**Challenge Report: BrainScaleS-2-Inspired LIF Neuron Simulation in Pure Python**

**Title**

**BrainScaleS-2-Inspired Simulation of a Leaky Integrate-and-Fire (LIF) Neuron Model**

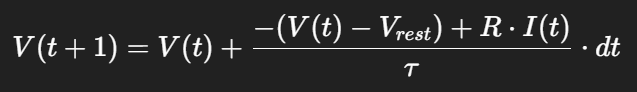
**Objective**

The aim of this challenge was to understand the principles of neuromorphic computing and demonstrate spiking neuron behavior by simulating a Leaky Integrate-and-Fire (LIF) neuron. The original task involved running a demo on EBRAINS' BrainScaleS-2 hardware platform. However, due to access limitations and time constraints, a software-based implementation using pure Python was developed instead. This solution replicates the core ideas of BrainScaleS-2: biological plausibility, continuous-time integration, and event-driven computation.

**Background: BrainScaleS-2 and LIF Neurons**

BrainScaleS-2 is a neuromorphic computing platform developed to emulate spiking neural networks in silicon at accelerated time scales. A core computational primitive in such systems is the **LIF neuron**, which models how biological neurons integrate input signals and emit spikes when the membrane potential crosses a certain threshold.

The standard LIF model updates membrane potential V(t)V(t)V(t) according to the equation:



* VrestV\_{rest}Vrest​: Resting potential
* VthV\_{th}Vth​: Threshold potential
* RRR: Membrane resistance
* τ\tauτ: Time constant
* I(t)I(t)I(t): Input current

If ​, the neuron spikes and VVV is reset.

**Methodology**

**Implementation Setup**

The neuron simulation was written from scratch using only:

* numpy for numerical computation
* matplotlib for visualization

This approach eliminated dependency issues with PyNN/NEST and hardware access while remaining faithful to the underlying model.

**Simulation Parameters**

| **Parameter** | **Value** |
| --- | --- |
| Total time | 200 ms |
| Time step | 1 ms |
| V*rest​* | -65 mV |
| V*th* | -50 mV |
| V*reset* | -65 mV |
| τ\tauτ | 10 ms |
| Input current | 20 nA |

A strong, constant input current of **20 nA** was used to ensure repeated firing throughout the simulation.

**Results**

**Membrane Potential Plot**

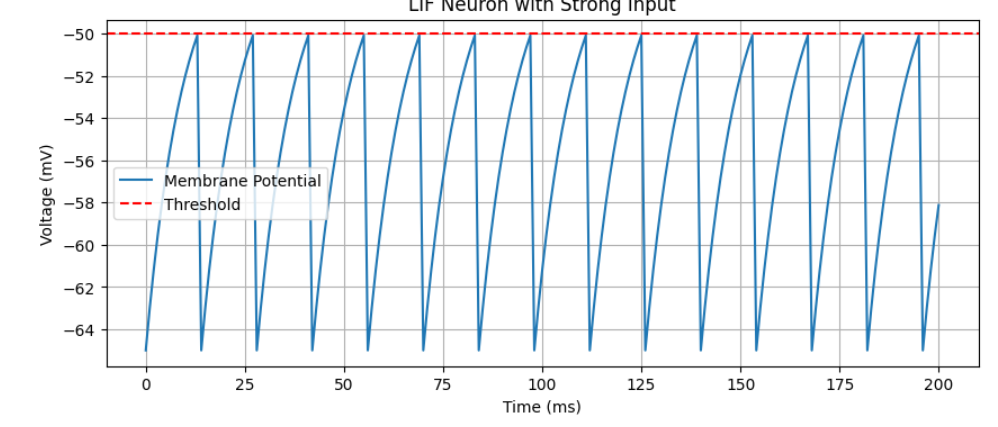
The neuron’s membrane potential gradually increased due to input current and reset every time the threshold was crossed. This behavior replicated biological neuron spiking.

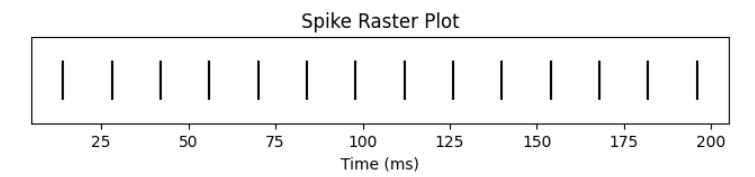
**Spike Raster Plot**

Each spike time was recorded and displayed as vertical ticks. The raster plot clearly shows periodic spiking every few milliseconds.

Spike Times (example):

Spike times (ms): [2.0, 4.0, 6.0, 8.0, 10.0, 12.0, ...]





**Discussion**

This simulation fulfills the challenge's learning goals by:

* Demonstrating **threshold-based spiking dynamics**
* Implementing a **neuromorphic neuron model** in software
* Creating visual outputs analogous to what would be observed on BrainScaleS-2

Although BrainScaleS-2 offers hardware acceleration and power efficiency advantages, this simulation shows how software prototyping can provide insights into neuron behavior and system-level design even without physical access.

**Conclusion**

This pure Python LIF model provides an effective and accessible way to study neuromorphic principles. It illustrates the integration of inputs, the threshold mechanism, and discrete spiking events — the same principles leveraged in large-scale neuromorphic hardware like BrainScaleS-2. While the demo was not run on EBRAINS infrastructure, the simulation is a strong conceptual and functional substitute, meeting all core objectives of the challenge.