Appendix

Here we provide all supplementary materials to our letter.

A Training parameters

Table 1: Training parameters used for training the networks for the MNIST and Fashion-MNIST datasets.

Parameters	Layer		
	Hasani	Euler	Our solution
optimizer	Adam	Adam	Adam
batch size	64	64	64
epochs	100	100	100
learning rate	0.001	0.05	0.05

B Steps for One Dimensional Case

In this section, we provide more details steps for computing the dynamcis of a liquid time-constant (LTC) neuron in the single presynaptic impulse case.

The ordinary differential equation (ODE) behind an LTC neuron with a single presynaptic input with no self-connections are described by [1]

$$\frac{\mathrm{d}}{\mathrm{d}t}x(t) = -\omega x(t) + f(\tilde{g}(t), \sigma, \mu)\big(A - x(t)\big), \tag{1}$$

where $x(t)$ is the time-dependent postsynaptic neuron's potential with $t \in \mathbb{R}$, ω is the inverted post-synaptic neuron's time constant, f is a synaptic release nonlinearity, where its parameters $\sigma \in \mathbb{R}$ and $\mu \in \mathbb{R}$ are learned during the training process, $A \in \mathbb{R}$ a synaptic reversal potential (usually referred to as *bias*), and $\tilde{g}$ a signal describing the presynaptic stimulus.

We represent a 1-dimensional input signal $\tilde{g}$ by a weighted sum of shifted basis functions on the sampling grid as

$$\tilde{g}(t) = \sum_{k\in\mathbb{Z}} c[k]\varphi(\frac{t}{T} - k), \tag{2}$$

where the integer T is the sampling step and $\{c[k]\}_{k\in\mathbb{Z}}$ a sequence of weights. Thereby, $\varphi \in L_2(\mathbb{R})$ is a generator, which we choose to be the constant B-spline interpolator, defined as

$$\beta(t) = \begin{cases} 1, & 0 \leq t < 1 \\ 0, & \text{otherwise.} \end{cases} \tag{3}$$

Here, to simplify readability, we restrict ourselves to sampling on the *regular* grid. However, the extension of the presented theory to the irregularly sampled grid is straight-forward. Without any loss of generality we can set $T = 1$ and thus, (2) is re-expressed as

$$\tilde{g}(t) = \sum_{k \in \mathbb{N}} c[k]\beta(t-k). \tag{4}$$

In the case where the B-spline interpolators are piecewise constants and $t \in [K; K+1[$ with $K = \lfloor t \rfloor \in \mathbb{N}$, it holds that

$$\tilde{g}(t) = c[k]. \tag{5}$$

The solution to (1) is given by

$$x(t) = \frac{1}{\alpha(t)} \left(x(0) + A \int_0^t f(\tilde{g}(u))\alpha(u)\mathrm{d}u \right) \tag{6}$$

with

$$\alpha(t) = e^{\omega t + \int_0^t f(\tilde{g}(u))\mathrm{d}u}.$$

Here we choose the synaptic release nonlinearity to be a sigmoid

$$f(t, \sigma, \mu) = f(t) = \frac{1}{1 + e^{-\sigma(t-\mu)}}. \tag{7}$$

Using(4) and (5) we now compute for $K \leq t < K + 1$ (using decomposition of the piecewise constant signal)

$$\begin{aligned} \int_0^t f(\tilde{g}(u))\mathrm{d}u &= \int_K^t f(c[K])\mathrm{d}u + \sum_{i=0}^{K-1} \int_i^{i+1} f(c[i])\mathrm{d}u \\ &= (t-K)f(c[K]) + \sum_{i=0}^{K-1} f(c[i]). \end{aligned} \tag{8}$$

The integral expression in (6) can also be decomposed for piecewise constant inputs with $K \leq t < K+1$

$$\begin{aligned} \int_0^t f(\tilde{g}(u))\alpha(u)\mathrm{d}u &= \int_K^t f(\tilde{g}(u))\alpha(u)\mathrm{d}u + \sum_{i=0}^{K-1} \int_i^{i+1} f(\tilde{g}(u))\alpha(u)\mathrm{d}u \\ &= f(c[K]) \int_K^t e^{\omega u + \int_0^u f(\tilde{g}(v))\mathrm{d}v}\mathrm{d}u + \sum_{i=0}^{K-1} f(c[i]) \int_i^{i+1} e^{\omega u + \int_0^u f(\tilde{g}(v))\mathrm{d}v}\mathrm{d}u. \end{aligned} \tag{9}$$

Using a change of variable with

$$z(u) = \omega u + \int_0^u f(\tilde{g}(v))\mathrm{d}v = \omega u + (u-K)f(c[K]) + \sum_{i=0}^{K-1} f(c[i]), \tag{10}$$

where

$$\frac{\mathrm{d}}{\mathrm{d}u} z(u) = \omega + f(c[K]), \tag{11}$$

the inner integral expression in Eq.(9) becomes

$$\int_i^{i+1} e^{\omega u + \int_0^u f(\tilde{g}(v))\mathrm{d}v} \mathrm{d}u = \frac{\alpha[i+1] - \alpha[i])}{\omega + f(c[K])}. \tag{12}$$

Therefore (9) becomes

$$\int_0^t f(\tilde{g}(u))\alpha(u)\mathrm{d}u = \frac{f(c[K])(\alpha[t] - \alpha[K])}{\omega + f(c[K])} + \sum_{i=0}^{K-1} \frac{f(c[i])(\alpha[i+1] - \alpha[i])}{\omega + f(c[i])}. \tag{13}$$

The solution to the ODE is given by the following, where we insert (13) into (6).

C Proof of Theorem 1

In this section, we provide the proof to the Theorem 1 given in our letter. It is an extension to the theory revised in the one-dimensional case, with multiple presynapctic inputs that are irregularly sampled.

We focus on the units of one neuron with multiple presynaptic stimulus. Here, we will solve the neuron's dynamic equation involving a total number of S synapses. The equation is derived as [1] [2]

$$\frac{\mathrm{d}}{\mathrm{dt}} x(t) = -\omega x(t) + \sum_{s=0}^{S} f_s(\tilde{g}_s(t), \sigma_s, \mu_s)(A_s - x(t)). \tag{14}$$

Consider a continuous stimulus signal sampled on the discrete domain with piecewise constant B-splines. For a total number of synapses S containing input synapses from a previous layer and connecting synapses between neuron associated to a unique neuron, we have, for all $s \in [1; S]$ and $t \in [\tau_k, \tau_{k+1}[$:

$\tilde{g}_s(t) = c_s[k]$: the presynaptic stimulus for synapse s

$f_s(t, \sigma_s, \mu_s) = f_s(t) = \frac{1}{1+e^{-\sigma_s(t-\mu_s)}}$: synaptic release non-linearity for synapse s

Theorem 1: We compute the exact solution of (14), for piecewise constant input $\tilde{g} = (\tilde{g}_s)_{s\in[1,S]}$ and regularly or irregularly sampled inputs on time samples denoted by $(\tau_k)_{k\in\mathbb{N}}$ and $t \in [\tau_k, \tau_{k+1}[$:

$$x(t) = \frac{1}{\alpha(t)} \left(x(0) + \Delta(t) \sum_{s=0}^{S} A_s u_s(\tau_k) + \sum_{i=0}^{k-1} \Delta(\tau_i) \sum_{s=0}^{S} A_s u_s(\tau_i) \right) \tag{15}$$

Where

$$\alpha(t) = e^{\omega t + \sum_{s=0}^{S}(t-\tau_k) f_s(c_s[k]) + \sum_{i=0}^{k-1}\sum_{s=0}^{S}(\tau_{i+1}-\tau_i) f_s(c_s[i])},$$
$$u_s(\tau_i) = \frac{f_s(c_s[i])}{\omega + \sum_{r=1}^{S} f_r(c_r[i])},$$
$$\Delta(\tau_i) = \alpha(\tau_{i+1}) - \alpha(\tau_i),$$
$$\Delta(t) = \alpha(t) - \alpha(\tau_k).$$

Proof theorem 1

Let's denote

$$\eta_S(t) = \sum_{s=1}^{S} f_s(\tilde{g}_s(t)), \tag{16}$$

$$c_S(t) = \sum_{s=1}^{S} A_s f_s(\tilde{g}_s(t)). \tag{17}$$

The equivalent problem to the ODE in (14) is

$$\frac{\mathrm{d}}{\mathrm{d}t} x(t) = -(\omega + \eta_S(t))x(t) + c_S(t). \tag{18}$$

Let us define for piecewise constant inputs and $t \in [\tau_k, \tau_{k+1}[$

$$\begin{aligned}
\alpha_S(t) &= e^{\int_0^t (\omega + \eta_S(u))du} = e^{\omega t + \int_0^t \eta_S(u)du} \\
&= e^{\omega t + \sum_{s=1}^{S}\int_{\tau_k}^{t} f_s(\tilde{g}_s(u)))du + \sum_{s=1}^{S}\sum_{i=0}^{k-1}\int_{\tau_i}^{\tau_{i+1}} f_s(\tilde{g}_s(u))du} \\
&= e^{\omega t + \sum_{s=1}^{S}(t-\tau_k) f_s(c_s[k]) + \sum_{s=1}^{S}\sum_{i=0}^{k-1}(\tau_{i+1}-\tau_i) f_s(c_s[i])} \\
&= \alpha(t).
\end{aligned} \tag{19}$$

By multiplying by α_S the equation (18), we obtain:

$$\alpha_S(t)\frac{\mathrm{d}}{\mathrm{d}t} x(t) = -(\omega + \eta_S(t))\alpha_S(t)x(t) + c_S(t)\alpha_S(t). \tag{20}$$

When taking the derivative of $\alpha_S(t)x(t)$, we observe that

$$\alpha_S(t)\frac{\mathrm{d}}{\mathrm{d}t} x(t) + (\omega + \eta_S(t))\alpha_S(t)x(t) = \frac{\mathrm{d}}{\mathrm{d}t}\alpha_S(t)x(t). \tag{21}$$

Inserting (21) into (20), we have

$$\alpha_S(t)x(t) = \int_0^t c_S(u)\alpha_S(u)du + C. \tag{22}$$

Let us find C:

$$\alpha_S(t=0)=1 \quad \Rightarrow \quad C=x(t=0) \tag{23}$$

$$\alpha_S(t)x(t)-x(0)=\int_0^t c_S(u)\alpha_S(u)du \tag{24}$$

Let's now compute $\int_0^t c_S(u)\alpha_S(u)du$:

$$\begin{aligned}
\int_0^t c_S(u)\alpha_S(u)du &= \int_0^t \sum_{s=1}^{S} A_s f_s(g_s(u))\alpha_S(u)du \\
&= \sum_{s=1}^{S} A_s \int_0^t f_s(g_s(u))\alpha_S(u)du \\
&= \sum_{s=1}^{S} A_s I_S(t,s),
\end{aligned} \tag{25}$$

where we define $I_S(t,s)=\int_0^t f_s(\tilde{g}_s(u))\alpha_S(u)du$. This expression for piecewise constant inputs becomes

$$\begin{aligned}
I_S(t,s) &= \int_0^t f_s(\tilde{g}_s(u))\alpha_S(u)du \\
&= \int_{\tau_k}^t f_s(\tilde{g}_s(u))\alpha_S(u)du + \sum_{i=0}^{k-1}\int_{\tau_i}^{\tau_{i+1}} f_s(\tilde{g}_s(u))\alpha_S(u)du \\
&= \int_{\tau_k}^t f_s(\tilde{g}_s(u))e^{\omega u+\int_0^u \eta_S(v)dv}du + \sum_{i=0}^{k-1}\int_{\tau_i}^{\tau_{i+1}} f_s(\tilde{g}_s(u))e^{\omega u+\int_0^u \eta_S(v)dv}du.
\end{aligned} \tag{26}$$

As $u \in [\tau_i, \tau_{i+1}[$, we have

$$\begin{aligned}
I_S(t,s) = &\int_{\tau_k}^t f_s(c_s[k])e^{\omega u+\sum_{r=1}^{S}(u-\tau_k)f_r(c_r[k])+\sum_{r=1}^{S}\sum_{j=0}^{k-1}(\tau_{j+1}-\tau_j)f_r(c_r[j])}du \\
&+\sum_{i=0}^{k-1}\int_{\tau_i}^{\tau_{i+1}} f_s(c_s[i])e^{\omega u+\sum_{r=1}^{S}(u-\tau_i)f_r(c_r[i])+\sum_{r=1}^{S}\sum_{j=0}^{i-1}(\tau_{j+1}-\tau_j)f_r(c_r[j])}du.
\end{aligned}$$

Using a change of variable with

$$z(u)=\omega u+\sum_{r=1}^{S}(u-\tau_i)f_r(c_r[i])+\sum_{r=1}^{S}\sum_{j=0}^{i-1}(\tau_{j+1}-\tau_j)f_r(c_r[j]) \tag{27}$$

where

$$\frac{\mathrm{d}}{\mathrm{d}u}z(u)=\omega+\sum_{r=1}^{S}f_r(c_r[k]), \tag{28}$$

we obtain

$$I_S(t,s) = f_s(c_s[k])\frac{e^{\omega t+\int_0^t \eta_S(u)du} - e^{\omega\tau_k+\int_0^{\tau_k}\eta_S(u)du}}{\omega+\sum_{r=1}^S f_r(c_r[k])} + \sum_{i=0}^{k-1} f_s(c_s[i])\frac{e^{\omega\tau_{i+1}+\int_0^{\tau_{i+1}}\eta_S(u)du} - e^{\omega\tau_i+\int_0^{\tau_i}\eta_S(u)du}}{\omega+\sum_{r=1}^S f_r(c_r[i])}. \tag{29}$$

Then, from (22), the final expression for the neuron's dynamic is the following:

$$\begin{aligned} x(t) = \alpha^{-1}(t)\Bigg(x(0) + \sum_{s=1}^S A_s\Bigg(& f_s(c_s[k])\frac{e^{\omega t+\int_0^t \eta_S(u)\,du} - e^{\omega\tau_k+\int_0^{\tau_k}\eta_S(u)\,du}}{\omega+\sum_{r=1}^S f_r(c_r[k])} \\ &+ \sum_{i=0}^{k-1} f_s(c_s[i])\frac{e^{\omega\tau_{i+1}+\int_0^{\tau_{i+1}}\eta_S(u)\,du} - e^{\omega\tau_i+\int_0^{\tau_i}\eta_S(u)\,du}}{\omega+\sum_{r=1}^S f_r(c_r[i])}\Bigg)\Bigg) \end{aligned} \tag{30}$$

$$x(t) = \alpha^{-1}(t)\left(x(0) + \sum_{s=1}^S A_s f_s(c_s[k])\frac{\alpha(t)-\alpha(\tau_k)}{\omega+\eta_S[\tau_k]} + \sum_{s=1}^S\sum_{i=0}^{k-1} A_s f_s(c_s[i])\frac{\alpha(\tau_{i+1})-\alpha(\tau_i)}{\omega+\eta_S(\tau_i)}\right) \tag{31}$$

To simplify the given expression for $x(t)$:

$$\begin{aligned} x(t) &= \alpha^{-1}(t)\left(x(0) + (\alpha(t)-\alpha(\tau_k))\sum_{s=0}^S A_s u_s(\tau_k) + \sum_{i=0}^{k-1}(\alpha(\tau_{i+1})-\alpha(\tau_i))\sum_{s=0}^S A_s u_s(\tau_i))\right) \\ &= \alpha^{-1}(t)\left(x(0) + \Delta(t)\sum_{s=0}^S A_s u_s(\tau_k) + \sum_{i=0}^{k-1}\Delta(\tau_i)\sum_{s=0}^S A_s u_s(\tau_i)\right). \end{aligned}$$

□

D Proof of Corollary 1

Corollary 1: We observe a recursive relationship between τ_{k+1} and τ_k:

$$x(\tau_{k+1}) = e^{-\omega(\tau_{k+1}-\tau_k)-\sum_{s=0}^S(\tau_{k+1}-\tau_k)f_s(c_s[k]))}\left(x(\tau_k) - \sum_{s=0}^S A_s w_s(\tau_k)\right) + \sum_{s=0}^S A_s w_s(\tau_k)$$

D.1 Proof corollary 1

$$\begin{aligned} x(\tau_{k+1}) &= \alpha^{-1}(\tau_{k+1})\left(x(0) + \sum_{i=0}^k \Delta(\tau_i)\sum_{s=0}^S A_s w_s(\tau_i)\right) \\ &= \alpha^{-1}(\tau_{k+1})\left(\alpha(\tau_k)x(\tau_k) + \Delta(\tau_k)\sum_{s=0}^S A_s w_s(\tau_k)\right) \\ &= \alpha^{-1}(\tau_{k+1})\alpha(\tau_k)\left(x(\tau_k) - \sum_{s=0}^S A_s w_s(\tau_k)\right)) + \sum_{s=0}^S A_s w_s(\tau_k) \end{aligned}$$

□

References

[1] R. Hasani, M. Lechner, A. Amini, D. Rus, and R. Grosu, "Liquid time-constant networks," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 9, pp. 7657–7666, May 2021. [Online]. Available: https://ojs.aaai.org/index.php/AAAI/article/view/16936

[2] S. R. Wicks, C. J. Roehrig, and C. H. Rankin, "A dynamic network simulation of the nematode tap withdrawal circuit: predictions concerning synaptic function using behavioral criteria," *J. Neurosci.*, vol. 16, no. 12, pp. 4017–4031, Jun. 1996.