeckman and Freeman, 1991).

within the PG, OB, AON and PC and the feed-forward and feed-back gains between them, are the system parameters modeling the synaptic interactions of neural ensembles. Other parameters required to model the full set are distributed feedback delays. We have been shown empirically that the solutions of the model suffice to simulate versatile EEG patterns by proper evaluation of all the parameters (Freeman, 1987, 1992; Yao and Freeman, 1990).

The asymmetric sigmoid function (Fig. 2) has been derived from the Hodgkin–Huxley equations (Freeman, 1979) and evaluated by fitting it to the normalized probability density conditional on EEG amplitude of single and multiple unit firing of neurons in the PG, OB, AON, PC and entorhinal cortex (Freeman, 1975; Eeckman and Freeman, 1991). The asymmetry is necessary for the input-dependent increase in gain that is required to simulate bursts of oscillation on inhalation. The function has only one parameter, $q^{(LR)}$, to specify slope, asymptotes and zero operating point, whereas alternatives such as the logistics curve, the arctangent and a cubic spline cell require three parameters.

1.2. The problem of parameter optimization

Because the synaptic connection weights cannot be directly measured from experiments, we optimize these values by simulating with the model the biologically measured activity (endogenous EEG waves) and average evoked potentials (AEPs impulse response evoked by electrical stimulation). Owing to a high dimensional parameter space and thus high complexity of parameter controls for the entire system, it is unrealistic to optimize all the parameters at the same time. We take a step-by-step approach to overcome this difficulty (Freeman, 1975; Freeman and Shimoide, 1994; Shimoide and Freeman, 1995).

1. Fix the values of parameters that can be measured experimentally, e.g. the values of passive membrane rate constants and conduction delays in the open loop, noninteractive state under deep anesthesia (Freeman, 1975).
2. Partition the olfactory system (e.g. the PG, OB, AON and PC) and optimize the values of their internal parameters with respect to AEPs in their near-linear ranges at their characteristic frequencies. (Freeman and Shimoide, 1994; Chang and Freeman, 1996).
3. Fix the values of parameters which have been determined and optimize the values of synaptic connection parameters between the parts with reference to the EEGs and multiple unit activity of the four parts.

In this paper, we focus on the third step. One of observed characteristics of EEGs is a $1/f$ -type power spectrum. Averaged auto and cross spectra for the OB and PC at rest and of the entorhinal cortex and dentate gyrus of hippocampus (DG), show that the log power decreased nearly linearly with log frequency. Temporal frequency analysis

on pre- and post-stimulus of neocortical EEG segments for prepyriform, visual, somatic and auditory recordings also reveal $1/f$ -type power spectra (Barrie et al., 1996). Thus, the $1/f$ Property becomes our criterion for the optimization. Our aim is to derive an analytical rule and determine the values of parameters to match the characteristic from normal neuro-physiological data.

In Section 2, a mathematical model of the olfactory system is presented. The parameter optimization rule is addressed in Section 3. Simulation results shown in Section 4 employ the mathematical analyses in the previous sections.

2. Dynamics of the olfactory neural ensemble model

A massively parallel distributed architecture with multiple layers can be used to describe the olfactory neural system (Freeman, 1974). We first express a vector–matrix representation of the olfactory neural dynamics and next give an example to show how it is derived. The definitions of variables and parameters are given in nomenclature.

2.1. Vector–matrix representation of the olfactory model

The dynamical behavior of each cell ensemble of the olfactory system can be governed by:

$$\frac{1}{\Phi_i}[\ddot{x}_i(t) + \Psi_i \dot{x}_i(t) + \Phi_i x_i(t)] = \sum_{j \neq i}^N w_{ij} y_j(t) + k_i r_i(t), \quad (1)$$

a general connectionist model (Chang and Freeman, 1996), where Φ_i and Ψ_i are derived from experimental rate constants. A single (double) dot over a variable means the first (second) derivative of the variable with respect to time. The variable, x , represents dendritic potential, as it is evaluated by extracellular measurements of EEGs. The variable y represents multineuronal pulse density as it is evaluated by multiunit recording. Both variables are continuous in the cortex. They are discretized by experimental measurement. Conversion from pulse density, y , to wave amplitude, x , is implicit in the synaptic weights. Conversion from wave amplitude, x , to pulse density, y , is implicit in the sigmoidal function, $Q(x, q^{(LR)})$ (see Fig. 2 with $q^{(LR)} = 5.000$). The pulse–wave conversion is kept in its linear range by the sigmoid (Freeman, 1975). The variable r stands for an external stimulus. Physiological experiments confirm that a linear second-order derivative is an appropriate choice.

By the topological diagram of Fig. 1, the n -channel olfactory model is implemented by Eq. (1) in vector–matrix notation with the specifications listed in Appendix A. This notation allows us to concisely present the olfactory model and its corresponding dynamics of parameter optimization (see Section 3).

2.2. ODEs for the periglomerular cell ensemble

To show how to evaluate Eq. (1) in vector–matrix representation, we begin with the dynamics of the P cell ensemble in the PG layer:

$$\begin{aligned} & \frac{1}{ab}[\ddot{P}_\mu(t) + (a+b)\dot{P}_\mu(t) + abP_\mu(t)] \\ &= \frac{W_{(PPL)}}{n-1} \sum_{\nu \neq \mu}^n Q(P_\nu(t), q^{(PG)}) + k_{(PR)} R_\mu(t) \\ & \quad + w_{(PD_2)} \int_0^t K(\tau) Q(E_1(\tau), q^{(AON)}) d\tau \end{aligned} \quad (2)$$

The RHS of Eq. (2) represents the summed input to a P neural ensemble from receptor R, from the other P-ensembles in the same layer and from delayed outputs of ensembles in the AON layer. Here, $K(\tau) \geq 0$ is used to simulate the distributed delay feedback from cell ensemble $E_1(t)$. It is a unimodal and normalized function, which means that for a current time $t \geq 0$ and $0 \leq \tau \leq t$, $K(\tau)$ has only one maximum with $K(0) = K(t) = 0$ and $\int_0^t K(\tau) d\tau = 1$. The LHS of Eq. (2) is the inside change of the ensemble due to the net input.

Let $t - T_2^{(s)}$ and $t - T_2^{(e)}$ be delay starting time and delay ending time, respectively, and take the kernel function $K(\tau)$ as:

$$K^{(\text{step})}(\tau) = \frac{1}{T_2^{(s)} - T_2^{(e)} + 1} \sum_{k=t-T_2^{(e)}}^{t-T_2^{(s)}} \delta(\tau - k)$$

i.e. a difference of two discrete step functions. The delay feedback term in Eq. (2) becomes

$$\frac{w_{(PD_2)}}{T_2^{(s)} - T_2^{(e)} + 1} \sum_{\tau=t-T_2^{(s)}}^{t-T_2^{(e)}} Q(E_1(\tau), q^{(AON)}) \quad (3)$$

The olfactory model is thus described by a set of differential-delay equations, which is the dynamics that Yao and Freeman studied in 1990. It is, however, biologically implausible, owing to distributed arrival times in multi-axonal tracts due to distributed conduction velocities and delays and it is also numerically error prone in digital simulations, due to coarse graining of distributed delays by the computational time step. By using a numerical solver for an ODE set, the actual integration time step is internally adjusted according to the slope at that moment. To have an accurate implementation on a numerical integration of Eq. (2) with $K(\tau) = K^{(\text{step})}(\tau)$, we should offer these data of $E_1(\tau)$ at each correct moment, but it is technically impossible.

In this paper, to model temporal distributed axonal delays we express the kernel function $K^{(\text{step})}$ by a difference of two exponential functions:

$$K^{(\text{exp})}(\tau) = \frac{1}{T_2^{(s)} - T_2^{(e)}} \left[\exp\left(-\frac{t-\tau}{T_2^{(s)}}\right) - \exp\left(-\frac{t-\tau}{T_2^{(e)}}\right) \right]$$

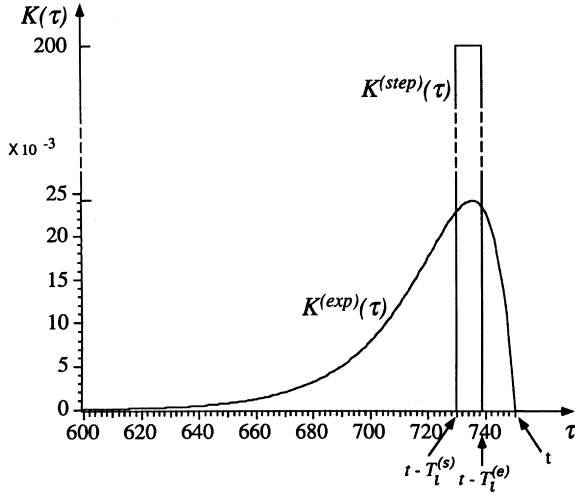


Fig. 3. Kernel function for delayed feedback (at time $t = 750$ ms, $T_1^{(s)} = 20.000$, $T_1^{(e)} = 11.000$ ms).

as shown in Fig. 3. Obviously, $K^{(exp)}(\tau)$ has the unique maximum at $\tau = t - T_m$ with

$$T_2^{(e)} < T_m = \frac{T_2^{(s)} T_2^{(e)}}{T_2^{(s)} - T_2^{(e)}} \ln \left(\frac{T_2^{(s)}}{T_2^{(e)}} \right) < T_2^{(s)}$$

and $K^{(exp)}(0) \approx 0$ (when t is large) and $K^{(exp)}(t) = 0$. In addition, because of:

$$\int_0^t K^{(exp)}(\tau) d\tau = 1 + \frac{1}{T_2^{(s)} - T_2^{(e)}} \left[T_2^{(e)} \exp\left(-\frac{t}{T_2^{(e)}}\right) - T_2^{(s)} \exp\left(-\frac{t}{T_2^{(s)}}\right) \right],$$

$K^{(exp)}(\tau)$ is a normalized function. (The second term becomes negligible when t is large.) $K^{(exp)}$ is thus a temporally smooth version of $K^{(step)}$. Through Laplace transformation, we then see that Eq. (2) with $K(\tau) = K^{(exp)}(\tau)$ is mathematically equivalent to:

$$\begin{cases} \frac{1}{ab} [\ddot{P}_\mu(t) + (a+b)\dot{P}_\mu(t) + abP_\mu(t)] \\ = \frac{W_{(PPL)}}{n-1} \sum_{v \neq \mu}^n Q(P_v(t), q^{(P)}) + k_{(PR)} \tau_\mu(t) + w_{(PD_2)} D_2(t), \\ T_2^{(s)} T_2^{(e)} \left[\ddot{D}_2(t) + \left(\frac{1}{T_2^{(s)}} + \frac{1}{T_2^{(e)}} \right) \dot{D}_2(t) + \frac{1}{T_2^{(s)} T_2^{(e)}} D_2(t) \right] \\ = Q(E_1(t), q^{(AON)}), \end{cases}$$

with $D_2(0) = \dot{D}_2(0) = 0$.

Moreover, because the Fourier transformation of $D_2(t)$ is:

$$\mathcal{F}[D_2(t)] = H(\omega) \mathcal{F}[Q(E_1(t), q^{(AON)})]$$

with

$$H(\omega) = \frac{(1 - T_2^{(s)} T_2^{(e)} \omega^2) - i(T_2^{(s)} - T_2^{(e)})\omega}{(1 - T_2^{(s)} T_2^{(e)} \omega^2)^2 + (T_2^{(s)} - T_2^{(e)})^2 \omega^2}, \quad (4)$$

the delayed feedback node acts as a low-pass filter. Using the same procedure to model the other delayed feedback loops, the ‘infinite dimensional’ differential-delay equations (Yao and Freeman, 1990) convert into finite dimensional ODEs. The dynamical model expressed in Section 2.1 and Appendix A is thus constructed. We call it the low-pass filter olfactory model.

3. An optimization technique for 1/f power spectra

We combine global and local search processes for parameter optimization. The global search has long jumps to avoid being trapped in a local minimum. The local search is a gradient descent to a local minimum.

3.1. Parameter optimization rule for local searching

By way of spatio-temporal error propagation, we obtain the parameter self-adjusting rule (Chang and Freeman, 1996). The criterion is to match the power spectrum of the model to the desired 1/f-type power spectrum (in the sense of minimizing a pre-defined error function within a frequency domain Ω).

We decompose each second-order ODE to two first-order ODEs. Let $l \in \{0, 1\}$ and $x_i^{(l)}(t) \triangleq x_i(t)$ and reformulate Eq. (1) as:

$$\begin{pmatrix} \dot{x}_i^{(0)}(t) \\ \dot{x}_i^{(1)}(t) \end{pmatrix}_{2N \times 1} = \begin{pmatrix} x_i^{(1)}(t) \\ -A_i x_i^{(1)}(t) - B_i x_i^{(0)}(t) + B_i \sum_{j \neq i}^N w_{ij} y_j + B_i k_i r_i(t) \end{pmatrix}_{2N \times 1}. \quad (5)$$

A time sequence $x_i^{(l)}(t)$ is then generated by Eq. (5) for t from $t_i \geq 0$ to $t_f > t_i$. Its power spectrum is $S_i^{(l)}(\omega) \triangleq |X_i^{(l)}(\omega)|^2$ with $X_i^{(l)}(\omega) = \mathcal{F}[x_i^{(l)}(t)]$ (a Fourier transformation of $x_i^{(l)}(t)$) for t from a state transition $t_{st} \geq t_i$ to the next state transition $t_{nst} \leq t_f$.

To derive parameter optimization rules, in the first step we have to choose an objective function E to be minimized in terms of power spectra of actual outputs of the mathematical model $S_i^{(l)}(\omega)$ s and its targets $T_i^{(l)}(\omega)$ s. There are only two requirements for the choices: $E \geq 0$ and $E = 0$

if $S_i^{(l)}(\omega) = T_i^{(l)}(\omega)$ for all $l \in \{0, 1\}$, $\omega \in \Omega$ and $i \in \{1, 2, \dots, N\}$. Here, $F(S_i^{(l)}(\omega), T_i^{(l)}(\omega))$ is an error function, which can be from the least-square fit, relative entropy, etc. (Kullback, 1978; Chang and Freeman, 1996). We also introduce a weight function $\eta_i^{(l)}(\omega)$ for $S_i^{(l)}(\omega)$ achieving $T_i^{(l)}(\omega)$ at frequency ω . For instance, we may have:

$$\eta_i^{(0)}(\omega) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(\omega - \bar{\omega})^2}{2\sigma^2}\right) \text{ and } \eta_i^{(1)}(\omega) = 0$$

to suppress the output time series of $x_i(t)$ from having too strong frequency components around $\omega = \bar{\omega}$ (see also Section 4 for the $\eta_i^{(l)}(\omega)$ we used in the simulations). Thus, the objective function $E = E^{(0)} + E^{(1)}$ with the form:

$$E^{(l)} = \sum_{i=1}^N \int_{\Omega} \eta_i^{(l)}(\omega) F(S_i^{(l)}(\omega), T_i^{(l)}(\omega)) d\omega. \quad (6)$$

In the second step, we derive dynamics of the network of perturbation variable $\lambda_i^{(l)}(t)$ [Eq. (8)] from the dynamics of the network of ensemble variable $x_i^{(l)}(t)$ [Eq. (5)]. We need to define and distinguish two kinds of partial differentiations with respect to $x_i^{(l)}(t)$.

(1) The derivative which accounts for the corresponding change of E due to an arbitrarily small change of $x_i^{(l)}(t)$ at some particular time t with everything else fixed; that is, the conventional partial derivative:

$$\frac{\partial E}{\partial x_i^{(l)}(t)} = p_i^{(l)}(t) dt \quad (7)$$

with

$$p_i^{(l)}(t) = \sqrt{\frac{2}{\pi}} \int_{\Omega} \eta_i^{(l)}(\omega) \frac{\partial F(S_i^{(l)}(\omega), T_i^{(l)}(\omega))}{\partial S_i^{(l)}(\omega)} \times \text{Re}[X_i^{(l)}(\omega) \exp(-i\omega t)] d\omega$$

from Eq. (6).

(2) The other derivative which accounts for the corresponding change of E due to an arbitrarily small change of $x_i^{(l)}$ at time t , when this change propagates throughout the trajectories of other ensembles in the entire time interval that is denoted by $\lambda_i^{(l)}(t) = (\partial^+ E) / (\partial x_i^{(l)}(t))$, an ordered partial derivative (Werbos, 1974, 1988).

The perturbation dynamics are then:

$$\begin{pmatrix} \dot{\lambda}_i^{(0)}(t) \\ \dot{\lambda}_i^{(1)}(t) \end{pmatrix}_{2N \times 1} = \begin{pmatrix} B_i \lambda_i^{(1)}(t) - \frac{\partial y_i(t)}{\partial x_i^{(0)}(t)} \sum_{j \neq i} B_j \omega_{ji} \lambda_j^{(1)}(t) - p_i^{(0)}(t) \\ A_i \lambda_i^{(1)}(t) - \lambda_i^{(0)}(t) - p_i^{(1)}(t) \end{pmatrix}_{2N \times 1} \quad (8)$$

with boundary conditions $\lambda_i^{(0)}(t_{nst}) = \lambda_i^{(1)}(t_{nst}) = 0$ for all i s. The complementary relationship of network structures between the two dynamics is expressed in Fig. 4.

We can next relate the perturbation of E with respect to parameters w_{ij} and k_i to $\lambda_i^{(l)}(t)$ by the following Eq. (9). Because of symmetric property of the network structure, assume that $\omega_{i_1 j_1}, \omega_{i_2 j_2}, \dots$ and $w_{i_1(\omega) j_1(\omega)}$ all have the value of the i th distinguishable connection strength $\tilde{\omega}_i$. Similarly, $k_{i_1} = k_{i_2} = \dots = k_{i_l(k)} = \tilde{k}_i$. Following this, the derivative which accounts for the corresponding change in E due to an arbitrarily small change of the parameter, p_i over the time interval from t_{st} to t_{nst} , is:

$$\frac{\partial E}{\partial p_i} = \int_{t_{st}}^{t_{nst}} \sum_{\kappa=1}^{L^{(p)}} [B_{i_{\kappa}} z_{\kappa}(t) \lambda_{i_{\kappa}}^{(1)}(\tau)] d\tau \quad (9)$$

where

$$z_{\kappa}(t) = \begin{cases} y_{j_{\kappa}}(t), & \text{when } p \text{ is } \tilde{\omega} \\ r_{i_{\kappa}}(t), & \text{when } p \text{ is } \tilde{k} \end{cases}$$

After obtaining the gradients, we can apply a proper optimization algorithm to derive the adaptation rules for parameters. Here, a gradient descent approach allows us to adjust the values of weight parameters according to $p_i^{(new)} = p_i^{(old)} + \Delta p_i$, with:

$$\Delta p_i = -C \frac{\partial E}{\partial p_i} \quad (10)$$

where the constant C governs a learning rate. In summary, for an interconnected neural ensemble model of the olfactory system described by Eq. (1), if the system parameters are updated cyclically at $t = t_{nst}$ according to Eq. (10) with the gradients given in Eq. (9), then the connection

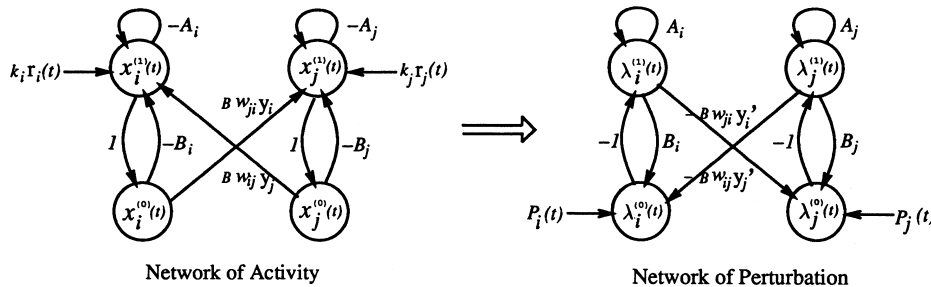


Fig. 4. Reformulation of dynamics in Eq. (1) for ensemble activity and activity perturbation from second-order to pairs of first-order ODEs. The upper loops resulting from the reformulation do not imply self-excitation, which does not occur in cortical neuropil, the dense layered fabric of axons, dendrites and synapses.

strength parameters can be locally optimized to give the output of the model a $1/f$ -type power spectrum.

The model in Eq. (1) and its parameter optimization rule [Eqs. (8)–(10)] are generalizations of the multilayer perceptrons and the back-propagation rule. The detail is shown in Appendix B. This convergence supports the correctness of the derivations and provides alternative approach to derive the back propagation rule (Chang and Freeman, 1996).

3.2. Approach to global optimization of parameters

The parameters which have been determined in the early stage are fixed to reduce the complexity/dimension of the parameter searching space. From neurobiological suggestions, we set initial guesses and a global search domain for these parameters to be optimized. In the global searching, we generate a set of random numbers with certain distributions for parameters updating and test on the updating whether the updated parameters are in the search domain (Matyas, 1965; Solis and Wets, 1981). Next, we numerically integrate Eq. (5) and calculate the corresponding error by Eq. (6). When this error is sufficiently small for transition to local tuning, we use the local optimization rule presented in Section 3.1 for the next parameter adaptation. In the local tuning, we numerically integrate Eq. (5). We set the boundary for $\lambda_i^{(0)}(t_{nst}) = \lambda_i^{(1)}(t_{nst}) = 0$ and integrate Eq. (8) from t_{nst} back to t_{st} . At the same time, we also integrate Eq. (9) to get the gradients for parameters and thus adapt these parameters according to Eq. (10). When a difference of the successive error or parameter vector is small enough, we revert to the global search. The whole process continues until a stop criterion (defined in Section 4.1) is met.

Unlike a traditional random optimization technique, in which the parameters are always updated randomly, we

update the parameters randomly only if a local optimum of an error function is reached. We choose the random distribution as a Gaussian function. Through the experience of several simulation runs, we are able to locate the appropriate mean and standard deviation. However, it is, in general, heuristic to determine the distribution functions which can quickly direct converge to the basin of a global optimum.

4. Simulations

The technique developed in Section 3 were used to optimize the parameters of the olfactory model with four low-pass filters representing distributed feedback delays. The numerical integration was performed by Livermore solver for ODEs with time step $\Delta t = 1.0$ ms in all cases. All the simulations were performed on Macintosh II computers.

We studied impulse responses with:

$$r_i(t) = \begin{cases} 1, & \text{for } i = 1 \text{ or } n + 1, \text{ and } 0 \leq t \leq \Delta t, \\ 0, & \text{otherwise,} \end{cases}$$

which was always used to initiate non zero output from the low-pass filter model. The initial values of $x_i(t)$ and $\dot{x}_i(t)$ were set zero. We used the fast Fourier transform algorithm (Ramirez, 1985) to calculate $X_i^{(l)}(\omega) = \mathcal{F}[x_i^{(l)}(t)]$ for time t from t_{st} to t_{nst} . A least-square fit led us to choose:

$$F(S_i^{(l)}(\omega), T_i^{(l)}(\omega)) = \frac{1}{2}(S_i^{(l)}(\omega) - T_i^{(l)}(\omega))^2.$$

A graph of $\log(\omega)$ versus $\log(S^{(0)}(\omega))$ of an experimental EEG power spectrum can be fitted by a ‘‘straight’’ line for ω ($\triangleq f/2\pi$) greater than a minimal value ω_{min} to the maximal frequency ω_{max} (see Fig. 5), with slope $-\beta_i$

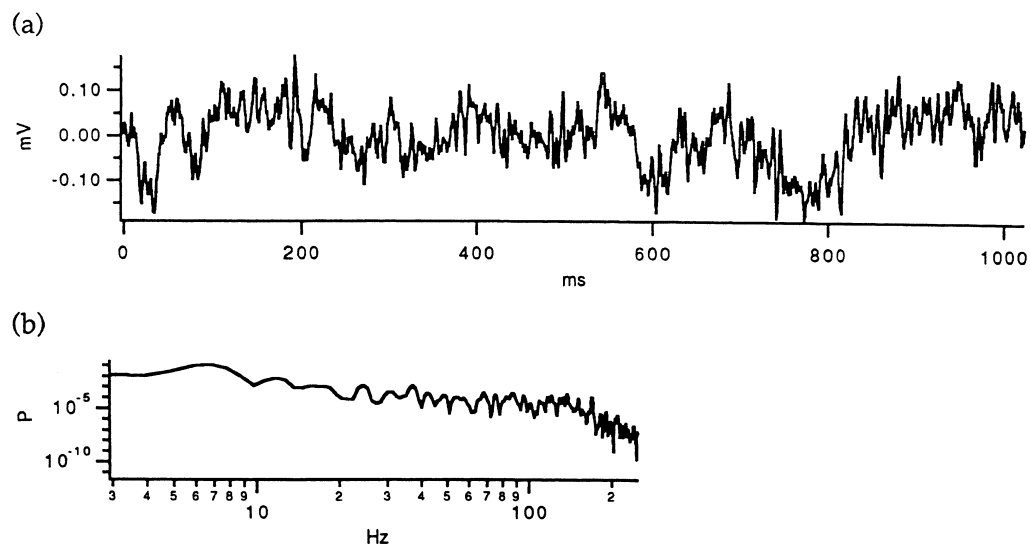


Fig. 5. (A) An EEG from OB of a non-motivated rabbit; (B) the corresponding log–log plot of spectrum of the OB.

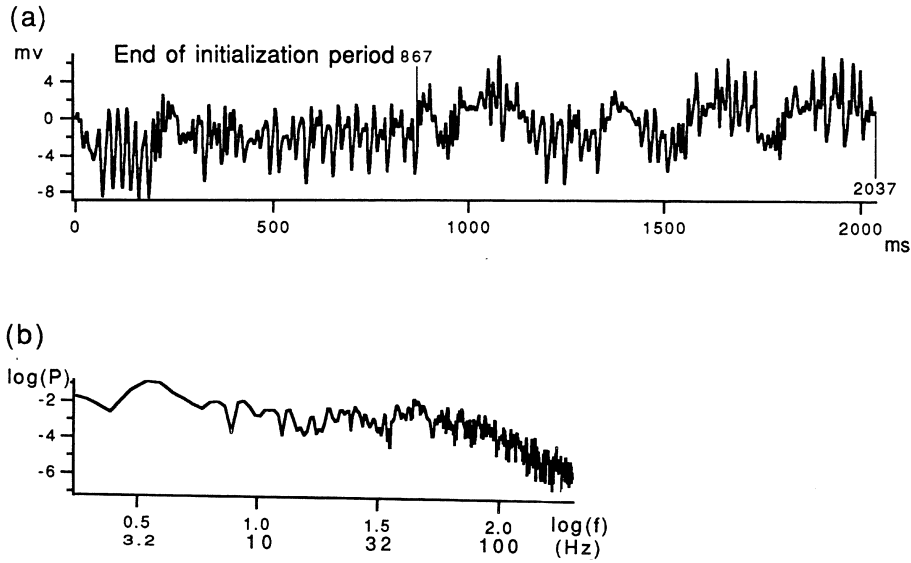


Fig. 6. (A) The example of time series from G_2 of the four-channel low-pass filter model, with $w_{(MP)}=0.779$, $w_{(M_1M_1L)}=2.500$, $w_{(E_1M_1)}=1.300$, $w_{(A_1M_1)}=1.700$, $w_{(G_1G_1L)}=1.000$, $w_{(CB_1)}=-1.300$, $w_{(B_1C)}=1.187$, and $w_{(G_1D_1)}=0.500$, $w_{(PD_2)}=4.000$, $w_{(I_1D_3)}=0.500$, $w_{(G_1D_4)}=4.000$, and $T_1^{(s)}=20.000$, $T_1^{(e)}=10.000$, $T_2^{(s)}=26.000$, $T_2^{(e)}=15.000$, $T_3^{(s)}=25.000$, $T_3^{(e)}=12.000$, $T_4^{(s)}=39.000$, $T_4^{(e)}=24.000$; (B) the corresponding log–log plot of power spectra from 867 to 2037 ms.

and y-intercept at $x = \log(\omega_c)$ as $\log(\alpha_i)$. We had the target pattern as $T_i^{(0)}(\omega) = \alpha_i \omega^{-\beta_i}$. $\eta_i^{(0)}(\omega) = \omega^{2\beta_i}$ was set to normalize across high frequency and $\eta_i^{(1)}(\omega) = 0$ because of no feature to optimize on $x_i^{(1)}(t)$. By the form of Eq. (6), the objective function for minimizing became:

$$E = \int_{\omega_{\min}}^{\omega_{\max}} \frac{1}{2} [\omega^{\beta_i} S_i^{(0)}(\omega) - \alpha_i]^2 d\omega \quad (11)$$

which gave the ordinary partial derivative

$$\frac{\partial E}{\partial X_i(t)} \triangleq p_i^{(0)}(t) = \sqrt{\frac{2}{\pi}} \int_{\Omega} [w^{\beta_i} S_i^{(0)}(\omega) - \alpha] \omega^{\beta} \text{Re}\{X_i^{(0)}(\omega) \times \exp(-i\omega t)\} d\omega$$

For an m th-order differentiable variable $x(t)$ with its Fourier

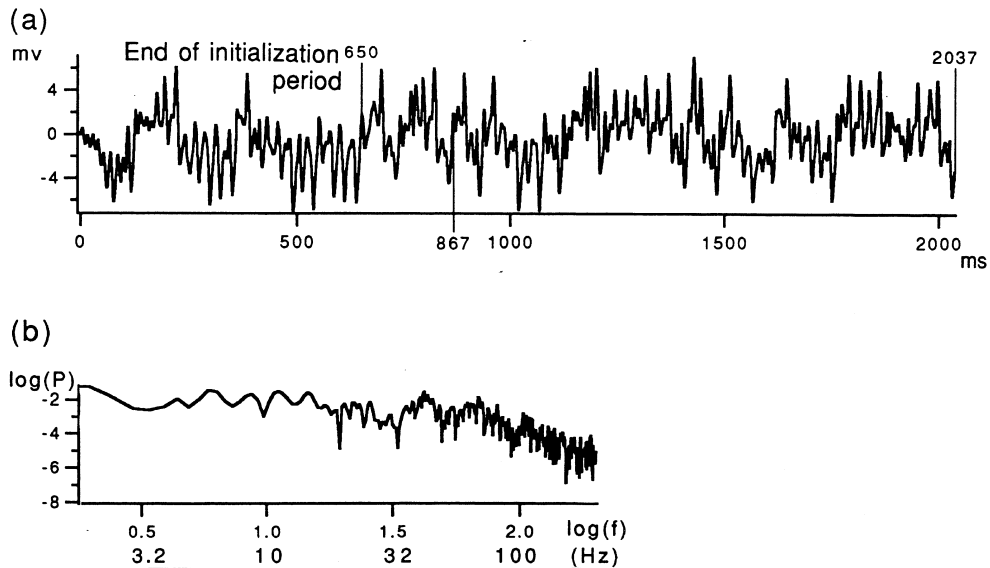


Fig. 7. (A) The example of time series from G_2 of the 16-channel low-pass filter model; (B) The corresponding log–log plot of power spectra from 867 to 2037 ms.

transform existing:

$$\left| \mathcal{F} \left[\frac{d^3 x(t)}{dt^3} \right] \right| = \omega^2 |\mathcal{F}[x(t)]|^2$$

holds for any non-negative integer $J \leq m$. Thus, the above objective function with respect to $S_i^{(0)}(\omega)$ can be transformed to that with respect to $S_i^{(1)}(\omega)$ and this result was more practical especially when $x_i(t)$ was a “pure” $1/f$ (i.e. $\beta_i = 2$, which was often the cases for our simulations). Therefore, the power spectrum of $x_i^{(1)}(t)$ was supposed to be a constant and:

$$E = \int_{\omega_{\min}}^{\omega_{\max}} \frac{1}{2} [S_i^{(1)}(\omega) - \alpha_i]^2 d\omega, \quad (12)$$

with the ordinary partial derivative as

$$\frac{\partial E}{\partial x_i^{(1)}(t)} \triangleq p_i^{(1)}(t) = \sqrt{\frac{2}{\pi}} \int_{\Omega} [S_i^{(1)}(\omega) - \alpha_i] \text{Re}\{X_i^{(1)}(\omega) \times \exp(-i\omega t)\} d\omega$$

Thus, the target $T_i^{(1)}(\omega) = \alpha_i$ and $\eta_i^{(0)}(\omega) = 0$ and $\eta_i^{(1)}(\omega) = 1$. From the viewpoint of the Livermore solver on the olfactory model Eq. (5), both $x_i^{(1)}(t)$ and $x_i^{(0)}(t)$ were independent variables to be solved. No further preparation was needed if using Eq. (12) and the numerical accuracy of $x_i^{(1)}(t)$ was just as good as $x_i^{(0)}(t)$.

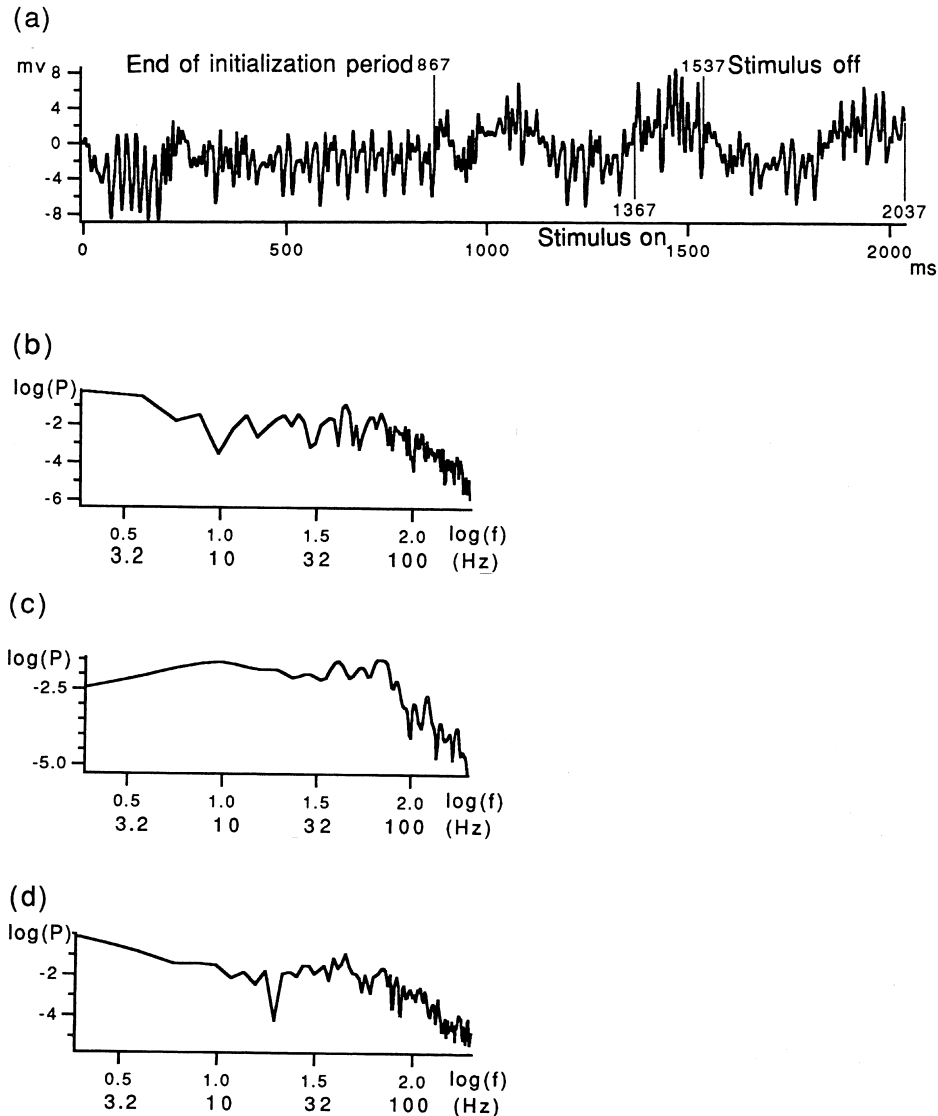


Fig. 8. (A) The example of time series from G_2 of the four-channel low-pass filter model with constant stimulus from 1367 to 1537 ms (simulated burst of input-induced state transition); (B) the corresponding log–log plot of power spectra from 867 to 1367 ms; (C) the corresponding log–log plot of power spectra from 1367 to 1537 ms; (D) The corresponding log–log plot of power spectra from 1537 to 2037 ms.

4.1. Numerical results of parameter optimization

We optimized ω_{CB_1} , ω_{B_1C} , $\omega_{M_1M_1L}$, $\omega_{G_1G_1L}$ and connection strengths ω_{ij} between layers PG, OB, AON, PC and delay feedback nodes D_1 , D_2 , D_3 and D_4 . Appendix C gives the fixed parameters, which had been determined. A step-by-step algorithm is listed in Appendix D.

In Fig. 6, the parameters are optimized for a four-channel low-pass filter model, yielding a $1/f$ -type power spectrum from the output time series of OB node (the granule cells G_2 of the first channel in Fig. 1). To express scaling invariance of the model, we extended the simulation to a 16-channel case and kept the parameter values unchanged. Fig. 7 shows the output of the first G_2 node having a $1/f$ -type power spectrum. (Although the initialization period in the 16-channel model simulation ended

around $t = 650$ ms, we still calculated power spectra from $t = 867$ ms for comparison with the four-channel case.) Both the examples had Eq. (5) numerically integrated from 0 to 2037 ms. Because $1/f$ power spectrum is only a necessary condition to simulate an EEG wave, after finding possible candidates by way of the parameter optimization algorithm, we visually checked if the time series was experimentally realistic.

Next, this four-channel model was given a constant stimulus from 1367 ms to 1537 ms to channel 1 (on node $i = 1$ and $i = n + 1$). Fig. 8 shows that the first 867 ms is a warm-up period and the following 500 ms gives a basal state. The model trajectory then jumps to a near “periodic” state (a burst-like state) in the next couple hundred milliseconds due to presenting a 170 ms constant stimulus input at R with intensity 0.68 and goes back to the same basal

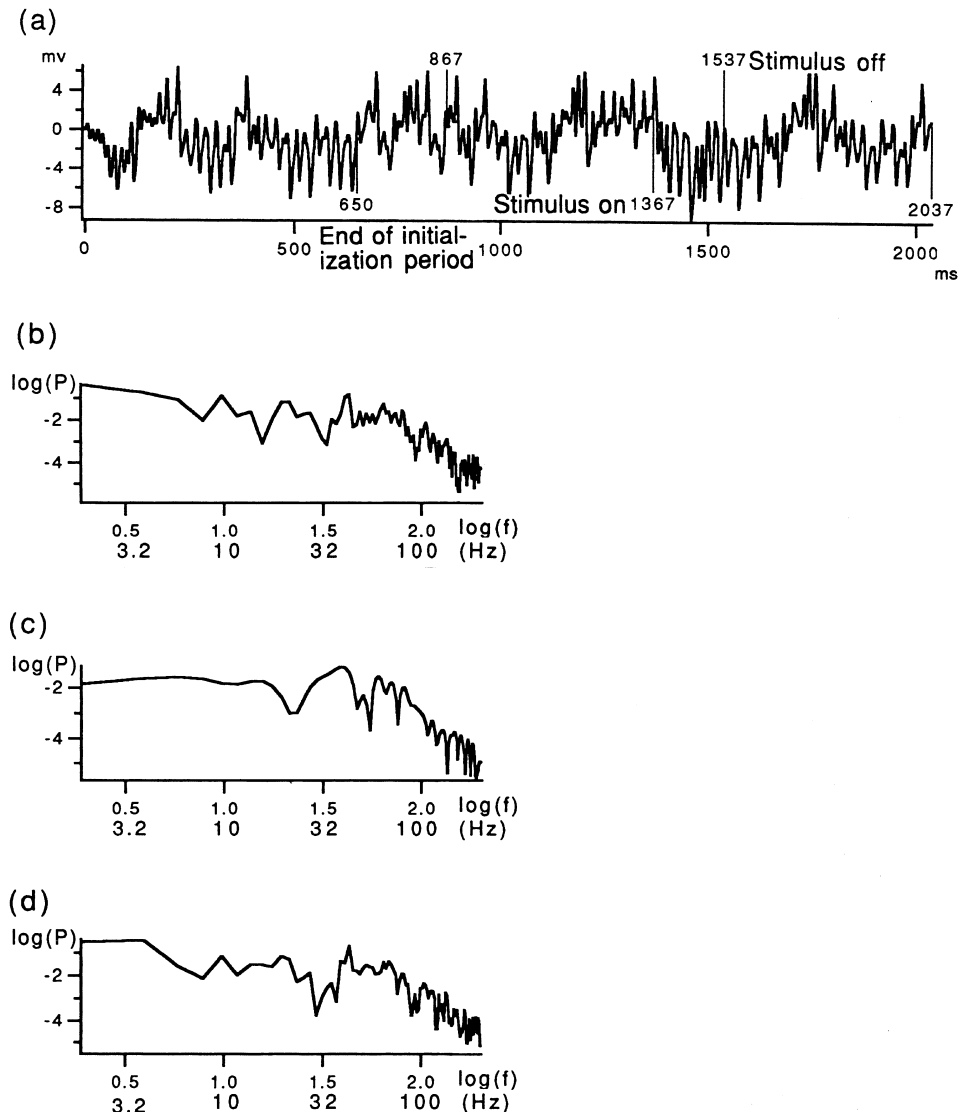


Fig. 9. (A) The example of time series from G_2 of the 16-channel low-pass filter model with constant stimulus from 1367 to 1537 ms (simulated burst of input-induced state transition); (B) the corresponding log–log plot of power spectra from 867 to 1367 ms; (C) the corresponding log–log plot of power spectra from 1367 to 1537 ms; (D) the corresponding log–log plot of power spectra from 1537 to 2037 ms.

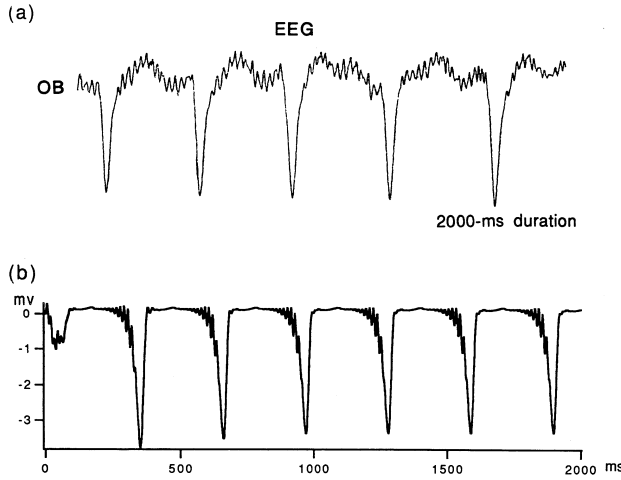


Fig. 10. (A) A 2000 ms segment from a rat's EEG recording during a seizure [Reprinted from Freeman (1987) by permission of Springer, Berlin]. (B) A segment 2000 ms in duration from the output of G_2 node of the one-channel low-pass filter model, with $w_{(M_1, P)} = 0.779$, $w_{(E_1, M_1)} = 0.600$, $w_{(A_1, M_1)} = 3.800$, $w_{(G_1, G_1, L)} = 1.000$, $w_{(CB_1)} = -0.900$, $w_{(B_1, C)} = 1.600$, and $w_{(G_1, D_1)} = 3.240$, $w_{(PD_2)} = 6.020$, $w_{(I_1, D_2)} = 3.7803$, $w_{(G_1, D_1)} = 0.240$, and $T_1^{(s)} = 19.510$, $T_1^{(e)} = 10.470$, $T_2^{(s)} = 25.790$, $T_2^{(e)} = 14.250$, $T_3^{(s)} = 24.870$, $T_3^{(e)} = 12.330$, $T_4^{(s)} = 37.960$, $T_4^{(e)} = 24.160$.

state, shown by the another 500 ms, after the stimulus is ended. Again, the simulation was extended to the 16-channel model (Fig. 9). An important property of the low-pass filter olfactory model is revealed by these examples: the system returns to the previously defined attractor after termination of the stimulus. Moreover, the $1/f$ state is stable in the sense that if the total perturbation amount (external input) to the model is not more than 0.68 (input intensity) \times 170 s (input duration), the $1/f$ state can persist. Because of its high nonlinearity, the low-pass filter model is destabilized when either the stimulus duration or the stimulus intensity becomes too long or too strong. The input stage of the olfactory system has a neural mechanism for input range compression by a logarithm transformation (Freeman, 1975), which, if over-ridden or by-passed, can lead to onset of seizures. Fig. 10 shows the one-channel model with optimized parameters to simulated the olfactory seizure activity. one-channel model has two connected P nodes (not a single P node) to form the layer of periglomerular cells (Freeman, 1987). When we compare the $1/f$ state shown in Fig. 11(A) with a non- $1/f$ state shown in Fig. 12(A), these simulation results also suggest that a $1/f$ state is less influenced by a tiny perturbation (e.g. the numerical truncation/round-off). Thus, we propose that maintenance of the $1/f$ spectrum is a good criterion to “stabilize” a biological neural system and that corresponds to having high correlation of output time series in numerical simulations of the low-pass filter olfactory model [see Fig. 11(A)]. Here, “stabilization” for the model means that sufficiently small perturbations to parameters, variables, or even the model's structure will still permit the trajectory to return to the same attractor.

We cannot expect to reproduce the identical results by

directly using Livermore package with the truncated data given in Figs. 6–12. When the low-pass filter model is operated in a domain of aperiodic oscillations, it resembles a chaotic system, in that tiny perturbations of state variables or parameters may lead to an exponentially diverging output. The exact parameter values used by the machines internally were binary representation with double precision (64 bits) and rounded off to give the decimal values shown. However, in any programming language with any proper ODE integrator and on any computer, given the initial guesses of the parameters and the starting condition, we can always optimize the values of parameters in respect to a specified criterion of output.

4.2. Shadowing of the chaotic solution

A chaotic system has the property that a relatively small numerical noise tends to grow exponentially fast. To integrate ODEs on a current digital computer, a continuous time function must be converted into a discrete time map. Following an iterated process with a proper time step, we get a numerical solution. We cannot accept this is “a true solution” of the original chaotic system. We agree with Dawson et al. (1993) who stated: “We believe that in systems with high-dimensional chaos, trajectories with intrinsic noise, such as computer-generated pseudo-trajectories, can be shadowed only for short times.”

Thus, the extended question we examine is the fuzzy word “short”: Can we control the shadowing time? Because the normalizing-rounding process in the floating point arithmetic is a deterministic process occurring at the least significant bit level, the shadowing lemma to the digital integration for a chaotic solution is not applicable. Fryska and Zohdy, 1992 injected controlled amounts of uniformly distributed random noise to a three-dimensional chaotic system (a piecewise linear system) during digital integration and thereby closely approximated the statistics of the invariant chaotic attractor. Due to simplicity of their system, an exact analytical solution was obtainable, which was used as a reference for comparing with numerical solutions from different precisions.

We have no way to analytically solve the high dimensional neural model. To check the shadowing property and the numerical sensitivity of the model, here, we gave two mathematically equivalent formulas for the low-pass filter model, say Implementation 1 and Implementation 2. Consider the ODE for the M_1 node of OB:

$$\begin{aligned} & \frac{1}{ab} [\ddot{M}_{1\mu}(t) + (a+b)\dot{M}_{1\mu}(t) + abM_{1\mu}(t)] \\ &= \omega_{(MP)} Q(P_\mu(T), q^{(PG)}) + \frac{w_{(M_1 M_1 L)}}{n-1} \sum_{v \neq \mu}^n Q(M_{1v}(t), q^{(OB)}) \\ & \quad + \omega_{(MM)} Q(M_{2\mu}(t), q^{(OB)}) + \omega_{(MG)} [Q(G_{1\mu}(t), q^{(OB)}) \\ & \quad + Q(G_{2\mu}(t), q^{(OB)})] + k_{(MR)} r_\mu(t). \end{aligned}$$

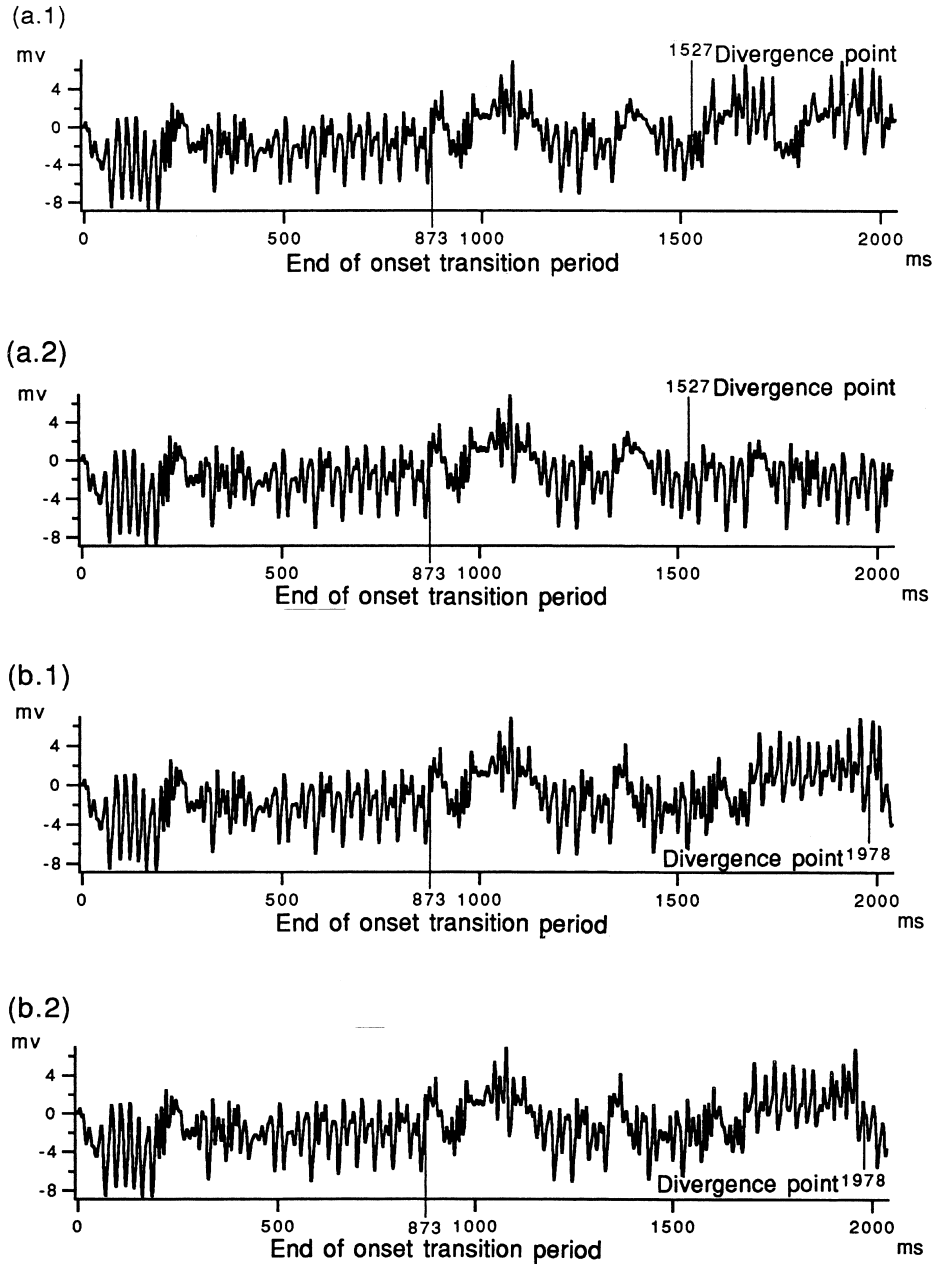


Fig. 11. Output waveforms from the four-channel low-pass filter model of G_2 cell ensemble. (A) Without terminal bit randomization: (a.1) numerical solution by Implementation 1 and (a.2) numerical solution by Implementation 2. (B) With terminal bit randomization: (b.1) numerical solution by Implementation 1, and (b.2) numerical solution by Implementation 2.

On Implementation 1, the second term of the RHS was coded by:

$$\frac{1.0}{n-1.0} \text{ times } \omega_{(M_1 M_1 L)} \sum_{v \neq \mu}^n Q(M_{1v}(t), q^{(OB)}).$$

On Implementation 2, it was coded by:

$$\omega_{(M_1 M_1 L)} \sum_{v \neq \mu}^n Q(M_{1v}(t), q^{(OB)}) \text{ divided by } n-1.0.$$

All the other parts on coding the ODEs were identical. Because of truncation/round-off on terminal bits, the two formulas are not identically represented by machine code. The difference can propagate out spatio-temporally and finally cause two divergent numerical solutions.

By injecting controlled amounts of random noise to the model, we used:

$$\dot{x}(t) = f(x(t), t) + \text{a tiny Gaussian random variable}$$

for each single equation of Eq. (5) to randomize terminal

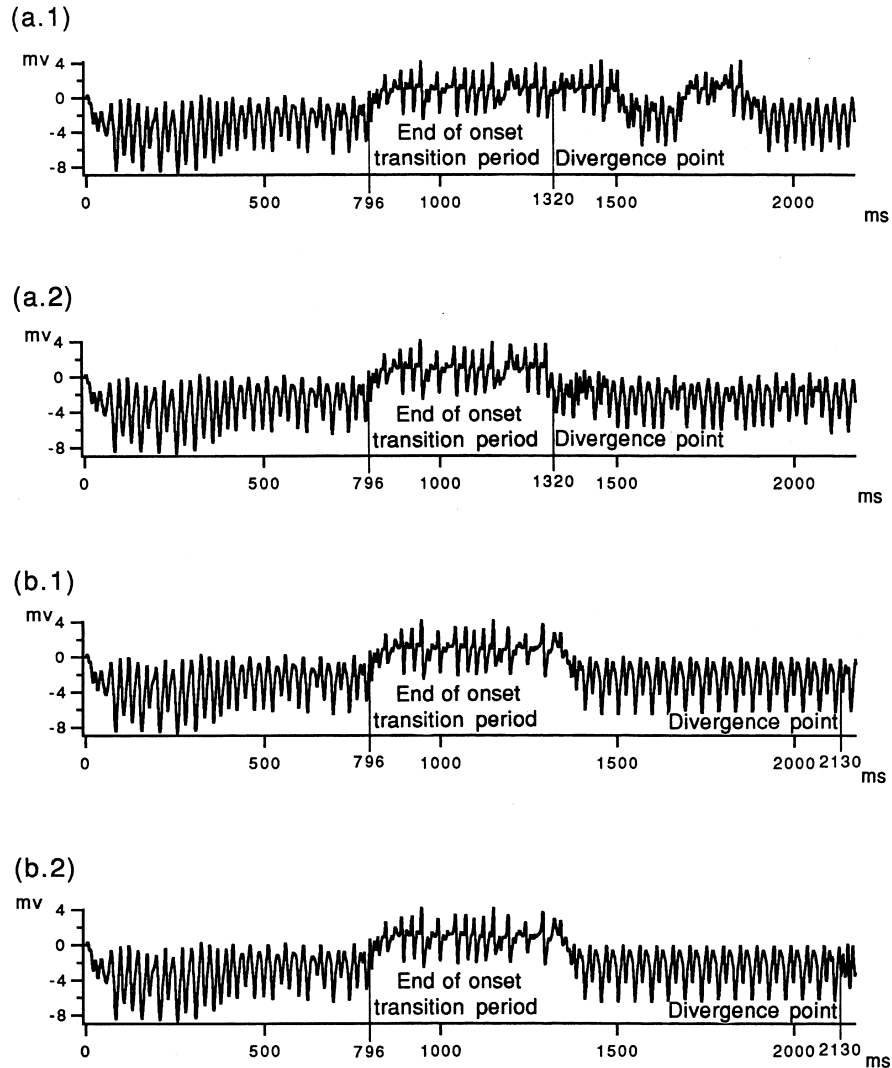


Fig. 12. Output waveforms from the four-channel low-pass filter model of G_2 cell ensemble (with $w_{(E_1, M_1)} = 3.300$ and $w_{(B_1, C)} = 1.700$ instead). (A) Without terminal bit randomization: (a.1) numerical solution by Implementation 1 and (a.2) numerical solution by Implementation 2. (B) With terminal bit randomization: (b.1) numerical solution by Implementation 1 and (b.2) numerical solution by Implementation 2.

bits. The numerical simulations extended the shadowing time. For the four-channel model with the identical parameter values used in Section 4.1, we still tested the G_2 node in the first channel. Fig. 11(A) shows an obvious divergence of both implementations from $t = 1527$ ms; Fig. 11(B) shows both implementations start to diverge obviously around $t = 1978$ ms. If we then set $\omega_{(E_1, M_1)} = 3.300$, $\omega_{(B_1, C)} = 1.700$ instead, the resultant trajectory becomes a “less chaotic” pattern: an obvious divergence from $t = 1320$ ms in Fig. 12(A), but from $t = 2127$ ms in Fig. 12(B). Thus, a random noise injection can “stabilize” the model’s trajectories (in the sense that the two numerical solutions from mathematically equivalent formulas, but not numerically identical implementations stay close longer).

The key question is how to control the random noise (e.g. how to set the values of mean and standard deviation for the Gaussian random variable) to reduce the divergence rate. It

is, of course, parameter dependent because the model trajectory depends on a given set of parameters. When the parameter optimization algorithm is applied the values of parameters are continually updated. To “entirely stabilize” the model with parameter adaptation is still an open problem. If the shadowing time period can be reasonably long due to such a “stabilization” process, we have a relatively high confidence in the numerical results of parameter optimization from a digital computer. The expected maximal duration of a stable state between transitions is around 200 ms in the biological neural system (Barrie et al., 1996). The desired safety factor for stability is 10×200 ms. For competing with experimental EEGs, ~ 2000 ms is a fairly good shadowing length for numerical simulation.

Because of such super sensitivity to numerical truncation/round-off, before the shadowing problem is “solved” we

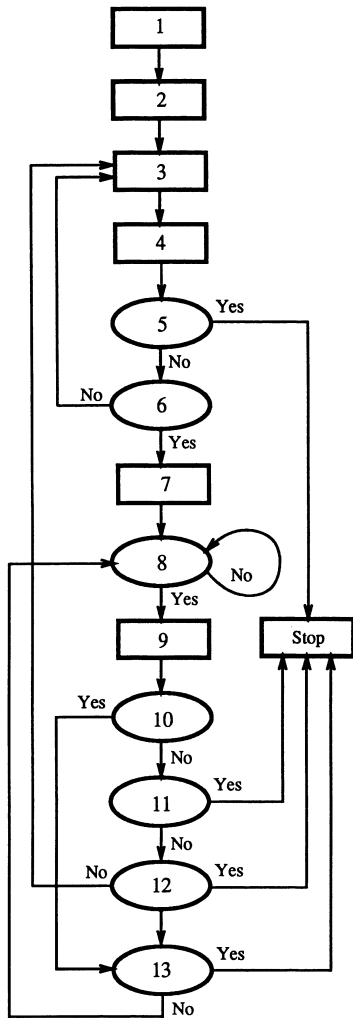


Fig. 13. Flow chart of parameter optimization algorithm for $1/f$ power spectra.

are obliged to regard it as part of the olfactory model. The current technology of numerical methods appears to be the only executable approach, which can give us a closest pseudo-solution to the true solution of such a high dimension, nonlinear ODE set.

5. Conclusion

In Chang and Freeman (1996), we optimized parameters according to an objective function measured in time domain (i.e. the output pattern is examined in time domain). The main advantage of this optimization technique is that it can give us derivatives of an objective function with respect to system variables under constraints, when we have the objective function formulated in terms of these variables. Following this, we can always derive the corresponding parameter adaptation rules. Here, in a frequency domain, the objective function which relates variables to the feature of a $1/f$ power spectrum is formulated. Thus, the

self-adaptation rule for parameter fine tuning is derived. To avoid being absorbed into a local minimum, we apply the random optimization technique for global parameter searching. How to choose random distribution functions to quickly locate a global minimum is heuristic. By using this parameter optimization algorithm, optimized values of parameters for the low-pass filter olfactory model become self-adaptively achievable to generate $1/f$ power spectra and match experimental data.

Earlier models of olfactory dynamics (Freeman, 1987; Yao and Freeman, 1990) used a set of differential-delay equations to model axonal propagation delays between layers PG, OB, AON and PC. ‘‘Differential’’ accounts for a discrete set of distributed delays in an array of stored values at fixed time steps. ‘‘Delay’’ accounts for the time-delay feedback and the dynamical model made mathematical analysis difficult by introducing unavoidable numerical error (inaccuracy) in using existing ODE solvers as described in Section 2.2. This is a significant problem for a chaotic system because we cannot tell whether the output chaotic patterns generated by computer simulations are from the dynamics or from the numerical error. In this paper, we simulate the time-delay feedback by using another second-order ODE. Biologically, this low-pass filter model is plausible because the onset and offset of delayed feedback by axonal tracts is smoothed by temporal dispersion. Mathematically, the entire model becomes a set of ODEs, which eliminates the numerical inaccuracy of differential-delay equations and simplifies the derivation of parameter optimization rule. However, on embodiment of the model in software, the ODEs are transformed into difference equations, in which the size of the digital time step and the word size discretize the model’s trajectories. This makes no difference for point and limit cycle attractors, but it is non-negligible for aperiodic, presumably chaotic, attractor solutions. The duration and stationarity decrease with the size of the model in the OB from four to 16 to 64 oscillators, indicating that the instability is due to attractor crowding (Wiesenfeld, 1989; Tsang and Wiesenfeld, 1990). Without noise, this is unavoidable in a model of realistic size operating with aperiodic attractors. Because a biological system is stable and robust, we would like its model to be robust in qualitative dynamics to perturbations in variables and parameters. Except for the injection of temporal noise as discussed in Section 4.2, the other neurobiologically guided method in the future to stabilize the model may be either to explore the introduction into the model of controlled random variation of gain and delay parameters, which simulates the known variation in the properties of neurons in the olfactory system as multiplicative noise, or to explore the utility of low-level random variation of state variables in the model in the form of additive noise, which simulates the presence of axonal noise in the synaptic inputs to neural ensembles. After adding this spatial noise to the model, we will re-optimize its parameters.

Acknowledgements

This research was supported by a grant from the Office of Naval Research ONR-N63373-Nonlinear Neurodynamics of Biological Pattern Recognition. The authors thank Leslie Kay for preparing Fig. 5.

Appendix A

By the following specification, Eq. (1) implements the olfactory neural system as shown in Fig. 1.

(1) The total number of neural ensembles, $N = 5n + 13$ ($i = 1$ to n for P; $i = n + 1$ to $2n$ for OB M_1 ; $i = 2n + 1$ to $3n$ for OB M_2 ; $i = 3n + 1$ to $4n$ for OB G_1 ; $i = 4n + 1$ to $5n$ for OB G_2 ; $i = 5n + 1$ to $5n + 4$ for AON E_1, E_2, I_1 and I_2 ; $i = 5n + 5$ to $5n + 8$ for PC A_1, A_2, B_1 and B_2 ; $i = 5n + 9$ for C; $i = 5n + 10$ to $5n + 13$ for DL D_1, D_2, D_3 and D_4).

(2) The N -dimensional vector of wave activity variables ($x_1(t), \dots, x_N(t)$) is:

$$\begin{aligned} & (P_1(t), \dots, P_n(t), M_{1_1}(t), \dots, M_{1_n}(t), M_{2_1}(t), \dots, M_{2_n}(t), \\ & G_{1_1}(t), \dots, G_{1_n}(t), G_{2_1}(t), \dots, G_{2_n}(t), E_1(t), E_2(t), \\ & I_1(t), I_2(t), A_1(t), A_2(t), B_1(t), B_2(t), C(t), D_1(t), \\ & D_2(t), D_3(t), D_4(t))^T \end{aligned}$$

where $P_\mu(t)$ denotes the input activity of P node at time t at the μ th channel with $1 \leq \mu \leq n$, etc.

(3) The N -dimensional vector of external stimuli, ($r_1(t) \dots r_N(t)$) is:

$$\begin{aligned} & (R_1(t), \dots, R_n(t), R_1(t), \dots, R_n(t), \mathbf{0}_n, \mathbf{0}_n, \mathbf{0}_n, 0, 0, 0, 0, 0, \\ & 0, 0, 0, 0, \mathbf{0}_4)^T \end{aligned}$$

(4) Let $t - T_l^{(s)}$ and $t - T_l^{(e)}$ be the delay starting time and the delay ending time with respect to current time t , respectively, ($l \in \{1, 2, 3, 4\}$). The six sets of rate constants are:

$$\left\{ \begin{aligned} & \psi_i = a + b \text{ and } \Phi_i = ab, \quad i \in LR \quad (1 \leq i \leq 5n + 9) \\ & \psi_i = \frac{1}{T_{i-(5n+9)}^{(s)}} + \frac{1}{T_{i-(5n+9)}^{(e)}} \text{ and} \\ & \Phi_i = \frac{1}{T_{i-(5n+9)}^{(s)} T_{i-(5n+9)}^{(e)}}, \quad i \in DL \\ & (5n + 10 \leq i \leq 5n + 13, \text{ i.e. when } x_i(t) \text{ is } D_l(t) \\ & \text{for } l \in \{1, 2, 3, 4\}). \end{aligned} \right.$$

(5) The structure of matrix $[w_{ij}]$ defines the connection network and the values of w_{ij} s determine the connection strengths. For the olfactory neural model, we have the $N \times N$ matrix of $[w_{ij}]$:

$$\begin{array}{cccccccccccccccccccc} \frac{w_{(PL)}J}{n-1} & \mathbf{0}_{n \times n} & \mathbf{0}_{n \times n} & \mathbf{0}_{n \times n} & \mathbf{0}_{n \times n} & \mathbf{0}_{n \times 1} & \mathbf{0}_{n \times 1} & \mathbf{0}_{n \times 1} & \mathbf{0}_{n \times 1} & \mathbf{0}_{n \times 1} & \mathbf{0}_{n \times 1} & \mathbf{0}_{n \times 1} & \mathbf{0}_{n \times 1} & \mathbf{0}_{n \times 1} & \mathbf{0}_{n \times 1} & \mathbf{0}_{n \times 1} & w_{(P_2)}M_2 \\ w_{(M_1P)}I & \frac{w_{(M_1M_1)}J}{n-1} & w_{(M_1M)}I & w_{(M_1G)}I & w_{(M_1G)}I & \mathbf{0}_{n \times 1} & \mathbf{0}_{n \times 1} & \mathbf{0}_{n \times 1} & \mathbf{0}_{n \times 1} & \mathbf{0}_{n \times 1} & \mathbf{0}_{n \times 1} & \mathbf{0}_{n \times 1} & \mathbf{0}_{n \times 1} & \mathbf{0}_{n \times 1} & \mathbf{0}_{n \times 1} & \mathbf{0}_{n \times 1} & \mathbf{0}_{n \times 4} \\ \mathbf{0}_{n \times n} & w_{(M_1M)}I & \mathbf{0}_{n \times n} & w_{(M_1G)}I & \mathbf{0}_{n \times n} & \mathbf{0}_{n \times 1} & \mathbf{0}_{n \times 1} & \mathbf{0}_{n \times 1} & \mathbf{0}_{n \times 1} & \mathbf{0}_{n \times 1} & \mathbf{0}_{n \times 1} & \mathbf{0}_{n \times 1} & \mathbf{0}_{n \times 1} & \mathbf{0}_{n \times 1} & \mathbf{0}_{n \times 1} & \mathbf{0}_{n \times 1} & \mathbf{0}_{n \times 4} \\ \mathbf{0}_{n \times n} & w_{(GM)}I & w_{(GM)}I & \frac{w_{(G_1G_1)}J}{n-1} & w_{(G_1G)}I & \mathbf{0}_{n \times 1} & \mathbf{0}_{n \times 1} & \mathbf{0}_{n \times 1} & \mathbf{0}_{n \times 1} & \mathbf{0}_{n \times 1} & \mathbf{0}_{n \times 1} & \mathbf{0}_{n \times 1} & \mathbf{0}_{n \times 1} & \mathbf{0}_{n \times 1} & \mathbf{0}_{n \times 1} & \mathbf{0}_{n \times 1} & w_{(G_1P_1)}M_1 + w_{(G_1P_4)}M_4 \\ \mathbf{0}_{n \times n} & w_{(GM)}I & \mathbf{0}_{n \times n} & w_{(G_1G)}I & \mathbf{0}_{n \times n} & \mathbf{0}_{n \times 1} & \mathbf{0}_{n \times 1} & \mathbf{0}_{n \times 1} & \mathbf{0}_{n \times 1} & \mathbf{0}_{n \times 1} & \mathbf{0}_{n \times 1} & \mathbf{0}_{n \times 1} & \mathbf{0}_{n \times 1} & \mathbf{0}_{n \times 1} & \mathbf{0}_{n \times 1} & \mathbf{0}_{n \times 1} & \mathbf{0}_{n \times 4} \\ \mathbf{0}_{1 \times n} & \frac{w_{(E_1M_1)}I}{n} & \mathbf{0}_{1 \times n} & \mathbf{0}_{1 \times n} & \mathbf{0}_{1 \times n} & 0 & w_{(EE)} & w_{(E)} & w_{(E)} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \mathbf{0}_{1 \times 4} \\ \mathbf{0}_{1 \times n} & \mathbf{0}_{1 \times n} & \mathbf{0}_{1 \times n} & \mathbf{0}_{1 \times n} & \mathbf{0}_{1 \times n} & w_{(EE)} & 0 & w_{(E)} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \mathbf{0}_{1 \times 4} \\ \mathbf{0}_{1 \times n} & \mathbf{0}_{1 \times n} & \mathbf{0}_{1 \times n} & \mathbf{0}_{1 \times n} & \mathbf{0}_{1 \times n} & w_{(IE)} & w_{(IE)} & 0 & w_{(U)} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & w_{(I_1P_3)}M_3 \\ \mathbf{0}_{1 \times n} & \mathbf{0}_{1 \times n} & \mathbf{0}_{1 \times n} & \mathbf{0}_{1 \times n} & \mathbf{0}_{1 \times n} & w_{(IE)} & 0 & w_{(U)} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \mathbf{0}_{1 \times 4} \\ \mathbf{0}_{1 \times n} & \frac{w_{(A_1M_1)}I}{n} & \mathbf{0}_{1 \times n} & \mathbf{0}_{1 \times n} & \mathbf{0}_{1 \times n} & 0 & 0 & 0 & 0 & 0 & w_{(AA)} & w_{(AB)} & w_{(AB)} & 0 & 0 & 0 & \mathbf{0}_{1 \times 4} \\ \mathbf{0}_{1 \times n} & \mathbf{0}_{1 \times n} & \mathbf{0}_{1 \times n} & \mathbf{0}_{1 \times n} & \mathbf{0}_{1 \times n} & 0 & 0 & 0 & 0 & 0 & w_{(AA)} & 0 & w_{(AB)} & 0 & 0 & 0 & \mathbf{0}_{1 \times 4} \\ \mathbf{0}_{1 \times n} & \mathbf{0}_{1 \times n} & \mathbf{0}_{1 \times n} & \mathbf{0}_{1 \times n} & \mathbf{0}_{1 \times n} & 0 & 0 & 0 & 0 & 0 & w_{(BA)} & w_{(BA)} & 0 & w_{(BB)} & w_{(B,C)} & 0 & \mathbf{0}_{1 \times 4} \\ \mathbf{0}_{1 \times n} & \mathbf{0}_{1 \times n} & \mathbf{0}_{1 \times n} & \mathbf{0}_{1 \times n} & \mathbf{0}_{1 \times n} & 0 & 0 & 0 & 0 & 0 & w_{(BA)} & 0 & w_{(BB)} & 0 & 0 & 0 & \mathbf{0}_{1 \times 4} \\ \mathbf{0}_{1 \times n} & \mathbf{0}_{1 \times n} & \mathbf{0}_{1 \times n} & \mathbf{0}_{1 \times n} & \mathbf{0}_{1 \times n} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & w_{(CB)} & 0 & \mathbf{0}_{1 \times 4} \\ \mathbf{0}_{4 \times n} & \mathbf{0}_{4 \times n} & \mathbf{0}_{4 \times n} & \mathbf{0}_{4 \times n} & \mathbf{0}_{4 \times n} & e_1 + e_2 & \mathbf{0}_{4 \times 1} & \mathbf{0}_{4 \times 1} & \mathbf{0}_{4 \times 1} & e_3 & \mathbf{0}_{4 \times 1} & \mathbf{0}_{4 \times 1} & \mathbf{0}_{4 \times 1} & e_4 & \mathbf{0}_{4 \times 1} & \mathbf{0}_{4 \times 1} & \mathbf{0}_{4 \times 4} \end{array}$$

with vectors and matrices as

$$\begin{aligned}
 \mathbf{I} &= \begin{pmatrix} 1 & 0 & \dots & \dots & 0 \\ 0 & 1 & 0 & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & 0 \\ 0 & \dots & \dots & 0 & 1 \end{pmatrix}_{n \times n}, \quad \mathbf{J} = \begin{pmatrix} 0 & 1 & \dots & \dots & 1 \\ 1 & 0 & 1 & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & 1 \\ 1 & \dots & \dots & 1 & 0 \end{pmatrix}_{n \times n}, \quad \mathbf{I} = (1 \ 1 \ \dots \ 1)_{1 \times n}, \\
 \mathbf{M}_1 &= \begin{pmatrix} 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots \\ 1 & 0 & 0 & 0 \end{pmatrix}_{n \times 4}, \quad \mathbf{M}_2 = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots \\ 0 & 1 & 0 & 0 \end{pmatrix}_{n \times 4}, \quad \mathbf{M}_3 = (0 \ 0 \ 1 \ 0)_{1 \times 4}, \quad \mathbf{M}_4 = \begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & 1 \end{pmatrix}_{n \times 4}, \\
 \mathbf{e}_1 &= \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}_{4 \times 1}, \quad \mathbf{e}_2 = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix}_{4 \times 1}, \quad \mathbf{e}_3 = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix}_{4 \times 1}, \quad \text{and} \quad \mathbf{e}_4 = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}_{4 \times 1}.
 \end{aligned}$$

The $[w_{ij}]$ is basically a block diagonal matrix. Each block starts for the layers of the n -channel OB, AON and PC, respectively. The last row, representing the delay feedback nodes, together with other sparse elements, form the connections between layers. When we want to extend the system, all we need to do is to add another block for a new layer and sparse elements for connecting it bidirectionally with the old layers. All results based on Eq. (1) are applicable. Each weight represents the strength (gain) of synaptic action of one group of neurons on another. Experimental testing has shown that the gains are kept within a small signal linear range during normal function (Freeman, 1975). They are time-invariant during recording periods of several seconds and can changes to new values abruptly during learning or pharmacological manipulation. The value can be a positive, negative or zero number to represent an excitatory, inhibitory or no connection from ensemble j to $i \neq j$, respectively. Here, $w_{(P_\mu, P_\nu L)}$ denotes the connection strength linking P_μ , with $P_\nu \neq \mu$ and $w_{(P_\mu, D_2)}$ is the one from D_2 to P_μ , etc. Physically, ‘‘from D_2 to P_μ ’’ means from $E_1(t)$ to P_μ with time delay implemented by passing the output of E_1 through D_2 node. Because our approach is to break the olfactory system into small modules, for simplification it is reasonable to allow:

$$\begin{aligned}
 \text{(a)} \quad w_{(P_\mu, P_\mu L)} &= w_{(PPL)}, \quad w_{(M_{1\mu}, P_\mu)} = w_{(M_1 P)}, \quad w_{(P_\mu, D_2)} = w_{(PD_2)}, \\
 w_{(M_{1\mu}, M_{2\mu})} &= w_{(M_{2\mu}, M_{1\mu})} = w_{(MM)}, \\
 w_{(M_{1\mu}, G_{1\mu})} &= w_{(M_{1\mu}, G_{2\mu})} = w_{(M_{2\mu}, G_{1\mu})} = w_{(MG)},
 \end{aligned}$$

$$\begin{aligned}
 w_{(G_{1\mu}, M_{1\mu})} &= w_{(G_{1\mu}, M_{2\mu})} = w_{(G_{2\mu}, M_{1\mu})} = w_{(GM)}, \\
 w_{(G_{1\mu}, G_{2\mu})} &= w_{(G_{2\mu}, G_{1\mu})} = w_{(GG)}, \quad w_{(M_{1\mu}, M_{1\mu} L)} = w_{(M_1 M_1 L)}, \\
 w_{(G_{1\mu}, G_{1\mu} L)} &= w_{(G_1, G_1 L)}, \quad w_{(G_{1\mu}, D_1)} = w_{(G_1 D_1)} \\
 \text{and } w_{(G_{1\mu}, D_4)} &= w_{(G_1 D_4)}
 \end{aligned}$$

for all $1 \leq \mu, \nu \leq n$ in the PG and OB layers;

$$\begin{aligned}
 \text{(b)} \quad w_{(E_1 E_2)} &= w_{(E_2 E_1)} = w_{(EE)}, \quad w_{(E_1 I_1)} = w_{(E_1 I_2)} = w_{(E_2 I_1)} = w_{(EI)}, \\
 w_{(I_1 E_1)} &= w_{(I_1 E_2)} = w_{(I_2 E_1)} = w_{(IE)} \quad \text{and} \quad w_{(I_1 I_2)} = w_{(I_2 I_1)} = w_{(II)}
 \end{aligned}$$

in the AON layer;

$$\begin{aligned}
 \text{(c)} \quad w_{(A_1 A_2)} &= w_{(A_2 A_1)} = w_{(AA)}, \\
 w_{(A_1 B_1)} &= w_{(A_1 B_2)} = w_{(A_2 B_1)} = w_{(AB)}, \\
 w_{(B_1 A_1)} &= w_{(B_1 A_2)} = w_{(B_2 A_1)} = w_{(BA)} \\
 \text{and } w_{(B_1 B_2)} &= w_{(B_2 B_1)} = w_{(BB)}
 \end{aligned}$$

in the PC layer.

(6) The N -dimensional vector of stimulus intensity weights, $(k_1(t) \dots k_N(t))$ is:

$$\begin{aligned}
 (k_{(P_1 R_1)}, \dots, k_{(P_n R_n)}, k_{(M_{11} R_1)}, \dots, k_{(M_{1n} R_n)}, \mathbf{0}_n, \mathbf{0}_n, \mathbf{0}_n, 0, 0, 0, \\
 \times 0, 0, 0, 0, 0, 0, 0, \mathbf{0}_4)^T
 \end{aligned}$$

with positive elements. Similarly, $k_{(P_\mu R_\mu)}$ and $k_{(M_{1\mu} R_\mu)}$ are the gains from R_μ to P_μ and to $M_{1\mu}$, respectively, and $k_{(P_\mu R_\mu)} = k_{(PR)}$ and $k_{(M_{1\mu} R_\mu)} = k_{(M_1 R)}$ for all $1 \leq \mu \leq n$.

(7) The output is operated on by:

$$y_i(t) = \begin{cases} Q(x_i(t), q^{(LR)}) \triangleq q^{(LR)} \left\{ 1 - \exp\left(-\frac{1}{q^{(LR)}}[\exp(x_i(t)) - 1]\right) \right\}, & i \in LR. \\ x_i(t), & i \in DL. \end{cases} \tag{A.1}$$

The static nonlinearity of the cell ensemble results from the function $Q(x_i, q^{(LR)})$. The $q^{(LR)}$ is a constant for all the x_i s belonging to one layer of cell ensembles. For later use in Section 3.1, we also calculate the derivation of the sigmoid to give the nonlinear gain:

$$\frac{\partial y_i}{\partial x_i} \triangleq y' = \begin{cases} \exp(x_i - \frac{1}{q^{(LR)}}[\exp(x_i) - 1]), & i \in LR. \\ 1, & i \in DL. \end{cases}$$

Appendix B

To gain an insight into this general mathematical model and the corresponding parameter optimization rule, we investigate the relationship to an artificial neural model and the traditional back propagation learning rule.

A frequently used discrete-time version of the standard artificial neural model follows from Eq. (1).

1. Let $y_i(t) = Q(x_i(t))$. (Remove delay feedback nodes and assume all the other nodes have identical I/O transformation.)
2. Let Δt be a small time duration and neglect $O(\Delta t^3)$.
3. Let $a + b = 2/\Delta t$ and $ab = 2/\Delta t^2$. Following this, take the Taylor series expansion of $x_i(t + \Delta t)$ at time t . Eq. (1) reduces to:

$$x_i(t + \Delta t) = \sum_{j \neq i}^N w_{ij} Q(x_j(t)) + k_i r_i(t)$$

This is a discrete-time artificial neural network (Rumelhart et al., 1986).

Next, we may relate the parameter optimization rule of the model of Eq. (1) to the weight learning rule of the artificial neural network. To have a close comparison, consider a multi-layer feed-forward network with error measure according to:

$$E = \frac{1}{2} [Q(x_i(t)) - \Upsilon_i^{(0)}(t)]^2 \quad (\text{B.1})$$

with $\Upsilon_i^{(0)}(t)$ a teaching signal. When the same three conditions listed above are satisfied, we have:

$$\lambda_i^{(1)}(t) = \frac{\Delta t}{2} \lambda_i^{(0)}(t) \quad (\text{B.2})$$

from the second half of Eq. (8). Using Eq. (B.2), the first half of Eq. (8) reduces to:

$$\lambda_i^{(0)}(t) = \begin{cases} \frac{\partial Q(x_i(t))}{\partial x_i(t)} [Q(x_i(t)) - \Upsilon_i^{(0)}(t)], & i \in \{\text{Output Layer}\}, \\ \frac{\partial Q(x_i(t))}{\partial x_i(t)} \sum_{j \neq i}^N w_{ji} \lambda_j^{(0)}(t), & i \in \{\text{Hidden Layers}\}, \end{cases}$$

and Eq. (10) (for $p = w$) combined with Eq. (9) becomes

$$\Delta w_{ij} = -CQ(x_j(t))\lambda_i^{(0)}(t),$$

i.e. the traditional back-propagation learning rule (Rumelhart et al., 1986). The term $p_i^{(l)}(t) dt$ of Eq. (8) is from the conventional partial derivative $\partial E / \partial x_i^{(l)}(t)$ with E being an integration, e.g. Eq. (6). Since E has been taken as Eq. (B.1), $p_i^{(0)}(t) dt$ and $p_i^{(1)}(t) dt$ should be replaced by $\partial Q(x_i(t)) / \partial x_i(t) [Q(x_i(t)) - \Upsilon_i^{(0)}(t)]$ and 0, respectively.

Appendix C

The rate constants $a = 0.220$ and $b = 0.720$. For n -channel PG layer $q^{(P)} = 1.824$, $w_{(PPL)} = 0.900$, $k_{(PR)} = 20.000$. For n -channel OB layer, $q^{(OB)} = 5.000$, $w_{(MM)} = 1.500$, $w_{(MG)} = -2.063$, $w_{(GM)} = 2.323$, $w_{(GG)} = -2.445$, $k_{(M,R)} = 3.000$. For AON layer, $q^{(AON)} = 5.000$, $w_{(EE)} = 1.202$, $w_{(EI)} = -1.426$, $w_{(IE)} = 1.372$ and $w_{(II)} = -1.571$. For PC layer, $q^{(PC)} = 5.000$, $w_{(AA)} = 0.823$, $w_{(AB)} = -1.938$, $w_{(BA)} = 1.947$ and $w_{(BB)} = -2.354$. These parameters were used in the simulations. The determination of these values were presented in reference (Freeman, 1975; Chang and Freeman, 1996).

Appendix D

In this optimization algorithm, one step means one update of a parameter set; ρ_1 and ρ_2 are used for transition between global searching and local tuning; M and ϵ are for stop criteria; Ξ_1 and Ξ_2 store the current and the preceding minimum errors; parameters are those we want to optimize. The corresponding flow chart is presented in Fig. 13.

1. Set the maximum number of steps in a simulation run to M and let the initial step count $c = 0$. Set a sufficiently small positive number, ϵ and two fractions $0 < \rho_1, \rho_2 < 1$. Define a global searching domain, \mathcal{D} and select the initial values of parameters, $p^{(0)}$ to be optimized in \mathcal{D} .
2. Run an integrator with the given $p^{(c)}$ to solve Eq. (1) and calculate Ξ according to Eq. (11) and assign $\Xi_1 = \Xi = p^{(c)}$.
3. Assign $\Xi_2 = \Xi_1$.
4. Run an integrator with the $x_i(t)$ obtained in (2) to solve Eq. (8). By using the $\lambda_i^{(1)}(t)$, $p^{(c)}$ is adapted according to Eqs. (9) and (10). Increase $c = c + 1$. Run an integrator with the given $p^{(c)}$ to solve Eq. (1), calculate Ξ according to Eq. (11) and assign $\Xi_1 = \Xi = p^{(c)}$.
5. Check if $c = M$ or $\Xi_1 < \epsilon$?

- Yes \rightarrow stop;
- No \rightarrow continue.

6. Check if $\left| 1 - \frac{|p^{(c-1)}|}{|p^{(c)}|} \right| < \rho_1$?

or check if

$$\left|1 - \frac{\Xi_1}{\Xi_2}\right| < \rho_2?$$

- Yes \rightarrow continue;
 - No \rightarrow go to (3).
7. Assign $\Xi_2 = \Xi_1$.
8. Generate the random numbers γ and check if $p^{(c)} + \gamma^{(c)} \epsilon D$?
- Yes \rightarrow continue;
 - No \rightarrow go to (8).
9. Update $p^{(c)} = p^{(c)} + \gamma$ and increase $c = c + 1$. Run an integrator with the given $p^{(c)}$ to solve Eq. (1), calculate E according to Eq. (11) and assign $\Xi_1 = \Xi(p^{(c)})$.
10. Check if $\Xi_2 < \Xi_1$?
- Yes \rightarrow go to (13);
 - No \rightarrow continue.
11. Check if $\Xi_1 < \epsilon$?
- Yes \rightarrow stop;
 - No \rightarrow continue.
12. Check if $c = M$?
- Yes \rightarrow stop;
 - No \rightarrow go to (3).
13. Check if $c = M$?
- Yes \rightarrow stop;
 - No \rightarrow go to (8).

There are many variations in random parameters updating for steps (8) and (9) (Solis and Wets, 1981). We have to assure the convergence to the global minimum of the objective function with probability 1 and then compare the convergence rates from modified methods with each other.

References

- Barrie J., Freeman W.J., & Lenhart M. (1996). Spatiotemporal analysis of prepyriform, visual, auditory and somesthetic surface EEGs in trained rabbits. *Journal of Neurophysiology*, 76, 520–539.
- Chang H.J., & Freeman W.J. (1996). Parameter optimization in an olfactory neural systems. *Neural Networks*, 1, 1–14.
- Dawson S., Grebogi C., Sauer T., & Yorke J.A. (1993). Obstructions to shadowing when a Lyapunov exponent fluctuates about zero. *Physical Review Letters*, 73, 1927–1930.
- Eeckman F.H., & Freeman W.J. (1991). Asymmetric sigmoid nonlinearity in the rat olfactory system. *Brain Research*, 557, 13–21.
- Freeman W.J. (1974). A model for mutual excitation in a neuron population in olfactory bulb. *IEEE Transactions on Biomedical Engineering*, 21, 350–358.
- Freeman, W.J. (1975). *Mass action in the nervous system*. New York: Academic Press.
- Freeman W.J. (1979). Nonlinear gain mediating cortical stimulus–response relations. *Biological Cybernetics*, 126, 237–247.
- Freeman W.J. (1987). Simulation of chaotic EEG patterns with a dynamic model of the olfactory system. *Biological Cybernetics*, 56, 139–150.
- Freeman W.J. (1992). Tutorial on neurobiology: from single neurons to brain chaos. *International J. of Bifurcation and Chaos*, 2, 451–482.
- Freeman, W.J. & Shimoide, K. (1994). New approaches to nonlinear concepts in neural information processing: parameter optimization in a large-scale biologically plausible cortical network. In S.F. Zornitser, J.L. Davis & T. Mckenna (Eds.), *An introduction to neural and electronic networks*, 2nd ed. New York: Academic Press.
- Fryska S.T., & Zohdy M.A. (1992). Computer dynamics and shadowing of chaotic orbits. *Physics Letters A*, 166, 340–346.
- Kullback, S. (1978). *Information theory and statistics*. Gloucester: Peter Smith.
- Matyas J. (1965). Random optimization. *Automation and Remote Control*, 26, 246–253.
- Ramirez, R.W. (1985). *The FFT Fundamentals and Concepts*. Englewood Cliffs, NJ: Prentice-Hall.
- Rumelhart, D.E., Hinton, G.E. & Williams, R.J. (1986). Learning internal representations by error propagation. In D.V. Rumelhart, J.L. McClelland & the PDP Research Group (Eds.), *Parallel Distributed Processing*, 1 (pp. 318–362). Cambridge, MA: MIT Press.
- Shimoide K., & Freeman W.J. (1995). Dynamic neural network derived from the olfactory system with examples of applications. *IEICE Transaction Fundamentals*, 78A, 869–884.
- Solis F.J., & Wets J.B. (1981). Minimization by random search techniques. *Mathematics of Operations Research*, 6, 19–30.
- Tsang K.Y., & Wiesenfeld K. (1990). Attractor crowding in Josephson junction arrays. *Applied Physics Letters*, 56, 495–496.
- Werbos, P.J. (1974). *Beyond regression: New tools for prediction and analysis in the behavioral science*. Unpublished doctoral thesis, Harvard University.
- Werbos P.J. (1988). Generalization of back propagation with application to a recurrent gas market model. *Neural Networks*, 1, 339–356.
- Wiesenfeld K. (1989). Attractor crowding in oscillator arrays. *Physical Review Letters*, 62, 1335–1338.
- Yao Y., & Freeman W.J. (1990). Model of biological pattern recognition with spatially chaotic dynamics. *Neural Networks*, 3, 153–170.