PRELIMINARY VERSION
FINAL VERSION TO APPEAR IN ADVANCES IN NEURAL INFORMATION PROCESSING 2010

Fractionally Predictive Spiking Neurons

Sander M. Bohte
CWI, Life Sciences
Amsterdam, The Netherlands
S.M.Bohte@cwi.nl

Jaldert O. Rombouts
CWI, Life Sciences
Amsterdam, The Netherlands
J.Rombouts@cwi.nl

Abstract

Recent experimental work has suggested that the neural firing rate can be interpreted as a fractional derivative, at least when signal variation induces neural adaptation. Here, we show that the actual neural spike-train itself can be considered as the fractional derivative, provided that the neural signal is approximated by a sum of power-law kernels. Empirically, we find that the online approximation of signals with a sum of power-law kernels is beneficial for encoding signals with slowly varying components, like long-memory self-similar signals. For such signals, the online power-law kernel approximation typically required less than half the number of spikes for similar SNR as compared to sums of similar but exponentially decaying kernels. We also find that for encoding such signals, the optimal power-law exponent is similar to that reported for neural adaptation in several experiments. As power-law kernels can be accurately approximated using sums or cascades of weighted exponentials, we demonstrate that the corresponding decoding of spike-trains by a receiving neuron allows for natural and transparant temporal signal filtering by tuning the weights of the decoding kernel.

1 Introduction

A key issue in computational neuroscience is the interpretation of neural signaling, as expressed by a neuron's sequence of action potentials. An emerging notion is that neurons may in fact encode information at multiple timescales simultaneously [1, 2, 3, 4]: the precise timing of spikes may be conveying high-frequency information, and slower measures, like the rate of spiking, may be relating low-frequency information. Such multi-timescale encoding comes naturally, at least for sensory neurons, as the statistics of the outside world often exhibit self-similar multi-timescale features [5] and the magnitude of natural signals can extend over several orders. Since neurons are limited in the rate and resolution with which they can emit spikes, the mapping of large dynamic-range signals into spike-trains is an integral part of attempts at understanding neural coding.

Experiments have extensively demonstrated that neurons adapt their response when facing persistent changes in signal magnitude. Typically, adaptation changes the relation between the magnitude of the signal and the neuron's discharge rate. Since adaptation thus naturally relates to neural coding, it has been extensively scrutinized [6, 7, 8], and among others, adaptation is found to additionally exhibit features like dynamic gain control, when the standard deviation but not the mean of the signal changes [1], and long-range time-dependent changes in the spike-rate response are found in response to large magnitude signal steps, with the changes following a power-law decay (e.g. [9]).

Tying the notions of self-similar multi-scale natural signals and adaptive neural coding together, it has recently been suggested that neuronal adaptation allows neuronal spiking to communicate a *fractional derivative* of the actual computed signal [10, 4]. Fractional derivatives are a generalization of standard 'integer' derivatives ('first order', 'second order'), to real valued derivatives (e.g. '0.5th order'). A key feature of such derivatives is that they are non-local, and rather convey information over essentially a large part of the signal spectrum [10]. Fractional derivatives, or fractional calculus, is closely related to self-similar scale-free functions, like fractals, as such functions may not have a well-defined derivative (e.g., for a fractal this would depend entirely on the considered scale), but they often do have well defined fractional derivatives.

As fractional derivatives contain information over many time-ranges, they are naturally suited for predicting signals. This links to notions of predictive coding, where neurons communicate deviations from expected signals rather than the signal itself. Predictive coding has been suggested as a key feature of neuronal processing in e.g. the retina [11]. For self-similar scale-free signals, future signals may be influenced by past signals over very extended time-ranges: so-called long-memory. For example, fractional Brownian motion (fBm) can exhibit long-memory, depending on their Hurst-parameter H. For $H > 0.5$ fBM models with exhibit long-range dependence (long-memory) where the autocorrelation-function follows a power-law decay [12].

Here, we show how neural spikes can encode this long-memory dependence of the signal when the spike-train itself is taken as the fractional derivative of an approximation of the signal with a sum of power-law kernels. Then, the long-memory nature of such power-laws naturally extends the approximation into the future along the autocorrelation of the signal, at least for self-similar $1/f^{\gamma}$ like signals. The key "prediction" assumption we make is that a neuron's spike-train up to time t contains all the information that the past signal contributes to the future signal $t' > t$. Taking the spike-train as the fractional derivative, the corresponding signal $x(t)$, composed of past and future signal, is proportional to a sum of power-law decaying kernels each started respectively at the time of past emitted spikes. For this to hold, the exponent of the power-law kernels has to correspond to one minus the order of the fractional derivative. For example, power-law adaptation with an exponent of $\beta = 0.2$ (e.g. [13, 9]) would correspond to a fractional derivative of $\alpha = 0.8$.

The correspondence between a spike-train as a fractional derivative and a signal approximated as a sum of power-law kernels is only exact when spike-trains are taken as a sum of Dirac-δ functions and the power-law kernels as $1/t^{\beta}$. As both responses are singular, neurons would only be able to approximate this. We show empirically how sums of (approximated) $1/t^{\beta}$ power-law kernels can accurately approximate long-memory fBm signals via simple difference thresholding, in an online greedy fashion. Thus encodings signals, we show that the power-law kernels approximate synthesized signals with about half the number of spikes to obtain the same Signal-to-Noise-Ratio, when compared to the same encoding method using similar but exponentially decaying kernels.

We further demonstrate the approximation of sine wave modulated white-noise signals with sums of power-law kernels. The resulting spike-trains, expressed as "instantaneous spike-rate", exhibit the phase-presession as in [4], with suppression of activity on the "back" of the sine-wave modulation, and stronger suppression for lower values of the power-law exponent (corresponding to a higher order for *our* fractional derivative). We find the effect is stronger when encoding the actual sine wave envelope, mimicking the difference between thalamic and cortical neurons reported in [4]. This may suggest that these cortical neurons are more concerned with encoding the sine wave envelope.

We remark that signal encoding with power-laws kernels allows for the transparent and straightforward implementation of temporal signal filtering by a post-synaptic, receiving neuron. Since neural *de*coding by a receiving neuron corresponds to adding a power-law kernel for each received spike, modifying this receiving power-law kernel then corresponds to a temporal filtering operation, effectively exploiting the wide-spectrum nature of power-law kernels. This is particularly relevant, since, as has been amply noted [9, 14], power-law dynamics can be closely approximated by a weighted sum or cascade of exponential kernels. Temporal filtering would then correspond to simply tuning the weights for this sum or cascade. We illustrate this notion with an encoding/decoding example for both a high-pass and low-pass filter.

2 Power-law Signal Encoding

Neural processing can often be reduced to a Linear-Non-Linear (LNL) filtering operation on incoming signals [15] (figure 1), where inputs are linearly weighted and then passed through a non-linearity to yield the neural activation. As this computation yields analog activations, and neurons communicate through spikes, the additional problem faced by spiking neurons is to decode the incoming signal and then encode the computed LNL filter again into a spike-train. The standard spiking neuron model is that of Linear-Nonlinear-Poisson spiking, where spikes have a stochastic relationship to the computed activation [16]. Here, we interpret the spike encoding and decoding in the light of processing and communicating signals with fractional derivatives [10].

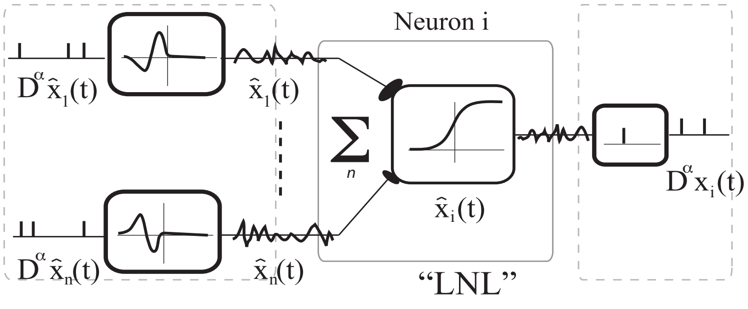

Figure 1: Linear-Non-Linear filter, with spike-decoding front-end and spike-encoding back-end.

At least for signals with mainly (relatively) high-frequency components, it has been well established that a neural signal can be decoded with high fidelity by associating a fixed kernel with each spike, and summing these kernels [17]; keeping track of doublets and triplet spikes allows for even greater fidelity. This approach however only worked for signals with a frequency response lacking low frequencies [17]. Low-frequency changes lead to “adaptation”, where the kernel is adapted to fit the signal again [18]. For long-range predictive coding, the absence of low frequencies leaves little to predict, as the effective correlation time of the signals is then typically very short as well [17].

Using the notion of predictive coding in the context of (possible) long-range dependencies, we define the goal of signal encoding as follows: let a signal $x_j(t)$ be the result of the continuous-time computation in neuron j up to time t, and let neuron j have emitted spikes t_j up to time t. These spikes should be emitted such that the signal $x_j(t')$ for $t' < t$ is decoded up to some signal-to-noise ratio, *and* these spikes should be predictive for $x_j(t')$ for $t' > t$ in the sense that no additional spikes are needed at times $t' > t$ to convey the predictive information up to time t.

Taking kernels as a signal filter of fixed width, as in the general approach in [17] has the important drawback that the signal reconstruction incurs a delay for the duration of the filter: its detection cannot be communicated until the filter is actually matched to the signal. This is inherent to any backward-looking filter-maching solution. Alternatively, a predictive coding approach could rely on only on a very short backward looking filter, minimizing the delay in the system, and continuously computing a forward predictive signal. At any time in the future then, only deviations of the actual signal from this expectation are communicated.

2.1 Spike-trains as fractional derivative

As recent work has highlighted the possibility that neurons encode fractional derivatives, it is noteworthy that the non-local nature of fractional calculus offers a natural framework for predictive coding. In particular, as we will show, when we assume that the predictive information about the future signal is fully contained in the current set of spikes, a signal approximated as a sum of power-law kernels corresponds to a fractional derivative in the form of a sum of Dirac-δ functions, which the neuron can obviously communicate through timed spikes.

The fractional derivative $r(t)$ of a signal $x(t)$ is denoted as $D^{\alpha}x(t)$, and intuitively expresses:

$$r(t) = \frac{d^{\alpha}}{dt^{\alpha}}x(t),$$

where α is the fractional order, e.g. 0.5. This is most conveniently computed through the Fourier transformation in the frequency domain, as a simple multiplication:

$$R(\omega) = H(\omega)X(\omega),$$

where the Fourier-transformed fractional derivative operator $H(\omega)$ is by definition $(i\omega)^{\alpha}$ [10], and $X(\omega)$ and $R(\omega)$ are the Fourier transforms of $x(t)$ and $r(t)$ respectively.

We assume that neurons carry out predictive coding by emitting spikes such that all predictive information is contained in the current spikes, and no more spikes will be fired if the signal follows this prediction. Approximating spikes by Dirac-δ functions, we take the spike-train up to some time t_0

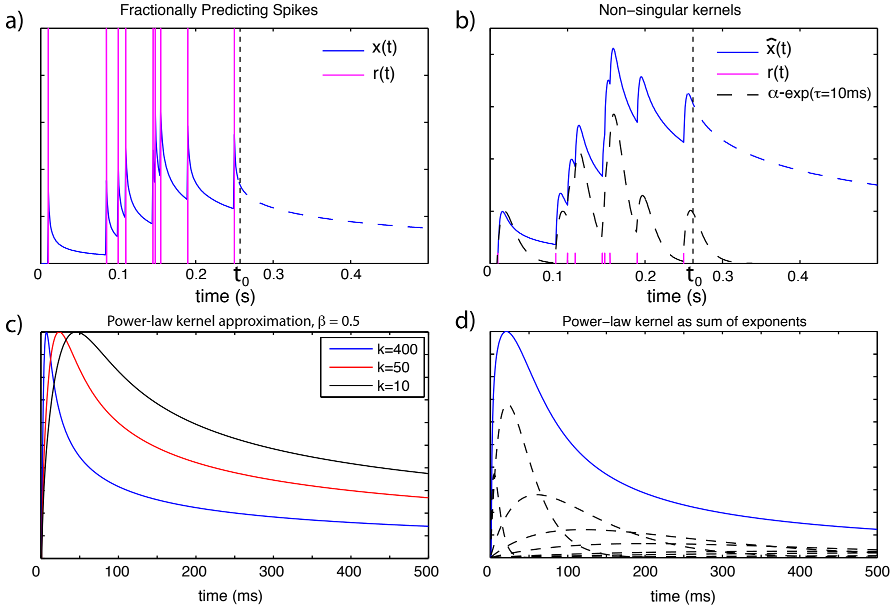

Figure 2: a) Signal $x(t)$ and corresponding fractional derivative $r(t)$: $1/t^\beta$ power-laws and delta-functions; b) power-law approximation, timed to spikes; compared to sum of α-functions (black dashed line). c) Approximated $1/t^\beta$ power-law kernel for different values of k from eq. (2). d) Approximating the approximated $1/t^\beta$ power-law kernel (blue line) with a weighted sum of α-functions with various decay time-constants (dashed lines).

to be the fractional derivative of the past signal *and* be fully predictive for the expected influence the past signal has on the future signal:

$$r(t) = \sum_{t_i < t_0} \delta(t - t_i)$$

The task is to find a signal $\hat{x}(t)$ that corresponds to an approximation of the actual signal $x(t)$ up to t_0, and where the predicted signal contribution $x(t)$ for $t > t_0$ due to $x(t < t_0)$ does not require additional future spikes. We note that a sum of power-law decaying kernels with power-law $t^{-\beta}$ for $\beta = 1 - \alpha$ corresponds to such a fractional derivative: the Fourier-transform for a power-law decaying kernel of form $t^{-\beta}$ is proportional to $(i\omega)^{\beta-1}$, hence for a signal that just experienced a single step from 0 to 1 at time t we get:

$$R(\omega) = (i\omega)^\alpha (i\omega)^{\beta-1},$$

and setting $\beta = 1 - \alpha$ yields a constant in Fourier-space, which of course is the Fourier-transform of $\delta(t)$. It is easy to check that shifted power-law decaying kernels, e.g. $(t - t_a)^{-\beta}$ correspond to a shifted fractional derivative $\delta(t - t_a)$, and the fractional derivative of a sum of shifted power-law decaying kernels corresponds to a sum of shifted delta-functions. Note that for decaying power-laws, we need $\beta > 0$, and for fractional derivatives we require $\alpha > 0$.

Thus, with the reverse reasoning, a signal approximated as the sum of power-law decaying kernels corresponds to a spike-train with spikes positioned at the start of the kernel, and, beyond a current time t, this sum of decaying kernels is is interpreted as a prediction of the extent to which the future signal can be predicted by the past signal.

Obviously, both the Dirac-δ function and the $1/t^\beta$ kernels are singular (figure 2a) and can only be approximated. For real applications, only some part of the $1/t^\beta$ curve can be considered, effectively leaving the magnitude of the kernel and the high frequency component (the extend to which the initial $1/t^\beta$ peak is approximated) as free parameters. Figure 2b illustrates the signal approximated by a random spikes train; as compared to a sum of exponentially decaying α-kernels, the long-memory effects of power-law decay kernels is evident.

2.2 Practical encoding

To explore the efficacy of the power-law kernel approach to signal encoding/decoding, we take a standard thresholding online approximation approach, where neurons communicate only deviations between the current computed signal $x(t)$ and the emitted approximated signal $\hat{x}(t)$ exceeding some threshold θ. The emitted signal $\hat{x}(t)$ is constructed as the (delayed) sum of filter kernels κ each starting at the time of the emitted spike:

$$\hat{x}(t) = \sum_{t_j < t} \kappa(t - (t_j + \Delta)),$$

the delay Δ corresponds to the time-window over which the neuron considers the difference between computed and emitted signal. Allowing for both positive and negative spikes (corresponding to tightly coupled neurons with reversed threshold polarity [17]), this would expand to:

$$\hat{x}(t) = \sum_{t_j^+ < t} \kappa(t - (t_j^+ + \Delta)) - \sum_{t_j^- < t} \kappa(t - (t_j^- + \Delta)).$$

Considering just the fixed time-window thresholding approach, a spike is emitted each time the difference between the computed signal $x(t)$ and the emitted signal $\hat{x}(t)$ plus (or minus) the kernel $\kappa(t)$ summed over some time-window exceeds the threshold θ:

$$\begin{aligned} r(t_0) = \delta(t_0) \qquad & \text{if} \sum_{\tau = t_0 - \Delta}^{t_0} |x(\tau) - \hat{x}(\tau)| - |x(\tau) - (\hat{x}(\tau) + \kappa(\tau))|) > \theta, \\ = -\delta(t_0) \qquad & \text{if} \sum_{\tau = t_0 - \Delta}^{t_0} |x(\tau) - \hat{x}(\tau)| - |x(\tau) - (\hat{x}(\tau) - \kappa(\tau))|) > \theta, \end{aligned} \tag{1}$$

the signal approximation improvement is computed here as the absolute value of the difference between the current signal noise and the signal noise when a kernel is added (or subtracted).

As an approximation of $1/t^\beta$ power-law kernels, we let the kernel first quickly rise, and then decay according to the power-law. For a practical implementation, we use a $1/t^\beta$ signal multiplied by a modified version of the logistic sigmoid function $\text{logsig}(t) = 1/(1 + \exp(-t))$: $v(t, k) = 2\,\text{logsig}(kt) - 1$, such that the kernel becomes:

$$\kappa(t) = \lambda v(t, k) 1/t^\beta, \tag{2}$$

where $\kappa(t)$ is zero for $t' < t$, and parameter k determines the angle of the initial increasing part of the kernel. The resulting kernel is further scaled by a factor λ to achieve a certain signal approximation precision (kernels for power-law exponential $\beta = 0.5$ and several values of k are shown in figure 2c). As an aside, the resulting (normalized) power-law kernel can very accurately be approximated over multiple orders of magnitude by a sum of just 11 α-function exponentials (figure 2d).

Next, we compare the efficiency of signal approximation with power-law predictive kernels as compared to the same approximation using standard fixed kernels. For this, we synthesize self-similar signals with long-range dependencies. We first remark on some properties of self-similar signals with power-law statistics, and on how to synthesize them.

2.3 Self-similar signals with power-law statistics

There is extensive literature on the synthesis of statistically self-similar signals with $1/f$-like statistics, at least going back to Kolmogorov [19] and Mandelbrot [20]. Self-similar signals exhibit slowly decaying variances, long-range dependencies and a spectral density following a power law. Importantly, for wide-sense self-similar signals, the autocorrelation functions also decays following a power-law. Although various distinct classes of self-similar signals with $1/f$-like statistics exist [12], fractional Brownian motion (fBm) is a popular model for many natural signals. Fractional Brownian motion is characterized by its Hurst-paramater H, where $H = 0.5$ corresponds to regular Brownian motion, and fBM models with $H > 0.5$ exhibit long-range (positive) dependence. The spectral density of an fBm signal is proportional to a power-law, $1/f^\gamma$, where $\gamma = 2H + 1$. We used fractional Brownian motion to generate self-similar signals for various H values, using the `wfbm` function from the Matlab wavelet toolbox.

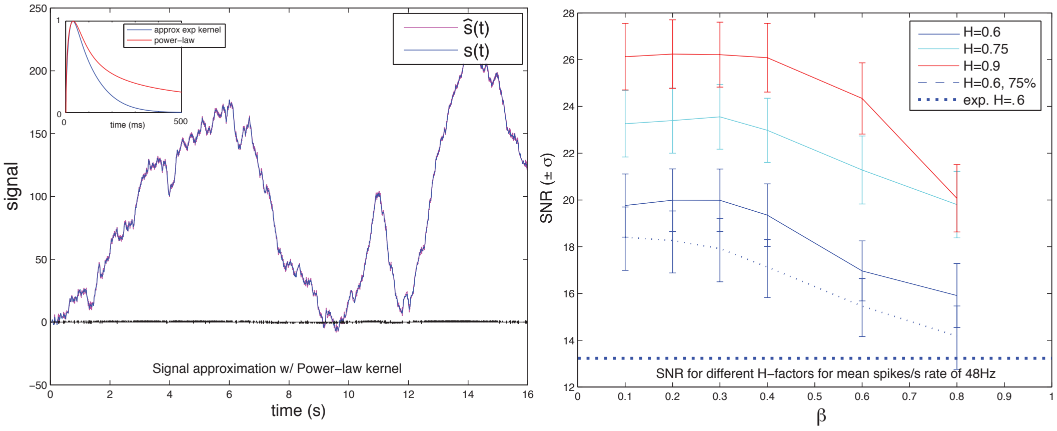

Figure 3: Left: example of encoding of fBm signal with power-law kernels. Using an exponentially decaying kernel (inset) required 1398 spikes vs. 618 for the power-law kernel ($k = 50$), for the same SNR. Right: SNR for various β power-law exponents using a fixed number of spikes (48Hz), with curves for different H-parameters, each curve averaged over five 16s signals. The dashed blue curve plots the $H = 0.6$ curve, using less spikes (36Hz); the flat bottom dotted line shows the average performance of the non-power-law exponentially decaying kernel, also for $H = 0.6$.

3 Signal encoding/decoding

3.1 Encoding long-memory self-similar signals

We applied the thresholded kernel approximation outlined above to synthesized fBm signals with $H > 0.5$, to ensure long-term dependence in the signal. An example of such encoding is given in figure 3, left panel, using both positive and negative spikes, (inset, red line: the power-law kernel used). When encoding the same signal with kernels without the power-law tail (inset, blue line), the approximation required more than twice as many spikes for the same Signal-to-Noise-Ratio (SNR).

In figure 3, right panel, we compared the encoding efficacy for signals with different H-parameters, as a function of the power-law exponent, using the same number of spikes for each signal (achieved by changing the λ parameter and the threshold θ). Rather unsurprisingly, we find that more slowly varying signals, corresponding to higher H-parameters, are better encoded by the power-law kernels. More surprisingly, we find consistently that the signals are optimally encoded for low β-values, in the order of $0.1 - 0.3$, corresponding well to the empirical values reported in e.g. [9]. Similar results were obtained for different values of k in equation (2).

We should remark that without negative spikes, there is no longer a clear performance advantage for power-law kernels (even for large β): where power-law kernels are beneficial on the rising part of a signal, they lose on downslopes where their slow decay cannot follow the signal.

3.2 Sine-wave modulated white-noise

Fractional derivatives as an interpretation of neuronal firing-rate has been put forward by a series of recent papers [10, 21, 4], where experimental evidence was presented to suggest such an interpretation. A key finding in [4] was that the instantaneous firing rate of neurons along various processing stages of a rat's whisker movement exhibit a phase-lead relative to the amplitude of the movement modulation. The phase-lead was found to be greater for cortical neurons as compared to thalamic neurons. When the firing rate corresponds to the α-order fractional derivative, the phase-lead would correspond to greater fractional order α in the cortical neurons [10] . We used the sum-of-power-laws to approximate both the sine-wave-modulated white noise and the actual sine-wave itself, and found similar results (figure 4): smaller power-law exponents, in our interpretation also corresponding to larger fractional derivative orders, lead to increasingly fewer spikes at the back of the sine-wave (both in the case where we encode the signal with both positive and negative spikes – then counting only the positive spikes – and when the signal is approximated with only positive spikes – not shown). We find an increased phase-lead when approximating the actual sine-wave ker-

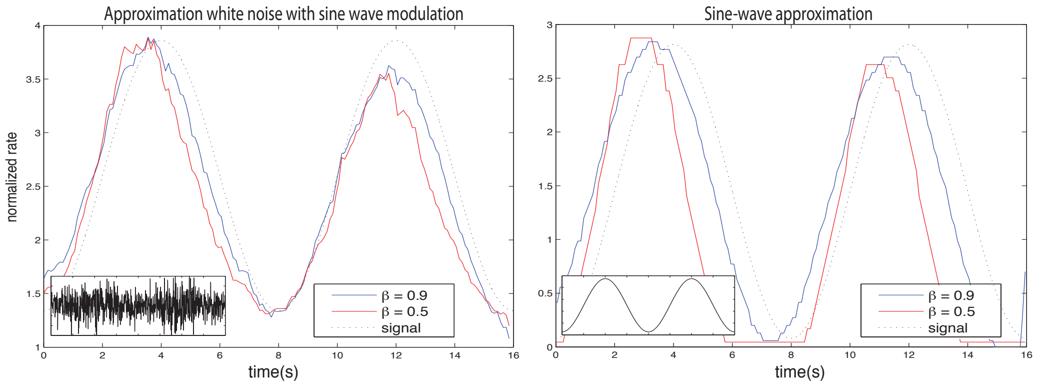

Figure 4: Sinewave phase-lead. Left: when encoding sine-wave modulated white noise (inset); right: encoding the sine-wave signal itself (inset). Average firing rate is computed over 100ms, and normalized to match the sine-wave kernel.

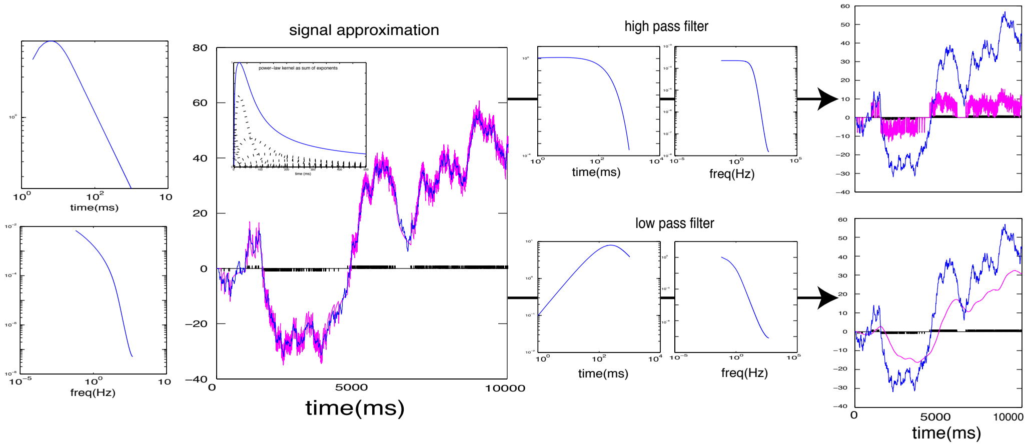

Figure 5: Illustration of frequency filtering with modified decoding kernels. The square boxes show the respective kernels in both time and frequency space. See text for further explanation.

nel as opposed to the white-noise modulation, suggesting that perhaps cortical neurons more closely encode the former as compared to thalamic neurons.

3.3 Signal Frequency Filtering

For a receiving neuron i to properly interpret a spike-train $r(t)_j$ from neuron j, both neurons would need to keep track of past events over extended periods of time: current spikes have to be added to or subtracted from the future expectation signal that was already communicated through past spikes. The required power-law processes can be implemented in various manners, for instance as a weighted sum or a cascade of exponential processes [9, 10]. A natural benefit of implementing power-law kernels as a weighted sum or cascade of exponentials is that a receiving neuron can carry out temporal signal filtering simply by tuning the respective weight parameters for the kernel with which it decodes spikes into a signal approximation.

In figure 5, we illustrate this with power-law kernels that are transformed into high-pass and low-pass filters. We first approximated our power-law kernel (2) with a sum of 11 exponentials (depicted in the left-center inset). Using this approximation, we encoded the signal (figure 5, center). The signal was then reconstructed using the resultant spikes, using the power-law kernel approximation, but with some zeroed out exponentials (respectively the slowly decaying exponentials for the high-pass filter, and the fast-decaying kernels for the low-pass filter). Figure 5, most right, shows the resulting filtered signal approximations. Obviously, more elaborate tuning of the decoding kernel with a larger sum of kernels can approximate a vast variety of signal filters.

4 Discussion

Taking advantage of the particular relationship between power-laws and fractional derivatives, we outlined the peculiar fact that a sum of Dirac-δ functions, when taken as a fractional derivative, corresponds to a signal in the form of a sum of power-law kernels. Exploiting the obvious link to spiking neural coding, combined with the recent experimental suggestion that neurons may indeed compute fractional derivatives, we showed how a simple thresholding neuron can compute a signal approximation as a sum of power-law kernels. We then demonstrated the usefulness of such an approximation when encoding slowly varying signals, finding that encoding with power-law kernels significantly outperformed similar but exponentially kernels that do not take long-range signal dependencies into account. We furthermore found that the optimal power-law exponent was rather small, but, curiously, corresponded roughly to values reported in at least some of the experimental literature.

Compared to the work where the firing rate is considered as a fractional derivative, e.g. [10], the advantage of the formulation presented here is that it extend the notion of a fractional derivative to finer temporal variations: each spike effectively encodes very local signal variations, while also keeping track of long-range variations.

It is worth remarking that the interpretation in [10] of the fractional derivative $r(t)$ as a *rate* leads to 1:1 relation between the fractional derivative order and the power-law decay exponent β [10], as compared to the relationship here derived, $\beta = 1 - \alpha$. Intriguingly, fitting a power-law to the adaptation reported in [13] yields an exponent of about 0.2, similar to the rate-based fractional derivative reported in [10], and in our experiments we also find that optimal signal approximation was found for $\beta = 0.2$.

As noted, the singularity of $1/t^{\beta}$ power-law kernels means that initial part of the kernel can only be approximated. Here, we initially focused our simulation on the use of long-range power-law kernels for encoding slowly varying signals. A more detailed approximation of this initial part of the kernel may be needed to incorporate effects like gain modulation [22, 8], and determine up to what extent the power-law kernels already account for this phenomenon. This would also provide a natural link to existing neural models of spike-frequency adaptation, e.g. [23], as they are primarily concerned with modeling the spiking neuron behavior rather than the computational aspects.

We used a greedy online thresholding process to determine when a neuron would spike to approximate a signal, this in contrast to offline optimization methods that place spikes at optimal times, like Smith & Lewicki [24]. The key difference of course is that the latter work is concerned with decoding a signal, and in effect attempts to determine the effective neural (temporal) filter. As we aimed to illustrate in the signal filtering example, these notions are not mutually exclusive: a receiving neuron could very well filter the incoming signal with a carefully shaped weighted sum of kernels, and then, when the filter is activated, signal the magnitude of the match through fractional spiking.

Predictive coding seeks to find a careful balance between encoding known information as well as future, derived expectations [25]. It does not seem unreasonable to formulate this balance as a no-going-back problem, where current computations are projected forward in time, and corrected where needed. In terms of spikes, this would correspond to our assumption that, absent new information, no additional spikes need to be fired by a neuron to transmit this forward information.

The kernels we find are somewhat in contrast to the kernels found by Bialek et. al. [17], where the optimal filter exhibited both a negative and a positive part and no long-range “tail”. Several practical issues may contribute to this difference, not least the relative absence of low frequency variations, as well as the fact that the signal considered is derived from the fly’s H1 neurons. These two neurons have only partially overlapping receptive fields, and the separation into positive and negative spikes is thus slightly more intricate. We need to remark though that we see no impediment for the presented signal approximation to be adapted to such situations, or situations where more than two neurons encode fractions of a signal, as in population coding, e.g. [26].

Finally, we would like to remark that the issue of long-range temporal dependencies such as discussed here seems to be relatively unappreciated. As pointed out in [9], long-range power-law dynamics would seem to offer a variety of “hooks” for computation through time, like for temporal difference learning and relative temporal computations (and possibly exploiting the many noted correspondences between spatial and temporal statistics [27]) .

References

[1] A.L. Fairhall, G.D. Lewen, W. Bialek, and R.R.R. van Steveninck. Multiple timescales of adaptation in a neural code. In *Advances in Neural Information Processing Systems*, volume 13, page 124. The MIT Press, 2001.

[2] B. Wark, A. Fairhall, and F. Rieke. Timescales of inference in visual adaptation. *Neuron*, 61(5):750–761, 2009.

[3] S. Panzeri, N. Brunel, N.K. Logothetis, and C. Kayser. Sensory neural codes using multiplexed temporal scales. *Trends in Neurosciences*, page in press, 2010.

[4] B.N. Lundstrom, A.L. Fairhall, and M. Maravall. Multiple Timescale Encoding of Slowly Varying Whisker Stimulus Envelope in Cortical and Thalamic Neurons In Vivo. *J. of Neuroscience*, 30(14):5071, 2010.

[5] JH Van Hateren. Processing of natural time series of intensities by the visual system of the blowfly. *Vision Research*, 37(23):3407–3416, 1997.

[6] N. Brenner, W. Bialek, and R. de Ruyter van Steveninck. Adaptive rescaling maximizes information transmission. *Neuron*, 26(3):695–702, 2000.

[7] B. Wark, B.N. Lundstrom, and A. Fairhall. Sensory adaptation. *Current opinion in neurobiology*, 17(4):423–429, 2007.

[8] M. Famulare and A.L. Fairhall. Feature selection in simple neurons: how coding depends on spiking dynamics. *Neural Computation*, 22:1–18, 2009.

[9] P.J. Drew and LF Abbott. Models and properties of power-law adaptation in neural systems. *Journal of neurophysiology*, 96(2):826, 2006.

[10] B.N. Lundstrom, M.H. Higgs, W.J. Spain, and A.L. Fairhall. Fractional differentiation by neocortical pyramidal neurons. *Nature neuroscience*, 11(11):1335–1342, 2008.

[11] T. Hosoya, S.A. Baccus, and M. Meister. Dynamic predictive coding by the retina. *Nature*, 436(7047):71–77, 2005.

[12] G.W. Wornell. *Signal processing with fractals: a wavelet based approach.* Prentice Hall, NJ, 1999.

[13] Z. Xu, JR Payne, and ME Nelson. Logarithmic time course of sensory adaptation in electrosensory afferent nerve fibers in a weakly electric fish. *Journal of neurophysiology*, 76(3):2020, 1996.

[14] S. Fusi, PJ Drew, and LF Abbott. Cascade models of synaptically stored models. *Neuron*, 45:1–14, 2005.

[15] C.M. Bishop. *Neural networks for pattern recognition.* Oxford University Press, USA, 1995.

[16] EJ Chichilnisky. A simple white noise analysis of neuronal light responses. *Network: Computation in Neural Systems*, 12(2):199–213, 2001.

[17] F. Rieke, D. Warland, and W. Bialek. *Spikes: exploring the neural code.* The MIT Press, 1999.

[18] A.L. Fairhall, G.D. Lewen, W. Bialek, and R.R.R. van Steveninck. Efficiency and ambiguity in an adaptive neural code. *Nature*, 412(6849):787–792, 2001.

[19] A. Kolmogorov. Wienersche Spiralen und einige andere interessante kurven in Hilbertschen raum. *Computes Rendus (Doklady) Academic Sciences USSR (NS)*, 26:115–118, 1940.

[20] B.B. Mandelbrot and J.W. Van Ness. Fractional Brownian motions, fractional noises and applications. *SIAM review*, 10(4):422–437, 1968.

[21] B.N. Lundstrom, M. Famulare, L.B. Sorensen, W.J. Spain, and A.L. Fairhall. Sensitivity of firing rate to input fluctuations depends on time scale separation between fast and slow variables in single neurons. *Journal of Computational Neuroscience*, 27(2):277–290, 2009.

[22] S. Hong, B.N. Lundstrom, and A.L. Fairhall. Intrinsic gain modulation and adaptive neural coding. *PLoS Computational Biology*, 4(7), 2008.

[23] R. Jolivet, A. Rauch, HR Luescher, and W. Gerstner. Integrate-and-Fire models with adaptation are good enough: predicting spike times under random current injection. *Advances in Neural Information Processing Systems*, 18:595–602, 2006.

[24] E. Smith and M.S. Lewicki. Efficient coding of time-relative structure using spikes. *Neural Computation*, 17(1):19–45, 2005.

[25] N. Tishby, F.C. Pereira, and W. Bialek. The information bottleneck method. *Arxiv physics/0004057*, 2000.

[26] Q.J.M. Huys, R.S. Zemel, R. Natarajan, and P. Dayan. Fast population coding. *Neural Computation*, 19(2):404–441, 2007.

[27] O. Schwartz, A. Hsu, and P. Dayan. Space and time in visual context. *Nature Reviews Neuroscience*, 8(11), 2007.