ICASSP 2020

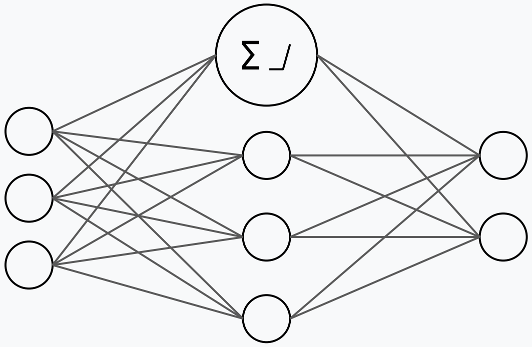
Temporal Coding in Spiking Neural Networks with Alpha Synaptic Function

Iulia M. Comșa

Google Research

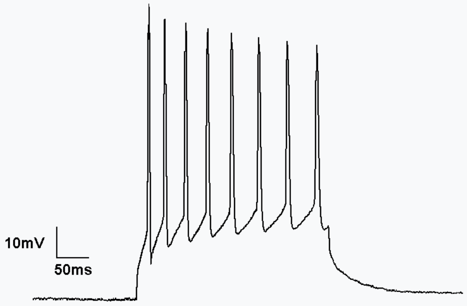
Conventional ANNs

- Inspired by the human brain
- Benchmarks on tasks solved by humans
- ...but compute in a fundamentally different way compared to the biological brain
- Lack a **time dimension**



Biological networks

- Neuron spikes are **discrete events**
- Asynchronous
- Can encode information in temporal patterns of activity
- Energy-efficient

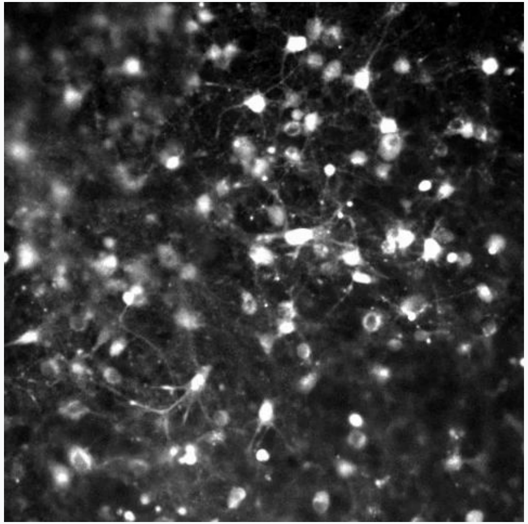


Source: https://en.wikipedia.org/wiki/Electrophysiology

Google Research

Information coding in biological brains

- Conventional approach: rate coding
  - **slowly** accumulate over spikes
- Alternative: temporal coding
  - single spikes at **precise times**
  - **fast** but possibly less accurate
- Information is carried by relative spike times
  - retinal ganglion cells encode the spatial structure of an image in the relative timing of their first spikes (Gollish & Meister, 2008)
  - tactile afferents encode information about fingertip events in the relative timing of the first spikes (Johansson & Birznieks, 2004)
- Single spikes carry information across brain areas
  - 10 synaptic stages crossed within 100ms in the visual system, suggesting that responses are made on the basis of single spikes (Thorpe & Imbert, 2016)

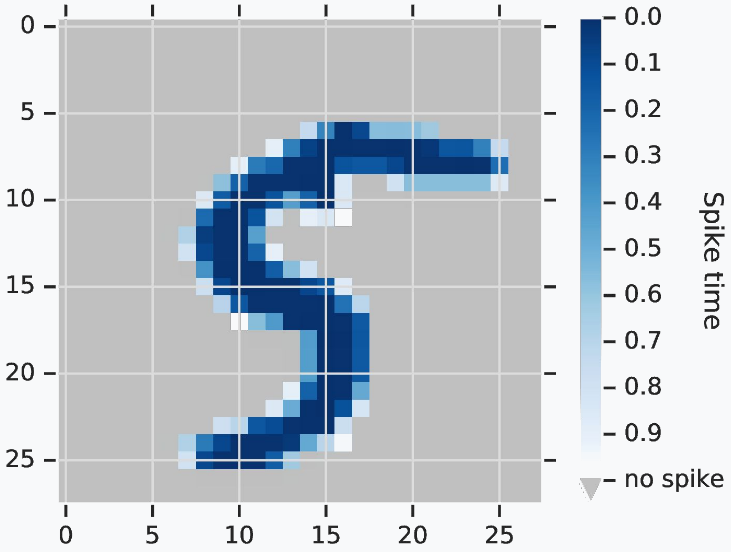


Source: https://www.youtube.com/watch?v=yy994HpFudc
by Michelle Kuykendal and Gareth Guvanasen

Information can be encoded in the timing of individual spikes.

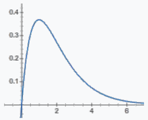
Artificial networks with temporal coding

- Earlier spikes encode more salient information
- Consider a classification problem with $\mathtt{m}$ inputs and $\mathtt{n}$ possible classes
- Temporal encoding
  - $\mathtt{m}$ input neurons
  - spike at time proportional to brightness of corresponding input pixel
- Temporal decoding
  - $\mathtt{n}$ output neurons
  - class $\mathtt{k}$ iff $\mathtt{k}^{\text{th}}$ neuron spikes earliest

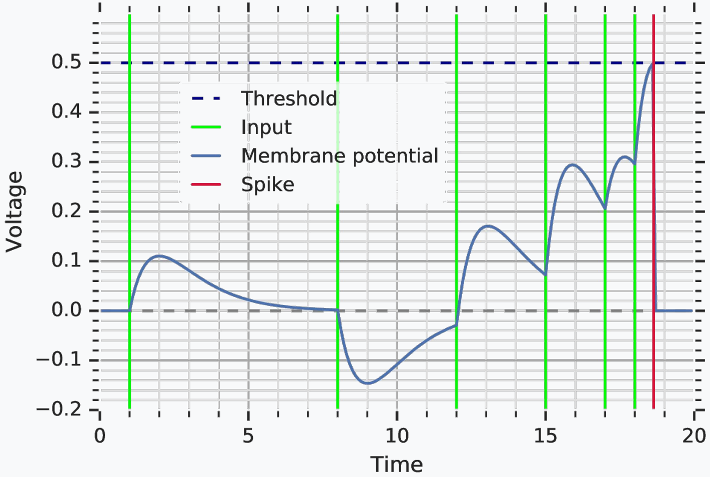


Google Research

Building a spiking model

- Alpha synaptic function: $t \cdot e^{-\tau t}$ 

- Custom decay: τ
- Weighted presynaptic inputs: w_i
- Fixed spiking threshold: θ
- May not spike
- Allows forgetting inputs
- Richer dynamics than nonleaky models
- Computing the spike time given a set I of inputs: $t_{out} = \frac{B_I}{A_I} - \frac{1}{\tau} W\left(-\tau \frac{\theta}{A_I} e^{\tau \frac{B_I}{A_I}}\right)$ where $A_I = \sum_{i \in I} w_i e^{\tau t_i}$, $B_I = \sum_{i \in I} w_i e^{\tau t_i} t_i$



$$V(t) = \sum_{i \in I} w_i (t - t_i) e^{\tau (t_i - t)}$$

Learning with backpropagation

- Usual problem with spiking networks: non-differentiable spike events
- Learning goal with temporal coding: adjust the timing of outputs
- Postsynaptic spike times depend on presynaptic spike times and their weights

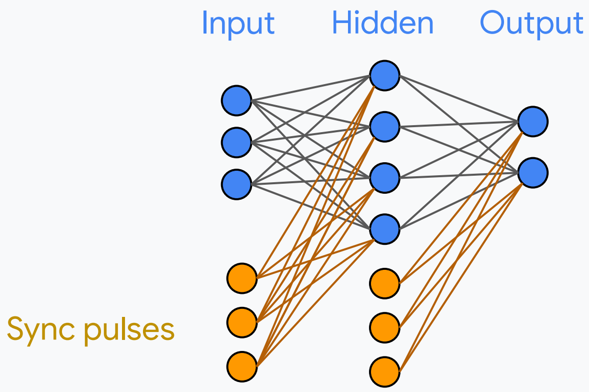
$$\frac{\partial t_{out}}{\partial t_j} = \frac{w_j e^{\tau t_j}((t_j - \frac{B_I}{A_I})\tau + W_I + 1)}{A_I(1 + W_I)} \qquad \frac{\partial t_{out}}{\partial w_j} = \frac{e^{\tau t_j}(t_j - \frac{B_I}{A_I} + \frac{W_I}{\tau})}{A_I(1 + W_I)}$$

where $\bar{W}_I = \bar{W}(-\frac{\theta}{A_I} e^{\frac{B_I}{A_I}})$

- Minimize cross-entropy loss with Adam optimizer
- At the output layer, minimize spike time: softmax of negative spike times

Synchronization pulses

- A set of neurons connected to each non-input layer
- Act like temporal biases
- Ensure there are spikes (eventually)
- Learnable spike times and weights



MNIST experiment

Hyperparameter	Search space	Chosen value
Decay constant τ	[0.1, 2]	0.18
Fire threshold θ	[0.1, 1.5]	1.17
Number of hidden layers.	[0, 4] × [2, 1000]*	1 × 340
Number of pulses per layer.	[0, 10]	10
Multiplier for non-pulse weights initialization.	[-10, 10]	-0.275
Multiplier for pulse weights initialization.	[-10, 10]	7.84
Learning rate for network weights.	$[10^{-5}, 1]$*	$10^{-4} \times 2.02$
Learning rate for pulse timings.	$[10^{-5}, 1]$*	$10^{-2} \times 5.95$
Mini-batch size for Adam optimization.	[1, 1000]*	5
Clipping value for derivatives.	[1, 1000]	539.7
Penalty added to presynaptic weights if a neuron didn't fire.	[0, 100]	48.38

* - logarithmic search space

MNIST experiment

- Network size 784x340x10 (plus 10 synchronization pulses)
- MNIST digits were encoded as spikes at times between 0 and 1
- Pulses were initialized to spike at evenly distributed times between 0 and 1

- Results: **97.96%** accuracy on MNIST test
- For comparison: a **non-convolutional** ReLU DNN achieves 97.9%

Google Research

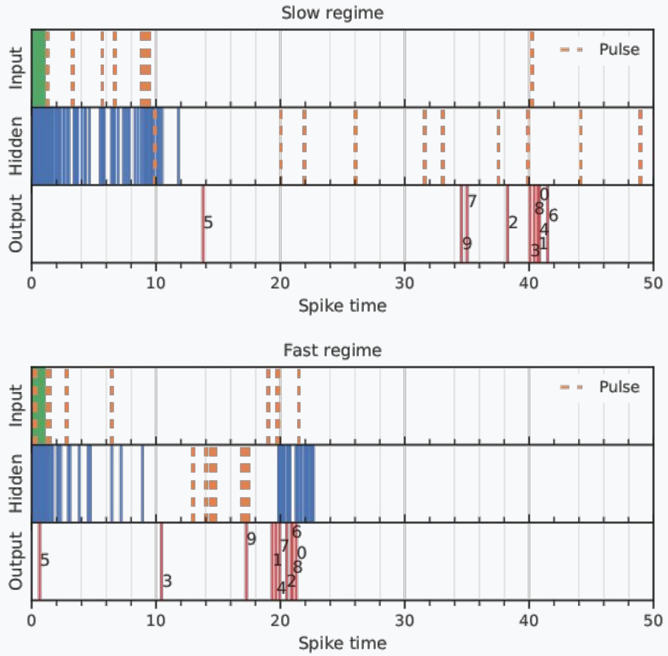
Slow and fast classification regimes

During training, the same network **spontaneously** switched between two operating regimes:

- slow but more accurate
- very fast but less accurate

	Slow regime	Fast regime
Training accuracy (%)	99.9633	99.885
Training loss (mean)	0.002884	0.00444
Test accuracy (%)	97.96	97.4
Test loss (mean)	0.173248	0.19768

The same speed-accuracy trade-off is observed in human decision making.



Google Research

Take-away points

- The timing of single spikes efficiently encodes information in biological brains.
- Spiking networks with temporal coding:
  - can be trained with backpropagation
  - can perform digit recognition at competitive accuracies
- Interesting from a multidisciplinary perspective:
  - shed light on the representational capabilities of biological-like networks
  - possible model for efficient neuromorphic computing
- Opening pathways towards spiking nets research:
  - recurrent, state-based spiking networks that perform efficient computation

Open-source code available

https://github.com/google/ihmehimmeli

Thank you!

Iulia Comșa

Krzysztof Potempa

Luca Versari

Thomas Fischbacher

Andrea Gesmundo

Jyrki Alakuijala

Google Research