

Brief Report

Not peer-reviewed version

Noise as an Optimization Tool for Spiking Neural Networks

[Yana Garipova](#)^{*}, [Shogo Yonekura](#), [Yasuo Kuniyoshi](#)

Posted Date: 9 August 2024

doi: 10.20944/preprints202408.0704.v1

Keywords: spiking neural networks; adaptability; noise



Preprints.org is a free multidiscipline platform providing preprint service that is dedicated to making early versions of research outputs permanently available and citable. Preprints posted at Preprints.org appear in Web of Science, Crossref, Google Scholar, Scilit, Europe PMC.

Copyright: This is an open access article distributed under the Creative Commons Attribution License which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Brief Report

Noise as an Optimization Tool for Spiking Neural Networks

Yana Garipova *, Shogo Yonekura and Yasuo Kuniyoshi

Intelligent Systems Informatics Laboratory, University of Tokyo, Tokyo, 113-0033, Japan; <https://www.isi.imi.i.u-tokyo.ac.jp/>

* Correspondence: garipova@isi.imi.i.u-tokyo.ac.jp;

Abstract: Standard ANNs lack flexibility when handling corrupted input due to their fixed structure. Spiking Neural Networks (SNNs) can utilize biological temporal coding features, such as noise-induced stochastic resonance and dynamical synapses to increase a model's performance when its parameters are not optimized for a given input. Using the analog XOR task as a simplified Convolutional Neural Network analog, this paper demonstrates two key results: (1) SNNs solve the problem linearly inseparable in ANN with over 10% improvement in the optimal setting, and (2) in SNNs, the synaptic noise and dynamical synapses compensate for non-optimal parameters, achieving near-optimal results.

Keywords: spiking neural networks; adaptability; noise

1. Introduction

Spiking Neural Networks (SNNs) have been shown to outperform conventional static Artificial Neural Networks (ANNs) [13] in energy efficiency when running models on Neuromorphic chips [3,17], ability to self-optimize to changing probability distributions, such as the likelihood of a reward [11], and adaptability of a dynamical motor control system to a changing environment [5]. This paper shows that noise sensitivity in SNNs drives changes in neural synchronization, leading to surprisingly better performance than ANNs when parameters are not fully optimized.

Unlike ANNs, training the synaptic weights of SNNs is challenging due to the instantaneous spiking activity and dynamic synaptic characteristics that hinder gradient computation. Several methods exist to achieve trained SNNs: (1) obtain synaptic weights by ANN-backpropagation and apply them to SNN [16], (2) SNN-backpropagation [10], (3) hybrid learning of STDP-based pre-training and fine-tuning [9], and (4) gradient-free training method for SNN [14].

Aiming to compare the adaptability of ANN and SNN models, this paper is based on method (1): we create two similarly structured networks designed to run the analog XOR function. The network structure diagrams are shown in (Figure 1) of the materials and methods section. The goal of the analog XOR task is to check whether the inputs are within a certain range of each other; if they are, then the output is low; otherwise, it is high. The analog XOR task is a simple way to analyze networks' internal processes that can be generalized to Convolutional Neural Networks, audio processing, and sensor systems.

Differently from the pioneering network comparison/translation research [3,16], our networks have a multiplicative noise component to leverage temporal coding features of spiking neurons. The interest in creating noisy spiking models originates from the conclusions that noise influences the synchronization of inputs and inhibition, and synchronous input is more strongly propagated [6,7].

Another difference from the weight translation method (1) [16], is the use of dynamical synapses (another temporal coding feature) in our SNN. Dynamical synapses maintain the output frequency and defined thresholds comparable while applying noise. While this setup gives us an opportunity to study noise sensitivity in ANN and SNNs, it disrupts the ANN activations-to-SNN firing rates alignment and requires additional input and weight gain parameters to compensate for the synaptic time constant, τ_{syn} (a dynamical synapse parameter). While parameter tuning in dynamical synapse models poses challenges for scalability, the inherent robustness of our spiking network mitigates the

need for extensive optimization. The intrinsic adaptability of our spiking network allows it to maintain functionality even with suboptimal parameter configurations.

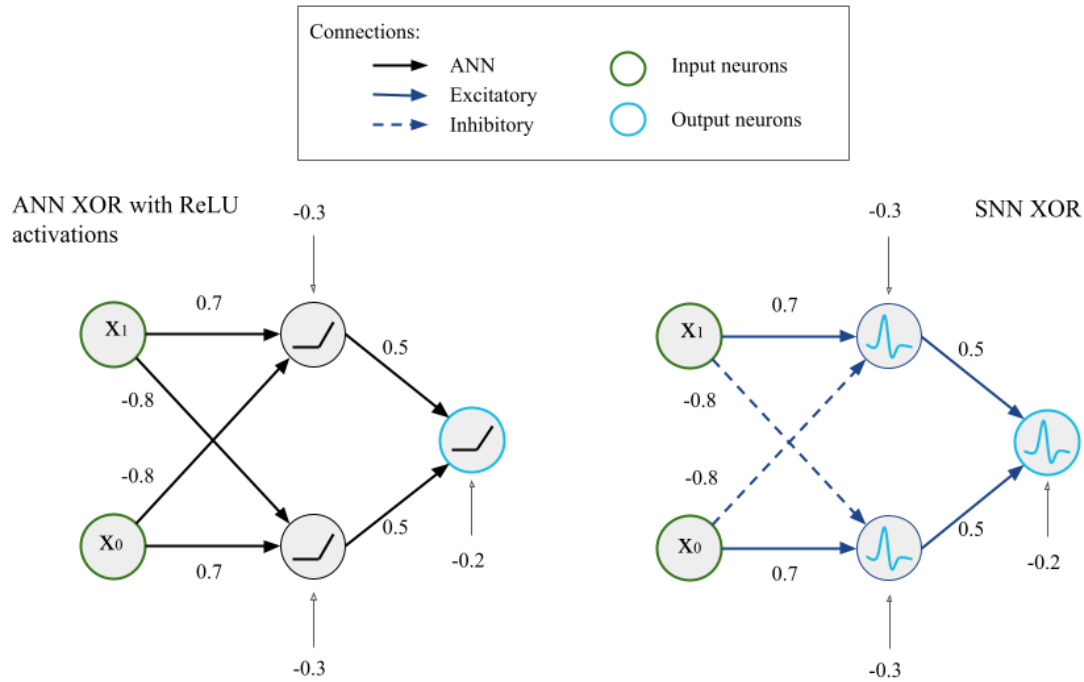


Figure 1. a (left): ANN XOR circuit and **b (right):** SNN XOR diagrams. Each circuit has two input neurons, two hidden layer neurons, and one output neuron. The circuits' input and weights are scaled.

2. Materials and Methods

Consider the analog XOR task. Given two inputs $(x_0, x_1) \in [0.00, 1.00]$, the output neuron will exceed the firing threshold h , when exactly one of the input neurons is greater or equal to a boundary value b . The output is normalized with $h \in (0.00, 1.00)$. The accuracy of a circuit is determined post-normalization assuming a decision boundary pair (b, h) .

For example, assume we have a boundary value of $b = 0.42$. If both inputs are smaller (or larger) than 0.42, then the output should be low. If only one input is less than 0.42, then the output should be high. The value that is considered "high" enough is called h and is decided upon normalized output. With (b, h) selected we observe how the circuit performs against it with noise added and parameters skewed.

The spiking circuit is simulated with NEST 2.20.1 [8], consisting of Leaky Integrate-and-Fire *iaf_psc_exp* neurons with dynamical synapses [19] and parameters listed in Table 1.

The neurons' membrane potential is defined by:

$$\tau_m \frac{dV_m}{dt} = -V_m(t) + I_{bias} + I(t) + I_{noise}(t), \quad (1)$$

where:

- I_{bias} is the base current (18 pA) plus the biases set by the network architecture
- $I(t)$ is the postsynaptic input current
- The synaptic noise current $I_{noise}(t)$ is given by the normal distribution with 0 mean and standard deviation D :

$$I_{noise}(t) = \frac{1}{D\sqrt{2\pi}} * e^{-\frac{1}{2}(\frac{x}{D})^2} * I(t). \quad (2)$$

Table 1. Neuron and simulation parameters for *iaf_psc_exp* used in the experiment.

Variable	Description, Units	Value
C_m	Membrane capacitance, pF	10.0
tau_m	Membrane time constant, ms	10.0
E_L	Resting membrane potential, mV	0.0
V_m	Initial membrane potential, mV	0.0
V_reset	Reset potential of the membrane, mV	16.0
V_min	Lowest value for the membrane potential	-1.798e+308
refractory_input	Discard input during refractory period	False
I_e	Constant input current, pA	18. + bias
V_th	Spike threshold, mV	20.
tau_fac	Synapse facilitation time constant, ms	1
tau_rec	Synapse recovery time constant, ms	10
tau_syn	Synaptic time constant (τ_{syn}), ms	1
-	Simulation resolution, ms	0.1
-	Simulation time, s	20.

In this experiment, noise is applied to the input neurons only and is scaled by the input values to mimic synaptic multiplicative noise, i.e. accumulated voltages that do not generate a spike [7]. The noise component can be common (equal) between both inputs or independent (different noise components of the same distribution).

In the non-spiking circuit of Figure 3, each input is tested with 20000 different noise components drawn from the same distribution, and then all 20000 trials are averaged for a single output. The same operation repeats 20 times for each noise level D to have more samples for the average accuracy result. The spiking circuit is simulated for 20 seconds per input, with a new noise component added every 0.1 ms (NEST resolution setting), with a total of 200,000 noise components per input. This repeats 20 times for each noise level D. Since accuracy is measured on normalized data, the standard deviation of the average accuracy is low.

There are fundamental differences in noise application in both networks. In SNN, noise is continuously interacting with the circuit before the final answer is given, affecting synchronization and inhibition (Figure 4). In ANN, on the other hand, negative activations are ReLU-ignored, and the contribution of noise is smoothed over by averaging the output.

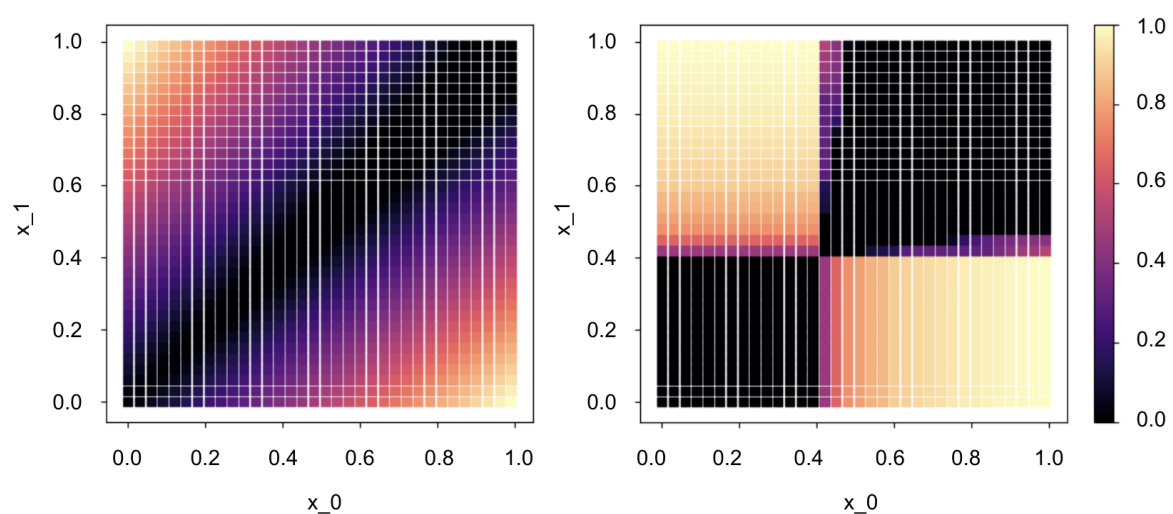


Figure 2. a (left): Output activation of the ANN XOR and **b (right):** output frequency of SNN XOR with $\tau_{syn}=20\text{ms}$, input gain=5, weight gain=17.

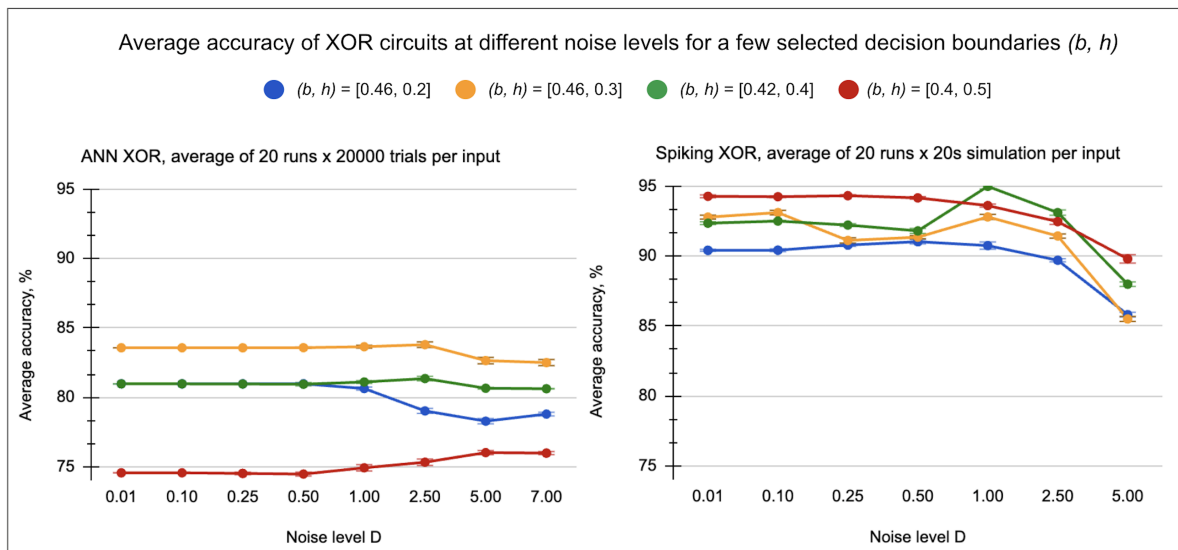


Figure 3. Accuracy percentage for selected "good" boundaries across different levels of common multiplicative noise. **a (left):** ANN XOR performance at different noise levels. **b (right):** The spiking circuit performance with "well-selected" gain parameters for $\tau_{syn}=1\text{ms}$: input gain=5, weight gain=110. ANN circuit is tested with the same gain parameters.

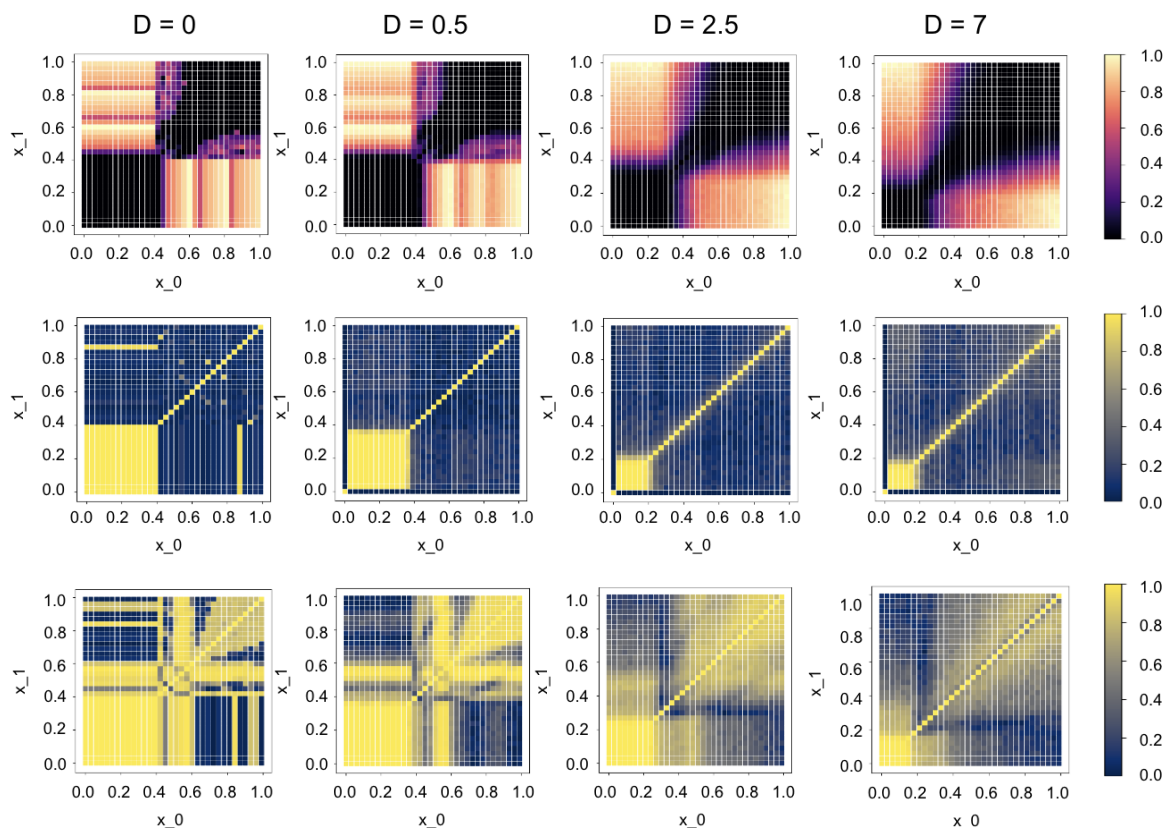


Figure 4. The spiking circuit behavior with "well-selected" parameters for $\tau_{syn}=1\text{ms}$: input gain=5, weight gain=110. **a (top row):** normalized firing frequencies output with different noise intensity D; **b (middle row):** Phase Locking Value (PLV) between the two input neurons; **c (bottom row):** PLV between the two hidden layer neurons.

3. Results

3.1. Firing Rates Output Example - Linear Separation with SNN

Note, that while the analog XOR problem is linearly inseparable for the ANN circuit, there is a set of parameters and a decision boundary (b, h) for which the spiking network produces a clear separation.

3.2. Comparing Models' Performance at Different Noise Levels

The non-spiking network in Figure 3 has the highest accuracy of 83.56% due to the linear inseparability; the highest accuracy for our current network parameters is achieved with the boundary input value of $b=0.46$, and firing frequency/activation threshold $h=0.3$. The spiking version of the circuit yields the following results (the first two are not relevant to noise): (1) 10% improvement for the same decision boundary due to an effect similar to Figure 2, (2) up to 20% improvement for the other non-optimal boundaries indicating non-equivalence gap, and (3) minor noise improvement up to 1-3% for some non-optimal boundaries in both networks.

3.3. Synchronization in Spiking Neurons at Different Noise Levels

3.4. the Independent Vs Common Noise in Spiking Circuit

Figure 5 shows other decision boundaries in the same "well-selected" setting ($\tau_{syn}=1\text{ms}$, input gain=5, weight gain=110). The independent noise gives slightly more improvement than the common noise at first, but the trend reverses for less optimal boundaries with high noise intensity. Overall, the noise boosts the performance to comparable levels for lower boundaries, making the network more sensitive to input as a consequence of stochastic resonance. A similar effect is observed in Figure 6 with a set of non-optimal gain parameters.

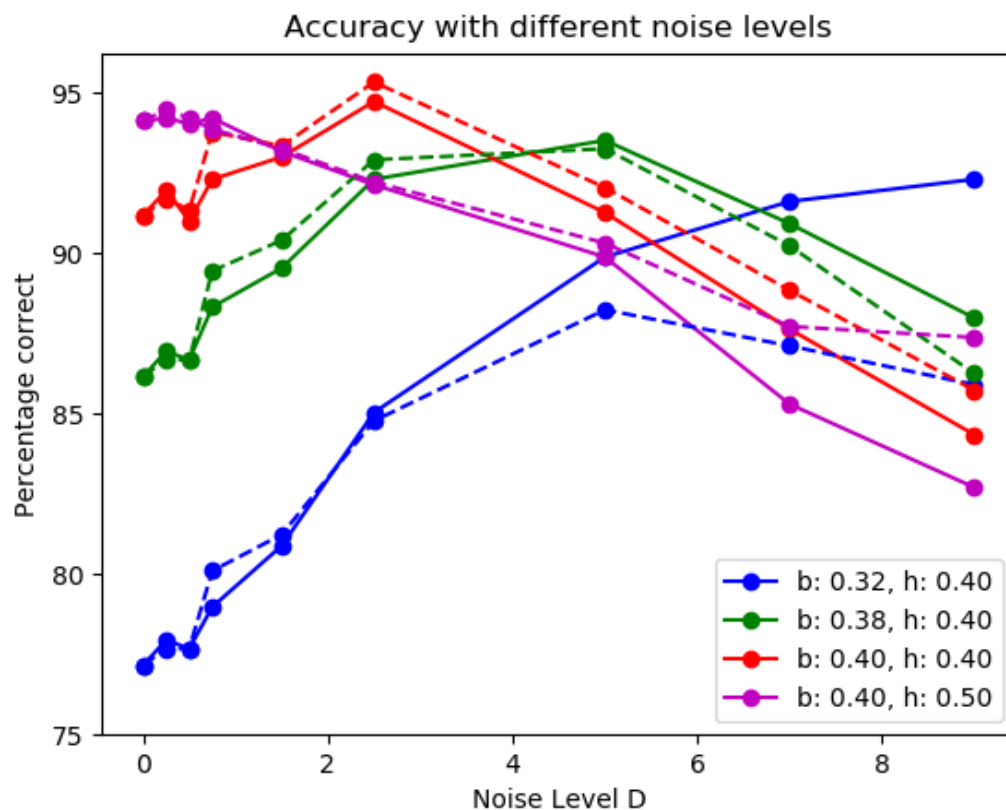


Figure 5. Accuracy percentage for some boundaries across different noise levels in spiking circuit with well-selected parameters for $\tau_{syn}=1\text{ms}$: input gain=5, weight gain=110. Solid line is common noise, dashed line is independent noise.

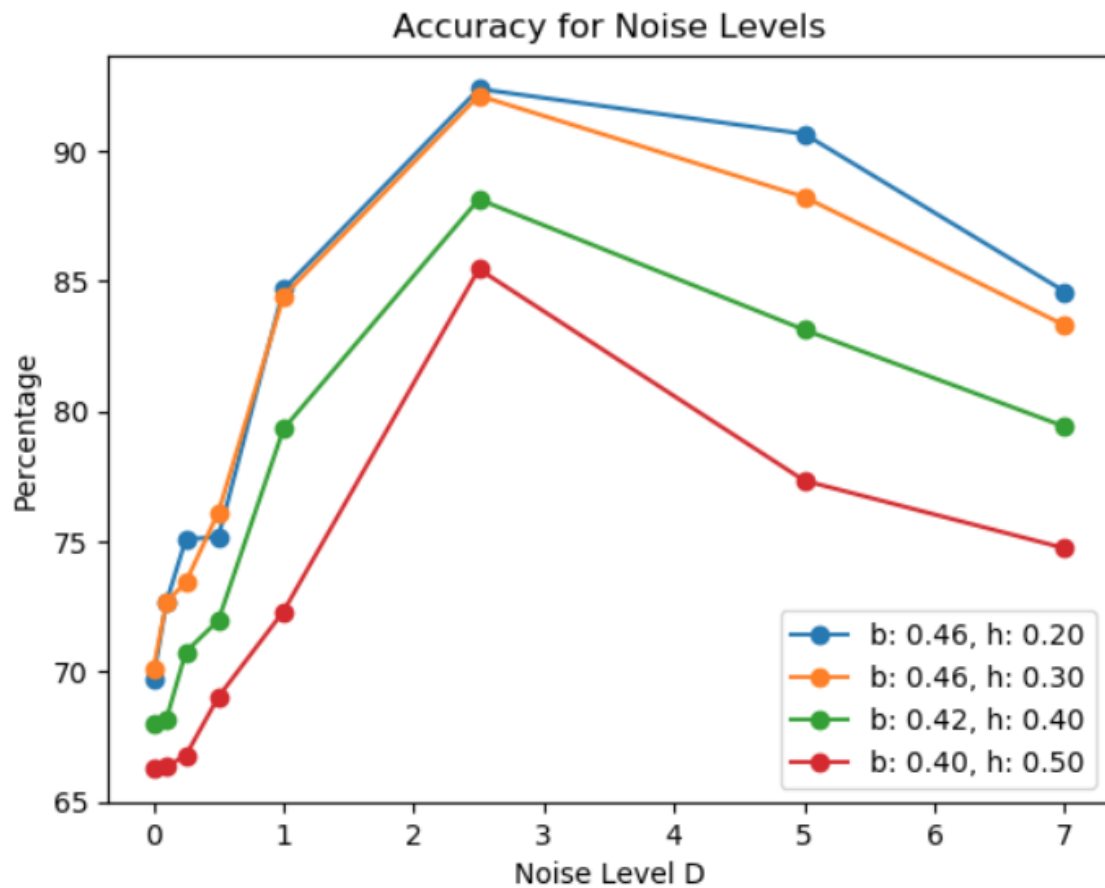


Figure 6. Accuracy for some boundaries across different common noise levels in SNN XOR with non-optimized gain parameters for $\tau_{syn}=1ms$: input gain=4, weight gain=105.

4. Discussion

The ability to adapt to changing or missing input is embedded in the brain's regime of stable propagation of synchronous neural activity among asynchronous global activity [2,6], a combination of rate and temporal coding [15]. The contributing physical properties to temporal coding are the inhibitory connections (feedback) [12], the presence of noise [18], and the short-term plasticity of dynamical synapses [19,20]. These properties are usually omitted in the standard ANN-to-SNN rate-coding-based conversion method [3,4,16].

This work implements temporal features in the XOR network by using synaptic noise and dynamical synapses. The synaptic noise has been shown to have amplifying power due to stochastic resonance, having a stronger effect when the network is not optimally tuned. The short-term plasticity of dynamical synapses has a role of temporal filtering of the firing rates [20], thus allowing the decision boundaries comparable between different noise levels. However, it requires adjusting the gain parameters depending on a selected synaptic time constant and corrupts the equivalency of the ANN and SNN models. Other ways to implement temporal features could be the addition of inhibitory neuron pools [1] or taking the synchronization index into account similar to [15].

While ANNs are fixed to the data they are trained on, SNNs can use noise and other perturbations to improve their performance. The XOR circuit example shows that SNNs can solve problems that ANNs cannot, and they can do so even when their parameters are not optimally tuned.

Author Contributions: Conceptualization, Y.G., S.Y. and Y.K.; methodology, Y.G.; software, Y.G. and S.Y.; validation, S.Y. and Y.K.; resources, Y.G. and S.Y.; writing—original draft preparation, Y.G.; writing—review and editing, Y.G. and S.Y.; supervision, S.Y. and Y.K. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding

Informed Consent Statement: Not applicable

Data Availability Statement: The raw data supporting the conclusions of this article will be made available by the authors on request.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. M. Abeles, G. Hayon, and D. Lehmann. Modeling compositionality by dynamic binding of synfire chains. *J. Comput. Neurosci.*, 17:179–201, 2004.
2. Y. Aviel, D. Horn, and M. Abeles. Synfire waves in small balanced networks. *Neurocomputing*, 58-60:123–127, 2004. Computational Neuroscience: Trends in Research.
3. Y. Cao, Y. Chen, and D. Khosla. Spiking deep convolutional neural networks for energy-efficient object recognition. *Int. J. Comput. Vis.*, 113:54–66, 2015.
4. P.U. Diehl, D. Neil, J. Binas, M. Cook, S.C. Liu, and M. Pfeiffer. Fast-classifying, high-accuracy spiking deep networks through weight and threshold balancing. In *IJCNN*, pages 1–8, 2015.
5. K. Favier, S. Yonekura, and Y. Kuniyoshi. Spiking neurons ensemble for movement generation in dynamically changing environments. *IEEE/RSJ*, pages 3789–3794, 2020.
6. P. Fries. A mechanism for cognitive dynamics: neuronal communication through neuronal coherence. *TiCS*, 9(10):474–480, 2005.
7. W. Gerstner and W.M. Kistler. Spiking neuron models: Single neurons, populations, plasticity, 2002.
8. M.O. Gewaltig and M. Diesmann. Nest (neural simulation tool). *Scholarpedia*, 2(4):1430, 2007.
9. C. Lee, P. Panda, G. Srinivasan, and Kaushik Roy. Training deep spiking convolutional neural networks with stdp-based unsupervised pre-training followed by supervised fine-tuning. *Front. Neurosci.*, 12, 2018.
10. J.H. Lee, T. Delbruck, and M. Pfeiffer. Training deep spiking neural networks using backpropagation. *Front. Neurosci.*, 10, 08 2016.
11. R. Legenstein and W. Maass. Ensembles of spiking neurons with noise support optimal probabilistic inference in a dynamically changing environment. *PLoS Comput Biol*, 10(10:e1003859), 2014.
12. H. Liang, X. Gong, M. Chen, Y. Yan, W. Li, and C.D. Gilbert. Interactions between feedback and lateral connections in the primary visual cortex. *Proc Natl. Acad. Sci.*, 114(32):8637–8642, 2017.
13. J. McGonagle. Feedforward neural networks. *Brilliant.org*, 2024.
14. M. Nakajima, K. Inoue, and K. et al. Tanaka. Physical deep learning with biologically inspired training method: gradient-free approach for physical hardware. *Nat Commun*, 13(7847), 2022.
15. D.P. Reichert and T. Serre. Neuronal synchrony in complex-valued deep networks, 2013.
16. B. Rueckauer, I.A. Lungu, Y. Hu, M. Pfeiffer, and S.C. Liu. Conversion of continuous-valued deep networks to efficient event-driven networks for image classification. *Front. Neurosci.*, 11, 2017.
17. C. Stöckl and W. Maass. Optimized spiking neurons classify images with high accuracy through temporal coding with two spikes, 2021.
18. C. Sutherland, B. Doiron, and A. Longtin. Feedback-induced gain control in stochastic spiking networks. *Biol. Cybern.*, 100:475–489, 2009.
19. M. Tsodyks, A. Uziel, and H. Markram. Synchrony generation in recurrent networks with frequency-dependent synapses. *J. Neurosci.*, 20, 2000.
20. M. Tsodyks and S. Wu. Short-term synaptic plasticity. *Scholarpedia*, 8(10):3153, 2013. revision #182521.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.