Project 3

100-cell network model

Network model of the amygdala to study fear learning during Pavlovian fear conditioning

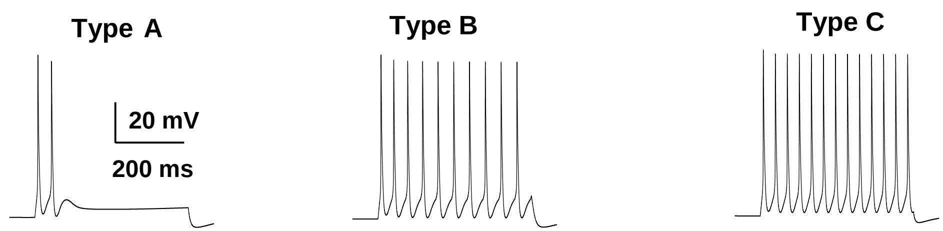
Project Objective

- Design a network 100-cell network model with specified network-level performance.
- Learn how to adjust network parameters, i.e. synaptic parameters, to obtain the network-level performance.
- Learn how to post-analyze the outputs from network simulation results.
- Specifically, you need to match the spontaneous firing rates, and number of long term plasticity (LP) cells from the simulation of network model.
- Note that, the codes (NEURON, Matlab) we provided here is not the only solution. Other tools, like C, Python are welcome too.

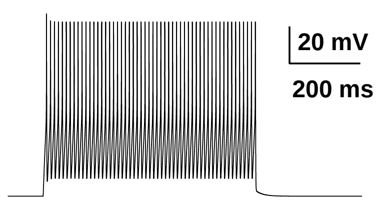
Before Network

- Before network, you already know how to build a single cell.
- In this project, we provided 4 different single-cell templates that reproduced 3 types of principal cells and 1 type of interneuron in LA (for review, see Sah et al., 2003).
- These 100 neurons consisted of 80 principal cells (A:B:C= 40:24:16) and 20 interneurons.

principal cells

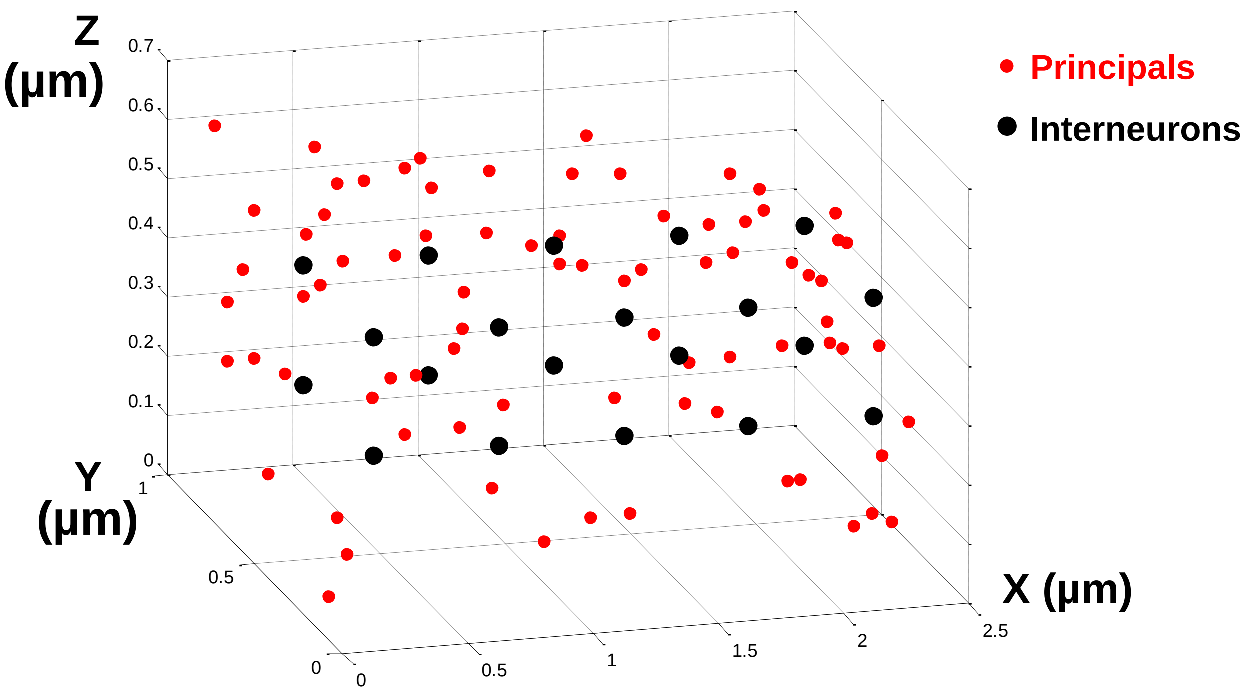


Interneurons



Setup Network

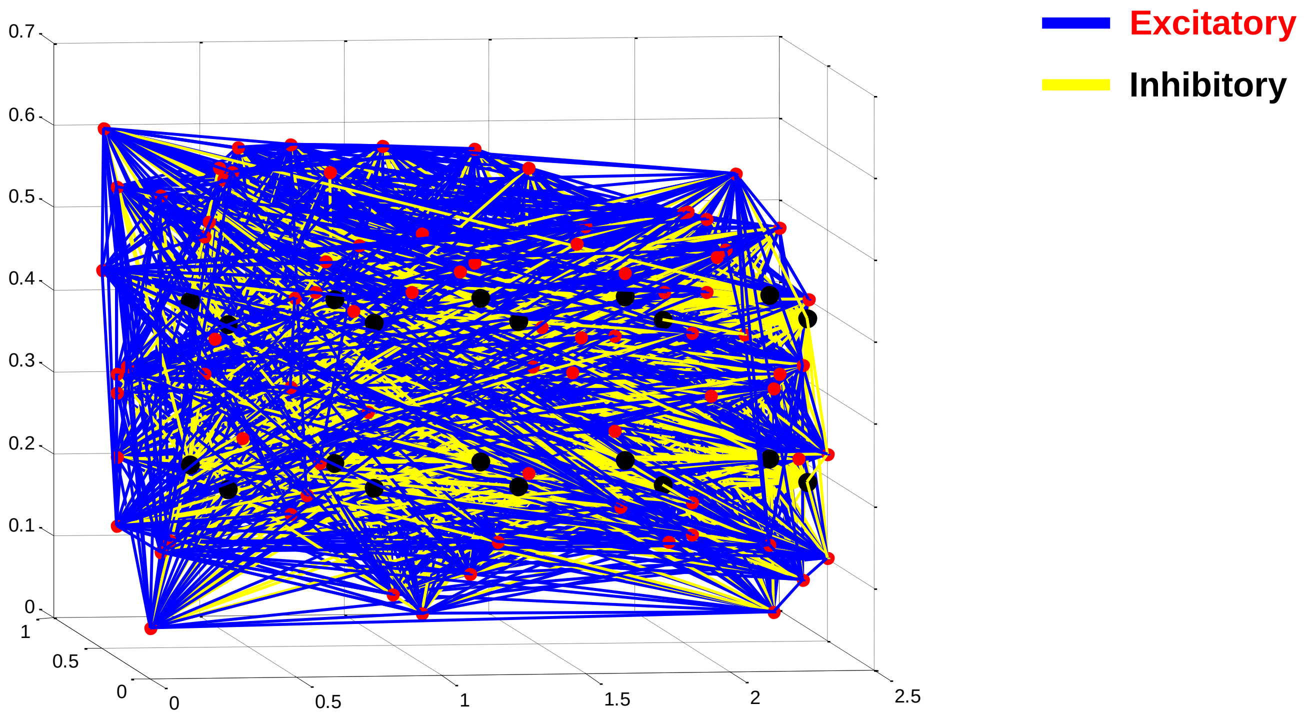
- 100 cells were randomly distributed in the cuboid 3D space



Setup Network (Connectivity)

- Connections were established in a random fashion.
- Between Pre- and Post- cells, connectivity probabilities are as in the following table.

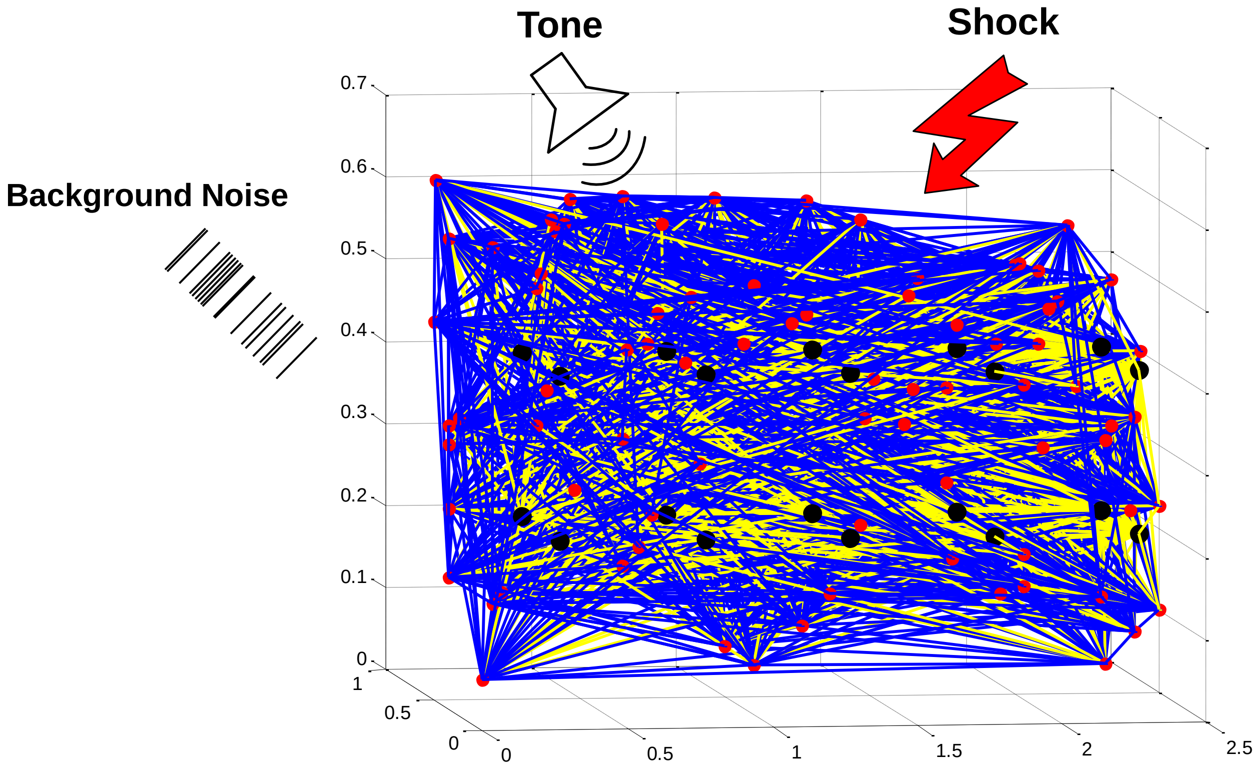
Pre \ Post	Principals	Interneurons
Principals	25%	33%
Interneurons	60%	0%

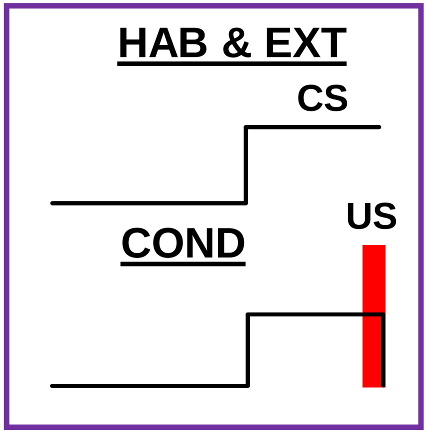


Setup Network (Tone and Shock)

- **Tone and shock were introduced to randomly chosen neurons, at following distribution:**

	Principals	Interneurons
Thalamic Tone	53%	50%
Cortical Tone	53%	50%
Shock	70%	70%





Setup Network (.txt files created)

- There were several .txt files created during the process of establishing network, which are needed for the simulation.

1. Cell_type.txt : list the cell type for each neuron. 1, 6 and10 stands for A,B and C type respectively.
2. NM.txt: list the NM type for each neuron. 0, 1, 2 and 3 stands for none, DA, NE and DA&NE.
3. Syn_Matrix.txt : 100×100 connections matrix file.
4. tone2LAd.txt, tone2LAd2.txt: list the ID of principals receive thalamic and cortical tone, respectively.
5. tone2Id.txt, tone2Id2.txt: list the ID of interneurons receive thalamic and cortical tone, respectively.
6. shock2LAd.txt, shock2Id.txt: list the ID of principals and interneurons respectively which receive shock.

Setup Network (synaptic .mod files)

- There are three types of synaptic .mod files in the folder, which are synapse files used for connecting neurons and introducing external drives (Noise, Tone and Shock).

1. bg2pyr.mod, bg2inter.mod : synapses for background noise for principals and interneurons.

2. tone2pyrD_new.mod, shock2pyrD.mod :Tone and shock synapses for principals.

3. tone2interD_new.mod, shock2interD.mod :Tone and shock synapses for interneurons.

4. pyrD2pyrD_STFD_new.mod, pyrD2interD_STFD.mod, interD2pyrD_STFD_new.mod : synapse files for P2P, P2I and I2P connections, respectively.

Run Simulation

- After compiling the folder, double click LA_model_main_file.hoc to run the simulation. You can see the progress percentage when you run the simulation.
- Based on the protocols, simulation were designed for 276s-long of computing.
- In this project, we only need the spike times of each neuron in the network. So, when simulation is finished, you will find a file named **data** in the same folder.
- **data** file records the spiking times (first column, unit in ms) and the corresponding IDs (second column).
- Based on the protocol, you can code via other tools, e.g. Matlab, to post-analyze **data** file (See "How to analyze simulation results" part).

Tuning Goals

- **Our tuning goals for this project are**

1. Spontaneous firing rates for principals and interneurons should be in the ranges of 0.5-1.5Hz, and 5-15Hz, respectively.

2. The number of LP cells needs to be in the range of 10-40% (Tovote et al. 2015) of total number of principals, i.e. 8-32 LPs.

Suggested tuning procedures

Note that : Except shock synapses files, we left all the initW values to blank in synapse mod files for you to tune and decide.

1. First, match spontaneous firing rates. In this stage, you don't need to run the whole protocol, but only HAB phase is needed. To do this, you can change tstop in main.hoc to 32000.

You can tweak the weights of background synapses to see its influences on spontaneous firing rates.

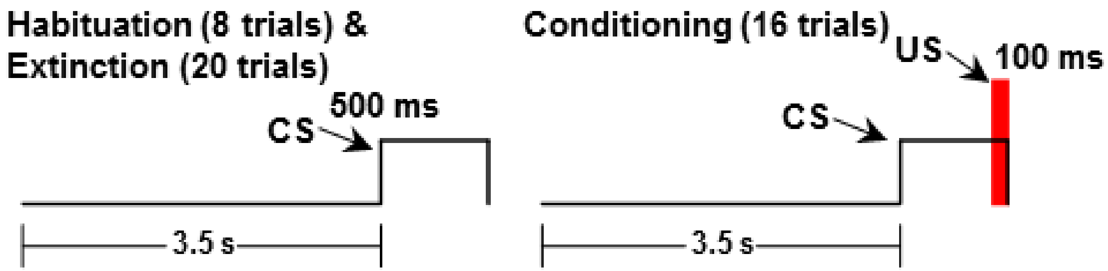
2. After tuning the spontaneous firing rates, run the compete protocol, analyze the data file and check how many LPs you have. Try to change several different initial weight values to check the certain synapse's influences on the number and responses of LP cells.

	suggested ranges for initW
bg2pyr.mod	0.7-2
bg2inter.mod	7-10
tone2pyrD_new.mod	0.7-2
tone2interD_new.mod	2-4
pyrD2pyrD_STFD_new	0.5-1
pyrD2interD_STFD_new.mod	1-2
interD2pyrD_STFD_new.mod	0.7-2

How to analyze simulation results (based on protocols)

- **1. To analyze simulation results, we detailed our protocol here:**

The auditory fear conditioning protocol included three phases (habituation, conditioning and extinction), comprised of 8, 16 and 20 trials, respectively. Each trial featured a 0.5 sec tone CS followed by a 3.5 sec gap. Only during conditioning, a shock (100 Hz) was administered 100 msec prior to the end of the tone, so that they co-terminated. The frequency of thalamic and cortical tone inputs was increased to 40 Hz after the first and sixth conditioning trials, respectively (Feng et al, 2016).



How to analyze simulation results (spon firing rate)

- **2. How to calculate spontaneous firing rates**

Run the Matlab program, Spon_cal.m, which calculates the average spontaneous firing rates (Hz) for principals and interneurons:

- Each neuron's spontaneous firing rate was calculated from calculating the frequency of spikes during the periods prior to tones for the 3-8 trials in HAB.

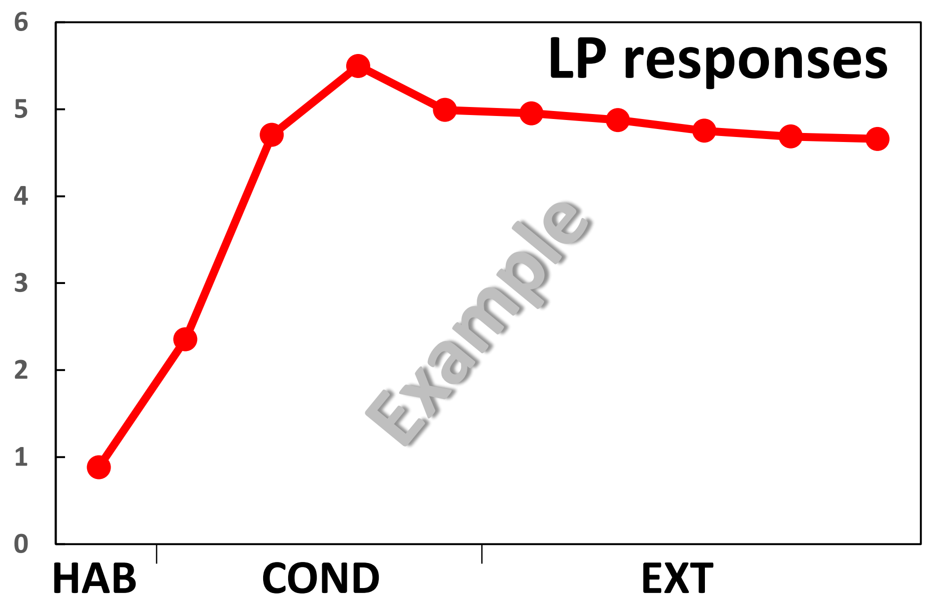
- **Pyr_Spon_mean** and **Int_Spon_mean** from Matlab are for the average spontaneous firing rates for principals and interneurons respectively.

How to analyze simulation results (LP #s and tone responses)

- **3. How to calculate LP #s and their tone responses**

Successively run the Matlab program, Spike_count_100.m and Test_LP.m, you would get the LP#s and tone responses :

- The criteria we used to classify LP cells are based on Kim et al. 2013a and Feng et al, 2016.
- However, a more stringent criterion was used to classify the output tone responses: a threshold of 10 Hz was used to define LP cells; if either of the average tone responses of last two conditioning blocks was greater than 10Hz, it was classified as LP.
- Variables from Matlab **LP_Num** and **LP_response** records the # of LPs and their average tone responses, respectively.
- An example of average LP tone responses was given in the :



References

1. Feng F, Samarth P, Pare D, Nair SS (2016) Mechanisms underlying the formation of the amygdalar fear memory trace: A computational perspective. Neuroscience 322:370-376.

2. Kim D, Pare D, Nair SS (2013) Mechanisms contributing to the induction and storage of Pavlovian fear memories in the lateral amygdala. Learning & memory 20:421-430.

3. Sah P, Faber ES, Lopez De Armentia M, Power J (2003) The amygdaloid complex: anatomy and physiology. Physiol Rev 83:803-834.

4. Tovote P, Fadok JP, Luthi A (2015) Neuronal circuits for fear and anxiety. Nat Rev Neurosci 16:317-331.