Spike-based models of neural computation

Romain Brette

December 2009

Mémoire d'Habilitation à Diriger des Recherches

Jury

Nicolas Brunel (Université Paris V)
Vincent Hakim (Ecole Normale Supérieure, Paris)
Claude Meunier (Université Paris V)
Walter Senn (Université de Berne, Suisse)
Simon Thorpe (Université Paul Sabatier, Toulouse)

Abstract

Neurons compute mainly with action potentials or "spikes", which are stereotypical electrical impulses. Over the last century, the operating function of neurons has been mainly described in terms of firing rates, with the timing of spikes bearing little information. More recently, experimental evidence and theoretical studies have shown that the relative spike timing of inputs has an important effect both on computation and learning in neurons. This evidence has triggered considerable interest for spiking neuron models in computational neuroscience, but the theory of computation in those models is sparse.

Spiking neuron models are hybrid dynamical systems, combining differential equations and discrete events. I have developed specific theoretical approaches to study this particular type of models. In particular, two specific properties seem to be relevant for computation: spiking models can encode time-varying inputs into trains of precisely timed spikes, and they are more likely fire to when input spike trains are tightly correlated. To simulate spiking models efficiently, we have developed specific techniques, which can now be used in an open source simulator (Brian).

These theoretical and methodological investigations now allow us to address spike-based modeling at a more global and functional level. Since the mechanisms of synaptic plasticity tend to favor synchronous inputs, I propose to investigate computational mechanisms based on neural synchrony in sensory modalities.

Contents

1 Introduction

Computational neuroscience is often depicted as a subfield of neuroscience which uses computer modelling as its main tool to understand the nervous system. A famous example is the Nobelized Hodgkin-Huxley model. In the beginning of the twentieth century, the action potential was thought to reflect a non-selective increase in membrane permeability, which would reduce the difference between intracellular and extracellular potential (Bernstein, 1912). This view was challenged by Hodgkin and Huxley who found, in the first intracellular recording of an action potential in an animal cell[1], that the membrane potential became positive during action potentials (Hodgkin and Huxley, 1939), and they hypothesized that the action potential was due to a selective increase in sodium permeability. To support their theory, they developed a quantitative model of the squid axon membrane which could account for many aspects of electrophysiological recordings (Hodgkin and Huxley, 1952). This achievement is all the more impressive since single channel recordings were made possible only 20 years later (Sakmann et al., 1976). The mathematical model was a central part of the discoveries for which Hodgkin and Huxley were awarded the Nobel prize.

This historical example highlights the idea that computer modelling allows testing ideas when experiments would be very challenging or impossible. It is also a way of addressing questions for which the experimental approach is intrinsically limited in complex systems such as the brain, that is, understanding how the properties of elementary units (neurons) combine to explain the properties of the organism. While experiments can help unravel the properties at a given complexity level (e.g., integration properties of single neurons), they do not directly explain how these properties transfer to the next complexity level (Anderson, 1972). Thus, computational modelling is an answer to the limits of reductionism in complex systems, where properties of distinct complexity levels (neuron, network, organism) might be qualitatively very different. Moreover, in living organisms, one might argue that it is not so obvious whether the lower level (neurons) should explain the higher one (organism) or the converse should be true, because lower level properties indirectly depend on higher level properties (e.g. ecological fitness) through evolution.

I will give two simple examples of how computational modelling, or theoretical neuroscience in general, has increased our understanding of neural function. Much of our current understanding of neuron function comes from *in vitro* experiments. Patch clamp experiments have shown that the membrane time constant of cortical neurons is several tens of milliseconds, suggesting that neurons integrate inputs over time and are mostly insensitive to fine correlations. However, computational modelling studies have later shown that the situation might be very different *in vivo*, because the intense synaptic activity increases the total conductance and therefore shortens the time constant, putting neurons in a so-called *high-conductance state* (Destexhe et al., 2003). This was confirmed in experiments, showing that the membrane time constant *in vivo* is only a few milliseconds (Pare

[1]The first intracellular recording of an action potential was in fact made by Umrath in 1930 in plant cells, which have very slow action potentials (Umrath, 1930).

et al., 1998; Leger et al., 2005) [2]. This simple fact implies that neurons are driven by fast fluctuations (possibly tightly correlated inputs) because their membrane time constant is many times shorter than typical interspike intervals.

The second example is the concept of the balanced regime: excitation is, on average, balanced by the same amount of inhibition in cortical neurons. It is an interesting example in that a constraint at network level transfers to a constraint for single cells, which is the converse of a standard reductionist approach. The idea comes from the observation of an apparent contradiction: cortical neurons receive inputs from many neurons (about 10,000 synapses per neuron) and fire very irregularly *in vivo* (in fact, almost Poisson-like), but the law of larger numbers would predict that any single neuron should then receive a quasi-constant drive and thus fire regularly (Softky and Koch, 1993). This contradiction can be raised if inhibition balances excitation, because then the mean drive is below threshold and spikes can only be triggered by fluctuations, which are irregular (Shadlen and Newsome, 1998).

While *computational neuroscience* often refers to the use of computer modelling to address neuroscience questions that cannot be addressed by experimentation, there is another view of the field which I want to develop more specifically in this thesis, which is the study of how the brain computes. The brain does not work as a computer, but it computes: it solves complex computational problems such as understanding speech in noisy environments or identifying faces. It does so in a way that is fundamentally different from a Turing machine, the model of classical computing. For example, it is massively parallel (billions of neurons) and very plastic, memory is distributed and addressed by content. All these facts are now well known and have led to the development of successful computational theories such as artificial neural network theory. But there is one other specific fact about how the brain computes, which is the subject of this thesis: to a first approximation[3], neurons communicate with discrete timed events: spikes. Thus, a neuron can be seen as an elementary computing device which maps input spike trains to an output spike train, which is why I chose to focus on spiking neuron models and to investigate what this fact might imply in terms of computation.

The first question to answer (section 2) is: what is a realistic spiking neuron model? I focussed on two specific aspects: spike initiation, leading to dynamical models of the spike threshold, and two-dimensional spiking models that can account for a variety of electrophysiological classes. Secondly, I will present some of the most relevant mathematical properties of single spiking neuron models (section 3), in particular two properties that are important for computation: reliable encoding of time-varying inputs into trains of precisely timed spikes, and coincidence detection. These models require specific techniques to simulate them, which I will discuss in section 4. Finally, I will sketch a theory of spike-based computation

[2]The value might be underestimated because those experiments used sharp microelectrodes, which damage the membrane and typically yield smaller time constant values than patch electrodes (Brette and Destexhe, 2009); in any case, it remains that the membrane time constant is several times shorter *in vivo* than at rest.

[3]Deviations from spike-based communication include electrical synapses and effects of subthreshold potential on target synapses (Shu et al., 2006).

that relies on selective synchronization (section 5).

2 What is a good spiking neuron model?

2.1 General considerations

What is a spiking neuron model?

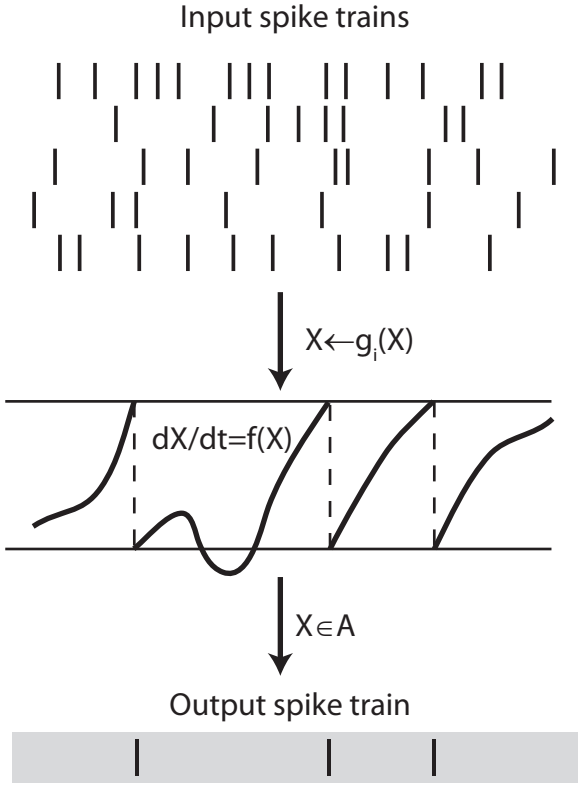


Figure 1: A spiking neuron model: input spikes modify state variables $\mathbf{X}$, which otherwise evolve continuously through differential equations. Spikes are emitted when a threshold condition is met.

A spiking model is a mathematical model which describes how input spike trains (sequences of timings) are mapped to an output spike train. Mathematically, neurons can be described as *hybrid systems* (Brette et al., 2007): the state of a neuron evolves continuously according to some biophysical equations, which are typically differential equations (deterministic or stochastic, ordinary or partial differential equations), and spikes received through the synapses trigger changes in some of the variables (see Fig. 1). Thus the dynamics of a neuron can be

described as follows:

$$\begin{aligned} \frac{d\mathbf{X}}{dt} &= f(\mathbf{X}) \\ \mathbf{X} &\leftarrow g_i(\mathbf{X}) \qquad \text{upon spike from synapse i} \end{aligned}$$

where $\mathbf{X}$ is a vector describing the state of the neuron[4]. Spikes are emitted when some threshold condition is satisfied, for instance $V_m \geq \theta$ for integrate-and-fire models (where V_m is the membrane potential and would be the first component of vector $\mathbf{X}$), and/or $dV_m/dt \geq \theta$ for Hodgkin-Huxley type models. This can be summarized by saying that a spike is emitted whenever some condition $\mathbf{X} \in \mathbf{A}$ is satisfied. For integrate-and-fire models, the membrane potential, which would be the first component of $\mathbf{X}$, is reset when a spike is produced[5].

A different formalism may be found in the litterature, where the membrane potential is expressed as a sum of postsynaptic potentials, but it can normally be equivalently restated in the hybrid system formalism. For example, consider the following integrate-and-fire model (described for example in Gtig and Sompolinsky (2006)):

$$V(t) = \sum_i w_i \sum_{t_i} K(t - t_i) + V_{\text{rest}}$$

where $V(t)$ is the membrane potential, V_{rest} is the rest potential, w_i is the synaptic weight of synapse i, t_i are the timings of the spikes coming from synapse i, and $K(t-t_i) = \exp(-(t-t_i)/\tau) - \exp(-(t-t_i)/\tau_s)$ is the post-synaptic potential (PSP) contributed by each incoming spike. The model can be restated as a two-variables differential system with discrete events as follows:

$$\begin{aligned} \tau \frac{dV}{dt} &= V_{\text{rest}} - V + J \\ \tau_s \frac{dJ}{dt} &= -J \\ J &\leftarrow J + \frac{\tau - \tau_s}{\tau} w_i \qquad \text{upon spike from synapse i} \end{aligned}$$

Virtually all post-synaptic potentials or currents described in the literature (e.g. α-functions, bi-exponential functions) can be expressed this way. Several authors have described the transformation from phenomenological expressions to the hybrid system formalism for synaptic conductances and currents (Destexhe et al., 1994; Rotter and Diesmann, 1999; Giugliano, 2000), short-term synaptic depression (Giugliano et al., 1999), and spike-timing-dependent plasticity (Song et al., 2000). In many cases, the Spike Response Model (Gerstner and Kistler, 2002) is also the integral expression of a hybrid system. To derive the differential formulation of a given post-synaptic current or conductance (PSC), one way is to see the latter as the impulse response of a linear time-invariant system (which

[4] In theory, taking into account the morphology of the neuron would lead to partial differential equations; however, in practice, one usually approximates the dendritic tree by coupled isopotential compartments, which also leads to a differential system with discrete events.

[5] The reset can be integrated into the hybrid system formalism by considering for example that outgoing spikes act on $\mathbf{X}$ through an additional (virtual) synapse: $\mathbf{X} \leftarrow g_0(\mathbf{X})$.

can be seen as a filter (Jahnke et al., 1999)) and use transformation tools from signal processing theory such as the Z-transform (Khn and Wrgtter, 1998) or the Laplace transform (the Z-transform is the equivalent of the Laplace transform in the digital time domain, i.e., for clock-driven algorithms).

What is a "realistic" neuron model?

It seems very reasonable to equate realism and level of detail in models, that is, that a realistic model is one that incorporates many biological components, for example Hodgkin-Huxley models are more realistic than integrate-and-fire models, but less realistic than multicompartmental biophysical models. Since simpler models neglect some aspects of cellular biophysics, they must be less realistic than models that do not. Although in principle this seems a very reasonable statement, I would argue that in many cases the converse might in fact be true, because adding more details introduces more uncertainty (e.g. about parameter values) at the expense of functional constraints, which are in the end more important. I will illustrate this point with an example, which will be explained in more detail in the next section.

The Hodgkin-Huxley model has been very successful in explaining action potential generation, and consequently it has been widely used in computational modelling studies, for example in single compartment models of cortical neurons (Meunier and Segev, 2002). It is generally assumed that it is more accurate that an integrate-and-fire model, which is used only as a simplification. However, it was meant to model the axon of a giant squid, and for example it predicts an unrealistic shape for action potentials of cortical neurons, which are much sharper (Naundorf et al., 2006).The reason for this is known, and is in fact not contradictory with Hodgkin-Huxley theory: it is due to active backpropagation of spikes from the site of action potential generation in the axon hillock to the soma (McCormick et al., 2007). But it remains that the single-compartment Hodgkin-Huxley model is a poor model of cortical neurons (see also section 2.2). On the other hand, simpler adaptive integrate-and-fire models are very good at predicting the rate and spike times of a cortical neuron in response to somatic current injection (Rauch et al., 2003; Badel et al., 2008; Jolivet et al., 2008; Camera et al., 2008), using an effective sharpness parameter (the slope factor) that is smaller than expected from sodium channel properties (slightly more than 1 mV vs. more than 6 mV). To account for the sharpness of cortical spikes in biophysical models, one needs to use a multicompartmental model in which action potentials are initiated in the axon hillock rather than in the soma. Still, such a model would predict that spikes should be even less sharp than at initiation site (since they are low-pass filtered) unless adequate active properties (sodium channels) are included between the soma and initiation site.

This example illustrates the idea that in many cases it might be more accurate to use a good phenomenological model than one that incorporates many biological components. In the following, I will consider that a *realistic* spiking model is one that provides an accurate input-output relationship (in terms of spike trains), even though some of the variables or parameters might only reflect effective properties.

The question is then what aspects of this relationship are most significant. Many studies have shown that adaptation and refractoriness are very important aspects of neural function (Camera et al., 2006; Gerstner and Naud, 2009). I will focus in the next section on spike initiation, then I will discuss electrophysiological classes.

2.2 Spike initiation and spike threshold

Spike initiation in neurons follows the *all-or-none* principle: spikes are produced when the neuron is sufficiently excited, while no spike is initiated below that threshold. This description suggests that there is a fixed voltage threshold above which spikes are initiated, as in the standard integrate-and-fire model. However, this is not exactly so in real neurons. First of all, the threshold depends on the type of stimulation (Koch et al., 1995), because spikes are not instantly produced. Secondly, spike threshold varies greatly *in vivo*: for example it depends on the preceding rate of depolarization (Azouz and Gray, 2000, 2003; Wilent and Contreras, 2005) and on the preceding interspike intervals (Henze and Buzsaki, 2001). Since spikes are produced when the membrane potential reaches the threshold, understanding threshold dynamics might be as important as understanding membrane potential dynamics, although it has not received as much attention. I have recently looked at this problem with Jonathan Platkiewicz (Platkiewicz and Brette, 2009).

The exponential approximation

Spikes are initiated by the opening of sodium channels. In the Hodgkin-Huxley formalism, the sodium current is $I_{Na} = g_{Na} P_{Na} (E_{Na} - V)$, where g_{Na} is the maximum conductance, E_{Na} is the sodium reversal potential, V is the membrane potential and P_{Na} is the proportion of open Na channels. We neglect inactivation for the moment. Activation is very fast compared to all other time constants (a fraction of ms). We make the approximation that it is instantaneous, so that the proportion of activated Na channels at any time equals the steady-state activation $P_a^\infty(V)$, which can be empirically described as a Boltzmann function (Angelino and Brenner, 2007):

$$P_a^\infty(V) = \frac{1}{1 + \exp(-(V - V_a)/k_a)}$$

where V_a is the half-activation voltage ($P_a^\infty(V) = 1/2$) and k_a is the activation slope factor. Spikes are initiated well below V_a (which is about -30 mV, Angelino and Brenner (2007)), so that $e^{-(V-V_a)/k_a} >> 1$ except during the action potential. Similarly, E_{Na} is very high (about 55 mV), so that $E_{Na} - V$ is not very variable below threshold. We make the approximation $E_{Na} - V \approx E_{Na} - V_a$ and we obtain:

$$I_{Na} = g_{Na}(E_{Na} - V_a) e^{(V-V_a)/k_a} = g_L \Delta_T e^{(V-V_T)/\Delta_T}$$

where $\Delta_T = k_a$, g_L is the leak conductance and $V_T = V_a - k_a \log \frac{g_{Na}}{g_L} \frac{E_{Na} - V_a}{k_a}$. This approximation is meaningful for spike initiation but not for spike shape (in particular, sodium activation dynamics cannot be neglected at high currents, i.e.,

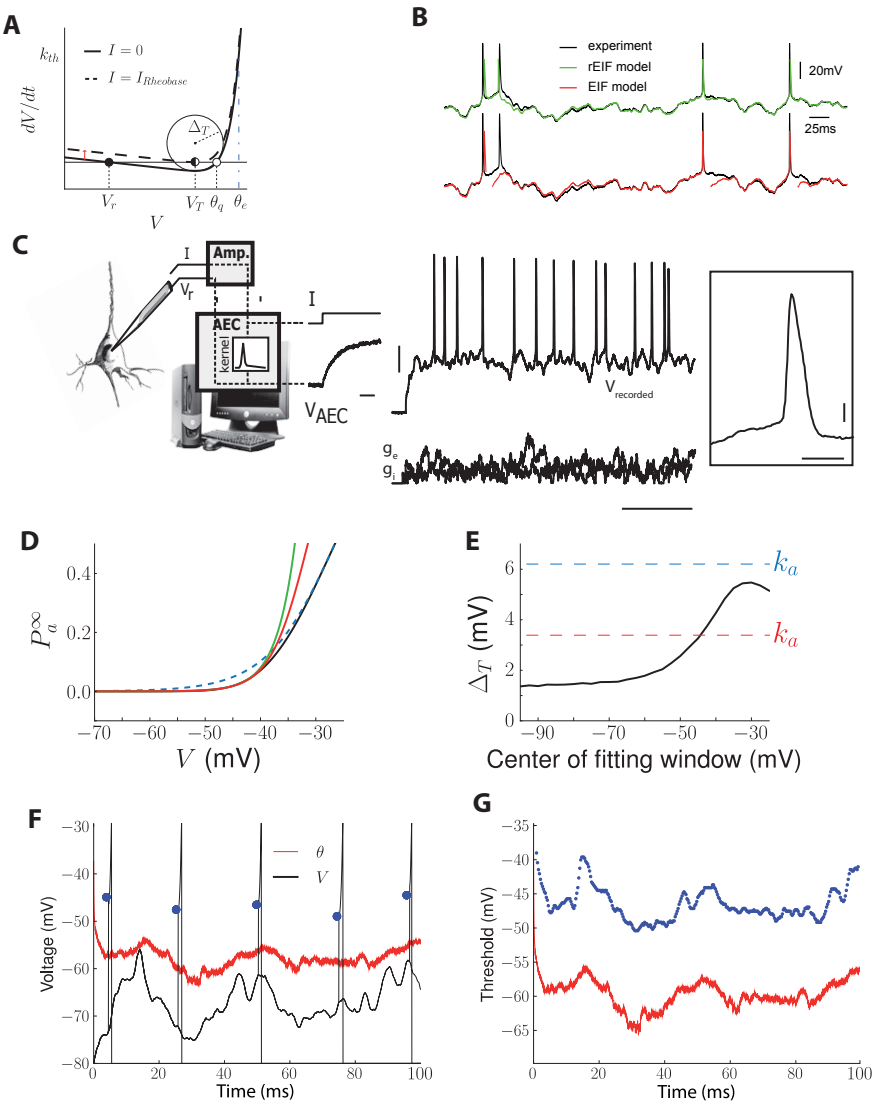


Figure 2: Spike initiation and spike threshold (D-F extracted from Platkiewicz and Brette (2009)). A. The I-V curve of the exponential integrate-and-fire (EIF) model. B. Variants of the EIF can be fit to *in vitro* traces in response to somatic current injection (from Badel et al. (2008)). C. I-V curves that include spike initiation as shown in A can be obtained with single-electrode recordings *in vitro* with the Active Electrode Compensation method, which consists in calibrating an electrode model using noise injection (Brette et al., 2008)). D. In standard biophysical models, Na activation curves (black) seems to be well fitted by Boltzmann functions on the entire voltage range (dashed blue), but fits are not so good in the spike initiation zone (red: Boltzmann fit in the spike initiation region; green: exponential fit). E. As a result, slope factors calculated from Boltzmann fits vary widely depending on the fit region. F. Measured (blue) and predicted (red) threshold in a simulated conductance-based model with fluctuating synaptic inputs (trace in black). G. Time-varying threshold: blue is measured, red is theoretical. The bias is partly predicted by the different definition for the threshold, and partly due to the fact that the activation curve was poorly fitted by an exponential function in that voltage region (close to half-activation voltage of the Na activation curve).

during the action potential (Meunier, 1992)). With a reset (ignoring inactivation and other ionic channels), we obtain the exponential integrate-and-fire model (Fourcaud-Trocme et al., 2003):

$$C\frac{dV}{dt} = g_L\Delta_T e^{(V-V_T)/\Delta_T} + g_L(E_L - V) + I$$

In this model, V_T is the voltage threshold for constant input currents I (i.e., such that $f'(V_T) = 0$, where $dV/dt = f(V)$) and Δ_T is the slope factor, which measures the sharpness of spikes (see Fig. 2A): in the limit $\Delta_T \to 0$ mV, the model becomes a standard integrate-and-fire model with threshold V_T. With spike-frequency adaptation, it predicts the response of cortical neurons to somatic injection with good accuracy, in terms of spike timings (Badel et al., 2008; Brette and Gerstner, 2005; Jolivet et al., 2008) (see Fig. 2B). I-V curves that include spike initiation (Fig. 2A,C) can be obtained with single-electrode recordings *in vitro* with the Active Electrode Compensation method, which consists in calibrating an electrode model using noise injection (Brette et al., 2008)) (see Fig. 2C).

Sharpness of spikes

Activation curves of Na channels measured in patch clamp show that the slope factor is about $\Delta_T = 6mV$, with little variability across channel types (Angelino and Brenner, 2007). When the I-V curve of cortical neuron is measured *in vitro* with fluctuating somatic current injection, it indeed fits the exponential model very well, but the slope factor is much smaller than expected: about 1.5 mV. This value is smaller than any reported value in the 40 patch clamp studies collected by Angelino and Brenner (2007). Jolivet et al. (2008) fitted an adaptive exponential integrate-and-fire model to *in vitro* recordings (somatic current injection) with very good results, but surprisingly the optimal slope factor was only 0.006 mV — in other words it was a standard adaptive integrate-and-fire model. We also tried to fit the same model to similar data from the 2009 INCF quantitative modelling competition and found that the optimal slope factor was about 5% of the difference between rest and threshold, meaning less than 1 mV. In fact similar or better results were obtained without the exponential nonlinearity. The winner of the competition used an IF model with adaptive threshold (Gerstner and Naud, 2009). This apparent contradiction was noted recently by Naundorf et al. (2006), who observed that spikes recorded in the soma of cortical neurons are much sharper than expected from Hodgkin-Huxley dynamics, which started a controversy (McCormick et al., 2007).

The reason for this surprising sharpness has in fact been known for quite a while: spikes are not initiated in the soma but in the axon hillock (axon initial segment, AIS) and then actively backpropagated to the soma (McCormick et al., 2007; Yu et al., 2008). This property is also seen in numerical simulations of multicompartmental models (Kole et al., 2008; Kole and Stuart, 2008). A two-compartment model could not account for the increased sharpness because the spike seen in the soma is a low-pass filtered version of the spike initiated in the

AIS, so that, if anything, it should look less sharp in the soma. Since the correct behavior appears in multicompartmental models, this discrepancy must be due to the fact that two compartments are not sufficient to model the active backpropagation of the action potential. Indeed, when Na and Kv1 channels open, the effective conductance increases, which reduces the electrotonic length λ[6]. Therefore, a two-compartment model is not valid anymore after spike initiation. The sharpening effect of backpropagation can be seen in the cable equation:

$$\tau_m \frac{\partial V}{\partial t} = E_L - V + \text{ionic currents} + \lambda^2 \frac{\partial^2 V}{\partial x^2}$$

It appears that the membrane equation is augmented by a diffusion term, which is positive and large in the rising phase of the action potential between the initiation site and the soma. Thus, for the same membrane potential V, the time derivative gets larger as this diffusion term increases, which sharpens action potentials. What is responsible of the sharpness of spikes seen at the soma is not only the fact that spikes are remotely initiated (this fact alone would predict the opposite effect), but also that they are actively back-propagated, with high sodium channel density between the soma and initiation site (not only at the initiation site).

Although the increased sharpness in the soma is due to the backpropagation of the action potential from the AIS (it looks softer at the initiation site), it cannot be considered completely artefactual because a model with sharp threshold predicts neuronal output more accurately than with softer threshold. Threshold variability, however, is still determined by channel properties at the initiation site, where spike initiation is soft. It is interesting to note that the fact that spikes are initiated in the axon rather than in the soma makes cortical neurons closer to integrate-and-fire models with hard threshold (considering only somatic current injection, not the effect of dendrites), and one might wonder whether this might be a desirable feature. It has been suggested that spikes are initiated in the axon hillock because it is energetically cheaper, but it is not clear that it would make a significant difference, because the initiation site is too close to the soma (less than 50 μm) to change the minimum current to elicit a spike. Another possible advantage is that it allows direct inhibitory control of spike output by synapses that specifically target the axon hillock (DeFelipe et al., 1985).

Threshold dynamics

The spike threshold varies very significantly *in vivo*. This property has been mainly attributed to sodium channel inactivation and potassium channels. We have recently looked at these hypotheses in biophysical models. It is a difficult problem to determine of the spike threshold when the input is fluctuating. Our approach is a quasi-static approximation: we consider that Na channel inactivation and voltage-gated conductances (e.g. K channels) are slow at the timescale of spike initiation (which is fast). With some elementary algebra, we find that the

[6] which is about 500–1000 μm at rest in the AIS, while spikes are initiated only 50 μm away from the soma.

membrane equation reads:

$$C\frac{dV}{dt} = g_{tot}\Delta_T e^{(V-\theta)/\Delta_T} + g_{tot}(E_L - V)$$

with

$$\theta = V_T - \Delta_T \log h \frac{g_L}{g_{tot}}$$

where g_{tot} is the total non-sodium conductance and h is the Na inactivation variable ($1-h$ is the proportion of inactivated Na channels). In the quasi-static approximation, θ is indeed the voltage threshold. This equation (which we called the *threshold equation*) provides an instantaneous value for the threshold, as a function of Na channel inactivation and other voltage-gated channels.

We compared our theoretical prediction with the empirically measured threshold in a standard biophysical conductance-based model (Fig. 2F,G). First of all, that biophysical model, as other models we have tried, does not exhibit much threshold variability. To increase threshold variability, we hyperpolarized Na inactivation by 12.5 mV. The threshold equation could account for 83 % of the variance. The differences between theory and measurement (in simulation) had two causes: the first one is that threshold definition is different (short pulses vs. slow inputs), but this difference can be calculated and compensated for; the second one is that Na activation curves found in most models are in fact not so well fitted by Boltzmann function, as in shown in Fig. 2D,E. Depending on the voltage region where the curve is fitted, the slope factor Δ_T ($= k_a$) varies between 1.5 mV and almost 6 mV. When fitted over the entire voltage range, one finds $k_a \approx 6$ mV, which is close to experimentally reported values (similarly obtained) However, in the spike initiation region it is closer to 1.5 mV, which implies low threshold variability, but it is not clear whether this is also the case in patch clamp measurements of Na currents. It might be that this low value of the slope factor near spike initiation comes from attempts to fit the shape of action potentials in single-compartment models, compensating for the absence of active backpropagation, but this property comes at the expense of lower threshold variability. Therefore, I suggest that the low threshold variability in most standard biophysical models comes from an anomaly in the Na activation curve of the model, which might be resolved by rexamining patch clamp measurements near spike initiation.

If we assume that h evolves according to a first-order kinetic equation, then it is possible to describe the dynamics of the threshold θ by a differential equation which depends on V, and can be approximated as

$$\tau_h(V)\frac{d\theta}{dt} = \theta_\infty(V) - \theta$$

where $\theta_\infty(V) = V_T - \Delta_T \log h_\infty(V)$ is the equilibrium value of the threshold. A similar equation can be obtained when considering the effect of voltage-gated conductances (e.g. potassium channels). The effect of an emitted action potential on h is a partial reset: $h \to he^{-\delta t/\tau_h}$, where δt is the duration of the spike. This reset translates to a shift for the threshold: $\theta \to \theta + (\delta t/\tau_h)\Delta_T$. In other words, the spike threshold increases by a fixed amount after each spike. This

effect was recently demonstrated *in vitro* (Badel et al., 2008) and explains *in vivo* observations where the threshold was found to be inversely correlated with the previous interspike interval (Henze and Buzsaki, 2001). In summary, we obtain the following integrate-and-fire model with dynamic threshold:

$$\begin{aligned} \frac{dV}{dt} &= g_L(E_L - V) + I \\ \tau_h(V)\frac{d\theta}{dt} &= \theta_\infty(V) - \theta \end{aligned}$$

and a spike is produced when V reaches θ. After spiking, both variables are reset: $V \to V_r$, $\theta \to \theta + \delta\theta$.

Effective postsynaptic potential

What is the effect of threshold adaptation on synaptic integration? There is a simple way to address this issue when the threshold equation and the membrane equation are linear (i.e., $\theta_\infty(V) = aV+b$). In this case, in response to a sequence of input spikes, the membrane potential is a superposition of postsynaptic potentials (PSPs):

$$V(t) = E_L + \sum_i \mathrm{PSP}_i(t - t_i)$$

where t_i is the time of incoming spike i, and so is the value of the threshold:

$$\theta(t) = \theta_0 + \sum_i \mathrm{PST}_i(t - t_i)$$

where PST_i is the postsynaptic threshold for incoming spike i, which is simply a low-pass filtered version of the PSP (with filtering time constant τ_h). A spike is produced when $V(t)$ reaches $\theta(t)$, i.e., when

$$\sum_i (\mathrm{PSP}_i(t - t_i) - \mathrm{PST}_i(t - t_i)) = \theta_0 - E_L$$

We call the difference the PSP $-$ PST the *effective postsynaptic potential*, which is the postsynaptic potential that is left when the effect of threshold adaptation is discounted. It appears that the effect of threshold adaptation is similar to a slightly delayed inhibition (with approximate delay τ_h), meaning that it shortens the effective integration time constant.

2.3 Electrophysiological classes

Integrate-and-fire models (with some form of spike-frequency adaptation) seem to be good models for regular spiking cortical neurons, but these simple models cannot account for the variety of electrophysiological behaviors of real neurons (see e.g. (Markram et al., 2004) for interneurons). Izhikevich (2003) introduced a two-variable spiking model based on the quadratic model augmented by an adaptive equation. Despite its simplicity, it can reproduce a large number of electrophysiological signatures such as bursting or regular spiking. Different sets

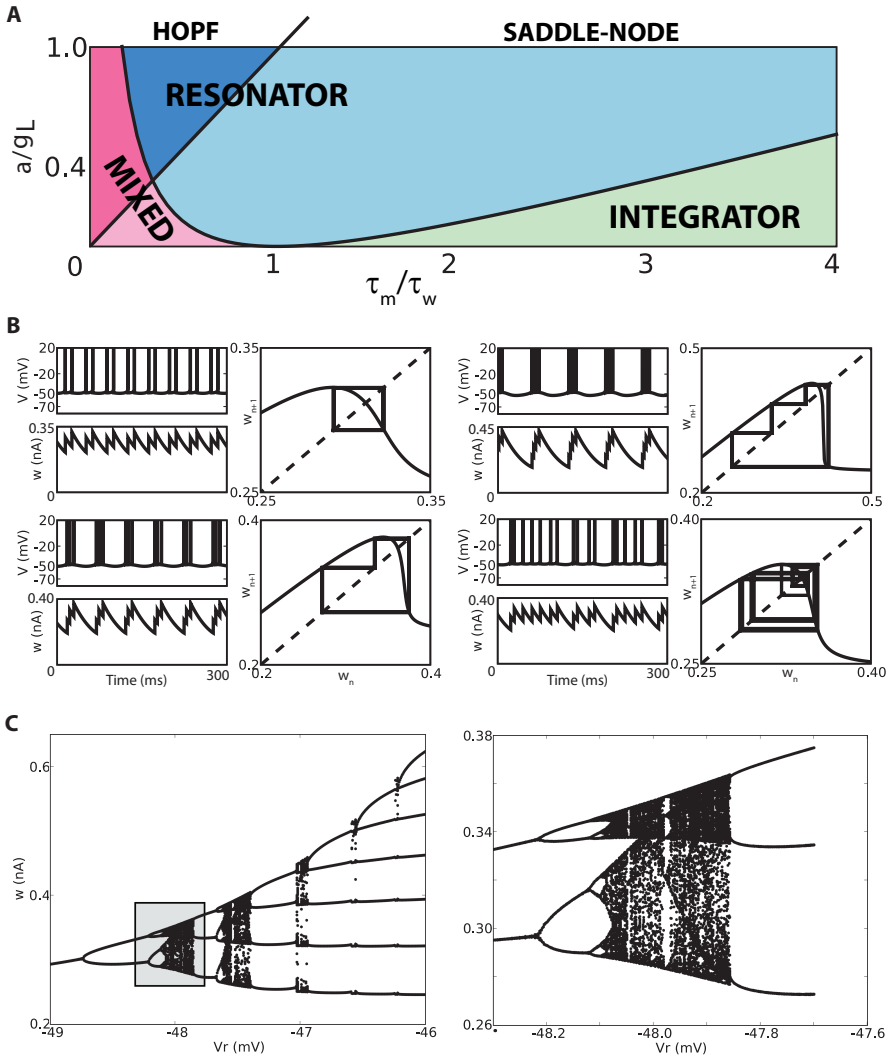


Figure 3: Neuron classes in the adaptive integrate-and-fire model (extracted from Touboul and Brette (2008)). A. Subthreshold behavior of the model as a function of a/g_L and τ_m/τ_w. Light (dark) colors indicate class A (class B) parameters. Blue indicates resonator mode (oscillations for any or almost any I). Green indicates integrator mode (oscillations for any I). Pink indicates mixed mode (resonator if I is large enough, otherwise integrator). B. Bursting and chaos. Each panel shows a sample response (V and w) from the model, with different values of the reset V_r, and the adaptation map Φ, which maps the value of the adaptation variable w at spike time to the value at the next spike time. A burst with n spikes corresponds to an n-periodic orbit under the adaptation map Φ. The last spike of each burst occurs in the decreasing part of Φ, inducing a slower trajectory. The last panel shows a chaotic behavior. C. Bifurcation structure with increasing V_r. Left: bifurcation diagram showing a period adding structure (orbits under the adaptation map Φ with varying values for V_r). Fixed points indicate regular spiking, periodic orbits indicate bursting, dense orbits indicate chaos. Right: zoom on the bifurcation diagram (as indicated by the shaded box), showing a period doubling structure.

of parameter values correspond to different electrophysiological classes. Brette and Gerstner (2005) proposed a variant of this model, the adaptive exponential integrate-and-fire model (AdEx), in which the quadratic equation is replaced by an exponential equation. Although these two models are qualitatively similar, the AdEx model is quantitatively more accurate for time-varying inputs. The reason is that the quadratic model is an abstract model that is equivalent to any type I neuron model near bifurcation, for constant injected currents. Therefore, it applies to near threshold inputs that change slowly compared to the typical interspike interval. The situation is very different *in vivo*, where spikes are triggered by fast fluctuations in synaptic current (Destexhe et al., 2003; Piwkowska et al., 2008).

The AdEx model is described by two variables, the membrane potential V and an adaptation current w, whose dynamics are governed by the following differential equations:

$$\begin{cases} C\frac{dV}{dt} &= -g_L(V-E_L) + g_L\Delta_T \exp\left(\frac{V-V_T}{\Delta_T}\right) - w + I \\ \tau_w \frac{dw}{dt} &= a(V-E_L) - w \end{cases} \tag{1}$$

When the membrane potential V is high enough, the trajectory quickly diverges because of the exponential term. This divergence to infinity models the spike (the shape of the action potential is ignored, as in the standard integrate-and-fire model). When a spike occurs, the membrane potential is instantaneously reset to some value V_r and the adaptation current is increased: $V \to V_r$, $w \to w + b$. Although the differential system is only two-dimensional, the reset makes the resulting hybrid system very rich. The AdEx model can reproduce many known electrophysiological features: spike-frequency adaptation, regular and fast spiking, phasic spiking, phasic and tonic bursting, post-inhibitory spiking and bursting, delayed spike inititation and delayed burst initiation, damped oscillations, overshoot or undershoot of the voltage in response to a subthreshold current step, type I and type II excitability.

The subthreshold behavior of the model can be analyzed using standard dynamical systems techniques. We were able to classify if as a function of two quantities (Touboul and Brette, 2008): a/g_L (ratio of adaptation and leak conductances) and τ_m/τ_w (ratio of adaptation and membrane time constants, where $\tau_m = C/g_L$), as shown on Fig. 3A. Light colors indicate loss of stability via saddle-node bifurcation and dark colors via subcritical Andronov-Hopf bifurcation. Subcritical Andronov-Hopf corresponds to f-I curves of type II, whereas saddle-node bifurcations can lead to type I or II. The blue region corresponds to the resonator mode (oscillations for any or almost any input current I), while the green region corresponds to the integrator mode (no oscillation for any I) and the pink region is the mixed mode (resonator if I is large enough, otherwise integrator). Similar results are obtained in integrate-and-fire models with an additional adaptive equation (Brunel et al., 2003).

To analyze the spiking patterns, we introduced the *adaptation map* Φ (Touboul and Brette, 2008): because V is always reset to the same value V_r after a spike, interspike intervals are determined by the value of the adaptation variable w at spike time. The sequence (w_n) of these values is the orbit of w_0 under the adapta-

tion map Φ, which maps w to the value of $w(t)$ at spike time plus b for a solution starting from (V_r, w). Thus, $w_n = \Phi^n(w_0)$. Regular spiking corresponds to a stable equilibrium of Φ while bursting corresponds to cycles of Φ. Examples are shown on Fig. 3B,C.

3 Mathematics of spiking models

3.1 General results

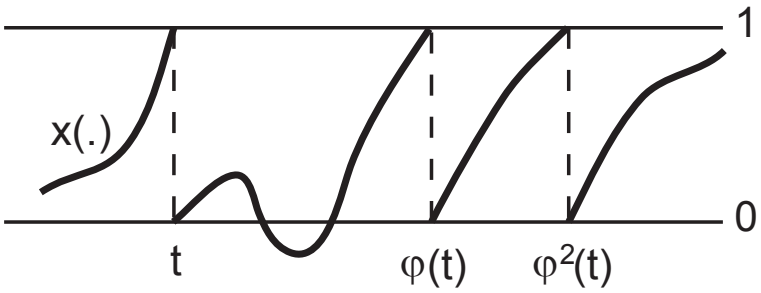


Figure 4: The spike map φ: $\varphi(t)$ is the time of the next spike for a trajectory starting from reset at time t.

Spiking models such as integrate-and-fire models are complicated mathematical objects, because although many such models are defined by simple (and often linear) differential equations, the threshold condition introduces discontinuities, which complicates their dynamics. I have studied the dynamical properties of one-dimensional integrate-and-fire models with time-varying inputs (Brette, 2004; Brette and Guigon, 2003; Brette, 2008), which are defined by a differential equation governing the dynamics of the membrane potential x:

$$\frac{dx}{dt} = f(x,t) \tag{2}$$

and a reset: when $x(t)$ reaches a threshold x_t, then a spike is produced and $x(\cdot)$ is instantaneously reset to x_r. Up to a change of variables, one can set $x_t = 1$ and $x_r = 0$. The conditional reset makes this model a hybrid dynamical system. This definition encompasses many types of integrate-and-fire models, including models with conductance-based synapses. However, it does not include models in which emitted spikes influence the subsequent dynamics (i.e., spike-frequency adaptation).

I showed that two particular classes of spiking models have especially interesting properties (Brette, 2004; Brette and Guigon, 2003):

- *leaky models*, such that $\frac{\partial f}{\partial x} \leq \alpha < 0$;
- *reflecting models*, such that $f(0,t) > 0$ for all t.

Standard integrate-and-fire models are leaky (and $\frac{\partial f}{\partial x} = -g/C$, where g is the total conductance and C is the membrane capacitance), the quadratic model (Ermentrout and Kopell, 1986) is reflecting. Models of these two classes have a unique firing rate (independent of initial condition).

An important mathematical object for these models is the *spike map* φ, which is defined such that a spike train produced by the model is the orbit of the first spike time under φ (Fig. 4). More precisely, $\varphi(t)$ is the minimal $s \geq t$ such that the forward solution starting at reset at time t reaches threshold at time s. For leaky and reflecting models, the spike map is (strictly) increasing on its range but often discontinuous. This is a fundamental property which has implications for the reliability of spike timing and phase locking to periodic inputs.

3.2 Reliability of spike timing

The responses of neurons to dynamic stimuli have been shown to exhibit high reliability *in vitro* (Mainen and Sejnowski, 1995; Hunter et al., 1998; Fellous et al., 2001; Beierholm et al., 2001) and *in vivo* (Berry et al., 1997; Nowak et al., 1997; Sanchez-Vives and McCormick, 2000; Reich et al., 1997; Berry and Meister, 1998; Bair and Koch, 1996). In this case, spike timing is reproducible on a trial-by-trial basis up to a precision of 1 ms or less, even a long time after stimulus onset (1 s in Mainen and Sejnowski (1995)). This is a non-trivial property because one might expect internal noise to accumulate as is the case with constant inputs (if the interval between two successive spikes is $\Delta + \xi$, where ξ is some internal noise, then the variability in the timing of the n^{th} spike increases as $\sqrt{n}$). It is in fact a convergence property of the class of spiking models defined above, which is robust to noise, changes in initial condition and small parameter changes (Fig. 5A).

Mathematically proving this property turns out to be very difficult. In (Brette and Guigon, 2003), I proposed an explanation for balanced inputs. When the input is balanced, the current at threshold is often negative: $f(x_t, t) < 0$. In all the time intervals where it occurs, no spike is possible (Fig. 5B). In other words, spiking is restricted to certain time intervals, independently of the initial condition. Using the fact the spike map is increasing (on its range), each such interval maps to an infinite number of intervals where spiking is also not possible. When constructing these forbidden spiking intervals, the size of the remaining set seems to go to zero, meaning that after some time, spiking is constrained in smaller and smaller sets (Fig. 5C). Although the construction is convincing, the proof is not complete because one would need to prove that the measure of the set of allowed spike times tends to zero, which is hard to do without further assumptions.

In (Brette, 2008), I addressed the problem from a different angle, asking an elementary dynamical system question: for a given initial condition, is there a unique spiking solution defined on $\mathbb{R}$? The answer is straightforward and positive on $\mathbb{R}^+$ (forward solutions), but not on $\mathbb{R}^-$ (backward solutions). It turns out that for each initial condition, there is a countable (and possibility finite) number of backward trajectories, with an increasing number of previous spikes. I called the maximum number of previous spikes the *degree* of an initial condition (Fig. 5D). I showed that for a model with noisy synaptic conductances, the degree of

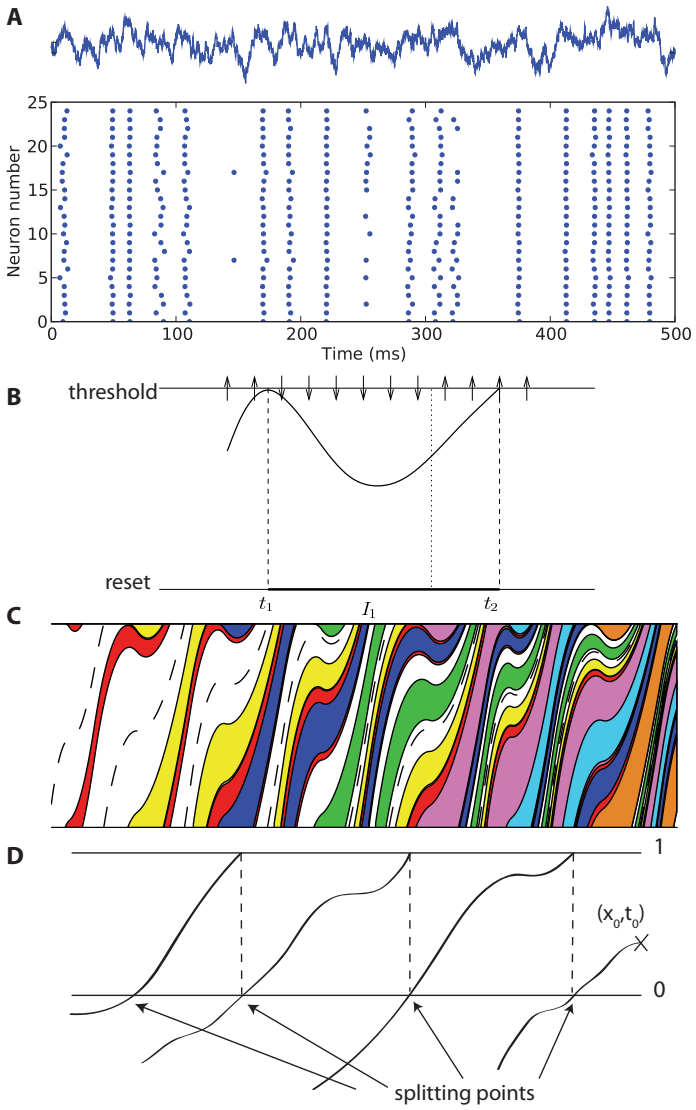


Figure 5: Reliability of spike timing in spiking models. A. When a spiking neuron is driven by a time-varying input (top), its output spike trains are reproducible over repeated trials (bottom, simulation with a noisy leaky integrate-and-fire model). B. The sign of the current at threshold is shown with arrows. At the time when sign changes, there is a trajectory that comes tangential to the threshold. The entire region between this trajectory and threshold cannot be reached by any other trajectory, which forbids any trajectory to spike in the time interval (t_1, t_2). C. By following all trajectories starting from that forbidden interval, we obtain an unbounded region (red) which cannot be entered by any trajectory. All trajectories are then constrained to spike in the remaining region (white), which gets smaller and smaller as new forbidden intervals appear. D. An alternative explanation is that most possible spike times correspond to trajectories which have only a few previous spikes. Starting from initial condition $x(t_0) = x_0$, we can construct several possible backward trajectories with an increasing number of spikes, which is bounded by a number called the *degree* of the initial condition.

initial conditions ($x(t) = x_r$) is geometrically distributed. In other words, as the number of spikes in a spike train increases, spike times are constrained in a smaller and smaller set whose size decreases exponentially. I conjectured that the set of points with infinite degree is just a single spike train. This would make the transformation input spike trains $\mapsto$ output spike train a function. However, this point is still lacking a sound mathematical proof.

Reliability of spike timing is a very important property for spike-based computation, because it implies that two spiking models receiving similar inputs should synchronize. Together with the dual property, which is coincidence detection, it implies that spiking models are well equipped to perform an important elementary operation: detecting similarities (see section 5). Before addressing the problem of coincidence detection in spiking neurons, I will shortly discuss how spiking models encode periodic inputs.

3.3 Phase locking

Phase locking, also called *mode locking*, is the property of spiking models to respond to periodic inputs with periodic spike trains, at particular phases of the input signal. In the auditory litterature, phase locking often refers to the notion of phase preference, i.e., the fact the distribution of spike phases with respect to the input signal is sharp.

The response of spiking models to periodic inputs can also be understood using the spike map φ (Brette, 2004; Brette and Guigon, 2003). Since $\varphi(t + T) = \varphi(t) + T$ and φ is increasing (more precisely, on its range), the spike map is the lift of an orientation-preserving circle map. When it is continuous, this map is a homeomorphism of the circle, which is a very well known mathematical object (Denjoy, 1932; Coddington and Levinson, 1955; Arnold, 1961; Herman, 1977; Keener, 1980; Veerman, 1989). It appears that the sequence of phases of successive spikes is an orbit under the circle map. Therefore a fixed point or a periodic point of the circle map corresponds to a periodic spike pattern for the spiking model. The dynamics of leaky spiking models depends on the value of the *rotation number*, defined as the ratio of the input frequency to the output spike rate:

- It is rational if and only if the output pattern of spikes is (asymptotically) periodic, the period being a multiple of the input period. When this pattern is stable under perturbations, this is called phase locking or mode locking, which has also been studied experimentally (Rescigno et al., 1970; Ascoli et al., 1977; Guttman et al., 1980; Koppl, 1997). There is $p : q$ phase locking when a stable pattern of p spikes is produced every q periods of the input. Thus when the model is phase-locked, noise does not accumulate over time. However, several stable responses may coexist: if a $p : q$ phase-locked solution is shifted by a multiple of the input period, a new solution is obtained, so that at least q distinct stable solutions exist (Fig. 6C, middle: 1:1 phase locking, bottom: 2:3 phase locking).

- When it is irrational, under some regularity conditions, the dynamics of the

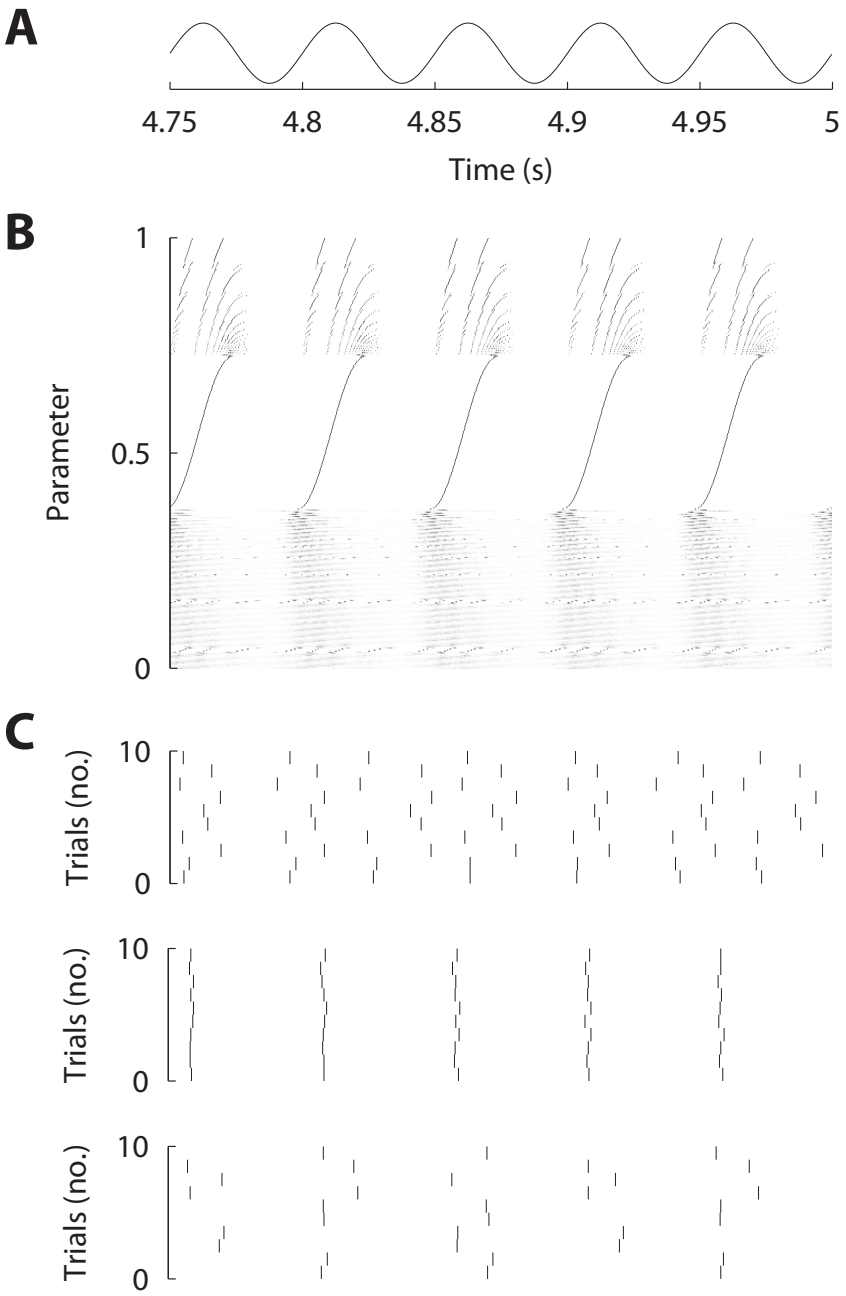


Figure 6: Phase locking (or mode locking) in spiking models (adapted from Brette and Guigon (2003)). A. Input current is a 20 Hz sine wave with varying mean and amplitude. B. For each input parameter value p, the model was run 2000 times with random initial potential and firing probability is shown as grey level for $t > 4.75$ s. C. For three different parameter values (0, 0.5, 1), spike trains for 10 trials of the model with noise and random initial potential: top: no phase locking; middle: 1:1 phase locking; bottom: 2:3 phase locking (2 spikes every 3 periods).

model are topologically equivalent to the dynamics of a model with constant input, which means that noise always accumulates over time, however small it may be (Fig. 6C, top).

Although this dynamical theory was initially developed for homeomorphisms of the circle, it also applies when the circle map is discontinuous (Brette, 2003). Interesting properties appear when one looks at how spike patterns change with a parameter (for example the input mean). It turns out that there is a qualitative difference between the case when the input current is entirely above threshold and when it crosses it (e.g., balanced input). When the current is entirely above threshold, the rotation number is irrational on a positive measure set of parameters (Herman, 1977), meaning that spike trains are sometimes *not* reproducible. In the other case (balanced input), that set has measure zero, and thus is not observable (Keener, 1980) (see Veerman (1989) for a rigorous mathematical proof): in other words, phase locking is always observed. This appears on the top half of Fig. 6B.

This theory should be directly relevant to the study of auditory neuron responses in the brainstem, but surprisingly it has only be applied very recently (Laudanski et al., in preparation).

3.4 Coincidence detection

We have seen in section 3.2 that the spike trains produced by spiking models in response to time-varying inputs are reproducible, which implies that similar neurons receiving the same inputs should synchronize (feedforward synchronization). If this property is to have any impact at all, neurons must be able to detect coincidences between spikes. Intuitively, it seems clear that a leaky neuron should be more likely to spike in response to two coincident spikes than to two spikes that are far away compared to the membrane time constant, because the effect of the first spike on the membrane potential has vanished when the second spike arrives. However, it is harder to quantify it outside of this toy situation, e.g. in active neurons with a background of fluctuating synaptic activity. Fig. 7A shows that spiking models can be very sensitive to the correlation of their input spike trains. Clearly, this can occur only if the input is balanced, i.e., if the temporal average of the total input is subthreshold, since otherwise the output firing rate is mainly determined by the mean input and therefore is insensitive to correlations.

To understand the coincidence detection properties of active neurons, I propose a new approach based on several observations about the properties of cortical neurons in vivo: the membrane time constant is short (a few ms) compared to typical interspike intervals; the membrane potential is stochastic, with a mean well below threshold and the standard deviation is several times smaller than the average distant to threshold (the distribution may not be Gaussian). Then the coincidence detection effect can be quantitatively understood as illustrated in Fig. 7B. If the membrane potential probability is decreasing near threshold, then two synchronous incoming spikes will be more likely to trigger a spike than two distant ones. This extra firing probability can be calculated as a function of the membrane potential distribution and the post-synaptic potentials, and it applies to tight correlations (shorter than the integration time constant). Interestingly, it appears

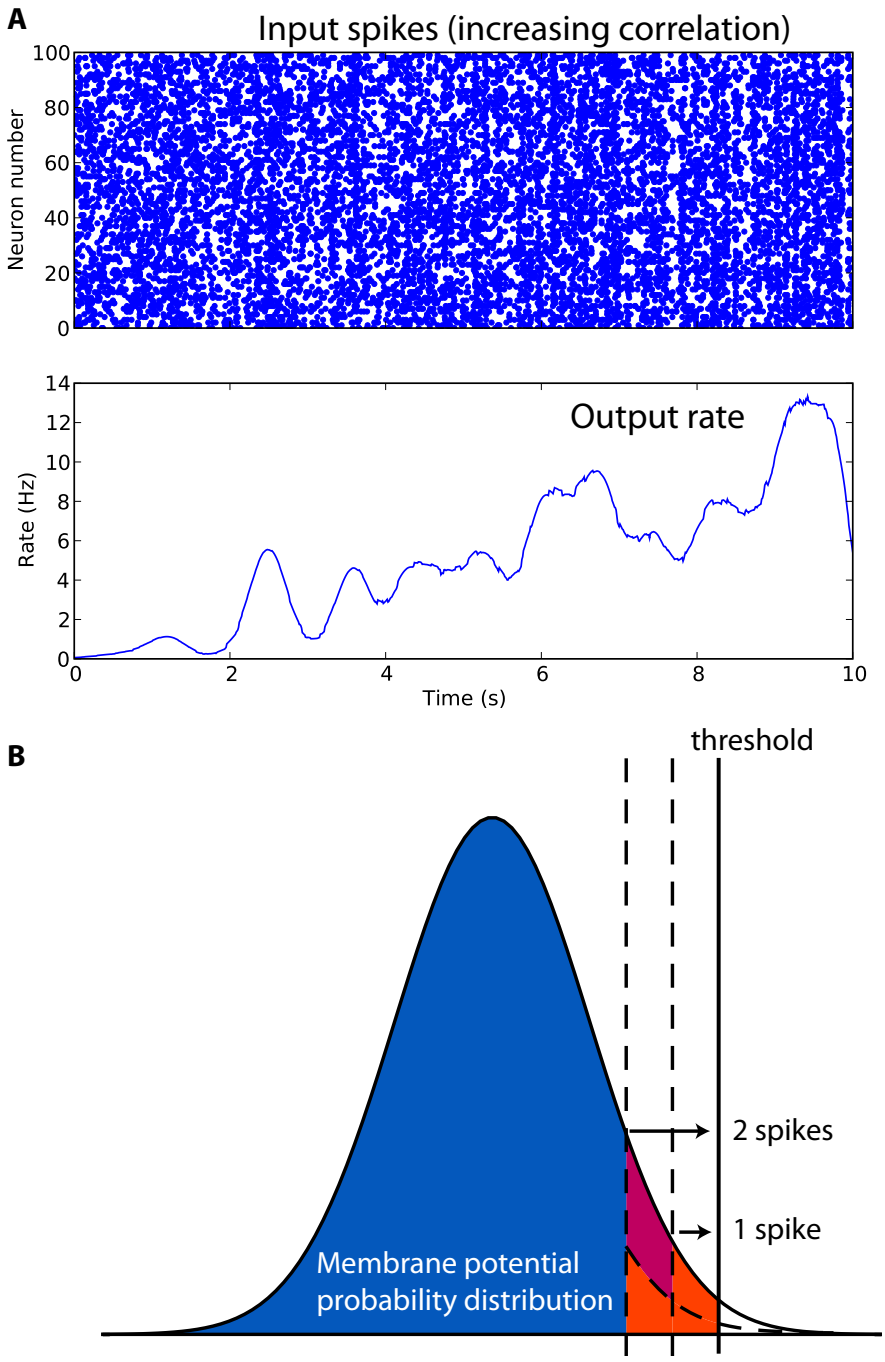


Figure 7: Coincidence detection in spiking models. A. An integrate-and-fire model receives 100 correlated input spike trains with increase correlation (generated with algorithms from (Brette, 2009)). The output firing rate increases with input correlation. B. Coincidence detection in active neurons. The purple area is the extra firing probability with two input synchronous spikes, compared to two non-interacting spikes.

that the background activity modulates the coincidence detection properties. The approach can be extended to more complex models by looking at the distribution of the stimulation parameter (current or conductance) instead of the potential.

With Cyrille Rossant, we started to study coincidence detection properties using this approach and we were able to quantify the time constant for coincidence detection when the membrane potential distribution reflects slow fluctuations. We found that this time constant can be expressed as the product of the time constant of the post-synaptic potential and of a factor defined by the membrane potential distribution, which is always smaller than 1 (work in progress). In other words, spiking models are more sensitive to fine correlations than expected from the value of the membrane time constant.

4 Simulation of spiking neuron models

4.1 Algorithms

There are two families of algorithms for the simulation of neural networks: synchronous or "clock-driven" algorithms, in which all neurons are updated simultaneously at every tick of a clock, and asynchronous or "event-driven" algorithms, in which neurons are updated only when they receive or emit a spike. These two families of algorithms were reviewed in (Brette et al., 2007), here I will briefly give an overview of complexity issues.

Algorithmic complexity

First of all, how much time can it possibly take for a good algorithm to simulate a large network? Suppose there are N neurons whose average firing rate is F and average number of synapses is p. If all spike transmissions are taken into account, then a simulation lasting 1s (biological time) must process $N \times p \times F$ spike transmissions. The goal of efficient algorithm design is to reach this minimal number of operations (of course, up to a constant multiplicative factor). Note that if the simulation is not restricted to spike-mediated interactions, e.g. if the model includes gap junctions or dendro-dendritic interactions, then the optimal number of operations can be much larger.

There are two techniques that are common to both types of strategies to minimize the number of operations. The first one is to express the model as a hybrid system, as previously described. With this formalism, it appears clearly that spike times need not be stored (except of course if transmission delays are included), even though it would seem so from more phenomenological formulations. For example, in the Spike Response Model (Gerstner and Kistler, 2002), it seems that one would need to calculate the sum over all synapses at each timestep, but this requirement disappears in the differential hybrid system form, where the cost of a spike transmission is typically just one operation (for one synapse). The second common strategy is to use linearities to reduce the number of equations. In general, the number of state variables of a neuron (length of vector $\mathbf{X}$) scales with the number of synapses, since each synapse has its own dynamics. This fact

constitutes a major problem for efficient simulation of neural networks, both in terms of memory consumption and computation time. However, several authors have observed that all synaptic variables sharing the same linear dynamics can be reduced to a single one (Wilson and Bower, 1989; Bernard et al., 1994; Lytton, 1996; Song et al., 2000). Some models of spike-timing dependent plasticity (with linear interactions between pairs of spikes) can also be simulated in this way (see e.g. Abbott and Nelson (2000)). However, some important biophysical models are not linear and thus cannot benefit from this optimization, in particular NMDA-mediated interactions and saturating synapses.

Clock-driven vs. event-driven

In a synchronous or "clock-driven" algorithm, the state variables of all neurons (and possibly synapses) are updated at every tick of a clock: $\mathbf{X}(t) \rightarrow \mathbf{X}(t+dt)$. With non-linear differential equations, one would use an integration method such as Euler or Runge-Kutta (Press et al., 1993) or, for Hodgkin-Huxley models, implicit methods (Hines, 1984); linear equations can be integrated exactly (each update is then just a matrix product). Then, after updating all variables, the threshold condition is checked for every neuron. Each neuron that satisfies this condition produces a spike which is transmitted to its target neurons, updating the corresponding variables ($\mathbf{X} \leftarrow g_i(\mathbf{X})$). For integrate-and-fire models, the membrane potential of every spiking neuron is reset. The total computational cost per second of biological time is of order

$$\begin{array}{ccc} \textit{Update} & + & \textit{Propagation} \\ c_U \times \dfrac{N}{dt} & + & c_P \times F \times N \times p \end{array}$$

where c_U is the cost of one update and c_P is the cost of one spike propagation; typically, c_U is much higher than c_P but this is implementation-dependent. Therefore, for very dense networks, the total is dominated by the propagation phase and is linear in the number of synapses, which is optimal. However, in practice the first phase is negligible only when the following condition is met: $\frac{c_P}{c_U} \times F \times p \times dt >> 1$, which is often not true in many practical situations.

The obvious drawback of clock-driven algorithms as described above is that spike timings are aligned to a grid (ticks of the clock), thus the simulation is approximate even when the differential equations are computed exactly. Other specific errors come from the fact that threshold conditions are checked only at the ticks of the clock, implying that some spikes might be missed. To solve this problem, one must use exact event-driven algorithms in which spike timings are computed exactly. Another advantage is a potential gain in speed due to not calculating many small update steps for a neuron in which no event arrives. In an asynchronous or "event-driven" algorithm, the simulation advances from one event to the next event. Events can be spikes coming from neurons in the network or external spikes (typically random spikes described by a Poisson process). Such algorithms typically use complex data structures to handle events (efficient queues) and elaborate algorithms to calculate spike times (e.g. (Brette, 2007) for

models with exponential currents and (Brette, 2006) for models with exponential conductances). With efficient structures, the total computational cost per second of biological time is of order $F \times N \times p$, which is optimal. However, in practice, the constant implementation factor is very large because the algorithms are complex, so that clock-driven algorithms are typically faster for dense networks (but slower when the update phase dominates the total cost). The main drawback of exact event-driven methods is that they are constrained to relatively simple types of neuron models. For example, Hodkgin-Huxley models cannot be simulated in this way.

In summary, event-driven algorithms are more precise than clock-driven algorithms and have optimal computational complexity. However, this comes at a cost of greater implementation complexity, which implies that they are less flexible and applicable (not all models can be simulated) and they are typically slower for dense networks (although only by a constant factor). Another advantage of clock-driven algorithms is that they can be simulated with vectorization techniques, which we have used to develop the Brian simulator (Goodman and Brette, 2008).

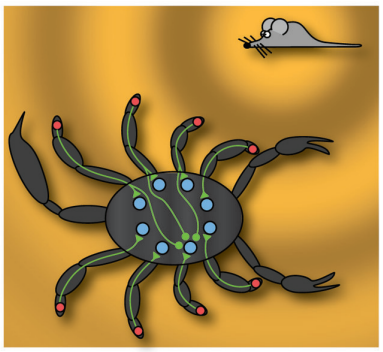
4.2 Brian: a spiking neural network simulator

Several successful neural simulators already exist (Brette et al., 2007), such as Neuron (Carnevale and Hines, 2006) and Genesis (Bower and Beeman, 1998) for compartmental modelling, and NEST (`http://www.nest-initiative.org`) for large scale network modelling. These simulators implement computationally efficient algorithms and are widely used for large scale modelling and complex biophysical models. However, computational efficiency is not always the main limiting factor when simulating neural models. In many practical cases, it takes considerably more time to write the code than to run the simulations. We developed the Brian simulator (`http://www.briansimulator.org`) to minimize learning and development time, while maintaining reasonable simulation efficiency (Goodman and Brette, 2008).

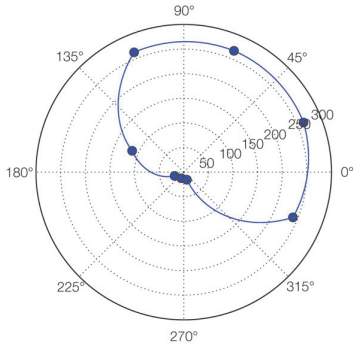
Brian is an extension package for the Python programming language (Goodman and Brette, 2008). A simulation using Brian is a Python program either executed from a script or interactively from a Python shell. The primary focus is on making the development of a model by the user as rapid and flexible as possible. For that purpose, the models are defined directly in the main script in their mathematical form (differential equations and discrete events). This design choice addresses three issues: flexibility, as the user can change the model by changing the equations; readability, as equations are unambiguous and do not require any specific knowledge of the Brian simulator to understand them; and simplicity of the syntax, as models are expressed in their original mathematical form, with little syntax to learn that is specific to Brian. Computational efficiency is achieved using vector-based computation techniques.

Figure 8 shows a Brian script adapted from (Stürzl et al., 2000), which models the neural mechanisms of prey localization by a sand scorpion. This model is fairly complex and includes in particular noise and delays, which would make equivalent code in Matlab or C very long, whereas the full script takes only about 20 lines

A



B



C

```
def wave(t):
    return .2*sin(2*pi*300*Hz*t)*cos(2*pi*25*Hz*t)
eqs_legs = '''                                          # Leg mechanoreceptors
dv/dt = (1+wave(t-d)-v)/tau+sigma*(2/tau)**.5*xi : 1  # xi is white noise
d : second                                              # wave delay
'''
legs = NeuronGroup(8, eqs_legs, threshold='v>1', reset='v=0', refractory=1*ms)
legs.d = radius/wave_speed*(1-cos(prey_angle-legs_angle))
eqs_neuron = '''        # Command neurons
dv/dt = (x-v)/tau : 1
dx/dt = -x/tau    : 1  # PSPs are alpha functions
'''
neurons = NeuronGroup(8, eqs_neuron, threshold='v>1', reset='v=0')
synapses_ex = IdentityConnection(legs, neurons, 'x', weight=4)
synapses_inh = Connection(legs, neurons, 'x', delay=.7*ms)
for i in range(8):
    synapses_inh[i, (4+i-1)%8] = -1
    synapses_inh[i, (4+i)%8] = -1
    synapses_inh[i, (4+i+1)%8] = -1
run(300*ms)
```

Figure 8: Brian implementation of a model of prey localisation in the sand scorpion, adapted from Stürzl et al. (2000). The movement of the prey causes a surface wave (function wave in the code in panel C) which is detected by mechanoreceptors (the red points in panel A) at the ends of each of the scorpion's legs. The mechanoreceptors are modelled by noisy, leaky integrate-and-fire neurons with an input current defined by the surface wave displacement at the ends of the legs (object legs in the code, defined by the equations eqs_legs). These neurons send an excitatory signal (the object synapses_ex) to corresponding command neurons (the blue points) modelled by leaky integrate-and-fire neurons (object neurons with equations eqs_neuron), which also receive delayed inhibitory signals (the object synapses_inh) from the three legs on the other side (the for loop). A wave arriving first at one side of the scorpion will cause an excitatory signal to be sent to the command neurons on that side causing them to fire, and an inhibitory signal to the command neurons on the other side, stopping them from firing when the wave reaches the other side. The result is that command neurons are associated to the directions of the corresponding legs, firing at a high rate if the prey is in that direction. Panel B shows the firing rates for the 8 command neurons in a polar plot for a prey at an angle of 60 degrees relative to the scorpion.

with Brian (plus the definition of parameter values). The script illustrates the fact that the code is close to the mathematical definition of the model, which makes it relatively easy to understand.

Computational efficiency

Brian can achieve very good performances by using the technique of vectorisation, similar to the same technique familiar to Matlab users. The idea is to replace loops by operations on large vectors, so that the interpretation overhead becomes negligible. Brian uses vectorisation for both the simulation and the construction of the model (e.g., initialisation of synaptic weights).

For example, for a single neuron i with state vector $\mathbf{x}_i$, the update step from $\mathbf{x}_i(t)$ to $\mathbf{x}_i(t+dt)$ might be $\mathbf{x}_i(t+dt) = M\mathbf{x}_i(t) + \mathbf{b}$ for a matrix M and vector $\mathbf{b}$. This operation is the same for every i so rather than looping through all the neurons carrying out the same operation, we write a state matrix S whose columns are the state vectors of each neuron. Now the loop carrying out the operation for each neuron i can be written in one operation, $S(t+dt) = MS(t) + B$ (where B is a matrix with every column equal to b). The number of mathematical operations is the same, but the interpretation overhead is reduced from N interpretation operations for N neurons to 1 interpretation operation. Brian uses the NumPy package for these vectorised operations. NumPy is written in optimised C code, and for linear algebraic operations uses the Basic Linear Algebra Subprograms (BLAS) application programming interface (API). This means that NumPy can be combined with an implementation of the BLAS API that is optimised for the specific details of the processor it is running on. For large networks, the time spent on mathematical operations is much larger than the time spent on interpretation operations and so Brian is very efficient. For smaller networks, the interpretation overhead is much larger in proportion but in many situations it is not critical because the simulation time is shorter too. The least favourable scenario for Brian is the simulation of a small network for a long biological time.

4.3 Parallel simulations

We are currently working on distributed computing techniques for the Brian simulator, which I will briefly outline here.

First of all, independent simulations can be run with job scheduling software such as Condor (Frey et al., 2002) and Boinc `http://boinc.berkeley.edu`. For example, this would allow simultaneously running simulations for different parameter values or different inputs. In principle, this technique already works with existing tools but in practice, it can be complicated to deploy on a medium or large scale (for example, to use all the unused computers in a lab) because of dependencies (Python and scientific Python modules) and heterogeneities (different operating systems). There are two (probably complementary) ways to deal with this problem: firstly, to use virtualization techniques to distribute pre-installed virtual appliances containing all the necessary software `http://www.grid-appliance.org`; secondly, to use a recently developed package that interfaces Python with Condor

and Boinc `http://pymw.sourceforge.net`.

Secondly, simulations can be parallelized on graphics processing units (GPUs): these are inexpensive pieces of hardware (around several hundred euros) consisting of a large number of parallel processing cores (in the hundreds for the latest models). Using these cards gives the equivalent of a small parallel cluster in a single machine at much lower cost. The vectorization techniques that we have developed are very well adapted to this type of hardware. There are additional difficulties with spike propagation and handling delays, but the required techniques are mostly known (Nageswaran et al., 2009) and we have just started an open source group to progress on the implementation `http://groups.google.fr/group/brian-on-gpu`.

5 Computation with spikes

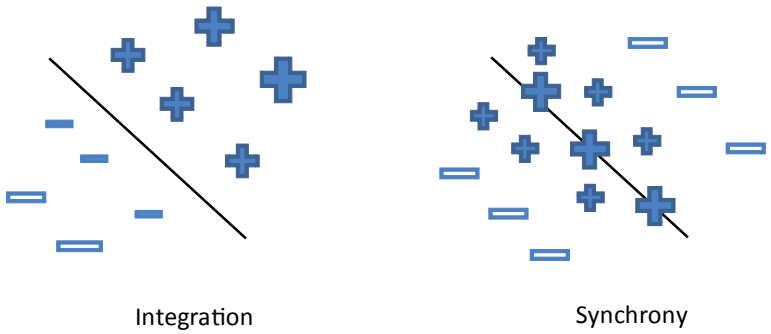


Figure 9: The two operating modes of spiking neuronal models : in the integration mode (perceptron), the neuron preferentially fires on one side of a hyperplane; in the synchrony-based mode, the neuron preferentially fires close to the hyperplane.

5.1 Rate-based theory of neural computation

Let us start with a brief overview of the classical theory of neural computation, based on firing rates. Several hypotheses lead to a description of the firing rate of a neuron as a function of the firing rates of presynaptic neurons. If we assume that synaptic integration is slow compared to the firing rate of neurons, i.e., if the leak current is neglected (the "perfect integrator" model), then one can prove (Brette, 2004) that the output firing rate is the sum of the input firing rates, weighted by the synaptic weights, or is zero when that sum is negative: $F = \sum [w_k F_k]^+$. Alternatively, if the inputs are independent Poisson processes and the synaptic weights w_k are the success probabilities of presynaptic spikes, then the output firing rate is $F = f(\sum w_k F_k)$, where f is a function, determined by the neuron model. Finally, if the inputs are not assumed to be independent but their average sum is greater than the spiking threshold and their variance is negligible, then

a similar expression $F = f(\sum w_k F_k)$ is found, where f is the current-frequency function of the neuron (Dayan and Abbott, 2001).

In these three cases, the operation that the neuron performs is close to that of a perceptron (Minsky and Papert, 1969): the neuron signals the position of the input vector $(F_1, \ldots, F_n)$ with respect to a hyperplane defined by the synaptic weights; the neuron fires more on one side of the hyperplane (Fig. 9).

However, the aforementioned hypotheses are not necessarily satisfied in the nervous system. In particular, the effective membrane time constant is short in vivo (Destexhe et al., 2003) while the average firing rate of cortical neurons is low (Attwell and Laughlin, 2001; Lennie, 2003) ; the irregularity of neuronal discharge *in vivo* (Shadlen and Newsome, 1998; Softky and Koch, 1993) and the membrane potential distributions in vivo (DeWeese and Zador, 2006) suggest that excitation is balanced by inhibition, which maintains the average input of neurons below the spiking threshold. Thus, it might be that the cortical neuron does not act as an integrator but rather as a coincidence detector (Konig et al., 1996), which suggests that neurons might compute in a different way.

5.2 Spike-based theory of neural computation

In this section, I describe some work in progress about a theory of spike-based computation that relies on selective synchronization.

Synchronization and neural computation

Neurons respond to repeated time-varying somatic current injections with reproducible patterns of precisely timed spikes (Mainen and Sejnowski, 1995). This is a stability property of spiking dynamical systems, which is shared by most spiking neuron models (Brette and Guigon, 2003). It implies in particular that similar neurons receiving similar dynamical inputs produce synchronous spike trains. On the other hand, neurons are sensitive to the synchronization of their inputs: more precisely, two input spikes are more likely to make a neuron fire when the time between these inputs is shorter than the integration time constant of the neuron. This sensitivity to spike correlations on fine timescales remains when the inputs are stochastic and numerous (Moreno et al., 2002). A number of recent physiological data tend to show that the integration time constant of neurons is short: the membrane time constant is short *in vivo* (a few milliseconds), because the intense synaptic activity increases the total conductance (Destexhe et al., 2003), while the average firing rate of cortical neurons is probably not greater than 1 Hz, according to metabolic arguments (Attwell and Laughlin, 2001; Lennie, 2003) ; the variability of the spike threshold (Azouz and Gray, 2000) and the coordination between inhibition and excitation (Wehr and Zador, 2003) also participate in shortening the integration time constant. Thus, neurons are equipped with two dual properties: selective synchronization to similar inputs and coincidence detection.

Let us consider a stimulus $X(t)$ (belonging to some space S). This stimulus, after a number of transformations (transduction, transformation by neuronal

circuits), arrives at neuron A as an input $F_A(X)$, and in neuron B as an input $F_B(X)$. When $F_A(X) = F_B(X)$ (as functions of time), the spike trains elicited by these two neurons are synchronous, which can be detected by a third neuron receiving inputs from A and B. Thus, this elementary circuit signals the identity $F_A(X) = F_B(X)$. If we consider more presynaptic neurons, the circuit signals the identities $F_1(X) = F_2(X) = \ldots = F_n(X)$, or, depending on the properties of the coincidence detector neurons, a subset of them. This operation has been described in a more specific case by Hopfield and Brody under the name "many-are-equal" (Brody and Hopfield, 2003). In their model, they considered static inputs and affine operations (synaptic weights and bias), and a coincidence detection mechanism relying on phase locking to a common oscillatory input, but the mechanism can in fact be generalized.

It is interesting to focus on the case when the operations F_i are linear (but not necessarily scalar). The operation performed by the synchrony-based mechanism is then recognizing the identity $(F_A - F_B)(X) = 0$, i.e., the property that X belongs to the kernel of the linear operator $(F_A - F_B)$, which is a linear space. If several neurons are considered, then the operation consists in detecting whether X belongs to the intersection of kernels of $(F_i - F_j)$. Thus, the operation performed by such a synchrony-based mechanisms (in the case of linear operations) is recognizing whether X belongs to some linear subspace. This is to be contrasted with the operation performed by a perceptron, which is detecting whether X is on a given side of a hyperplane (see Fig. 9). This description generalizes to non-linear operations (the linear subspace is replaced by a manifold), but a number of sensory problems can already be expressed in this simple linear framework, as is detailed in next section.

Synchrony-based computation in sensory modalities

A number of elementary percepts can be expressed in the proposed framework. In audition, the pitch of a sound corresponds in general to the period of the signal. Thus, the sounds with the same pitch, corresponding to the fundamental frequency $f_0 = 1/T$, are all those sounds $S(t)$ such that $S(t+T) = S(t)$ for all times t. This set is a linear subspace of sounds, and whether a given sound belongs to that set is signaled by the identity $F_A(S) = F_B(S)$, where F_A if the application that delays the sound by time T and F_B is the identity. This description is similar to the classical model of Licklider, based on the autocorrelation of the signal (Licklider, 1951).

The problem of sound source localization can be expressed in a similar way (see Fig. 10). The arrival time of a sound to the two ears depends on the azimuth of the sound source. If only the interaural time differences are taken into account, the sound arriving at the ears are $S_1(t) = S(t - d_1)$ and $S_2(t) = S(t - d_2)$. The azimuth of the source can be inferred from the difference $d_2 - d_1$. That difference is signaled by the identity $F_1(S_1) = F_2(S_2)$, where F_1 is the transformation that delays the sound by time d_2 and F_2 by time d_1. This description is close to the classical Jeffress model (Jeffress, 1948). More generally, taking into account the filtering differences between the two ears, due to scattering by the head and

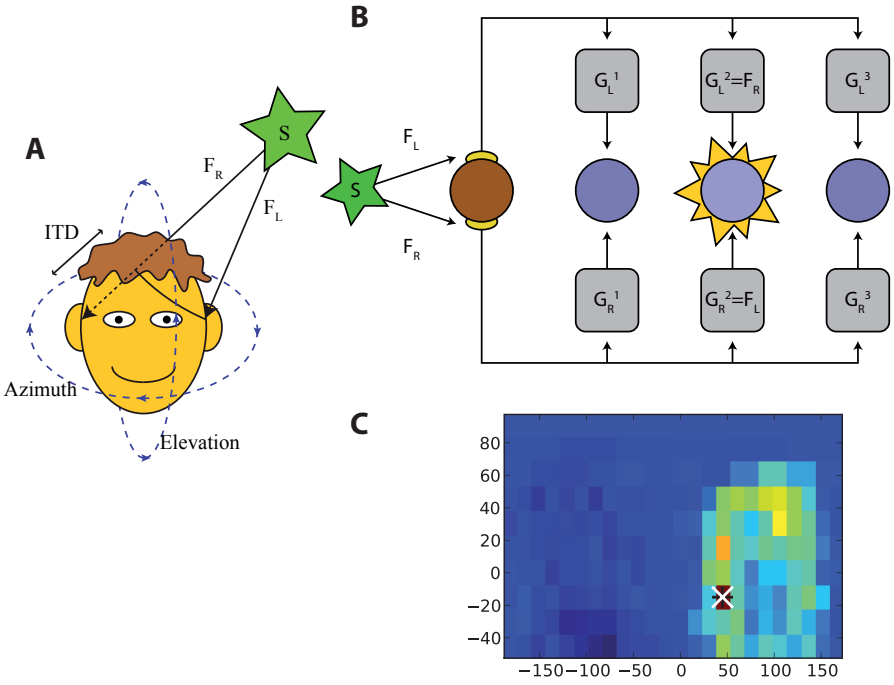


Figure 10: Sound localisation model based on selective synchronisation (adapted from Goodman, Pressnitzer and Brette (2009)). A. Sounds are filtered by the head and pinnae in a way that depends on the azimuth and elevation of the source. The filters F_L (left) and F_R (right) are different for both ears, which can be used to locate the source. B. Signals at the two ears are further processed by the auditory periphery and brainstem. When the combination of acoustical and neural filtering matches on both side, corresponding neurons fire synchronously, which can be detected by a binaural neuron. C. Color-coded firing rate of location-specific neural assemblies (horizontal: azimuth, vertical: elevation). The white cross is the location of the sound, the black cross corresponds to the neural assembly with maximum activity.

pinnae, the sound arrives at the two ears as $(G_1 * S, G_2 * S)$, where the filters G_1 et G_2 ($*$ is the convolution) are location-specific. Inferring the spatial location of the sound amounts to identifying the couple (left sound, right sound) as an element of a linear subspace, which is the image of the application $S \mapsto (G1 * S, G2 * S)$. This identification can be done with the identity $F_1(S_1) = F_2(S_2)$, where $F_1(S) = G_2 * S$ and $F_2(S) = G_1 * S$, or $F_1(S) = U * G_2 * S$ and $F_2(S) = U * G_1 * S$ for any linear filter U (for example a band-pass filter). This framework applies to all wave localization problems, whether acoustical or not (mechanical, electrical), and whether there are two or more receptors (for example, the localization of preys by sand scorpions (Stürzl et al., 2000)).

The problem of identifying an odor can also be expressed in this framework. An odor is a mixture of a number of components in specific proportions (more precisely, an odor can be defined by its affinity to each one of the olfactory receptors, which are sensitive to a large number of odorants), and the olfactory system is able to identify an odor independently of its overall concentration (Uchida and Mainen, 2008). Thus, if c_i is the concentration of component i, then an odor is the (half-)line $(\lambda c_1, \ldots, \lambda c_n)$ $(\lambda > 0)$. To explain the concentration invariance of the olfactory system, Hopfield and Brody assume that the olfactory receptors encode the concentrations of odorants in a logarithmic and static fashion, and consider neurons with various biases (Brody and Hopfield, 2003). These hypotheses are in fact unnecessary. One can simply consider a set of olfactory receptors for a wide distribution of affinities, reacting to odorant i by an activation $S(c_i/s_i)$, which may be dynamical (with adaptation for example). Then two receptors i and j produce synchronous spike trains if the vectors (c_i, c_j) and (s_i, s_j) are collinear. Thus a given odor activates sets of synchronous receptors, in an odor-specific way. In audition, the spectral aspect of the timbre of sounds can be described in a similar way (where the timbre is intensity-invariant and odor components are replaced by frequency bands).

In vision, a number of elementary percepts can be expressed in a similar fashion, such as lines with a specific orientation, periodic textures and interocular disparity. There is an original and interesting aspect of color perception, which deserves closer attention. Philipona and O'Regan recently showed that colors that are considered as "pure" in most human cultures (some specific hues of red, green, blue and yellow) correspond to particular ways in which natural light interacts with surfaces and photopigments (Philipona and O'Regan, 2006). There are three types of photopigments, so that when the illumination conditions change, a given surface seen from the photoreceptors spans a three-dimensional linear space. For these 4 "pure" colors, that space is only a two-dimensional space (plane) or a one-dimensional space (line). Thus, recognizing one of those pure colors amounts to detecting whether the color signal (R, G, B) belongs to a specific plane or line. In both cases, that operation can be performed as previously described with a neuronal synchronization mechanism.

The elementary sensory problems mentioned above can thus be expressed in terms of detecting whether the stimulus belongs to some linear subspace, and that sort of operation is most directly implemented by neurons with synchrony-based mechanisms, rather than with integration mechanims.

Learning synchrony-based neural mechanisms

The previous discussion shows that a number of elementary perceptual operations can be simply implemented with synchrony-based neural mechanisms, but it does not allow us to conclude that those mechanisms are indeed used by the nervous system. That question must be addressed by experimental work and the study of physiological, anatomical and psychophysical data. However, a complementary theoretical aspect can provide some additional information about the physiological plausibility of those mechanisms: the investigation of synaptic plasticity in the proposed sensory models. Indeed, showing that, in realistic conditions, the previously described neural circuits can naturally emerge under the action of known synaptic plasticity mechanisms is a strong point in favor of those models. It seems plausible given the recent results about spike-timing dependent synaptic plasticity, since it has been shown that these physiological mechanisms favor correlated inputs (Gerstner et al., 1996; Song et al., 2000; Senn, 2002).

How can synchrony-based neural mechanisms develop? A simple general process can be described in two stages, as follows. Stimuli S are represented through a large number of transformations F_i by a set $\{F_i(S)\}$, which are inputs to neurons (the encoding stage). In the context of audition, these transformations can correspond to the spectral decomposition of sounds by the basilar membrane, the inner hair cells and the auditory nerves, and possibly by subsequent cochlear nucleus neurons. In the context of olfaction, this multiple transformation is performed by the olfactory receptors, which have a wide distribution of affinities to odorants. These transformed stimuli $F_i(S)$ trigger stimulus-specific spike trains, with a specific correlation structure, as previously described. In a second layer (whether anatomically distinct or not), neurons receive inputs from the first layer (the decoding stage). Under the effect of spike-timing dependent plasticity, these neurons select groups of correlated inputs, as described for example in (Song and Abbot, 2001). From a mathematical point of view, the synaptic plasticity mechanism tends to detect and select subspaces with low dimensionality.

6 Discussion

Neurons compute mainly with action potentials or "spikes", which are stereotypical electrical impulses. Over the last century, the operating function of neurons has been mainly described in terms of firing rates, with the timing of spikes bearing little information. This classical point of view has led to a considerable number of developments in computing, from the perceptron (Minsky and Papert, 1969) to modern artificial neural network theories for pattern recognition (Bishop, 1996). However, recent experimental evidence and theoretical studies show that the relative spike timing of inputs has an important effect both on computation and learning in neurons: correlated inputs are more likely to make neurons fire *in vivo* (Usrey et al., 2000) (which is confirmed by models (Moreno et al., 2002)), and synaptic plasticity mechanisms favor correlated synaptic inputs (Gerstner et al., 1996; Song et al., 2000; Senn, 2002). Besides, neurons are known to respond to dynamic somatic input (*in vitro*) with precisely timed spikes (Street and Manis,

2007; Mainen and Sejnowski, 1995) and some sensory tasks can be performed so fast that they probably rely on spike timing rather than firing rate (Thorpe et al., 1996; VanRullen et al., 2005). This evidence has triggered considerable interest for spiking neuron models in computational neuroscience (Gerstner and Kistler, 2002), but the theory of computation in those models is sparse, although there has been a lot of progress about the dynamics of spiking neural networks (Brunel, 2000; Brunel and Hakim, 1999).

Spiking models can be described as hybrid systems, i.e., sets of differential equations and discrete events (spikes). Virtually all models in the litterature can be described in this formalism. The sodium current, which is responsible spike initiation, can be approximated by an exponential function near threshold, which suggests a new spiking model: the exponential integrate-and-fire (Fourcaud-Trocme et al., 2003). Although this is more accurate than the hard threshold of standard integrate-and-fire models, it turns out that spikes are in fact much sharper than suggested by sodium channel patch clamp measurements, because of active backpropagation between spike initiation site (in the axon hillock) and soma. It explains why conventional integrate-and-fire models (with adaptation) are in fact not a bad approximation of cortical neurons (as far as somatic injection is concerned). The spike threshold also adapts, both to the membrane potential and to emitted spikes. This adaptation, due to sodium channel inactivation and possibly adaptive voltage-gated conductances (potassium), shortens effective postsynaptic potentials, which makes the temporal resolution of neurons finer. Such models that include spike initiation (either exponential or quadratic) can also reproduce many different spiking patterns such as bursting or post-inhibitory spiking when augmented with an adaptive equation.

Spiking models require specific techniques for efficient simulation. For large and dense networks, the computational cost of simulation is dominated by synaptic transmissions, although in many practical situations, the cost of state updates is significant or even dominant in clock-driven simulations. We have developed vectorization techniques which allow efficient simulation in a high-level language (Python), providing great flexibility. The simulator is developed as an open source project (Brian, `http://www.briansimulator.org`) and it should be able to run on graphics processing units in the near future, providing great efficiency without expensive dedicated hardware.

Because of the discontinuities introduced in the dynamics of spiking models by the threshold condition, these models are difficult to analyze mathematically. Spiking models exhibit two important dual properties: selective synchronization and coincidence detection. Spiking models respond with precisely timed spike trains to time-varying inputs, which implies that neurons receiving similar inputs should synchronize. The dual property is coincidence detection: a neuron is more likely to fire when input spike trains are finely correlated than when they are uncorrelated. To quantify this property in realistic situations, when neurons are subjected to background synaptic activity, I propose a stochastic approach based on the membrane potential distribution. These two mechanisms suggest that neurons are well equipped to perform a computationally interesting elementary operation: detecting similarities.

Since synaptic plasticity mechanisms favor synchronous inputs (Gerstner et al., 1996; Song et al., 2000; Senn, 2002), I propose to investigate more specifically computational mechanisms based on synchrony codes. In audition for example, the fine temporal structure of sounds is thought to play an important role in several aspects of auditory perception, including pitch perception and the spatial localization of sounds (Lorenzi et coll., 2006), but it is still unclear how this temporal structure is extracted with neuronal mechanisms. A number of simple perceptual properties can be identified as low-dimensional subspaces of sensory space: for example, an odor is the (half-)line $(\lambda c_1, \ldots, \lambda c_n)$ (λ corresponds to odor intensity) and the location of a sound source corresponds to the linear subspace which is the image of the the location-specific binaural filtering operation $S \mapsto (F_L * S, F_R * S)$ (S is the sound). This lower dimensionality translates to synchronous spiking in neurons, which increases the spiking probability of target neurons. We have shown that this model does indeed provide very good estimates of sound source location in realistic settings (using measured head-related transfer functions to reproduce a virtual acoustic environment), and we are currently trying to address other perceptual problems.

References

Abbott, L. F. and Nelson, S. B. (2000). Synaptic plasticity: taming the beast. *Nat Neurosci*, 3 Suppl:1178.

Anderson, P. W. (1972). More is different. *Science*, 177(4047):393–396.

Angelino, E. and Brenner, M. P. (2007). Excitability constraints on voltage-gated sodium channels. *PLoS Computational Biology*, 3(9):1751–60.

Arnold, V. I. (1961). Small denominators. i. mapping the circle onto itself. *Izv Akad Nauk SSSR Ser Mat*, 25:21—86.

Ascoli, C., Barbi, M., Chillemi, S., and Petracchi, D. (1977). Phase-locked responses in the limulus lateral eye. theoretical and experimental investigation. *Biophys J*, 19(3):219.

Attwell, D. and Laughlin, S. B. (2001). An energy budget for signaling in the grey matter of the brain. *Journal of Cerebral Blood Flow & Metabolism*, 21:1133–1145.

Azouz, R. and Gray, C. (2000). Dynamic spike threshold reveals a mechanism for synaptic coincidence detection in cortical neurons in vivo. *PNAS*, 97(14):8110.

Azouz, R. and Gray, C. (2003). Adaptive coincidence detection and dynamic gain control in visual cortical neurons in vivo. *Neuron*, 37(3):513.

Badel, L., Lefort, S., Brette, R., Petersen, C. C. H., Gerstner, W., and Richardson, M. J. E. (2008). Dynamic I-V curves are reliable predictors of naturalistic pyramidal-neuron voltage traces. *Journal of Neurophysiology*, 99(2):656–66.

Bair, W. and Koch, C. (1996). Temporal precision of spike trains in extrastriate cortex of the behaving macaque monkey. *Neural Comput*, 8(6):1185.

Beierholm, U., Nielsen, C., Ryge, J., Alstrom, P., and Kiehn, O. (2001). Characterization of reliability of spike timing in spinal interneurons during oscillating inputs. *J Neurophysiol*, 86(4):1858—68.

Bernard, C., Ge, Y., Stockley, E., Willis, J., and Wheal, H. (1994). Synaptic integration of NMDA and non-NMDA receptors in large neuronal network models solved by means of differential equations. *Biological Cybernetics*, 70(3):267–273.

Bernstein, J. (1912). *Elektrobiologie*. Die Wissenschaft,, 44 Hft, Wissenschaft, Braunschweig.

Berry, M. and Meister, M. (1998). Refractoriness and neural precision. *J Neurosci*, 18(6):2200—2211.

Berry, M., Warland, D., and Meister, M. (1997). The structure and precision of retinal spike trains. *PNAS*, 94(10):5411.

Bishop, C. M. (1996). *Neural Networks Pattern Recognition*. Oxford University Press, USA.

Bower, J. M. and Beeman, D. (1998). *The book of GENESIS*. TELOS.

Brette, R. (2003). Rotation numbers of discontinuous Orientation-Preserving circle maps. *Set-Valued Analysis*, 11(4):359–371.

Brette, R. (2004). Dynamics of one-dimensional spiking neuron models. *Journal of Mathematical Biology*, 48(1):38–56.

Brette, R. (2006). Exact simulation of integrate-and-fire models with synaptic conductances. *Neural Computation*, 18(8):2004–27.

Brette, R. (2007). Exact simulation of integrate-and-fire models with exponential currents. *Neural Computation*, 19(10):2604–9.

Brette, R. (2008). The cauchy problem for one-dimensional spiking neuron models. *Cognitive Neurodynamics*, 2(1):21–27.

Brette, R. (2009). Generation of correlated spike trains. *Neural Computation*, 21(1):188–215.

Brette, R. and Destexhe, A. (2009). Intracellular recording (in press). In *Handbook of neural activity measurement*. Cambridge University Press.

Brette, R. and Gerstner, W. (2005). Adaptive exponential integrate-and-fire model as an effective description of neuronal activity. *Journal of Neurophysiology*, 94(5):3637–42.

Brette, R. and Guigon, E. (2003). Reliability of spike timing is a general property of spiking model neurons. *Neural Computation*, 15(2):279–308.

Brette, R., Piwkowska, Z., Monier, C., Rudolph-Lilith, M., Fournier, J., Levy, M., Frgnac, Y., Bal, T., and Destexhe, A. (2008). High-resolution intracellular recordings using a real-time computational model of the electrode. *Neuron*, 59(3):379–91.

Brette, R., Rudolph, M., Carnevale, T., Hines, M., Beeman, D., Bower, J. M., Diesmann, M., Morrison, A., Goodman, P. H., Harris, F. C., Zirpe, M., Natschlager, T., Pecevski, D., Ermentrout, B., Djurfeldt, M., Lansner, A., Rochel, O., Vieville, T., Muller, E., Davison, A. P., Boustani, S. E., and Destexhe, A. (2007). Simulation of networks of spiking neurons: a review of tools and strategies. *Journal of Computational Neuroscience*, 23(3):349–98.

Brody, C. D. and Hopfield, J. J. (2003). Simple networks for spike-timing-based computation, with application to olfactory processing. *Neuron*, 37(5):843.

Brunel, N. (2000). Dynamics of sparsely connected networks of excitatory and inhibitory spiking neurons. *J Comput Neurosci*, 8(3):183.

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

Brunel, N., Hakim, V., and Richardson, M. J. E. (2003). Firing-rate resonance in a generalized integrate-and-fire neuron with subthreshold resonance. *Physical Review E*, 67(5):051916.

Camera, G. L., Giugliano, M., Senn, W., and Fusi, S. (2008). The response of cortical neurons to in vivo-like input current: theory and experiment : I. noisy inputs with stationary statistics. *Biological Cybernetics*, 99(4-5):279–301.

Camera, G. L., Rauch, A., Thurbon, D., Lscher, H., Senn, W., and Fusi, S. (2006). Multiple time scales of temporal response in pyramidal and fast spiking cortical neurons. *Journal of Neurophysiology*, 96(6):3448–3464.

Carnevale, N. T. and Hines, M. L. (2006). *The NEURON book*. Cambridge University Press.

Coddington, E. A. and Levinson, N. (1955). *Theory of ordinary differential equations*. McGraw-Hill, New York.

Dayan, P. and Abbott, L. F. (2001). *Theoretical neuroscience*. MIT Press Cambridge, Mass.

DeFelipe, J., Hendry, S. H., Jones, E. G., and Schmechel, D. (1985). Variability in the terminations of GABAergic chandelier cell axons on initial segments of pyramidal cell axons in the monkey sensory-motor cortex. *Journal of Comparative Neurology*, 231(3):364–384.

Denjoy, A. (1932). Sur les courbes d\'efinies par les \'equations diff\'erentielles la surface du tore. *J Math Pures Appl*, 9(11):333—375.

Destexhe, A., Mainen, Z. F., and Sejnowski, T. J. (1994). An efficient method for computing synaptic conductances based on a kinetic model of receptor binding. *Neural Comput*, 6(1):14–18.

Destexhe, A., Rudolph, M., and Pare, D. (2003). The high-conductance state of neocortical neurons in vivo. *Nat Rev Neurosci*, 4(9):739.

DeWeese, M. R. and Zador, A. M. (2006). Non-Gaussian membrane potential dynamics imply sparse, synchronous activity in auditory cortex. *Journal of Neuroscience*, 26(47):12206–18.

Ermentrout, B. and Kopell, N. (1986). Parabolic bursting in an excitable system coupled with a slow oscillation. *Siam J Appl Math*, 46(2):233–253.

Fellous, J., Houweling, A., Modi, R., Pao, R., Tiesinga, P., and Sejnowski, T. (2001). Frequency dependence of spike timing reliability in cortical pyramidal cells and interneurons. *J Neurophysiology*, 85:1782—1787.

Fourcaud-Trocme, N., Hansel, D., van Vreeswijk, C., and Brunel, N. (2003). How spike generation mechanisms determine the neuronal response to fluctuating inputs. *J Neurosci*, 23(37):11628–40.

Frey, J., Tannenbaum, T., Livny, M., Foster, I., and Tuecke, S. (2002). Condor-G: a computation management agent for Multi-Institutional grids. *Cluster Computing*, 5(3):237–246.

Gerstner, W., Kempter, R., van Hemmen, J. L., and Wagner, H. (1996). A neuronal learning rule for sub-millisecond temporal coding. *Nature*, 383(6595):76.

Gerstner, W. and Kistler, W. M. (2002). *Spiking Neuron Models*. Cambridge University Press.

Gerstner, W. and Naud, R. (2009). How good are neuron models? *Science*, 326(5951):379–380.

Giugliano, M. (2000). Synthesis of generalized algorithms for the fast computation of synaptic conductances with markov kinetic models in large network simulations. *Neural Computation*, 12(4):903–931.

Giugliano, M., Bove, M., and Grattarola, M. (1999). Fast calculation of short-term depressing synaptic conductances. *Neural Computation*, 11(6):1413–1426.

Goodman, D. and Brette, R. (2008). Brian: a simulator for spiking neural networks in python. *Frontiers in Neuroinformatics*, 2:5.

Gtig, R. and Sompolinsky, H. (2006). The tempotron: a neuron that learns spike timing-based decisions. *Nature Neuroscience*, 9(3):420–428.

Guttman, R., Feldman, L., and Jakobsson, E. (1980). Frequency entrainment of squid axon membrane. *J Membrane Biol*, 56:9—18.

Henze, D. A. and Buzsaki, G. (2001). Action potential threshold of hippocampal pyramidal cells in vivo is increased by recent spiking activity. *Neuroscience*, 105(1):121–30.

Herman, M. R. (1977). Mesure de lebesgue et nombre de rotation. In Palis, J. and do Cormo, M., editors, *Geometry and Topology*, Lecture Notes in Mathematics, page 271. Springer, Berlin.

Hines, M. (1984). Efficient computation of branched nerve equations. *International Journal of Bio-Medical Computing*, 15(1):69–76.

Hodgkin, A. and Huxley, A. (1939). Action potentials recorded from inside a nerve fibre. *Nature*, 144(3651):710.

Hodgkin, A. and Huxley, A. (1952). A quantitative description of membrane current and its application to conduction and excitation in nerve. *J Physiol (Lond)*, 117:500.

Hunter, J., Milton, J., Thomas, P., and Cowan, J. (1998). Resonance effect for neural spike time reliability. *J Neurophysiol*, 80(3):1427.

Izhikevich, E. M. (2003). Simple model of spiking neurons. *IEEE Transactions on Neural Networks*, 14(6):1569–1572.

Jahnke, A., Roth, U., and Schnauer, T. (1999). Digital simulation of spiking neural networks. In *Pulsed neural networks*, pages 237–257. MIT Press.

Jeffress, L. A. (1948). A place theory of sound localisation. *J Comp Physiol Psychol*, 41:35.

Jolivet, R., Schrmann, F., Berger, T. K., Naud, R., Gerstner, W., and Roth, A. (2008). The quantitative single-neuron modeling competition. *Biological Cybernetics*, 99(4-5):417–426.

Keener, J. P. (1980). Chaotic behavior in piecewise continuous difference equations. *Trans AMS*, 261(2):589—604.

Khn, J. and Wrgtter, F. (1998). Employing the Z-Transform to optimize the calculation of the synaptic conductance of NMDA and other synaptic channels in network simulations. *Neural Computation*, 10(7):1639–1651.

Koch, C., Bernander, O., and Douglas, R. J. (1995). Do neurons have a voltage or a current threshold for action potential initiation? *Journal of Computational Neuroscience*, 2(1):63–82.

Kole, M. H. P., Ilschner, S. U., Kampa, B. M., Williams, S. R., Ruben, P. C., and Stuart, G. J. (2008). Action potential generation requires a high sodium channel density in the axon initial segment. *Nat Neurosci*, 11(2):178–186.

Kole, M. H. P. and Stuart, G. J. (2008). Is action potential threshold lowest in the axon? *Nat Neurosci*, 11(11):1253–1255.

Konig, P., Engel, A. K., and Singer, W. (1996). Integrator or coincidence detector? the role of the cortical neuron revisited. *Trends Neurosci*, 19(4):130.

Koppl, C. (1997). Phase locking to high frequencies in the auditory nerve and cochlear nucleus magnocellularis of the barn owl, tyto alba. *J Neurosci*, 17(9):3312.

Leger, J. F., Stern, E. A., Aertsen, A., and Heck, D. (2005). Synaptic integration in rat frontal cortex shaped by network activity. *J Neurophysiol*, 93(1):281–93.

Lennie, P. (2003). The cost of cortical computation. *Current Biology*, 13(6):493–497.

Licklider, J. C. R. (1951). A duplex theory of pitch perception. *Experientia*, 7:128.

Lytton, W. W. (1996). Optimizing synaptic conductance calculation for network simulations. *Neural Comput*, 8(3):501–9.

Mainen, Z. and Sejnowski, T. (1995). Reliability of spike timing in neocortical neurons. *Science*, 268:1503.

Markram, H., Toledo-Rodriguez, M., Wang, Y., Gupta, A., Silberberg, G., and Wu, C. (2004). Interneurons of the neocortical inhibitory system. *Nature Reviews Neuroscience*, 5(10):793–807.

McCormick, D. A., Shu, Y., and Yu, Y. (2007). Neurophysiology: Hodgkin and huxley model–still standing? *Nature*, 445(7123):E1–2; discussion E2–3.

Meunier, C. (1992). Two and three dimensional reductions of the Hodgkin-Huxley system: separation of time scales and bifurcation schemes. *Biological Cybernetics*, 67(5):461–468.

Meunier, C. and Segev, I. (2002). Playing the devil's advocate: is the Hodgkin-Huxley model useful? *Trends in Neurosciences*, 25(11):558–563.

Minsky, M. L. and Papert, S. A. (1969). *Perceptrons*. MIT Press.

Moreno, R., de la Rocha, J., Renart, A., and Parga, N. (2002). Response of spiking neurons to correlated inputs. *Phys Rev Lett*, 89(28 Pt 1):288101.

Nageswaran, J. M., Dutt, N., Krichmar, J. L., Nicolau, A., and Veidenbaum, A. (2009). Efficient simulation of large-scale spiking neural networks using CUDA graphics processors. In *Neural Networks, IEEE - INNS - ENNS International Joint Conference on*, volume 0, pages 2145–2152, Los Alamitos, CA, USA. IEEE Computer Society.

Naundorf, B., Wolf, F., and Volgushev, M. (2006). Unique features of action potential initiation in cortical neurons. *Nature*, 440(7087):1060–1063.

Nowak, L., Sanchez-Vives, M., and McCormick, D. (1997). Influence of low and high frequency inputs on spike timing in visual cortical neurons. *Cereb Cortex*, 7(6):487.

Pare, D., Shink, E., Gaudreau, H., Destexhe, A., and Lang, E. J. (1998). Impact of spontaneous synaptic activity on the resting properties of cat neocortical pyramidal neurons in vivo. *J Neurophysiol*, 79(3):1450.

Philipona, D. L. and O'Regan, J. K. (2006). Color naming, unique hues, and hue cancellation predicted from singularities in reflection properties. *Visual Neuroscience*, 23(3-4):331–339.

Piwkowska, Z., Pospischil, M., Brette, R., Sliwa, J., Rudolph-Lilith, M., Bal, T., and Destexhe, A. (2008). Characterizing synaptic conductance fluctuations in cortical neurons and their influence on spike generation. *Journal of Neuroscience Methods*, 169(2):302–22.

Platkiewicz, J. and Brette, R. (2009). A threshold equation for action potential initiation. *(Submitted)*.

Press, W. H., Flannery, B. P., Teukolsky, S. A., and Vetterling, W. T. (1993). *Numerical Recipes in C: The Art of Scientific Computing*. Cambridge University Press.

Rauch, A., Camera, G. L., Luscher, H., Senn, W., and Fusi, S. (2003). Neocortical pyramidal cells respond as Integrate-and-Fire neurons to in Vivo-Like input currents. *J Neurophysiol*, 90(3):1598–1612.

Reich, D., Victor, J., Knight, B., Ozaki, T., and Kaplan, E. (1997). Response variability and timing precision of neuronal spike trains in vivo. *J Neurophysiol*, 77(5):2836.

Rescigno, A., Stein, R., Purple, R., and Poppele, R. (1970). A neuronal model for the discharge patterns produced by cyclic inputs. *Bull Math Biophys*, 32:337.

Rotter, S. and Diesmann, M. (1999). Exact digital simulation of time-invariant linear systems with applications to neuronal modeling. *Biol Cybern*, 81(5-6):381–402.

Sakmann, B., Patlak, J., and Neher, E. (1976). Single-channel currents recorded from membrane of denervated frog muscle fibers. *Nature*, 260:799–802.

Sanchez-Vives, M. V. and McCormick, D. A. (2000). Cellular and network mechanisms of rhythmic recurrent activity in neocortex. *Nature Neurosci*, 3(10):1027—1034.

Senn, W. (2002). Beyond spike timing: the role of nonlinear plasticity and unreliable synapses. *Biological Cybernetics*, 87(5):344–355.

Shadlen, M. and Newsome, W. (1998). The variable discharge of cortical neurons: implications for connectivity, computation, and information coding. *J Neurosci*, 18(10):3870.

Shu, Y., Hasenstaub, A., Duque, A., Yu, Y., and McCormick, D. A. (2006). Modulation of intracortical synaptic potentials by presynaptic somatic membrane potential. *Nature*, 441(7094):761–765.

Softky, W. and Koch, C. (1993). The highly irregular firing of cortical cells is inconsistent with temporal integration of random EPSPs. *J Neurosci*, 13(1):334.

Song, S. and Abbot, L. (2001). Cortical development and remapping through spike timing-dependent plasticity. *Neuron*, 32:339—350.

Song, S., Miller, K. D., and Abbott, L. F. (2000). Competitive hebbian learning through spike-timing-dependent synaptic plasticity. *Nature Neurosci*, 3(9):919—926.

Street, S. E. and Manis, P. B. (2007). Action potential timing precision in dorsal cochlear nucleus pyramidal cells. *Journal of Neurophysiology*, 97(6):4162–72.

Stürzl, W., Kempter, R., and van Hemmen, J. L. (2000). Theory of arachnid prey localization. *Physical Review Letters*, 84(24):5668.

Thorpe, S., Fize, D., and Marlot, C. (1996). Speed of processing in the human visual system. *Nature*, 381(6582):520.

Touboul, J. and Brette, R. (2008). Dynamics and bifurcations of the adaptive exponential integrate-and-fire model. *Biological Cybernetics*, 99(4-5):319–34.

Uchida, N. and Mainen, Z. F. (2008). Odor concentration invariance by chemical ratio coding. *Frontiers in Systems Neuroscience*, 1.

Umrath, K. (1930). Untersuchungen ber plasma und plasmastrmung an characeen. *Protoplasma*, 9(1):576–597.

Usrey, W., Alonso, J., and Reid, R. (2000). Synaptic interactions between thalamic inputs to simple cells in cat visual cortex. *J Neurosci*, 20(14):5461.

VanRullen, R., Guyonneau, R., and Thorpe, S. J. (2005). Spike times make sense. *Trends in Neurosciences*, 28(1):1–4.

Veerman, J. J. P. (1989). Irrational rotation numbers. *Nonlinearity*, 2(3):419—428.

Wehr, M. and Zador, A. M. (2003). Balanced inhibition underlies tuning and sharpens spike timing in auditory cortex. *Nature*, 426(6965):442.

Wilent, W. B. and Contreras, D. (2005). Stimulus-dependent changes in spike threshold enhance feature selectivity in rat barrel cortex neurons. *J Neurosci*, 25(11):2983–91.

Wilson, M. A. and Bower, J. M. (1989). The simulation of large-scale neural networks. In *Methods in neuronal modeling: From synapses to networks*, pages 291–333. MIT Press.

Yu, Y., Shu, Y., and McCormick, D. A. (2008). Cortical action potential backpropagation explains spike threshold variability and Rapid-Onset kinetics. *J. Neurosci.*, 28(29):7260–7272.