Matching encoding and decoding with spiking neurons

Fleur Zeldenrust, Sophie Denève, Boris Gutkin

Abstract The Generalized Linear Model (GLM) is a powerful tool in assessing neural spike responses ([1]; for an overview, see [2]). The model assumes that the output of a neuron is an inhomogeneous Poisson process, of which the instantaneous rate is given by a thresholded sum of the linearly filtered input and output. It can incorporate effectively both the neuron's receptive field and history-dependent effects such as the refractory period and spike-frequency adaptation [3], [4]. While the GLM is a descriptive model of how neurons respond to their input, we show how it can be used to unify encoding (how a neuron represents its input in its output spike train) and decoding (how the input can be reconstructed from the output spike train) properties. We analytically derive a GLM that can be interpreted as a recurrent network of neurons that optimally tracks a continuously varying input. In this GLM, every neuron only fires a spike if this reduces the mean-squared error between the received input and a prediction of the input based on the output spike trains of the network, implementing a form of Lewicki's 'matching pursuit' [5]. Contrary to the standard GLM, where input and output filters are independently fitted to a neuron's response, here the filters have a direct interpretation. This theory predicts that the feature the neuron represents directly determines its input and the output filters. Moreover, the representing feature determines the neuron's spike-generating dynamics and its connectivity to other neurons. Therefore, we predict that the encoding and decoding properties of sensory neurons are two sides of the same coin. We use this approach to investigate the coding properties of several types of neurons recorded in in vitro patch clamp experiments.

Derivation Here we derive the temporal input filters (input kernels), response functions (output kernels) and functional connections (lateral kernels) of neurons in a network that optimally tracks its input. Suppose a neuron j receives an input $s(t)$ and represents this with a kernel $k_j(t)$. Estimating the input from the spike trains of these neurons is equivalent to convolving the kernels $k_j(t)$ with the spike trains $O_j = \sum_i \delta(t - t^i)$:

$$\hat{s}(t) = \sum_j k_j(t) * O_j = \sum_{i,j} k_j(t - t_j^i), \tag{1}$$

where t_j^i is the i^{th} spike time of neuron j. Now suppose that these neurons aim to minimize the mean squared error $E(t)$ between the estimated input and the real input

$$E(t) = \int_0^t (s(u) - \hat{s}(u))^2 \, du. \tag{2}$$

This can be done by assuming that each neuron only fires a spike if this reduces the overall error, so if $E_{\text{spike}}(t) < E_{\text{no spike}}(t)$. This will result in a spike rule, where a neuron j places a spike a $t + \Delta$ if $E_{\text{spike}}(t + \Delta) - E_{\text{no spike}}(t + \Delta) < -\nu^2$, so if

$$\int_0^{t+\Delta} (k_j(u - t - \Delta)\,(s(u) - \hat{s}(u)))\, du > \frac{1}{2} \int_0^{t+\Delta} (k_j(u - t))^2 \, du + \nu^2 = T_j, \tag{3}$$

where T_j is the threshold for neuron j and ν is a sparseness constraint. If we define a kernel $\bar{k}_j(t)$ as

$$\bar{k}_j(t) = \begin{cases} k_j(-t) & \text{if } t \geq -\Delta \\ 0 & \text{if } t < -\Delta \end{cases} \tag{4}$$

we can rewrite the spike rule as

$$\begin{gathered} \int_{-\infty}^{\infty} \left(\bar{k}_j(t + \Delta - u)\,(s(u) - \hat{s}(u))\right) du > T_j \\ \bar{k}_j * s - \bar{k}_j * \sum_m k_m * O_m > T_j. \end{gathered} \tag{5}$$

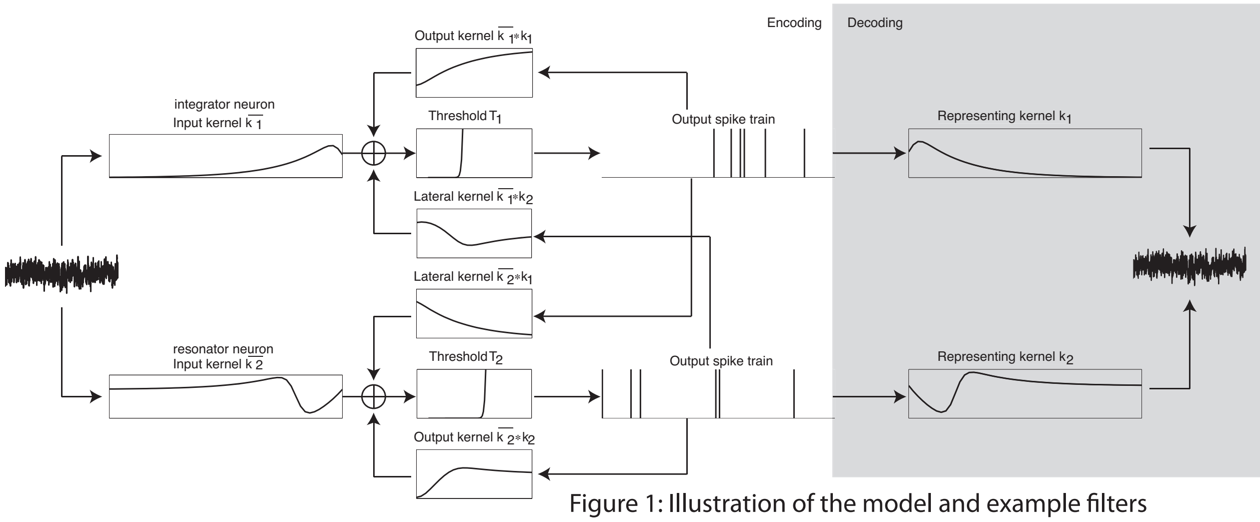

Figure 1: Illustration of the model and example filters

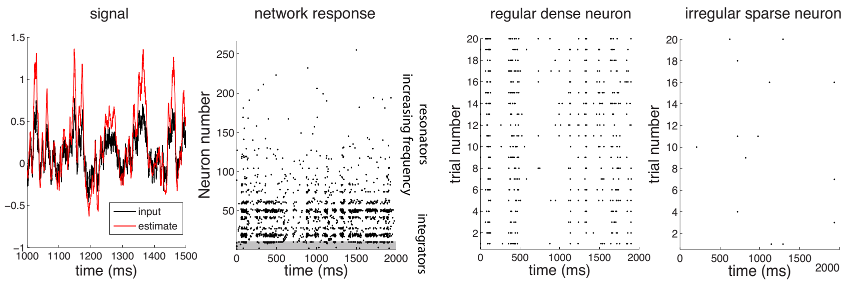

Figure 2: network response to frozen noise input

Equation (5) is equivalent to a GLM with input filter $\bar{k}_j$, response function $\Omega_{jj} = \bar{k}_j * k_j$ and functional lateral connections $\Omega_{jm} = \bar{k}_j * k_m$. Each neuron has a local estimate of the overall error due to the lateral connections Ω. The threshold function is very steep, making the spike rule effectively deterministic.

Model predictions This framework gives a functional explanation to the GLM, which is in its original form a descriptive model. It shows that encoding and decoding properties are related, but not the same, in contrast to a linear-nonlinear model (spike-triggered average). The predicted relation between the filters (figure 1) can be tested experimentally, since for experimental data, the input and output filters can be found with the usual ascent methods [2]. Using this framework, we can also assess which kernels are optimal given the structure of the input. Decoding could be done by a neuron that receives converging inputs. The input filters of this neuron should match the output filters of the network, so the encoding properties of the network are determined by the decoding properties of the next layer.
To illustrate how this model can be implemented, we built a network with a heterogeneous set of integrator neurons (unimodal Gaussian input filters with different width) and resonator neurons (input filters are wavelets with different frequency and phase and a small offset) [6]. The integrator neurons estimate the amplitude of a step or the mean of the input, the resonator neurons fluctuating features. We found this network can correctly track a variety of inputs, such as steps, ramps, oscillations, and (filtered) frozen noise signals. When white noise was added to the input, the prediction stayed stable, even though the firing of some of the neurons was very irregular (figure 2). We replicated known aspects of neural firing: the characteristics of the input, such as the power of the preferred frequency of the neuron, determine the sparseness and (ir)regularity of each neuron's response [7], [8].

References
[1] Pillow, J.W. et al (2008) *Nature* **454(7207)**, 995-9.
[2] Pillow, J. (2007) In Doya, K. et al, Bayesian Brain: Probabilistic Approaches to Neural Coding, ch. 3, 53-70 MIT Press.
[3] Paninski, L. (2004) *Network: Comput. Neural Syst.* 15(4), 243-262.
[4] Truccolo, W. et al. (2005) *J Neurophysiol* **93(2)**, 1074-89.
[5] Smith, E. and Lewicki, M. S. (2005) *Neural Comput.* **17(1)**, 19-45.
[6] Izhikevich, E. M. (2001) *Neural Netw.* **14(6-7)**, 883-94.
[7] Gutkin, B. S. et al (2003) *J Comput. Neurosci.* **15**, 91-103.
[8] Schreiber, S et al (2004) *J Neurophysiol.* **91**, 194-205.