A unifying framework for neuronal assembly dynamics

Julian Eggert, Jan van Hemmen

2000

Preprint:

A Unifying Framework for Neuronal Assembly Dynamics

J. Eggert and J.L. van Hemmen
Physik Department, Technische Universität München,
D-85747 Garching bei München, Germany
(October 12, 2000)

Starting from single, spiking neurons, we derive a system of coupled differential equations for the description of the dynamics of pools of extensively many equivalent neurons. Contrary to previous work, the derivation is exact and takes into account microscopic properties of single neurons, such as axonal delays and refractory behavior. Simulations show a good quantitative agreement with microscopically modeled pools of spiking neurons. The agreement holds both in the quasistationary and in the non-stationary dynamical regime, including fast transients and oscillations. The model is compared with other pool models based on differential equations. It turns out that models of the graded-response category can be understood as a first-order approximation of our pool dynamics. Furthermore, the present formalism gives rise to a system of equations that can be reduced straightforwardly so as to gain a description of the pool dynamics to any desired order of approximation. Finally, we present a novel stability criterion that is suitable for handling pools of neurons. Due to its exact derivation from single-neuron dynamics, the present model opens new simulation possibilities for studies that rely upon biologically realistic large-scale networks composed of assemblies of spiking neurons.

I. INTRODUCTION

What kind of mathematical models should be chosen to study and simulate large, biologically realistic neural networks? In the computational-neuroscience literature, one can find a growing number of models that describe neurons at the single-cell level [11,20] as well as many models that describe the joint activity of groups of equivalent neurons [16,22,23]. Between the two modeling levels, only in a few cases [10,19,23] has a connection been made. To bridge this gap between the microscopic and the assembly level, we derive here a model for the activity dynamics of a pool of equivalent neurons starting from a single-cell model. Contrary to previous work on the subject, no a-priory time averaging is necessary, making the derivation concise and systematic.

The model is motivated by the experimental observation that cortical neurons of the same type that are near to each other tend to receive similar inputs. In experiments one often finds that neurons of the same type that are close to each other are activated simultaneously, or in a correlated fashion. In cortical networks, this may be due to reciprocal connections and common convergent input. In modeling studies it therefore seems sensible to consider all neurons of the same type in a small cortical volume as a computational unit of a neuronal network. We will call this computational unit a neuronal "pool" or "assembly". All pool neurons have to be equivalent in the sense that they have the same input/output connection characteristics and, additionally, the same dynamics parameters. This is explained in detail in Figure 1. All neurons that constitute a pool feel a common synaptic input field, but each neuron evolves according to its own internal dynamics.

How should we build a model based on these neuronal assemblies? We can start at the microscopic level and use single spiking neurons to compose the pools of a network. For large-scale simulations, the single-cell model has to be numerically efficient and easy to implement. Single neuron models that fulfill these requirements normally neglect the spatial structure of the dendritic tree and focus on the spike generation process. Examples are the spike–response model (see e.g. [11]) and the integrate–and–fire type models (see e.g. [20]), which constitute a special case of the spike–response model. Typically, a neuron has an internal state variable and a spike or action potential is released when the state variable reaches some threshold from below. After releasing the spike, the state variable or the threshold is temporarily modified to account for refractory effects.

Alternatively, one could start at a macroscopic level and use models that describe directly the behavior of some macroscopic variables of a neuronal pool. The most prominent models of this category are of the assembly-averaged graded-response type. Models of this type normally describe dynamics of the assembly activity through relatively simple differential equations.

In this paper we will follow a third approach. We start with single spiking neurons and take advantage of the assembly characteristics to derive a differential equation model for the dynamics of the pool activity. In this way,

the model focuses on the macroscopic parameters of cell assemblies but retains the quantitative behavior and the microscopic parameters of the neuronal model.

With the pool model presented in this paper, large-scale simulations with networks that are composed of pools of equivalent neurons become possible. It also allows for the modeling of complex spatio-temporal activity dynamics that is thought to rely upon the properties of spiking neurons. Coherent activity oscillations found in the visual cortex and other areas of the brain [2–4,13,14] constitute a well-known example. Another advantage of the presented model is the possibility to compare the microscopic parameters that are retained by the derivation from single-neuron dynamics with data from neurophysiological measurements. In addition, the differential equation form of our model allows for a comparison with other, more heuristically based pool models that are used throughout the neuroscience community.

In the next sections, we proceed as follows. First, we sketch the essentials of some of the commonly encountered assembly-averaged graded-response models used for the simulation of pool dynamics. We then introduce the single-neuron dynamics that constitutes the basis of our pool model. In sections III, IV and V, we present the derivation of the differential equation pool model. Consequences for the application of the model follow in section VI. In sections VII we compare the model with graded-response models and pools modeled by using single spiking neurons of the spike–response and the integrate–and–fire type. In section VIII we show that the model has the same stability and locking characteristics as pools of spiking neurons. In section IX, we demonstrate how finite-size effects can be included into simulations with our pool model. Finally, a summary gives a short overview of the model.

II. GRADED-RESPONSE MODELS

In this section we discuss some standard and enhanced graded-response models that are used throughout the neuroscience community. We indicate some flaws that are inherent to them. In later sections they will be compared with our pool model.

A. Standard Graded-Response Models

In the standard neural network literature, the simplest neuronal models use gain functions g to express the dependence of the "firing rate" or activity A_i of some neuronal entity i, with $1 \leq i \leq N$, upon its synaptic input field h_i, see e.g. [16],

$$A_i = \mathrm{g}[h_i(\{A_j\})] \; . \tag{1}$$

The synaptic input field depends on the set of present and past activities $\{A_j\}$ of all other neuronal entities j, $1 \leq j \leq N$, that contribute to the input of i. (In the rest of this section, we will not specify $h_i(t)$ any further.) A solution of the system (1) for all A_i defines a stationary state of the network.

In the assembly-averaged graded-response interpretation, the macroscopic variable $A_i(t)$ designates the pool-averaged spike density at time t, that is, $A_i(t)\Delta t$ is the total number of spikes released by neurons of the pool i during the interval $(t, t + \Delta t]$. We will use this interpretation throughout the rest of the paper.

Usually, a dynamics is introduced by choosing equations that have as fixpoints the same stationary solutions as the system (1). An easy way of achieving this is by adding an exponential relaxation term. In the time-discrete case this returns

$$A_i(t + \Delta t) = -A_i(t) + \mathrm{g}[h_i(\{A_j\})] \tag{2}$$

where Δt is the discretization interval, and in the continuous case one obtains

$$\tau \frac{\mathrm{d}}{\mathrm{d}t} A_i(t) = -A_i(t) + \mathrm{g}[h_i(\{A_j\})] \tag{3}$$

with some suitably chosen relaxation time constant τ.

A modified assembly-averaged graded-response model has been introduced by Wilson and Cowan [23]. They derived a differential equation model for neurons with absolute refractory period of length γ^{abs}, using a "time-coarse-graining" averaging method. Their final result reads

$$\tau \frac{\mathrm{d}}{\mathrm{d}t} A_i(t) = -A_i(t) + \mathrm{g}[h_i(\{A_j\})]\,[1 - \gamma^{\mathrm{abs}} A_i(t)] \; . \tag{4}$$

Compared with (3), this equation has a slightly modified dynamics near to the saturating activity $1/\gamma^{\rm abs}$. Both (3) and (4) will be encountered again in later sections when we discuss the connection of our pool model to graded-response models.

The equations (2) or (3), are, by construction, only suited to describe activities near to a *stationary state of the whole network*. Similarly, in equation (4), time averaging generates a dynamics that neglects fast, transient, behavior. In many cases, however, the above models are used to generate oscillatory pool activities. For example, it has been postulated that two reciprocally coupled pools, one composed of excitatory and the other of inhibitory neurons, could constitute a kind of processing unit capable of generating oscillations. Using equation (3) and designating the activity of the excitatory pool by $E_i(t)$ and that of the inhibitory pool by $I_i(t)$, we get

$$\begin{aligned} \tau^E \frac{\mathrm{d}}{\mathrm{d}t} E_i(t) &= -E_i(t) + \mathrm{g}^E[h_i^E(\{E_j\},\{I_j\})] \ , \\ \tau^I \frac{\mathrm{d}}{\mathrm{d}t} I_i(t) &= -I_i(t) + \mathrm{g}^I[h_i^I(\{E_j\},\{I_j\})] \ . \end{aligned} \tag{5}$$

It is plain that such a model can show oscillatory behavior, if the time constants τ^E, τ^I and the input fields h_i^E, h_i^I are suitably chosen; for more details we refer to e.g. [22,23].

After the derivation of our exact pool dynamics, we will return to the models presented in this section. We will compare the model of this paper with the assembly-averaged graded-response models. We will also show how graded-response models can be incorporated into a broader framework of pool activity dynamics which allows for a comparison of the dynamics parameters with single-neuron parameters.

B. Graded-Response Models with Refractory Effects

The necessity of having at least two variables for the generation of oscillatory pool behavior has been exploited in another graded-response-like model that relies upon a more realistic neuronal basis [1]. Instead of a second pool as in the equations (5), a subpopulation of refractory neurons (which are refractory because of recent spiking) takes care of the inhibitory effects.

Consider a pool i with pool neurons that can be in either of three different states: active, refractory and quiescent. They can be active, that means, they released a spike in a certain past time interval; or they can be in a refractory state (after firing), or they are quiescent, that is, they do not fire and do not feel the refractory effects any more.

Between the three states, transitions are allowed with a certain probability. We define a_i as the number of neurons of pool i that fired recently, r_i as the number of pool neurons that are in the refractory state and q_i as the number of quiescent pool neurons. Neurons from the a_i subpopulation decay towards the refractory state with a rate α, similarly neurons from the r_i subpopulation decay towards the quiescent state with a rate β. On the other hand, neurons from the quiescent and the refractory subpopulation can be activated by a synaptic input field h_i with transition rates $\sigma_1(h_i)$ and $\sigma_2(h_i)$, respectively, with $\sigma_1(h_i) \leq \sigma_2(h_i)$. The three states with their subpopulations and the allowed transitions are illustrated in Figure 2.

Assuming a first-order decay between the three subpopulations, we find

$$\frac{\mathrm{d}}{\mathrm{d}t} a_i(t) = -\alpha\, a_i(t) + q_i(t)\, \sigma_1[h_i(t)] + r_i(t)\, \sigma_2[h_i(t)] \ , \tag{6}$$

$$\frac{\mathrm{d}}{\mathrm{d}t} q_i(t) = \beta\, r_i(t) - q_i(t)\, \sigma_1[h_i(t)] \ , \tag{7}$$

$$\frac{\mathrm{d}}{\mathrm{d}t} r_i(t) = \alpha\, a_i(t) - r_i(t)\, \sigma_2[h_i(t)] - \beta\, r_i(t) \ , \tag{8}$$

with a synaptic field h_i that depends on the set of activities $\{a_j\}$ of all other pools j. Because all neurons participate in the process, we have $q_i(t) + a_i(t) + r_i(t) = 1$ so that only two quantities are independent,

$$\begin{aligned} \frac{\mathrm{d}}{\mathrm{d}t} a_i(t) = &-\alpha\, a_i(t) + \{1 - a_i(t) - r_i(t)\}\, \sigma_1[h_i(t)] \\ &+ r_i(t)\, \sigma_2[h_i(t)] \ , \end{aligned} \tag{9}$$

$$\frac{\mathrm{d}}{\mathrm{d}t} r_i(t) = \alpha\, a_i(t) - r_i(t)\, \{\beta + \sigma_2[h_i(t)]\} \ . \tag{10}$$

Neurons that release a spike participate in the $a_i(t)$ subpopulation for a short time period. Afterwards, they enter the refractory phase, during which the probability of a new spike release $\sigma_1(h_i)$ due to the input field h_i is low. After

a longer time period without spiking, the refractory effects disappear, and the neuron can release a new spike with a greater probability $\sigma_1(h_i) \leq \sigma_2(h_i)$.

In this model, we have written $a_i(t)$ instead of $A_i(t)$, because we are dealing with the total number of spikes released during a certain period of time, instead of the spike density. That means, $a_i(t) \approx \tau A_i(t)$, with some time constant τ. This creates a difficulty in the normalization condition $q_i(t)+a_i(t)+r_i(t) = 1$ and in the interpretation of the outcome of simulations within this model. Similarly, it is difficult to identify the parameters of the model with experimental data.

Again, after the derivation of our pool dynamics, we will see that the pool model of this subsection can be understood as an approximation of a more general pool dynamics.

III. MICROSCOPIC MODEL: SPIKING NEURONS

In this section we introduce the basic notions that define a pool dynamics.

A. Single Spiking Neurons

Imagine a pool composed of extensively many $N \gg 1$ neurons with the same neuronal dynamics parameters. Inspired by the three-state system description of [1] presented in section II B, we now introduce a three-state *neuron*. A single neuron i, $1 \leq i \leq N$ can be in one of three different states: it can be *inactivated* (**i**), it can be *activated* (**a**), or it can be *firing* (**f**). A neuron can only fire, i.e., release an action potential (or spike), if it is activated. If this is the case, the neuron fires with some probability $\Delta t/\tau[h_i(t)]$ during the interval $(t, t+\Delta t]$, depending on its synaptic input field $h_i(t)$. After the release of a spike, the neuron is to remain inactivated for a certain time period of length γ^{abs}. During this period, it cannot spike, so that it is in an absolute refractory state. Following the absolute refractory period, the neuron enters a relative refractory period during which the neuron has a certain probability $p_{\text{A}} > 0$ of getting activated, and thus a non-vanishing total probability for a spike release. We assume that only the time elapsed since the very last spike of a neuron at t_i^* determines its refractoriness, so that we end up with an activation probability $1 > p_{\text{A}}(t-t_i^*) \geq 0$ for $t \geq t_i^*$. This is the "renewal hypothesis" for spiking neurons; see e.g. [20].

Figure 3 shows the three possible internal states of a single neuron and the allowed transitions. We assume the transitions of a neuron's state between the inactivated and the activated state to occur at a fast timescale as compared to the transition of a neuron from the activated to the firing state and the modification of the activation probability with time. It is therefore sufficient to regard the mean occupation p_{A} of the activated state of a neuron. From the activated state, and depending on the synaptic input field $h_i(t)$, a neuron can be pushed into the firing state with a rate $\{\tau[h_i(t)]\}^{-1}$. A neuron in the firing state releases a single spike and drops immediately back into the inactivated state.

In summary, the total firing probability of a neuron i during an interval $(t-\Delta t, t]$ is given by the joint probability that a neuron is in an activated state and that it is pushed into the firing state by the synaptic field $h_i(t)$ during that time interval,

$$\begin{aligned} &\text{Prob}\{i \in \mathbf{a} \text{ and } i \text{ fires in } (t, t+\Delta t] \text{ due to field } h_i\} \\ &= \text{Prob}\{i \text{ fires in } (t, t+\Delta t] \text{ due to field } h_i \mid i \in \mathbf{a}\} \\ &\quad \times \text{Prob}\{i \in \mathbf{a}\} \\ &= \frac{\Delta t}{\tau[h_i(t)]}\, p_{\text{A}}(t-t_i^*) \ . \end{aligned} \tag{11}$$

The refractory properties are governed by the time course of the activation probability function $1 > p_{\text{A}}(t-t_i^*) \geq 0$, which in this paper will also be called in short the "activation function". It is divided into two parts. For a period of length γ^{abs}, we have the *absolute refractory period*, with $p_{\text{A}}(s) = 0$ and $s = t - t_i^*$ the elapsed time since the last spike. After that period, the neuron enters the *relative refractory period*, during which $p_{\text{A}}(s)$ rises from some value $p_{\text{A}}(\gamma^{\text{abs}})$ towards 1 for $s \to \infty$, according to a differentiable function $P_{\text{A}}(s)$. Between the two refractory periods, we allow a discontinuity of the function $p_{\text{A}}(s)$ at γ^{abs},

$$p_{\text{A}}(s) = \begin{cases} 0 & \text{for } 0 \leq s < \gamma^{\text{abs}} \\ P_{\text{A}}(s) & \text{for } s \geq \gamma^{\text{abs}} \end{cases} \ . \tag{12}$$

B. Synaptic Field

The synaptic field of a single neuron is calculated as follows. Each pool neuron releases a series of action potentials each of which, after a fixed delay period, reaches a synapse of another neuron. This causes a temporal variation of the membrane potential at the postsynaptic neuron. The total variation of the postsynaptic membrane potential due to incoming action potentials is the synaptic field $h_i(t)$. Since we assume passive conducting characteristics of the dendritic tree, the synaptic field is calculated as a sum of the contributions of single action potentials. If we neglect the form of a spike we can characterize action potentials uniquely by their firing times t_i^f, with $f \geq 1$. Here $t_i^1 = t_i^*$ is the most recent action potential of a neuron i in a spike train of delta-functions

$$S_i(t) = \sum_f \delta(t - t_i^f) \ , \tag{13}$$

with $t_i^f \leq t$. The synaptic field is then calculated by inserting the coupling strength J_{ij} for connections from neuron j to neuron i and fixing the temporal variation of the postsynaptic membrane potential $\alpha(s)$,

$$h_i(t) = \sum_{j=1}^{N} J_{ij} \int_0^\infty \mathrm{d}s \, \alpha(s) \, S_j(t-s) \ . \tag{14}$$

In may be well to realize that we have defined pool neurons to have the same input/output connectivity characteristics. This means that all neurons i of the *same* pool $\mathbf{x}$ feel the *same* synaptic field $h(\mathbf{x},t)$. If the coupling strength from a connection conveying signals from a neuron $j \in \mathbf{y}$ of pool $\mathbf{y}$ to a neuron $i \in \mathbf{x}$ of pool $\mathbf{x}$ is designated with $J(\mathbf{x},\mathbf{y})$, we get

$$\begin{aligned} h(\mathbf{x},t) &= \sum_{\mathbf{y}} \sum_{j \in \mathbf{y}} J(\mathbf{x},\mathbf{y}) \int_0^\infty \mathrm{d}s \, \alpha(s) \, S_j(t-s) \\ &= \sum_{\mathbf{y}} J(\mathbf{x},\mathbf{y}) \int_0^\infty \mathrm{d}s \, \alpha(s) \, A(\mathbf{y}, t-s) \end{aligned} \tag{15}$$

with the pool activity

$$A(\mathbf{x},t) = \sum_{j \in \mathbf{x}} S_j(t) \ . \tag{16}$$

The pool activity $A(\mathbf{x},t)$ has the dimension spikes/time and is extensive; if desired, it can be normalized. The integration of $A(\mathbf{x},t)$ over a small time interval of length Δt is then the total number of released spikes of all pool neurons during that interval.

The time course of the membrane potential variation due to the input of a single spike follows qualitatively the form of an "alpha" function: It rises to a maximum value and then decays back towards zero. In our case, the kernel $\alpha(s)$ will be chosen to be a delta function, $\alpha(s) = \delta(s - \Delta^{\mathrm{ax}})$, so that $h(\mathbf{x},t) = \sum_{\mathbf{y}} J(\mathbf{x},\mathbf{y}) A(\mathbf{x}, t - \Delta^{\mathrm{ax}})$ or to have a Poisson-like time course $\alpha(s) = \alpha^{(k)}(s)$ with

$$\alpha^{(k)}(s) = \Theta(s - \Delta^{\mathrm{ax}}) \, \frac{(s - \Delta^{\mathrm{ax}})^k}{c_k} \exp\left[-(s - \Delta^{\mathrm{ax}})/\tau_\alpha\right] \ . \tag{17}$$

Here τ_α is the rise time, s the time difference to the firing of the presynaptic neuron, c_k the normalization factor, Δ^{ax} the axonal delay time between two pools and Θ the Heavyside step function [1]. The constant c_k can be used to normalize the alpha-function by its maximum amplitude or its area. The maximum of the alpha function is attained at $s_{\max} = \Delta^{\mathrm{ax}} + k\tau_\alpha$ and is equal to $\alpha_{\max} = (k\tau_\alpha)^k e^{-k}$. The area of the function gives $F_\alpha = \tau_\alpha^{k+1} \Gamma(k+1)$ with $\Gamma(k+1) = k!$ for $k \in \mathbb{N}$.

The pool form of the synaptic field, viz., the last equation of (15), will be used in the following sections for the formulation of our pool dynamics.

---

[1] $\Theta(x) = 1$ for $x \geq 0$ and 0 otherwise.

IV. POOLS OF SPIKING NEURONS

The present section is devoted to a derivation of the time evolution of the key variable $A(\mathbf{x},t)$, the activity. To this end, we start with the survival function that tells us how long a neuron survives without spiking. This will allow us to obtain an expression for calculating $A(\mathbf{x},t)$, viz., Eq. (26), which constitutes the key to the ensuing analysis.

A. Survival Function

In a pool of extensively many equivalent neurons, we take advantage of the property that all neurons of a pool $\mathbf{x}$ feel a common synaptic field. We now group the pool neurons into subgroups with the same last firing times t^*; for example, $n(\mathbf{x},t,t^*)\Delta t^*$ is the total number of pool neurons $i \in \mathbf{x}$ found at time t with $t_i^* \in (t^*-\Delta t^*, t^*]$. For times $t \geq t^*$, we can then look at the time development of these subgroups. Let [2] $S_h(\mathbf{x},t,t^*)$ be the fraction of a group of neurons that has spiked at least once during $(t^*,t]$. Then this fraction changes in the interval $(t-\Delta t,t]$ by

$$\Delta S_h(t,t^*) = \frac{\Delta t}{\tau[h(\mathbf{x},t)]}\, p_{\rm A}(t-t^*)\,[1-S_h(\mathbf{x},t,t^*)]\ . \tag{18}$$

Taking the limit $\Delta t \to 0$, we get for the time derivative of the fraction of neurons that did *not* spike again during $(t^*,t]$ [i.e. of $D_h(\mathbf{x},t,t^*) := 1-S_h(\mathbf{x},t,t^*)$],

$$\frac{\rm d}{{\rm d}t} D_h(\mathbf{x},t,t^*) = -\frac{1}{\tau[h(\mathbf{x},t)]}\, p_{\rm A}(\mathbf{x},t-t^*)\, D_h(\mathbf{x},t,t^*)\ . \tag{19}$$

Integration including the boundary condition $D_h(\mathbf{x},t^*,t^*)=1$ yields the "survival function" [11]

$$D_h(\mathbf{x},t,t^*) = \exp\left\{-\int_{t^*}^{t} {\rm d}t'\, \frac{1}{\tau[h(\mathbf{x},t')]}\, p_{\rm A}(t'-t^*)\right\}\ , \tag{20}$$

that is a measure of the fraction of neurons that have spiked last at t^* and did not spike again until t.

B. Time Evolution of the Pool-Averaged Activity

Using the survival function, we can now calculate $n(\mathbf{x},t,t^*)\Delta t^*$. It is equal to the number of neurons $A(\mathbf{x},t^*)\,\Delta t^*$ that actually spiked during the interval $(t^*-\Delta t^*,t^*]$, multiplied by the fraction of surviving neurons at time t,

$$n(\mathbf{x},t,t^*)\Delta t^* = D_h(\mathbf{x},t,t^*)\, A(\mathbf{x},t^*)\, \Delta t^*\ . \tag{21}$$

Taking the limit $\Delta t^* \to 0$ and adding over all possible last firing times t^* (that is, over all possible refractory states of the pool neurons), we get the total number of pool neurons,

$$\begin{aligned} N(\mathbf{x}) &= \int_{-\infty}^{t} {\rm d}t^*\, D_h(\mathbf{x},t,t^*) A(\mathbf{x},t^*) \\ &= \int_{0}^{\infty} {\rm d}s\, D_h(\mathbf{x},t,t-s) A(\mathbf{x},t-s)\ . \end{aligned} \tag{22}$$

By exploiting the same kind of argument, we can calculate another important macroscopic pool variable. The mean fraction of activated pool neurons that have spiked for the last time at t^* is given by $p_{\rm A}(t-t^*)$. Therefore the mean number of *inactivated* neurons under the same conditions is $1-p_{\rm A}(t-t^*)$. The number of pool neurons that spiked last during $(t^*-\Delta t,t^*]$ and that are inactivated at time t is then

$$n_{\rm I}(\mathbf{x},t,t^*)\Delta t^* = [1-p_{\rm A}(t-t^*)]\, D_h(\mathbf{x},t,t^*)\, A(\mathbf{x},t^*)\, \Delta t^*\ . \tag{23}$$

---

[2] The suffix h denotes a functional dependence of the function upon the field $h(t')$ during the past time $t' \in (t^*,t]$.

From this we can calculate the total number of inactivated neurons of the pool,

$$N_{\mathrm{I}}(\mathbf{x},t) = \int_{-\infty}^{t} \mathrm{d}t^* \left[1 - p_{\mathrm{A}}(t-t^*)\right] D_h(\mathbf{x},t,t^*) A(\mathbf{x},t^*)$$
$$= \int_{0}^{\infty} \mathrm{d}s \left[1 - p_{\mathrm{A}}(s)\right] D_h(\mathbf{x},t,t-s) A(\mathbf{x},t-s) \ . \tag{24}$$

The number of pool neurons that *can* contribute to the activity $A(t+\Delta t)\Delta t$ during the next time step $(t, t+\Delta t]$, is given by the total number of *activated* neurons, $N(\mathbf{x}) - N_{\mathrm{I}}(\mathbf{x},t)$. Because the activated neurons contribute to spiking with a probability $\{\tau[h(\mathbf{x},t)]\}^{-1}\Delta t$, we get for the activity

$$A(\mathbf{x},t+\Delta t)\Delta t = \frac{\Delta t}{\tau[h(\mathbf{x},t)]} \left[N(\mathbf{x}) - N_{\mathrm{I}}(\mathbf{x},t)\right] \ . \tag{25}$$

This equation is valid as long as $A(\mathbf{x},t+\Delta t)\Delta t \ll N_{\mathrm{I}}(\mathbf{x},t)$. That is to say, as long as $N_{\mathrm{I}}(\mathbf{x},t)$ can be considered as approximately constant during a time interval of length Δt. We note that in this case the activation $A(\mathbf{x},t+\Delta t)$ does not depend on the length of the time interval Δt. For any small enough Δt, the result will be the same. Hence we can take the limit $\Delta t \to 0$,

$$A(\mathbf{x},t) = \frac{1}{\tau[h(\mathbf{x},t)]} \left[N(\mathbf{x}) - N_{\mathrm{I}}(\mathbf{x},t)\right] \ . \tag{26}$$

This is a nonlinear integral equation for the time evolution of the activity $A(\mathbf{x},t)$; cf. (25). It is the central equation from which we derive the pool dynamics. The nonlinearity of (26) is hidden in the synaptic field $h(\mathbf{x},t)$, which can depend on the activities of all other pools that provide synaptic input, including itself [according to the second line of Eq. (15) in section III B].

Starting from this equation, we will gain a differential equation system that describes the dynamics contained implicitly in Eq. (26) and that is better suited to analytical treatment and numerical simulations. This will provide us with a straightforward and natural description of the activity dynamics of neuronal assemblies. In addition, the differential equation form will allow us to compare the system with the models discussed in section II.

V. DIFFERENTIAL EQUATION POOL DYNAMICS

In this section, we are going to reduce the integral equations for the pool synaptic field $h(\mathbf{x},t)$ and the pool activity $A(\mathbf{x},t)$ to a system of coupled differential equations. We begin with the synaptic field because its derivation is straightforward. Then we derive the dynamics of the pool activity.

A. Synaptic Field

The aim of this section is to express the synaptic field acting on the neurons of a specified pool through differential equations. For a synaptic field calculated according to the second line of (15), with an alpha kernel of the type $\alpha(s) = \alpha^{(k)}(s)$, $k \in \mathbb{N}$ from Eq. (17), we define additional fields

$$h^{(l)}(\mathbf{x},t) = \sum_{\mathbf{y}} J(\mathbf{x},\mathbf{y}) \int_0^\infty \mathrm{d}s \, \alpha^{(l)}(s) \, A(\mathbf{y},t-s) \tag{27}$$

with $l \in \mathbb{N}$, $0 \le l \le k$. With this definition, the field we are looking for is $h^{(k)}(\mathbf{x},t)$, and it is straightforward to see that the field dynamics can be expressed by the differential equation system

$$\frac{\mathrm{d}}{\mathrm{d}t} h^{(l)}(\mathbf{x},t) = l\frac{c_{l-1}}{c_l} h^{(l-1)}(\mathbf{x},t) - \frac{1}{\tau_\alpha} h^{(l)}(\mathbf{x},t) \ ,$$
$$\frac{\mathrm{d}}{\mathrm{d}t} h^{(0)}(\mathbf{x},t) = \sum_{\mathbf{y}} J(\mathbf{x},\mathbf{y}) \, A(\mathbf{y},t-\Delta^{\mathrm{ax}})$$
$$- \frac{1}{\tau_\alpha} h^{(0)}(\mathbf{x},t) \ . \tag{28}$$

These equations have been calculated by differentiating repeatedly the field $h^{(k)}(\mathbf{x},t)$ and by extracting the terms of the additional fields $h^{(l)}(\mathbf{x},t)$. The coupling with other pools enters the differential equation system only in the last equation of (28).

A similar procedure can be used if a sum of several alpha functions of the type $\alpha^{(k)}(s)$, $k \in \mathbb{N}$, is used as synaptic kernel. In this case, separate differential equation systems of the form (28) have to be used to compute the different field contributions. The corresponding fields $h^{(k)}(\mathbf{x},t)$ have then to be added. This also opens the possibility to approximate alpha functions with delay by weighted additions of alpha functions without delay, thus resulting in a differential equation system without delay for the synaptic field.

For $k \notin \mathbb{N}$, the integral equation [second line of Eq. (15)] instead of the differential equations (28) has to be used.

B. Activation Probability Functions

The behavior of a neuron i during its relative refractory period (see section III A) is characterized by the differentiable function $P_{\mathrm{A}}(s)$, with $s = (t - t_i^*)$. From now on, we will restrict ourselves to the exponential case (exp), the case of a sigmoid-like time evolution of the activation function after the absolute refractory period (sigm), and the case of an inverse decay (inv):

$$P_{\mathrm{A}}(s) = \begin{cases} 1 - p_0 \exp[-(s-\gamma^{\mathrm{abs}})/\tau_{\mathrm{ref}}] & \mathrm{exp} \\ 1 - p_0/\{1+\exp[(s-s_0)/\tau_{\mathrm{ref}}]\} & \mathrm{sigm} \\ 1 - \tau_{\mathrm{ref}}/(s-s_0) & \mathrm{inv} \end{cases} . \tag{29}$$

The constants p_0, τ_{ref} and s_0 are free parameters of the activation function. It has to be verified that $0 \le P_{\mathrm{A}}(s) \le 1$ for $\gamma^{\mathrm{abs}} \le s \le \infty$; for example, for the inverse activation function in (29), this means that we effectively set $\gamma^{\mathrm{abs}} > \tau_{\mathrm{ref}} + s_0$.

The activation functions obey the differential equations

$$\frac{\mathrm{d}}{\mathrm{d}s} P_{\mathrm{A}}(s) = \begin{cases} \frac{1}{\tau_{\mathrm{ref}}}[1 - P_{\mathrm{A}}(s)] & \mathrm{exp} \\ \frac{1}{\tau_{\mathrm{ref}}}[1 - P_{\mathrm{A}}(s)]\{1 - [1 - P_{\mathrm{A}}(s)]/p_0\} & \mathrm{sigm} \\ \frac{1}{\tau_{\mathrm{ref}}}[1 - P_{\mathrm{A}}(s)]^2 & \mathrm{inv} \end{cases} . \tag{30}$$

These properties will be used in the subsequent sections for the derivation of the model. Figure 4 shows the different activation functions.

C. Time Evolution of the Number of Inactivated Neurons

Here and in the following subsection, we will reduce the integral equation for the pool activity (26) to a differential equation system. To this end, we consider the time development of the total number of inactivated neurons $N_{\mathrm{I}}(\mathbf{x},t)$. We use the t^*-form of equation (24), the property $\mathrm{d}/\mathrm{d}t\; D(\mathbf{x},t,t^*) = -\{\tau[h(\mathbf{x},t)]\}^{-1} p_{\mathrm{A}}(t-t^*)\, D(\mathbf{x},t,t^*)$, and note that for a function of the type $h(t,t^*)$ we have

$$\frac{\mathrm{d}}{\mathrm{d}t} \int_{f(t)}^{g(t)} \mathrm{d}t^*\, h(t,t^*) = F[t,f(t)]\, f'(t) - F[t,g(t)]\, g'(t) + \int_{f(t)}^{g(t)} \mathrm{d}t^*\, \frac{\partial}{\partial t} h(t,t^*)$$

so as

$$\frac{\mathrm{d}}{\mathrm{d}t} N_{\mathrm{I}}(\mathbf{x},t) = A(\mathbf{x},t) - \int_0^\infty \mathrm{d}s \left[\frac{\mathrm{d}}{\mathrm{d}s} p_{\mathrm{A}}(s)\right] D_h(\mathbf{x},t,t-s) A(\mathbf{x},t-s) - \frac{1}{\tau[h(\mathbf{x},t)]} \int_0^\infty \mathrm{d}s\, [1 - p_{\mathrm{A}}(s)] p_{\mathrm{A}}(s)\, D_h(\mathbf{x},t,t-s) A(\mathbf{x},t-s) . \tag{31}$$

Thus the number of inactivated neurons grows with $A(\mathbf{x},t)$. This makes sense, because neurons that spike get inactivated immediately afterwards. On the other hand, the number of inactivated neurons decreases with time as the refractory effect on neurons decreases. Moreover, the term $p_{\mathrm{A}}(s)[1 - p_{\mathrm{A}}(s)]$ selects a time window of the activity that contributes to further changes of $N_{\mathrm{I}}(\mathbf{x},t)$.

Exploiting the properties (30) of the chosen activation functions (29) of section V B, and taking into account the discontinuity of $p_{\mathrm{A}}(s)$ at $s = \gamma^{\mathrm{abs}}$, we get

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}N_{\mathrm{I}}(\mathbf{x},t) &= A(\mathbf{x},t) - P_{\mathrm{A}}(\gamma^{\mathrm{abs}})A(\mathbf{x},t-\gamma^{\mathrm{abs}}) \\ &\quad -\frac{1}{\tau[h(\mathbf{x},t)]}\int_0^\infty \mathrm{d}s\, P_{\mathrm{A}}(s)[1-P_{\mathrm{A}}(s)]D_h(\mathbf{x},t,t-s)A(\mathbf{x},t-s) \\ &\quad -\begin{cases} \frac{1}{\tau_{\mathrm{ref}}}\int_{\gamma^{\mathrm{abs}}}^\infty \mathrm{d}s\,[1-P_{\mathrm{A}}(s)]\, D_h(\mathbf{x},t,t-s)A(\mathbf{x},t-s) & \text{exp. } p_{\mathrm{A}}(s) \\ \frac{1}{\tau_{\mathrm{ref}}}\int_{\gamma^{\mathrm{abs}}}^\infty \mathrm{d}s\,[1-P_{\mathrm{A}}(s)]\left\{1-\frac{[1-P_{\mathrm{A}}(s)]}{p_0}\right\} D_h(\mathbf{x},t,t-s)A(\mathbf{x},t-s) & \text{sigm. } p_{\mathrm{A}}(s) \\ \frac{1}{\tau_{\mathrm{ref}}}\int_{\gamma^{\mathrm{abs}}}^\infty \mathrm{d}s\,[1-P_{\mathrm{A}}(s)]^2 D_h(\mathbf{x},t,t-s)A(\mathbf{x},t-s) & \text{inv. } p_{\mathrm{A}}(s) \end{cases}\end{aligned} \tag{32}$$

Instead of $P_{\mathrm{A}}(s)$, we want to express our dynamics using the original activation function $p_{\mathrm{A}}(s)$. For this purpose, we introduce a quantity

$$\begin{aligned}M(\mathbf{x},t) &= \int_0^{\gamma^{\mathrm{abs}}} \mathrm{d}s\, D_h(\mathbf{x},t,t-s)A(\mathbf{x},t-s) \\ &= \int_0^{\gamma^{\mathrm{abs}}} \mathrm{d}s\, A(\mathbf{x},t-s)\ ,\end{aligned} \tag{33}$$

which is interpreted as the number of inactivated neurons for a pool with absolute refractory period only, and rewrite the previous equation (32) in the form

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}N_{\mathrm{I}}(\mathbf{x},t) &= A(\mathbf{x},t) - p_{\mathrm{A}}(\gamma^{\mathrm{abs}})A(\mathbf{x},t-\gamma^{\mathrm{abs}}) \\ &\quad -\frac{1}{\tau[h(\mathbf{x},t)]}\int_0^\infty \mathrm{d}s\, p_{\mathrm{A}}(s)[1-p_{\mathrm{A}}(s)]D_h(\mathbf{x},t,t-s)A(\mathbf{x},t-s) \\ &\quad -\begin{cases} \frac{1}{\tau_{\mathrm{ref}}}\int_0^\infty \mathrm{d}s\,[1-p_{\mathrm{A}}(s)]\, D_h(\mathbf{x},t,t-s)A(\mathbf{x},t-s) & \text{exp. } p_{\mathrm{A}}(s) \\ \frac{1}{\tau_{\mathrm{ref}}}\int_0^\infty \mathrm{d}s\,[1-p_{\mathrm{A}}(s)]\left\{1-\frac{[1-p_{\mathrm{A}}(s)]}{p_0}\right\} D_h(\mathbf{x},t,t-s)A(\mathbf{x},t-s) & \text{sigm. } p_{\mathrm{A}}(s) \\ \frac{1}{\tau_{\mathrm{ref}}}\int_0^\infty \mathrm{d}s\,[1-p_{\mathrm{A}}(s)]^2 D_h(\mathbf{x},t,t-s)A(\mathbf{x},t-s) & \text{inv. } p_{\mathrm{A}}(s) \end{cases} \\ &\quad +\frac{1}{\tau_{\mathrm{ref}}}M(\mathbf{x},t)\ .\end{aligned} \tag{34}$$

What is the relevance of this equation? Since the dynamics of $A(\mathbf{x},t)$ is determined by the dynamics of $h(\mathbf{x},t)$ and of $N_{\mathrm{I}}(\mathbf{x},t)$, it is of primary importance to understand the time development of the number of inactivated neurons. In the next section, we will see how this allows us to derive systematically a system of differential equations for the dynamics of $A(\mathbf{x},t)$.

D. Pool Dynamics

The number $N_{\mathrm{I}}(\mathbf{x},t)$ of inactivated neurons of a pool is the assembly-averaged mean inactivation probability. For the calculation of the mean, we need the momentary density $\rho(\mathbf{x},t,t^*) = D_h(\mathbf{x},t,t^*)A(\mathbf{x},t^*)$ of neurons with a refractory state defined by their last spike at t^*. With this density and the definition

$$\langle \ldots \rangle := \int_{-\infty}^{t} \mathrm{d}t^*\, \rho(\mathbf{x},t,t^*)\ \ldots \tag{35}$$

we can write

$$N_{\mathrm{I}}(\mathbf{x},t) = \langle 1 - p_{\mathrm{A}}(t-t^*)\rangle\ . \tag{36}$$

The kernel $1 - p_{\mathrm{A}}(t-t^*)$ determines the influence of the past activity on the quantity $N_{\mathrm{I}}(\mathbf{x},t)$. Instead of using integral equations which incorporate the past activity by means of equidistant time slices (imagine a Riemann sum approximation of the integral equations), we could try to incorporate the past using a set of kernels similar to

$1 - p_{\rm A}(t - t^*)$. The underlying problem is that of the reducibility of an integro-differential equation to a system of differential equations. It has been treated by a number of authors, e.g. see ([8,9]). In principle, a reduction of Eqs. (31) or (34) into a chain of differential equations is possible for a suitable choice of intermediary variables. The problem is that there is no systematic derivation of these additional variables, so that we have to guess. As indicated above we will use the function $1 - p_{\rm A}(t - t^*)$ for this purpose.

To accomplish the reduction of (34), the number of inactivated neurons $N_{\rm I}(\mathbf{x}, t)$ will be treated in a way equivalent to $N^{(1)}(\mathbf{x}, t)$, the total number of pool neurons $N(\mathbf{x})$ will be handled as $N^{(0)}(\mathbf{x})$, and the number $M(\mathbf{x}, t)$ of inactivated neurons for a pool with absolute refractory period only, as $N^{(\infty)}(\mathbf{x}, t)$. Furthermore, we remark that the definitions (22) of $N(\mathbf{x}) = N^{(0)}(\mathbf{x})$, (24) of $N_{\rm I}(\mathbf{x}, t) = N^{(1)}(\mathbf{x}, t)$ and (33) of $M(\mathbf{x}, t) = N^{(\infty)}(\mathbf{x}, t)$ are equivalent to

$$\begin{aligned} N^{(0)}(\mathbf{x}) &= \left\langle [1 - p_{\rm A}(t - t^*)]^0 \right\rangle \ , \\ N^{(1)}(\mathbf{x}, t) &= \left\langle [1 - p_{\rm A}(t - t^*)]^1 \right\rangle \ , \\ N^{(\infty)}(\mathbf{x}, t) &= \left\langle [1 - p_{\rm A}(t - t^*)]^\infty \right\rangle \ . \end{aligned} \tag{37}$$

Extending these definitions, we introduce additional time-dependent inactivation quantities, or "recovery variables",

$$N^{(m)}(\mathbf{x}, t) := \langle [1 - p_{\rm A}(t - t^*)]^m \rangle \tag{38}$$

so that, for $m \in \mathbb{N}$,

$$\begin{aligned} N^{(m)}(\mathbf{x}, t) &= \int_{-\infty}^{t} \mathrm{d}t^* \, \{1 - p_{\rm A}(t - t^*)\}^m \, D_h(\mathbf{x}, t, t^*) A(\mathbf{x}, t^*) \\ &= \int_0^\infty \mathrm{d}s \, \{1 - p_{\rm A}(s)\}^m \, D_h(\mathbf{x}, t, t - s) A(\mathbf{x}, t - s) \ . \end{aligned} \tag{39}$$

The $N^{(m)}(\mathbf{x}, t)$ obey the relationship

$$N^{(0)}(\mathbf{x}) \geq N^{(1)}(\mathbf{x}, t) \geq N^{(2)}(\mathbf{x}, t) \geq \ldots \geq N^{(\infty)}(\mathbf{x}, t) \quad \forall t \tag{40}$$

and have the property

$$\begin{aligned} &\int_0^\infty \mathrm{d}s \, \{1 - p_{\rm A}(s)\}^m p_{\rm A}(s) \, D_h(\mathbf{x}, t, t - s) A(\mathbf{x}, t - s) \\ &\qquad = N^{(m)}(\mathbf{x}, t) - N^{(m+1)}(\mathbf{x}, t) \ . \end{aligned} \tag{41}$$

Figure 5 shows an example of a sigmoidal activation function $p_{\rm A}(s)$ with an absolute refractory period of length $\gamma^{\rm abs}$ and the recovery kernels $[1 - p_{\rm A}(s)]^m$.

For these recovery variables, we can calculate the time derivative in the same way as for $N_{\rm I}(\mathbf{x}, t)$ in Eq. (31),

$$\begin{aligned} \frac{\mathrm{d}}{\mathrm{d}t} N^{(m)}(\mathbf{x}, t) = A(\mathbf{x}, t) &- m \int_0^\infty \mathrm{d}s \, [1 - p_{\rm A}(s)]^{m-1} \left[\frac{\mathrm{d}}{\mathrm{d}s} p_{\rm A}(s) \right] D_h(\mathbf{x}, t, t - s) A(\mathbf{x}, t - s) \\ &- \frac{1}{\tau[h(\mathbf{x}, t)]} \int_0^\infty \mathrm{d}s \, [1 - p_{\rm A}(s)]^m p_{\rm A}(s) \, D_h(\mathbf{x}, t, t - s) A(\mathbf{x}, t - s) \ . \end{aligned} \tag{42}$$

Finally, with the property (41), we obtain a recursive set of differential equations

$$\begin{aligned} \frac{\mathrm{d}}{\mathrm{d}t} N^{(m)}(\mathbf{x}, t) = A(\mathbf{x}, t) &- \{1 - [1 - p_{\rm A}(\gamma^{\rm abs})]^m\} A(\mathbf{x}, t - \gamma^{\rm abs}) \\ &- \frac{1}{\tau[h(\mathbf{x}, t)]} [N^{(m)}(\mathbf{x}, t) - N^{(m+1)}(\mathbf{x}, t)] \\ &- \begin{cases} \frac{m}{\tau_{\rm ref}} [N^{(m)}(\mathbf{x}, t) - M(\mathbf{x}, t)] & \text{exp. } p_{\rm A}(s) \\ \frac{m}{\tau_{\rm ref}} \{N^{(m)}(\mathbf{x}, t) - M(\mathbf{x}, t) - [N^{(m+1)}(\mathbf{x}, t) - M(\mathbf{x}, t)]/p_0\} & \text{sigm. } p_{\rm A}(s) \\ \frac{m}{\tau_{\rm ref}} [N^{(m+1)}(\mathbf{x}, t) - M(\mathbf{x}, t)] & \text{inv. } p_{\rm A}(s) \end{cases} . \end{aligned} \tag{43}$$

The last recovery variable $N^{(\infty)}(\mathbf{x}, t) = M(\mathbf{x}, t)$ increases with the number of spiking neurons and decreases with the number of neurons that are released from their absolute refractory phase,

$$\frac{\mathrm{d}}{\mathrm{d}t}M(\mathbf{x},t) = A(\mathbf{x},t) - A(\mathbf{x},t-\gamma^{\mathrm{abs}}) \ . \tag{44}$$

This completes our derivation of the pool dynamics. The system (43) looks linear but it is not since the field $h(\mathbf{x},t)$ (28) contains the recovery variable $N^{(1)}(\mathbf{x},t)$ through the activity $A(\mathbf{x},t)$. The complete dynamics is defined by the field dynamics given by Eqs. (28) of section V A together with the dynamics of the recovery variables given by Eqs. (43) and (44). The spike density acts only as an auxiliary variable that is calculated from the first recovery variable using the main equation (26) of section IV B

$$A(\mathbf{x},t) = \frac{1}{\tau[h(\mathbf{x},t)]}[N(\mathbf{x}) - N^{(1)}(\mathbf{x},t)] \ . \tag{45}$$

Other pools $\mathbf{y}$ influence the dynamics of pool $\mathbf{x}$ through $A(\mathbf{y},t)$ in the last equation of the field $h(\mathbf{x},t)$ in (28). Axonal delays appear in the last equation of (28), in the dynamics of the recovery variables (43) due to the discontinuity of $p_{\mathrm{A}}(s)$ at γ^{abs}, and in Eq. (44) also because of the absolute refractory period.

To model pool dynamics using differential equations without delays, a differentiable activation function $p_{\mathrm{A}}(s)$ without absolute refractory period has to be chosen. In this case, the system (43) reduces to

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}N^{(m)}(\mathbf{x},t) = A(\mathbf{x},t) &- \frac{1}{\tau[h_i(\mathbf{x},t)]}[N^{(m)}(\mathbf{x},t) - N^{(m+1)}(\mathbf{x},t)] \\ &- \begin{cases} \frac{m}{\tau_{\mathrm{ref}}}N^{(m)}(\mathbf{x},t) & \text{exp. } p_{\mathrm{A}}(s) \\ \frac{m}{\tau_{\mathrm{ref}}}[N^{(m)}(\mathbf{x},t) - N^{(m+1)}(\mathbf{x},t)/p_0] & \text{sigm. } p_{\mathrm{A}}(s) \\ \frac{m}{\tau_{\mathrm{ref}}}N^{(m+1)}(\mathbf{x},t) & \text{inv. } p_{\mathrm{A}}(s) \end{cases} \ .\end{aligned} \tag{46}$$

The remaining axonal delay in (28) can be avoided using the procedure for synaptic fields with alpha functions with delay as explained in section V A.

VI. CONSEQUENCES

In this section, we are going to analyze in detail the consequences that follow from using the system (43) for the calculation of assembly dynamics. Equations (43) are exact for assemblies composed of extensively many spiking neurons. Nevertheless, for a numerical implementation of the dynamics, the infinite chain of differential equations has to be approximated by a finite differential equation system. Breaking the chain earlier or later leads to a dynamics that follow the exact result in a smooth fashion or in every detail. One can therefore approximate the pool dynamics with the desired accuracy. Here we discuss systematic approximations to the differential-equation system and show simulation results for the different approximation schemes.

A. Systematic Approximations

Contrary to previous work on pool dynamics, the present procedure is exact for pools of extensively many neurons since it does not rely upon time-averaging for its derivation. This allows us to quantitatively model pool activities well beyond the quasistationary regime. But for numerical simulations, the infinite chain of differential equations has to be approximated by a finite system.

Because of property (40) of the recovery variables, we can approximate the infinite chain of differential equations (43) by breaking it at a desired recovery variable $N^{(n+1)}(\mathbf{x},t)$ and by introducing an appropriate dynamics for this quantity. In this section, we will proceed to analyze different approximations of the differential equation system for our pool model.

There are two sensible ways of approximating $N^{(n+1)}(\mathbf{x},t)$, which differ according to the desired dynamical simulation range. Assuming that n is large enough, the influence of the relative refractory field on the $(n+1)$-th recovery variable can be neglected, and we can approximate $N^{(n+1)}(\mathbf{x},t) \approx M(\mathbf{x},t)$; or $N^{(n+1)}(\mathbf{x},t) \approx 0$ if we are dealing with neurons without absolute refractory period.

- For *fast, transient dynamics* with sharp activity steps, $N^{(n+1)}(\mathbf{x},t)$ is then calculated according to the dynamics of $M(\mathbf{x},t)$,

$$\frac{\mathrm{d}}{\mathrm{d}t}N^{(n+1)}(\mathbf{x},t) \approx \frac{\mathrm{d}}{\mathrm{d}t}M(\mathbf{x},t) = A(\mathbf{x},t) - A(\mathbf{x},t-\gamma^{\mathrm{abs}}) \ , \tag{47}$$

or, for neurons without absolute refractory period, we use equation (46) for $\mathrm{d}/\mathrm{d}t\, N^{(n)}(\mathbf{x},t)$ and $N^{(n+1)}(\mathbf{x},t) \approx 0$. In the case of exponential or sigmoidal $p_{\mathrm{A}}(s)$, this gives for the n'th recovery variable

$$\frac{\mathrm{d}}{\mathrm{d}t} N^{(n)}(\mathbf{x},t) \approx A(\mathbf{x},t) - \left[\frac{1}{\tau[h(\mathbf{x},t)]} + \frac{n}{\tau_{\mathrm{ref}}}\right] N^{(n)}(\mathbf{x},t) \ . \tag{48}$$

- For *slow* dynamics, we can approximate $N^{(n+1)}(\mathbf{x},t)$ by its stationary value. For slow dynamics, the activity $A(\mathbf{x},t)$ and the field $h(\mathbf{x},t)$ are approximately constant during the time period of the kernel $[1-p_{\mathrm{A}}(s)]^{n+1}$ of $N^{(n+1)}(\mathbf{x},t)$. This means that

$$N^{(n+1)}(\mathbf{x},t) \approx [\gamma^{\mathrm{abs}} + \kappa_h^{(n+1)}(\mathbf{x})] A(\mathbf{x},t) \tag{49}$$

with $\kappa_h^{(n+1)}(\mathbf{x})$ being the time constant of the $(n+1)$-th kernel due to relative refractory effects,

$$\kappa_h^{(n+1)}(\mathbf{x}) = \int_{\gamma^{\mathrm{abs}}}^{\infty} \mathrm{d}s\, [1-p_{\mathrm{A}}(s)]^{n+1}\, D_h(\mathbf{x},t,t-s) \ . \tag{50}$$

Here $\kappa_h^{(n+1)}(\mathbf{x})$ has been evaluated for a quasistationary field $h(\mathbf{x},t)$ (i.e., the field is assumed to be constant for a period during which the expression in the integral is large), and, thus, it is written without an explicit dependency on t. It depends, however, on the field h. For neurons with *absolute refractory period only*, $\kappa_h^{(n+1)}(\mathbf{x}) = 0$ and we can use $N^{(n+1)}(\mathbf{x},t) \approx \gamma^{\mathrm{abs}} A(\mathbf{x},t)$ as the stationary approximation. Similarly, in case of pools with *relative refractory period only*, we use $N^{(n+1)}(\mathbf{x},t) \approx \kappa_h^{(n+1)}(\mathbf{x}) A(\mathbf{x},t)$.

The slow approximation is exact when the activity approaches a stationary value.

B. Zeroth-Order Approximation: Stationary Solution and Gain Function

For constant input field $h(\mathbf{x},t) \equiv h(\mathbf{x})$ and stationary activity $A(\mathbf{x},t) \equiv A(\mathbf{x})$, it is $\mathrm{d}/\mathrm{d}t\, N^{(m)}(\mathbf{x},t) = 0\ \forall m$, so that from the assembly dynamics (43) and (45) only remains

$$A(\mathbf{x}) = \frac{1}{\tau[h(\mathbf{x})]} [N(\mathbf{x}) - N^{(1)}(\mathbf{x})] \ . \tag{51}$$

Using this equation, and the expression (49) for $N^{(1)}(\mathbf{x})$ in the stationary case,

$$N^{(1)}(\mathbf{x}) = [\gamma^{\mathrm{abs}} + \kappa_h^{(1)}(\mathbf{x})] A(\mathbf{x}) \ , \tag{52}$$

we can calculate the normalized stationary spike density to

$$G[\mathbf{x}, h(\mathbf{x})] := \frac{A(\mathbf{x})}{N(\mathbf{x})} = \frac{1}{\gamma^{\mathrm{abs}} + \tau[h(\mathbf{x})] + \kappa_h^{(1)}(\mathbf{x})} \ . \tag{53}$$

If $\tau[h]$ is a monotonously decreasing function of h, we get a *gain function* $G[h]$ that saturates at large h and has a sigmoidal-like appearance.

With the often used ansatz $\tau[h] = \tau_0 \exp[-2\beta(h-\theta)]$ with spike-rate at threshold τ_0^{-1}, noise parameter β, and threshold θ (see e.g. [11]), we get

$$G[\mathbf{x}, h(\mathbf{x})] = \frac{1}{\gamma^{\mathrm{abs}}} \frac{1}{1 + \exp\{-2\beta[h(\mathbf{x}) - \theta']\} + \kappa_h^{(1)}(\mathbf{x})/\gamma^{\mathrm{abs}}} \tag{54}$$

with the modified threshold

$$\theta' = \theta + 1/(2\beta) \ln(\tau_0/\gamma^{\mathrm{abs}}) \tag{55}$$

and the relative refractory period time constant $\kappa_h^{(1)}(\mathbf{x})$ as defined in Eq. (50).

Figure 6 shows the stationary spike density $A(\mathbf{x})$ as a function of the synaptic field $h(\mathbf{x})$. The pool spike rate $A(\mathbf{x})$ saturates at $N(\mathbf{x})/\gamma^{\text{abs}}$ as it is bounded by the inverse length of the absolute refractory period. The time constant $\kappa_h^{(1)}(\mathbf{x})$ quantifies the influence of the relative refractory period on the stationary pool spike rate. It reduces the activity for intermediate fields $h(\mathbf{x}) \approx \theta$. The noise factor β and the effective threshold θ' determine the slope and the inflection point of the gain function. Increasing the length of the absolute refractory period γ^{abs} or decreasing the firing rate at threshold τ_0^{-1} shifts the effective threshold towards higher values. We see that the present model lets us understand the gain function quantitatively in terms of the *microscopic neuronal parameters* γ^{abs}, $\kappa_h^{(1)}$, τ_0, β, and θ. This marks a difference to standard gain functions as those used with other graded-response pool models.

C. First-Order Approximation: Quasistationary Dynamics and Graded-Response

The graded-response models presented in the introductory sections II A and II B have a serious disadvantage: they have free dynamical parameters which can be chosen at will. Of course we could fit the model parameters with some measured curves, but still it would be difficult to interprete the data, because the free parameters stem from the dynamics derivation procedure (more precisely, from temporal averaging), and not from the microscopic properties of the neurons.

Here we move in the opposite way. We start from our main Eqs. (43) and derive a closed expression for the simplest possible assembly dynamics. This results in a graded-response-like relaxation dynamics that follows smoothly and coarsely the real dynamics of the assembly. Additionally, all its parameters can be interpreted in terms of the microscopic neuronal parameters.

Since we want to gain a relaxation dynamics without delays, we assume that there is no discontinuity in the activation function $p_{\text{A}}(s)$, i.e. the neurons have a relative refractory behavior but no absolute refractory period. We start with a first-order approximation of our main Eqs. (46). We use the slow-dynamics approximation (49) [3], and chop the differential equation system at $n = 1$, so that

$$N^{(2)}(\mathbf{x}, t) \approx \kappa_h^{(2)}(\mathbf{x}) A(\mathbf{x}, t) \tag{56}$$

with $\kappa_h^{(2)}(\mathbf{x})$ calculated as specified in (50) (depending on $h(\mathbf{x}, t)$). This means that we only have two state variables, namely the activity $A(\mathbf{x}, t)$ and the first recovery variable $N^{(1)}(\mathbf{x}, t) = N_{\text{I}}(\mathbf{x}, t)$, which is the number of neurons in the inactivated state.

The number of inactivated neurons is obtained from $A(\mathbf{x}, t) = \tau[h(\mathbf{x}, t)]^{-1}[N(\mathbf{x}) - N_{\text{I}}(\mathbf{x}, t)]$ so as to give

$$N_{\text{I}}(\mathbf{x}, t) = N - \tau[h(\mathbf{x}, t)] A(\mathbf{x}, t) \ . \tag{57}$$

We now turn to the activity $A(\mathbf{x}, t)$. We assume that, for quasistationary activity, the fields evolve more slowly than the activity, and neglect the changes of $h(\mathbf{x}, t)$ [4]. This leaves us with $\mathrm{d}/\mathrm{d}t\, A(\mathbf{x}, t) \approx -\tau[h(\mathbf{x}, t)]^{-1}\, \mathrm{d}/\mathrm{d}t\, N_{\text{I}}(\mathbf{x}, t)$, and inserting Eqs. (56) and (57) into the dynamics equation (46) for $N_{\text{I}}(t)$ with exponential $p_{\text{A}}(s)$ [the same steps can be applied to the other functions $p_{\text{A}}(s)$], we arrive at

$$\begin{aligned} \tau_{\text{g}}[h(\mathbf{x}, t)] \frac{\mathrm{d}}{\mathrm{d}t} A(\mathbf{x}, t) &= -A(\mathbf{x}, t) \\ &+ \frac{1}{\tau[h(\mathbf{x}, t)]} \left\{ N(\mathbf{x}) - \tau_{\text{g}}[h(\mathbf{x}, t)] \left[1 - \frac{\kappa_h^{(2)}(\mathbf{x})}{\tau[h(\mathbf{x}, t)]} \right] A(\mathbf{x}, t) \right\} \end{aligned} \tag{58}$$

---

[3] The slow dynamics approximation now involves temporal averaging. The difference to standard graded-response models is that the averaging occurs over intrinsic neuronal time intervals, and not a priori over some arbitrary interval of length T. Therefore, it does not introduce additional dynamical parameters. Additionally, since we look at slow, or even quasistationary dynamics in this case, the temporal averaging is justified. We also remind that we can avoid temporal averaging if we use the fast dynamics approximation, resulting in a first order approximation that has the form of a differential equation with delay.

[4] Without loss of generality, we can include a term that considers the variation of $A(\mathbf{x}, t)$ due to changes of $h(\mathbf{x}, t)$, so that this assumption is not really necessary for the calculation of first-order dynamics. It is ommitted only to gain an equation that can be compared to other graded-response models.

with

$$\frac{1}{\tau_{\rm g}[h(\mathbf{x},t)]} = \frac{1}{\tau[h(\mathbf{x},t)]} + \frac{1}{\tau_{\rm ref}} \ . \tag{59}$$

Not only has this graded-response type equation the microscopically correct stationary solutions but it also provides us with the relaxation time constant $\tau_{\rm g}[h(\mathbf{x},t)]$. This means that if we are interested in a realistic quasistationary pool behavior, graded-response equations with fixed relaxation time constants as those of sections II A and II B are not sufficient.

Equation (58) is the correct way to introduce a systematically derived graded-response-type dynamics for pools of spiking neurons using the chain of differential equations. The effect of (58) is a dynamics that follows the real activity dynamics by smoothing out sharp activity peaks. Nevertheless, it will do so following the envelope curve of the activity, and it will still approach the correct stationary solutions for a constant field $h(\mathbf{x})$.

D. Higher-Order Approximations: Realistic Assembly Dynamics

Higher-order approximations serve to model in a quantitatively accurate way the dynamics of assemblies of spiking neurons. Using the fast dynamics approximation (47), the model is capable of reproducing the time-course of the activity of a pool composed of extensively many neurons, including fast transients and sharp activity peaks as those occurring when the activity approaches oscillatory solutions.

The different recovery variables serve as memory buffers for the past activity. Higher (with larger n) recovery variables are responsible for the more recent past and influence the response of the pool to fast transients. Taking only one or two recovery variables results in activities that follow the real activity in a smooth, approximated way. If we include more recovery variables, the assembly dynamics also follows the smaller details of the real activity.

Figs. 10 and 11 show simulations of transient and oscillatory pool dynamics calculated using Eq. **??**dynamics). The activity is compared with results gained from simulations using assemblies of explicitly modeled spiking neurons.

VII. CONNECTION WITH OTHER MODELS

In this section, we compare the model with other neuronal models. Specifically, we show that standard gain functions and graded-response models can be understood in terms of our pool dynamics, and that this allows us to interprete the parameters of those functions in terms of the microscopic parameters of our underlying neuronal model. Furthermore, we show that our model is equivalent to a pool of spike–response or integrate–and–fire neurons.

A. Gain Function

In section VI B, we have shown that the stationary solution of the pool dynamics is the sigmoidal gain function (54). In case we have an absolute refractory period only, $\kappa_h^{(1)}(\mathbf{x})$ vanishes, and we get an equation of the same form as the standard logistic gain function,

$$\begin{aligned} G[h(\mathbf{x})] &= \frac{1}{\gamma^{\rm abs}} \frac{1}{1+\exp\{-2\beta[h(\mathbf{x})-\theta']\}} \\ &= \frac{1}{\gamma^{\rm abs}} \frac{1}{2} \left(1+\tanh\{\beta[h(\mathbf{x})-\theta']\}\right) \ . \end{aligned} \tag{60}$$

Since $A_{\rm max} := 1/\gamma^{\rm abs}$ is the maximal spiking activity of the neurons, and normalizing the activity $A \to A/N$, we get

$$A(\mathbf{x}) = A_{\rm max} \frac{1}{2} \left(1+\tanh\{\beta[h(\mathbf{x})-\theta']\}\right) \ . \tag{61}$$

This means that for pools of spiking neurons we can use the standard logistic gain function to get realistic stationary results, and we know how each parameter of the gain function can be interpreted in terms of the microscopic parameters of the underlying neuronal model.

B. Standard Graded-Response Models

The standard graded-response models of section II A can be motivated as follows from our pool dynamics. We look at the normalized form ($A \to A/N$) of equation (45). In a quasistationary regime we define a dynamics by an exponential relaxation towards the stationary solution of (43) and (45), given by $A(\mathbf{x}) = \tau[h(\mathbf{x})]^{-1}\{1 - [\gamma^{\text{abs}} + \kappa_h^{(1)}(\mathbf{x})]A(\mathbf{x})\}$:

$$\begin{aligned}\tau\frac{\mathrm{d}}{\mathrm{d}t}A(\mathbf{x},t) &= -A(\mathbf{x},t) \\ &+\frac{1}{\tau[h(\mathbf{x},t)]}\{1 - [\gamma^{\text{abs}} + \kappa_h^{(1)}(\mathbf{x})]A(\mathbf{x},t)\} \ . \end{aligned} \tag{62}$$

This equation, and its simpler variant (for small $[\gamma^{\text{abs}} + \kappa_h^{(1)}(\mathbf{x})]A(\mathbf{x},t) \ll 1$)

$$\tau\frac{\mathrm{d}}{\mathrm{d}t}A(\mathbf{x},t) = -A(\mathbf{x},t) + \frac{1}{\tau[h(\mathbf{x},t)]} \ , \tag{63}$$

are of the same form as the assembly-averaged graded-response models presented in section II A. Eq. (62) will relax towards the *correct microscopic solutions* (i.e., solutions that are in accordance with those obtained from simulations with single spiking neurons), incorporating absolute and relative refractory effects. There is no necessity of "time-coarse-graining" or other temporal averaging procedures to arrive at equation (62) for quasistationary activity. Graded-response models as in (62) and (63) may thus present a valid approach, if the assembly dynamics are always close to the stationary state calculated from the microscopic parameters. For fast, transient, dynamics, the full differential equation system (43) is to be used instead. Again, as in the preceding section, it is now possible to understand how each parameter of the graded-response model can be interpreted in terms of the microscopic parameters of the underlying neuronal model. The only exception is the arbitrary relaxation time constant τ. For a calculation of the relaxation time constant using intrinsic neuronal parameters refer back to section VI C.

C. Graded-Response Models with Refractory Effects

We can enhance the standard graded response model (62) by incorporating an additional term for the dynamics of the first recovery variable. Together with an exponential relaxation dynamics of $A(\mathbf{x},t)$, using the approximation (48) for the number of inactivated neurons, and assuming neurons with relative refractory period only, we find

$$\begin{aligned}\tau\frac{\mathrm{d}}{\mathrm{d}t}A(\mathbf{x},t) &= -A(\mathbf{x},t) + \frac{1}{\tau[h(\mathbf{x},t)]}[1 - N_{\text{I}}(\mathbf{x},t)] \ , \\ \frac{\mathrm{d}}{\mathrm{d}t}N_{\text{I}}(\mathbf{x},t) &= A(\mathbf{x},t) - \left[\frac{1}{\tau[h(\mathbf{x},t)]} + \frac{1}{\tau_{\text{ref}}}\right] N_{\text{I}}(\mathbf{x},t) \ . \end{aligned} \tag{64}$$

This system is similar to that of section II B, Eq. (9). Neurons can be firing, inactivated (=quiescent), and activated (=refractory). Between the three states transitions are allowed, some with a field-dependent rate and others with a fixed rate. Integrating the spike density over a small fixed interval T during which $A(\mathbf{x},t)$ can be regarded as constant, we get the absolute number of neurons that released a spike recently, $a(\mathbf{x},t) \approx TA(\mathbf{x},t)$. We define further $r(\mathbf{x}) := N_{\text{I}}(\mathbf{x},t)$, $\beta := 1/\tau_{\text{ref}}$, $\sigma_1[h(\mathbf{x},t)] := (1/\tau[h(\mathbf{x},t)])\,(T/\tau)$, $\sigma_2[h(\mathbf{x},t)] := 1/\tau[h(\mathbf{x},t)]$, $\alpha_a = 1/\tau$ and $\alpha_r = 1/T$ and rewrite the equations (64) as

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}a(\mathbf{x},t) &= -\alpha_a a(\mathbf{x},t) + \{1 - r(\mathbf{x},t)\}\sigma_1[h(\mathbf{x},t)] \ , \\ \frac{\mathrm{d}}{\mathrm{d}t}r(\mathbf{x},t) &= \alpha_r a(\mathbf{x},t) - r(t)\{\beta + \sigma_2[h(\mathbf{x},t)]\} \ . \end{aligned} \tag{65}$$

The result is a system that is very similar to the model of section II B. Again, we can interprete the parameters of the model in terms of their microscopic parameters. The system now depends as before in section II B on an arbitrary integration time constant T and a relaxation time constant τ. For quantitative modeling it is therefore better to use the assembly model presented in this paper, which is based exclusively on microscopic parameters.

D. Connection with Models of Spiking Neurons

In this subsection we compare one of the most general types of model of single neuron threshold dynamics, the "spike–response model" (SRM), with the presented pool model. We show how the parameters of our pool models can be mapped to parameters of the SRM. It turns out that our pool model is exact for pools of spike–response neurons with special refractory functions. In other cases, our model can be used as an approximation.

In the SRM, the response of a neuron, say i, is determined by a total field that has two contributions: one from the synaptic inputs from other neurons and another that accounts for the neuron's refractory behavior due to the release of action potentials,

$$h_i^{\text{total}}(t) = h_i(t) + h_i^{\text{ref}}(t) \ . \tag{66}$$

The neuron fires deterministically or with a certain probability, if the total field reaches a fixed threshold from below.

The *synaptic input field* $h_i(t)$ is usually defined using an alpha function as in Eq. (14) of section III B. The *refractory field* $h_i^{\text{ref}}(t)$ is defined by a *refractory function* $\eta_i(s)$. For spike trains of a neuron i, $S_i(t) = \sum_f \delta(t - t_i^f)$, we have

$$\begin{aligned} h_i(t) &= \sum_j J_{ij} \sum_f \alpha_{ij}(t - t_j^f) \\ &= \sum_j J_{ij} \int_0^\infty \mathrm{d}s \, \alpha_{ij}(s) S_j(t-s) \ , \\ h_i^{\text{ref}}(t) &= \sum_f \eta_i(t - t_i^f) \\ &= \int_0^\infty \mathrm{d}s \, \eta_i(s) S_i(t-s) \ . \end{aligned} \tag{67}$$

In this paper, we consider neurons with renewal. This means that only the last spike at t_i^* accounts for refractory effects and thus contributes to $h_i^{\text{ref}}(t)$,

$$h_i^{\text{ref}}(t) = \eta_i(t - t_i^*) \ . \tag{68}$$

The $\alpha(s)$ and $\eta(s)$ functions [5] of the spike–response model can be used to model a broad range of types of neuronal models. For example, it is possible to express the so-called "integrate–and–fire" (I&F) type models in terms of special functions $\alpha(s)$ and $\eta(s)$.

Using the total field of the SRM, we introduce an exponential total spike probability density $1/\tau_{\text{SRM}}$ for neurons with renewal [11],

$$\begin{aligned} \tau_{\text{SRM}}[h_i(t), h_i^{\text{ref}}(t)] &= \tau_0 \exp\{-2\beta[h_i(t) + h_i^{\text{ref}}(t) - \theta]\} \\ &= \tau_0 \exp\{-2\beta[h_i(t) + \eta(t - t_i^*) - \theta)]\} \ . \end{aligned} \tag{69}$$

Then we can identify the spike probability density for activated neurons and the activation probability for refractory neurons from section III A with

$$\{\tau[h_i(t)]\}^{-1} := \tau_0^{-1} \exp\{2\beta[h_i(t) - \theta]\} \tag{70}$$

and

$$p_{\text{A}}(t - t_i^*) := \exp\{2\beta\eta(t - t_i^*)\} \ . \tag{71}$$

This means that the differential equation system for the pool dynamics (43) is exact in the limit of pools composed of extensively many spike–response neurons with renewal and refractory functions $\eta(s)$ of the form

$$\eta(s) = \frac{1}{2\beta} \ln[p_{\text{A}}(s)] \tag{72}$$

---

[5] For the sake of simplicity, we will drop the neuron indices i and j of the α and η-functions from here on.

with $p_{\rm A}(s)$ being one of the activation functions presented in section V B. Figure 7 shows an example of the exponential $p_{\rm A}(s)$ and the corresponding refractory function calculated through (72).

Alternatively, we can start from frequently used refractory functions $\eta(s)$ and search for systematic approximations of these function through the corresponding $p_{\rm A}(s)$. This is for example the case for I&F neurons, which use an exponential $\eta(s)$. Two of the most frequently used refractory functions are

$$\eta_{\rm exp}(s) = \begin{cases} 0 & \text{for } s < 0 \\ -\infty & \text{for } 0 \le s < \gamma^{\rm abs} \\ -\eta_0 \exp[-\frac{s-\gamma^{\rm abs}}{\tau_\eta}] & \text{for } s \ge \gamma^{\rm abs} \end{cases} \tag{73}$$

and

$$\eta_{\rm inv}(s) = \begin{cases} 0 & \text{for } s < 0 \\ -\infty & \text{for } 0 \le s < \gamma^{\rm abs} \\ -\frac{\tau_\eta}{s-\gamma^{\rm abs}} & \text{for } s \ge \gamma^{\rm abs} \end{cases} . \tag{74}$$

For small $\eta(s)$, i.e., in the case that the synaptic field is small enough so that neurons do not spike again until their refractory field has already decreased considerably, we can approximate the activation function $p_{\rm A}(s)$ in (72) corresponding to the refractory function (73) by

$$\begin{aligned} p_{\rm A}(t-t^*) &= \exp[2\beta\eta(t-t^*)] \\ &\approx 1 + 2\beta\eta(t-t^*) = 1 - 2\beta\eta_0 \exp\left[-\frac{s-\gamma^{\rm abs}}{\tau_\eta}\right] . \end{aligned} \tag{75}$$

Comparing this with the exponential activation function or the sigmoidal activation function in (29), we get

$$p_0 = 2\beta\eta_0, \ \tau_{\rm ref} = \tau_\eta, \text{ and } s_0 = \gamma^{\rm abs} . \tag{76}$$

Similarly, in the case of the inverse refractory function (74) we can approximate

$$\begin{aligned} p_{\rm A}(t-t^*) &= \exp\{2\beta\eta(t-t^*)\} \\ &\approx 1 + 2\beta\eta(t-t^*) = 1 - 2\beta\frac{\tau_\eta}{s-\gamma^{\rm abs}} \end{aligned} \tag{77}$$

and compare this with the inverse activation function [last equation of (29)] so as to get

$$\tau_{\rm ref} = 2\beta\tau_\eta, \text{ and } s_0 = \gamma^{\rm abs} . \tag{78}$$

Figure 8 shows an exponential refractory function as used for I&F neurons and its approximation in terms of $p_{\rm A}(s)$. We see that, for large s, the curves coincide. This means that specially in undercritical synaptic driving conditions, during which the synaptic input is much smaller than the highest amplitude of the refractory field, the presented approximation scheme should allow for a precise quantitative description of the activity of pools composed of stochastic SRM or I&F neurons.

Of course any other approximation scheme can be used as well. This allows us to simulate pools of neurons with different refractory fields by means of the model presented in this paper.

VIII. STABILITY AND OSCILLATIONS

In this section, we analyze the stability problem concerning assembly dynamics and present a novel stability criterion that is well-suited to handle pools of neurons.

A. Non-Stationary Activity

The exact correspondence between the dynamics generated by a chain of differential equations and that of pools of spike–response type neurons (although restricted to special refractory functions) allows for a simple derivation of some known analytical results. Two points are of special interest. First, the stability of a pools' state of stationary activity is relevant to the capability of a pool to develop coherent oscillations. A stability analysis for spike–response neurons

has been worked out by Gerstner et al. [11]. Simulations with the differential equation system confirm the stability conditions calculated analytically for the spike–response model (SRM). In passing we remind that integrate–and–fire neurons constitute a special case of the SRM.

Second, the conditions for the existence of stable coherent oscillations have been stated in the so-called "locking theorem" for the noise-free case [12]. In the noise-free case, a neuron i spikes exactly when its total field reaches a fixed threshold θ from below, i.e., when

$$h_i(t) + h_i^{\text{ref}}(t) - \theta = 0 \ . \tag{79}$$

The locking theorem states that the activity of a pool of SRM neurons with renewal has a stable oscillatory solution, if all neurons fire according to their threshold condition (79) while their synaptic field is increasing in time. This is illustrated by Fig. 9.

The differential equation pool model presented in this paper uses stochastic neurons, and thus does not apply in the noise-free limit. Nevertheless, we can approximate the noise-free case to any accuracy. We therefore expect that the stability conditions stated in the locking theorem are applicable to our pool model as well. We will see in the next subsection, however, that if we stick to the integral Eq. (26) instead of using the differential equation system (43), we can perform the noise-free limit and prove the locking theorem for our dynamics. To what extent the locking theorem can be applied to the noisy case and to the case of an approximated dynamics (43) (with a limited number of recovery variables) is a question that still remains open.

In Fig. 11, we show a simulation of a single pool of SRM neurons using our differential equation model and compare it with the pool activity calculated according to Eq. (26). The pool is coupled reciprocally with itself (i.e., the coupling strength J_{ij} between any two pool neurons i and j is the same) and the parameters of the synaptic and the refractory fields fulfill the conditions of the locking theorem. The simulation shows that a small perturbation grows until the pool activity shows a marked oscillation. It also demonstrates the good agreement between the dynamics according to (26) and (43). A similarly good quantitative and qualitative agreement as in Fig. 11 is found between our pool model data and microscopically modeled pools (not shown).

B. Conditions for Locking and Oscillatory Activity

To understand what happens with the pool dynamics in the low-noise limit, we will return to the original integral equation (26) for the activity $A(\mathbf{x},t)$. Because of Eq. (22), we see that the number of inactivated neurons $N_{\text{I}}(\mathbf{x},t)$ (24) can also be expressed by

$$\begin{aligned} N_{\text{I}}(\mathbf{x},t) &= N(\mathbf{x}) - \int_{-\infty}^{t} \mathrm{d}t^*\, p_{\text{A}}(t-t^*)\, D_h(\mathbf{x},t,t^*) A(\mathbf{x},t^*) \\ &= N(\mathbf{x}) - \int_{0}^{\infty} \mathrm{d}s\, p_{\text{A}}(s)\, D_h(\mathbf{x},t,t-s) A(\mathbf{x},t-s) \ . \end{aligned} \tag{80}$$

Using this in Eq. (26) we immediately get

$$\begin{aligned} A(\mathbf{x},t) &= \int_{-\infty}^{t} \mathrm{d}t^*\, F_h(\mathbf{x},t,t^*) A(\mathbf{x},t^*) \\ &= \int_{0}^{\infty} \mathrm{d}s\, F_h(\mathbf{x},t,t-s) A(\mathbf{x},t-s) \end{aligned} \tag{81}$$

with the firing probability at time t,

$$\begin{aligned} F_h(\mathbf{x},t,t^*) &= -\frac{\mathrm{d}}{\mathrm{d}t} D_h(\mathbf{x},t,t^*) \\ &= \frac{1}{\tau[h(\mathbf{x},t)]}\, p_{\text{A}}(t-t^*) D_h(\mathbf{x},t,t^*) \ . \end{aligned} \tag{82}$$

This is the integral-equation form for the activity of a pool of spiking neurons as it has been presented by [11]. Eq. (81) is equivalent to Eq. (26) for any finite β. We will use Eq. (81) here to explain the low-noise limit of our pool dynamics.

With the same equations as in section VII D for $\tau[h(\mathbf{x},t)]$ (70) and $p_{\text{A}}(t-t^*)$ (71), taking the low-noise limit for spike–response neurons means $\beta \to \infty$. Near to this limit, the great majority of the neurons will spike when their total field $h^{\text{total}}(t)$ (66) gets close to the threshold θ. For a continuous synaptic field $h(\mathbf{x},t)$ and neurons with renewal,

this is equivalent to saying that the time s elapsed since their last firing will be close to the "ideal" time $s^*(\mathbf{x},t)$ defined implicitly by the noise-free threshold condition

$$h(\mathbf{x},t) + h^{\mathrm{ref}}(t) - \theta = h(\mathbf{x},t) + \eta(s^*) - \theta = 0 \ . \tag{83}$$

Because of the spiking of the neurons, the survival function $D_h(\mathbf{x},t,t-s)$ (20) will present a sharp drop from 1 to 0 for $s > s^*(\mathbf{x},t)$. At the same time, we have $\{\tau[h(\mathbf{x},t)]\}^{-1}\, p_{\mathrm{A}}(s) \approx 0$ for $s < s^*(\mathbf{x},t)$. Therefore, the spiking probability $F_h(\mathbf{x},t,t-s)$ is nearly zero everywhere with the exception of the region where $s \approx s^*(\mathbf{x},t)$. In the low-noise limit, the maximum of $F_h(\mathbf{x},t,t-s)$ diverges to $+\infty$, and the location of the maximum converges towards $s = s^*(t)$. In addition, we see from (82) that $F_h(\mathbf{x},t,t-s)$ is normalized over t, since

$$\begin{aligned}\int_{-\infty}^{\infty} \mathrm{d}t\, F_h(\mathbf{x},t,t^*) &= \int_{t^*}^{\infty} \mathrm{d}t\, F_h(\mathbf{x},t,t^*) \\ &= -\int_{t^*}^{\infty} \mathrm{d}t \left[\frac{\mathrm{d}}{\mathrm{d}t} D_h(\mathbf{x},t,t^*)\right] = 1 \ .\end{aligned} \tag{84}$$

Taking advantage of these properties of $F_h(\mathbf{x},t,t^*)$, we choose in the limit $\beta \to \infty$ the following firing probability function

$$F(\mathbf{x},t,t-s) = f_0\, \delta[s - s^*(\mathbf{x},t)] \ . \tag{85}$$

Using the normalization property of the firing probability function we know that

$$\int_{t^*}^{\infty} \mathrm{d}t'\, F(\mathbf{x},t',t^*) = f_0 \int_{t^*}^{\infty} \mathrm{d}t'\, \delta[t' - t^* - s^*(\mathbf{x},t)] = 1 \ . \tag{86}$$

From this equation we get

$$f_0 = \left| 1 - \frac{\mathrm{d}}{\mathrm{d}t'} s^*(\mathbf{x},t') \big|_{t'-t^* = s^*(\mathbf{x},t')} \right| \tag{87}$$

and thus, together with (81) for the activity dynamics,

$$A(\mathbf{x},t) = \left| 1 - \frac{\mathrm{d}}{\mathrm{d}t} s^*(\mathbf{x},t) \right| \ A[\mathbf{x}, t - s^*(\mathbf{x},t)] \ , \tag{88}$$

with the implicitly defined $s^*(\mathbf{x},t)$. This expression is valid for the activity of a pool of equivalent neurons with renewal in the noise-free case.

We will elaborate conclusion (88) a bit further. A linearization of $s^*(\mathbf{x},t)$ in a small interval of length Δt around t_0, during which the synaptic field can be regarded as constant, $h(\mathbf{x},t) \approx h(\mathbf{x},t_0)$, and during which we can invert the refractory field function so that $s^*(\mathbf{x},t) = -\eta^{-1}[h(\mathbf{x},t) - \theta]$, directly gives [6] $s^*(\mathbf{x},t_0 + \Delta t) = s^*(\mathbf{x},t_0) - h'(\mathbf{x},t_0)/\eta'(\mathbf{x},t_0)\, \Delta t$ (the primes denote time derivatives), so that we find for the activity

$$A(\mathbf{x},t_0+\Delta t) = \left| 1 + \frac{h'(\mathbf{x},t_0)}{\eta'[s^*(\mathbf{x},t_0)]} \right| \ A\left(\mathbf{x}, t_0 - s^*(\mathbf{x},t_0) + \left[1 + \frac{h'(\mathbf{x},t_0)}{\eta'[s^*(\mathbf{x},t_0)]}\right] \Delta t\right) \ . \tag{89}$$

Let us now consider a pool that is only coupled to itself. Starting with a constant activity, a small perturbation at time $t_{-1} = t_0 - s^*(t_0)$ causes a further increase/decrease of the perturbation at the next spike-time at t_0 if the factor $1 + h'(t_0)/\eta'[s^*(\mathbf{x},t_0)]$ is greater/smaller than 1. Since for monotonous $\eta(s)$ it is $\eta'[s^*(\mathbf{x},t_0)] > 0$, this requirement is fulfilled, if the synaptic field $h(\mathbf{x},t)$ caused by the perturbation has a *positive* slope at time t_0. Thus, an increasing synaptic field at the time of spiking caused by the perturbation is a sufficient condition for the instability of the state of constant activity.

Similarly, a pool that has already developed an oscillatory activity, say, with narrow activity peaks at times $t_{-1}, t_0, \ldots$ will present a contraction of its activity peak and at the same time an increase of the activity maximum, if the synaptic field has a positive slope at the activity peak times. This can be seen by rewriting Eq. (89) as

---

[6] $\eta'[\eta^{-1}(x)](\eta^{-1})'(x) = 1 \Rightarrow (\eta^{-1})'(x) = \{\eta'[\eta^{-1}(x)]\}^{-1}$

$$A(\mathbf{x}, t_0 + \Delta t_0) = c(t_0)\, A\left(\mathbf{x}, t_{-1} + \Delta t_{-1}\right) \ , \tag{90}$$

with the compression factor

$$c(t_0) := \left| 1 + \frac{h'(\mathbf{x}, t_0)}{\eta'[s^*(\mathbf{x}, t_0)]} \right| \ , \tag{91}$$

the past spiking time

$$t_{-1} := t_0 - s^*(\mathbf{x}, t_0) \tag{92}$$

and the difference to the last spiking time

$$\Delta t_{-1} := c(t_0)\, \Delta t_0 \ . \tag{93}$$

The neurons that contributed to the activity at $t_{-1} + \Delta t_{-1}$ now contribute to the activity at $t_0 + \Delta t_0$. For $c(t_0) > 1$, however, the activity at $t_0 + \Delta t_0$ is larger than the activity at $t_{-1} + \Delta t_{-1}$. This growth goes hand in hand with a contraction of the activity peak, because for $c(t_0) > 1$ we get

$$\Delta t_0 < \Delta t_{-1} \ . \tag{94}$$

This means that neurons that where delayed by Δt_{-1} with respect to the oscillatory peak at t_{-1} present a smaller delay Δt_0 at the new peak at t_0. They are therefore "pulled" back into the oscillatory peak, i.e., they lock. Otherwise, the time difference to the oscillatory peak gets larger and the neurons fire more asynchronously, i.e., the oscillatory peak broadens and the coherence decreases.

The condition $c(t_0) > 1$ is identical to the condition stated by the locking theorem [12]. It is a sufficient condition to determine if a pool of noise-free neurons has a stable solution in form of an oscillatory activity. Equation (88) is a more general form of the locking theorem, and can be used to directly calculate the time course of the activity of a pool.

IX. FINITE-SIZE POOLS AND THE CENTRAL-LIMIT THEOREM

The dynamics represented by Eqs. (26) and (43) is valid under the assumption that there exist extensively many pool neurons for each interval $(t^* - \Delta t^*, t^*]$. For *finite* pool sizes, Eqs. (26) and (43) are valid for the *mean* values of the activity and the recovery variables. It should be asked, then, how noise influences the pool dynamics since the strong law of large numbers does not suffice any more and, because of finite-size effects, noise has to be taken into account by the central limit theorem and variations thereof [18]. This is necessary for understanding stability criteria of a pool's activity, for the estimation of the number of neurons that compose a pool, or for comparison of the presented pool dynamics with microscopically modeled pools. In this section, we present a "cooking recipe" for calculating the variance of the pool activity. This variance can be used afterwards for a realistic simulation of finite-size pools.

Consider a single pool $\mathbf{x}$. At time t, there are $\bar{n}(\mathbf{x}, t, t^*) = n(\mathbf{x}, t, t^*)\Delta t^*$ neurons that have spiked for the last time during the interval $(t^* - \Delta t^*, t^*]$. Since all these neurons feel the same refractory field, the present firing probability during $(t - \Delta t, t]$ is the same, and equals

$$\begin{aligned} &\text{Prob}\{i \text{ fires in } [t, t + \Delta t) \text{ due to field } h_i\} \\ &= \frac{\Delta t}{\tau[h(\mathbf{x}, t)]}\, p_{\text{A}}(t - t^*) \ . \end{aligned} \tag{95}$$

The probability that $n_{\text{s}}(\mathbf{x}, t, t^*)$ of these $\bar{n}(\mathbf{x}, t, t^*)$ neurons emit a spike during the interval $(t^* - \Delta t^*, t^*]$ is then given by the binomial distribution (the stochastic variable x_{t^*} is the number of spiking neurons)

$$\begin{aligned} &\text{Prob}\{x_{t^*} = n_{\text{s}}(\mathbf{x}, t, t^*)\} \\ &= \binom{\bar{n}(\mathbf{x}, t, t^*)}{n_{\text{s}}(\mathbf{x}, t, t^*)} p^{n_{\text{s}}(\mathbf{x}, t, t^*)} q^{\bar{n}(\mathbf{x}, t, t^*) - n_{\text{s}}(\mathbf{x}, t, t^*)} \end{aligned} \tag{96}$$

with $p = \Delta t\, \tau[h(\mathbf{x}, t)]^{-1}\, p_{\text{A}}(t - t_i^*)$ and $q = 1 - p$.

The mean number of firing neurons of the subgroup of $\bar{n}(\mathbf{x}, t, t^*)$ neurons during the interval $[t - \Delta t, t)$ can then be calculated as

$$\begin{aligned}\langle x_{t^*}\rangle(\mathbf{x},t,t) &= \bar{n}(\mathbf{x},t,t^*)\,p \\ &= \frac{\Delta t}{\tau[h(\mathbf{x},t)]}\,p_{\mathrm{A}}(t-t^*)\,n(\mathbf{x},t,t^*)\,\Delta t\ ,\end{aligned} \tag{97}$$

and for the variance of the number of firing neurons we get

$$\begin{aligned}\sigma^2(\mathbf{x},t,t^*) &= \bar{n}(\mathbf{x},t,t^*)\,p\,q \\ &= \langle x_{t^*}\rangle(\mathbf{x},t,t^*)\left[1-\frac{\Delta t}{\tau[h(\mathbf{x},t)]}\,p_{\mathrm{A}}(t-t^*)\right]\ .\end{aligned} \tag{98}$$

To calculate the number of spiking neurons of the entire pool, we have to consider the sum of the stochastic variables from the subgroups characterized by their last firing times t^*,

$$X = \sum_{t^*} x_{t^*}\ . \tag{99}$$

According to the central limit theorem, the probability distribution function of the stochastic variable X has a mean that can be calculated as the sum of the means of the single stochastic variables x_{t^*},

$$\langle X\rangle(\mathbf{x},t) = \sum_{t^*}\langle x_{t^*}\rangle(\mathbf{x},t,t^*)\ . \tag{100}$$

This gives

$$\begin{aligned}\langle X\rangle(\mathbf{x},t) &= \sum_{t^*}\frac{\Delta t}{\tau[h(\mathbf{x},t)]}\,p_{\mathrm{A}}(t-t^*)\,n(\mathbf{x},t,t^*)\,\Delta t^* \\ &= A(\mathbf{x},t)\Delta t\ .\end{aligned} \tag{101}$$

Therefore, for a pool of finite size, our calculation of the activity $A(\mathbf{x},t)$ [using Eqs. (26) or (43)] is equivalent to the calculation of the mean number (or the *expectation value*) of neurons that emit a spike at time t.

Similarly, the central limit theorem states that the variance of the probability distribution function of the stochastic variable X is equal to the sum of the variances of the single stochastic variables x_{t^*},

$$\sigma^2(\mathbf{x},t) = \sum_{t^*}\sigma^2(\mathbf{x},t,t^*)\ . \tag{102}$$

Together with $p_{\mathrm{A}}^2 = (1-p_{\mathrm{A}})^2 - 2(1-p_{\mathrm{A}}) + 1$ this leads to the result

$$\begin{aligned}\sigma^2(\mathbf{x},t) &= \sum_{t^*}\langle x_{t^*}\rangle(\mathbf{x},t,t^*)\left[1-\frac{\Delta t}{\tau[h(\mathbf{x},t)]}\,p_{\mathrm{A}}(t-t^*)\right] \\ &= A(\mathbf{x},t)\Delta t - \sum_{t^*}\frac{\Delta t^2}{\tau[h(\mathbf{x},t)]^2}\,p_{\mathrm{A}}(t-t^*)^2 \\ &= A(\mathbf{x},t)\Delta t\left[1-\frac{\Delta t}{\tau[h(\mathbf{x},t)]}\right] \\ &\quad -\frac{\Delta t^2}{\tau[h(\mathbf{x},t)]^2}\left[N^{(1)}(\mathbf{x},t)-N^{(2)}(\mathbf{x},t)\right]\end{aligned} \tag{103}$$

where $N^{(1)}(\mathbf{x},t) = N_{\mathrm{i}}(\mathbf{x},t)$ and $N^{(2)}(\mathbf{x},t)$ have been defined in section V D.

We see that, for small discretization time intervals of length Δt [and neglecting the terms of the order $(\Delta t)^2$], we have a relative width of the probability distribution function

$$\frac{\sigma(\mathbf{x},t)}{\langle X\rangle(\mathbf{x},t)} = \frac{1}{\sqrt{A(\mathbf{x},t)\Delta t}}\ . \tag{104}$$

In other words, the signal-to-noise ratio increases for higher activity. This is important for oscillatory pool activity with high activity peaks. In this case the effect of the noise induced by finite-size pool effects during the high activity peaks is reduced considerably.

For simulations of pools with a *finite* number of neurons, we can now calculate the mean activity and the recovery variables as before, and then assume for the activity a Gaussian probability distribution function (central limit theorem) with a variance calculated according to the last lines of Eq. (103).

X. DISCUSSION

The differential equation model (43) presents many advantages over pools of explicitly modeled spiking neurons. First of all, for large pools the numerical cost is reduced. E.g., the simulations of Fig. 11 were calculated using the same sampling stepsize of 0.5 ms (the differential equation system additionally used an adaptive-stepsize integration method in these 0.5 ms intervals). The simulation times of the pool of spiking neurons and the differential equation pool model were 1451.5 s resp. 15.5 s. In our implementation, the numerical cost of both types of simulations gets comparable for pools with less than 70 neurons. It can also be seen from Fig. 5 that the recovery kernels decay very fast, so that usually only a few recovery variables are needed to describe a pool's activity quantitatively well.

A second advantage results from the fact that, for simulations, the differential equation system with delay requires the past activity of the system to be remembered only up to a point in the past specified by the longest delay present in the system. Typical values for the axonal delay Δ^{ax} during the synaptic transmission are $1-5$ ms, and the length of the absolute refractory period γ^{abs} is usually < 10 ms. Furthermore, there is also the possibility to reduce the pool dynamics to a differential equation system without delays, which has been discussed in sections V A and V D and which completely eliminates the system's functional dependence upon the past activity.

In summary, starting from a stochastic single-neuron threshold model with renewal, we have derived a system of differential equations with or without delays that describe the activity dynamics of a pool or assembly of *equivalent* neurons. Contrary to previous derivations of differential equation pool dynamics from microscopic models, the derivation is exact for any dynamical range. This means that the model can operate equally well in the near-stationary condition and when fast, transient, dynamics is required. For numerical simulations, the real behavior of the pool is approximated by breaking the chain of differential equations at the desired level. The chain of differential equations allows to move gradually from a crude approximation of the real pool dynamics (corresponding to a graded-response approach) towards a biologically realistic dynamics of a pool of spiking neurons.

The model serves as a basis for understanding the more heuristic graded-response type models in terms of microscopic (i.e., neuronal), dynamics parameters. Furthermore, it is exact in the case of pools composed of extensively many spiking neurons of the spike–response or integrate–and–fire type. Simulations show good quantitative agreements of the resulting pool activity with the activity of pools modeled using spiking neurons. It is also shown that analytical results from the microscopic models are applicable to the presented model. Pools modeled by our dynamics show the capability of developing oscillatory behavior in the parameter regimes predicted by the locking theorem of spike–response neurons. This is shown both in simulations and by presenting a proof of the locking theorem for our pool dynamics. Finally, it is explained how the model can account for finite-size effects.

The key advantage of the model presented in this paper is that it relies upon macroscopic (i.e., pool-averaged) parameters but retains many biologically relevant neuronal parameters that are subject to experimental observation. In summary, it closes an existing gap between the microscopic and the macroscopic neuronal modeling levels.

Acknowledgements
Thanks are due to Armin Bartsch for stimulating discussions and helpful comments. This work has been supported by the Deutsche Forschungsgemeinschaft (DFG) under grants number He 1729/7-2 and 8-1.

APPENDIX A: MAIN NOTATIONAL DEFINITIONS

Notation	Definition
$S_i(t) = \sum_f \delta(t - t_i^f)$	Spike train, with spikes f released by neuron i
$A(\mathbf{x},t) = \sum_{j\in\mathbf{x}} S_j(t)$	Activity (spike density) of pool $\mathbf{x}$ with neurons j
Δ^{ax}	Synaptic axonal delay
$\alpha(s) = \Theta(s-\Delta^{\text{ax}}) \frac{(s-\Delta^{\text{ax}})^k}{c_k} \exp\left[-\frac{(s-\Delta^{\text{ax}})}{\tau_\alpha}\right]$	Synaptic alpha function (normalization const c_k)
$h_i(t) = \sum_{j=1}^N J_{ij} \int_0^\infty \mathrm{d}s\, \alpha(s)\, S_j(t-s)$	Synaptic field on neuron i, coupling weights J_{ij}
$h(\mathbf{x},t) = \sum_{\mathbf{y}} J(\mathbf{x},\mathbf{y}) \int_0^\infty \mathrm{d}s\, \alpha(s)\, A(\mathbf{y},t-s)$	Synaptic field on any neuron of pool $\mathbf{x}$, pool-to-pool coupling weights $J(\mathbf{x},\mathbf{y})$
$D_h(\mathbf{x},t,t^*)$	Probability that a pool $\mathbf{x}$ neuron that spiked last at time t^* did not spike again until t
$\rho(\mathbf{x},t,t^*) = D_h(\mathbf{x},t,t^*)A(\mathbf{x},t^*)$	Momentary density of pool $\mathbf{x}$ neurons that spiked last at time t^*
$\langle f\rangle(\mathbf{x},t) = \int_{-\infty}^t \mathrm{d}t^*\, \rho(\mathbf{x},t,t^*)\, f(\mathbf{x},t,t^*)$	Function f averaged over all pool $\mathbf{x}$ neurons
$[\tau(h)]^{-1}$	Firing probability density for neurons that are in the activated state
γ^{abs}	Length of the absolute refractory period
$p_{\text{A}}(s)$	Activation function, probability that a neuron that spiked last at $t-s$ is in the activated state
τ_{ref}, p_0 and s_0	Time constant and parameters of $p_{\text{A}}(s)$
$N^{(0)}(\mathbf{x}) = \langle[1-p_{\text{A}}(t-t^*)]^0\rangle$ $N^{(1)}(\mathbf{x},t) = \langle[1-p_{\text{A}}(t-t^*)]^1\rangle$ $N^{(\infty)}(\mathbf{x},t) = \langle[1-p_{\text{A}}(t-t^*)]^\infty\rangle$	Recovery variables
$N(\mathbf{x}) = N^{(0)}(\mathbf{x})$	Total number of pool neurons
$N_{\text{I}}(\mathbf{x},t) = N^{(1)}(\mathbf{x},t)$	Number of inactivated pool neurons
$M(\mathbf{x},t) = N^{(\infty)}(\mathbf{x},t)$	Last recovery variable

APPENDIX B: ASSEMBLY DYNAMICS - MAIN RESULTS

1. Dynamics of the Synaptic Field

Assumptions: $k \in \mathbb{N}$ in $\alpha(s)$, $h(\mathbf{x},t) = h^{(k)}(\mathbf{x},t)$. Dynamics for $l \in \mathbb{N}$, $0 \le l \le k$:

$$\boxed{\begin{array}{l} \frac{\mathrm{d}}{\mathrm{d}t} h^{(l)}(\mathbf{x},t) = l\frac{c_{l-1}}{c_l} h^{(l-1)}(\mathbf{x},t) - \frac{1}{\tau_\alpha} h^{(l)}(\mathbf{x},t) \\ \frac{\mathrm{d}}{\mathrm{d}t} h^{(0)}(\mathbf{x},t) = \sum_{\mathbf{y}} J(\mathbf{x},\mathbf{y})\, A(\mathbf{y},t-\Delta^{\text{ax}}) - \frac{1}{\tau_\alpha} h^{(0)}(\mathbf{x},t) \end{array}}$$

2. Dynamics of the Activity and Recovery Variables

Assumptions: $p_{\text{A}}(s)$ monotonous and differentiable except at $s = \gamma^{\text{abs}}$. Without absolute refractory period, the terms with $M(\mathbf{x},t)$ and $p_{\text{A}}(\gamma^{\text{abs}})$ vanish. Dynamics for $m \le 1 < \infty$:

$$\boxed{\begin{array}{l} A(\mathbf{x},t) = \frac{1}{\tau[h(\mathbf{x},t)]}[N(\mathbf{x}) - N^{(1)}(\mathbf{x},t)] \\ \frac{\mathrm{d}}{\mathrm{d}t} N^{(m)}(\mathbf{x},t) = A(\mathbf{x},t) - \{1-[1-p_{\text{A}}(\gamma^{\text{abs}})]^m\} A(\mathbf{x},t-\gamma^{\text{abs}}) - \frac{1}{\tau[h(\mathbf{x},t)]}[N^{(m)}(\mathbf{x},t) - N^{(m+1)}(\mathbf{x},t)] \\ \qquad - \begin{cases} \frac{m}{\tau_{\text{ref}}}[N^{(m)}(\mathbf{x},t) - M(\mathbf{x},t)] & \text{exp. } p_{\text{A}}(s) \\ \frac{m}{\tau_{\text{ref}}}\{N^{(m)}(\mathbf{x},t) - M(\mathbf{x},t) - [N^{(m+1)}(\mathbf{x},t) - M(\mathbf{x},t)]/p_0\} & \text{sigm. } p_{\text{A}}(s) \\ \frac{m}{\tau_{\text{ref}}}[N^{(m+1)}(\mathbf{x},t) - M(\mathbf{x},t)] & \text{inv. } p_{\text{A}}(s) \end{cases} \\ \frac{\mathrm{d}}{\mathrm{d}t} M(\mathbf{x},t) = A(\mathbf{x},t) - A(\mathbf{x},t-\gamma^{\text{abs}}) \end{array}}$$

3. Approximation Schemes

Implementation of the dynamics by breaking the chain and approximation of $N^{(n+1)}$:

Approximation	Dynamical regime and assumptions
$\frac{\mathrm{d}}{\mathrm{d}t}N^{(n+1)}(\mathbf{x},t) \approx A(\mathbf{x},t) - A(\mathbf{x},t-\gamma^{\mathrm{abs}})$	For fast, transient dynamics. Without absolute refractory period it is $\mathrm{d}/\mathrm{d}t\, N^{(n+1)}(\mathbf{x},t) \equiv 0$.
$N^{(n+1)}(\mathbf{x},t) \approx [\gamma^{\mathrm{abs}} + \kappa_h^{(n+1)}(\mathbf{x})]A(\mathbf{x},t)$	For slow dynamics. $A(\mathbf{x},t)$ must be approximately constant during the past time $(t-s,t]$ during which $[1-p_{\mathrm{A}}(s)]^{n+1}$ is large. Without absolute/relative refractory period it is $\gamma^{\mathrm{abs}} = 0$ resp. $\kappa_h^{(n+1)}(\mathbf{x}) = 0$

---

[1] J. D. Cowan. In: volume 3 of *Advances in Neural Information Processing Systems*, p 62, San Mateo, 1991. Morgan Kaufmann.

[2] R. Eckhorn, R. Bauer, W. Jordan, M. Brosch, W. Kruse, M. Munk, and H. J. Reitboeck. Coherent oscillations: A mechanism of feature linking in the visual cortex? *Biol. Cybern.*, 60:121–130, 1988.

[3] A. K. Engel, P. König, A. K. Kreiter, and W. Singer. Interhemispheric synchronization of oscillatory neural responses in cat visual cortex. *Science*, 252:1177–1179, 1991.

[4] A. K. Engel, P. König, and W. Singer. Direct physiological evidence for scene segmentation by temporal coding. *Proc. Natl. Acad. Sci. USA*, 88:9136–9140, 1991.

[5] G. B. Ermentrout and J. D. Cowan. A mathematical theory of visual hallucination patterns. *Biol. Cybern.*, 34:137–150, 1979.

[6] G. B. Ermentrout and J. D. Cowan. Temporal oscillations in neuronal nets. *J. Math. Biol.*, 7:265–280, 1979.

[7] G. B. Ermentrout and J. D. Cowan. Large scale spatially organized activity in neural nets. *SIAM J. Appl. Math.*, 38:1–21, 1980.

[8] D. Fargue. Réductibilité des systèmes héréditaires à des systèmes dynamiques. *Compt. Rend. Acad. Sci.*, B(277):471–473, 1973.

[9] D. Fargue. Réductibilité des systèmes héréditaires. *Int. J. of Non-Linear Mechanics*, 9:331–338, 1974.

[10] J. L. Feldman and J. D. Cowan. Large-scale activity in neural nets I: Theory with application to motoneuron pool responses. *Biol. Cybern.*, 17:29–38, 1975.

[11] W. Gerstner and J. L. van Hemmen. Coding and information processing in neural networks. In E. Domany, J. L. van Hemmen, and K. Schulten, editors, *Models of Neural Networks II*, pages 1–93, Berlin, 1994. Springer.

[12] W. Gerstner, J. L. van Hemmen, and J. D. Cowan. What matters in neuronal locking? *Neural Comput.*, 8(8):1653–1676, 1996.

[13] C. M. Gray, P. König, A. K. Engel, and W. Singer. Oscillatory responses in cat visual cortex exhibit inter-columnar synchronization which reflects global stimulus properties. *Nature*, 338:334–337, 1989.

[14] C. M. Gray and W. Singer. Stimulus-specific neuronal oscillations in orientation columns of cat visual cortex. *Proc. Natl. Acad. Sci. USA*, 86:1698–1702, 1989.

[15] D. O. Hebb. *The Organization of Behavior*. Wiley, New York, 1949.

[16] J. Hertz, A. Krogh, and R. G. Palmer. *Introduction to the theory of neural computation*. Addison-Wesley, Redwood City CA, 1991.

[17] W. M. Kistler, R. Seitz, and J. L. van Hemmen. Modelling collective excitations in cortical tissue. *Physica D*, 114:273–295, 1998.

[18] J. Lamperti. *Probability*. Benjamin, New York, 1966. Chapter 7.

[19] T. Ohira and J. D. Cowan. Master-equation approach to stochastic neurodynamics. *Physical Review E*, 48(3):2259–2266, 1993.

[20] H. C. Tuckwell. *Introduction to Theoretical Neurobiology*. Cambridge University Press, Cambridge, 1988.

[21] J. L. van Hemmen and R. Kühn. Collective phenomena in neural networks. In E. Domany, J. L. van Hemmen, and K.Schulten, editors, *Models of Neural Networks.*, pp 1–105, Berlin, 1991. Springer.

[22] D. Wang, J. Buhmann, and C. von der Malsburg. Pattern segmentation in associative memory. *Neural Comput.*, 2:94–106, 1990.

[23] H. R. Wilson and J. D. Cowan. Excitatory and inhibitory interactions in localized populations of model neurons. *Biophys. J.*, 12:1–24, 1972.

[24] H. R. Wilson and J. D. Cowan. A mathematical theory of the functional dynamics of cortical and thalamic nervous tissue. *Kybernetik*, 13:55–80, 1973.

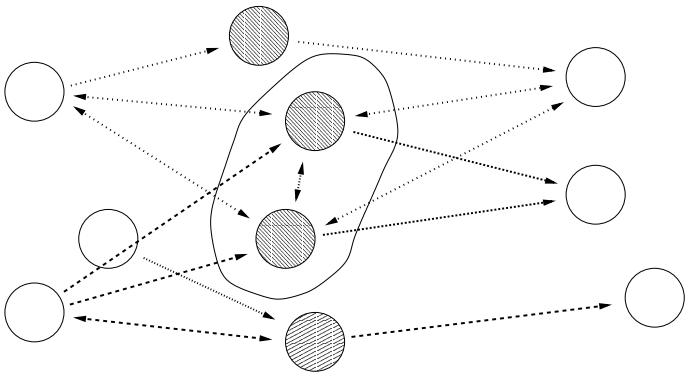

FIG. 1. The notion of a "pool" or "assembly" of neurons is often encountered when dealing with large-scale biological neural networks. It has originally been introduced by Hebb [15]. In this paper, neurons belonging to the same pool or assembly are characterized by having the same input/output connectivity pattern. Furthermore, all neurons of the same pool have the same parameters. (A very similar concept is that of a "sublattice"; see [21] §1.2.4, which arises in relation to associative networks. It contains the key to the pool idea in that a sublattice has been defined implicitly as all neurons being identical and having the same input.) In the figure, different types of neurons and connections are characterized by different textures (white neurons are of any type). According to the assembly definitions, only the two neurons in the oval belong to the same pool.

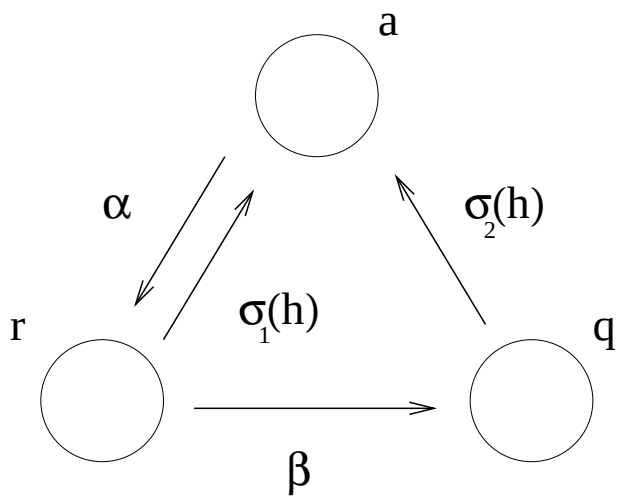

FIG. 2. A graded-response model including refractory effects, according to [1]. In this model, the neurons of a single pool can be grouped into three subpopulations: neurons that fired recently (a), neurons that are in a relative refractory state (r) and neurons that are quiescent (q). Neurons from a decay with a rate α towards the state r, and neurons from r decay with a rate β towards the state q. A synaptic field h induces field-dependent transitions from the refractory (r) or the quiescent (q) subpopulation towards the firing (a) subpopulation with rates $\sigma_1(h)$ and $\sigma_2(h)$, respectively. With an appropriate a-dependent field h, a single recursively coupled pool can generate sustained oscillations.

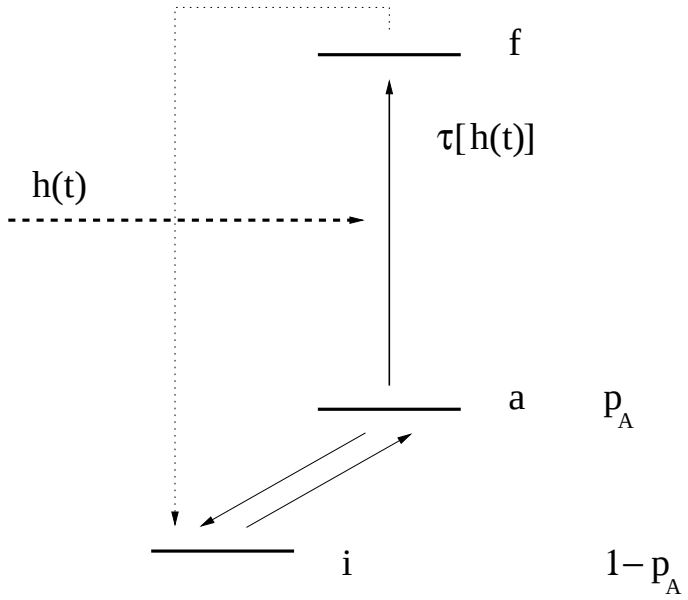

FIG. 3. Definition of the microscopic model of a spiking neuron. In our description, a neuron can be in either of three states: inactivated (**i**), activated (**a**), or firing (**f**). Transitions between the three states are allowed between the **i** and the **a** levels (fast, with transition rates that depend on the last spike time t^*), from the **a** to the **f** level [slow, with a transition rate that depends on the synaptic input field $h(t)$] and from the **f** back to the **i** level (fast). The mean occupation of the **a** level is given by the *activation probability* $p_{\rm A}(t-t^*)$. Refractoriness means the neuron bounces back and forth between the **i** and the **a** level, with $1 > p_{\rm A}(t-t^*) \geq 0$. A neuron can only release a spike if it is activated (**a**). The *firing probability for activated* neurons in a time interval of length Δt is field-dependent and equal to $\Delta t/\tau[h(t)]$. After firing, t^* is reset and the neuron is inactivated.

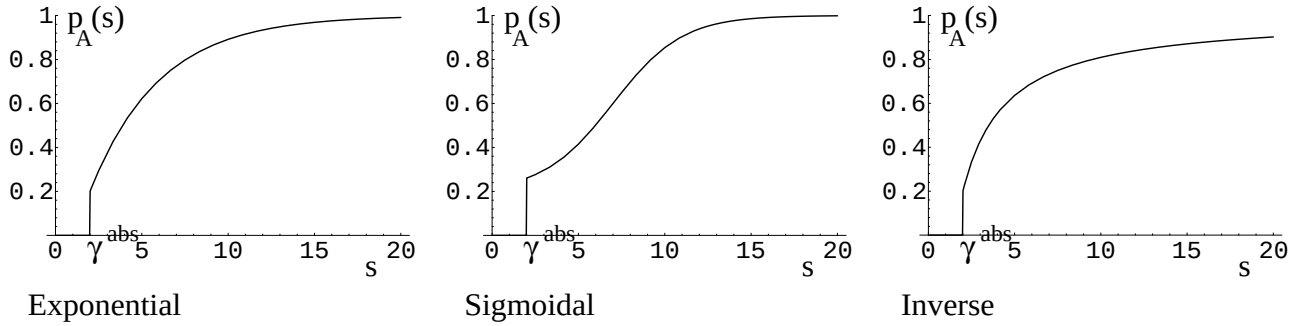

FIG. 4. The different activation functions $p_{\rm A}(t-t^*)$ used for the derivation of the pool dynamics. At $s=\gamma^{\rm abs}$, the function may have, and here has, a discontinuity. For $t \searrow t^*$, the neuron is in an absolute refractory state because it has just spiked and $p_{\rm A}(t-t^*) \searrow 0$. For $t \to \infty$, the refractory effects vanish and $p_{\rm A}(t-t^*) \to 1$. In case of activation functions with a discontinuity, $\gamma^{\rm abs}$ is the length of the absolute refractory period.

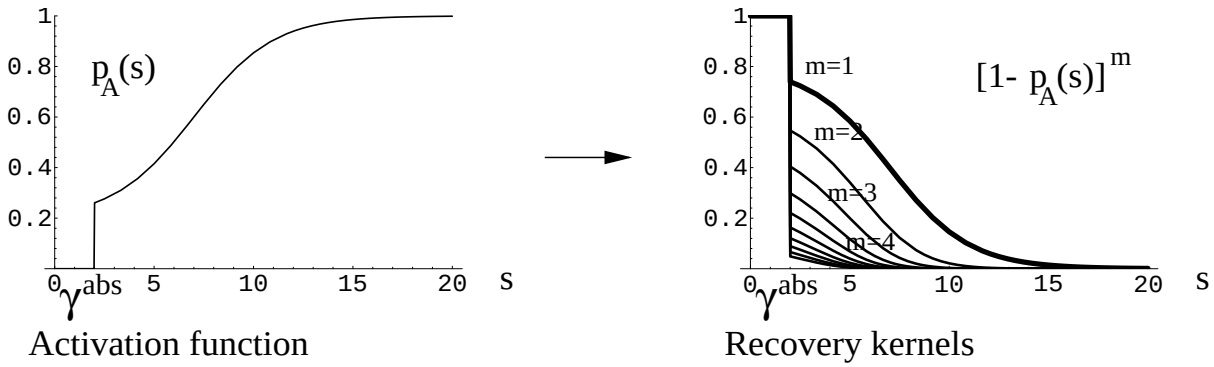

FIG. 5. The activation function $p_A(s)$ is shown with its recovery kernels $[1 - p_A(s)]^m$. With growing m, the kernels include less and less of the past time s. The pool dynamics is expressed with the help of a series of recovery variables $N^m(\mathbf{x}, t) := \langle [1 - p_A(t - t^*)]^m \rangle$ calculated by computing the pool average of the recovery kernel function.

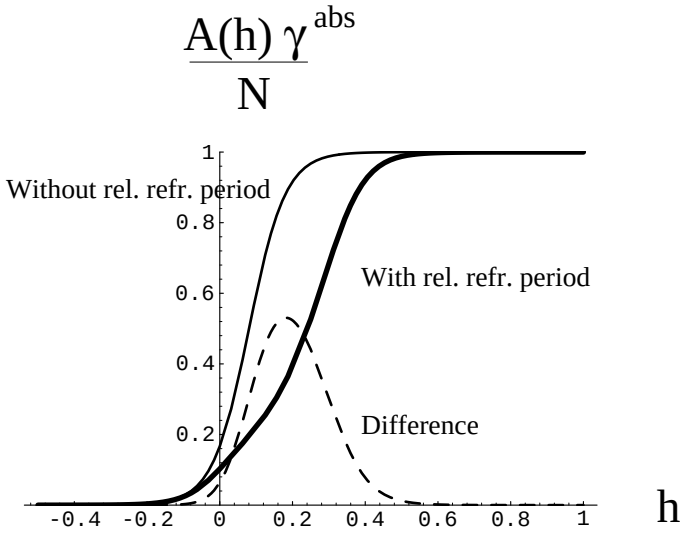

FIG. 6. The stationary spike density for a pool of spiking neurons which receives a constant synaptic input field h can be expressed by a gain function that is similar to the logistic gain function. Here we show the gain function for a pool of neurons with absolute refractory period only (thin solid line) and for a pool with absolute and relative refractory period (thick solid line). The relative refractory period reduces the activity for fields h close to the threshold (dashed line, difference between the gain function without and the gain function including relative refractory effects).

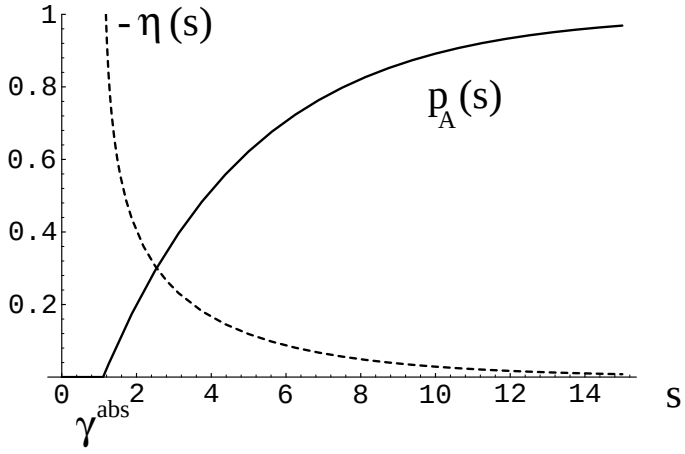

FIG. 7. Correspondence between the activation function $p_A(s)$ (solid line) and the negative refractory function $-\eta(s)$ of the SRM (dotted line). In this case, we used an exponential $p_A(s)$ with an absolute refractory period of $\gamma^{abs} = 3$. At $s = \gamma^{abs}$, the refractory function $\eta(s)$ diverges to $-\infty$.

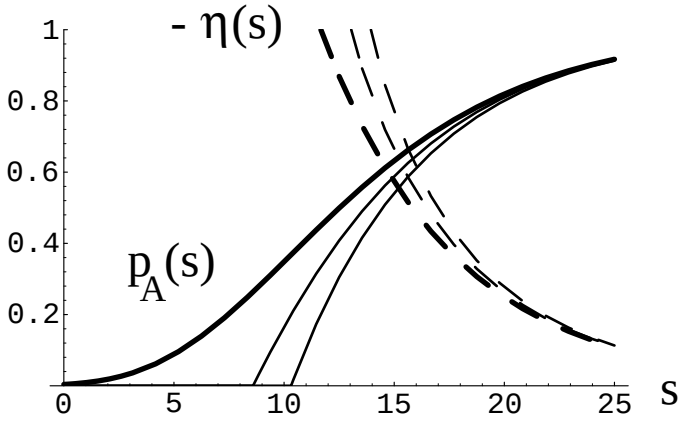

FIG. 8. The exponential refractory field function $-\eta(s)$ (dashed thick line) is plotted together with its corresponding activation probability function $p_A(s)$ (solid thick line). The other four functions are approximations of the desired refractory function (dashed thin line) and the desired activation function (solid thin line) using a sigmoidal (better fit of the thin curves) and an exponential $p_A(s)$. This approximation is particularly suitable for undercritical stimulation conditions since the curves coincide for large s, i.e., when neurons spike again after their refractory field has already decreased noticeably.

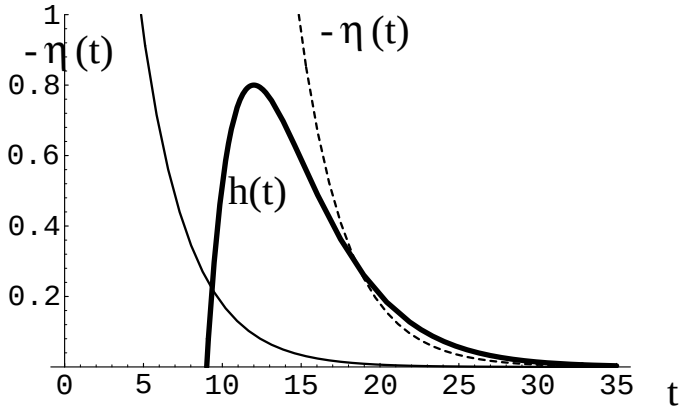

FIG. 9. Locking theorem for pools of deterministic (i.e., noise-free), equivalent spike–response neurons with $\theta = 0$. Two cases, corresponding to pools with different refractory functions $\eta(s)$ are illustrated. Let us assume that all neurons fire exactly at the same moment $t = 0$. Then the refractory field $h^{\mathrm{ref}}(t)$ of all neurons evolves according to their refractory function $\eta(t)$ [solid thin line for one pool and dotted thin line for the other pool; $-\eta(t)$ is shown]. The locking theorem states that, if the threshold condition $h(t) + \eta(t) - \theta = 0$ is fulfilled at a *rising* synaptic field h, an oscillatory solution for the pool activity is stable; otherwise it is unstable. Therefore, in the figure, the pool with the refractory field indicated by the solid thin line has a stable oscillatory solution, whereas the oscillatory activity of the pool with the dotted thin line will decay ($\theta = 0$ in this figure).

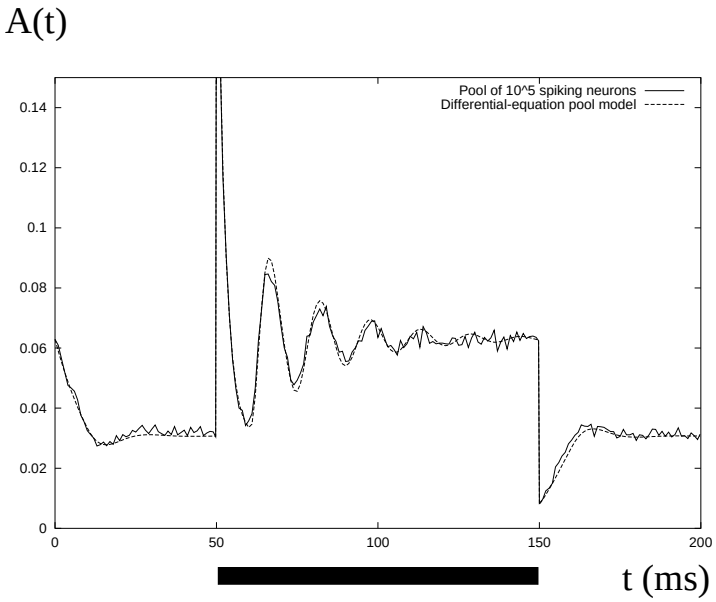

FIG. 10. Simulation of the activity $A(t)$ of a single pool of neurons without couplings, using spike–response neurons (solid line) or Eqs. (43) (dashed line). From $50 - 150$ ms, a constant external field is applied (black bar). The sudden onset of the external field evokes a sharp activity peak which decays in a damped oscillation towards the new stationary state. We have used the fast dynamics approximation with $n = 4$ recovery variables and fast approximation $N^{(4)}(t) \approx N^{(\infty)}(t)$, meaning $\mathrm{d}/\mathrm{d}t\, N^{(4)}(t) = A(t) - A(t - \gamma^{\mathrm{abs}})$, and fifth-order Runge-Kutta integration with adaptive stepsize. The simulations show a good quantitative agreement.

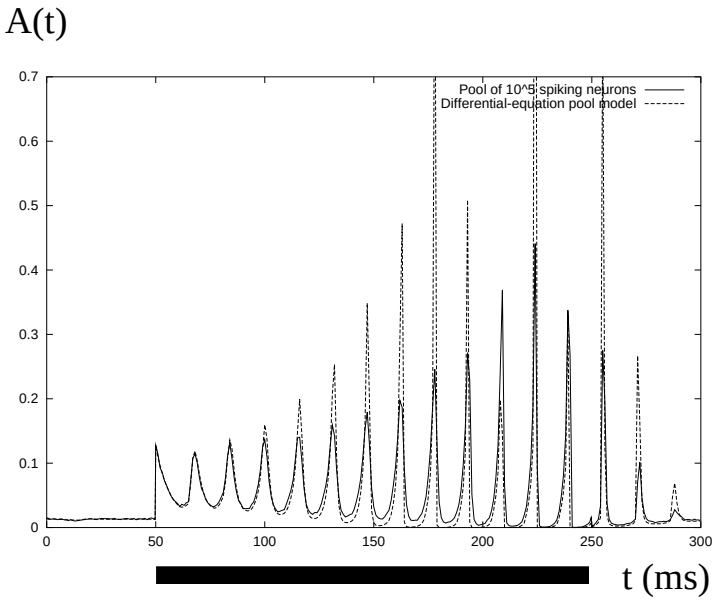

FIG. 11. Simulation of the activity $A(t)$ of a single pool of reciprocally coupled neurons, using spike–response neurons (solid line) or Eqs. (43) (dashed line). From $50 - 250$ ms, a constant external field is applied (black bar). The parameters of the pool neurons fulfill the locking theorem. The onset of the external field evokes a small activity peak, which grows and generates a self-sustained oscillation. As in the previous simulation, we have used the fast dynamics approximation with $n = 4$. The simulations show a good quantitative agreement, except in the tips of the activity peaks (finite-size effects).