# Evolving Spiking Circuit Motifs Using Weight Agnostic Neural Networks

**Abrar Anwar**
Department of Computer Science, The University of Texas at Austin
Sandia National Laboratories
abraranwar@utexas.edu

### Abstract

Neural architecture search (NAS) has emerged as an algorithmic method of developing neural network architectures. Weight Agnostic Neural Networks (WANNs) are an evolutionary-based NAS approach. Fundamentally, WANNs find network structures that are relatively insensitive to shifts in weight values and are typically much smaller than an equivalent performance dense network. Here, we extend the WANN framework to search for spiking circuits and in doing so investigate whether these circuit motifs can also yield task performance that is weight agnostic. We analyze properties such as the complexity of the solution, as well as performance. Our results successfully show the performance of spiking WANNs on several exemplar tasks.

## Introduction

Neural networks are becoming exceedingly commonplace, with applications in various domains; however, limitations of traditional hardware which neural networks run on are becoming apparent, specifically in the low-power domain. For edge computing applications, such as drones and satellites, running large neural networks are not feasible due to the energy cost.

Neuromorphic, or brain-inspired, computing introduces a new paradigm that offers low energy usage. This non-Von Neumann architecture relies on event-based spiking communication between neurons. Conversely, a typical artificial neural network (ANN) relies on dense communication of continuous values. In order for ANNs to work under this new paradigm, they must be converted into a spiking neural network (SNN). An SNN mimics biological neural networks by incorporating time into the neuron model by using discrete spikes to transfer information between neurons. The primary motivation for this difference is the promise of energy-efficient compute evidenced by biological systems.

In terms of an ANN, we can define a spiking neuron computation by a threshold (step/binary) activation function. Although an ANN with threshold activation functions is not strictly an SNN due to the lack of a temporal domain, they are compatible with spiking neuromorphic hardware, therefore it is referred to as spiking. Whetstone (Severa et al. 2019), an algorithm that iteratively sharpens activation functions in deep neural networks to become binary, shows that converting ANNs to SNNs is a non-trivial process. We show that evolving weight agnostic neural networks with threshold and linear activation functions perform well and are be suitable to be transferred onto neuromorphic hardware.

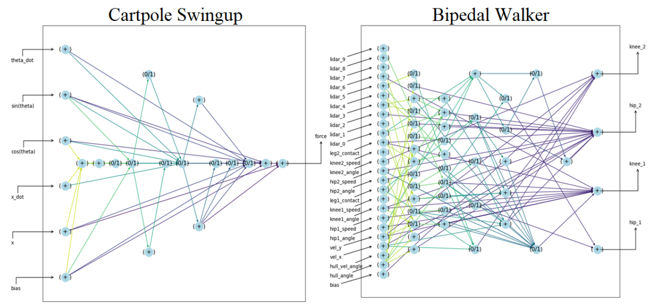


Figure 1: Sparse network topologies of the top individuals for the swingup cartpole task (left) and the bipedal walker task (right). (+) represents the linear activation functions and (0/1) is the threshold activation function.

## Weight Agnostic Neural Networks

Weight Agnostic Neural Networks (WANNs) (Gaier and Ha 2019) are inspired by the fact precocial species can carry out several tasks at birth, without any training. WANNs follow an iterative topology search algorithm inspired by the NEAT evolutionary search method (Stanley and Miikkulainen 2002). Using multi-objective optimization, the size of the network is minimized and the performance is maximized by adding and removing connections to various nodes that are sampled from a large set of activation functions. This process generates a population of sparse network topologies.

Most architecture search methods involve individual weight training for each generated topology, causing the searching process to be computationally expensive. WANNs enforce weight sharing across the entire network, so rather than training a network several times, WANNs evaluate their performance on a set of shared weight values between -2 and 2. The optimal network topology through this search method then has its weights individually trained. Gaier and Ha speculated the variety of activation functions was important to

| WANN | Tuned Shared Weight | Tuned Weights | Connections in Network |
|---|---|---|---|
| Swingup Cartpole | $723 \pm 16$ | $932 \pm 6$ | 62 |
| Bipedal Walker | $261 \pm 58$ | $322 \pm 7$ | 338 |
| MNIST | 91.9% | 94.2% | 1228 |
| **Spiking WANN** | **Tuned Shared Weight** | **Tuned Weights** | **Connections in Network** |
| Swingup Cartpole | $745 \pm 11$ | $912 \pm 5$ | 56 |
| Bipedal Walker | $290 \pm 22$ | $281 \pm 31$ | 210 |
| MNIST | 87.7% | 88.2% | 576 |

Table 1: Results for the various tasks for the WANN (top) and Spiking WANN (bottom). The first two results show average reward over 100 rollouts with standard deviations. For MNIST, accuracy on the test set is reported.

their result. We show that threshold and linear activations are sufficient.

## Contribution

To generate networks compatible with neuromorphic hardware, the set of activation functions are restricted to threshold and linear. Threshold activation functions themselves can easily be transferred onto neuromorphic hardware; however, when combined with a linear activation function, they mimic additive dendritic trees and can be approximated by leaky integrate-and-fire neurons with delays.

After devising this approach with the help of my mentors, I evaluated it on three tasks: the cartpole swingup task, the bipedal walker task, and MNIST digit classification. These tasks are the same as the WANN paper and are evolved using the same parameters to ensure fair comparisons are made between them. WANNs perform slowly on high dimensional tasks due to mutating through a large number of connections from the input layer, thus the dimensions of MNIST were reduced from 28x28 to 16x16. Evaluation of these tasks are highly parallelizable, so I implemented asynchronous evaluation across hundreds of processes to speed up training.

Results can be seen in Table 1. The tuned shared weight category is the best shared weight value within the range for the evolved network topology. The tuned weights is when the network's weights are individually trained using evolutionary strategies. Interestingly, the tuned shared weights for the spiking WANNs have generally higher performance than the WANN, but the finetuned weights perform worse. Experimentation has shown this can be attributed to a fewer number of weights to finetune, as we see spiking WANNs consistently generate smaller networks, seen in Figure 1.

We note, however, that in the control tasks the agent interacts with the environment and may be able to adjust behavior during an episode. In contrast, classification is a one-shot determination. Table 1 shows the performance to be worse across the board for the classification task; however, this is related to the significantly smaller sized network generated by the spiking WANN. Changes to the multi-objective optimization problem may mitigate this and help find a more optimal network topology.

## Future Work

We hope to map these, or similar networks, to neuromorphic hardware. Some inputs and outputs may not be fully spiking, such as the softmax operation used for classification tasks. Developments in these methods will aid in eventual neuromorphic deployment. In addition, the size of the network was minimized as a rough approximation of energy usage and complexity, but architectures could perform differently depending on the type of network topology. Exploring energy-based constraints by changing this metric depending on the target platform would allow for neural network-hardware co-design. For example, certain neuromorphic platforms restrict the number of connections but have a large number of neurons, and vice versa.

Whetstone refines the activation functions of a typical deep neural network to become threshold activation functions, which can then be used on neuromorphic hardware. Leveraging the representational capabilities of a Whetstone network with a spiking WANN may increase performance on tasks with large input sizes, such as Atari game playing.

Lastly, I have been evaluating the robustness of spiking WANNs to noise. Since the evolved network topology is weight agnostic, there is potential for noise resilience in the input space or in the synaptic weights. Future exploration of this domain would make it an ideal candidate for generating networks on hardware where the weights are noisy or low precision.

## Acknowledgments

This work was supported by DOE NA-22 funding at Sandia National Laboratories. Sandia National Laboratories is a multimission laboratory managed and operated by National Technology and Engineering Solutions of Sandia, LLC., a wholly owned subsidiary of Honeywell International, Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

## References

Gaier, A.; and Ha, D. 2019. Weight Agnostic Neural Networks. In *Advances in Neural Information Processing Systems 32*, 5364–5378.

Severa, W.; Vineyard, C. M.; Dellana, R.; Verzi, S. J.; and Aimone, J. B. 2019. Training deep neural networks for binary communication with the Whetstone method. *Nature Machine Intelligence* 1(2): 86–94.

Stanley, K. O.; and Miikkulainen, R. 2002. Evolving Neural Networks Through Augmenting Topologies. *Evolutionary Computation* 10(2): 99–127.