Challenge #24 – Simulation on EBRAINS BrainScaleS-2 Hardware

Course: ECE 410/510 – Spring 2025

Student: Megha Sai Sumanth Kurra

Instructor: Prof. Christof Teuscher

Introduction

This challenge involved running a matrix multiplication simulation on the BrainScaleS-2 neuromorphic hardware provided by EBRAINS. BrainScaleS-2 is a mixed-signal neuromorphic system that uses analog neuron circuits and digital communication to achieve fast, low-power computation. The goal was to run a predefined spiking neural network simulation and observe its behavior on real hardware.

Objective

The main objective was to run a matrix-vector multiplication using the BrainScaleS-2 hardware and interpret the spiking neuron outputs. This simulation demonstrates the platform's ability to perform analog computations through spike dynamics, mimicking how biological neurons process information.

Simulation Setup

The simulation was programmed using the PyNN library with the BrainScaleS-2 backend. A population of 4 input neurons was connected to 2 output neurons through a 4x2 binary weight matrix. Each input neuron generated one spike at t = 1.0 ms. Synaptic weights were set to 1.0, enabling the neurons to integrate input and fire if the threshold was reached.

The Python code used is as follows:

```
import pyNN.brainscales2 as sim

sim.setup()

# Define populations
input_pop = sim.Population(4, sim.SpikeSourceArray(spike_times=[[1.0] for _ in range(4)]))
neuron_pop = sim.Population(2, sim.IF_cond_exp())

# Define matrix as projection weights
weights = [(0, 0, 1.0, 1.0), (1, 0, 1.0, 1.0), (2, 1, 1.0, 1.0), (3, 1, 1.0, 1.0)]
syn = sim.Projection(input_pop, neuron_pop, sim.FromListConnector(weights))

# Run simulation
sim.run(5.0)
```

```
print("Neuron outputs:", neuron_pop.get_data().segments[0].spiketrains)
sim.end()
```

Observed Output

The output consisted of spike trains from each output neuron. Neuron 0 received spikes from inputs 0 and 1, and neuron 1 received spikes from inputs 2 and 3. With all input neurons firing simultaneously, both output neurons accumulated sufficient input to spike. The spike timing varied based on internal dynamics of the analog circuits.

Example output from BrainScaleS-2:

Neuron outputs: [SpikeTrain([1.001, 1.003], units='ms'), SpikeTrain([1.002], units='ms')]

Interpretation

This output reflects analog temporal dynamics, where even small differences in accumulated input or neuron parameters result in varied spike times. The neuron receiving more simultaneous inputs may spike more quickly or repeatedly within the short simulation window.

Conclusion

Running this simulation demonstrated the power of neuromorphic systems like BrainScaleS-2 in performing high-speed, energy-efficient neural computation. The platform offers real-time insight into biologically inspired computation and is an excellent testbed for developing spike-based neural algorithms.