

I implemented the first version of the ring attractor following the paper from Vergani and Huyck. To do that, I coded from scratch an Integrate and fire neuron model, with conductance based synapses and exponential currents with Python, using a timestep constant of 1ms and implementing the Euler method to approximate the differential equations. I then connected the neurons with a 2-4 connectivity rule (2 excitatory synapses and then 4 inhibitory by each side) to form a ring.

After that, I injected noise into the network, trying with different sources (correlated noise, Ornstein-Uhlenbeck process, white noise) and I concluded that white noise was the most relevant for our study. The noise has mean 0 and standard deviation between 0 and $3e-3$, and is computed independently for every neuron at every time step. I manually explored the behaviour of the network under different amounts of noise, to find stable weight strengths.

Then I started investigating the role of fixed points on reducing the variability of the bump position. I manually found the parameters to have a stable fixed point, and made multiple large-scale simulations, varying the number of fixed points and the amount of noise injected. The simulations drove us toward fixing some setup mistakes and exploring different error measures (mean, circular mean, centroid, fitted linear regression parameters) before developing a method able to capture:

- absence of activation
- explosion of activation
- small differences in variability of the bump shape
- small differences in the neurons activate at the beginning vs the end

To achieve that, Peter suggested fitting the first and the last third of the time course of the bump to a Von-Mises distribution, then to take the Kullback-Leibler divergence between the two distributions.

Then I moved the code on Julia for performance reasons, reducing the time to simulate 10 seconds of activity of 256 neurons from ~ 80 seconds with Python and Numba to ~ 0.3 seconds with plain Julia.

This allowed to simulate the activity of the network with high sensibility and speed over different values of global weights, observing a bifurcation between stability and explosion over the diagonal of the weights matrix, so that every time the excitatory weights are greater than the inhibitory, the activity of the whole network becomes active. Adding noise in the same simulation setup revealed that the bifurcation line moved so that excitatory weights need to be quite smaller than inhibitory to have stability.

I also repeated a similar simulation, but only changing local weights around the ignition point, and measuring activity only in that range. That allowed us to study how weights need to be changed locally to obtain a stable fixed point, so that I can use this knowledge to infer plasticity parameters that allow the autonomous formation of fixed points on neurons that have been stimulated.