

Mapping Function Onto Neuronal Morphology

Klaus M. Stiefel¹ and Terrence J. Sejnowski^{1,2}

¹Howard Hughes Medical Institute, Computational Neurobiology Lab, The Salk Institute for Biological Studies; and ²Department of Biology, University of California, San Diego, La Jolla, California

Submitted 15 August 2006; accepted in final form 10 April 2007

Stiefel KM, Sejnowski TJ. Mapping functions onto neuronal morphology. *J Neurophysiol* 98: 513–526, 2007; doi:10.1152/jn.00865.2007. Neurons have a wide range of dendritic morphologies the functions of which are largely unknown. We used an optimization procedure to find neuronal morphological structures for two computational tasks: first, neuronal morphologies were selected for linearly summing excitatory synaptic potentials (EPSPs); second, structures were selected that distinguished the temporal order of EPSPs. The solutions resembled the morphology of real neurons. In particular the neurons optimized for linear summation electrotonically separated their synapses, as found in avian *nucleus laminaris* neurons, and neurons optimized for spike-order detection had primary dendrites of significantly different diameter, as found in the basal and apical dendrites of cortical pyramidal neurons. This similarity makes an experimentally testable prediction of our theoretical approach, which is that pyramidal neurons can act as spike-order detectors for basal and apical inputs. The automated mapping between neuronal function and structure introduced here could allow a large catalog of computational functions to be built indexed by morphological structure.

INTRODUCTION

Neurons receive incoming synapses on dendritic trees that are often intricate branching patterns encompassing distinct subdomains (such as basal and apical dendrites in cortical pyramidal neurons). The dendritic morphologies can vary greatly between different classes of neurons, in length, arborization patterns, and presence of dendritic spines. Differences in the morphologies are believed to be related to functional differences (Krichmar et al. 2002; Mainen and Sejnowski 1996; Rall 1995; Vetter et al. 2001). There is, however, no systematic way to determine what the functions might be. In this study, we introduce a general method to map¹ neuronal function onto neuronal structure.

We used genetic algorithms (GAs) to optimize the structure of neuronal dendrites for a number of computational tasks (Holland 1975; Mitchell 2001). A population of possible solutions was tested against the problem and the individuals with the best performance (the most fit) were mutated, recombined, and used to make up the next generation. In this way, the

fitness of the population was increased in every generation, and a biased random walk toward a local or global optimal solution is performed. GAs find good solutions that are made up of good subsolutions (schemes or building blocks) (Mitchell 2001).

Two computational functions were optimized, linear summation and spike-time order detection, and the resulting structures were compared with dendritic morphologies. In both cases, real neuronal morphologies were found to match the optimized structures, suggesting possible functions that they might perform.

METHODS

We used a Lindenmayer-system (L-system) (Lindenmayer 1968) for the algorithmic construction of model neuron morphologies. Simulations of electrophysiological dynamics were carried out in the neuronal simulation language NEURON (version 5.7) (Hines and Carnevale 1997, 2000). The NEURON code used in this study is available as supplementary material by request from K. M. Stiefel and will be submitted to the Yale Senselab neuronal model-database (<http://senselab.med.yale.edu/senselab/modeldb/>). Further analysis and display of simulation results was done in *Mathematica* 4.2. (Wolfram Research, Champaign, IL). Running one generation of the GA, including the construction of 64 neurons' morphologies (averaging 575 compartments/neuron), the electrophysiological simulations (60 ms) and the selection, mutation and recombination of the genomes of the initial population took approximately ~1 min on an Intel Xeon 2.4-GHz processor. All simulations were run on this processor under Windows XP or on an Athlon AMD 1.7 GHz processor under Linux.

The optimization of neural structure used in this study is divided into four separate but linked processes. These are the construction of the model neuron morphologies, the electrophysiological neuronal dynamics, the assessment of the fitness value of the neuronal dynamics and reproductive/genetic processes. We want to stress that only the final morphologies and simulations of the electrophysiological neuronal dynamics should be compared with biological data. The compartmental models we used are based on the "well-established" description of current flow in neurons by the cable equation (Rall 1995).

The four different processes are computed using four sets of equations (Fig. 1). They are the L-system to generate the neuronal morphology, the cable equations to calculate the membrane dynamics of the neurons, the fitness functions, and the genetic algorithm. The sets of equations are only weakly connected. By this we mean that one set of equations provides only parameters, initial or boundary conditions to the other set of equations. There is, however, no continuous coupling with the state variables (and thus dynamics) of the other equation set. For example, the GA provides parameters and initial conditions to the L-system. The state variables (m) are not affected by the GA during the growth of the dendritic tree. At the next stage, the

Present address and address for reprint requests and other correspondence: K. M. Stiefel, Theoretical and Experimental Neurobiology Unit, Okinawa Institute of Science and Technology, 12-22, Suzaki, Uruma, Okinawa 904-2234, Japan (E-mail: stiefel@oist.jp).

¹ Our use of the term "map" is different from the use in mathematics, where it denotes a time-discrete function. Rather we use it as an analogy for connecting a point in the function space to a region in the structure space. More concretely, we use it to describe finding members of a set of dendritic structures capable of performing a desired computational function. We do not imply that the "mapping" is one-to-one, that we find the complete set of adequate structures, or that we have an analytical expression for the connection between function and structure space (also see DISCUSSION).

The costs of publication of this article were defrayed in part by the payment of page charges. The article must therefore be hereby marked "advertisement" in accordance with 18 U.S.C. Section 1734 solely to indicate this fact.

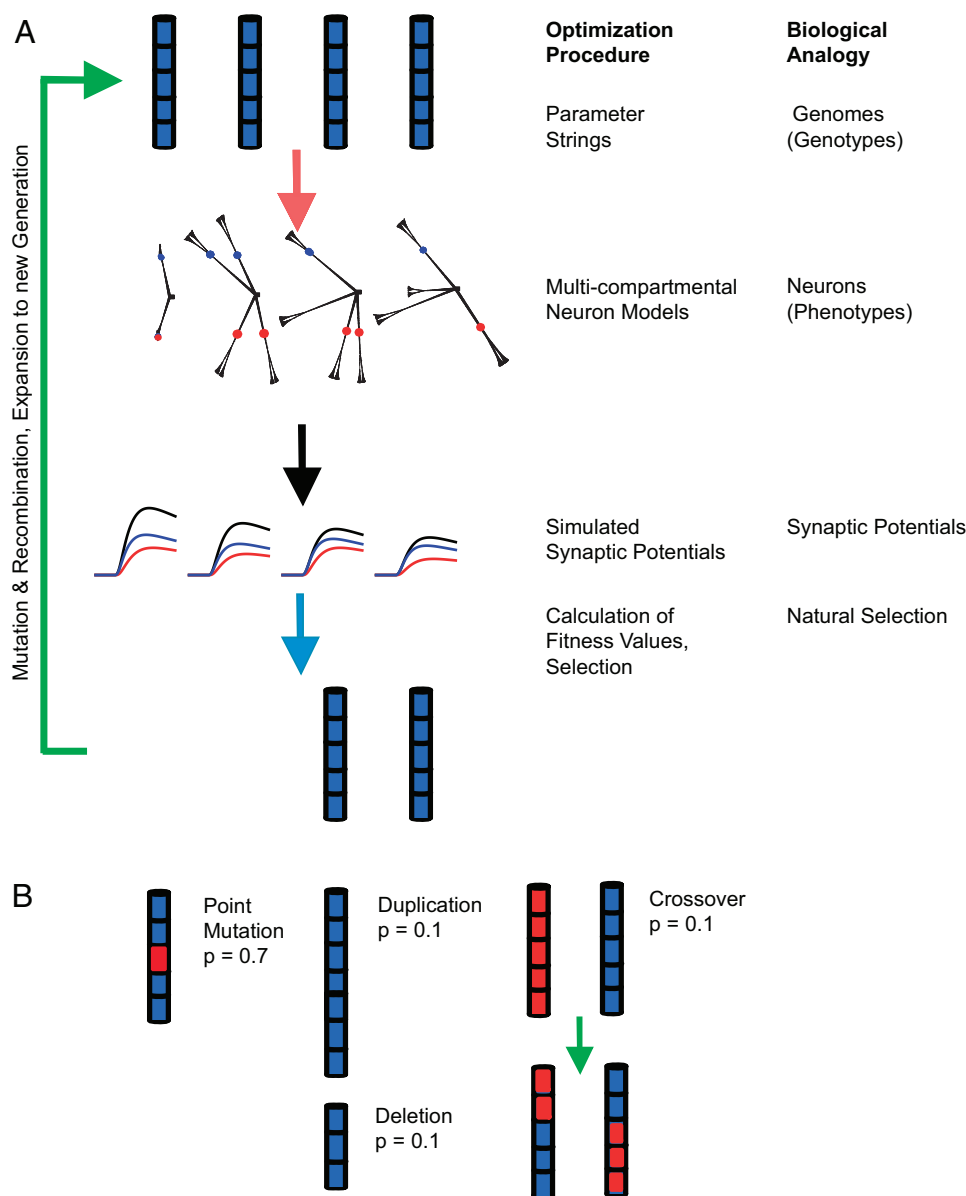


FIG. 1. Genetic algorithm (GA) used to optimize neuronal structure. **A:** flow diagram of the GA. The population of genomes is translated into neuronal phenotypes. The electrophysiological properties of these neurons are then tested in a computational task (linear summation of 2 input populations). The performances of the neurons are used to sort the corresponding genomes. The best-ranked genome is deterministically promoted to the next generation in an unchanged state (elitism). The remaining genomes above a cut-off are used to randomly draw the remaining members of the new generation. The genomes of the new generation are mutated and recombined and the algorithm restarts. **B:** sources of genetic variation. Genomes were altered by point mutation, duplication, deletion and crossover between 2 genomes. Probabilities of the respective operations are noted.

L-system sets boundary conditions and parameters for the electrophysiological simulations. The state variable, (V) is unaffected by the L-system during the simulations. Similar relationships hold between the electrophysiological simulations and the fitness function and between the fitness function and the GA. The four sets of equations influence each other in a circular, unidirectional manner (Fig. 1). For example, the L-system influences only the electrophysiological simulations, but there is no influence in the reverse direction.

Construction and coding of neuronal morphologies

An important choice in applying GAs to a specific problem is how the possible solutions, in our case neuronal morphologies, are encoded in the artificial genome. We adopted a method recently proposed by Ascoli and Samsonovich et al. (Ascoli et al. 2001a,b; Samsonovich and Ascoli 2003) for constructing neuronal morphologies using a L-system. L-systems are parallel string rewriting systems in which a set of rules is applied to all the elements of a string at once. An L-system is interpreted by assigning a geometric operation to every symbol of the alphabet producing the string. The string then becomes a program for a construction process that resembles but does not

model a developmental process. This method encodes dendritic morphologies in a compact way and allows the GAs to search efficiently for optimized neuronal morphologies. The approach used by Samsonovich and Ascoli (2005) was simplified here to encode the input parameters and initial conditions to make the search more efficient. The main, but not only, modifications we made were the omission of a stochastic component in the recursive morphogenetic algorithm and the use of a simpler formalism for the determination of branching angles. The stochastic component was omitted because having such a component in the calculation of the fitness function of a GA “smooths out” the fitness landscape. Fitness peaks become less pronounced, and the duration of the optimization procedure is prolonged. This is because either the GA will miss peaks due to an unlucky random number generator seed or multiple evaluations have to be performed for every genome.

Our implementation is described in the following text (see also the NEURON simulation code provided as supplementary material). The set of structures this L-system can produce is not as rich in variety as neuronal dendrites. However, they share a number of features with natural dendrites: they are all branched, noncircular, and progressively thinning cable structures attached to a common sink, the soma. They

project away from the soma with a freely variable direction and rotational angle. These crucial features together with the ease of encoding them into a “genome” make them suitable subjects for our study.

The L-system constructs one dendritic tree (connected to the soma) at a time (Fig. 2). The parameters for the L-system of one dendritic tree are encoded in a string of 14 parameters (“genes”, Figs. 1 and 2, Table 1), and the length of the parameter string (genome), always a multiple of 14, determines the number of dendritic trees of a neuron. The parameter values are initially drawn from a normal distribution with the half-width given in Table 1, and are then subject to mutation, crossover during the optimization by the GA (see following text).

A dendritic tree starts out with a total terminal degree, m_0 . At every branch point, m is asymmetrically distributed to the two daughter branches, according to

$$m_n^{\text{left}} = \frac{m_{n-1}}{a(l) + 2} \quad (1)$$

for the left dendritic branch and

$$m_n^{\text{right}} = m_{n-1} - m_n^{\text{left}} \quad (2)$$

for the right dendritic branch, where $a(l)$ is the asymmetry as a function of dendritic path length, l . When a branch reaches $m = 1$, no more branching is possible and the dendrite terminates. For technical reasons, our use of the term asymmetry is different from the more

common use in the literature, where asymmetry is defined as $a = \Delta m / (\Sigma m + 2)$.

The total terminal degree, m_0 , the branching asymmetry, $a(l)$, and angle, $\theta(l)$, the dendritic diameter, $d(l)$, and the length to the next branch point, $L(l)$ determine the morphology of the dendritic tree. All but for m_0 vary as a function of path distance to the soma, l . Either they serve as initial conditions (m_0) or as parameters (all others) for the morphogenetic algorithm. They were measured in the case of Samsonovich et al. and were artificially evolved in our case. The genome encodes these initial conditions or parameters (see following text, Table 1). They are grouped in 14-tuples (blocks), each of which contains the initial conditions or parameters for a morphogenetic algorithm that will give rise to exactly one dendritic tree of the neuron. The number of 14-tuples determines the number of dendritic branches of a neuron. This number, and thus the number of dendritic branches, is not fixed but can change due to deletion or duplication (see following text).

The diameter of dendrites in the L-system was a linear function of l

$$d(l) = -l\Delta d + d_0 \quad (3)$$

which leads to a progressive thinning of the dendrite as it grows away from the soma. This is a good approximation of observed dendritic diameters. In the case of the length to branch point, the function was a Gaussian

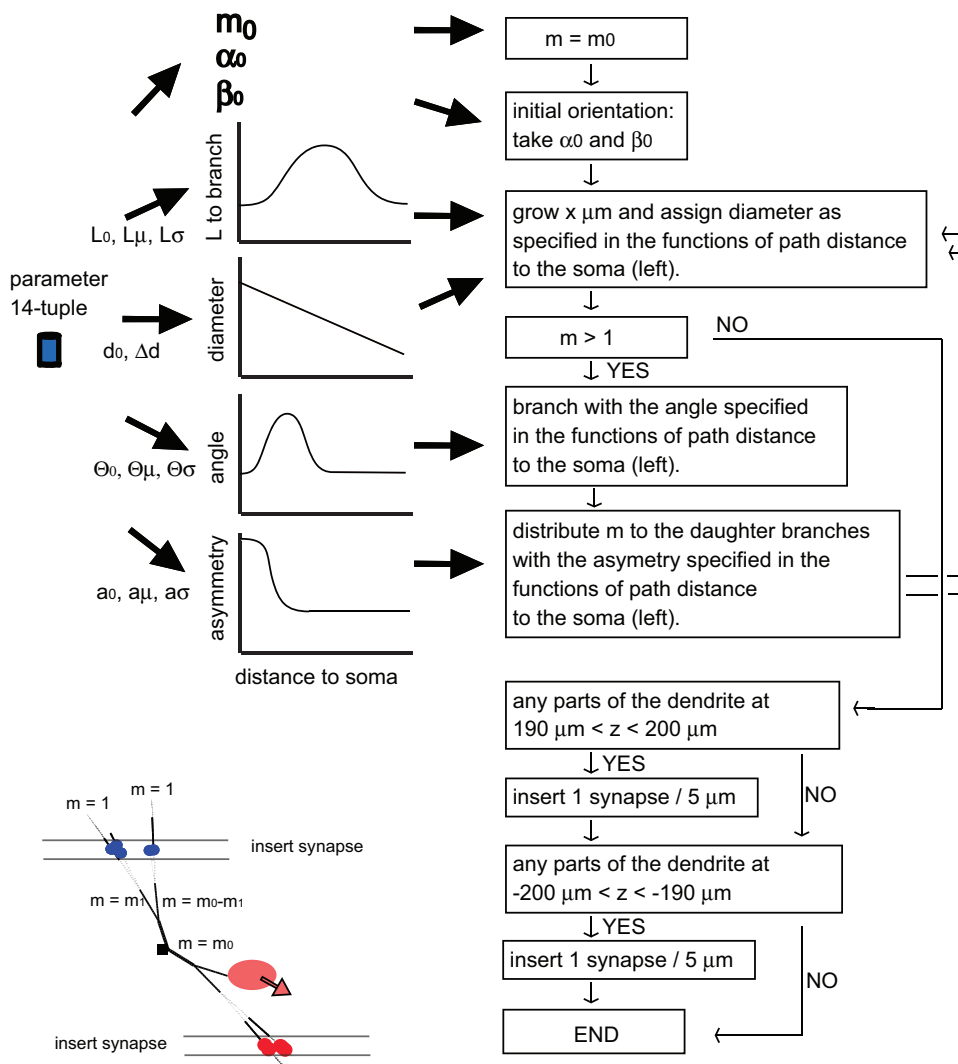


FIG. 2. Encoding of the neuronal morphology by the genome. One of the 14-tuples of parameters (left) directly specifies the values for the initial orientation of the dendrite (α_0 , β_0) and the terminal degree (m_0). The 14-tuples also specifies functions of parameters dependent on the path distance to the soma: the intersection and the slope for the function determining the dendritic diameter (d_0 , Δd) and the width, mean and amplitude (σ , μ , σ) for the functions determining the branching angle, asymmetry and length to branch point (middle). An L system uses these values and functions to build a dendritic tree. At each branch point, m is distributed onto the 2 daughter branches (hence 2 arrows originating from this box), until $m = 1$. Synapses are inserted in a location-dependent manner after completion of the dendritic tree. Right: each 14-tuple of parameters (Table 1) codes for 1 dendritic tree connected to the soma (inset bottom left).

TABLE 1. Encoding parameters and initial conditions of the recursive algorithm used to generate the neuronal morphologies

Index	Parameter	Encodes	Initial Values for GA
0	m_0	Terminal degree	32 ± 16
1	L_0	Peak length to branch point	100 ± 100
2	L_μ	Position of the peak length to branch point	100 ± 10
3	L_σ	Half width of the Gaussian determining the length to branch point.	200 ± 100
4	a_0	Peak asymmetry for the splitting of m .	0.5 ± 0.2
5	a_μ	Position of the peak asymmetry for the splitting of m .	100 ± 10
6	a_σ	Half width of the Gaussian determining the asymmetry for the splitting of m .	100 ± 10
7	Θ_0	Peak branching angle.	$\pi/10 \pm \pi/10^5$
8	Θ_μ	Position of the peak branching angle.	10 ± 10
9	Θ_σ	Half width of the Gaussian determining the branching angle.	100 ± 10
10	d_0	Initial dendritic diameter.	10 ± 2
11	Δd	Slope of the change of the dendritic diameter.	0.04 ± 0.004
12	α_0	Initial orientation.	$\pi/2 \pm 4\pi$
13	β_0	Initial orientation.	$0 \pm 4\pi$

The values are means \pm SD of the respective values in the initial population of neurons. Each 14-tuple parameters codes for one dendritic tree. The number of such 14-tuples in a parameter string determines the number of dendritic trees of a neuron.

$$L(l) = L_0 e^{\frac{(l-L_\mu)^2}{L_\sigma^2}} \quad (4)$$

where μ and σ are the position of the peak and the half-width of the function, respectively. The functions for the branching asymmetry, $a(l)$, and angle, $\theta(l)$, were analogous to Eq. 4. As in biological neurons, the dendritic diameter was not allowed to go $< 0.2 \mu\text{m}$. The size of the soma was fixed at $20 \mu\text{m}$ (length = diameter). The Gaussian function was chosen because it can, depending on the choice of parameters, approximate a number of other functions, such as constant or piecewise linear.

Synapses were attached to the dendritic tree whenever it passed through predetermined spatial regions (1 synapse per $5 \mu\text{m}$ dendrite). Two such regions were specified, one for synapses of each group (termed *left* and *right*). These regions were specified as layers, thus a synapse was attached to every $5 \mu\text{m}$ of a dendrite when it was within a pair of z coordinates. A consequence of this type of assignment of synapses to dendrites is that features of the dendritic tree which do not influence cable properties, such as the branching angle, influence dendritic function because they affect the placement of synapses.

In short, a parameter string composed of 14-tuples specified one neuron. Each 14-tuple of parameters specified one dendritic tree. The 14 parameters determine how an L-system builds this dendritic tree. Three of the parameters determined the initial orientation of the dendritic tree and the tree's terminal degree. The remaining parameters were used to calculate linear or Gaussian functions of the path distance to the soma. These functions determined when the dendritic tree branched, the angle and the terminal degrees of the daughter branches. They also determined the thickness of the dendrites. Whenever the dendrites passed through a predetermined region, a synapse was attached to every $5 \mu\text{m}$ of its length.

Test of computational performance: electrophysiological simulations

Simulations of the neuronal dynamics were run after the L-system constructed the dendritic trees. The simulations consisted of solving the coupled ordinary differential equations describing a circuit-equivalent multi-compartmental model of a neuron (Johnston and Wu 1995; Rall 1995). The membrane potential was governed by

$$c_m \frac{dV_n}{dt} = -g_l(V_n - E_l) - g_{ax}(V_n - V_{n-1}) - g_{ax}(V_n - V_{n+1}) - H(t_s)g_s(V - E_s)(e^{\frac{t-t_s}{\tau_2}} - e^{\frac{t-t_s}{\tau_1}}) \quad (5)$$

where c_m is the membrane capacitance, V_n , V_{n-1} and V_{n+1} are the voltages of the current and adjacent compartments, E_l and E_s are the reversal potentials of the leak and synaptic currents, g_l , g_{ax} , and g_s are the conductances of the leak, axial, and synaptic currents, respectively, $H(t)$ is the Heaviside function and τ_1 and τ_2 and t_s are the synaptic time constants and onset time.

The last term was included only in compartments with a synapse and gave rise to an excitatory postsynaptic potential (EPSP). The synaptic onset time was set by a presynaptic input stream. There were two inputs streams, right and left (giving rise to EPSP_r and EPSP_l), one for each subsection of synapse-space. The coincident activation of both inputs is denoted as EPSP_{rl}, the activation of the right synapses 15 ms before the left synapses as EPSP_{rl}($\Delta t = 15$ ms) and the inverse order as EPSP_{lr}($\Delta t = -15$ ms). The maximum amplitude of these EPSPs is denoted as M , with the corresponding subscripts.

We did not include voltage-dependent (active) currents in the model. Although active currents are important, the first step is to explore the influence of the passive properties, which are the underlying basis for the electrophysiological behavior of a neuron. Many aspects of neuronal signal processing can already be explained due to passive properties (Rall 1995). The effects of active properties build on passive properties and often act synergistically with them. Therefore we expect that their inclusion would modify, but not necessarily qualitatively change our results. Below spiking threshold, many aspects of synaptic integration (saturation, shunt) are governed by passive properties. Here we set out to develop a novel method for mapping neuronal function onto structure, and demonstrate its usefulness. In the interest of simplicity, we are using a passive model at this stage (but see DISCUSSION).

Linear summation task

The fitness of each neuron was determined by a cost function that was chosen to measure its performance on a computational task. The greater the fitness value, the better the performance. The first task was the *linear summation* of synaptic potentials (EPSPs). This is a non-trivial but relatively simple problem for which a solution is known and should thus serve as a good starting point: nontrivial because a linear input-output relationship runs counter to the sublinear nature of the cable properties of passive dendrites; simple because it can be described in a few words or with a short equation. Second, there is a class of neurons in the avian auditory brain stem (*nucleus laminaris*, Fig. 3C) that is believed to perform a similar function (Carr et al. 2005). These neurons compute interaural time differences. They react to the simultaneous activation of two groups of synapses but not to massive activity of a single input stream. This is difficult to compute with a neuron because cross-saturation occurs between co-activated spatially adjacent synapses. Cross-saturation is the reduction of each other's *driving force* [of the term $(V - E_s)$ in Eq. 5] and local input resistance, and leads to a reduced compound EPSP. The compound EPSP will be smaller than the algebraic sum of the individual EPSPs and sum sublinearly. To maximize the compound EPSP, cross-saturation has to be avoided. The neurons in the *nucleus laminaris* achieve this by placing their synapses at the ends of two long and thin

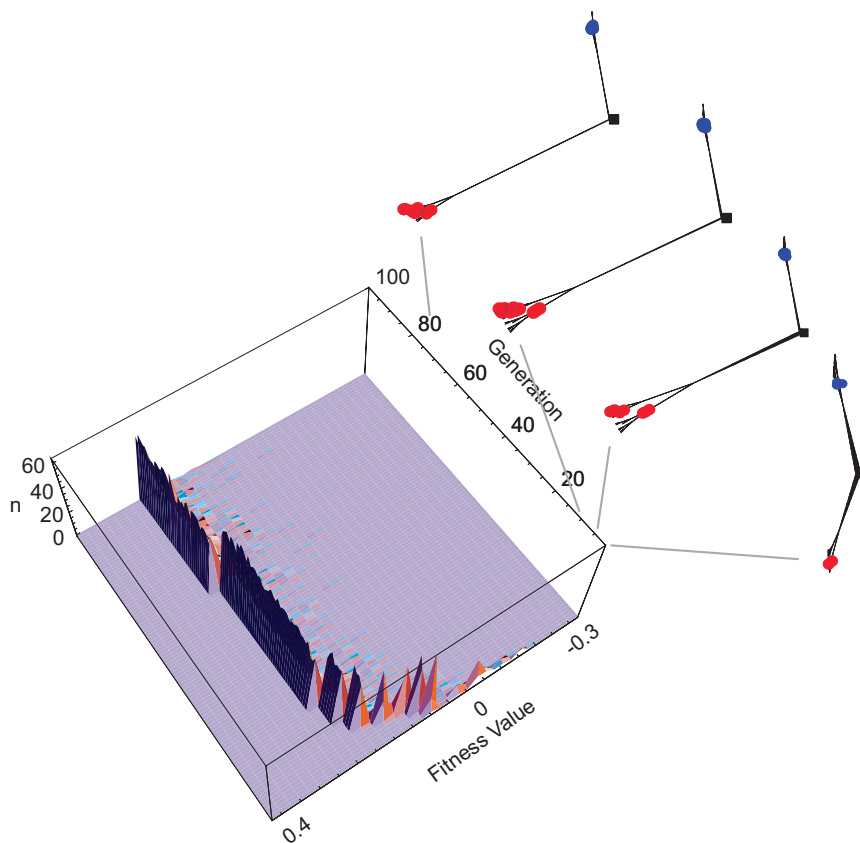


FIG. 3. Selection for linear summation. Simulations using a GA with 64 parameter strings and 400 generations. Time-resolved fitness-histograms (only the 1st 100 generations are shown). The number of parameter strings, n , in the vertical axis is plotted as a function of generation and fitness. Morphologies of the best solutions from at different stages of the optimization are shown. The red and blue dots correspond to *right* and *left* synapses.

dendrites originating from the soma. Due to the large electrotonic distance between them, the mutual influence on the driving force will be minimized. This class of neurons is one of the few for which the structure–function relationship is understood. The function performed by these neurons can be interpreted as an AND operation or as a linear summation. Irrespectively, they are optimized to react as strongly as possible to the joint activation of two groups of synapses. A linear sum of the individual EPSPs is the best possible performance which can be reached with a passive neuron model. We chose this problem to see if the optimization procedure finds a solution similar to the one observed in biological systems.

The goal was to select for small dendritic trees that summed large EPSPs of equal size in a linear manner. The fitness function for this task, F , was

$$F = -\frac{M_r + M_l - M_{rl}}{M_{rl}} - \alpha_1 \left(\frac{M_l}{M_r} + \frac{M_r}{M_l} \right) + \alpha_2 \log(M_{rl}) - \alpha_3 \frac{c}{c_0} \quad (6)$$

where M_l , M_r , and M_{rl} are the peak amplitudes of the excitatory postsynaptic potentials caused by synapses activated by a right, left, and both input streams, respectively, c and c_0 are the current and average initial dendritic tree sizes and α_1 , α_2 , and α_3 are weighing factors for terms two to four. This function is small when the arithmetic sum of the amplitudes of the individual EPSPs (M_l and M_r) is close to the EPSP amplitude observed when both synapses are activated at the same time (M_{rl}). Simulations with only the left, right, or both sets of synapses active were run, and the peak EPSP amplitudes were calculated from these simulations.

The second term in the fitness function punished unequal EPSP sizes to avoid the trivial solution of achieving linear summation by summing two EPSPs of very unequal sizes. A neuron which, due to the spatial configuration of its dendrites, received input not exceeding 0.2 mV from only the left or right group of synapses was assigned an extremely poor fitness value (−99).

The third term selected for large compound EPSPs.

The fourth term penalized unnecessarily large dendritic trees and was included to avoid the inclusion of superfluous, nonfunctional dendritic segments and to steer the optimization toward small neurons. c is the total number of segments of the current neuron and c_0 is the average number of segments in the initial population of neurons. The three additional terms were weighted with factors of $\alpha_1 = 0.1$, $\alpha_2 = 0.1$, and $\alpha_3 = 0.1$ relative to the term testing for linear summation. These values were selected according to the perceived relative importance of these terms.

Spike-order detection task

The second task was *spike-order detection*, which could be used in neural coding (VanRullen et al. 2005). We selected neuronal morphologies capable of distinguishing the temporal order in which EPSPs occur. A neuron successful in this task should react strongly (with a large compound EPSP) when the EPSPs were evoked in the temporal order left e right, but weakly when they were evoked in the order right e left, with onset times separated by 15 ms.

The fitness function for the spike-order detection task was

$$F = -\frac{M_{rl(\Delta t=15\text{ms})}}{M_{lr(\Delta t=15\text{ms})}} - \alpha_1 \left(\frac{M_l}{M_r} + \frac{M_r}{M_l} \right) + \alpha_2 \log(M_{rl}) - \alpha_3 \frac{c}{c_0} \quad (7)$$

with $\alpha_1 = 0.1$, $\alpha_2 = 0.1$, and $\alpha_3 = 0.02$. As before, better performance in the task corresponded to larger fitness values and the additional terms penalized small or unequal EPSP amplitudes and a large dendritic tree.

Selection

The initial population of genomes was drawn from a Gaussian distribution with the means and SDs given in Table 1. From these genomes, or from the genomes resulting from later generations, neurons were constructed and their membrane dynamics were simu-

lated. The results of these simulations were used to calculate fitness values, and the neurons were ranked according to their fitness values (Fig. 1A). The genomes giving rise to the neurons with the best ranks were then selected to populate the next generation. The selection probability was a half-Gaussian function of the rank with a peak at the best rank and a half-width equivalent to 50% of the population size. The selected genomes were used to replace 80% of the population drawn randomly in each generation. A randomly chosen proportion of the genomes (the remaining 20%) were carried over to slow down the loss of genetic diversity. Therefore not every genome in any given generation was present because of its success in the previous round of testing and selection.

The best individual genome was deterministically carried over to the next generation, a process called elitism. This avoided losing a rare beneficial genetic event due to the probabilistic selection process.

Mutation and recombination

Two types of genetic operators were used to introduce variation into the population, mutation and recombination (Fig. 1B). After creating a new generation of parameter sets, 50% were randomly drawn to be the subject of a stochastically chosen genetic operator.

The parameters were mutated by point mutations (relative probability of alteration of one parameter string, $P = 0.7$), deletions ($P = 0.1$), and duplications ($P = 0.1$). Point mutations consisted of multiplying a randomly chosen parameter with a random variable drawn from a Gaussian distribution with 1 ± 0.25 SD.

Care had to be taken to ensure the preservation of correct coding of the parameter sets for the L-system under the mutational operators (to not leave the language of the L-system). To that end, only complete sets of 14 parameters, each containing all the specifications for one dendritic tree, were subjected to deletion and duplication. A duplication consisted of taking a randomly chosen 14-tuple of parameters and attaching a copy of it at the end of the parameter string. This led to the addition of a new dendritic tree. The two parameters specifying the initial orientation were newly drawn for this 14-tuple to avoid superposition of the dendritic tree it specifies on the parent tree.

A deletion consisted of the opposite, deleting a randomly chosen 14-tuple and thus a dendritic tree. Deletions were not applied to neurons with only one tree.

The only type of recombination used was crossover, which occurred with $P = 0.1$. Two parent parameter string, M and N with m and n parameters gave rise to two daughters $G_N 0 - (x - 1)$, $G_M x - m - 1$ and $G_M 0 - (x - 1)$, $G_N x - n - 1$ with x being the crossover point. Thus the parameters were switched between strings up to the crossover point.

The parameter string of the neuron that performed best (the subject of *elitism*, see preceding text) was not subjected to the operators introducing genetic variation.

This population of neurons generated by selection and variation was used as a starting point for the next generation. Four hundred generations (25,600 fitness tests) were computed and the genomes, their fitness values and ranks were saved after every generation.

The parameters and exact algorithms used for the GA were chosen on a heuristic basis, and they produced successfully performing dendritic trees sometimes within only a few generations (see following text). It is not the objective of this study to explore the multiple parameter dimensions or implementation details of a GA. Rather we want to show that it is possible to use GAs to achieve a function-structure mapping of neuronal dendritic trees.

All values are given as means \pm SD.

Note that the terms gene, genome, mutation deletion, duplication, and recombination are used here in the sense of the genetic algorithm literature (Holland 1975; Mitchell 2001) in analogy to the genes encoded in the DNA of real biological organisms. We do not expect actual biological genes to individually determine the diameter,

branching angle, symmetry, and number of endpoints of dendrites as in our simulations (see Table 2).

The genetic algorithm, although inspired by real biological evolution and sharing some of its algorithmic structure (the interplay between selection and the introduction of variability), lacks its complexity. We use it only as an optimization technique and not as a model of evolutionary biology. Similarly, the construction of the dendritic tree is not a model of dendritic development but a method to parameterize branched, three-dimensional structures. The goal is not to simulate the evolution of vertebrate neurons, but to discover possible mapping of neuronal function to structure.

RESULTS

Selection for linear summation

The first task was to select for neurons performing well in linear summation of a compound EPSP. This necessitates the reduction of synaptic cross-saturation between two groups of synapses.

We ran 10 GA simulation runs with populations of 64 neurons for 400 generations using different random number generator seeds. In every GA run, the best neuron found was not present in the initial population but rather was generated by mutation and recombination from the neurons contained in previous generations.

The evolutionary development of neuronal shape in all runs qualitatively followed the same time course. In the time-resolved fitness histogram (Fig. 3), the initial population was represented as a shallow elevation at the bottom right edge, an initially relatively widely spread distribution of fitness values. Selection enriched a narrower ridge on the left side of the initial population. This ridge represents the currently best performing neuron. To its right, in lower-fitness bins are isolated, less fit genomes. These genomes remain from previ-

TABLE 2. Parameters of GA runs and electrophysiological simulations

Property	Value
GA	
Generations	400
Number of parameter strings (genomes)	64
Number of dendritic trees neuron	1–12
Parameters specifying one dendritic tree	14
Selection probability half-width	0.5 number of parameter string
Proportion of parameter strings replaced per generation	0.8
Total mutation & crossover probability per parameter string	0.5
Relative point mutation probability per parameter string	0.7
Relative deletion probability per parameter string	0.1
Relative duplication mutation probability per parameter string	0.1
Relative crossover probability per parameter string	0.1
Electrophysiology	
Compartments per segment	$2 + 1$ per $30 \mu\text{m}$
R_a	$100 \Omega \text{ cm}$
C_m	$0.8 \text{ pF } \mu\text{m}^2$
E_l	-70 mV
g_l	$2 \cdot 10^{-5} \text{ pS } \mu\text{m}^2$
τ_s on/off	0.2, 1 ms
g_s	$6 \cdot 10^{-4} \text{ pS}$
simulated time	20 to 100 ms

GA, genetic algorithm.

ous generations or have a decreased fitness due to deleterious mutation or recombination events after the last generation.

When a neuron with a better fitness value was stochastically generated by mutation or recombination, it accumulated in the population, which led to a collapse of the old ridge and the erection of a new ridge. Between these transitions the ridges were stable for 2–120 generations. These events were stochastic and varied between simulation runs, reflecting the stochastic nature of our simulations. In general, transitions became rarer with time. The magnitude of the jump to the left in the time-resolved fitness histogram corresponds to the amount of improvement in the computational task (increase in the fitness value). This magnitude was determined by the mapping of the genotype onto the phenotype as determined by the electrophysiological simulations. It varied around 0.02 and was not correlated with the generation at which the jump occurred. Although later increases in fitness value occurred, in most cases no or only minor improvements of the best performing neuron occurred after generation 100.

The best neurons, all of which had similar morphologies (Fig. 5D), generally had two thin dendrites (at the base: $0.89 \pm 1.92 \mu\text{m}$, range: 0.2–8.98 μm) projecting in different directions from the soma. Only one type of synapse was located on each dendrite. The dendrites thinned out further and branched once they entered the subspace containing synapses. The highly branched dendritic barbarizations were only 0.2 μm thin, the minimum diameter allowed by the L-system.

This dendritic tree was electrotonically large, especially during the activation of the synaptic conductances, and the distal tips were electrotonically isolated (Fig. 4). This led to a minimal change in input resistance and depolarization at the site of one set of synapses caused by activity at the other set of synapses. Cross-saturation between groups of synapses was thus minimized and the summation of the two EPSPs was close to linear ($M_{\text{pl}} = 14.7 \pm 2.87 \text{ mV}$, $M_{\text{l}} = 7.66 \pm 1.8 \text{ mV}$, $M_{\text{r}} =$

$6.53 \pm 1.31 \text{ mV}$). In fact, the amplitude of the compound EPSP (M_{pl}) was 0.992 times the arithmetic sum of the individual EPSP amplitudes ($M_{\text{r}} + M_{\text{l}}$). This is only 0.82% below optimum of 1 that is theoretically attainable with a passive dendritic tree. The average best fitness value reached after 400 generations was 0.233 ± 0.018 .

Comparison to nucleus laminaris neurons

The best performing neurons from each simulation run (Fig. 5) resembled the nucleus laminaris neurons in the crocodilian and avian brain stem and the medial superior olive neurons in the mammalian brain stem in a number of morphological and anatomical respects.

First the groups of synapses carrying each input stream were restricted to different primary dendrites in all classes of real neurons and in the artificial neurons (Grau-Serrat et al. 2003). In the real neurons, axons with auditory information originating from one ear synapse onto only one of the dendritic trees. This is reproduced in the artificial neurons. This is not an artifact of the zonal way of assigning synapses in our simulations, as a sideward oriented primary dendrite could carry synapses of both input populations.

The dendrites were thin (0.89 μm for artificial neuron and the 1.78 μm for the real avian neuron) (Grau-Serrat et al. 2003) and elongated. The lengths of the dendrites of auditory brain stem neurons is dependent on the sound frequency they are encoding and a quantitative comparison in this respect is therefore not possible since we stimulated the model neurons with only one EPSP. The dendrites of both the “artificially evolved” as well as the real neurons were, however, both morphologically as well as electrotonically significantly elongated. This led to an electrotonic separation of the synapses carrying the two input streams.

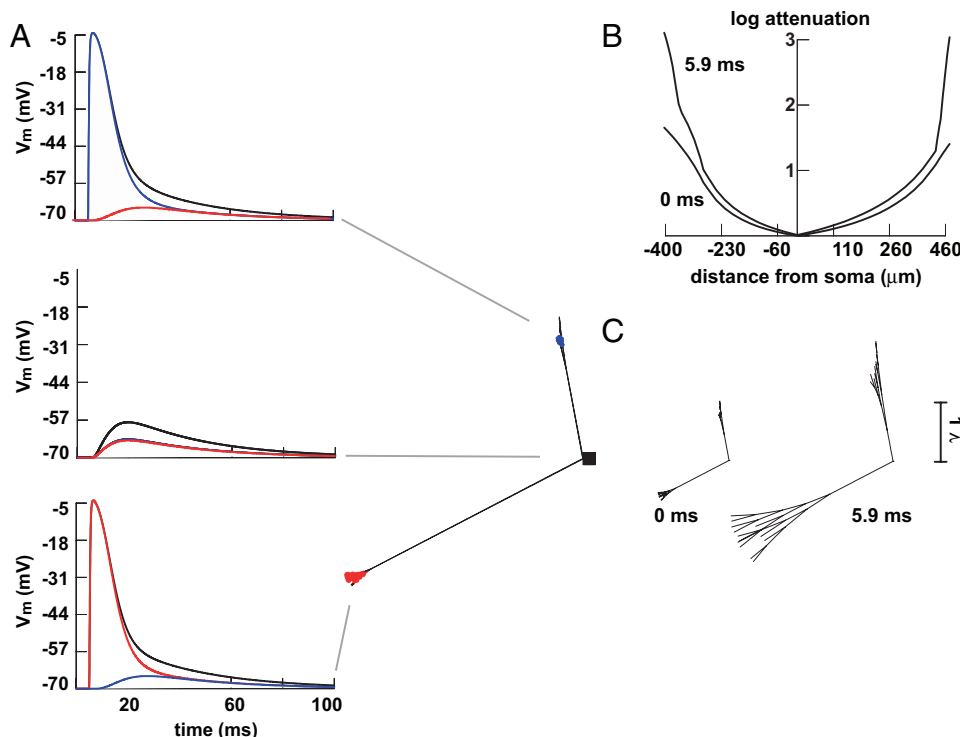


FIG. 4. The electrotonic properties of the best neurons optimized for the task of linear summation. *A*: excitatory postsynaptic potentials (EPSPs) at the soma and at the sites of the synapses with both groups (black) or only 1 group of synapses active (red/blue). *Middle*: neuronal morphology, indicating synapse position. *B*: voltage attenuation as a function of distance to the soma before (0 ms) and during the peak (5.9 ms) of the synaptic conductance. *C*: somatocentric electrotonic transformation of the neuronal morphologies before (0 ms, left) and during the peak (5.9 ms, right) of the synaptic conductance.

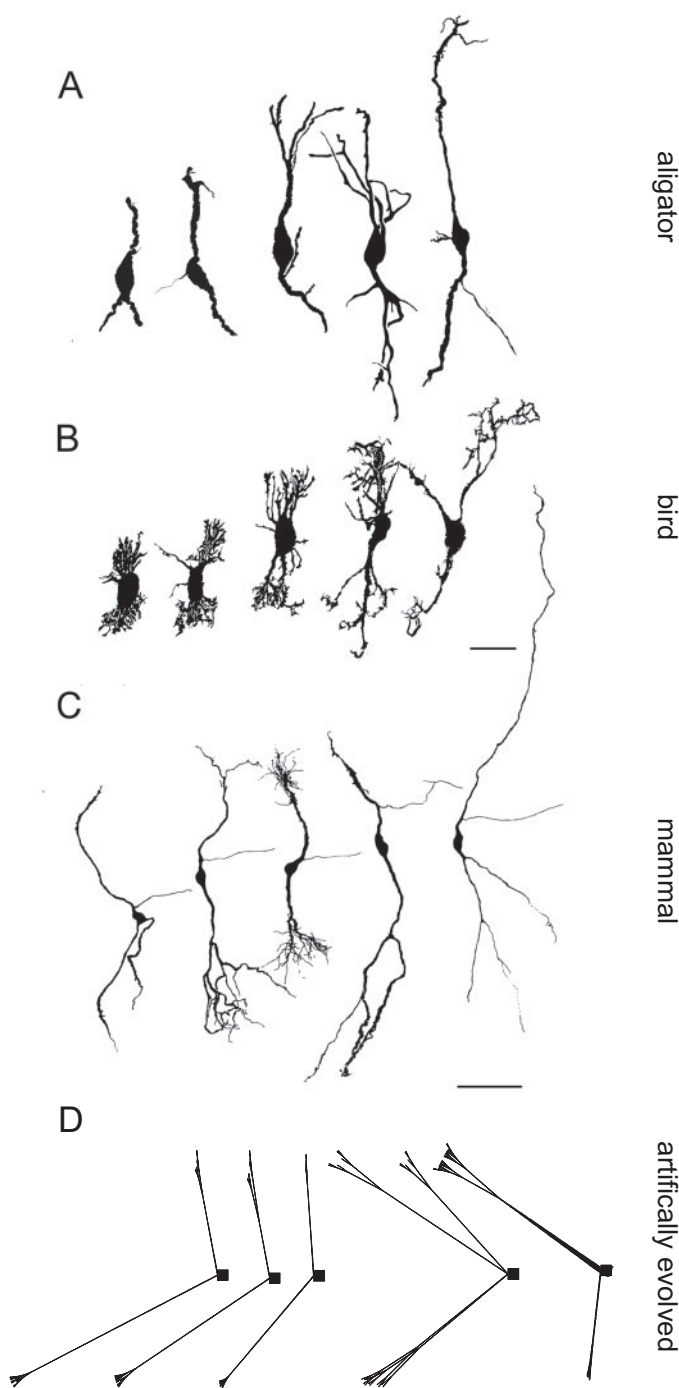


FIG. 5. Morphology of synaptically early auditory brain stem coincidence detector neurons. Golgi-stained neurons from an alligator nucleus laminaris (A), a bird nucleus laminaris (B), and a mammalian medial superior olive (C). Reproduced with permission from Carr et al. (2005). D: neurons artificially evolved toward linear summation of 2 groups of coincident inputs. The best performing neurons after 400 generations from 5 simulation runs with different random number generator seeds are shown.

Next we analyzed the parameter values that the GA had converged to when finding the most successful solutions. We compared the parameters of the best performing neurons after 400 generations from 10 different runs (using different random number generator seeds). We pooled the parameters of dendrites containing the right and left synapses, as their functions are symmetric. One exception are the initial orientation and

rotational angles, α_0 , and β_0 , which we treated separately for left (α_{0l} , β_{0l}) and right (α_{0r} , β_{0r}). We computed the mean and coefficient of variation ($CV = SD/mean$) of the parameters over all runs (Table 3). Surprisingly, only three parameters were conserved over all 10 runs ($CV < 0.15$), L_σ , α_{0r} , and α_{0l} . These parameters specify the half-width of the Gaussian function determining the length to branch point, and the initial orientation of both dendrites. What is therefore important for this task is that the length to branch point is initially large, but then drops quickly, so that a large number of short dendritic segments are localized in the synaptic regions. In addition, the dendrites must point in the correct direction. However, other similarities of the neuronal morphologies, such as the small diameter of the dendrites, are not reflected in values conserved across several genomes. This indicates that with the L-system and computational task used, the optimal type of morphology can be obtained in more than one way.

Active, voltage-dependent currents, have also been shown to contribute to the computational properties of the neurons in the nucleus laminaris (Parameshwaran-Iyer et al. 2003). These properties were not included in our simulations. However, they act synergistically with the passive properties, and the passive properties alone allow a good approximation of the behavior of these cells. Active currents can also be included in the optimization process, as discussed in the DISCUSSION.

There are also obvious differences between the “artificially evolved” neurons and nucleus laminaris neurons. The real neurons have more secondary dendrites branching off along the stem of the primary dendrite. This indicates that although a reduction of cross-saturation is one of the functions these neurons may have, there might be additional functions. Also the curvature of the real dendrites was not reproduced, although this was not relevant for their electrical properties.

Despite these minor differences, the “artificially evolved” neurons function according to the same principle as nucleus laminaris neurons (Carr et al. 2005), by electrotonic separation of groups of inputs. We have used the GA to automatically establish a mapping of a function, linear summation, to a neuronal structure, elongated dendrites with separate sets of synapses, and have found a real type of neuron that resembles this structure. These results confirm the structure–function relationship believed to exist in auditory brain stem neurons. The GA produced dendritic trees which are optimized for linear summation, and they resembled the nucleus laminaris neurons in important respects.

Selection for spike-order detection

Next we used a GA to find neurons that performed spike-order detection (Figs. 6 and 7). The desired performance in this task was to react strongly when the EPSPs were evoked in the temporal order left to right, but weakly when evoked in the order right to left. As in the first task, high-performance solutions evolved that were not present in the initial population. The evolutionary dynamics were similar to those observed when selecting for linear summation, although the best performing neurons were found more slowly.

The best neurons responded differentially to the order of inputs by making the low-pass filter properties of the dendrites carrying the two groups of synapses as different as possible. This was achieved by placing the two groups of synapses on

TABLE 3. Parameters of best-performing neurons in the linear summation task

Parameter	m_0	L_0	L_μ	L_σ	a_0	a_μ	a_σ	Θ_0	Θ_μ	Θ_σ	d_0	Δd	α_{0r}	α_{0l}	β_{0r}	β_{0l}
Mean	8.57	316.4	67.2	200.6	0.82	102.6	106.0	0.35	13.9	96.06	2.92	0.1	0.92	2	0.9	0.02
CV	0.36	0.23	0.47	0.06	0.87	0.23	0.19	0.16	0.43	0.28	0.59	0.57	0.109	0.006	2.82	89.7

Mean and coefficient of variation (CV) of parameters of the best-performing neurons in the linear summation task after 400 generations from 10 different runs (using different random number generator seeds). All parameters but α_{0l} and β_{0r} , were pooled for the dendrites carrying the left ($n = 14$) and right ($n = 15$) type synapses.

dendrites with different diameters. The dendrites carrying the right synapses were thick (at the base: $1.5 \pm 1.23 \mu\text{m}$), unbranched, and were thus weak low-pass filters. They did not strongly filter the EPSP_r and its waveform was preserved at the soma. In contrast, the dendrites carrying the left synapses were thin (at the base: $0.8 \pm 0.72 \mu\text{m}$) and branched into the segments ($0.2 \mu\text{m}$ diam) carrying the synapses. These dendrites strongly low-pass filtered the EPSP_l and its waveform was much flatter at the soma than at the synapse. A side effect of the increased low-pass filtering was an increased amplitude reduction during propagation to the soma compared with the other group of inputs (Fig. 8).

Because of the different waveforms, the EPSPs summed differently when the order was interchanged. When EPSP_l ($M_l = 10.91 \pm 0.78 \text{ mV}$) preceded EPSP_r ($M_r = 7.13 \pm 0.59 \text{ mV}$) by 15 ms, EPSP_r started from a higher voltage, bringing the peak amplitude of the compound EPSP_{lr($\Delta t=15\text{ms}$)}} to $M_{lr(\Delta t=15\text{ms})} = 16.89 \pm 1.01 \text{ mV}$. In contrast, when EPSP_l followed EPSP_r by 15 ms, EPSP_l started from a lower voltage and EPSP_{lr($\Delta t=15\text{ms}$)}} reached only $M_{rl(\Delta t=15\text{ms})} = 10.99 \pm 0.69 \text{ mV}$, which was only 0.08 mV larger than the amplitude of the individual M_r . The ratio $M_{lr(\Delta t=15\text{ms})}/M_{rl(\Delta t=15\text{ms})}$ in these neurons was 1.54, and their average fitness value was 0.041 ± 0.023 . In summary, the different low-pass filter properties of the dendrites bearing the two groups of synapses lead to

different EPSP shapes and thus to summation properties sensitive to the temporal order of EPSPs.

We analyzed the parameter values of the most successful solutions. In the case of the spike-order detection task, the two groups of dendrites carrying the right and left synapse had different properties so we calculated the mean and CV separately from the best performing neurons of 10 runs after 400 generations for both groups (Tables 4 and 5). As found in the first task, only a few parameters each were conserved over all 10 runs ($\text{CV} < 0.15$).

The first group of dendrites were the ones carrying the left synapses to which the neuron was supposed to react strongly when they were activated first and were therefore strong low-pass filters in all the optimized neurons. In these dendrites, a_μ and Θ_s were conserved over all parameter sets. These parameters specify the position of the peak of the Gaussian function determining the asymmetry for the splitting of m , and the half-width of the Gaussian function determining the branching angle. These two parameters work in conjunction to place the dendrites appropriately in the synaptic zones.

The second group of dendrites were the ones carrying the right synapses to which the neuron was expected to react strongly when they were activated second and were therefore weak low-pass filters in all the optimized neurons. In these dendrites, L_0 was conserved over all parameter sets. This

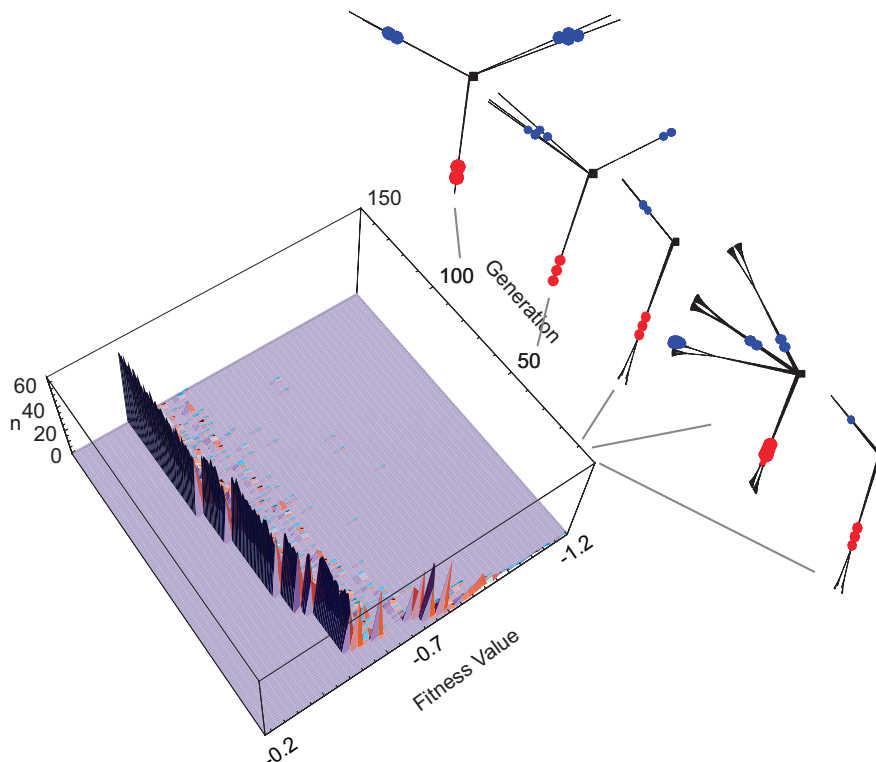


FIG. 6. Selection for spike-order detection. Simulations using a GA with 64 genomes and 400 generations. Time-resolved fitness-histograms (only the 1st 150 generations are shown). The number of genomes, n , in the vertical axis is plotted as a function of generation and fitness. Morphologies of the best solutions from at different stages of the optimization are shown. The red and blue dots correspond to right and left synapses.

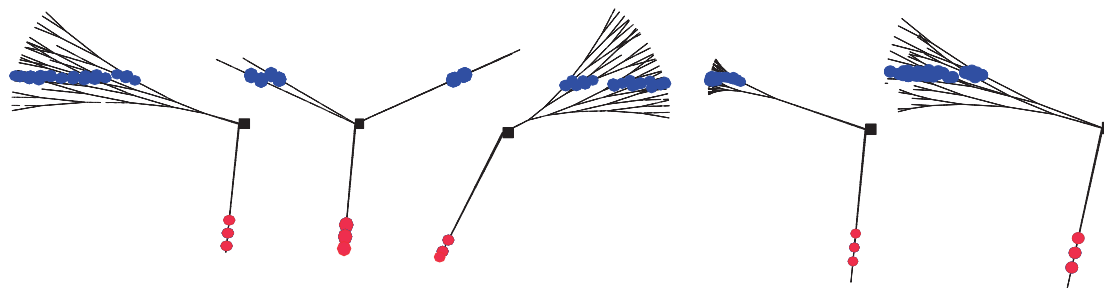


FIG. 7. Neurons artificially evolved toward spike-order detection. The best performing neurons after 400 generations from 5 simulation runs with different random number generator seeds are shown. The red and blue dots correspond to *right* (red) and *left* (blue) synapses.

parameter specifies the value of the peak of the Gaussian function determining the length to branch point.

The relatively small number of conserved parameters indicates that the task under investigation can be computed by neurons in a number of ways.

Comparison to cortical pyramidal neurons

Cortical pyramidal neurons, particularly the large pyramids of layer V, have two types of primary dendrites of significantly different diameter. In these neurons, the apical dendrites are $\sim 5 \mu\text{m}$ in diameter, whereas the basal dendrites are $\sim 1 \mu\text{m}$. A

prediction of our function-to-structure mapping is that EPSPs evoked in the apical and basal dendrites should sum with a higher peak amplitude when the EPSPs are first evoked in the basal dendrites compared with the reverse temporal order. This prediction should hold for EPSPs evoked at synapses at an approximately equal distance from the soma. Active, voltage-dependent properties, which are known to be different in basal and apical dendrites, can potentially modify the result of the passive dendritic signal processing. However, the responses obtained with passive properties are a good first approximation in many cases, especially when the EPSPs are small (Rall

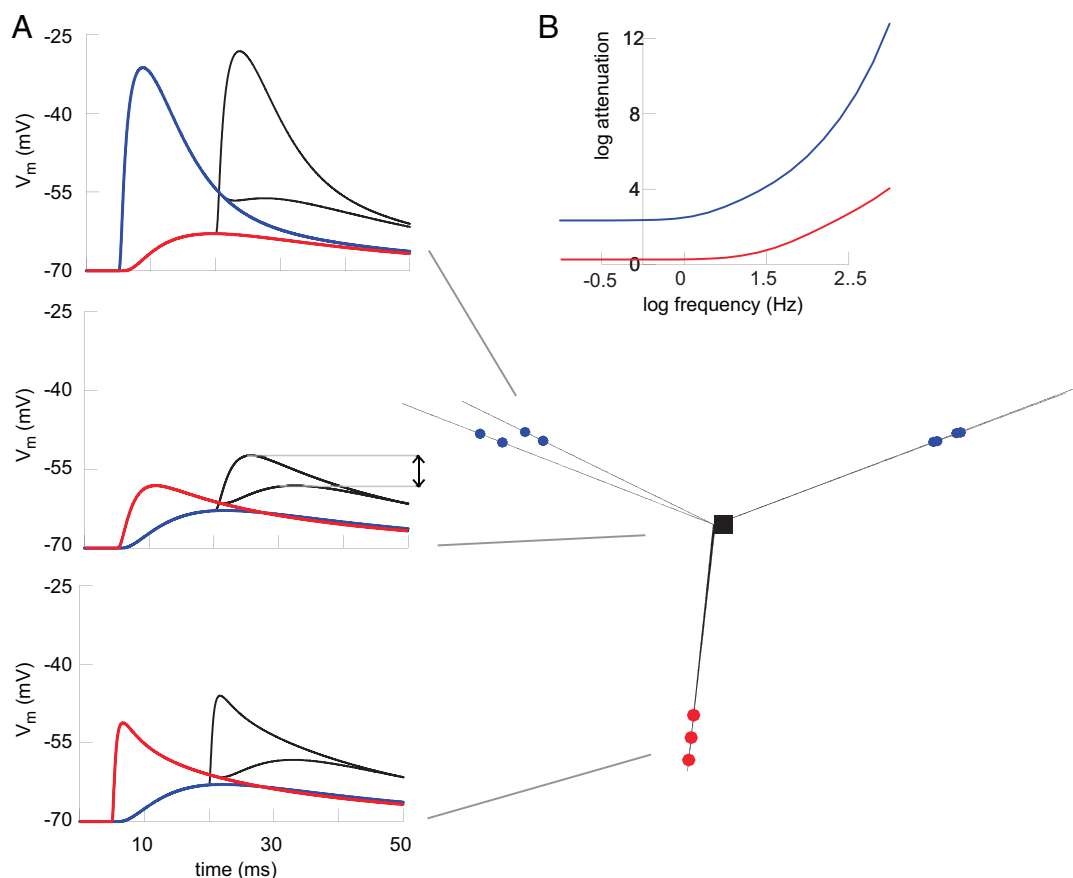


FIG. 8. The electrophysiological properties of the winners are optimized for the task of spike-order detection by making the low-pass filter properties of the dendrites as different as possible. Neurons are optimized to react strongly to the activation of synapses in the order *left* (blue) – *right* (red) but weakly to the reverse temporal order. Example winner neuron. *Left*: EPSPs evoked individually and in both temporal permutations at the soma and the sites of the synapses. The colors of the traces indicate the synapses that evoked the EPSPs, black traces indicate EPSPs evoked jointly by both groups of synapses. The larger amplitude black waveform corresponds to the activation of synapses in the temporal order *left* – *right*. The arrow indicates the difference between $M_{rl}(\Delta t = 15\text{ms})$ and $M_{lr}(\Delta t = 15\text{ms})$. *Middle*: neuronal morphology. The red and blue dots correspond to *right* and *left* synapses. *Inset*: low-pass filter path properties as a function of the frequency of the dendritic branches indicated by color.

TABLE 4. *Parameters of best-performing neurons in the spike-order detection task, left synapses*

Parameter	m_0	L_0	L_μ	L_σ	a_0	a_μ	a_σ	Θ_0	Θ_μ	Θ_σ	d_0	Δd	α_0	β_0
Mean	32	191.76	104.84	234.92	0.62	100.17	108.19	0.26	11.18	102.86	5.36	0.13	2.24	0.37
CV	0.53	0.88	0.6	0.32	0.39	0.01	0.3	0.46	0.45	0.09	0.65	0.64	0.46	1.09

Mean and CV of parameters of the best-performing neurons in the spike-order detection task after 400 generations from 10 different runs (using different random number generator seeds). These are the values of the dendrites carrying the *left* synapses ($n = 15$), which the neuron is supposed to react strongly to when activated first. Therefore these dendrites are strong low-pass filters.

1995). This study thus suggests that one possible function of cortical pyramidal neurons is to more strongly respond to basal EPSPs followed by apical EPSPs than to the reverse order. This is an experimentally testable prediction of our theoretical approach.

DISCUSSION

This study is the first report of the optimization of biophysically realistic neurons for solving a computational task. GAs acting on the parameters of an L-system are capable of finding neuronal morphologies for nontrivial computational tasks. Starting from random initial values, similar solutions were found by an optimization process composed of fitness value jumps at variable times of variable magnitudes. The number of generations needed to reach the best solution was about the same for a given task.

Two different computational functions were studied. When selecting for linear summation of EPSPs, neuron morphologies were found that had groups of synapses located distantly from each other on long, thin dendrites that were electrotonically isolated and thus ideally suited for the task of linear summation.

When selecting for temporal spike-order detection, neuron morphologies were found that made the low-pass filter properties of the dendrites carrying the two groups of synapses as different as possible. By comparing the morphology to real neurons, we were able to predict a new property of cortical pyramidal cells that had not been suggested previously: They should react more strongly to compound EPSPs evoked in the proximal basal before the apical dendrites than to the reverse order. This novel prediction can be tested experimentally, possibly using laser-uncaging of glutamate at two different sites on the dendrite, milliseconds apart, in both temporal permutations (Shoham et al. 2005). There are, of course, features of pyramidal neuron morphology that are not replicated by the artificially evolved morphologies. This indicates that spike-order detection may be only one of the functions of dendrites in pyramidal neurons. In general, the mapping from functions to morphologies is many to one.

Although a human armed with knowledge of dendritic cable properties (Rall 1995) could have designed neuronal morphologies to perform these tasks, *this knowledge was not explicitly*

built into the GA. Rather solutions were independently discovered by the GA on the basis of variation and selection. By examining these models, two insights can be gained: first, that synaptic cross saturation can be minimized by electrotonically separating synapses on long and thin dendrites or by minimizing the input resistance of a neuron and second, that the low-pass filter properties of dendrites vary as a function of their thickness. These are well-known properties of dendrites (Rall 1995). What is novel is that the method *automatically rediscovered* these dendritic properties.

The relatively small number of parameters that were conserved over all the dendrites fulfilling one type of function is indicative of a one-to-many mapping of structures to functions. In the study of Achard and De Schutter (2006), the regions of parameter space leading to a satisfactory fit of the model results to experimental data were “foam”-shaped and connected but irregular. In our study, combinations of parameters rather than definite values of single parameters were important for the generation of functional neurons.

The broader significance of our study is that the same method can be used for automatically establishing function-structure relationships for many other neuronal functions (Fig. 9). Only a few such relationships are known with any certainty. Previously, such relationships could only be found by the unreliable process of guessing and testing. Humans will inevitably have biases that might prevent the discovery of certain hypotheses. The automated optimization method presented here significantly speeds up the process of discovery with different biases.

Comparison of artificial and biological neurons

L-systems, which were used in this study to generate neural morphologies, can generate artificial morphologies that are statistically virtually indistinguishable from real neurons (Samsonovich and Ascoli 2005).

A class of solutions for the task of linear summation was found that resembles the neuronal morphologies of real neurons believed to carry out a task we selected for: synaptic summation avoiding cross-saturation. These neurons, located in the auditory brain stem of birds and mammals, react most strongly when auditory signals from both ears coincide but respond suboptimally to massive input from one ear only. This

TABLE 5. *Parameters of best-performing neurons in the spike-order detection task, right synapses*

Parameter	m_0	L_0	L_μ	L_σ	a_0	a_μ	a_σ	Θ_0	Θ_μ	Θ_σ	d_0	Δd	α_0	β_0
Mean	5.67	305.4	101.95	171.43	0.63	106.7	90.85	0.23	13.03	132.3	4.99	0.1	0.07	-0.05
CV	1.12	0.09	0.46	0.4	0.31	0.26	0.32	0.58	0.66	0.61	0.22	0.46	1.3	-2.87

Mean and CV of parameters of the best-performing neurons in the spike-order detection task after 400 generations from 10 different runs (using different random number generator seeds). These are the values of the dendrites carrying the *right* synapses ($n = 12$), which the neuron is supposed to react strongly to when activated second. Therefore these dendrites are weak low-pass filters.

Function Space → Genetic Algorithm → Morphology Space

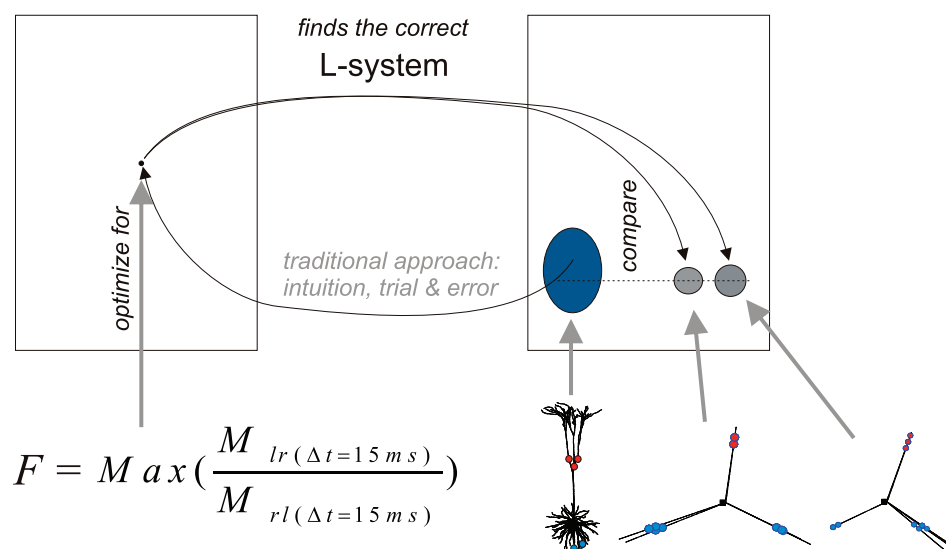


FIG. 9. Mapping of neuronal function to structure. Schematic depiction of the function space (left), which is automatically mapped onto the morphology space (right) by the L-system. The GA finds the L-system for the mapping that maximizes the fitness function F . The computation of spike-order detection is a member of the function space. It is mapped onto the 2 populations of best performing artificial neurons found (gray). They are then compared with a reconstruction of a real neuron (blue). Similarities (indicated by alignment along 1 axis) suggest a common function. The traditional approach of the discovery of neuronal structure-function relationships, going in the reverse direction, is indicated below.

can be achieved by electrotonically isolating the inputs from the left and right ear, something that can be implemented by separating them on two long thin dendrites. We did not include active currents in the dendrites, which might have altered the solutions (Agmon-Snir et al. 1998).

The solutions found for spike-order detection had one set of thick and one set of thin primary dendrites. This is a salient feature of cortical pyramidal neurons.

These two examples demonstrate the usefulness of our method (Fig. 8). In both cases, a computation was first proposed. Then the optimization was used to automatically determine neuronal morphologies capable of carrying out these computations. It essentially found an L-system that maps from a function in function space to a set of morphologies in morphology space. A comparison was made in morphology space between the artificial neuron and a real neuron. The similarity suggested that the real neuron could perform the computation that the artificial neuron was selected for. In the case of selection for linear summation, a similarity was found between the artificial neurons and neurons of the avian brain stem *nucleus laminaris*. This confirmed a hypothesis about the structure-function relationship in these neurons. In the case of selection for spike-order detection, a similarity was found between the artificial neurons and cortical pyramidal neurons. This generated a hypothesis about the structure-function relationship in pyramids that can be experimentally tested.

Collecting input signals constitutes another evolutionary pressure on neural morphology. This would mean optimizing axonal arbors as well as dendritic trees. In axons, the propagation of an action potential is regenerative, and therefore the addition of more axonal length would not change the height or shape of the signal at the termination of the axon. Axons are thus more flexible than the dendrites in terms of morphology. Where this might prove to be essential is in structures like the cerebellum where the geometry may be dominated by the need to maximize synaptic inputs. Some work along these lines has already been taken up by Chlovskii and colleagues (Chen et al. 2006).

In most instances, the real and the artificially optimized neuron will not be completely identical. Rather they will share

only certain aspects of their morphologies, which are the critical ones for the computation under question. Typically, we expect the real neurons to be more complex as they most likely evolved to fulfill more than one function.

We also asked whether the solutions found by nature are unique or one of several possible solutions (whether or not the function to structure mapping is degenerate). If alternative solutions are found by the optimization, the question arises whether the alternative are found in nature. If not, it is possible that additional biological constraints are needed in the optimization.

Rather than trying to guess the function of a particular neuron, it should be possible to generate a large dictionary of morphologies, based on a large set of possible functions, and to find the closest of the neurons in the dictionary because each entry in the dictionary is labeled with the function that generated it. The match automatically provides a possible computational function for the neuron (Fig. 9).

An important point is to distinguish between finding a structure-function relationship and understanding the nature of this relationship. The first can be achieved automatically by our method. The second, once a neural model able to perform the desired function has been found, still needs the knowledge and insight of the scientist. In this sense, our method does not replace the human but acts as an "intuition pump."

Relation to previous work

GAs have been applied to L-systems modeling plant growth (Jacob 1995). The aim was to maximize the number of leaves and flowers on a plant. GAs were extremely successful in solving this problem. Solutions were found in even fewer generations than in our study possibly because leaf and flower number is a more direct consequence of the structure generated by the L-system than electrophysiological performance.

GAs have also been used to evolve highly abstract models of nervous systems, artificial neural networks. These approaches have led to useful solutions for problems in machine learning (Jacob 1993; Moriarty 1997, 1998).

There have been a number of studies where GAs were used to search the parameter space of biophysically realistic single-neuron simulations (Achard and De Schutter 2006; Eichler West 1997; Keren et al. 2005; Vanier and Bower 1999). In these studies, GAs were used to find the conductance densities for a model of a hippocampal pyramidal neuron so that it would most closely reproduce experimental data. This approach successfully solved the problem, frequently encountered in neuronal modeling, of fitting a model to experimental data. This serves a different purpose than our approach, which searches for neurons that optimally perform a certain computation.

Future directions

We included only passive properties in the neuron model used in this study as a first step. The next step is to include active conductances in the optimized neurons. As with dendritic morphologies, there is a wide variety of active conductances in neurons (Johnston and Wu 1995) that are differentially distributed on the dendritic trees (Colbert et al. 1997; Hoffman et al. 1997; Magee 1999). The application of our method will help elucidate the computational significance of these differences. The studies mentioned in the preceding text that use GAs to fit models to experimental data (Achard and De Schutter 2006; Eichler West 1997; Keren et al. 2005; Vanier and Bower 1999) indicate that GAs can find solutions in higher-dimensional parameters spaces that include the parameters for active conductances.

In principle, active properties of neurons can be encoded in the parameters of an L-system along with passive parameters, such as the dendritic diameter or branching angle. The increased dimensionality and the stronger parameter sensitivity of active systems will, however, make this a more challenging task for the method presented here. It remains to be seen how successful it will be in such tasks. A fine tuning of the GA (mutation rate, population size etc.) might be necessary to tackle these advanced problems. In this sense, the current report constitutes a first step in a research program designed to automatically determine function-structure relationships in neurons.

Some computational functions of neurons are more reliant on the passive properties of dendrites, others on active properties. The two functions we investigated in this report are of the first type. The GA found neurons that could perform these functions quite well. There may be other functions that cannot be easily computed with passive dendrites. It will be interesting to investigate which functions necessitate the presence of active currents.

The present study has investigated instances of the mapping of computational tasks onto neural morphologies. Libraries of reconstructed neurons are available as well as sophisticated software packages to determine their statistical properties (Ascoli et al. 2001b; Samsonovich and Ascoli 2003, 2005). It would be intriguing to comprehensively map computational tasks onto morphologies. An automatization of not only the first step, the mapping of function to structure, but also the second step, the comparison of the optimized neurons to real neurons, will further increase the power of this method. This would generate predictions for the types of computations that

different classes of neurons evolved to perform in the functioning brain.

APPENDIX: USAGE OF THE SIMULATION CODE

The simulation code used in this study is available via the Senselab model database (<http://senselab.med.yale.edu/senselab/modeldb/>) or can be obtained from K.M.S. To run the simulations, install NEURON (www.neuron.yale.edu), and execute GA.hoc.

To look at the best-performing neuron in every generation, execute LOAD_WINNER.hoc, enter the desired generation and press the "load" button. To modify the basic parameters of the GA (mutation rates, population size, random number generator seed), change them in the first block of GA.hoc. The procedure performing the fitness test is called *lineartest()*, executed at line 99 of GA.hoc and contained in the file *lineartest.hoc*. The actual score of a neuron is computed in a subroutine of *lineartest()* called *scoreme()*, at line 11. There, *score[]* is assigned the sum of the variables *linearity* (computing the linearity of the EPSP summation), *equalize* (computing how close in size the EPSPs are), and *dsize* (computing the size of the dendritic tree as a fraction of the average dendritic tree of the 1st generation). The user can easily change the calculation of *score[]* here and for instance select for neurons that sum EPSPs in a particular sublinear manner.

In *lineartest()* and *linearrun()*, contained in *lineartest.hoc*, the times at which the EPSPs are evoked are specified, and the simulations are executed. A wide variety of computational functions can be specified by altering these routines. The user can also execute a completely different procedure at line 99 in GA.hoc. The fitness values need to be passed into the array variable *score[]* (with the array size equal to the population size) and the electrophysiological simulations should then be executed.

ACKNOWLEDGMENTS

We thank Drs. Michael Hines and Ted Carnevale for supporting NEURON, the Computational Neurobiology Laboratory at the Salk Institute, and Dr. Diego Raskin-Gutman for discussions and Dr. Melanie Mitchell for introducing K. M. Steifel to GAs at the 2001 Santa Fe Institute Summer School. We also thank the anonymous reviewers for valuable suggestions regarding the presentation and interpretation of our results.

GRANTS

This research was supported by the Deutsche Forschungsgemeinschaft to K. M. Steifel and the Howard Hughes Medical Institute to T. J. Sejnowski.

REFERENCES

- Achard P, De Schutter E. Complex parameter landscape for a complex neuron model. *PLoS Comput Biol* 2: e94, 2006.
- Agmon-Snir H, Carr CE, Rinzel J. The role of dendrites in auditory coincidence detection. *Nature* 393: 268–272, 1998.
- Ascoli GA, Krichmar JL, Nasuto SJ, Senft SL. Generation, description and storage of dendritic morphology data. *Philos Trans R Soc Lond B Biol Sci* 356: 1131–1145, 2001a.
- Ascoli GA, Krichmar JL, Scorcioni R, Nasuto SJ, Senft SL. Computer generation and quantitative morphometric analysis of virtual neurons. *Anat Embryol* 204: 283–301, 2001b.
- Carr CE, Iyer S, Soares D, Kalluri S, Simon JZ. Are neurons adapted for specific computations? examples from temporal coding in the auditory system. In: *23 Problems in Systems Neuroscience*, edited by Sejnowski TS, Van Hemmen L. Oxford, UK: Oxford Univ. Press, 2005, p. 245–265.
- Chen BL, Hall DH, Chklovskii DB. Wiring optimization can relate neuronal structure and function. *Proc Nat Acad Sci* 103: 4723–4728, 2006.
- Colbert CM, Magee JC, Hoffman DA, Johnston D. Slow recovery from inactivation of Na⁺ channels underlies the activity-dependent attenuation of dendritic action potentials in hippocampal CA1 pyramidal neurons. *J Neurosci* 17: 6512–6521, 1997.
- Eichler West RMW GL. Robust parameter selection for compartmental models of neurons using evolutionary algorithms. In: *Computational Neuroscience Conference 1997*. New York: Plenum, 1997, p. 75–80.

- Grau-Serrat V, Carr CE, Simon JZ.** Modeling coincidence detection in nucleus laminaris. *Biol Cybern* 89: 388–396, 2003.
- Hines ML, Carnevale NT.** The NEURON simulation environment. *Neural Comput* 9: 1179–1209, 1997.
- Hines ML, Carnevale NT.** Expanding NEURON's repertoire of mechanisms with NMODL. *Neural Comput* 12: 995–1007, 2000.
- Hoffman DA, Magee JC, Colbert CM, Johnston D.** K^+ channel regulation of signal propagation in dendrites of hippocampal pyramidal neurons. *Nature* 387: 869–875, 1997.
- Holland JH.** *Adaptation in Natural and Artificial Systems*. Cambridge, MA: MIT Press, 1975.
- Jacob CRJ.** Evolution of neural net architectures by a Hierarchical grammar-based genetic system. *International Conference on Artificial Neural Networks and Genetic Algorithms*, Innsbruck, Austria: Springer, 1993, p. 72–79.
- Jacob C.** Genetic L-system programming: breeding and evolving artificial flowers with Mathematica. In: *IMS'95, Proc. First International Mathematics Symposium, Southampton, Great Britain*. Computational Mechanics Publication, 1995, p. 215–222.
- Johnston D, Wu SM-S.** *Foundations of Cellular Neurophysiology*. Cambridge, MA: MIT Press, 1995.
- Keren N, Peled N, Korngreen A.** Constraining compartmental models using multiple voltage-recordings and genetic algorithms. *J Neurophysiol* 94: 3730–3742, 2005.
- Krichmar JL, Nasuto SJ, Scorcioni R, Washington SD, Ascoli GA.** Effects of dendritic morphology on CA3 pyramidal cell electrophysiology: a simulation study. *Brain Res* 941: 11–28, 2002.
- Lindenmayer A.** Mathematical models for cellular interaction in development. *J Theor Biol* 18: 280–315, 1968.
- Magee JC.** Dendritic Ih normalizes temporal summation in hippocampal CA1 neurons. *Nat Neurosci* 2: 848, 1999.
- Mainen ZF, Sejnowski TJ.** Influence of dendritic structure on firing pattern in model neocortical neurons. *Nature* 382: 363–366, 1996.
- Mitchell M.** *An Introduction to Genetic Algorithms*. Cambridge, MA: MIT Press, 2001.
- Moriarty DEM, R.** Hierarchical evolution of neural networks. *IEEE Conference on Evolutionary Computation, Anchorage, AK* IEEE, 1998.
- Moriarty DEM, R.** Forming neural networks through efficient and adaptive coevolution. *Evol Comput* 5: 373–399, 1997.
- Parameshwaran-Iyer S, Carr CE, Perney TM.** Localization of KCNC1 (Kv3.1) potassium channel subunits in the avian auditory nucleus magnocellularis and nucleus laminaris during development. *J Neurobiol* 55: 165–178, 2003.
- Rall W.** *The Theoretical Foundation of Dendritic Function (Selected Papers of Wilfrid Rall with Commentaries)*. Cambridge, MA: MIT Press, 1995.
- Samsonovich AV, Ascoli GA.** Statistical morphological analysis of hippocampal principal neurons indicates cell-specific repulsion of dendrites from their own cell. *J Neurosci Res* 71: 173–187, 2003.
- Samsonovich AV, Ascoli GA.** Statistical determinants of dendritic morphology in hippocampal pyramidal neurons: a hidden Markov model. *Hippocampus* 15: 166–183, 2005.
- Shoham S, O'Connor DH, Sarkisov DV, Wang SS.** Rapid neurotransmitter uncaging in spatially defined patterns. *Nat Methods* 2: 837–843, 2005.
- Vanier MC, Bower JM.** A comparative survey of automated parameter search methods for compartmental neural models. *J Comp Neurosci* 7: 149–171, 1999.
- VanRullen R, Guyonneau R, Thorpe SJ.** Spike times make sense. *Trends Neurosci* 28: 1–4, 2005.
- Vetter P, Roth A, Hausser M.** Propagation of action potentials in dendrites depends on dendritic morphology. *J Neurophysiol* 85: 926–937, 2001.