LETTER Communicated by Wulfram Gerstner

Event-Driven Simulation of Spiking Neurons with Stochastic Dynamics

Jan Reutimann
jan.reutimann@cns.unibe.ch
Michele Giugliano
michi@cns.unibe.ch
Stefano Fusi
fusi@cns.unibe.ch
Computational Neuroscience, Institute of Physiology, University of Bern, 3012 Bern, Switzerland

We present a new technique, based on a proposed event-based strategy (Mattia & Del Giudice, 2000), for efficiently simulating large networks of simple model neurons. The strategy was based on the fact that interactions among neurons occur by means of events that are well localized in time (the action potentials) and relatively rare. In the interval between two of these events, the state variables associated with a model neuron or a synapse evolved deterministically and in a predictable way. Here, we extend the event-driven simulation strategy to the case in which the dynamics of the state variables in the inter-event intervals are stochastic. This extension captures both the situation in which the simulated neurons are inherently noisy and the case in which they are embedded in a very large network and receive a huge number of random synaptic inputs. We show how to effectively include the impact of large background populations into neuronal dynamics by means of the numerical evaluation of the statistical properties of single-model neurons under random current injection. The new simulation strategy allows the study of networks of interacting neurons with an arbitrary number of external afferents and inherent stochastic dynamics.

1 Introduction

Most of the observed in vivo cortical phenomena are believed to be the expression of the collective dynamics of large populations of interacting neurons. The richness and the complexity of these phenomena call for powerful new techniques for investigating in a systematic way the emergent behavior of large networks of neurons. In this respect, computer simulations are becoming an increasingly important tool to model realistic cortical modules (10^4–10^5 neurons and 10^7–10^9 synapses), to relate single-neuron and synapse properties to the collective dynamics of large networks and to

check the validity of the numerous assumptions underlying the theories of neural networks as dynamical systems.

Traditional fixed time-step (synchronous) computer simulations are based on numerical methods for solving a set of coupled differential equations that describe the dynamics of the basic elements of the networks (the neurons and the synapses). The time axis is uniformly discretized, and each state variable is updated at every time step. Under such conditions, the algorithmic complexity usually scales with the number of elements contained in the network, multiplied by the number of time steps into which the simulated interval is divided. Such an approach also sets a cutoff on the temporal resolution of the simulation (the minimal time step), which is imposed on all the state variables, no matter whether they are characterized by completely different timescales. This represents an obvious drawback when slow and fast dynamic variables coexist, as all of them are updated at the very same pace, determined by the fastest variable. When the number of neurons and synapses approaches realistic numbers for a cortical column ($\sim 10^5$), the traditional techniques become impracticable, and new alternative simulation strategies should be devised.

One possible solution takes natural inspiration from the way the neurons exchange information and interact: most of the dynamical variables are coupled for relatively short time intervals only, during the emission and the transmission of action potentials. The emission of a spike is a relatively rare event (the typical cortical frequencies being of the order of several Hz) and well localized in time (i.e., the width of a spike is of the order of 1 ms) with respect to the overall simulation times. Interestingly, it can be observed that within the interval between two consecutive relevant events (e.g., spikes as in our case, although our approach can be generalized to other classes of events), the neurons and the synapses are essentially isolated, and each state variable is usually described by a deterministic, uncoupled differential equation. Therefore, given the initial state at the beginning of the interval and the length of the interval, the final state can be determined without iteratively updating the state variable at every time step. This approach is usually known as *event driven*; it constitutes the fundamental principle of modern operating systems and has already been applied successfully to large-scale simulations of networks of spiking neurons (Watts, 1994; Delorme, Gautrais, van Rullen, & Thorpe, 1999; Giugliano, 2000; Mattia & Del Giudice, 2000). The CPU time of these simulations depends mostly on the frequency of occurrence of relevant events, not on the time simulated. If the relevant events are rare, as would be the spikes in the cortex, then the event-driven approach reduces the computational loads dramatically.

One drawback of this strategy manifests itself when the neuronal dynamics does not evolve deterministically between the emission of two successive spikes—for example, when the network to be studied is embedded in a very large population of neurons that provide an external, noisy background activity. This spike activity is usually meant to emulate the sensory

inputs or, in general, the inputs coming from other areas of the brain (Amit & Brunel, 1997a; Wang, 1999). This interaction, at least on a first approximation, is modeled as a one-way connection (feedforward input) in the sense that the local simulated activity of the network of neurons does not affect the background activity that represents the external input. Each simulated (internal) neuron receives synaptic inputs from a group of M_{ext} external neurons. The external population is assumed to be large, and the overlap among the external groups corresponding to different internal neurons is usually small. Hence, the total synaptic currents generated by the external neurons and injected into different internal neurons can be considered statistically independent. These inputs provide the external stimulation and allow the network to sustain the spontaneous activity observed in cortical recordings (Amit & Brunel, 1997a).

In the case of the event-based simulation approach of Mattia and Del Giudice (2000), such an external activity was implemented by generating, for each of the N neurons in the simulated networks, a large number of additional random presynaptic spike events, fired on average at a frequency ν_{ext} by each of the M_{ext} neurons of the external background populations. Such hypotheses give rise to an additional algorithmic complexity that can be quantified as $\mathcal{O}(N \cdot \nu_{ext} \cdot M_{ext})$. As a consequence, the increased realism introduced in the simulated neuronal dynamics, by appropriately increasing the size of the individual background populations for each unit of the network, has to be paid in terms of a considerable portion of CPU loads, substantially slowing the event-driven approach. Moreover, as the size of the external population increases, the external spikes can very quickly saturate the list of events that have to be handled by the simulator, and the inter-event interval can be as short as a few nanoseconds (it scales as $\mathcal{O}(1/(N \cdot \nu_{ext} \cdot M_{ext}))$). In particular, such short time intervals can easily produce roundoff errors (see Press, Flannery, Teukolsky, & Vetterling, 1986) and require double-precision floating-point variables to encode the time of occurrence of the events.

In this article, we propose a possible solution to these problems. If the synaptic background activity is irregular, as in the in vivo phenomena to be modeled, the impact of the external neurons on each simulated neuron of the network can be approximated by a continuous random current injection with appropriate statistical properties, directly related to ν_{ext} and M_{ext} (Gerstein & Mandelbrot, 1964; Tuckwell, 1988). Furthermore, by numerically solving the master equation that governs the density distribution of the state variables, it is possible to predict the behavior of each neuron under such a noisy stimulating current, during those time intervals when no synaptic input (relevant event) from the internal simulated neurons occurs. The same approach can be applied to the case in which the state variables are inherently stochastic because of some source of noise related to the single-neuron dynamics.

Here we focus on an example in which the dynamics are deterministic and the simulated neurons are embedded in a very large network. If the

(excitatory or inhibitory) postsynaptic potentials evoked by a single presynaptic spike are small compared to the threshold for triggering postsynaptic action potentials (i.e., diffusion approximation), the density equation can be reduced to a simpler Fokker-Planck equation, which in many cases can be solved analytically (Ricciardi, 1977). These theoretical approaches have already proved to constitute a powerful tool for studying network properties in stationary conditions, such as in studies on spontaneous activity, on the persistent stimulus-selective delay activity observed in the cortex (Amit & Brunel, 1997b; Yakovlev, Fusi, Berman, & Zohary, 1998; Wang, 1999; Fusi & Mattia, 1999; Brunel, 2000), or to study the network response properties to particular transient or oscillatory inputs (for a review, see Knight, 1972; Nykamp & Tranchina, 2000; Gerstner & Kistler, 2002). By combining these strategies with an event-driven simulation approach, we show in this article how to implement the global background activity, effectively reducing its computational complexity to $\mathcal{O}(N \cdot \nu_{ext})$.

2 The Simulation Algorithm

2.1 The General Idea. In the event-driven approach described in Mattia and Del Giudice (2000), the dynamics of the neurons and the synapses were updated only on the arrival of a spike. Therefore, the action potential was the relevant event to trigger an update of the state variables. The new state was then computed on the basis of two elements: (1) the initial state, following the previous update, and (2) the time passed since the previous update. Because the dynamics of both synapses and neurons was deterministic in this interval, the final state could be determined for an arbitrary time interval by the solution of the differential equations governing the dynamics of the neurons.

In our case, the development of the neuronal state variables in the interspike interval is not deterministic. In particular, the external spikes are not explicitly treated as synaptic events. Instead, we assume that the external activity is highly irregular and the total synaptic current coming from outside can be described as a stochastic process with a known distribution. In a traditional fixed time-step simulation, at each step the external current should be randomly generated, independently for each neuron, and the state variables affected by it should be updated. Therefore, the dynamics of the state variables would be nondeterministic, and the final state could not be determined by the initial condition and the time since the previous update. However, given the distribution of the external current, it is possible to determine the distribution of the state variables as a function of time. It is as if each state variable starts from some determined initial value at the preceding update time (when the distribution is a delta function) and then evolves along all the possible trajectories determined by the different realizations of the input currents. This packet of trajectories would evolve until another relevant event occurs. At this

time, the packet should be reduced to a single trajectory, which is the one that has actually been followed by the neuron. The advantage is that it is not necessary to integrate iteratively, and presumably many times, the state variables during the interval between two relevant events. Deciding at the end what was the actual trajectory is completely equivalent to what would be obtained in a fixed time-step approach, provided that the trajectory is chosen according to the correct distribution of the state variables.

2.2 An Example: Networks of Integrate-and-Fire Neurons. The approach for the proposed new event-driven algorithm is quite general and can be applied to a large class of neuronal models and synaptic interactions (see also Mattia & Del Giudice, 2000; Giugliano, 2000). In order to illustrate the algorithm, we will focus on a network of randomly connected integrate-and-fire (IF) model neurons, whose state is fully determined by a single internal variable, the membrane potential V. In section 4, we show how to generalize this approach to more complex models.

Formally, the differential equation that governs the dynamics of the membrane voltage below threshold for a generic neuron can be written as follows,

$$\frac{dV}{dt} = -L(V,t) + I_{int}(t) + I_{noise}(t) \qquad V_{\min} \le V < V_{\vartheta}, \tag{2.1}$$

where $L(V,t)$ is a generic leakage term. We will show two examples: (1) the case in which $L(V,t) = (V_{rest} - V)/\tau_m$, where V_{rest} is the resting potential and τ_m is the passive membrane time constant, and (2) the case in which the leak is constant: $L(V,t) = \lambda$. The two models respond in the same way to noisy currents provided that the dynamics of the neuron with the constant leakage are complemented by the condition that the membrane voltage is limited from below by a rigid barrier at V_{rest}. Interestingly, the stationary response function of these two model neurons to noisy currents can be fitted to the response of cortical pyramidal cells measured in in vitro experiments (Rauch, La Camera, Lüscher, Senn, & Fusi, 2002).

$I_{int}(t)$ represents the net charging current generated by synaptic inputs from other neurons within the simulated network, and $I_{noise}(t)$ is the noisy current that in our case represents the external background activity, coming from the feedforward neurons. We assume that each internal neuron receives the external input from a different group of M_{ext} external neurons. If V reaches a threshold V_{ϑ}, the neuron emits a spike, and V is reset to a value $V_{reset} < V_{\vartheta}$, where it remains during an absolute refractory period τ_{arp}. A rigid (reflecting) barrier $V_{\min}$ was included to restrict the membrane potential to the interval $V_{\min} \le V(t) < V_{\vartheta}$. Although unusual compared to the IF dynamics described in the literature, such a choice is reasonable, as the physiological range for V in a real neuron is bounded by the ionic concentration ratios across the membrane.

If the spikes are point processes (i.e., their time width is zero), $I_{int}(t)$ can be written as a sum of Dirac deltas,

$$I_{int}(t) = \sum_k J_k \delta(t - t_k),$$

where the sum extends over all the afferent spikes. The efficacy of each spike is given by J_k. In the interval between two successive synaptic inputs, $I_{int}(t)$ is 0, and V follows a random walk induced by the noisy input $I_{noise}(t)$. If the external spikes are statistically independent, the total noisy current impinging on a generic neuron of the network can be well approximated by a Gauss-distributed, delta-correlated stochastic process characterized by a mean μ and a variance σ^2 per unit time (Amit & Tsodyks, 1991; Gerstein & Mandelbrot, 1964; Tuckwell, 1988). μ and σ^2 are explicitly related to the mean firing rate ν_{ext} of the external population, its probability of feedforward connectivity c_{ext}, its size M_{ext}, and the average synaptic strength J_{ext}, according to the following equations:

$$\mu = \nu_{ext} \cdot c_{ext} \cdot M_{ext} \cdot J_{ext} \qquad \sigma^2 = \nu_{ext} \cdot c_{ext} \cdot M_{ext} \cdot {J_{ext}}^2. \tag{2.2}$$

As a consequence, the temporal evolution of the membrane potential V in the intervals between two consecutive internal spikes can be regarded as a stochastic process whose statistical properties are completely described by $P_V(V, t)$, representing the probability density of having $V(t) = V$ (Ricciardi, 1977; Tuckwell, 1988). Given that the neuron starts from $V(t_0) = V_0$ at time t_0 (the time of the last relevant event), $P_V(V, t)$ describes the packet of depolarization trajectories generated by the external input when the neuron had a particular initial value V_0. Before the arrival of the next spike emitted by one of the network neurons, it might happen that the external synaptic current drives the voltage across the threshold V_ϑ. These crossings occur at any time t at a rate $P_{fpt}(t_0, V_0; t)$, which expresses the probability density of having a first passage time at t, given the initial state $V = V_0$ at time t_0. These probabilities can be computed numerically by solving the diffusion equations that govern $P_V(V, t)$ (see appendix A). A solution is shown in Figure 1, where the temporal evolution of the probability density for the V and the corresponding interspike interval distribution are plotted. Note that there is an overall decrease of the distribution in time, because the fraction of neurons that cross the firing threshold are eliminated (we are interested in the first passage time only). As apparent from the inspection of the interspike interval distribution, there is at any time a nonzero probability of crossing the firing threshold at $V_\vartheta = -50$ mV.

2.3 The Extended Event-Driven Algorithm. Unlike in the fully deterministic case, where the state of a neuron is influenced only by events having taken place before the current time, we now have to consider the possibility of having uncertain events—predictions about the neuron's future that may or may not happen. By drawing a random number according to the

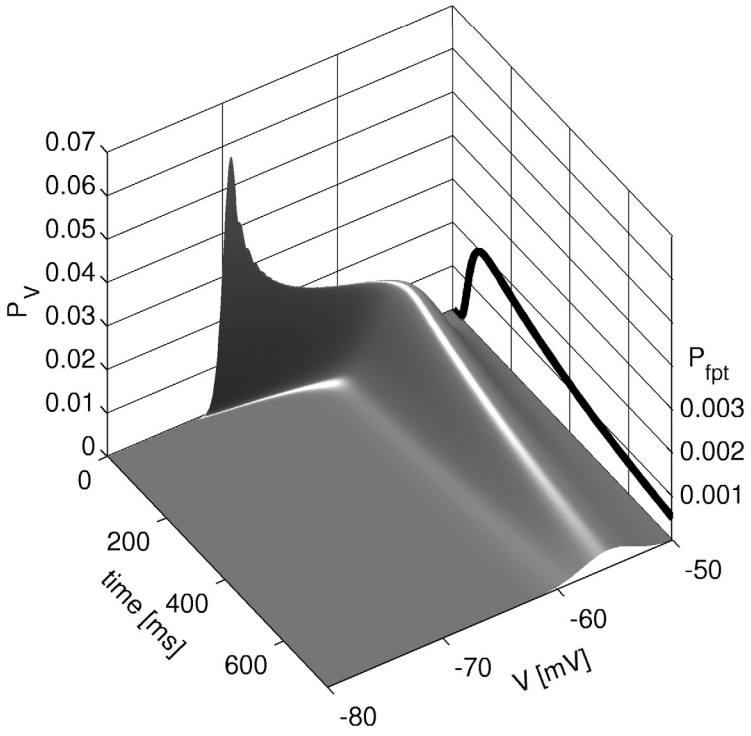

Figure 1: Numerical solution of the Fokker-Planck equation for the linear IF neuron (see appendix A): the temporal evolution of the probability density for the V (surface plot) and the interspike interval distribution (thick line) are shown. Neuron parameters: $\tau_m = 20$ ms, $V_{rest} = -70$ mV, $V_\vartheta = -50$ mV, $V_{\min} = -80$ mV, $\tau_{arp} = 2$ ms, $V_0 = -70$ mV, $\mu\tau_m = 14.5$ mV, and $\sigma^2\tau_m = 10.2$ mV^2.

probability distribution given by P_{fpt}, we obtain the predicted time (T_1) of a first threshold crossing. However, if the state of the neuron is influenced in any way before T_1, a new prediction has to be made based on the current state, and the old prediction becomes irrelevant.

To describe in some detail our algorithm, we consider a generic neuron, whose initial condition is $V(t_0) = V_0$ (see Figure 2), embedded in a network. First, we draw a pseudorandom number according to the distribution given by $P_{fpt}(t_0, V_0; T_1)$, which yields time T_1 as the predicted time of the next threshold crossing according to our neuron model (see Figure 2, A1). Two possible situations might now occur, and they can be outlined as follows:

1. No presynaptic (internal) spike arrives in the time interval $[t_0, T_1]$ from other neurons within the network. At time $t = T_1$, the neuron emits a spike (as determined in the first step) and enters the absolute

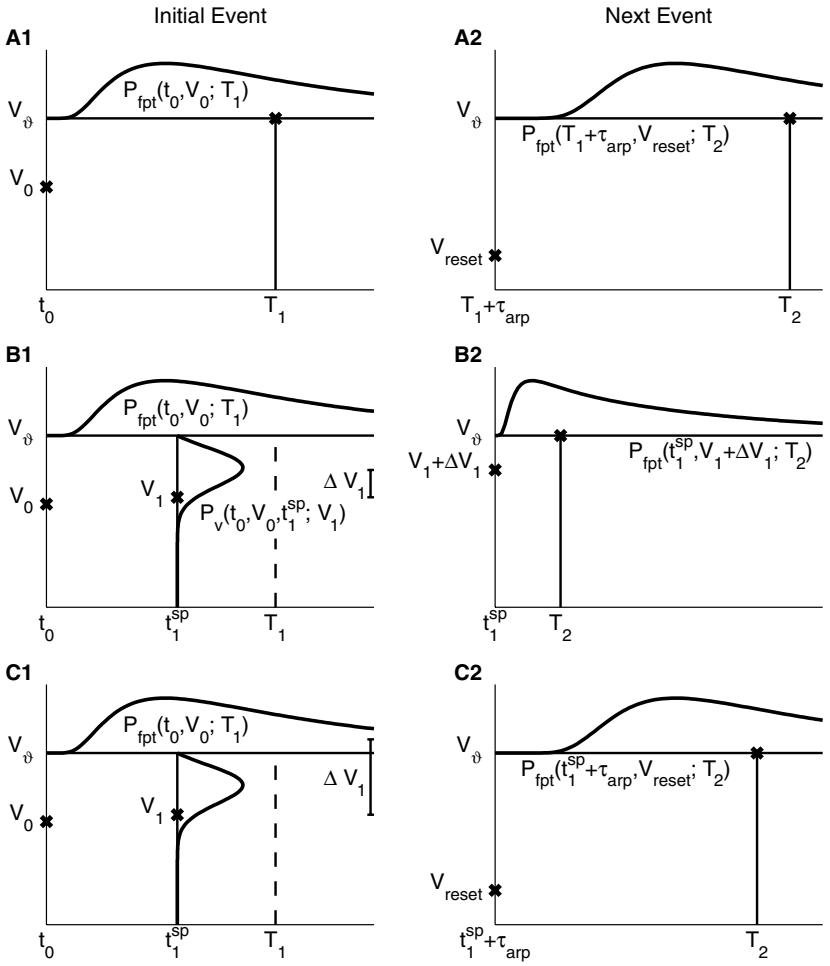

Figure 2: An illustration of the extended algorithm (see section 2.3). Column 1: Initial event in three different situations. Column 2: Corresponding next event. Initial voltage is $V(t_0) = V_0$ for all cases (A1, B1, C1). (A1) First-passage-time T_1 is determined randomly according to $P_{fpt}(t_0, V_0)$. As no further event occurs, a spike is emitted at time T_1. (A2) After the absolute refractory period τ_{arp}, the voltage is set to V_{reset}, and the next first-passage-time T_2 is determined according to $P_{fpt}(T_1+\tau_{arp}, V_{reset}; T_2)$. (B1) After T_1 has been determined (as in *A1*), a synaptic event occurs at time t_1^{sp} ($t_0 < t_1^{sp} < T_1$). The current membrane voltage $V_1 = V(t_1^{sp})$ is chosen randomly according to $P_V(t_0, V_0, t_1^{sp})$ and subsequently updated by the synaptic contribution ΔV_1. T_1 is now obsolete. (B2) Since the membrane voltage remains below threshold, the next first-passage-time T_2 is drawn according to $P_{fpt}(t_1^{sp}, V_1 + \Delta V_1; T_2)$. (C1) Same as *B1*, but ΔV_1 is large enough to cause a threshold crossing and emission of a spike. (C2) After the refractory period, the next first-passage-time is determined randomly according to $P_{fpt}(t_1^{sp}+\tau_{arp}, V_{reset})$.

refractory period, after which the neuron continues at the first step using $V_0^{new} = V_{reset}$ and $t_0^{new} = T_1 + \tau_{arp}$ (see Figure 2, A2). The spike emission leads to the creation of further events, which represent the synaptic interactions that will take place in response to this spike after some axonal delay period.

2. A single internal spike arrives at time $t_1^{sp}(t_0 < t_1^{sp} < T_1)$. At this point, the membrane voltage V_1 is drawn from $P_V(t_0, V_0; V, t_1^{sp})$ and subsequently updated according to the synaptic strength $\Delta V_1 = J$ associated with the incoming spike. Again, two scenarios are possible, depending on whether the new membrane potential $(V_1 + \Delta V_1)$ is (a) below threshold, in which case the neuron continues at step 1 using $V_0^{new} = V_1 + \Delta V_1$ and $t_0^{new} = t_1^{sp}$ (see Figure 2B), or (b) equal to or above threshold V_ϑ, in which case the neuron emits a spike and the membrane voltage is reset to V_{reset} during the absolute refractory period, after which the neuron continues at step 1 using $V_0^{new} = V_{reset}$ and $t_0^{new} = t_1^{sp} + \tau_{arp}$ (see Figure 2C). As before, the spike emission results in the generation of new events to be processed later in the simulation.

This is repeated during all the simulation time, or until no more events are to be processed.

A critical situation occurs when the statistics of the background activity change, because this also affects the probability distributions. In order to propagate the change to the whole network, all neurons must immediately be visited, and the current state variables must be updated according to P_V, using the old statistics. Then, using $P_{fpt}$with the new statistics, spontaneous spike times are determined as usual, and the simulation continues as described above.

It is clear from this procedure that the statistics of the background activity can change in time only by discrete steps rather than in a continuous way. This is an inherent limitation of the event-driven approach. However, most physiological situations can be adequately described by piecewise constant (discontinuous) changes in the statistical properties of the external background populations. Examples of such changes are given in Figures 3B and 4. For an extension to more complex transients or periodical noisy inputs, see section 4.

2.4 Implementation of the Event List. As a consequence of the event-based simulation approach, a large number of events have to be processed in chronological order. The strategies involved in handling the events have to be chosen with care, given that a lack of efficiency could lower the performance of the algorithm considerably.

The problem reduces to having a single ordered list with random insertions (when creating new events), repetitive deletion of the lowest element

(when the next event is processed), and occasional random deletions (when an uncertain event becomes irrelevant). An appropriate data structure for the storage of the events is given by the class of balanced trees, which ensure good performance: searching for one element out of E costs $\mathcal{O}(\log(E))$ operations. However, maintaining the balance of the tree causes considerable overhead, especially in our situation where repetitive deletions occur at one place (Wirth, 1986).

In our implementation, we used the "skip list" data structure, which is a probabilistic alternative to balanced trees (Pugh, 1990; see also appendix B), based on linked lists. In this structure, maintenance of the balance comes as a consequence of the probabilistic nature of the structure, and the insertion and deletion algorithms are as simple as those for linked lists. Therefore, skip lists strongly reduce the overhead of the event handling, while offering the performance benefits of balanced trees.

2.5 Look-up Tables. In the general case, the processing of an event requires a partial differential equation (e.g., equation A.1) to be solved numerically, which is time-consuming. Although this computational load does not directly depend on the assumed size of the external populations (M_{ext}), and therefore does not affect the run-time performance scaling of our algorithm with respect to M_{ext}, it nevertheless reduces the overall performance of the simulation in a dramatic way. However, we can make use of the fact that the parameters governing the evolution of the probability distributions can change only in a discontinuous manner, and prepare tables of all needed distributions off-line. These tables can be stored and used during the simulation, therefore reducing this computationally delicate task to simple table look-ups.

For a specific example, refer again to Figure 1, where such an off-line evaluated solution has been graphically represented: the time-varying probability density distribution P_V and the corresponding interspike interval distribution density $P_{fpt}$are plotted for a particular $V_0 = -70$ mV. The density distribution represented by the surface starts as a Dirac-delta located at $V = V_0$, and as time goes by with no internal event occurring, it drifts toward the asymptotic value ($\mu\tau_m = 14.5$ mV) and diffuses at a rate σ (with $\sigma^2\tau_m = 10.2$ mV^2). Accordingly, the first-passage time probability density distribution P_{fpt}first increases and then slowly decreases to zero for large time intervals. Of course, different initial conditions V_0 yield different profiles of P_V and P_{fpt}.

3 Performance Evaluation

The proposed algorithm provides fast simulations of large networks. In order to check the simulation results for correctness, the emergent behaviors were compared to the mean-field theory predictions.

3.1 Numerical Accuracy. The introduction of tabular discretizations will introduce numerical approximations that influence the accuracy of the simulations. The fact that a higher table resolution produces more accurate results leads to a trade-off between accuracy and table resolution (which directly affects memory consumption). This is illustrated in Figure 3: the different lines in Figure 3B correspond to different table resolutions. The mean frequency response to noisy external inputs with stationary statistics is compared to the one predicted analytically by the mean field theory (see Amit & Brunel, 1997a). The deviations from the predicted frequency in Figure 3B (23.6 Hz in the high noise regime) are plotted in Figure 3C as a function of table resolution. Of the resolutions tested in Figure 3C, the best results were obtained with 0.1 mV and 0.5 ms in the voltage and time domains, respectively, and the number of elements contained in the corresponding table was of the order of 10^6. Figure 3D demonstrates that the accuracy is very sensitive to the discretization of the voltage and less sensitive to the time resolution. Minor discrepancies in the global firing rate were expected, since an additional reflecting lower barrier for the depolarization of the single neuron was set at $V = V_{\min}$, slightly altering the IF dynamics compared to those employed in the theory (see equation A.1). The achieved accuracy was nevertheless satisfying, and the typical transient overshoots in the collective mean firing rate, expected after a sudden increase in the statistics of the background activity, were captured.

We also compared directly the simulations and the theoretical prediction of the transient response of a population of coupled neurons to a stepwise change in the statistics of the external current (see Figure 4). In this particular case, we were using the IF neuron with a constant leakage because the analytical solution of the network dynamics during transients is simpler (Mattia & Del Giudice, 2002). Note that changing the neuron model affects only the generation of the look-up tables, as all the information about the neural dynamics is contained within these tables.

Figure 4 shows simulations of a network of 5000 interacting, inhibitory neurons. At time $t = 0$ (2 seconds after the start of the simulation), the statistics of the external current are changed from ($\mu = 125$ mV/s, $\sigma = 5.15\,\mathrm{mV/s}$) to ($\mu = 700\,\mathrm{mV/s}, \sigma = 12.20\,\mathrm{mV/s}$). The transient response of the simulation follows very closely the theoretical prediction reported in Mattia and Del Giudice (2002).

In order to maximize accuracy and minimize memory usage, nonuniform discretizations and efficient interpolation methods can be used. In particular, voltages near the threshold and short time intervals should be represented with a refined discretization. In this last example, we obtain the exact steady-state values of the population frequency, which reflects the fact that we were using nonuniform, high-resolution discretizations of the underlying tables (minimal resolution of 0.1 mV at the reflecting barrier $V_{\min}$, and maximal resolution of 0.01 mV near the threshold V_ϑ).

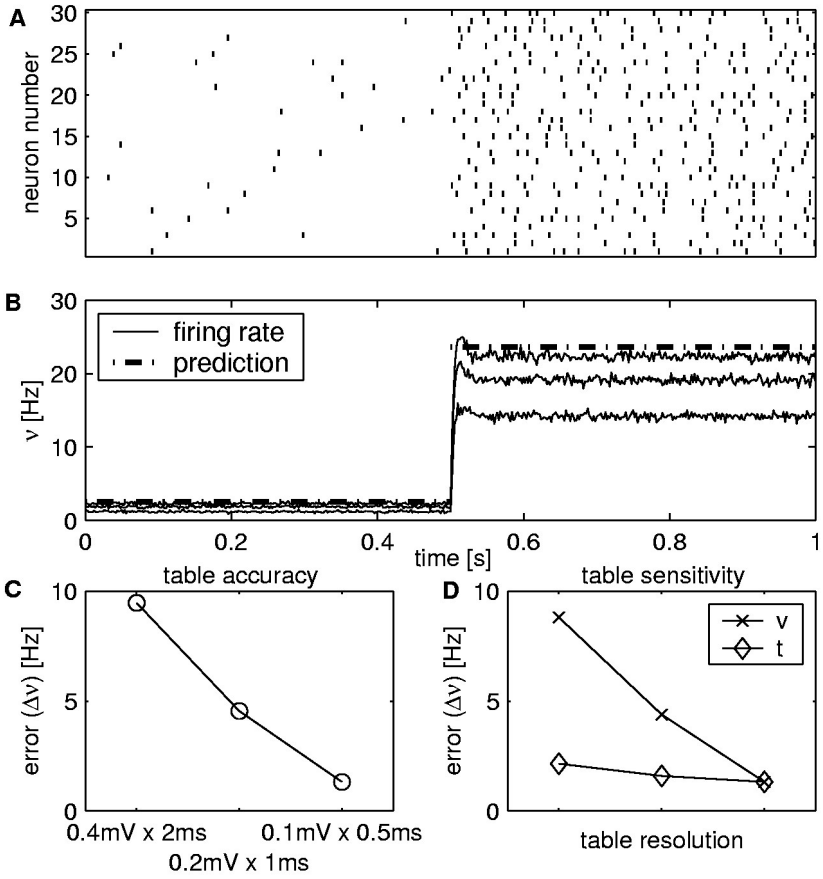

Figure 3: Simulation of a population of IF neurons for two different regimes of the external input ($\mu_1\tau_m = 14.5$ mV, $\sigma_1^2\tau_m = 10.2$ mV^2, $\mu_2\tau_m = 16.5$ mV, $\sigma_2^2\tau_m = 27$ mV^2). (A) Raster plot of 30 of 500 neurons and the corresponding instantaneous network firing rate (B, average over 100 runs) compared to the analytical prediction (dotted line) by the mean-field theory (see text; Amit & Tsodyks, 1991; Brunel, 2000; Fusi & Mattia, 1999). The three lines correspond to three different look-up table resolutions that were used. (C) Estimate of the inaccuracies induced by the discretizations of the look-up tables, measured in the last 200 ms of B. (D) Sensitivity of the inaccuracies to changes in table resolution (v: change in voltage axis [0.4 mV $\times$ 0.5 ms, 0.2 mV $\times$ 0.5 ms, 0.1 mV $\times$ 0.5 ms; from right to left], t: change in time axis [0.1 mV $\times$ 2 ms, 0.1 mV $\times$ 1 ms, 0.1 mV $\times$ 0.5 ms]).

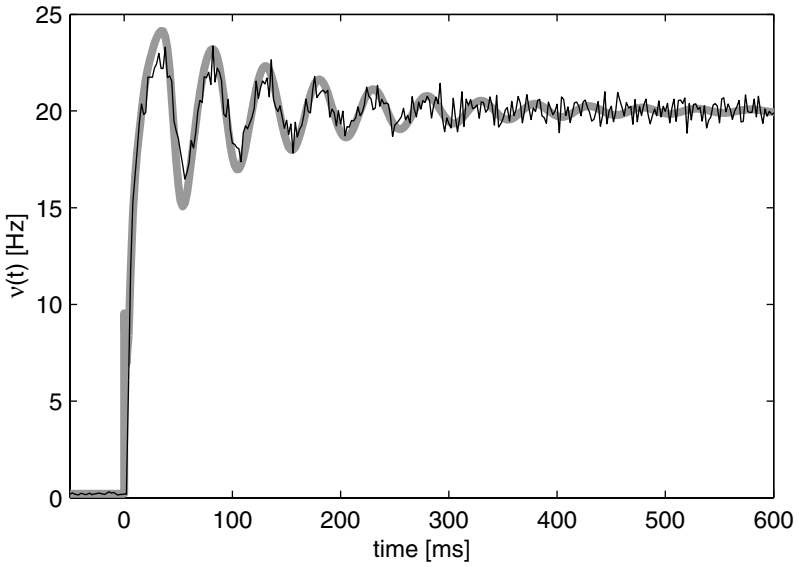

Figure 4: Transient response to a step change in the statistics of the external current: simulation versus theory. For $t < 0$, the network is in an asynchronous stationary state with mean emission rate $\nu = 0.2$ Hz. At $t = 0$, an instantaneous increase of the external current from ($\mu = 125$ mV/s, $\sigma = 5.15$ mV/s) to ($\mu = 700$ mV/s, $\sigma = 12.20$ mV/s) drives the activity toward a new stable state with $\nu = 20$ Hz. The solid black line is the mean of the activity from 10 simulations of a coupled network (5000 inhibitory LIF neurons; see the text). The thick gray line is the theoretical prediction. (Used with permission from Mattia & Del Giudice, 2002).

3.2 Run-Time Performance. All the simulations were performed on a Pentium III 850 MHz system running Linux, and they confirmed that excellent run-time performance can be achieved if the computation of the probabilistic dynamics (i.e., the on-line probability tables search and look-up, and the consequent random number extractions) is fast enough. Figure 5A reports a benchmark comparing our algorithm with the event-driven approach of Mattia and Del Giudice (2000). Both simulations were performed over a simulated time of 1 second with a network of 1000 IF neurons, characterized by a weak average internal synaptic coupling ($J = 0.1$ mV) and by a low connectivity probability ($c = 0.1$). With the aim of accounting for differences in the output frequency, the execution times of both simulation approaches were normalized and plotted as a function of the mean number of spikes received by each neuron from the external neurons (background activity, F_{ext}) divided by the number of spikes received by the other simu-

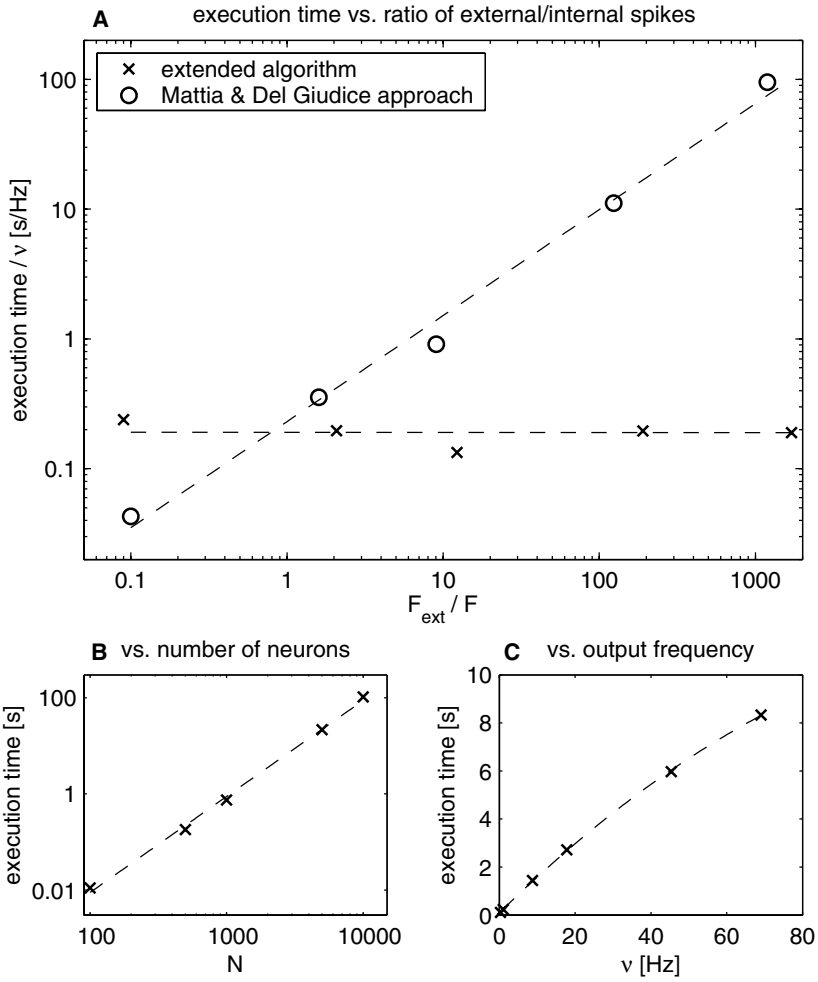

Figure 5: Execution performance of the proposed algorithm. (A) Comparison of the execution times of the state-of-the-art event-driven approach (as described in Mattia & Del Giudice, 2000, circles) and of the extended event-driven approach (crosses), as a function of the ratio between the number of the external and internal spikes (see section 3). The execution time is given per simulated second and averaged over five runs of 4 s simulation time each. (B) The execution time scales with the square of the number of simulated neurons ($c = 0.05$, mean output frequency 4 Hz). (C) Execution time scales almost linearly with output frequency ($c = 0.05$, $J = 0.1$ mV).

lated neurons (internal activity, F):

$$\frac{F_{ext}}{F} = \frac{M_{ext} \cdot c_{ext} \cdot \nu_{ext}}{N \cdot c \cdot \nu},$$

where c and c_{ext} are set to 0.1 and 1, respectively, ν_{ext} is calculated from the statistics of the background populations, and ν is the fraction of simulated neurons that emit a spike per unit time. The ratio M_{ext}/N was varied systematically (0.1, 1, 10, 100, 1000). The figure demonstrates that the execution time of the extended event-driven approach scales very well (does not increase) with respect to the increase in the size of the background populations. Only in the case of very few external spikes per unit time does the approach of Mattia and Del Giudice perform better than ours, which can be explained by the lower time required to process a single event: only one random number has to be generated compared to at least two in the extended algorithm (one for P_V and P_{fpt} each and for each newly inserted event in the event list).

The execution time is also affected by the size N of the network to be simulated, scaling with the square of N (see Figure 5B), and for a given network size, it increases almost linearly with the output frequency, as expected from the analysis of the traditional event-driven simulation paradigm.

As demonstrated by the simulation results, the main strength of the proposed approach is therefore related to the fact that the external populations were not modeled explicitly, resulting in an execution time independent of the number of external neurons (M_{ext}) and of the ratio of the external to internal spikes: $\mathcal{O}(N \cdot \nu_{ext})$. As a consequence, M_{ext} enters the simulation only as a parameter in the expressions of the mean μ and the variance σ^2 of the external current (see equation 2.2).

4 Discussion

The considerable speed-up of simulation times makes our approach suitable for the investigation of the emergent collective behaviors in large-scale networks of neurons and extends the perspectives of the event-driven simulation approach. In our tests, a simulation of 1000 neurons with a connectivity probability of 0.1 was running in real time (i.e., 1 s of the simulated network was executed in 1 s, while the mean frequency was 4 Hz), no matter how large the external population was.[1] It is important to stress that our approach does not imply any approximation provided that the number of independent external afferents to be mimicked is large enough and that the solution of the density equation is accurate. Our simulations produce the same results as any traditional fixed-time-step approach, in which the external afferents are statistically independent. In this, our strategy differs from

[1] The software implementing the described algorithm can be obtained from the authors upon request.

that of Knight (1972), who suggested studying the whole network dynamics by solving the density equations that govern the assembly of neurons. His strategy strongly relies on the assumption that the large number of neurons in the simulated network lose their identity and, hence, that the network state can be fully described by the probability density function of the state variables. In many cases, this represents a good approximation, although finite-size effects should be added artificially to the dynamic equations of the probability density function, to account for the finite size of the network under consideration (see Brunel & Hakim, 1999; Brunel, 2000; Mattia & Del Giudice, 2002). Moreover, the weak statistical correlations introduced by the synaptic couplings, which cannot be modeled in a density approach, must be neglected. In our case, all of these elements emerge naturally from the simulation, which still constitutes a useful tool to check the hypotheses of the density approach.

Our simulation strategy can be easily extended to events that are not well localized in time. For instance, when the synaptic currents generated by the arrival of presynaptic spikes are not delta correlated, the depolarization V cannot be updated instantaneously. A persistent current would affect the state variable of the neuron throughout the duration of the synaptic input. If the synaptic current goes instantaneously up upon the arrival of a spike and then decays exponentially with a time constant τ, we have that (see also Srinivasan & Chiel, 1993; Destexhe, Mainen, & Sejnowski, 1994; Lytton, 1996; Giugliano, 2000; Nykamp & Tranchina, 2001)

$$I_{int}(t) = \sum_k J_k e^{-(t-t_k)/\tau} = e^{-t/\tau} \sum_k J_k e^{t_k/\tau} = e^{-t/\tau} I_{sp},$$

where I_{int}, the total synaptic current, is updated when a spike arrives (a term is added to the sum I_{sp}) and decays exponentially during the interspike intervals. Such a dynamic behavior can be introduced in our approach by defining an effective external mean current $\mu_{eff}(t)$:

$$\mu_{eff}(t) = \mu(t) + e^{-t/\tau} I_{sp}.$$

The density equations should then be solved for every I_{sp}. This makes the look-up tables for the probability density functions more complicated (it adds a dimension corresponding to I_{sp}), but it provides the possibility of having more realistic currents.

Another interesting extension to our model can deal with situations in which we have a time-varying periodic change in the background statistics (i.e., an oscillatory, noisy input current) or a transient with a complex temporal profile that is limited in time. Such a situation can be simulated without changing continuously the statistics of the input current, which would require a frequent update of all the state variables of the neurons even in the absence of incoming spikes (see section 2.3). The solution is to add a time-dependent state variable describing the neuron's position within

the phase ϑ_0 of the period. At the update time, the neuronal state variable would be updated according to the appropriate look-up table, that is, the one corresponding to the neuronal dynamics under the time-varying stimulus starting at the update time. As an example, $P_{fpt}(t_0, V_0; t)$ would then become $P_{fpt}(t_0, V_0, \theta_0; t)$, and $P_V(t_0, V_0; V, t)$ would have to be extended to $P_V(t_0, V_0, \theta_0; V, t)$. The extra cost of this approach is an added dimension in the look-up tables whose extension depends on the time correlation length of the input signal, the length of the temporal profile (the period in the case of a cyclic input), and the time resolution. However, the simulation performance in terms of execution time is not impaired.

In general, simulations with more complex model neurons or synapses also require the addition of other dimensions in the probability density function. Moreover, solving high-dimensional partial differential equations is difficult, and the look-up tables would require a huge amount of memory. This is a major limitation of our approach because the problem becomes intractable even with few state variables. Nevertheless, our strategy (as most of the event-driven approaches) is efficient when the problem and the model are already well defined, the relevant internal variables are known, and extensive simulations are needed to explore the parameter space.

Appendix A: Fokker-Planck Equation

The probability density function $P_V(V, t)$ that describes the statistics of the dynamics during the intervals between two successive spikes evolves in the voltage-time domain according to the Fokker-Planck equation (Cox & Miller, 1965),

$$\frac{\partial}{\partial t} P_V(V, t) = -\frac{\partial}{\partial V} \phi(V, t), \tag{A.1}$$

where $\phi(V, t) = -\frac{\sigma^2}{2} \cdot \frac{\partial}{\partial V} p(V, t) + [(\mu - L(V, t)) \cdot p(V, t)]$.

In the most general case, equation A.1 must be complemented by the initial condition $P_V(V, t_0) = \delta(V - V_0)$ and by appropriate boundary conditions restricting, in the case of the leaky IF neuron, the potential

- at $V = V_{\min}$, in terms of a reflecting barrier, since no neuron can have a depolarization that passes below $V_{\min}$

$$\forall\, t,\ \phi(V_{\min}, t) = 0 \tag{A.2}$$

- at $V = V_\vartheta$, in terms of an absorbing barrier, since neurons that are crossing such a threshold are absorbed and leave the interval $(V_{\min}; V_\vartheta)$:

$$\forall\, t,\ P_V(V_\vartheta, t) = 0. \tag{A.3}$$

Finally, at any time t, the probability density $P_V(V, t)$ must satisfy a normalization condition,

$$\int_{V_{min}}^{V_\vartheta} p(V, t)\, dV + \int_{t_0}^{t} \nu(t')\, dt' = 1 \tag{A.4}$$

where $\nu(t) = \phi(V_\vartheta, t) = P_{fpt}(t) = -\frac{\sigma^2}{2} \cdot \frac{\partial}{\partial V} P_V(V, t)|_{V_\vartheta}$, represents the rate at which neurons are crossing the threshold V_ϑ. We note that equation A.4 accounts for the deflating time course of the surface $P_V(V, t)$, sketched in Figure 1: with time, the realizations of the process $V(t)$ leave the integration interval $(V_{\min}, V_\vartheta)$ after absorption at the threshold, and they irreversibly accumulate in the term $\int_{t_0}^{t} \nu(t')\, dt'$.

By preliminarily choosing a fixed set of values for (μ, σ) and discretizing the interval $(V_{\min}; V_\vartheta)$ into a finite number of bins, the numerical integration of equation A.1 determines the desired time-dependent probability density distributions $P_V(\mu, \sigma^2, t_0, V_0, t_1; V_1)$ and $P_{fpt}(\mu, \sigma^2, t_0, V_0; t)$, as a function of (μ, σ) and given that $V(t_0) = V_0$.

Appendix B: Skip List

Skip lists are a data structure introduced by W. Pugh. Skip lists can be used in place of balanced trees (Wirth, 1986), and they use probabilistic balancing rather than strictly enforced balancing.

The underlying data structure is an ordered linked list of length n. When searching a linked list, we may have to visit all the nodes of the list. If every second node also has a pointer to the node two ahead it in the list, we have to examine no more than $\lceil n/2 \rceil + 1$ nodes, since we can skip over half the nodes and go back only one node in case we had skipped over the element we were looking for. If every (2^i)th node has a pointer 2^i nodes ahead, the number of nodes that must be examined can be reduced to $\lceil \log_2 n \rceil$ while only doubling the number of pointers. A node that has k forward pointers is called a *level k* node. In a skip list, the levels of nodes are chosen randomly; the probabilities of choosing a certain level follow a geometric series (50% are level 1, 25% level 2, 12.5% level 3, and so on). A node's ith forward pointer, instead of pointing 2^{i-1} nodes ahead, points to the next node of level i or higher. Thus, insertions and deletions require only local modifications; the level of a node, chosen randomly when the node is inserted, never has to be changed.

In summary, the data structure is a linked list with extra pointers that skip over intermediate nodes—hence the name. Skip lists are balanced by consulting a random number generator and do not require rearranging of the structure, as is the case for balanced trees. Furthermore, the simplicity of skip list algorithms (only local modifications) makes them easier to implement and provides significant constant factor speed improvements over

balanced and self-adjusting tree algorithms. A more detailed analysis of the algorithmic performance can be found in Pugh (1990).

Acknowledgments

We thank Maurizio Mattia for providing his software and assistance and for very useful discussions and remarks on the manuscript. This work was supported by SNF grants 21-57076.99 and 31-61335.00 and by the Human Frontier Science Foundation (LT00561/2001-B).

References

Amit, D. J., & Brunel, N. (1997a). Model of global spontaneous activity and local structured activity during delay periods in the cerebral cortex. *Cereb. Cortex, 7*(3), 237–252.

Amit, D. J., & Brunel, N. (1997b). Dynamics of a recurrent network of spiking neurons before and following learning. *Network: Computation in Neural Systems, 8*, 373–404.

Amit, D. J., & Tsodyks, M. V. (1991). Quantitative study of attractor neural networks retrieving at low spike rates I: Substrate-spikes, rates and neuronal gain. *Network, 2*(3), 259–274.

Brunel, N. (2000). Dynamics of sparsely connected networks of excitatory and inhibitory spiking neurons. *J. Comp. Neurosci., 8*, 183–208.

Brunel, N., & Hakim, V. (1999). Fast global oscillations in networks of integrate-and-fire neurons with low firing rates. *Neural Comp., 11*(7), 1621–1671.

Cox, D. R., & Miller, H. D. (1965). *The theory of stochastic processes.* London: Methuen.

Delorme, A., Gautrais, J., van Rullen, R., & Thorpe, S. (1999). SpikeNET: A simulator for modeling large networks of integrate and fire neurons. *Neurocomputing, 26/27*, 989–996.

Destexhe, A., Mainen, Z. F., & Sejnowski, T. J. (1994). An efficient method for computing synaptic conductances based on a kinetic model of receptor binding. *Neural Comp., 6*, 14–18.

Fusi, S., & Mattia, M. (1999). Collective behavior of networks with linear (VLSI) integrate and fire neurons. *Neural Computation, 11*, 633–652.

Gerstein, G. L., & Mandelbrot, B. (1964). Random walk models for the spike activity of a single neuron. *Biophys. J., 4*, 41–68.

Gerstner, W., & Kistler, W. (2002). *Spiking neuron models: Single neurons, populations, plasticity*. Cambridge: Cambridge University Press.

Giugliano, M. (2000). Synthesis of generalized algorithms for the fast computation of synaptic conductances with Markov kinetic models in large network simulations. *Neural Comput., 12*(4), 771–799.

Knight, B. W. (1972). Dynamics of encoding in a population of neurons. *Journal of General Physiology, 59*, 734–766.

Lytton, W. W. (1996). Optimizing synaptic conductance calculation for network simulations. *Neural Comp., 8*, 501–509.

Mattia, M., & Del Giudice, P. (2000). Efficient event-driven simulation of large networks of spiking neurons and dynamical synapses. *Neural Comput., 12*(10), 2305–2329.

Mattia, M., & Del Giudice, P. (2002). *Population dynamics of interacting spiking neurons*. *Phys. Rev. E 66(5)*, 051917.

Nykamp, D. Q., & Tranchina, D. (2000). A population density approach that facilitates large-scale modeling of neural networks: Analysis and an application to orientation tuning. *J. Comp. Neurosci., 8*, 19–50.

Nykamp, D. Q., & Tranchina, D. (2001). A population density approach that facilitates large-scale modeling of neural networks: Extension to slow inhibitory synapses. *Neural Comp., 13*, 511–546.

Press, W. H., Flannery, B. P., Teukolsky, S. A., & Vetterling, W. T. (1986). *Numerical recipes: The art of scientific computing*. Cambridge: Cambridge University Press.

Pugh, W. (1990). Skip lists: A probabilistic alternative to balanced trees. *Communications of the ACM, 33*(6), 668–676.

Rauch, A., La Camera, G., Lüscher, H.-R., Senn, W., & Fusi, S. (2002). *Neocortical pyramidal cells respond as integrate-and-fire neurons to in vivo-like input currents*. Manuscript submitted for publication.

Ricciardi, L. M. (1977). *Diffusion processes and related topics in biology*. Berlin: Springer-Verlag.

Srinivasan, R., & Chiel, H. J. (1993). Fast calculation of synaptic conductances. *Neural Comp., 5*, 200–204.

Tuckwell, H. C. (1988). *Introduction to theoretical neurobiology, Vol. 2: Non linear and stochastic theories.* Cambridge: Cambridge University Press.

Wang, X.-J. (1999). Synaptic basis of cortical persistent activity: The importance of NMDA receptors to working memory. *J. Neurosci., 19*(21), 9587–9603.

Watts, L. (1994). Event-driven simulation of networks of spiking neurons. In J. D. Cowan, G. Tesauro, & J. Alspector (Eds.), *Advances in neural information processing systems, 6* (pp. 927–934). San Mateo, CA: Morgan Kaufmann.

Wirth, N. (1986). *Algorithms and data structures*. Upper Saddle River, NJ: Prentice Hall.

Yakovlev, V., Fusi, S., Berman, E., & Zohary, E. (1998). Inter-trial neuronal activity in infero-temporal cortex: A putative vehicle to generate long term associations. *Nature Neuroscience, 1*(4), 310–317.

Received May 3, 2002; accepted October 1, 2002.

This article has been cited by:

1. Niceto R. Luque, Richard R. Carrillo, Francisco Naveros, Jesús A. Garrido, M.J. Sáez-Lara. 2014. Integrated neural and robotic simulations. Simulation of cerebellar neurobiological substrate for an object-oriented dynamic model abstraction process. *Robotics and Autonomous Systems* **62**, 1702-1716. [CrossRef]
2. Alberto Seseña-Rubfiaro, Juan Carlos Echeverría, Jose Rafael Godínez-Fernández. 2014. Fractal-like correlations of the fluctuating inter-spike membrane potential of a Helix aspersa pacemaker neuron. *Computers in Biology and Medicine* **53**, 258-264. [CrossRef]
3. Dejan Pecevski, David Kappel, Zeno Jonke. 2014. NEVESIM: event-driven neural simulation framework with a Python interface. *Frontiers in Neuroinformatics* **8**. . [CrossRef]
4. S.C. Chan, R.R. Poznanski, S.Y. Goh. 2014. Network activity in a Morris–Lecar population density model. *Neurocomputing* **138**, 332-338. [CrossRef]
5. Aaron Miller, Dezhe Z. Jin. 2013. Potentiation decay of synapses and length distributions of synfire chains self-organized in recurrent neural networks. *Physical Review E* **88**. . [CrossRef]
6. Thibaud Taillefumier, Jonathan Touboul, Marcelo Magnasco. 2012. Exact Event-Driven Implementation for Recurrent Networks of Stochastic Perfect Integrate-and-Fire Neurons. *Neural Computation* **24**:12, 3145-3180. [Abstract] [Full Text] [PDF] [PDF Plus]
7. Michelle Rudolph-Lilith, Mathieu Dubois, Alain Destexhe. 2012. Analytical Integrate-and-Fire Neuron Models with Conductance-Based Dynamics and Realistic Postsynaptic Potential Time Course for Event-Driven Simulation Strategies. *Neural Computation* **24**:6, 1426-1461. [Abstract] [Full Text] [PDF] [PDF Plus]
8. Yi Sun, Aaditya V. Rangan, Douglas Zhou, David Cai. 2011. Coarse-grained event tree analysis for quantifying Hodgkin-Huxley neuronal network dynamics. *Journal of Computational Neuroscience* . [CrossRef]
9. Yi Sun, Douglas Zhou, Aaditya V. Rangan, David Cai. 2010. Pseudo-Lyapunov exponents and predictability of Hodgkin-Huxley neuronal network dynamics. *Journal of Computational Neuroscience* **28**, 247-266. [CrossRef]
10. Derek Daniel. 2009. Fokker-Planck Solution for a Neuronal Spiking Model. *Transport Theory and Statistical Physics* **38**, 383-391. [CrossRef]
11. Yi Sun, Douglas Zhou, Aaditya V. Rangan, David Cai. 2009. Library-based numerical reduction of the Hodgkin–Huxley neuron for network simulation. *Journal of Computational Neuroscience* **27**, 369-390. [CrossRef]
12. Romain Brette, Michelle Rudolph, Ted Carnevale, Michael Hines, David Beeman, James M. Bower, Markus Diesmann, Abigail Morrison, Philip H. Goodman, Frederick C. Harris, Milind Zirpe, Thomas Natschläger, Dejan Pecevski, Bard Ermentrout, Mikael Djurfeldt, Anders Lansner, Olivier Rochel, Thierry Vieville, Eilif Muller, Andrew P. Davison, Sami El Boustani, Alain Destexhe. 2007. Simulation of networks of spiking neurons: A review of tools and strategies. *Journal of Computational Neuroscience* **23**, 349-398. [CrossRef]
13. R. Agís, E. Ros, J. Díaz, R. Carrillo, E. M. Ortigosa. 2007. Hardware event-driven simulation engine for spiking neural networks. *International Journal of Electronics* **94**, 469-480. [CrossRef]
14. Abigail Morrison, Sirko Straube, Hans Ekkehard Plesser, Markus Diesmann. 2007. Exact Subthreshold Integration with Continuous Spike Times in Discrete-Time Neural Network Simulations. *Neural Computation* **19**:1, 47-79. [Abstract] [PDF] [PDF Plus]
15. Eduardo Ros, Richard Carrillo, Eva M. Ortigosa, Boris Barbour, Rodrigo Agís. 2006. Event-Driven Simulation Scheme for Spiking Neural Networks Using Lookup Tables to Characterize Neuronal Dynamics. *Neural Computation* **18**:12, 2959-2993. [Abstract] [PDF] [PDF Plus]
16. Michelle Rudolph, Alain Destexhe. 2006. Analytical Integrate-and-Fire Neuron Models with Conductance-Based Dynamics for Event-Driven Simulation Strategies. *Neural Computation* **18**:9, 2146-2210. [Abstract] [PDF] [PDF Plus]
17. E. Ros, E.M. Ortigosa, R. Agis, R. Carrillo, M. Arnold. 2006. Real-Time Computing Platform for Spiking Neurons (RT-Spike). *IEEE Transactions on Neural Networks* **17**, 1050-1063. [CrossRef]
18. M. Giugliano, M. ArsieroBiological Neuronal Networks, Modeling of . [CrossRef]
19. Abigail Morrison, Carsten Mehring, Theo Geisel, Ad Aertsen, Markus Diesmann. 2005. Advancing the Boundaries of High-Connectivity Network Simulation with Distributed Computing. *Neural Computation* **17**:8, 1776-1801. [Abstract] [PDF] [PDF Plus]
20. P Del Giudice. 2003. Modelling the formation of working memory with networks of integrate-and-fire neurons connected by plastic synapses. *Journal of Physiology-Paris* **97**, 659-681. [CrossRef]