Adaptive Sampling by Information Maximization

Christian K. Machens*

Innovationskolleg Theoretische Biologie, Invalidenstrasse 43, Humboldt-University Berlin, 10115 Berlin, Germany

(Received 4 January 2002; published 20 May 2002)

The investigation of input-output systems often requires a sophisticated choice of test inputs to make the best use of limited experimental time. Here we present an iterative algorithm that continuously adjusts an ensemble of test inputs on-line, subject to the data already acquired about the system under study. The algorithm focuses the input ensemble by maximizing the mutual information between input and output. We apply the algorithm to simulated neurophysiological experiments and show that it serves to extract the ensemble of stimuli that a given neural system "expects" as a result of its natural history.

DOI: 10.1103/PhysRevLett.88.228104 PACS numbers: 87.19.La, 07.05.Fb, 84.35.+i, 89.70.+c

Biophysical systems often have many degrees of freedom, and thus one needs large numbers of variables and parameters to describe them. Without strong prior knowledge about the intrinsic dynamics of such a system, one is left with inferring its function from data obtained by experiments or observations. Given a system where we control a set of "input" variables $x = (x^{(1)}, x^{(2)}, \ldots, x^{(n)})$ and measure another set of "output" variables $y = (y^{(1)}, y^{(2)}, \ldots, y^{(m)})$, we can actively manipulate the data acquisition by selecting the most informative test inputs. Yet how should one choose the test inputs to learn the most about the input-output relation?

Within the classical Volterra-Wiener system identification methods [1], the input space is sampled by drawing inputs from a probability distribution $p(x)$; a common choice is Gaussian "white noise." However, not all aspects of the system's input-output relation may be equally important. In neurobiology, for instance, one is especially interested in inputs x about which a given sensory system conveys the most information. In the spirit of importance sampling [2], one might therefore focus the data acquisition on those x that contribute most to the information transfer. For a given input distribution, the information provided by a single input can be quantified as $I(x) = H_y - H_y(x)$ where H_y is the entropy of the output distribution $p(y)$ and $H_y(x)$ is the entropy of the conditional probability distribution $p(y \mid x)$ which characterizes the input-output relation [3,4]. Hence, the appropriate focusing is achieved by an input distribution $p_{\mathrm{opt}}(x)$ that maximizes the mutual information $I = \langle I(x) \rangle$ where the angular brackets denote averaging over $p_{\mathrm{opt}}(x)$.

Without any information about the system and its input-output relation, the optimal input distribution $p_{\mathrm{opt}}(x)$ is unknown. Any experimental test of the system must therefore start with drawing the test inputs from some predefined model distribution $p_\phi(x)$ that depends on a set of parameters $\phi = (\phi^{(1)}, \ldots, \phi^{(L)})$. Once data about the system has been acquired, however, one need not adhere to this initial choice of an input distribution. Instead, one should adapt the parameters or even the structure of $p_\phi(x)$ to better focus on the important inputs. In this Letter, we show how to systematically perform this adaptation. By iterating the adaptation procedure, the acquired data become ever more useful and the input distribution approaches an optimum.

Adapting the input distribution.—For mathematical simplicity, we assume that both input and output take discrete values. Say that we have already tested the system with N different inputs x_i each of which was presented M_i times while measuring the outputs y_{ij} with $i = 1, \ldots, N$ and $j = 1, \ldots, M_i$. We define the set of all different output values measured so far by $\{y_k : k = 1, \ldots, K\}$. Our present knowledge about the system is summarized by the conditional probability that an output y_k was obtained from the input x_i,

$$q(y_k \mid x_i) = \frac{1}{M_i} \sum_{j=1}^{M_i} \delta_{y_{ij}, y_k} . \tag{1}$$

The estimated probabilities $q(y_k \mid x_i)$ allow us to reevaluate the relative importance of the inputs x_i in terms of their potential contribution to the mutual information. To measure this contribution, we assign a probability or "weight" $q(x_i)$ to every input. Initially we assume that all inputs x_i contribute equally and set $q_1(x_i) = 1/N$. To find a combination of weights that maximizes the information transfer, we use the Blahut-Arimoto algorithm [5] and readjust the weights,

$$q_{n+1}(x_i) = \frac{1}{Z} q_n(x_i) \exp\left(\sum_{k=1}^{K} q(y_k \mid x_i) \log \frac{q(y_k \mid x_i)}{q_n(y_k)} \right). \tag{2}$$

Here $q_n(y_k) = \sum_{i=1}^{N} q(y_k \mid x_i) q_n(x_i)$ and Z is a normalization constant so that $\sum_{i=1}^{N} q_{n+1}(x_i) = 1$. According to Eq. (2), the weight of an input x_i is decreased if its conditional output distribution $q(y_k \mid x_i)$ is similar to the total output distribution $q_n(y_k)$. In contrast, the weight of an input x_i is increased if the respective distributions differ. When Eq. (2) is iterated, the weights converge and reach a global maximum of the mutual information [5]. In practice, we terminate the process once $|1 - q_{n+1}(x_i)/q_n(x_i)| < \epsilon$ for all i and some chosen precision ϵ and set $q_{\mathrm{opt}}(x_i) = q_{n+1}(x_i)$.

The weights or probabilities $q_{\mathrm{opt}}(x_i)$ describe the relative frequencies with which the respective inputs x_i should be drawn. Consequently, we need to adapt the parameters ϕ so as to find a matching distribution $p_\phi(x)$. Here we determine the new parameters ϕ by maximizing the log-likelihood function [6],

$$\log L(x_1, \ldots, x_N \mid \phi) = \sum_{i=1}^{N} q_{\mathrm{opt}}(x_i) \log p_\phi(x_i), \quad (3)$$

where the contribution of the inputs x_i is weighted according to $q_{\mathrm{opt}}(x_i)$. For some models, e.g., Gaussians, the maximum can be found analytically. In general, however, one has to evaluate the maximum numerically.

The input distribution given by the new parameter values can be used to draw new test inputs, present them to the system, and measure the respective outputs. After a certain amount of data has been acquired, the parameters ϕ can be adapted again. The resulting iterative algorithm moves the input distribution towards an optimal ensemble.

Model quality and convergence.—Every maximum of the mutual information with respect to $p(x)$ is a global maximum [4]. Hence, if the input distribution does not rule out any inputs, i.e., $p_\phi(x) > 0$ for all x and ϕ, the estimates of the input-output relation, Eq. (1), converge, and therefore $q_{\mathrm{opt}}(x_i) \rightarrow p_{\mathrm{opt}}(x_i)$. Accordingly, the mutual information $I_{\mathrm{D}} = \langle H_y^q - H_y(x) \rangle_q$ achieves the information capacity of the system; here the index q denotes that the respective quantities and averages are calculated with respect to $q_{\mathrm{opt}}(x_i)$.

The model distribution $p_\phi(x)$ converges towards an optimal fit of $p_{\mathrm{opt}}(x)$. To control how well $p_\phi(x)$ captures the structure of $p_{\mathrm{opt}}(x)$, one can check the mutual information achieved by the model, $I_{\mathrm{M}} = \langle H_y^\phi - H_y(x) \rangle_\phi$, which is calculated with respect to $r_\phi(x_i) = p_\phi(x_i) / [\sum_{j=1}^{N} p_\phi(x_j)]$. The fraction γ of the mutual information reached by the model is then defined as

$$\gamma = \frac{I_{\mathrm{M}}}{I_{\mathrm{D}}} \quad (4)$$

and provides a measure for the quality of the model. Hence, if γ falls significantly below one, the distribution $p_\phi(x)$ no longer captures the structure of the optimal ensemble $p_{\mathrm{opt}}(x)$; in such a case, one might increase the complexity of the model.

In general, the algorithm will not be able to adapt the input ensemble if the presented inputs always result in the same output value. Similarly, there is no possibility to weight the inputs x_i differently if every input elicits a new, different output. However, the latter problem can be solved by discretizing the output side into a smaller number of possible outputs. The input space, on the other hand, can be discretized as fine as needed without impeding the convergence of $p_\phi(x)$.

Example.—To illustrate the method, we study a numerical simulation of a Hodgkin-Huxley–type model neuron [7]. The model neuron transforms an input current I into a voltage output V. For constant current values $I < 0\ \mu\mathrm{A/cm^2}$, the voltage approaches a stable equilibrium. For current values $I > 0\ \mu\mathrm{A/cm^2}$, the model undergoes a saddle-node bifurcation and generates periodically occurring action potentials, also called spikes [8]. Stochastic aspects of neural activity are incorporated by adding Gaussian "white" noise with a fixed standard deviation σ_η and a cutoff frequency f_η to the input.

We start with a simple one-dimensional parametrization of input and output. The inputs are 100-ms-long, discretized current steps ($\Delta I = 1\ \mu\mathrm{A/cm^2}$), restricted to a physiologically realistic range of $I \in [-12, 28]\ \mu\mathrm{A/cm^2}$. The outputs are given by the number of spikes, C, during the corresponding time window. The resulting probabilistic relation of spike count versus current is displayed in Fig. 1(a).

For this one-dimensional input-output system, we can compute an exact solution of the information maximization problem. The optimal input distribution $p_{\mathrm{opt}}(I)$ is depicted by the vertical bars in Fig. 1(b); the shape of $p_{\mathrm{opt}}(I)$ corresponds to the slope of the input-output relation [9]. Inputs far below the spiking threshold result almost certainly in no spikes. As reliable inputs allow one to convey more information than unreliable inputs, the optimal input distribution prefers inputs far below threshold to inputs closer to threshold.

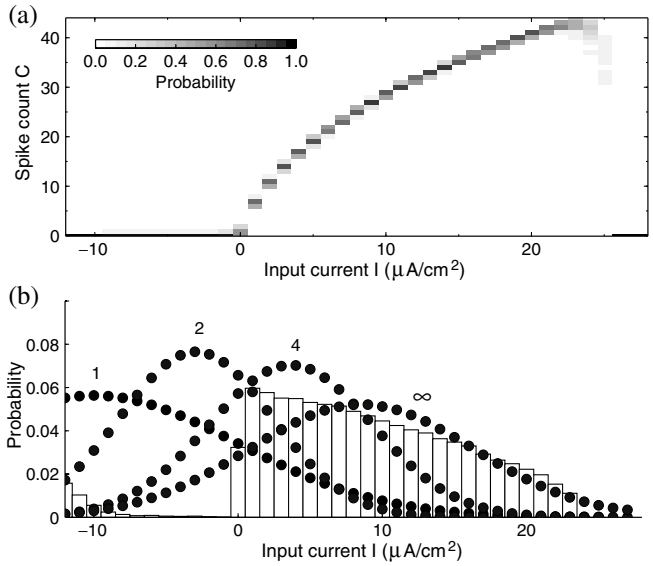

FIG. 1. Approaching the optimal input ensemble of a neuron with static, one-dimensional input and output. (a) Plot of the conditional probability distribution $p(C \mid I)$ with spike count C and input current I. The uncertainties at $I \approx 22\ \mu\mathrm{A/cm^2}$ are due to a decline in spike size that makes it impossible to detect the spikes in the noisy voltage output. For $I \approx 28\ \mu\mathrm{A/cm^2}$, the model neuron ceases to generate spikes. (b) Approaching the optimal input distribution (bars). Shown are the initial distribution (1), the distributions of the iterations (2) and (4), as well as the final distribution (∞). (Simulation parameters: $n = 1$, $m = 1$, $L = 2$, $A = 10$, $B = 5$, $\epsilon = 0.1$, $\sigma_\eta = 4\ \mu\mathrm{A/cm^2}$, $f_\eta = 1000$ Hz.)

To study the performance of the iterative algorithm, we model the optimal input distribution by a truncated Gaussian. As initial parameter values, we choose a mean $\phi^{(1)} = -10\ \mu\mathrm{A/cm^2}$ and a standard deviation $\phi^{(2)} = 10\ \mu\mathrm{A/cm^2}$. In each iteration, we draw A current values from the Gaussian, test them B times on the system, and adapt the parameters. For our Gaussian model, the maximum likelihood estimate of the new parameters is given by $\phi^{(1)} = \sum_{i=1}^{N} I_i q_{\mathrm{opt}}(I_i)$ and $\phi^{(2)} = [\sum_{i=1}^{N}(I_i - \phi^{(1)})^2 q_{\mathrm{opt}}(I_i)]^{1/2}$.

The Gaussian model distributions are displayed in Fig. 1(b) for the first few iterations. Most of the current values drawn from the initial distribution fall below the spiking threshold of the neuron. Consequently, the algorithm shifts the Gaussian distribution into the spiking regime of the neuron. After about 10 iterations, the mutual information rate saturates at ≈40 bits/sec and the final Gaussian model approximately covers the range of inputs relevant to information transmission. Note that due to the maximum-likelihood estimation, Eq. (3), the final Gaussian distribution has the same mean and variance as the optimal distribution.

Multidimensional example.—The computational power of the algorithm becomes clearly visible for high-dimensional input spaces. As an example, consider the above model neuron when the input consists of time-varying, statistically stationary currents, discretized in time steps of Δt_1. Following [10], we slide overlapping windows of length $T = n\Delta t_1$ across the input current trace and use the values within each window as input vector $I_i = (I_i^{(1)}, \ldots, I_i^{(n)})$. For each of these inputs I_i, the output C_{ij} is given by the spike times, discretized in time steps of $\Delta t_2 = T/m$, during the corresponding window. Hence, each input consists of n real-valued numbers bounded within the interval $I \in [-12, 28]\ \mu\mathrm{A/cm^2}$, and each output consists of m binary values that are either zero (no spike) or one (spike). Note that we do not explicitly discretize the current values; we instead assume that every input I_i is unique. For simplicity, we use a Gaussian input distribution. As the input is real and stationary, it suffices to parametrize the Gaussian with average and power spectrum.

To test the system, we choose an initial distribution with an average $\mu = \phi^{(1)} = 0\ \mu\mathrm{A/cm^2}$ and a flat power spectrum with standard deviation $\sigma = [\sum_{i=2}^{L} \phi^{(i)}]^{1/2} = 10\ \mu\mathrm{A/cm^2}$. For this prior, only 50% of the input values lie above threshold and the inputs will rarely lead to high firing rates; cf. Fig. 1. Consequently, we do not properly explore the full range of the input-output relation; if, for example, we test the system for 30 min with input currents drawn from this initial distribution, the information rate I_{D} does not exceed ≈300 bits/sec.

When using the iterative algorithm to adapt the parameters of the input ensemble, on the other hand, the information rate I_{D} saturates around ≈670 bits/sec after about 20 min. Figure 2(a) shows how the power spectrum is shaped during the iterations. Only input frequencies below 500 Hz are well suited for the information transfer, the cutoff is roughly determined by the maximum firing rate of the model neuron. The overall increase in power leads to input currents that override the additive noise η of the model neuron.

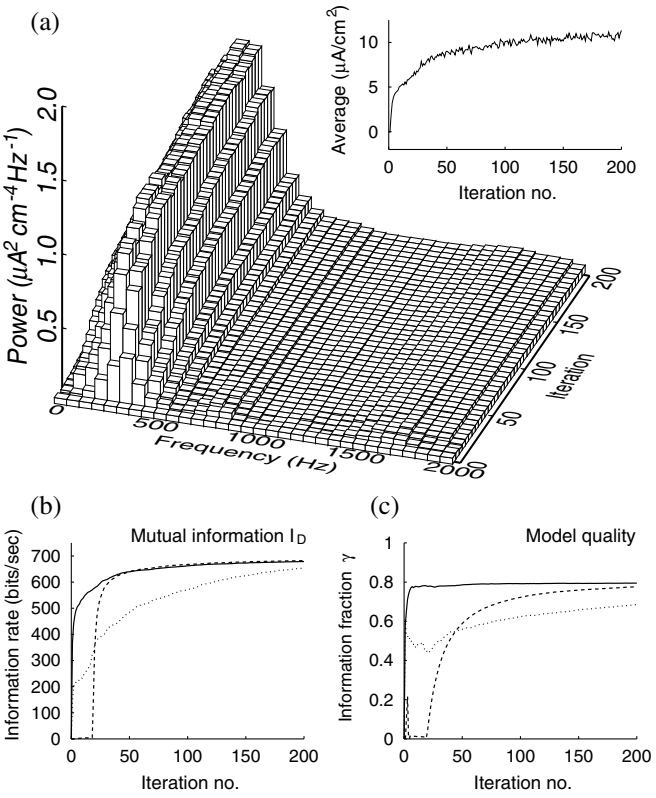

FIG. 2. Approaching the optimal input ensemble of a neuron with time-varying input and output. (a) Evolution of average and power spectrum. (b) Evolution of information rate and (c) model quality for three different initial conditions: $\mu = 0\ \mu\mathrm{A/cm^2}$, $\sigma = 10\ \mu\mathrm{A/cm^2}$ (solid lines); $\mu = -6\ \mu\mathrm{A/cm^2}$, $\sigma = 10\ \mu\mathrm{A/cm^2}$ (dashed lines); $\mu = 20\ \mu\mathrm{A/cm^2}$, $\sigma = 1\ \mu\mathrm{A/cm^2}$ (dotted lines). (Simulation parameters: $n = 64$, $m = 16$, $L = 33$, $A = 1000$, $B = 20$, $\epsilon = 0.1$, $\sigma_\eta = 4\ \mu\mathrm{A/cm^2}$, $f_\eta = 2000$ Hz, $\Delta t_1 = 0.25$ ms, $\Delta t_2 = 1$ ms, $T = 16$ ms; windows slided by Δt_2; accordingly, $A\Delta t_2 B \times 100$ iterations ≈ 34 min.)

Initial conditions, convergence, and degeneracies.—When the initial distribution is very narrow [flat power spectrum up to $f_c = 1000$ Hz, with $\sigma = 1\ \mu\mathrm{A/cm^2}$, $\phi^{(1)} = 20\ \mu\mathrm{A/cm^2}$, Fig. 2(b), dotted line], most of the input currents drive the neuron maximally and thereby very reliably. The strong initial bias leaves the algorithm with little maneuvering space for the parameter reestimation so that it takes longer to approximate an optimal input distribution. In the worst case, every input leads to the same output value. With the initial choice $\phi^{(1)} = -6\ \mu\mathrm{A/cm^2}$ and $\sigma = 10\ \mu\mathrm{A/cm^2}$, the input does not elicit any spikes during the first iterations; cf. Fig. 2(b), dashed line. However, once a spike has appeared, the statistics of the model distribution immediately

moves into the direction of the statistics of the inputs I_i that caused a spike. When the algorithm has tracked the relevant input range, a rapid increase of the information rate follows.

In the examples studied, the mutual information reaches approximately the same value independent of the initial conditions; cf. Fig. 2(b). Although there is always a preference for frequencies below 500 Hz, however, the parameters of the optimal input ensemble do not converge to the same set of values. Consequently, there is no unique combination of parameters that maximizes the mutual information, an observation that generalizes beyond the specific examples chosen. This indeterminacy is caused by "degenerate" subsets in input space, i.e., sets of inputs that lead to the same output value. The total probability assigned to such a subset can be distributed in an arbitrary way on the subset, and any statistical parameters ϕ that depend on these subsets can assume different values without significant consequences for the information transfer. Accordingly, all final input distributions capture about the same amount (80%) of the mutual information I_D; cf. Fig. 2(c).

Neurophysiological interpretation.—Recent studies indicate that sensory neurons convey large amounts of information if the properties of the stimulus ensembles used match those of natural stimuli [11]. Here we have shown how to extract a stimulus ensemble that conveys the maximum possible information without any prior knowledge. The proposed method could therefore serve to find the ensemble of stimuli that a given neuron naturally "expects." Note that in contrast to previous on-line algorithms such as Alopex or Simplex [12], we are not looking for a single optimal stimulus but rather for a complete ensemble of stimuli. The examples demonstrate also that such an optimal stimulus ensemble depends on the choice of a particular neural code.

Conclusion.—Conventionally, input-output systems are investigated by using either a predefined set of inputs or inputs drawn at random from a predefined probability distribution. However, both approaches risk missing important regions in input space. If interest concerns the system's function in terms of information transmission, then the data acquisition can be significantly improved by using the iterative algorithm proposed in this Letter. The optimal input ensemble itself might be interpreted as representing that region in input space that a particular system seeks to encode.

I thank A. V. M. Herz and M. B. Stemmler for stimulating discussions and H. Herzel for helpful comments on the manuscript. This work was supported by the DFG through the Innovationskolleg Theoretische Biologie and the Graduiertenkolleg 120.

*Present address: Cold Spring Harbor Laboratory, 1 Bungtown Road, Cold Spring Harbor, NY 11724.
Electronic address: c.machens@biologie.hu-berlin.de

[1] N. Wiener *Nonlinear Problems in Random Theory* (MIT Press, Cambridge, MA, 1958); Y. L. Lee and M. Schetzen, Int. J. Control **2**, 237 (1965); G. Palm and T. Poggio, SIAM J. Appl. Math. **34**, 524 (1978).

[2] C. Itzykson and J. M. Drouffe *Statistical Field Theory, Vol. 2* (Cambridge University Press, Cambridge, MA, 1989).

[3] C. E. Shannon and W. Weaver, *The Mathematical Theory of Communication* (University of Illinois Press, Urbana, IL, 1949); M. R. DeWeese and M. Meister, Netw., Comput. Neural Syst. **10**, 325 (1999).

[4] T. M. Cover and J. A. Thomas, *Elements of Information Theory* (Wiley, New York, 1991).

[5] S. Arimoto, IEEE Trans. Inf. Theory **IT-18**, 14 (1972); R. E. Blahut, IEEE Trans. Inf. Theory **IT-18**, 460 (1972).

[6] R. Barlow *Statistics* (Wiley, New York, 1989).

[7] X.-J. Wang and G. Buzsáki, J. Neurosci. **16**, 6402 (1996).

[8] E. M. Izhikevich, Int. J. Bifurcation Chaos Appl. Sci. Eng. **10**, 1171 (2000).

[9] For a deterministic input-output system, the mutual information is given by $I = H_y$ since $H_y(x) = 0$ for all x. Maximizing the mutual information results in a uniform distribution of the outputs y. For a one-dimensional system with a monotonic relation $y = f(x)$, we have $p(x)dx = p(y)dy$ and the optimal input distribution is simply given by $p(x) \propto dy/dx$. This relation is approximately preserved in the stochastic case.

[10] S. P. Strong, R. Koberle, R. R. de Ruyter van Steveninck, and W. Bialek, Phys. Rev. Lett. **80**, 197 (1998).

[11] F. Rieke, D. A. Bodnar, and W. Bialek, Proc. R. Soc. London Sect. B **262**, 259 (1995); H. Attias and C. E. Schreiner, in *Advances in Neural Information Processing Systems 10,* edited by M. I. Jordan *et al.* (MIT Press, Cambridge, MA, 1998), pp. 103–109; C. K. Machens, M. B. Stemmler, P. Prinz, R. Krahe, B. Ronacher, and A. V. M. Herz, J. Neurosci. **21**, 3215 (2001).

[12] E. Harth and E. Tzanakou, Vis. Res. **14**, 1475 (1974); I. Nelken, Y. Prut, E. Vaadia, and M. Abeles, Hear. Res. **72**, 237 (1994).