Optimized spiking neurons classify images with high accuracy through temporal coding with two spikes [5]

Nature Machine Intelligence 2021

May 18, 2021

Content

Introduction

The spiking neural network (SNN) is emerging as a promising candidate for the third generation of neural networks. Compared with the previously-proposed artificial neural networks (ANNs), SNN has the advantage of temporal information processing capability, low power consumption, and high biological plausibility.

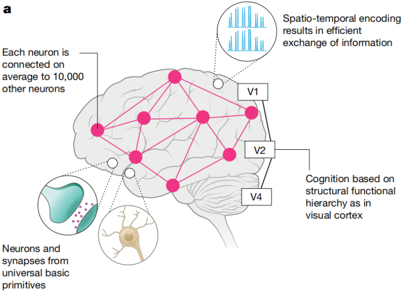

Figure 1: Inspiration of spiking neural networks

Background

Spiking Neural Network

In spiking neural networks, the medium that carry information are spike trains. Inputs are encoded into input spike trains and output spike trains are translated into final result. Between them is the neurons that collect input spikes and fire output spikes.

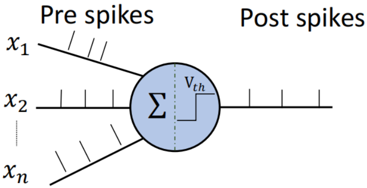

Figure 2: An overview of spiking neural networks

The neuron are responsible for receive spikes from previous layer and fire spikes to the next layer. The most most commonly used model is integrate-and-fire (IF) neuron.

In a fully connected layer, The input current at time t is the weighted sum of pre-synaptic spikes:

$$I_i^n(t) = \sum_j w_{ij} s_j^{n-1}(t) \tag{1}$$

where n is the layer index, I^n is the input current, $s(t)$ is spike train.

The current are then accumulated into the membrane voltage V.

$$V^n(t+1) = V^n(t) + I^n(t)$$

Then, the neuron will emit an output spike if the membrane voltage exceed the threshold:

$$s^n(t) = \Theta(V^n(t) - V_{threshold}), \tag{2}$$

where $\Theta(x)$ is the Heaviside function

$$\Theta(x) = \begin{cases} 1 & x > 0, \\ 0 & x \leqslant 0, \end{cases} \tag{3}$$

Background

Spiking Neural Network - Neurons

When the neuron fires a spike, the membrane voltage will be reset. For hard reset method, the membrane voltage will be reset to a fixed value V_{reset}:

$$V^n(t) = V^{reset}(t) \cdot s^n(t) + V^n(t) \cdot (1 - s^n(t)) \tag{4}$$

For soft reset, the membrane voltage will be deduct for a certain value:

$$V^n(t) = -V^{reset}(t) \cdot s^n(t) + V^n(t) \tag{5}$$

ANN-SNN Conversion

Up to now, the method with highest accuracy is ANN-SNN conversion method, which directly inherit the weight from trained ANN and substitute the ReLU with IF neuron. [4, 6]

The input values are interpreted as the frequency of input spike trains and the post-synaptic frequency are corresponding to the activation value. It has been proved that IF neuron is mathematically equivalent to ReLU function and the slope is inversely proportional to the threshold voltage.

ANN-SNN Conversion

However, there are several drawbacks of such method.

The first problem is the large number of time steps. Usually several thousands of time steps are needed to reach comparative result with ANNs. One of reason is that frequency based encoding method ignored the temporal information that spikes have. For a spike train of 1023 time steps long, frequency encoding only have a resolution of 1024: $0/1023$ — $1023/1023$. However, in an ideal situation, only 10 time steps are needed to perform such a resolution: 0000000000_2 — 1111111111_2.

Also, the frequency based encoding will inevitably generate a large number of spikes, which will reduce the main advantage of SNN, low power consumption.

ANN-SNN Conversion

In addition, the method can only simulate ReLU activation. It can not simulate activation functions that are not linear or have negative output, such as swish: $x \cdot \sigma(x)$, where σ is the sigmoid function. In ImageNet, one of the most successful model up to now is EfficientNet, which uses swish function as its activation function.

Methodology

To solve the above-mentioned problem, the paper proposed a method called FS-conversion. To perform the method, a modified IF neuron called FS-neuron is introduced.

The neuron only has one-off input at the first time step and uses soft reset. In addition, the threshold $T(t)$ and reset amount $h(t)$ is a function of time. Given the initial input x, the membrane voltage $v(t)$ and $z(t)$ can be represented as

$$v(t+1) = v(t) - h(z)z(t), \qquad v(0) = x,$$

$$z(t) = \Theta(v(t) - T(t))$$

where Θ is again the Heaviside function.

Methodology

After that the output value that serves as the activation output is calculated as $y = \sum_{i=1}^{K} d(t)z(t)$, where K is number of time steps.

Therefore, we can see that y is a function of x, with the parameter $h(t)$, $T(t)$ and $d(t)$.

$$y = \hat{f}(x; h, d, T)$$

Then, h, d, T can be trained to approximate $\hat{f}$ to any activation f.

Methodology

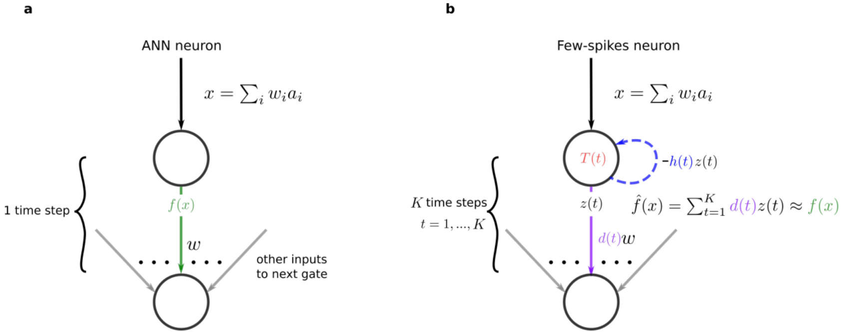

Figure 3: Comparison of tradition ANN neuron and FS-neuron.

Methodology

For example, to approximate the ReLU function at the range $[0, 2^K - 1]$, $T(t) = h(t) = d(t) = 2^{K-t}$ are set. Then $\hat{f}(x) = ReLU(x)$ for integers in $[0, 2^K - 1]$.

$e.g.$, $x = 7$, then spikes will appear at the last 3 time steps and $y = \sum_{i=1}^{K} d(t)z(t) = 4 + 2 + 1 = 7$. In addition, the range can be changed to α by applying an factor of $\alpha 2^{-K}$ to h, d, T.

Methodology

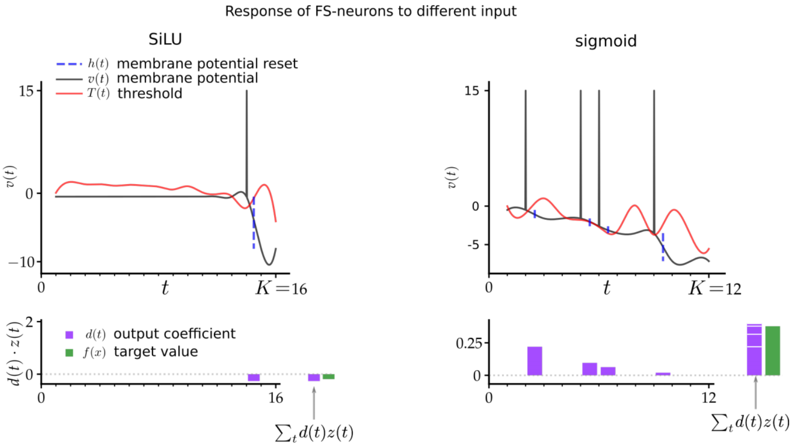

Figure 4: Response of FS-neuron to different input. Here all the curves should be scattered dots, but the author fitted them into a line for unknown reason, which makes the result puzzling to read.

Methodology

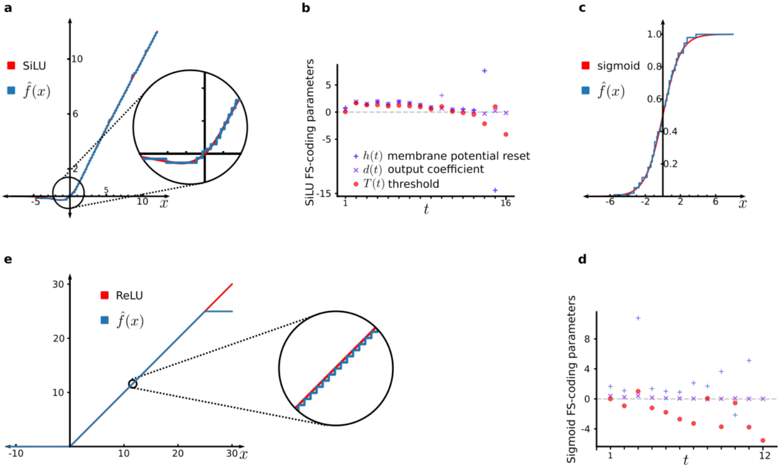

Figure 5: Approximation result of FS-neuron and activation functions. a,b: swish; c,d: sigmoid; e: ReLU.

Hardware Compatibility

The authors have also shown that their method are compatible with hardware SNN implementations such as SpiNNaker [3], Intel Loihi [2] and BrainScaleS-2 [1], without much modification. The only extra memory needed is the cost to save $h(t), d(t)$ and $T(t)$. Since the same activation share the same set of them, which will consume little memory.

Experiment

Model	ANN accuracy	accuracy of the SNN produced by FS-conversion	# params	# layers	# neurons	# spikes
ImageNet2012						
EfficientNet-B7	85% (97.2 %)	83.57% (96.7%)	66M	218	259M	554.9M
ResNet50	75.22% (92.4%)	75.10% (92.36%)	26M	50	9.6M	14.045M
CIFAR10						
ResNet8	87.22%	87.05%	78k	8	73k	103k
ResNet14	90.49%	90.39%	174k	14	131k	190k
ResNet20	91.58%	91.45%	271k	20	188k	261k
ResNet50	92.99%	92.42%	755k	50	475k	647k

Figure 6: Result of FS-conversion on ImageNet and Cifar10. The value in parentheses is top-5 accuracy and almost all the neurons fire for less than or equal to 2 times.

Reference I

[1] Sebastian Billaudelle et al. "Versatile emulation of spiking neural networks on an accelerated neuromorphic substrate". In: *2020 IEEE International Symposium on Circuits and Systems (ISCAS)*. IEEE. 2020, pp. 1–5.

[2] Mike Davies et al. "Loihi: A neuromorphic manycore processor with on-chip learning". In: *Ieee Micro* 38.1 (2018), pp. 82–99.

[3] Muhammad Mukaram Khan et al. "SpiNNaker: mapping neural networks onto a massively-parallel chip multiprocessor". In: *2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence)*. Ieee. 2008, pp. 2849–2856.

[4] Bodo Rueckauer et al. "Conversion of continuous-valued deep networks to efficient event-driven networks for image classification". In: *Frontiers in neuroscience* 11 (2017), p. 682.

Reference II

[5] Christoph Stöckl and Wolfgang Maass. "Optimized spiking neurons can classify images with high accuracy through temporal coding with two spikes". In: *Nature Machine Intelligence* 3.3 (2021), pp. 230–238.

[6] Amirhossein Tavanaei and Anthony Maida. "BP-STDP: Approximating backpropagation using spike timing dependent plasticity". In: *Neurocomputing* 330 (2019), pp. 39–47.