Computers in Physics, Vol. 11, No. 5, Sep/Oct 1997:

Computer Simulations of Neural Networks

P.M.C. de Oliveira

Presented by Charles Tahan, April 18, 2000.

1 Overview

— A Physicist's Guide to the Brain —

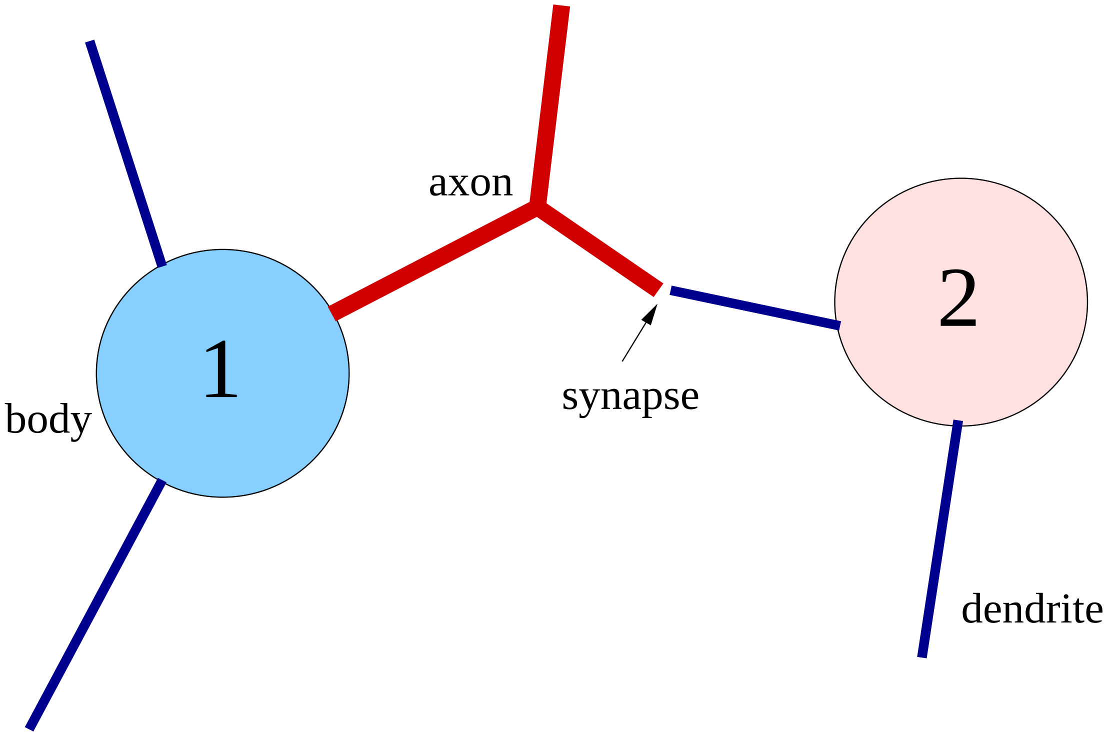
1.1 First, a pinch of biology: *Neurons*

- The brain is made up of $\sim 10^{12}$ neurons.

- Neurons communicate via electrical and chemical signals carried by **dendrites** and **axons**.

- Very interconnected, inputs from $\sim 10^4$ other neurons. Not two-way.

- Neurons **fire** as a function of their inputs (high, roughly periodic rate, or much lower rate). ($>>$**inhibitory** and **exhibitory** synapses)



- Neuron's **state**: firing or not firing; a function of its inputs.

- Connections are not limited to neighboring neurons, but can extend to neurons in remote areas of the brain.

- Many Neurons = Neural Network = **Brain = Remarkable**

1.2 Neural Networks

- Associative Memory (classification, reconstruction, training, learning).

- Understanding how a Neural Network = brain is a big (and cool) problem. (AI, pattern recognition, memory).

– A Physicist's Guide to the Brain 2 –

1.3 The **Ising Model**! (What else?)

- **Discrete** matrix of neurons with two states: firing or not firing (up spin, down spin ... sound familiar?)

- With **Long-Range interactions**. What are interactions?

- Assume no time-dependence of synaptic signals. (ok, biological evidence for non-synchronousness ... maybe)

1.4 A Modified Ising Model

• In the typical Ising Model, the effect of a spin on its neighbors was through the exchange energy. (remember, magnet)

• Similarly here, but we must consider every pair of spins.

• We define the effective energy of the neural network as

$$E = -\sum_{i,j} J_{i,j} s_i s_j,$$

where $J_{i,j}$ is the interaction energy: describes the influence of neuron i on the firing rate of neuron j.

$\sum_j J_{i,j} s_j$ determines the firing rate, ie spin, of neuron i. (Think of J as the effective field (eg magnetic) established by its neighbor's spins.)

• The connections in a real neural network are **not symmetric**: $(i \to j) \neq (j \to i)$; but, hell, what if they were? (Question: Why is this asymmetry important?)

$\Rightarrow$ We could use some Statistical Mechanics!

• E gives us a way to model a neural net via **Monte Carlo** by seeking a lower energy state,

only we define what the lowest energy state is through J (more on this later).

$\Rightarrow$ Apply these assumptions to the application of **memory**.

2 Application

— Memory via a Neural Net —

• In order to recall a pattern in a neural net memory, we require that the spin directions change with time such that the spin configuration ends up in a desired state.

The MC Approach:

1. Start with a particular configuration of the entire network.

2. Choose a spin and calculate the energy required to flip it, ΔE, from our relation for E.

3. If $\Delta E_{flip} < 0$, flip it.

4. If $\Delta E_{flip} \geq 0$, don't flip it.

2.2 Storing a Pattern

- The energy landscape, J, of the system defines how a memory is "stored" in the neural net.

- A very rudimentary definition is

$$J_{i,j} = s_i(m)s_j(m)$$

where $s_i(m)$ denotes the configuration of spin i in pattern m.

- Effectively we are defining the lowest energy state of the system as our pattern.

- We are not limited to one pattern,

$$J_{i,j} = \frac{1}{M}\sum_m s_i(m)s_j(m).$$

- But we are limited...

2.3 Limitations

- N spins $\rightarrow N^2$ different $J_{i,j}$.

- We'd think $N^2/N = N$ different patterns could be held.

- Actually, $\sim 0.14N$.

- The **Hamming distance**, energy minima, and interference.

$$\Delta_{m,n} = \frac{1}{N}\sum_i [s_i(m) - s_i(n)]^2$$

- Still fairly resilient, content addressable memory...

3 Examples

— Letter Recognition —

4 Enhancements and Other Oddities

• Learning,

$$J_{i,j}(new) = \beta J_{i,j}(old) + \alpha s_i(p) s_j(p),$$

(handwriting recognition, ...)

• More states for neuron: color picture memory, image processing.

• **feedforward networks**, layered neuron only communicate through the next layer (perceptron).

• An NN has been shown to be capable of universal computation (Turing machine).

• Seem capable of massively parallel processing.