Evolutionary Adaptation of Nonlinear Dynamical Systems in Computational Neuroscience

Stefan Schneider
(stefan.schneider@neuroinformatik.ruhr-uni-bochum.de), Christian
Igel (christian.igel@neuroinformatik.ruhr-uni-bochum.de),
Christian Klaes, Hubert R. Dinse and Jan C. Wiemer*
Lehrstuhl für theoretische Biologie, Institut für Neuroinformatik, Ruhr-Universität Bochum, 44780 Bochum, Germany

Abstract. We propose evolutionary "analysis by synthesis" as a powerful tool in computational neuroscience. We present applications of evolution strategies to the adaptation of dynamical systems for brain modeling. First, we compare evolutionary and gradient-based optimization of dynamic neural fields on an artificial benchmark problem. Then we adjust a few-neuron model developed for explaining our recent findings in a neurobiological experiment, in which we studied the processing of temporal sequences of stimuli in the cortex.

Keywords: evolution strategy, neural fields, dynamical systems, computational neuroscience

1. INTRODUCTION

We propose evolutionary "analysis by synthesis" guided by neurobiological knowledge as a powerful tool in computational neuroscience. The challenge is to force artificial evolution to favor solutions that are reasonable from the biological point of view. Such solutions are only likely to evolve if as much neurobiological knowledge as possible is used in the design process. This can be achieved by providing sufficient experimental data to evaluate the evolved systems and by a deliberate choice of the basic system structure. Additional knowledge can be incorporated into the objective function and constraints that ensure biological plausibility.

The first step towards the automated design of biological models is parameter adaptation of predefined systems. In the following, we report on some recent applications of the covariance matrix adaptation evolution strategy (CMA-ES) for parameter optimization of different neurobiological models that describe population-coded information processing in the cortex, in particular in terms of dynamic neural fields [8]. These models have comparatively few, but meaningful parameters (time constants, coupling strengths, etc.). We believe that such kind of descriptions are particularly helpful to understand the principles of neural computation. Neural fields are a mean-field approach assuming that mainly mean anatomical and neurophysiological properties of neuron populations are relevant for the information processing.

In the next section, we briefly review the CMA-ES. Then the two main parts of the article follow, a methodological and a biological one: In section 3, we

[*] present address: German Cancer Research Center (DKFZ), 69120 Heidelberg, Germany

To appear in *Genetic Programming and Evolvable Machines*

consider the adaptation of dynamic neural fields. Evolutionary and gradient-based optimization are compared in a benchmark scenario. In section 4, we present an application to the adaption of a few-neuron model. The model is fitted to the data of our recent experimental findings regarding processing of temporal sequences of tactile stimuli in rat sensory cortex.

2. THE CMA-ES

For the adaptation of the real-valued parameters of dynamical systems we use the elaborated CMA (covariance matrix adaptation) evolution strategy (CMA-ES). A general introduction to evolution strategies can be found in [3], for details of the CMA-ES the reader is referred to [11, 12].

In the CMA-ES the individuals are m-dimensional real-valued vectors, where m is the problem dimension. The offspring are generated by (global intermediate) recombination and subsequent mutation. Here, mutation means adding an m-dimensional normally distributed random vector with zero mean. Deterministic (μ, λ)-selection is used, i.e., the best μ of the λ offspring form the next parent population.

The CMA-ES implements an efficient and reliable way for adapting the covariance matrix of the mutation distribution. Thereby, the algorithm becomes invariant under orthogonal transformations of the search space (except for the initialization) and can perform efficient optimization with small population sizes. The adaptation of the covariance matrix makes use of two important concepts: *derandomization* and *cumulation*. The former means that the adaptation of the covariance matrix is deterministic after variation and selection of the individuals: The mutation steps in the object parameter space that resulted in selected offspring are used directly for adapting the covariance matrix such that similar steps become more likely. *Cumulation* means that the path of the population over a number of past generations is taken into account for the adaptation of the mutation distribution in order to use the information from previous generations more efficiently.

3. ADAPTATION OF DYNAMIC NEURAL FIELDS

Dynamic neural fields (NFs) have been introduced as models of cortical information processing [24, 1] and there is a growing interest in using them for biological and psychological modeling as well as for technical applications (e.g., see [7, 10, 4, 22, 9]). However, the problem arises how to find suitable parameters of NFs for a given task.

In [14], we adjusted the parameters of a one-dimensional single-layered NF [1] using gradient-based and evolutionary optimization as well as a simple hybrid algorithm, which switches from evolutionary to gradient-based optimization when the error drops below a predefined threshold. It turned out that gradient-based optimization showed better *local* search performance and that the hybrid strategy combined positive aspects of both evolutionary and gradient-based methods (similar to the findings in [2]).

In the following, we extend our investigation to gradient-based and evolutionary optimization of two-layered fields. Computation of the gradient of NFs is comparatively costly, see section 3.2. Hence, the question arises whether the increase in the number of object parameters leads to qualitatively different results compared to [14], e.g., to better local search performance of the evolution strategy. This hypothesis is supported by theoretical considerations (e.g., in [21]) showing that for increasing problem dimension a *simple* evolution strategy outperforms *simple* gradient-descent (needing $m+1$ target function evaluations per optimization step, where m is the problem dimension) on *simple* artificial test functions.

In section 3.1, we introduce the two-layered NF. Then we discuss the calculation of the partial derivatives of the model. In section 3.3 we present our benchmark problem and in 3.4 the results of our empirical comparison.

3.1. Model

We study the two-layered NF introduced in [8, 16] for modeling early visual information processing in the primary visual cortex of the cat. It is a coupled system of integro-differential equations describing an inhibitory and an excitatory layer of neurons. For a discrete space of n neurons, their dynamics are governed by

$$\begin{aligned}\tau_u \dot{u}_x(t) &= -u_x(t) + \sum_{x'=1}^{n} w_s(x-x') s_{x'}(t) + h \\ &\quad + f_{\mathrm{sh}}[u_x(t)] \left[\sum_{x'=1}^{n} w_u(x-x')\, f_u[u_{x'}(t)] - v_x(t) \right] \qquad (1)\\ \tau_v \dot{v}_x(t) &= -v_x(t) + \sum_{x'=1}^{n} w_v(x-x')\, f_u[u_{x'}(t)] \ , \qquad (2)\end{aligned}$$

where $u_x(t)$ and $v_x(t)$ describe the average membrane potential at time t of a neuron located at site $x \in \{1,\dots,n\}$ of the inhibitory and excitatory layer, respectively. The external input into the field—the stimulus—is denoted by $s_x(t)$. The resting potential (the potential in the absence of any input) is determined by h. The (membrane) time constants are given by τ_u and τ_v. The strengths of the couplings are determined by $w_u(.) = w_{g_u,\sigma_u}(.)$, $w_v(.) = w_{g_v,\sigma_v}(.)$, and $w_s(.) = w_{g_s,\sigma_s}(.)$, which are bell-shaped functions with two parameters each:

$$w_{g,\sigma}(x) = \frac{g}{\sqrt{2\pi}\sigma} \exp\left(\frac{-x^2}{2\sigma^2}\right) \ . \qquad (3)$$

The firing rate function $f_u[.] = f_{\alpha_u,\beta_u,\vartheta_u}[.]$ and the shunting term $f_{\mathrm{sh}}[.] = f_{\alpha_{\mathrm{sh}},\beta_{\mathrm{sh}},\vartheta_{\mathrm{sh}}}[.]$ are scaled sigmoidal functions with three parameters each:

$$f_{\alpha,\beta,\vartheta}[u_x(t)] = \frac{\alpha}{1+\exp(\vartheta - \beta u_x(t))} \ . \qquad (4)$$

The partial derivatives of f_u and f_{sh} with respect to their arguments are denoted by f'_u and f'_{sh}, respectively.

Overall, we have $m = 15$ free parameters gathered in the vector

$$\boldsymbol{\theta} = (g_u, \sigma_u, g_v, \sigma_v, g_s, \sigma_s, \alpha_u, \beta_u, \vartheta_u, \alpha_{\mathrm{sh}}, \beta_{\mathrm{sh}}, \vartheta_{\mathrm{sh}}, \tau_u, \tau_v, h)' \ .$$

3.2. Gradient-based optimization

The goal of supervised optimization of NFs is to find parameters such that, given a stimulus pattern $s_x(t)$, the potential $u_x(t)$ (or the average activity $f[u_x(t)]$) fits data points $d_x(t)$, i.e., measured membrane potentials (or firing rates), and eventually satisfies neurophysiological constraints. We assume that the measurements describe a potential pattern of excitatory neurons over the time interval $[0,T]$. The fit to the data can be achieved by minimizing the sum-of-squares error

$$E = \int_0^T E(t)\mathrm{d}t \quad \text{with} \quad E(t) = \sum_{x=1}^{n} \mu_x(t)\left(u_x(t) - d_x(t)\right)^2 \ , \tag{5}$$

where the masking function $\mu_x(t)$ is one if a target value at time t and position x is available and zero otherwise [6].

In order to apply gradient-based optimization methods, we have to calculate

$$\frac{\mathrm{d}E}{\mathrm{d}\theta_i} = \int\limits_{t=0}^{T} \sum_{x=1}^{n} \mu_x(t)\, \frac{\partial E(t)}{\partial u_x(t)} \frac{\mathrm{d}u_x(t)}{\mathrm{d}\theta_i} \mathrm{d}t \ , \tag{6}$$

for each parameter θ_i, $i = 1, \dots, m$.

The derivatives $\mathrm{d}u_x(t)/\mathrm{d}\theta_i$ can be determined by solving the so called variation system "forward in time" [6, 18]. As pointed out in [14], methods like computing the gradient "backward in time" do not reduce the complexity in case of NFs. The variation system that estimates the effect of parameter changes on the NF dynamics is derived by differentiating the NF equations (1) and (2) with respect to the parameters, cf. [13]. This means, we solve the system of differential equations describing $\mathrm{d}u_x(t)/\mathrm{d}\theta_i$ for $x = 1, \dots, n$ and $i = 1, .., m$. This method is also known as *real time recurrent learning*, *forward propagation*, *variation method*, or *forward propagation*. For example, for a parameter θ_i ($i \in \{1, 2\}$) of the interaction kernel w_u we have

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\mathrm{d}u_x(t)}{\mathrm{d}\theta}\right) =& \frac{1}{\tau_u}\Bigg(-\frac{\mathrm{d}u_x(t)}{\mathrm{d}\theta} \\ & + f'_{\mathrm{sh}}[u_x(t)]\frac{\mathrm{d}u_x(t)}{\mathrm{d}\theta}\left[\sum_{x'=1}^{n} w_u(x-x') f_u[u_{x'}(t)] - v_x(t)\right] \\ & + f_{\mathrm{sh}}[u_x(t)]\left[\sum_{x'=1}^{n}\left[\frac{\partial w_u(x-x')}{\partial\theta} f_u[u_{x'}(t)]\right.\right. \\ & \left.\left. + w_u(x-x') f'_u\left[u_{x'}(t)\right]\frac{\mathrm{d}u_{x'}(t)}{\mathrm{d}\theta}\right] - \frac{\mathrm{d}v_x(t)}{\mathrm{d}\theta}\right]\Bigg)\end{aligned} \tag{7}$$

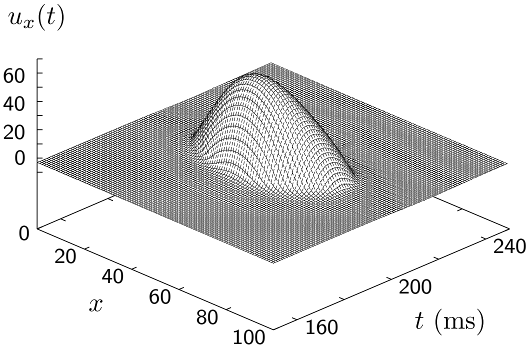

Figure 1. Artificial target data for the adaptation of the two-layered neural field. The spatio-temporal pattern was generated by a "teacher field" defined by Eqs. (1) and (2).

and

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\mathrm{d}v_x(t)}{\mathrm{d}\theta}\right) = \frac{1}{\tau_v}\left(-\frac{\mathrm{d}v_x(t)}{\mathrm{d}\theta} + \sum_{x'=1}^{n} w_v(x-x')f_u'[u_{x'}(t)]\frac{\mathrm{d}u_{x'}(t)}{\mathrm{d}\theta}\right) . \tag{8}$$

The initial conditions are given by $\mathrm{d}u_x(0)/\mathrm{d}\theta_i = 0$ for $x = 1, \dots, n$ and $i = 1, \dots, m$.

Equations like (7) and (8) have to be solved for each parameter for all positions. In this article, all dynamical systems are solved using the forward Euler method. Thus, calculating the gradient requires $\mathcal{O}(mn^2)$ computations per simulated (discrete / Euler) time-step.

Two problems become obvious: First, the systems has to be, at least piecewise, differentiable w.r.t. the m parameters. For example, if non-differentiable transfer functions are used in the model, gradient-based optimization becomes problematic. Second, for more complex dynamical systems, computing the partial derivatives quickly becomes tedious.

3.3. Experiments

The task was to reproduce the artificial spatio-temporal pattern shown in Fig. 1, which was produced by a "teacher" NF. The number of neurons of the field was $n = 101$ and the parameters were $h = -3$, $\tau_u = \tau_v = 10$, $\sigma_u = 15$, $\sigma_v = 25$, $g_u = 195$, $g_v = 250$, $\beta_u = \beta_{\mathrm{sh}} = \alpha_u = \alpha_{\mathrm{sh}} = 1$, $\vartheta_u = \vartheta_{\mathrm{sh}} = 0$, $\sigma_S = 10$, and $g_S = 60$. The time interval was $[0, \dots, 400]$. The stimulus was given by $s_{50}(t) = 1$ for $175 \leq t < 180$ and $s_x(t) = 0$ otherwise. Prior to the 400 time steps the field was iterated until $\dot{u}_x(t) < 10^{-6}$ for all positions x.

We conducted two sets of experiments. First, the initial parameters were randomly chosen from the intervals $h \in [-5, 0]$, $\tau_u, \tau_v \in [1, 10]$, $\sigma_u, \sigma_v, \sigma_S \in [3, 50]$, $g_u, g_v, g_S \in [10, 300]$, $\beta_u, \beta_{\mathrm{sh}} \in [0.5, 10]$, $\alpha_u, \alpha_{\mathrm{sh}} \in [0.5, 2]$, and $\vartheta, \vartheta_{\mathrm{sh}} \in [-0.1, 0.1]$. Second, the models were initialized ±10% around the optimal values. The latter scenario corresponds to fine-tuning an already (e.g., manually) adjusted model.

Before learning and fitness evaluation, respectively, the neurons were initialized to $u_x(0) = v_x(0) = 0$. The first 150 time steps were not considered for fitness and gradient calculation, i.e., we set $\mu_x(t) = 0$ for $t < 150$ and $\mu_x(t) = 1$ otherwise. This is necessary as the models need some time to relax when starting from the homogeneous initialization, see [13] for an analysis.

In the CMA-ES, we set $\mu = 4$ and $\lambda = 10$ and started with a global step size of $\sigma^{(1)} = 0.5$ (see [12]). For gradient-based optimization, we used the Broyden-Fletcher-Goldfarb-Shanno (BFGS) algorithm, a quasi-Newton method, cf. [20]. For the line search, a first-order method was used. Note that both the CMA-ES and the BFGS incrementally build up a matrix for second-order information about the (local) error surface (the covariance matrix of the mutation distribution and the approximated inverse Hessian, respectively).

We performed 200 independent trials for each scenario, where the initial centers of mass of the parent populations and the starting points of the gradient-based optimization were the same. We compared the success rates vs. the computational costs of the algorithms. A trial was regarded as successful if the sum-of-squares error dropped below a threshold of either 10^{-3} or 10. The computational costs were approximated in the following way. One evaluation of a NF model, during a line search or for fitness evaluation, corresponded to one computational cost unit. The costs of computing the gradient were set to $m + 1$, see previous section and [14].

3.4. Results

When the NFs were initialized from the larger intervals, the lower threshold was not reached by any evolutionary trial and only by a single gradient-based one. In case of initialization close to the optimum, the BFGS passed the lower error threshold considerably more often. This means, the local search performance of the CMA-ES compared to the BFGS is *not* better than in the case of fewer objective parameters (at least for our deliberate choice of the test problem and the parameters of the algorithms).

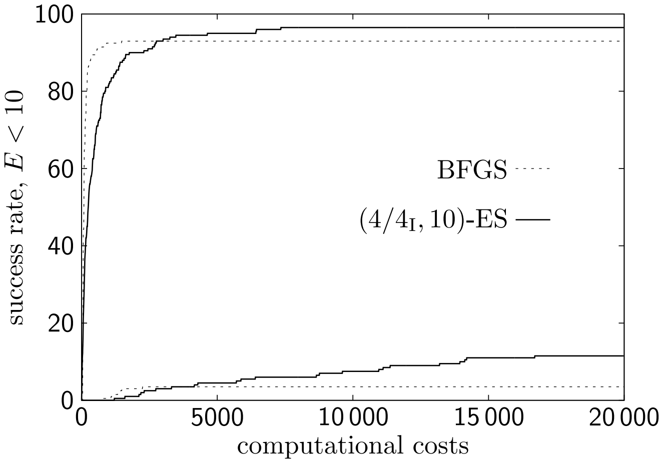

Figure 2. Success rate vs. computational costs. The upper two lines correspond to initialization close to the optimum, the lower ones to starting farther away.

The success rates for the larger threshold are shown in Fig. 2. Here the results are similar to the ones presented for the single-layered NFs in [14]. The CMA-ES is slower, but has a higher success rate. The difference is not statistically significant for the trials started close to the optimum (χ^2-test after computational costs of 20 000, $p > 0.1$), but for the trials starting farther away from the optimum (χ^2-test after computational costs of 20 000, $p < 0.005$). The BFGS gets stuck in local minima more often than the evolution strategy (however, it also converges faster to these local optima).

4. ADAPTION OF A FEW-NEURON MODEL

In this section we present an application of the previously discussed CMA-ES to a real-world computational neuroscience task. In order to reveal mechanisms of cortical temporal stimulus processing, we carried out a neurobiological experiment, developed a model, and fitted the model to the experimental data by applying the CMA evolution strategy.

We employed the CMA-ES for adjusting model parameters for the following reasons: First, adjusting the parameters by hand, as it is common in computational neuroscience, is usually time-consuming and may not lead to good fits, i.e., to models with small fitting errors. Second, we did not want to restrict a priori the search for parameter values to regions of desirable, i.e., biologically plausible parameter values. This helps judging the model. Obviously, this more global search is impossible by hand. Third, we wanted to test if the CMA-ES could serve as a a convenient tool for computational neuroscientists, i.e., can be used in an out-of-the-box fashion.

4.1. Neurobiological experiment

We examined how temporal sequences of sensory stimuli are processed by the sensory cortex. We applied trains of three tactile stimuli separated by identical interstimulus intervals (ISIs) to the hind paw of rats, and recorded the neural stimulus-response activity in the rats' primary somatosensory cortex (S1) [25]. The following ISIs were used: 22 ms, 30 ms, 45 ms, 60 ms, 80 ms, 100 ms, 120 ms, 180 ms, and 240 ms. For each ISI, a mean firing rate (post-stimulus-time histogram) was calculated by averaging the neural response data of all 64 stimulus trains for this specific ISI. The maxima of the three response peaks in the mean firing rate were read out as the stimulus responses $a_{1,\mathrm{exp}}$, $a_{2,\mathrm{exp}}$, and $a_{3,\mathrm{exp}}$. For this study, only the relative response strengths $a_{2,\mathrm{exp}}/a_{1,\mathrm{exp}}$, $a_{3,\mathrm{exp}}/a_{1,\mathrm{exp}}$, and $a_{3,\mathrm{exp}}/a_{2,\mathrm{exp}}$ were considered.

4.2. Model

In order to understand the biological neural mechanisms of temporal cortical stimulus processing, we modeled processing stages of the sensory signal in the brain. These include the thalamus and the primary somatosensory cortex which receives the signal from the thalamus [17]. In thalamus and cortex, processing of the sensory signal is done by populations of functionally similar neurons. Our approach follows a basic notion of computational neuroscience: each biological neuron population is represented by a single model neuron [23]. Therefore, our model consists of one excitatory thalamic neuron and two cortical neurons, one excitatory and one inhibitory, cf. Fig. 3. The model constitutes a limit case of a neural field, which is not spatially extended but restricted to a single location [23, 9]. We also modeled depletion of synaptic transmitter. Thereby, we assume that the experimentally observed effects were caused by cortical feedback inhibition and synaptic depletion.

The model is described by the following system of coupled ordinary differential equations:

$$\tau_e \frac{\mathrm{d}x_e(t)}{\mathrm{d}t} = -x_e(t) + f[p\, w_{\mathrm{ff}}\, x_m(t) - w_{\mathrm{int}}\, x_i(t)] \tag{9}$$

$$\tau_i \frac{\mathrm{d}x_i(t)}{\mathrm{d}t} = -x_i(t) \; + f[w_{\mathrm{int}}\, x_e(t)] \tag{10}$$

$$\tau_e \frac{\mathrm{d}x_m(t)}{\mathrm{d}t} = -x_m(t) + F_{m,\mathrm{ISI}}(t) \tag{11}$$

$$\frac{\mathrm{d}p(t)}{\mathrm{d}t} = -c_t\, p(t)\, F_{m,\mathrm{ISI}}(t) + \frac{1-p(t)}{\tau_d} \; . \tag{12}$$

The variables x_e, x_i, and x_m are the synaptic drives of excitatory, inhibitory, and thalamic neuron [19]. τ_e and τ_i denote time constants for excitation and inhibition. We set $\tau_e = 10\,\mathrm{ms}$, which is neurobiologically plausible, and τ_i was adjusted by the CMA-ES. w_{int} is the coupling strength between the excitatory and the inhibitory neuron determining their interaction. The value of w_{int} was adjusted by the CMA-ES. $f[.]$ is a nonlinear firing rate function. We chose the linear-threshold function $f[x] = \max(0, x)$. The parameter w_{ff} denotes the feed-forward coupling strength between the thalamic neuron and the excitatory cortical neuron, and was arbitrarily set to $w_{\mathrm{ff}} = 10$. $F_{m,\mathrm{ISI}}$ is the firing rate of the thalamic neuron and models the stimulus for the different ISIs. $F_{m,\mathrm{ISI}}(t)$ was chosen to be a stepwise function for each ISI: $F_{m,\mathrm{ISI}}(t) = 1$ for $t \in [0, 15] \cup [\mathrm{ISI}, \mathrm{ISI}+15] \cup [2\cdot\mathrm{ISI}, 2\cdot\mathrm{ISI}+15]$ ms and $F_{m,\mathrm{ISI}}(t) = 0$ otherwise. This corresponds to the time course of the application of the physical, tactile stimulus to the animals' hind paw. Synaptic depletion is described by the fraction of available transmitter, p, the depletion factor c_t, and the transmitter recovery time constant τ_d [5]. c_t and τ_d were adjusted by the CMA-ES.

For each ISI (stimulus $F_{m,\mathrm{ISI}}$) the differential equations are integrated and the maxima of the three response peaks in the excitatory firing rate $f[x_e(t)]$ are read out as the stimulus responses $a_{1,\mathrm{mod}}$, $a_{2,\mathrm{mod}}$, and $a_{2,\mathrm{mod}}$. These model responses are compared with the corresponding experimental responses in the form of relative response strengths $a_{2,\mathrm{mod}}/a_{1,\mathrm{mod}}$, $a_{3,\mathrm{mod}}/a_{1,\mathrm{mod}}$, and $a_{3,\mathrm{mod}}/a_{2,\mathrm{mod}}$.

4.3. Evolutionary optimization

We applied the CMA-ES for determining the 4 model parameters τ_i, w_{int}, τ_d, and c_t for feedback inhibition and synaptic depletion, which we hypothesized to be mainly responsible for the experimentally observed effects. The goal of

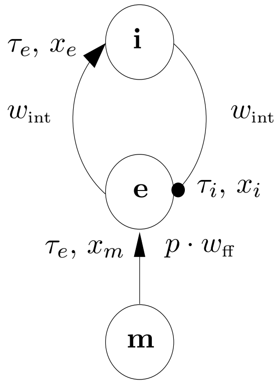

Figure 3. Structure of the model with thalamic neuron **m**, excitatory cortical neuron **e** and inhibitory cortical neuron **i**. Excitatory and inhibitory synapses are marked by arrows and circles, respectively. Depletion of synaptic transmitter occurs at the synapse between thalamic and excitatory cortical neuron. The strength of this synapse is the product $p \cdot w_{\mathrm{ff}}$ of a fixed weight w_{ff} and the fraction p of available transmitter.

our optimization was to find values for the four parameters such that the model data $a_{2,\mathrm{mod}}/a_{1,\mathrm{mod}}$ and $a_{3,\mathrm{mod}}/a_{1,\mathrm{mod}}$ fit the corresponding experimental data over all 9 ISIs[1] and also satisfies three constraints:

1. $a_{3,\mathrm{mod}} > a_{2,\mathrm{mod}}$ for ISI $= 30\,\mathrm{ms}$
2. $a_{2,\mathrm{mod}} > a_{3,\mathrm{mod}}$ for ISI $\geq 100\,\mathrm{ms}$
3. $a_{2,\mathrm{mod}}, a_{3,\mathrm{mod}} \neq 0$ for all ISIs .

These constraints stress features of the experimental data we considered to be of particular significance.

The error function E to be minimized consists of the mean-squared error E_0 evaluating the deviation of the model data from the experimental data, and three additive penalty terms E_1, E_2, and E_3 measuring violation of the constraints:

$$E = E_0 + E_1 + E_2 + E_3 \tag{13}$$

$$E_0 = \sqrt{\frac{1}{9}\sum_{\mathrm{ISI}}\left(\frac{a_{2,\mathrm{mod}}}{a_{1,\mathrm{mod}}} - \frac{a_{2,\mathrm{exp}}}{a_{1,\mathrm{exp}}}\right)^2} + \sqrt{\frac{1}{9}\sum_{\mathrm{ISI}}\left(\frac{a_{3,\mathrm{mod}}}{a_{1,\mathrm{mod}}} - \frac{a_{3,\mathrm{exp}}}{a_{1,\mathrm{exp}}}\right)^2} \tag{14}$$

$$E_1 = -10\left(\tfrac{a_{3,\mathrm{mod}}}{a_{2,\ \mathrm{mod}}} - 1\right) + 100 \quad : \quad \text{if } a_{3,\mathrm{mod}} < a_{2,\mathrm{mod}} \text{ for ISI} = 30\,\mathrm{ms} \tag{15}$$

$$E_2 = \begin{cases} 100 & : \text{ if } a_{2,\mathrm{mod}} < a_{3,\mathrm{mod}} \text{ for ISI} \geq 100\,\mathrm{ms} \\ 0 & : \text{ else} \end{cases} \tag{16}$$

$$E_3 = \begin{cases} 100 & : \text{ if } a_{2,\mathrm{mod}} \text{ or } a_{3,\mathrm{mod}} = 0 \text{ for at least one ISI} \\ 0 & : \text{ else} \end{cases} \quad . \tag{17}$$

The values for the penalty terms were chosen to be much larger than the average mean-squared error forcing the optimization to stick to the constraints with high priority. A similar procedure has proved to be successful in the case of fitting a neural field model to neurobiological data [15]. In test series of optimization trials we found that the simple penalty terms E_2 and E_3 are sufficient for satisfying constraints 2 and 3. However, we also found that an identical treatment of constraint 1 is insufficient: the majority of solutions violated the constraint. We concluded that there are large connected regions in the parameter space containing models that violate constraint 1. Adding a constant to the error function only "lifts" the error landscape on these regions. Due to the extendedness of the regions a lot of optimization trials start within these, and since there is no error gradient concerning the violation of constraint 1, the trials also get stuck in local optima within the regions. Accordingly, the linear penalty term (15) had to be added measuring how strongly the constraint is violated.

We carried out a series of twelve CMA-ES trials minimizing the error function (13). The parameter intervals for the search were restricted to reasonable values: $c_t \in [0, 0.1]$, $w_{\text{int}} \in [0, 10]$, and $\tau_d, \tau_i \in [10, 1000]$ ms. Parameter values that are larger or smaller than the boundaries of these intervals do not make sense because the corresponding models exhibit total response suppression, or time scales of dynamics much longer or shorter than those given by the experimental data. The other parameters were set to plausible values beforehand and kept fixed, cf. subsection 4.2. Numeric integration of the differential equations was done by means of the Euler method with step size 1 ms and initial conditions $x_e(0) = x_i(0) = x_m(0) = 0$ and $p(0) = 1$. In the CMA-ES we set $\mu = 1$ and $\lambda = 4$. The initial parameter values for τ_i, w_{int}, τ_d, and c_t were randomly chosen equally distributed from the intervals described above. We determined the fitness of a parameter set by integrating the differential equation system for every ISI, cf. subsection 4.2, and evaluating the model data according to the error function.

4.4. Results

The series of CMA-ES trials yielded several local optima of different quality. This indicates the difficulty of the optimization task. The computational costs of the single trials were low. In each trial, the CMA-ES settled on a local minimum within at most 1000 generations.

We adopted the solution with the smallest error, $E = 0.16$, as the solution to our model selection problem. It fits the experimental data very well, cf. Fig. 4. Furthermore, its parameter values are biologically plausible. The other four valid solutions, i.e., solutions satisfying the constraints 1 to 3, fit the experimental data clearly worse ($E \geq 0.53$). Additionally, their parameter values are less plausible than those of the best solution. These relations between the quality of the solution and the biological plausibility of the parameter values support our modeling approach.

Interestingly, seven out of the twelve CMA-ES trials violated the first constraint, although we used the penalty term E_1 providing a strong error gradient. Exploring the reasons for this behavior is one of the aims of our future research.

For our quite typical application, the CMA-ES proved to be a convenient tool. Setting rough boundaries to the parameter search and introducing simple

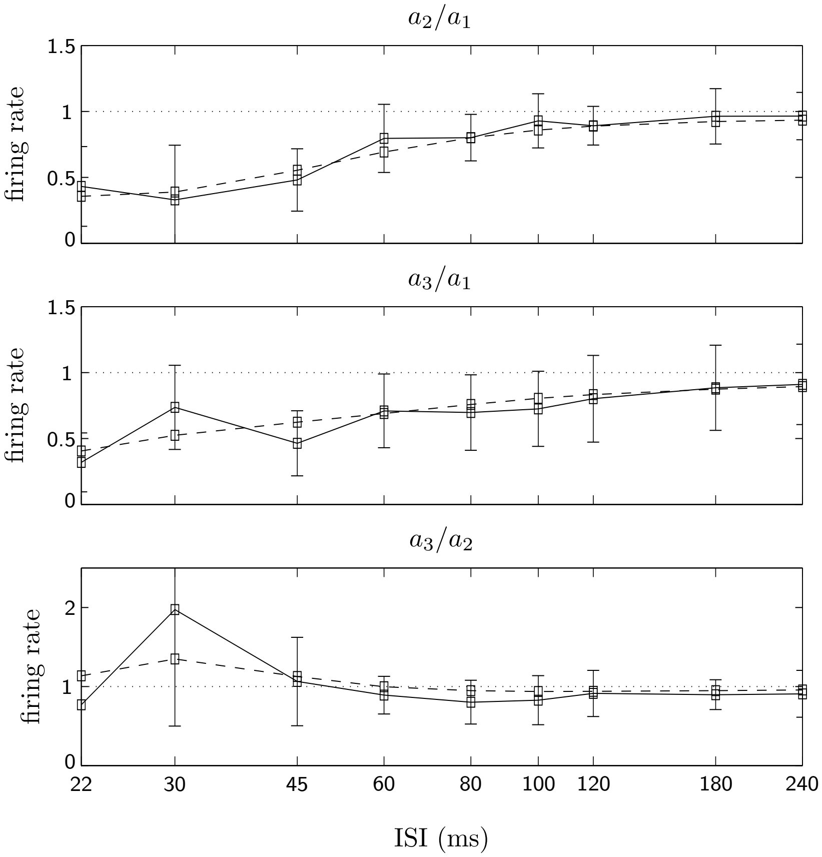

Figure 4. Experimental data (solid line) and model data of the best solution (dashed line). a_1, a_2 and a_3 are the neural response strengths to the 1st, 2nd, and 3rd stimulus.

penalty terms for the constraint was sufficient to find an adequate solution within reasonable time.

5. DISCUSSION AND CONCLUSIONS

Evolutionary algorithms have proved to be beneficial for the optimization of models in computational neuroscience. They can be applied even if the quality function is non-differentiable, discontinuous, and noisy. Further, they allow for the incorporation of expert knowledge in several ways. In the case where the gradient can be derived with reasonable effort, it it should be exploited, e.g., in a hybrid algorithm as in [14], because of the better local search performance of gradient-based methods. However, even if the fitness function is differentiable, there are reasons to prefer evolutionary optimization, as the derivatives may be very complicated and tedious to compute. Evolution strategies can be used nearly out-of-the-box and the increase in performance gained by a specialized (e.g., gradient-based) method can often be neglected compared to the time needed to get the specialized algorithm running.

These properties of evolution strategies make them suited for practical applications in neuroscience. The process of fitting a model to experimental data is strongly accelerated. By enabling a “global” search in parameter space, the application of evolution strategies allow for a more thorough understanding of the model. The evaluation of the model hypotheses, i.e., how well they explain the experimental findings, is improved. The development of a basic model structure and evolutionary optimization can be an iterative and interleaving process, where the (unsatisfying) outcome of the optimization may lead to a refinement of the model.[2]

Regardless of the used optimization method, parameter adaptation of nonlinear dynamical systems has turned out to be a difficult optimization task, where—at least in our examples—the optimization algorithms tend to get trapped in local minima; bifurcations of the dynamics exacerbate the search process. These difficulties can partly be overcome by incorporating additional expert knowledge and multi-start strategies, as done in our experiments.

Notes

[1] The quantity $a_{3,\mathrm{mod}}/a_{2,\mathrm{mod}}$ was not included in the data comparison because it is derived from the other two quantities.

[2] An example of model refinement after evolutionary optimization is the parameterization of the kernel function in [15]: The evolved models showed the need for steeper filter characteristics; after adding degrees of freedom to the kernel function, the evolutionary adaptation led to quantitatively better fits.

References

1. Amari, S.: 1977, ‘Dynamics of pattern formation in lateral-inhibition type neural fields’. *Biological Cybernetics* **27**, 127–132.
2. Arai, K., S. Das, E. L. Keller, and E. Aiyoshi: 1999, ‘A distributed model of the saccade system: Simulations of temporally perturbed saccades using position and velocity feedback’. *Neural Networks* **12**, 1359–1375.
3. Beyer, H.-G. and H.-P. Schwefel: 2002, ‘Evolution strategies – A comprehensive introduction’. *Natural Computing* **1**(1), 3–52.
4. Bicho, E., P. Mallet, and G. Schöner: 2000, ‘Target representation on an autonomous vehicle with low level sensors’. *The International Journal of Robotics Research* **19**(5), 424–447.
5. Dayan, P. and L. F. Abbott: 2001, *Theoretical euroscience: Computational and mathematical modeling of neural systems*. MIT Press.
6. Doya, K.: 1995, ‘Recurrent networks: Supervised learning’. In: M. A. Arbib (ed.): *The Handbook of Brain Theory and Neural Networks*. MIT Press, pp. 796–800.
7. Engels, C. and G. Schöner: 1995, ‘Dynamic fields endow behavior-based robots with representation’. *Robotics and Autonomous Systems* **14**, 55–77.
8. Erlhagen, W., A. Bastian, D. Jancke, A. Riehle, and G. Schöner: 1999, ‘The distribution of neural population activation as a tool to study interaction and integration in cortical representations’. *Journal of Neuroscience Methods* **94**(1), 53–66.
9. Erlhagen, W. and G. Schöner: 2002, ‘Dynamic field theory of movement preparation’. *Psychological Review* **109**(3), 545–572.
10. Giese, M. A.: 1999, *Dynamic Neural Field Theory for Motion Perception*. Kluwer Academic Publishers.

11. Hansen, N. and A. Ostermeier: 1997, 'Convergence properties of evolution strategies with the derandomized covariance matrix adaptation: The $(\mu/\mu, \lambda)$-CMA-ES'. In: *5th European Congress on Intelligent Techniques and Soft Computing (EUFIT'97)*. Aachen, Germany, pp. 650–654.
12. Hansen, N. and A. Ostermeier: 2001, 'Completely derandomized self-adaptation in evolution strategies'. *Evolutionary Computation* **9**(2), 159–195.
13. Igel, C.: 2003, *Beiträge zum Entwurf neuronaler Systeme*. Aachen, Germany: Shaker Verlag.
14. Igel, C., W. Erlhagen, and D. Jancke: 2001, 'Optimization of dynamic neural fields'. *Neurocomputing* **36**(1-4), 225–233.
15. Igel, C., W. von Seelen, W. Erlhagen, and D. Jancke: 2002, 'Evolving field models for inhibition effects in early vision'. *Neurocomputing* **44-46**(C), 467–472.
16. Jancke, D., W. Erlhagen, H. R. Dinse, A. C. Akhavan, M. Giese, A. Steinhage, and G. Schöner: 1999, 'Parametric population representation of retinal location: Neuronal interaction dynamics in cat primary visual cortex'. *Journal of Neuroscience* **19**(20), 9016–9028.
17. Kandel, E. R. and J. H. Schwartz: 1985, *Principles of Neural Science*. Elsevier.
18. Pearlmutter, B. A.: 1995, 'Gradient calculations for dynamic recurrent neural networks: A Survey'. *IEEE Transactions on Neural Networks* **6**(5), 1212–1228.
19. Pinto, D. J., J. C. Brumberg, D. J. Simons, and G. B. Ermentrout: 1996, 'A quantitative population model of whisker barrels: Re-examining the Wilson-Cowan equations'. *Journal of Computational Neuroscience* **3**, 247–264.
20. Press, W., S. Teukolsky, W. Vetterling, and B. Flannery: 1994, *Numerical Recipes in C*. Cambridge University Press, 2. edition.
21. Rechenberg, I.: 1994, *Evolutionsstrategie '94*, Werkstatt Bionik und Evolutionstechnik. Stuttgart: Frommann-Holzboog.
22. Thelen, E., G. Schöner, C. Scheier, and L. Smith: 2001, 'The dynamics of embodiment: A field theory of infant perserverative reaching'. *Brain and Behavioral Sciences* **24**, 1–33.
23. Wilson, H. R. and J. D. Cowan: 1972, 'Excitatory and inhibitory interactions in localized populations of model neurons'. *Kybernetik* **12**, 1–24.
24. Wilson, H. R. and J. D. Cowan: 1973, 'A mathematical theory of the functional dynamics of cortical and thalamic nervous tissue'. *Kybernetik* **13**, 55–80.
25. Zepka, R., J. Wiemer, S. Schneider, S. Cords, and H. R. Dinse: 2001, 'Plasticity of short-term plasticity: Cortical neuron responses to tactile stimuli of variable ISI recorded in somatosensory cortex of adult rats before and after intracortical microstimulation (ICMS)'. In: *Society for Neuroscience Abstracts*, Vol. 27, 49.12.