

A fast and accurate approximation of power-law adaptation for auditory computational models (L)

Daniel R. Guest^{1,a)}  and Laurel H. Carney^{1,2}

¹Department of Biomedical Engineering, University of Rochester, Rochester, New York 14627, USA

²Department of Neuroscience, University of Rochester, Rochester, New York 14627, USA

ABSTRACT:

Power-law adaptation is a form of neural adaptation that has been recently implemented in a popular model of the mammalian auditory nerve to explain responses to modulated sound and adaptation over long time scales. However, the high computational cost of power-law adaptation, especially for longer simulations, means it must be approximated to be practically usable. Here, a straightforward scheme to approximate power-law adaptation is presented, demonstrating that the approximation improves on an existing approximation provided in the literature. Code that implements the new approximation is provided. © 2024 Acoustical Society of America.

<https://doi.org/10.1121/10.0034457>

(Received 8 April 2024; revised 27 September 2024; accepted 5 November 2024; published online 11 December 2024)

[Editor: Sarah Verhulst]

Pages: 3954–3957

I. INTRODUCTION

This letter to the editor suggests a new strategy to improve the speed and accuracy of the popular mammalian auditory-nerve (AN) model of Zilany *et al.* (2014), hereafter referred to as the ZBC model. The model has gained popularity because of its high level of detail and fidelity to physiological data, but such detail comes at a substantial computational cost, which can dissuade users from applying the model to large-scale problems. One notable source of computational cost in the ZBC model is power-law adaptation (PLA) (Drew and Abbott, 2006; Zilany *et al.*, 2009), a form of neural adaptation that strikes a balance between adaptation with infinite memory of past responses, or perfect adaptation, and adaptation with a fixed time constant and exponential forgetting of past responses, or exponential adaptation (EA) (Drew and Abbott, 2006). Including PLA in the model improved predictions of AN response features such as synchrony to sinusoidally amplitude modulated (SAM) stimuli (Zilany *et al.*, 2009). This letter describes a strategy for approximating PLA that is as fast as previous approximations while improving accuracy over a wide range of time scales.

II. DERIVATION OF THE PLA

Throughout, we denote scalar-valued variables in *italic* and vector-valued variables in ***bold italic***. In models of adaptation, the time-varying output rate, $r(t)$, is described as the half-wave rectified difference between an input rate, $s(t)$, and the output of an integrator, $I(t)$:

$$r(t) = \max(0, s(t) - I(t)). \quad (1)$$

In PLA, the integrator is governed by power-law temporal dynamics:

$$I(t) = \int_0^t \frac{r(u)}{t - u + \beta} du. \quad (2)$$

Here, β is a parameter with units of time that governs the rate of adaptation (Drew and Abbott, 2006). It will be of use to note that Eq. (2) can be expressed as convolution (*) with a power-law kernel:

$$I(t) = r(t) * \frac{1}{t + \beta}. \quad (3)$$

Here the focus is on how to efficiently compute PLA. We adopt Big O notation (Knuth, 1976) to characterize PLA's computational complexity, or how many elementary operations are required to compute PLA for an input of n samples. $O(1)$ indicates that complexity is a constant factor (i.e., it does not depend on n), $O(n)$ indicates that complexity is a linear function of n , and $O(n^2)$ indicates that complexity is a quadratic function of n . Direct calculation of $I(t)$ for all time steps using Eq. (2) has complexity $O(n^2)$, the same as general one-dimensional convolution. That is, the cost of computing PLA grows superlinearly with the number of samples in the simulation, limiting PLA to short simulations at low sampling rates, for which the factor n^2 does not grow too large.

To address this problem, we can seek to replace direct implementation of Eq. (3) with some function that approximates Eq. (3) but has complexity less than $O(n^2)$. Infinite impulse response (IIR) filters have complexity $O(n)$, so we can reframe our goal as finding an IIR filter that has an impulse response approximating the power-law kernel in Eq. (3). The same approach was adopted in previous versions of the ZBC model, which approximates convolution in Eq. (3) as an IIR filter with numerically optimized weights¹ (one 10th-order filter for the “slow” PLA pathway and one 6th-order filter for the “fast” PLA pathway). Hereafter, this scheme is called “ZBC 2009/14 PLA” to indicate the approximation scheme in Zilany *et al.* (2009) and provided in the model code of Zilany *et al.* (2009) and Zilany *et al.* (2014).

^{a)}Email: Daniel_Guest@urmc.rochester.edu

ZBC 2009/14 PLA requires numerical optimization of the filter coefficients for specific combinations of β and sampling rate f_s , and the coefficients thus cannot readily be translated to simulations with different values of β or f_s . Moreover, as expected, the particular coefficients provided by Zilany *et al.* (2009) yield poor predictions of PLA for long simulations where $n \gg f_s$. An alternative approach is hinted at by Drew and Abbott (2006), who point out that “... we can mimic power-law adaptation, at least over a finite time interval, using a [sum] of exponential functions” and argue that power-law dynamics could in fact arise from multiple underlying processes with exponential dynamics and a range of time constants (see also Newman, 2005). This approach suggests approximating the power-law kernel, $f(t) = 1/(t + \beta)$, as the sum of many exponential kernels of the form

$$g(t) = \sum_{i=0}^{N-1} g_i(t) = \sum_{i=0}^{N-1} w_i \exp(-t/\tau_i). \tag{4}$$

Convolution with a kernel of the form $w \exp(-t/\tau)$ can be efficiently implemented via a first-order IIR filter with the difference equation

$$y[n] = w x[n] + d y[n - 1]. \tag{5}$$

Filters are as in Eq. (5), where $0 < d < 1$, are lowpass filters with a 3-dB cutoff frequency of $f = 1/2\pi\tau$ Hz. The decay coefficient, d , can be calculated as $d = \exp(-1/f_s\tau)$. Note that we set the coefficients for x such that the filter has an impulse response that peaks at w at time $t = 0$, rather than having unity gain at direct current (dc) making it easier to see the connection between this filter and Eq. (4). We can efficiently compute convolution with a weighted sum of exponentials via a set of these filters implemented in parallel. If we select w and τ appropriately, so as to match the combined impulse response of our set of filters with the power-law kernel $f(t)$, we can in turn approximate power-law adaptation, with $O(n)$ complexity.

Bochud and Challet (2007) provide a useful starting point for analytic heuristics for w and τ . The authors point out that a given exponential can approximate a power-law function reasonably well over a corresponding time scale (i.e., order of magnitude of time) near τ but makes little contribution for time scales beyond that due to the nature of exponential decay. Figure 1(A) displays an approximation of $f(t)$ by $\exp(-t/\beta)$ on linear axes. On log-log axes [Fig. 1(B)], the point made by Bochud and Challet (2007) is clear: For time $t < \beta$, the approximation is reasonably good, and by time $t = \beta$ the approximation differs from the target only by a factor of $e/2$ (i.e., still non-negligible). Beyond approximately 5β , however, the exponential function’s value is nearly zero, whereas the power-law function is still (relatively) large. This phenomenon can be seen on log-log axes as an obvious difference in slope: for $t \gg \beta$, the exponential function has a steep and accelerating slope, whereas the power-law function has its signature straight-line slope, which yields its characteristic “long tail.”

These observations and the forms of each function on log-log axes suggest that a sum of exponentials with time constants spaced evenly in log time and with weights

inversely proportional to their time constants could provide a reasonable approximation of the power-law kernel (Bochud and Challet, 2007) [Fig. 1(C)]. Each exponential contributes substantially only for $t \approx 5\tau$, and their sum, weighted with decreasing weights for longer τ , yields a match to the slope of the power-law function at long times. The approximation is imperfect, perhaps most importantly because the constant slope of the power-law kernel on the log-log axes for times well beyond can only be well approximated by exponentials up to about the longest τ used. At longer times, the lack of exponential functions with correspondingly long time constants results in a systematic truncation error [e.g., Fig. 1(C) for $t \gtrsim 1 \times 10^1$].

We adopted two strategies for selecting w , one based on a heuristic adjustment of the weights and another based on numerical optimization. In both cases, we start with

$$\tau_i = \beta \times 10^{i/e} \tag{6}$$

for $i = 0$ to $N - 1$. Note that this formula indicates time constants starting at β and increasing in increments of $1/e$ decades, thus each decade of time is approximated by about e (i.e., 2–3) exponentials. The ratio $1/e$ is a rule of thumb that works well, with smaller values leading to systematic gaps between adjacent exponentials, and larger values leading to too much overlap between adjacent exponentials (Bochud and Challet, 2007). The value of N/e then determines the longest time constant used and thus the longest times that can be simulated before the aforementioned truncation error dominates. We set $N = 14$, which yields a good approximation up to about 5 decades after β .

For the heuristic scheme, given time constants τ , the weights can be adjusted from $w_i = 1/\tau_i$, as in Fig. 1(C), to $w_i = c/(\tau_i + \beta)$, where

$$c = \left(2\beta \sum_{i=0}^{N-1} \frac{1}{\tau_i + \beta} \exp(-\beta/\tau_i) \right)^{-1}. \tag{7}$$

Intuitively, the weights $1/(\tau_i + \beta)$ reduce the amplitude of exponentials with short time constants relatively more than exponentials with long time constants, as compared to $1/\tau_i$, addressing the systematic overestimate of kernel amplitude at early times. The factor c further adjusts the weights to match $f(t)$ and $g(t)$ exactly at time $t = \beta$. This heuristic gives a surprisingly good fit over many time scales, only slightly underestimating the kernel amplitude for very short times and then again at somewhat longer times [Fig. 1(D), top row].

Next, we used numerical optimization to select optimal values for w , starting from the initial estimate $w_i = 1/\tau_i$ and subject to a loss function consisting of the sum of squared errors between $\log(f(t))$ and $\log(g(t))$ at 1000 query points spaced evenly in log time from $1/f_s$ to $10^5 \beta$ s. [f_s was set to 10 kHz in our simulations to match the synapse stage of Zilany *et al.* (2009).] This optimization was carried out in JULIA (Bezanson *et al.*, 2017) using Newton’s method via the OPTIM.JL package (Mogensen and Riseth, 2018) with default hyperparameters using forward-mode automatic differentiation (Rall, 1986; Revels *et al.*, 2016). Optimization was performed

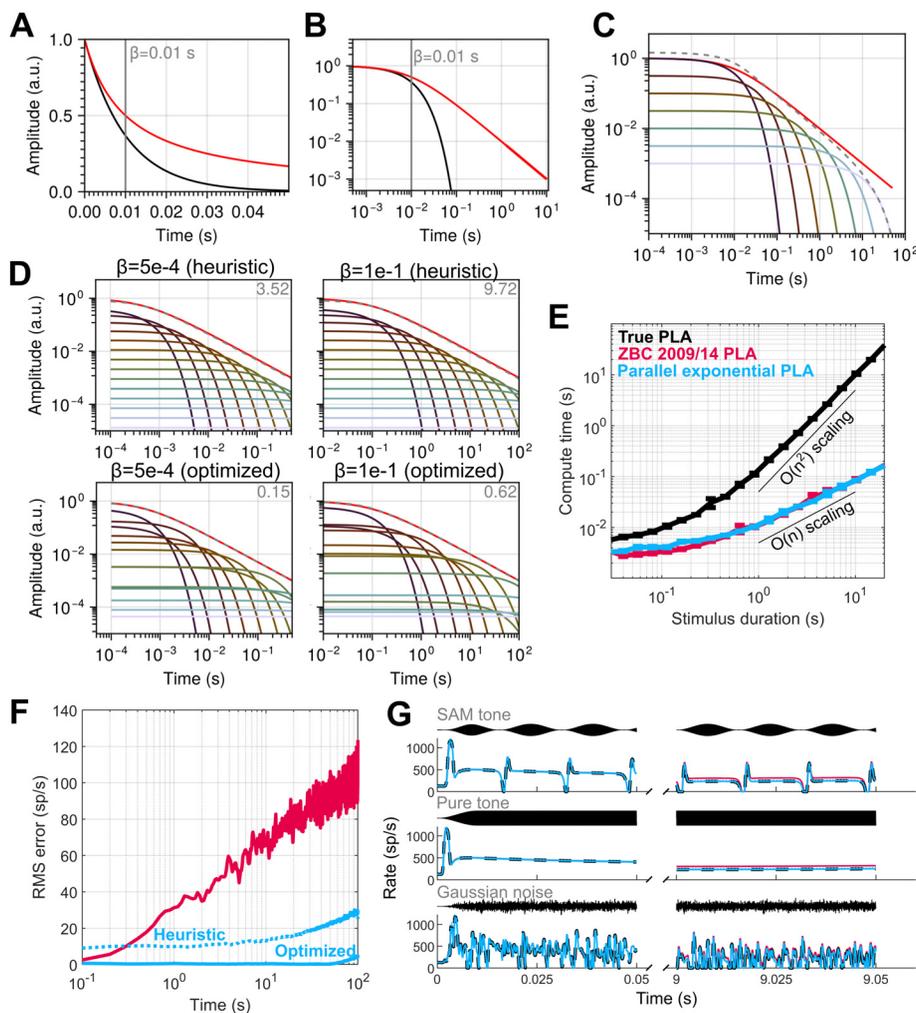


FIG. 1. (Color online) (A) Value of the power-law kernel $1/(t + \beta)$ (red) or of the exponential kernel $\exp(-t/\beta)$ (black) where $\beta = 0.01$ s. (In this and other figures, the peak amplitude of both kernels is normalized to 1 for ease of visualization and labeled in arbitrary units [a.u.]). The vertical gray line indicates β along the time axis. (B) Same as (A), except on a log-log axis. (C) The power-law kernel in (A) (solid red) approximated with seven exponential functions with time constants at every half decade starting at β and ending at $10^3 \beta$ and with weight w_i equal to $1/\tau_i$. The approximation is plotted in dashed gray; contributing exponentials are plotted in solid lines below. (D) Value of the power-law kernel (red) for $\beta = 5 \times 10^{-4}$ s (left column) or $\beta = 1 \times 10^{-1}$ s (right column) compared to the parallel-exponential approximation (gray dashed) with heuristic weights (top row) or with optimized weights (bottom row). The loss metric described in Sec. II is printed in the corner of each panel. (E) Compute time for the synapse stage in the ZBC model versus input stimulus duration. Error bars indicate 95% confidence intervals. The thin black lines indicate predicted scaling between stimulus duration and compute time based on or complexity, adjusted vertically to underline the measurements. (F) RMS error over time in a 200-ms rectangular sliding window for different approximation schemes in response to a SAM tone (carrier frequency = CF = 10 kHz, modulation frequency = 10 Hz, modulation depth = 0 dB, level = 50 dB SPL). The dashed line indicates the heuristic weight scheme while the solid line shows optimized weights. (G) Example high-spontaneous-rate responses using different methods for calculating PLA in different time windows (0–0.2 s or 9–9.2 s) and for different stimuli (from top to bottom): stimulus from C, 10-kHz pure tone at 50 dB SPL, Gaussian noise at 50 dB SPL overall level. The black trace at the top of each panel shows the stimulus waveform. The model response under true PLA is shown in dashed black. Responses approximated with the new scheme are shown in cyan (dotted = heuristic; solid = optimized), while responses approximated with ZBC 2009/14 PLA are shown in red.

on the logarithm of the weights to accommodate the large scale involved and to avoid the need to use parameter boundaries. Numerically optimized weights yielded exceptionally accurate predictions of the power-law kernel [Fig. 1(D), bottom row].

III. RESULTS

Two values of β are used in the ZBC model in the two PLA pathways: 5×10^{-4} and 1×10^{-1} s. Power-law kernels for these values of β were well approximated by both the heuristic and optimized weight schemes [Fig. 1(D)]. The values of the optimized weights are reported in Table I. Our approximation scheme was about as efficient as ZBC 2009/

14 PLA, achieving the expected $O(n)$ scaling behavior as compared to true PLA's $O(n^2)$ scaling behavior [Fig. 1(E)]. To evaluate the approximation in terms of responses to sound, we simulated high-spontaneous-rate AN responses to sound with the Zilany *et al.* (2014) cat model using true PLA or the different approximation schemes. Over the course of a 100-s SAM tone, ZBC 2009/14 PLA matched true PLA well in very early time windows (i.e., first several hundred ms), but consistent with the advice of Zilany *et al.* (2009) that ZBC 2009/14 PLA only be used for short simulations, the match became progressively poorer in later time windows [Fig. 1(F) and 1(G)]. This mismatch was the result of ZBC 2009/14 PLA systematically underestimating the

TABLE I. Value of the weight parameter corresponding to each time constant for numerically optimized parallel-exponential PLA, for either the slow or fast PLA pathway.^a

Coefficient number	Coefficient value for:	
	Slow pathway	Fast pathway
1	1.05413×10^3	6.10664×10^0
2	2.35420×10^2	1.15590×10^0
3	3.51309×10^2	1.30960×10^0
4	9.90012×10^1	7.85696×10^{-1}
5	5.51842×10^1	2.18355×10^{-1}
6	2.89945×10^1	1.03448×10^{-1}
7	6.55613×10^0	8.41393×10^{-2}
8	6.55838×10^0	1.59636×10^{-3}
9	1.15761×10^0	1.88867×10^{-2}
10	9.95489×10^{-1}	8.08962×10^{-4}
11	3.58867×10^{-1}	2.80610×10^{-3}
12	1.57345×10^{-1}	6.52993×10^{-4}
13	1.04288×10^{-2}	3.13954×10^{-5}
14	8.77389×10^{-2}	4.49067×10^{-4}

^aShown are the value of the weight parameter corresponding to each time constant [calculated via Eq. (6) with $N=14$] for numerically optimized parallel-exponential PLA, for either the slow PLA pathway ($\beta = 5 \times 10^{-4}$ s) or fast PLA pathway ($\beta = 1 \times 10^{-1}$ s).

magnitude of adaptation in later time windows (i.e., approximation’s impulse response underestimated the “long tail” of the power-law kernel). In contrast, our new approximation scheme produced a more uniform error across time. The heuristic weight scheme was slightly outperformed by ZBC 2009/14 PLA before ~ 300 ms, but then systematically outperformed it in later time windows, maintaining a constant moderate error before worsening slightly in time windows later than ~ 20 s. In comparison, the numerically optimized weights yielded a nearly perfect simulation of true PLA at all tested times up to 100 s.

IV. DISCUSSION

PLA is a form of neural adaptation that has been observed in AN (van Gendt *et al.*, 2020; Zilany *et al.*, 2009) but is computationally impractical to simulate directly in most applications. We provide here a straightforward strategy to approximate the convolution with a power-law kernel that underlies PLA Eq. (3) with a set of simple IIR filters implemented in parallel. The approximation should be most helpful when simulating responses to long stimuli (e.g., continuous speech) or when high sampling rates are required to avoid aliasing (Heinz *et al.*, 2001) or due to technical constraints, such as simulating feedback pathways (Farhadi *et al.*, 2023). The approach is also portable, in that the optimized weights can be applied to any value of β and the values of τ can be computed on-the-fly for any sampling rate. Future work could explore a number of fruitful directions. First, better analytical strategies for determining parameter values could be developed (e.g., Bochud and Challet, 2007). Second, AN data used to fit PLA parameters could be reanalyzed to determine if a small number of exponential processes might provide a better fit to the data than power-law adaptation. Third, the approximation could be incorporated into the AN

model of Bruce *et al.* (2018). We focused on the Zilany *et al.* (2014) model because its instantaneous rate output (which is popular for modeling psychophysical data) is more suitable for use than the equivalent output of the Bruce *et al.* (2018) model, and because it served as the foundation of the recent efferent model of Farhadi *et al.* (2023). However, the approximation could be easily extended to other models that include PLA.

ACKNOWLEDGMENTS

This work was supported by National Institute on Deafness and Other Communication Disorders (NIH) Grant Nos. R01-DC010813 (L.H.C.) and F32-DC022143 (D.R.G.).

AUTHOR DECLARATIONS

Conflict of Interest

The authors have no conflicts to disclose.

DATA AVAILABILITY

The code and data that support the findings of this study are openly available on the OSF repository at <https://osf.io/gmzh5/>.

¹Note that this convolution is actually computed at a lower sampling rate (10 kHz) following downsampling of its inputs before the output is upsampled back to the main sampling rate (e.g., 100 kHz), exploiting the fact that the inner-hair-cell model’s lowpass filters remove most energy above ~ 5 kHz from the response. This strategy was also adopted for the new approximation for the results shown here, but downsampling is not required by the new approximation.

Bezanson, J., Edelman, A., Karpinski, S., and Shah, V. B. (2017). “Julia: A fresh approach to numerical computing,” *SIAM Rev.* **59**, 65–98.

Bochud, T., and Challet, D. (2007). “Optimal approximations of power laws with exponentials: Application to volatility models with long memory,” *Quant. Finance* **7**, 585–589.

Bruce, I. C., Erfani, Y., and Zilany, M. S. A. (2018). “A phenomenological model of the synapse between the inner hair cell and auditory nerve: Implications of limited neurotransmitter release sites,” *Hear. Res.* **360**, 40–54.

Drew, P. J., and Abbott, L. F. (2006). “Models and properties of power-law adaptation in neural systems,” *J. Neurophysiol.* **96**, 826–833.

Farhadi, A., Jennings, S. G., Strickland, E. A., and Carney, L. H. (2023). “Subcortical auditory model including efferent dynamic gain control with inputs from cochlear nucleus and inferior colliculus,” *J. Acoust. Soc. Am.* **154**, 3644–3659.

Heinz, M. G., Colburn, H. S., and Carney, L. H. (2001). “Evaluating auditory performance limits: I. One-parameter discrimination using a computational model for the auditory nerve,” *Neural Comput.* **13**, 2273–2316.

Knuth, D. E. (1976). “Big omicron and big omega and big theta,” *SIGACT News* **8**, 18–24.

Mogensen, P. K., and Riseth, A. N. (2018). “Optim: A mathematical optimization package for Julia,” *JOSS* **3**, 615.

Newman, M. (2005). “Power laws, Pareto distributions and Zipf’s law,” *Contemp. Phys.* **46**, 323–351.

Rall, L. B. (1986). “The arithmetic of differentiation,” *Math. Mag.* **59**, 275–282.

Revels, J., Lubin, M., and Papamarkou, T. (2016). “Forward-mode automatic differentiation in Julia,” [arXiv:1607.07892v1](https://arxiv.org/abs/1607.07892v1) (Last viewed September 27, 2024).

van Gendt, M. J., Siebrecht, M., Briare, J. J., Bohte, S. M., and Frijns, J. H. M. (2020). “Short and long-term adaptation in the auditory nerve stimulated with high-rate electrical pulse trains are better described by a power law,” *Hear. Res.* **398**, 108090.

Zilany, M. S. A., Bruce, I. C., and Carney, L. H. (2014). “Updated parameters and expanded simulation options for a model of the auditory periphery,” *J. Acoust. Soc. Am.* **135**, 283–286.

Zilany, M. S. A., Bruce, I. C., Nelson, P. C., and Carney, L. H. (2009). “A phenomenological model of the synapse between the inner hair cell and auditory nerve: Long-term adaptation with power-law dynamics,” *J. Acoust. Soc. Am.* **126**, 2390–2412.