Classifying Russian speech commands with a hardware-deployable spiking neural network transferred from an artificial neural network

Alexey Serenko, Roman Rybka, Alexander Naumov, Alexander Sboev

National research centre Kurchatov Institute

National research nuclear university MEPhI

The purpose of this work

is to establish a baseline accuracy on the task of classifying Russian speech commands

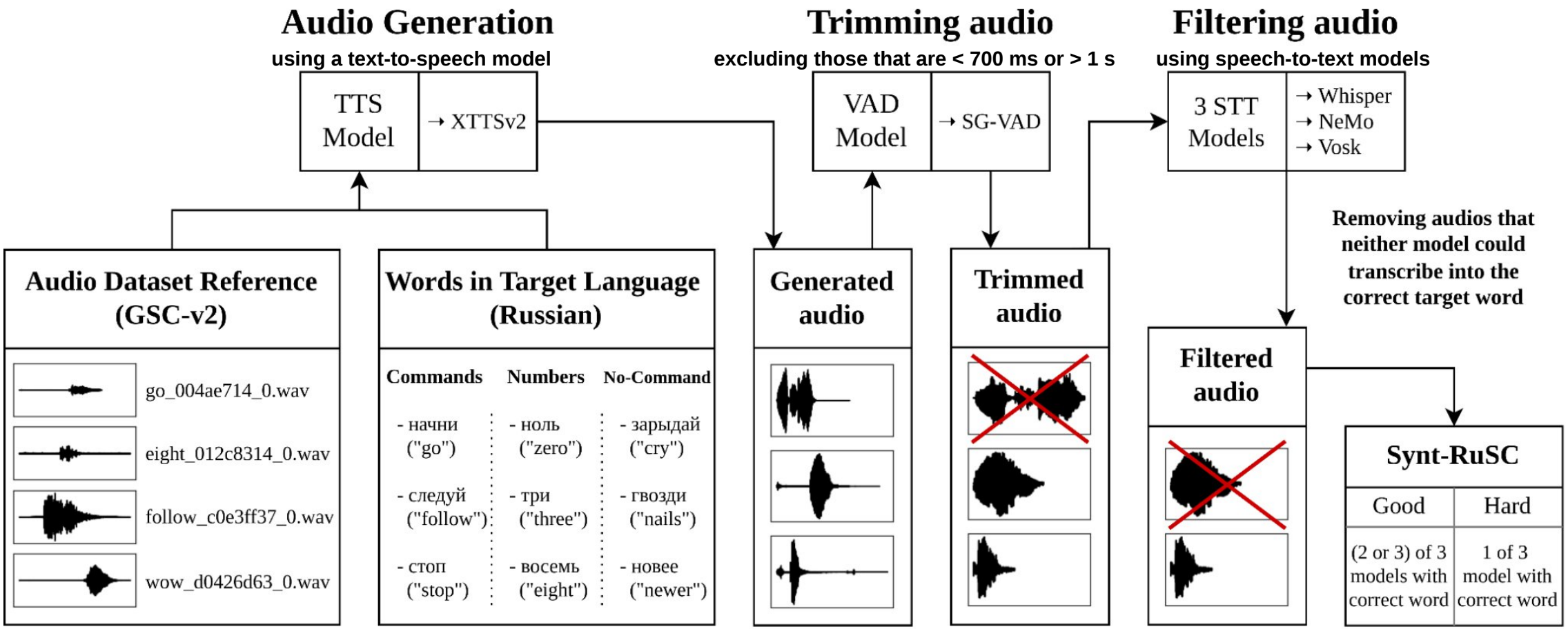
- using a spiking neural network
- that would satisfy the constraints of digital neuroprocessors

and evaluate accuracy loss with respect to a conventional neural network

We thus present:

- A spiking neural network obtained by converting a trained conventional network;
  - with its weights quantized into 8-bit integer;
  - with its neurons' thresholds adjusted so as to minimize the conversion loss.
- Its accuracies on the recently-created RuSC dataset.

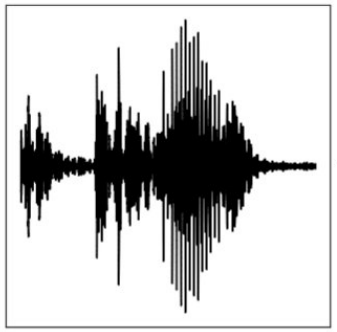
The RuSC dataset



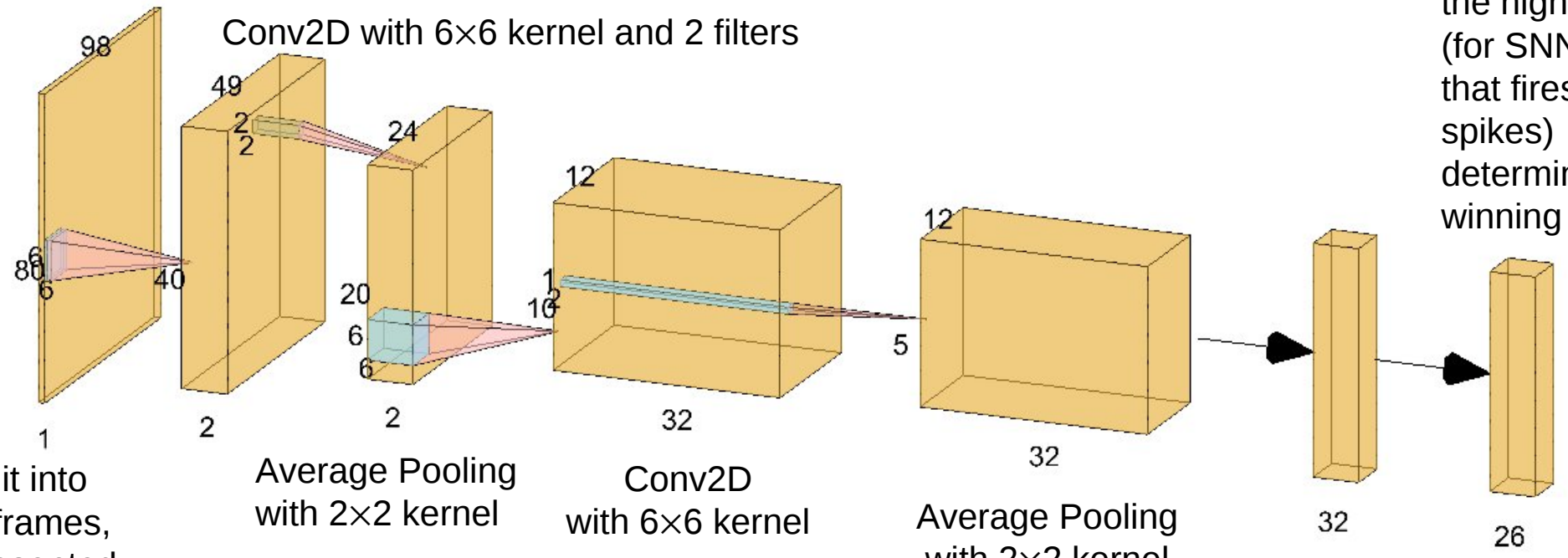
Words:	15	10	14	26 classes total
Recordings: training set	27 402	26 115	3 005	56 522 total
validation set	3 267	3 059	419	6 745 total
testing set	3 285	2 295	420	6 000 total

Synth-ruSC: Construction and Validation of Synthetic Dataset to Solve the Problem of Keyword Spotting in Russian
// Rybka R.B., Naumov A.V., Gryaznov A.V., Sboev A.G.
Neuroinformatics 2024

Network scheme



1-s-long audio



split into
98 frames,
represented
by 80 MFCC

For the spiking
network,
then encoded with
98×80 spike
sequences

Conv2D with 6×6 kernel and 2 filters

Average Pooling
with 2×2 kernel

Conv2D
with 6×6 kernel

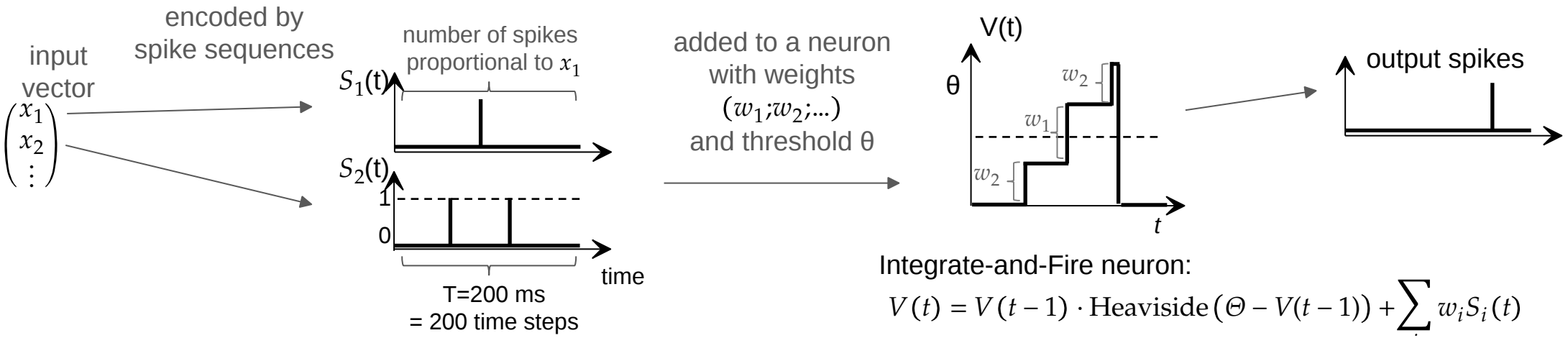
Average Pooling
with 2×2 kernel

Fully-connected
layers

The neuron with
the highest output
(for SNN, the one
that fires the most
spikes)
determines the
winning class

Transferring weights from a ReLU neuron to a spiking neuron

Spiking neuron: the number of output spikes is $\mathrm{ClipFloor}\left(\vec{x}\cdot\vec{w}\right)=\min\left(\left\lfloor\frac{\max\left(\vec{x}\cdot\vec{w},0\right)}{\Theta}\right\rfloor,T\right)$

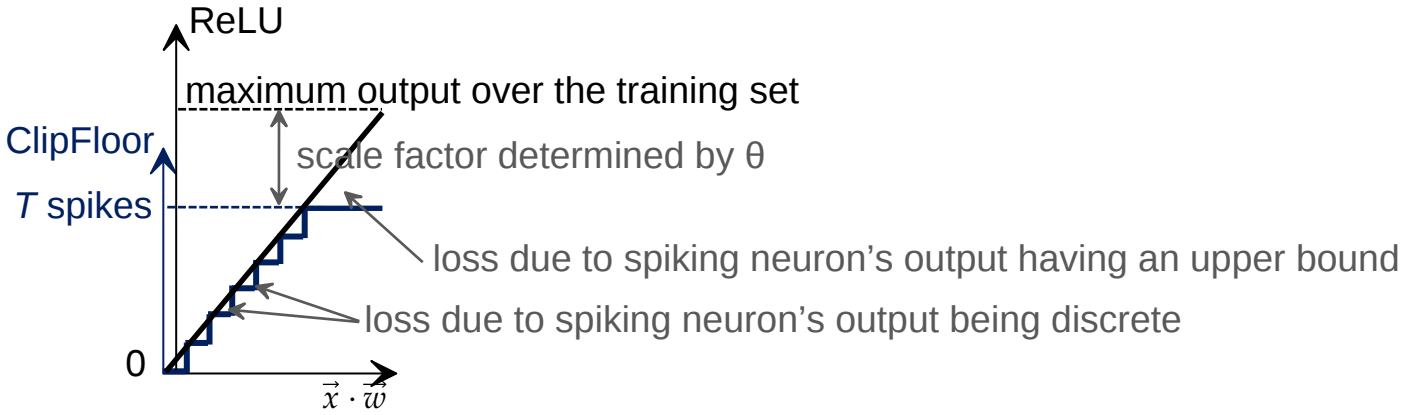


Integrate-and-Fire neuron:

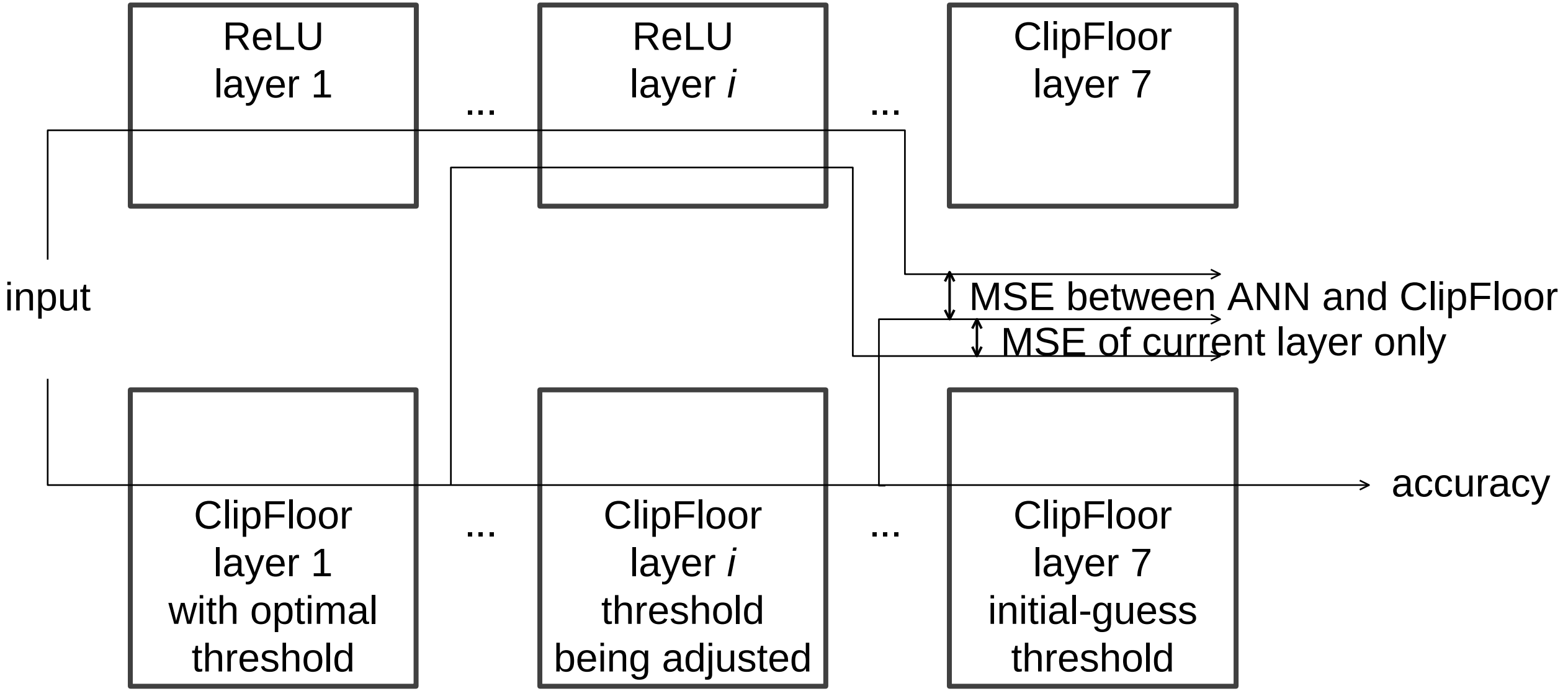
$$V(t)=V(t-1)\cdot\mathrm{Heaviside}\left(\Theta-V(t-1)\right)+\sum_i w_i S_i(t)$$

Conventional neuron: $\mathrm{ReLU}\left(\vec{x}\cdot\vec{w}\right)=\max\left(\vec{x}\cdot\vec{w},0\right)$

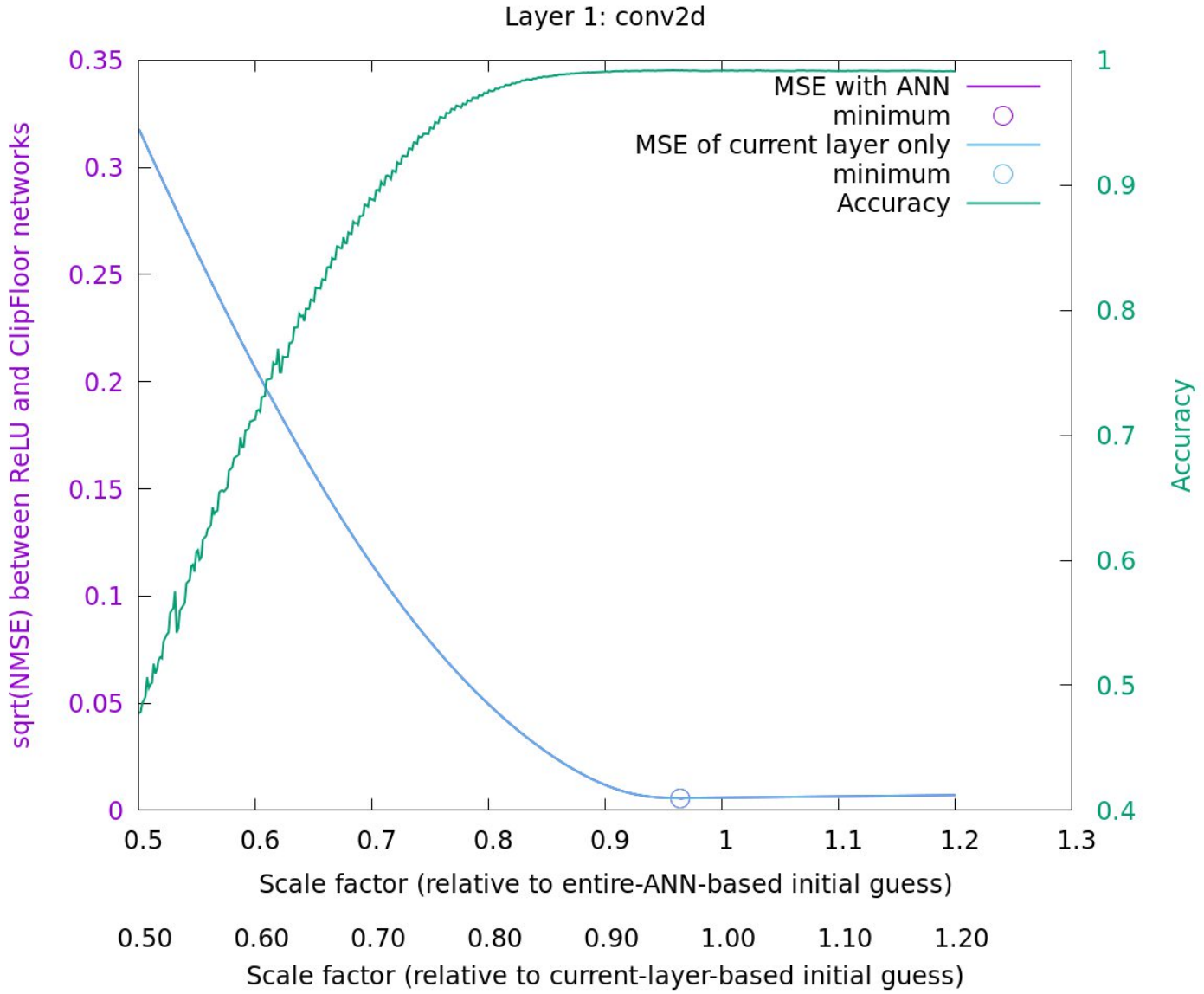
Transferring weights cause accuracy loss due to the number of spikes differing from the ReLU activation function:



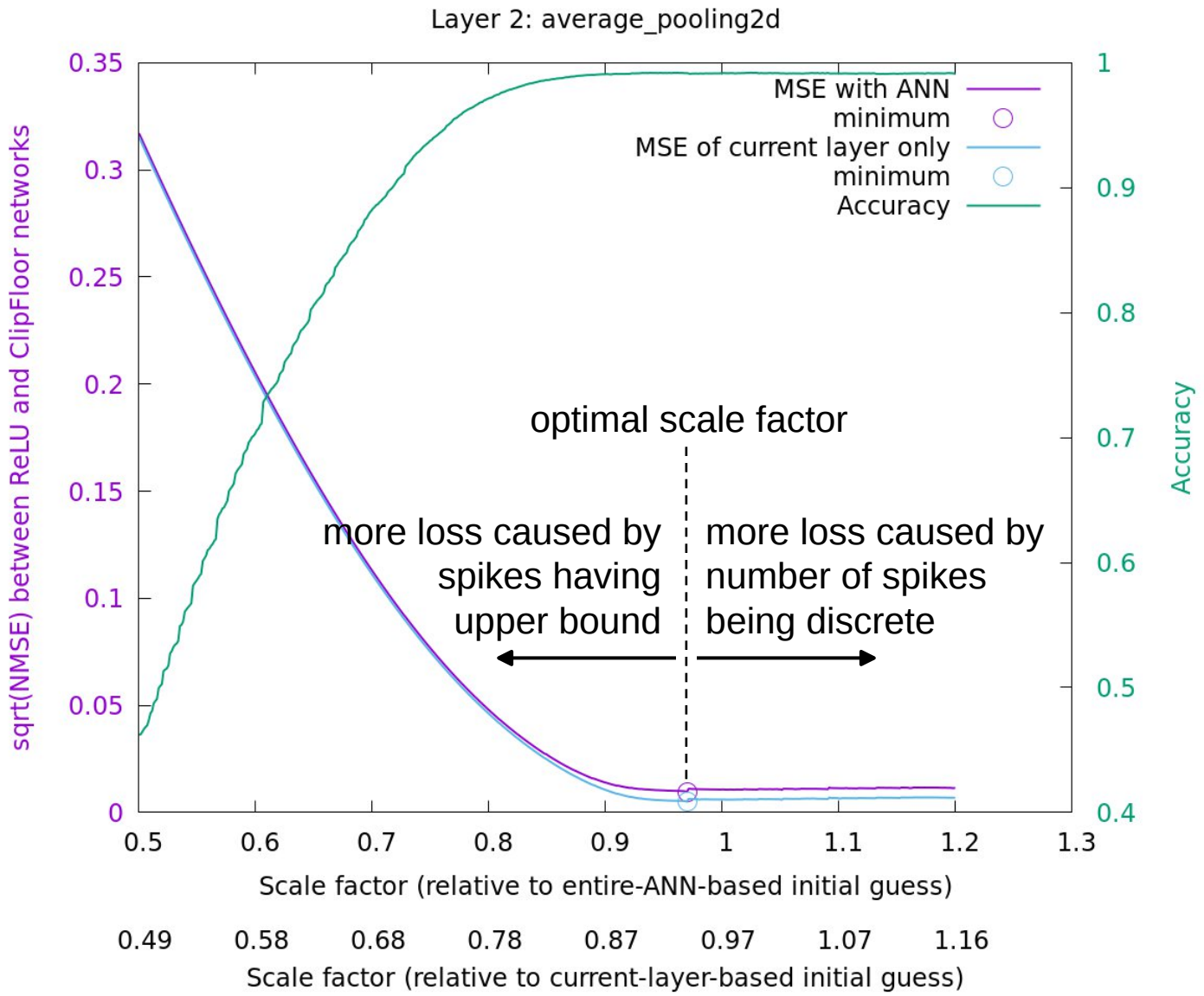
Finding optimal thresholds of spiking neurons: loss metrics to minimize



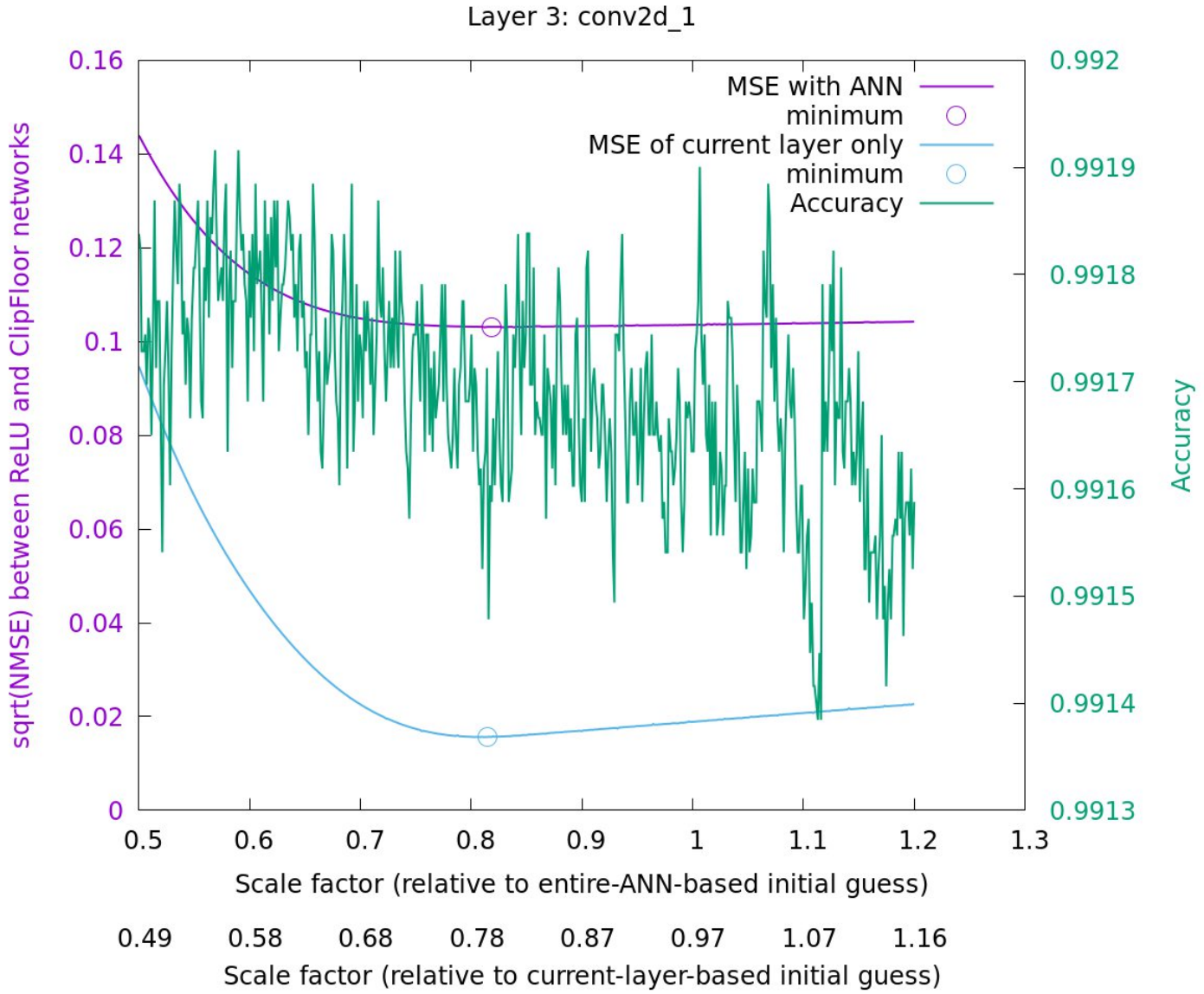
Finding optimal thresholds of spiking neurons: dependence of loss on the scale factor



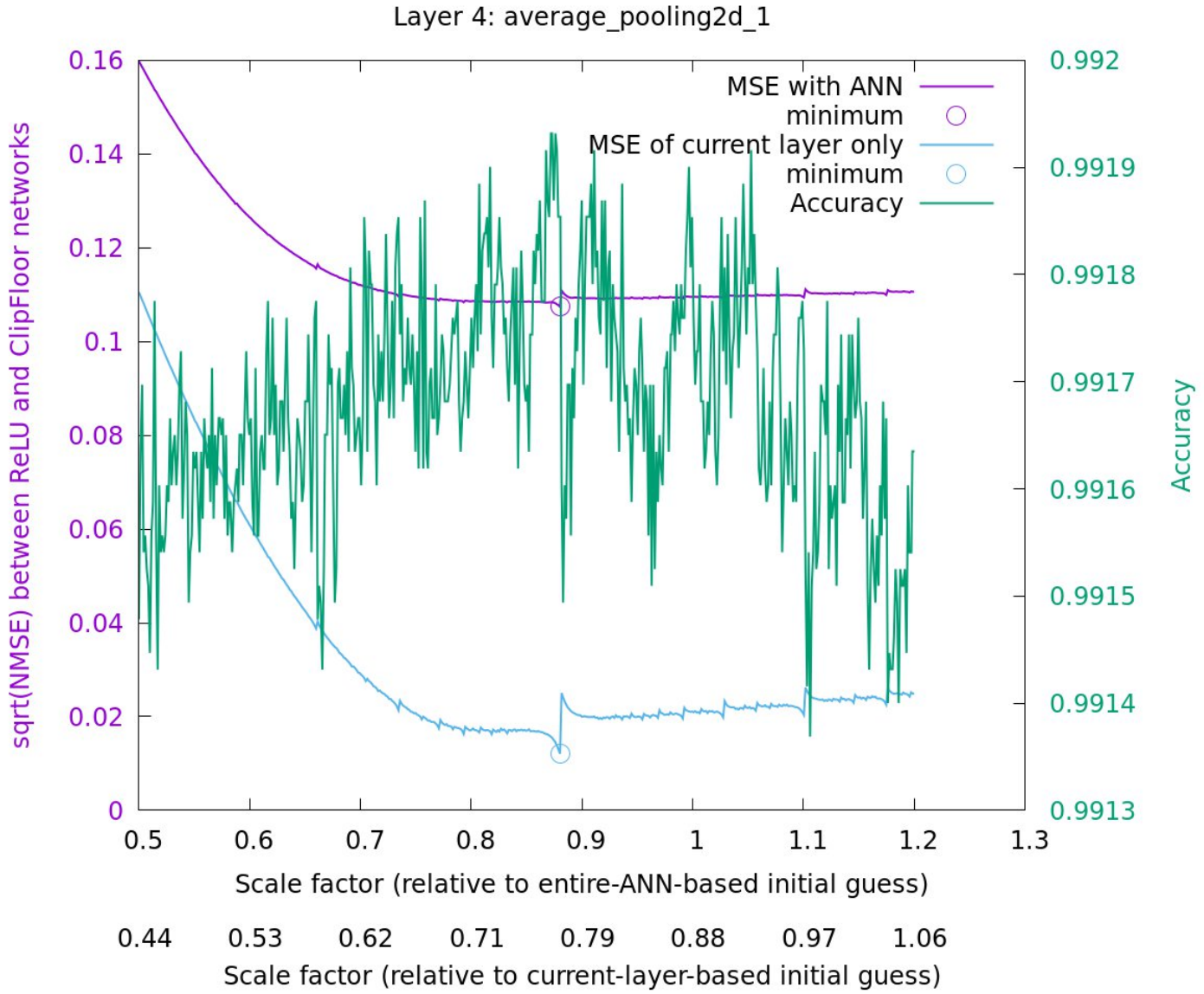
Finding optimal thresholds of spiking neurons: dependence of loss on the scale factor



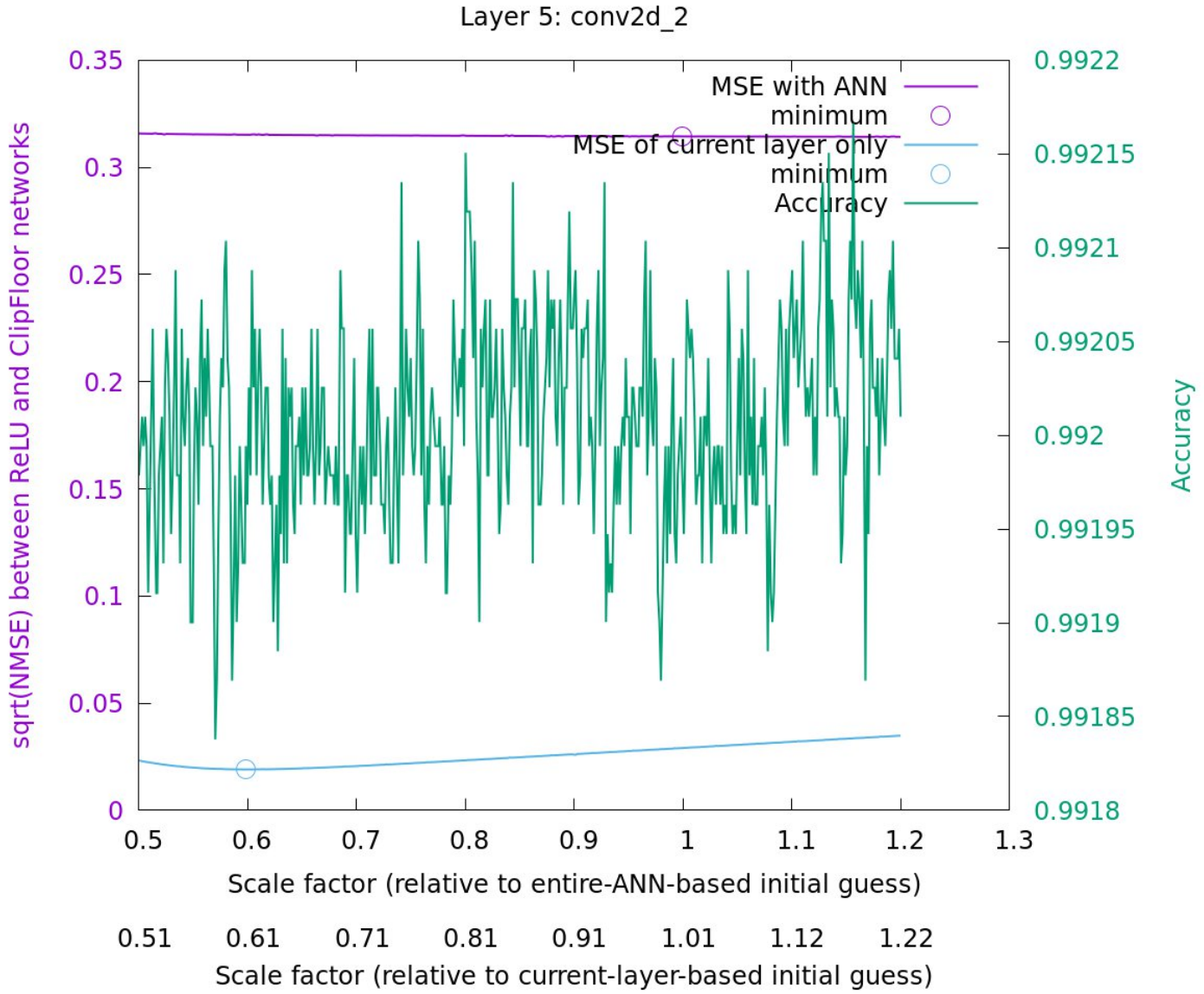
Finding optimal thresholds of spiking neurons: dependence of loss on the scale factor



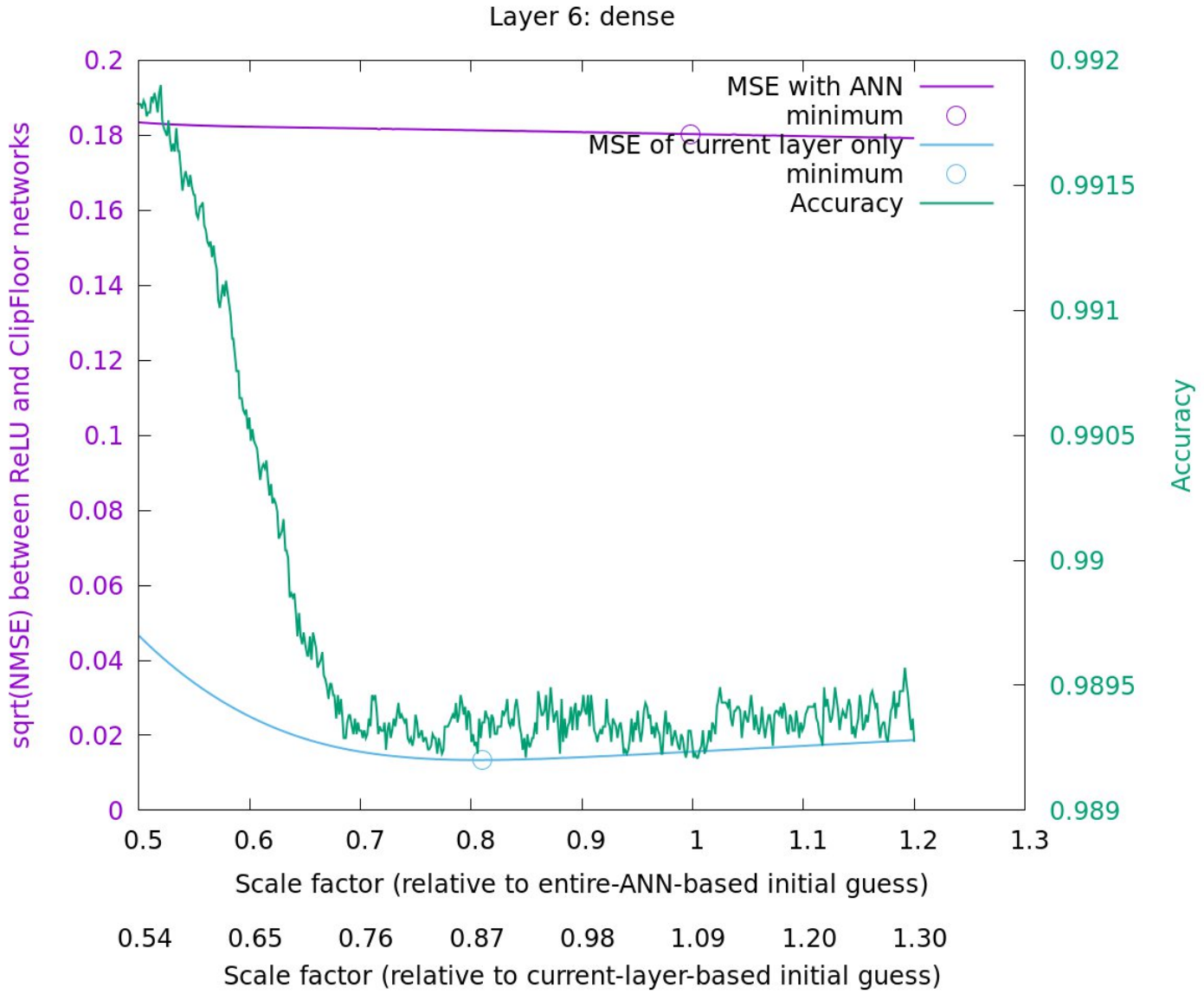
Finding optimal thresholds of spiking neurons: dependence of loss on the scale factor



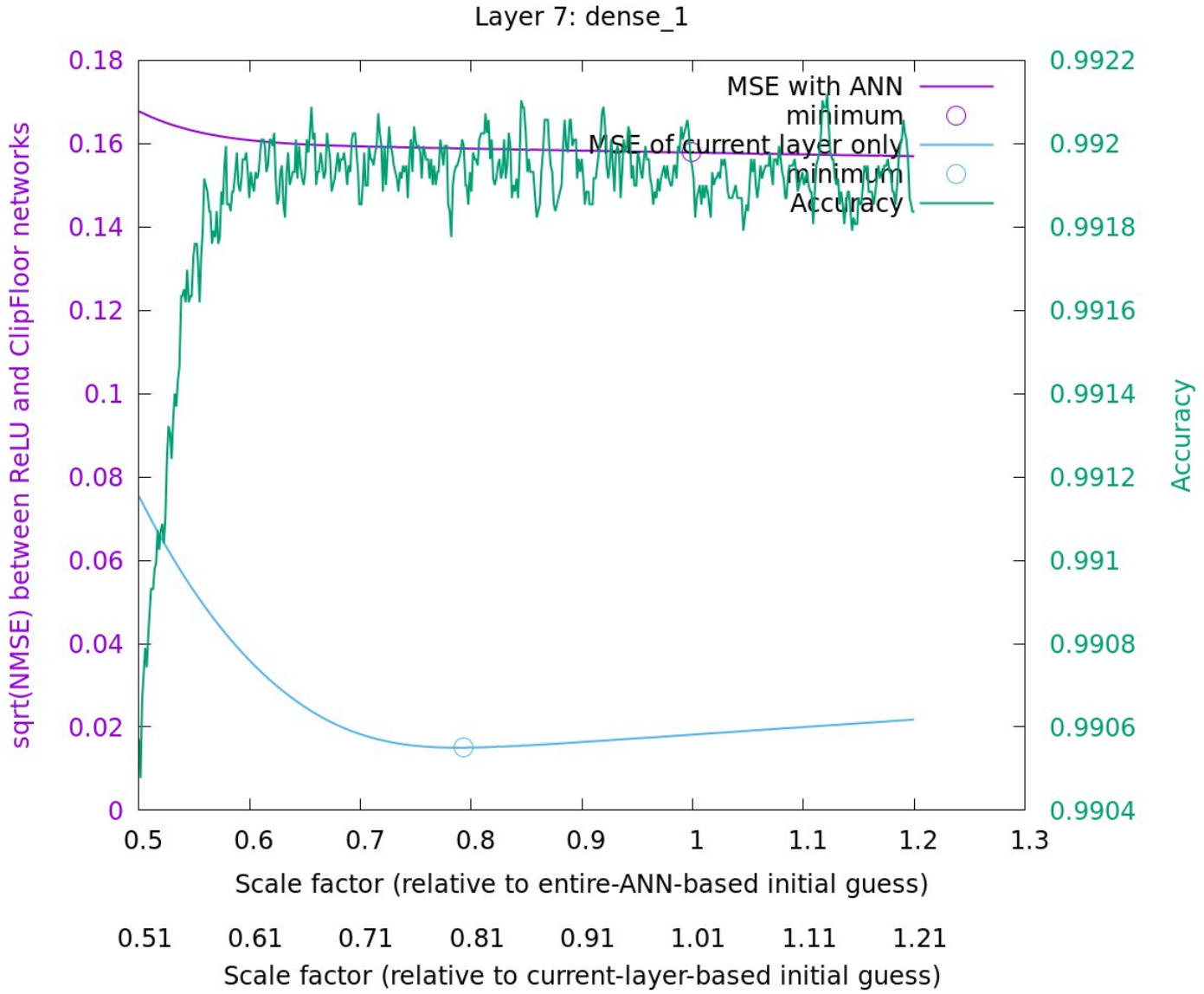
Finding optimal thresholds of spiking neurons: dependence of loss on the scale factor



Finding optimal thresholds of spiking neurons: dependence of loss on the scale factor



Finding optimal thresholds of spiking neurons: dependence of loss on the scale factor



Resulting accuracies

Network	F1-micro
Originally reported by the dataset authors	0.99
Conventional 2dCNN with float32 weights	0.99
Conventional 2dCNN with int8 weights	0.98
2dCNN with ClipFloor activation with T = 200 time steps	0.98
2dCNN with ClipFloor activation with T = 100 time steps	0.92
2dCNN with ClipFloor activation with T = 50 time steps	0.77

Conclusion

- The accuracy is 98%, with less than 1% loss relative to the original network,
  - with 8-bit synaptic weights,
  - with audio represented by Mel-Frequency Cepstral Coefficients,
  - and then encoded by spike rates 200 time steps per input audio (which allows real-time processing),
  - with the number of weights and neurons in the network allowing deployment to off-the-shelf neuroprocessors such as TrueNorth or AltAI.
- Optimal thresholds of spiking neurons can be obtained by finding minimal MSE between outputs of layers with ReLU and ClipFloor activation functions.