

# C. elegans connectome dataset exploration

Sergio Peignier

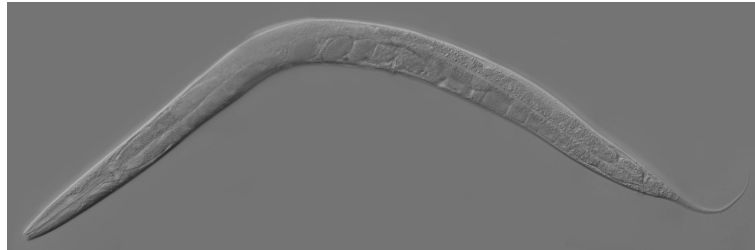


Figure 1: Adult *Caenorhabditis elegans*

## 1 C. elegans and the Openworm project

*Caenorhabditis elegans* is a small ( $\sim 1$ mm length) not parasitic transparent nematode (roundworm) that lives in temperate soil environments. *C. elegans* worms live around 3 years and have large progeny (around 300 children). These worms are mostly hermaphrodite ( $\sim 99\%$ ), and like other nematodes it lacks a respiratory and a circulatory system. They have a digestive system that includes a mouth, pharynx and intestine. The locomotion of their bodies (i.e., dorsal and ventral bending) is ensured by four longitudinal bands of muscles that receive signal from the neural system. The movements of the head are controlled by four muscles wired independently. When the waves of contractions of the body muscles proceed from the back (resp. front), then the animal moves backward (resp. forward). An interesting video that illustrates the locomotion of *C. elegans* can be found in <https://www.youtube.com/watch?v=GgZHziFWR7M>.

*C. elegans* is often used as a model organism and recently an interesting international project called the OpenWorm project [KHL98] has been developed (<http://www.openworm.org/>). This project aims to simulate *C. elegans* at the cellular level. The first stage is to study the locomotion system of the worm by modeling the 302 neurons of the hermaphrodite worm and the 95 muscular cells. The long term goal of the OpenWorm project is to simulate the entire worm (959 somatic cells).

In this practical course we will analyze the connectome of *C. elegans* that has been found by the OpenWorm project so far.

## 2 Connectome exploration

- Download the data from <https://github.com/3BIM20162017/CElegansTP/tree/master/data>.
- Three files are available at the previous url. What does each file contains?
- Open the files, read them and build two dictionaries:
  - The first dictionary contains the names of the pre-synaptic cell (or function) as keys and dictionaries as values. Inner dictionaries have the names of post-synaptic cells as keys and the weights of the corresponding synapses as values (excitatory connections are positive and inhibitory ones are negative, and the absolute value is equal to the number of synaptic connections between both cells). This dictionary will be called *ConnectomeDictionary* from now on.

- The second dictionary simply contains the names of the cells (or functions) as keys and their types as values (muscular-cell, inter-neuron, motor-neuron, sensory-neuron, sensory-motor-neuron, sensory-signal). This dictionary will be named *TypeDictionary* from now on.
- Using both dictionaries compute, for each type of cell (or function), the number of signals received from each other types (frequency matrix).
- Using this matrix compute the incoming type probabilities and outgoing type probabilities. What do you observe?
- Compute for each cell the number of incoming connections and outgoing connections. Save this information in two lists. Sort the given lists and plot the number of incoming and outgoing connections for each cell. What do you observe?
- Open the file in the folder called spatialpositions, and plot in 3D the position of each neuron.

### 3 Leaky Integrate and Fire (LIF)

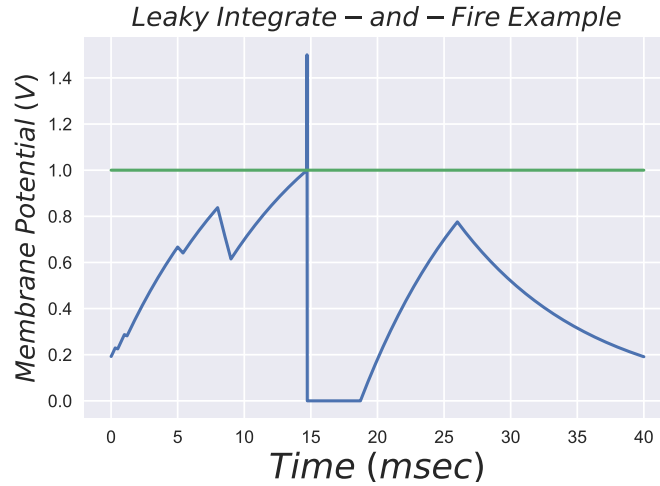


Figure 2: Leaky Integrate and Fire membrane potential evolution

The Leaky Integrate and Fire [Abb99] is a very simple model of neurons. Here, the voltage changes of the neuron behave in two different ways:

- When  $V$ , the voltage of the membrane of the neuron is under a given *threshold value*, the membrane acts as a leaky capacitor (With capacitance  $C_m$  and resistance  $R_m$ ) and its voltage decays to a resting level  $U_{rest}$ . The neuron receives an excitatory input (electric current)  $I$ .

$$\tau_m \frac{dV}{dt} = U_{rest} - V + R_m I$$

$$\tau_m = R_m C_m$$

- When the voltage reaches the action potential threshold the neuron spikes immediately to a level  $V_{spike}$  and then its potential is immediately reset to a hyper-polarized level  $V_{reset}$  and remains in a refractory state during a given period of time  $\tau_{ref}$ .

In Figure 2, we see the typical behavior of the voltage of a neuron following the Leaky Integrate and Fire model ( $V_{spike} = 1$ ,  $V_{reset} = 0$ ,  $U_{rest} = 0$ ,  $R_m = 1$ ,  $C_m = 10$ ,  $\tau_{ref} = 4$  and  $V_{spike} = 0.5$ ).

- Write a function that receives the required parameters and computes the derivatives of  $V$ .
- Test this function using an Euler integration scheme. Excite the neuron during a period of time, then leave the neuron without input and excite again ... You should be able to reproduce similar results than those depicted in Figure 2.

## 4 Neural Network

- Create a class "Neuron" that contains:
  - All the parameters for the LIF function
  - A variable encoding the actual voltage of the neuron.
  - A variable encoding the next state of the neuron.
  - A dictionary with all the cells to which the neuron fires and the weight of the connection.
  - A function that computes the next state of the neuron and returns 1 if the neuron is firing.
  - A function that excites the neighboring cells.
  - A function that updates the voltage of the neuron.
- Create a class "Neural Network" that has:
  - A dictionary containing the name of each cell as a key associated to an instance of the class "Neuron" that represents the given cell.
  - A list with the neurons that should firing at the current step.
  - A function that fires the neurons that should fire at the current step, updates the list of firing neurons and updates each neuron.
- Test the program exciting some of the neurons repeatedly during a large amount of time and recording the evolution of the entire system.
- According to some studies [CSW<sup>+</sup>85] the following neurons ('PLML', 'PLMR', 'PVM', 'IL1VR', 'IL1VL', 'IL1DR', 'IL1DL', 'ALML', 'ALMR', 'AVM') are involved in the transmission of signals when the "tail" of the animal is touched. While the following set of neurons would be involved in the transmission of sensory information when the "nose" of the animal is touched ('ALML', 'ALMR', 'AVM').

How the system behaves when the back or the front of the animal is touched? Which neurons are activated when the front is touched but are not activated when the tail is touched?
- Plot the spatial distribution of the motor neurons that are activated with a different color according to the time stamp when they are activated.
- Take a look to <http://wormwiring.org/data/analyticTools.html> to visualize the connectivity of the network. Here <http://wormatlas.org/neurons/Individual%20Neurons/Neuronframeset.html> you can find a more detailed description of each neuron. Using the spatial information you can draw the flow of information in the worm.
- Choose some neurons and excite them, test which other neurons are activated and analyze the patterns produced.

## References

- [Abb99] Larry F Abbott. Lapique's introduction of the integrate-and-fire model neuron (1907). *Brain research bulletin*, 50(5):303–304, 1999.
- [CSW<sup>+</sup>85] Martin Chalfie, John E Sulston, JOHN G White, Eileen Southgate, J Nicol Thomson, and Sydney Brenner. The neural circuit for touch sensitivity in *caenorhabditis elegans*. *Journal of Neuroscience*, 5(4):956–964, 1985.
- [KHL98] Hiroaki Kitano, Shugo Hamahashi, and Sean Luke. The perfect c. elegans project: An initial report. *Artificial Life*, 4(2):141–156, 1998.