Spiking neural networks as a model for Hydra nerve nets

Michael Ivanitsky, Connor Puritz

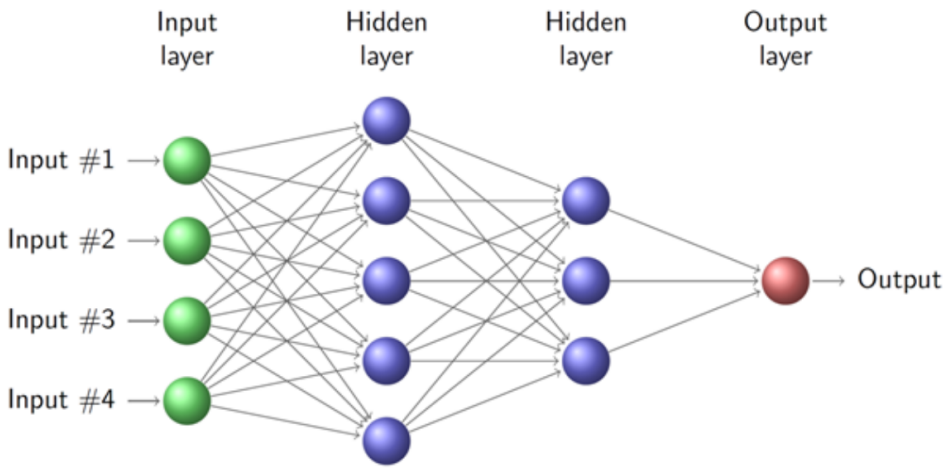
LSA COLLEGE OF LITERATURE, SCIENCE, AND THE ARTS
UNIVERSITY OF MICHIGAN

Math 463
December 5, 2018

Table of Contents

Artificial Neural Network (ANN) Introduction

- Neurons represented by vertices in a weighted directed graph
- Output of a neuron is a function of the weighted sum of inputs
- Usually have multiple hidden layers of neurons
- Learning accomplished through modifying edge weights according to some algorithm
- Originally inspired by biological nervous systems, but most implementations exchange biological realness for simplicity and computational efficiency

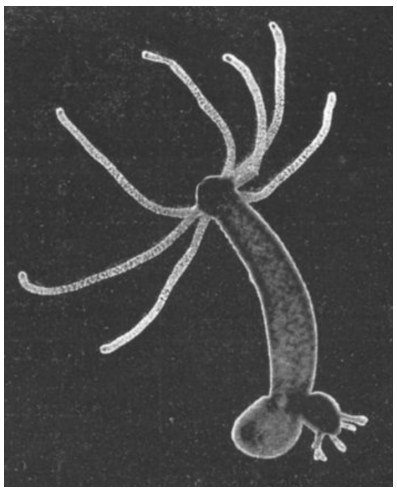


Spiking Neural Network (SNN) Introduction

- More closely resemble biological neural nets
- Fire not during a propagation cycle, but when a specified 'threshold membrane potential' crossed
- A neuron's state is modeled by a differential equation that depends on input strength and as well as timing
- More promise compared to other types of ANNs, but much more computationally intensive to both train and run
- It is thought a primary advantage of SNNs comes from their ability to encode information in the frequency/rate of pulses and not just in their intensity, as with traditional networks

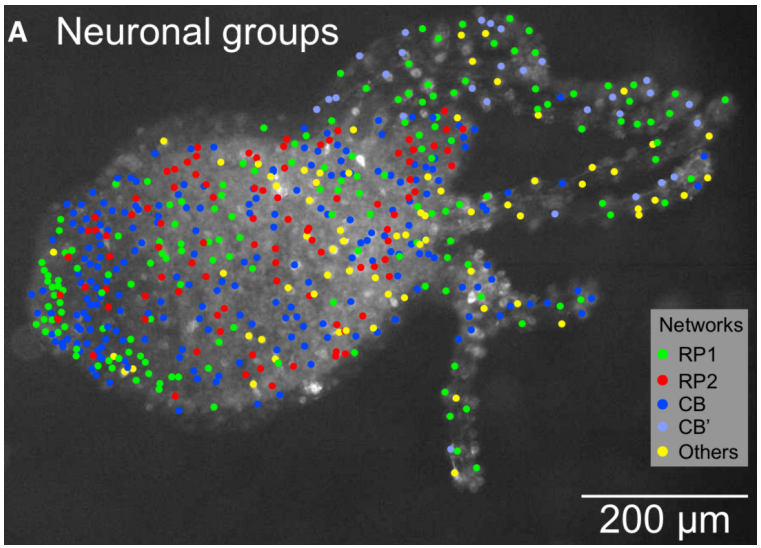
Evolutionary History of H. vulgaris

- *Hydra* are small, freshwater hydrozoans (family Cnidaria)
- Believed to have originated around 60 Mya
- Cnidarians first appeared around 580 Mya, haven't changed much since
- Very small repertoire of behaviors which is consistent across individuals and environments



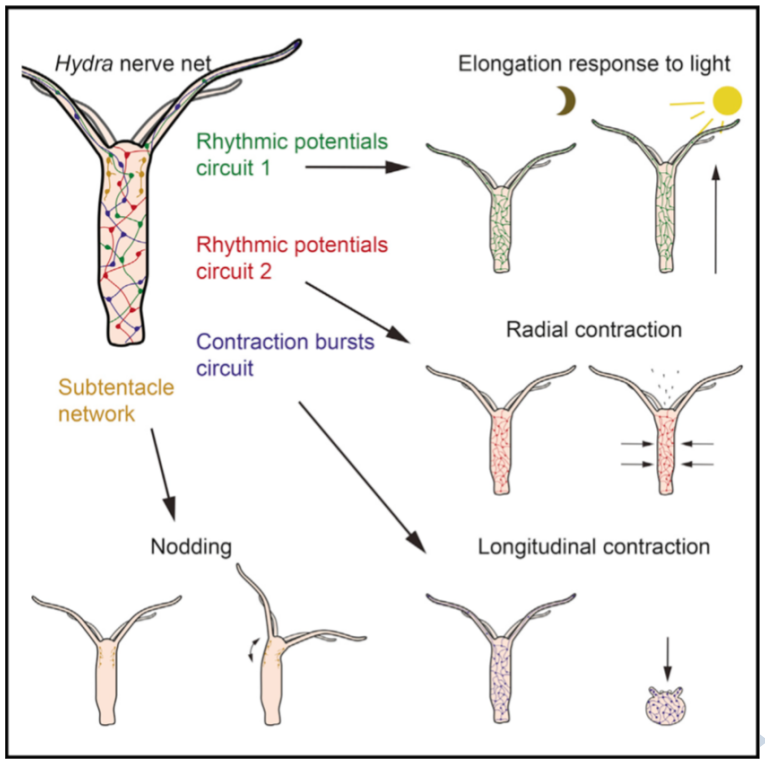
Nerve Nets

- *Hydra* have a diffuse nerve net rather than a CNS
- Comprised of ganglia and a few types of sensory neurons
- Once mature, a constant density gradient of neurons is maintained. Adults have no more than a few thousand neurons
- Turns out nerve net is actually composed of non-overlapping circuits that correspond to specific sets of behaviors



Separate Circuits

- RP1 associated with longitudinal elongations
- RP2 associated with radial contractions
- CB associated with longitudinal contractions
- STN associated with 'nodding' behavior
- RP1 and CB are antagonistic, no other interactions



Summary

Advantages of *Hydra* as a model organism:

- Very small size, so currents travel quickly and therefore the topology of the nerve net can be ignored at first approximation
- Distinct behaviors appear to be controlled by distinct circuits, so analysis can be broken into many parts
- Small amount of neurons and fixed neuronal density make simulation much easier

Neuron Model

- Leaky integrate-and-fire (LIF) model:

$$\frac{dV_m}{dt} = \frac{1}{C_m}\left(-\frac{(V_m - V_m^{eq})}{R_m} + I_{ext}\right)$$

- Computationally simpler than Hodgkin-Huxley
- Models neuron as RC circuit with leak term
- Doesn't explicitly specify spiking behavior or refractory period, but easy to implement using iterative ODE methods
- Possible implementation:

```
if V(t+1) > threshold:
   V(t)   <- spike
   V(t+1) <- hyperpolarize
if t in refractory period:
   V(t+1) <- hyperpolarize
```

Antagonistic Neural Circuits

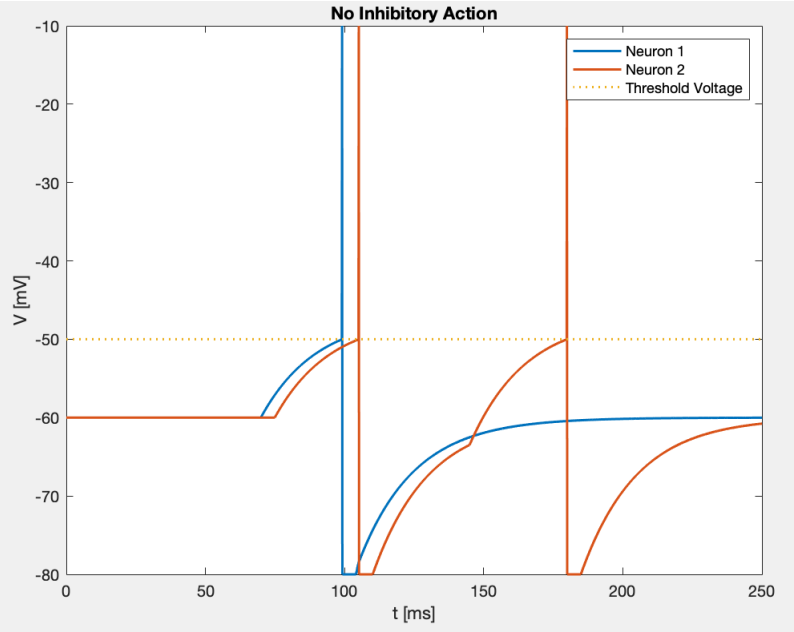
- Assume each neuron of RP1 emits an inhibitory neurotransmitter E_{RP1} when spiking, and similarly for CB with E_{CB}. Using concentrations, the model is:

$$\frac{dV_{RP1}}{dt} = \frac{1}{C_{RP1}}\left(-\frac{(V_{RP1} - V_{RP1}^{eq})}{R_{RP1}} + I_{ext}\,(1 - E_{CB})\right)$$

$$\frac{dV_{CB}}{dt} = \frac{1}{C_{CB}}\left(-\frac{(V_{CB} - V_{CB}^{eq})}{R_{CB}} + I_{ext}\,(1 - E_{RP1})\right)$$

$$\frac{dE_{RP1}}{dt} = d_{RP1}E_{RP1} \qquad \frac{dE_{CB}}{dt} = d_{CB}E_{CB}$$

Antagonistic Neural Circuits (cont.)



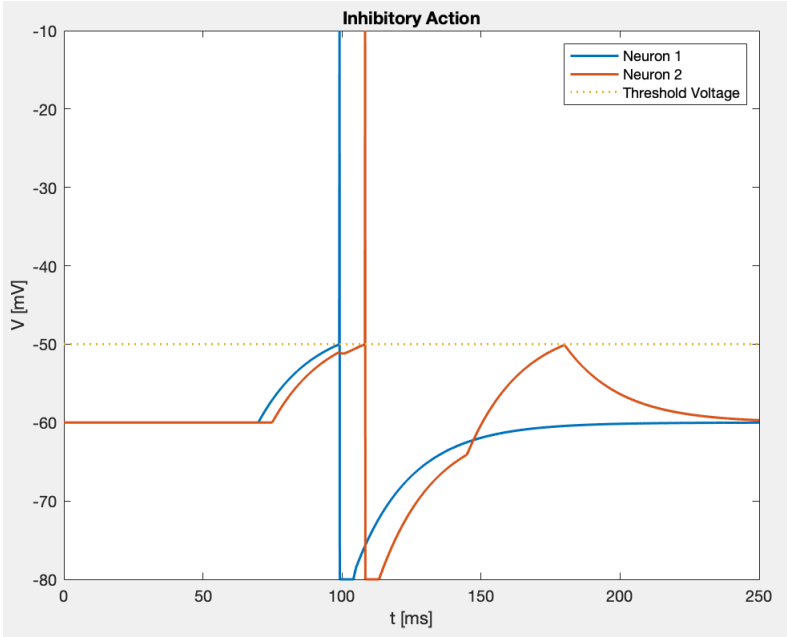


Figure: Simulation of two LIF neurons not acting on each other, and acting antagonistically, respectively. Input current is same in both simulations.

SNN simplifications

In our nerve network model, instead of fully simulating a continuous waveform for the state of the neuron, we keep track of the spikes at discrete timesteps instead. The primary advantage of spiked neural networks comes from the temporal coding, and this can be preserved with discrete timesteps.

One advantage of this approach is that the timestep size can be decreased if we find that information is not being encoded temporally as expected, or can be increased to save on processing power.

Tensor Product Optimization

We propose imposing non self-similar structure on the "blank" (before learning) nerve nets in the following fashion:
Where $H_1, H_2, \ldots, H_L$ are all graphs, we set

$$G = H_1 \otimes H_2 \otimes \cdots \otimes H_L$$

Where "$\otimes$" is the graph tensor product, equivalent to the matrix tensor product of the adjacency matrices of the graphs.
Tensor product $H \otimes K$ is equivalent to replacing every vertex in H with a copy of K. Vertices have connections between them if their counterparts in either H or K have edges between them. This means a random walk on $H \otimes K$ can be simulated by a random walk on H and K at the same time.

Tensor Product Optimization

- Note that the space required to store a graph as an adjacency matrix is $O(n^2)$ on the number of vertices
- If we assume H_i has n_i vertices, then we only require $\sum_{i \in [1,L]} (n_i)^2$ space to store the component graphs
- On the other hand, storing G by itself requires space $\prod_{i \in [1,L]} (n_i)^2$

Tensor Product Optimization, continued

For large networks, the aforementioned size differences become very significant

- Network with 5 layers of 100 neurons each will take only $5 \cdot 10^4$ units of memory to store by adjacency matrices
- the above 5-layer network describes a complex network with $100^5 = 10^{10}$ neurons that would take an adjacency matrix of size 10^{20} to describe
- assuming 1 byte memory per connection (very low estimate), the 5-layer network takes up about 50MB, while it's counterpart takes around a million TB to store directly.
- Simulating and training this network is still computationally difficult, but space complexity of is reduced

Biological Motivation

Making no assumptions about which genes affect the structure of the human brain, it is clearly impossible for the exact structure of the $> 10^{10}$ neurons and $> 10^{12}$ connections between them to be encoded in the mere $4 \cdot 10^9$ base pairs of the human genome. This tells us that there must be *some* sort of repeated structure in most nerve networks. Granted, this repeated structure might not be in the form of graph tensor products,b

Short term goals

- Using existing data in the literature, finish creating out model of the *Hydra* antagonistic nerve nets
- After creating a network that exhibits temporal signalling behavior, we hope to find the largest timestep Δt that preserves this behavior. Knowing the largest possible timestep lets us run simulations faster, and grants insights about the nature of the temporal signaling.
- Extend the tensor product optimization to allow replacement of vertices in H_i with different graphs, not just H_{i+1}. This will in theory allow the representation of a wider variety of graph patterns, notably allowing differences in local structure between larger regions

Long term future work

- Train the framework described here for *Hydra* on actual behavioral data from specimens. With sufficient training, the network described should be able to, in more or less real time, predict the behavior of *Hydra* in response to external stimuli.
- Chemical signalling in the human brain plays a far more complex role than the relatively simple antagonistic networks in *Hydra*. Extending our model to the human brain is far outside the scope of this project or modern technology, but perhaps some insight into chemical signalling systems in nerve nets can be gained.
- With more information about the nature of the temporal signalling, it may be possible to model the frequency domain instead of the time domain. This would be both easier for a human to read, and less computationally expensive.