# Hebbian Training for Jansen and Rit Model Networks

In this word document I will note the different problems and achievements I encounter during the development of the Hebbian Learning part of my thesis. This way I will be able to have a more thorough description of my work process and clearer ideas when I have to attend meetings or start writing the thesis.

## Until 6/04/2021

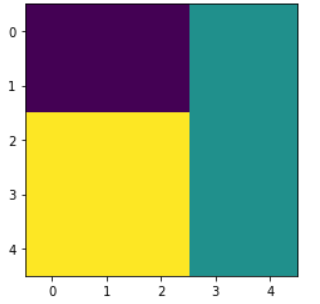
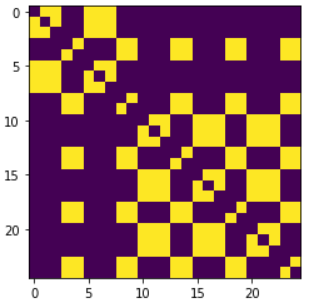
Today almost all the first layer part (excitatory connection matrix / auto-associative memory) is pretty much completed. Some questions have arisen which I will try to solve writing a mail to the Tonis.

Problems until now:

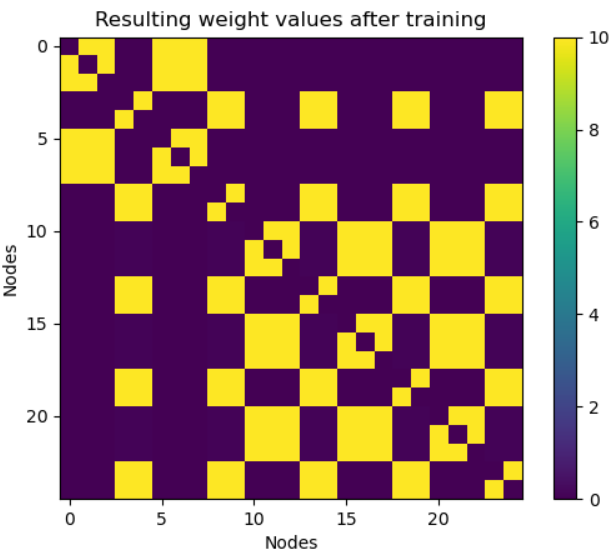
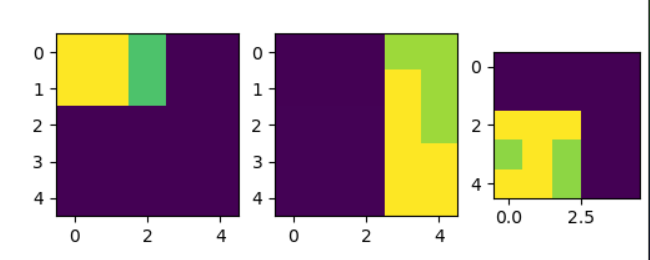
* In the JR model, the needed white noise of the external input to achieve alpha rhythm is between 100 and 300 Hz. This induces a certain firing rate in the pyramidal column that makes the Hebbian learning and simulation harder. As in the Ursino paper, it all simplifies when the resting (non stimulus) external noise is close to 0. However, the JR model for this case does not exhibit alpha rhythms. What could be possibly done?
* Furthermore, I don’t know if the fact that the same pattern, non-stimulated nodes, exhibit perfect 10 Hz, alpha rhythm is good or not.

Interesting figures:

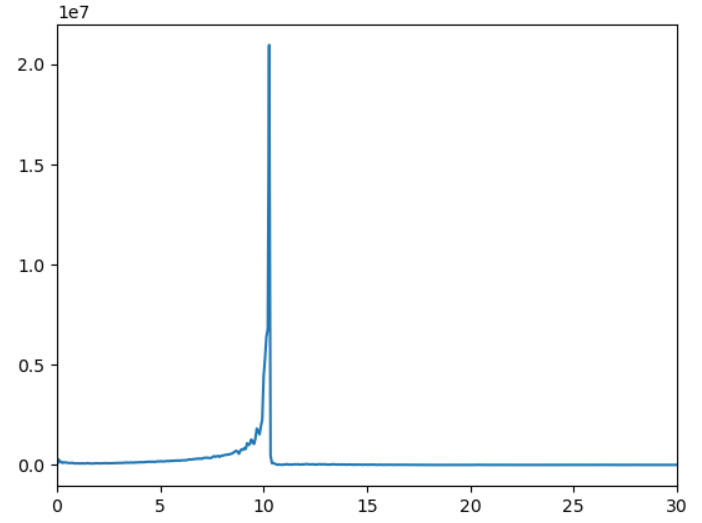
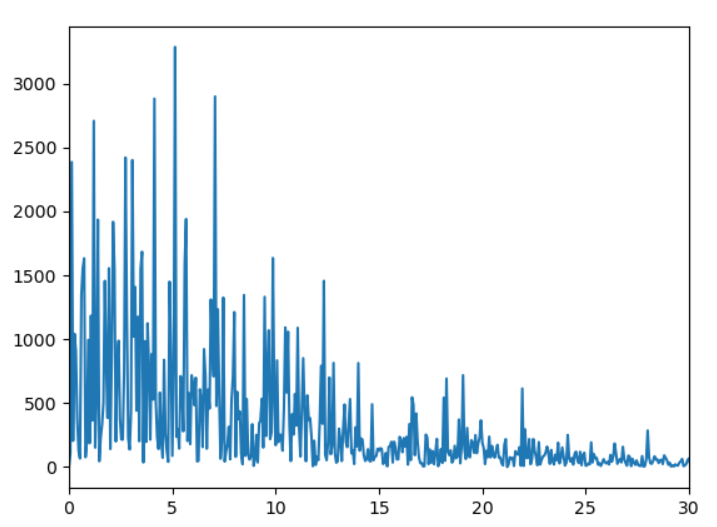
3 different patterns: Connection matrix of these 3 patterns:

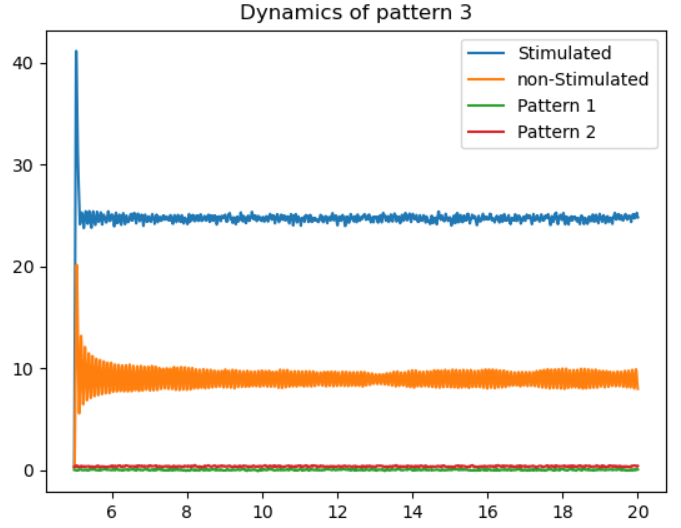
Actual connection matrix: Results when stimulating some of the nodes:

PSD of other patt, non-stimulated node: PSD of same patt, non-stimulated node:

Clear differences in the PSP potential value of the same patt, non-stimulated nodes.



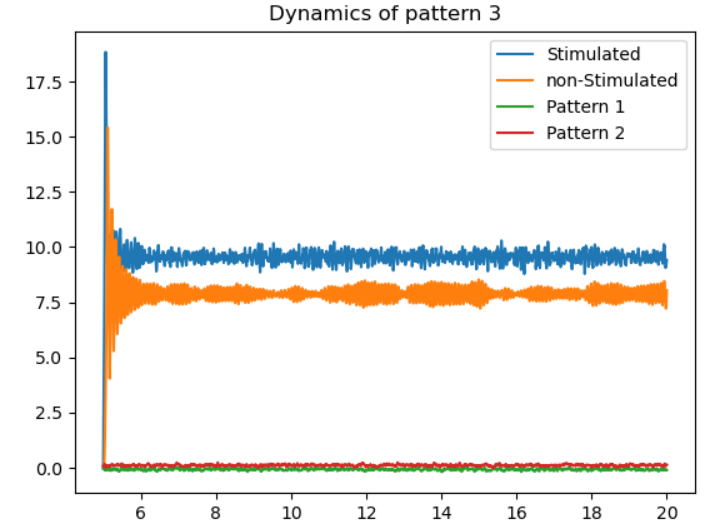
## 07/04/2021

Today I will try to start building the second layer, the connection between the nodes from the first and second layer, that should be constant and maybe think about the training of the second layer (two trainings). Ideas are to build a second matrix of dynamics like the first one, it will make everything clearer when having to index things. It will have its own connection matrices that will have to be trained in a similar way than the W matrix. It would be interesting to send the email to Toni and see what he thinks about the progress.

**I didn’t do it, I worked on the Wavenet paper**

## 9/04/2021

I will do what I wrote for the other day. Doing some tests on the first layer it’s possible to obtain closer values of nonstimulated nodes to the stimulated ones by reducing the magnitude of noise. It seems that there is a maximum that the excitatory coupling can reach before messing all the system up. When noise is set up to around 300 it is possible to obtain very close values:



Might be interesting to study correlations between signals in the same and in different nodes, see if they are synchronized or something like that.

The maximum values of the connectivity matrices will change when increasing the number of nodes in the network. More connections with the same value will increase the arriving excitatory signal.

It might also be interesting to reduce the value of the connection as a function of the distance.

Find the meaning/sense in whether I should use S(y1-y2) or y0 in the different excitatory/inhibitory coupling values. One is pulse density of the pyramidal population incoming to the populations and the other is the average density potential of the pyramidal population.

Done in the day: I think I have managed to build and debug the most part of the algorithm for this first case. However, the matrix W has changed resulting in some rare configuration that shouldn’t appear. The Hebbian learning still seems to work well, yellow where there should be, however there are differences with the first layer only matrix. It looks like increasing vL seems to fix it without incurring in problems.

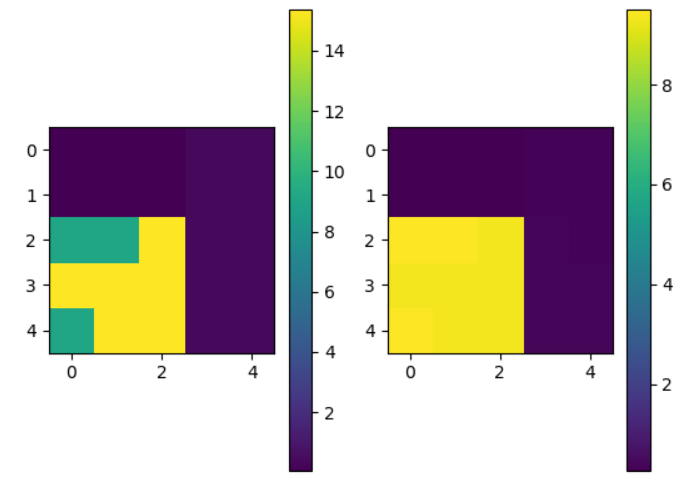
Next day I want to see how the dynamics of the second layer react when patterns are being stimulated, like one of the last figures I obtained. Some parameter exploration will have to be performed to see if it’s possible to start activating the W and O matrices.

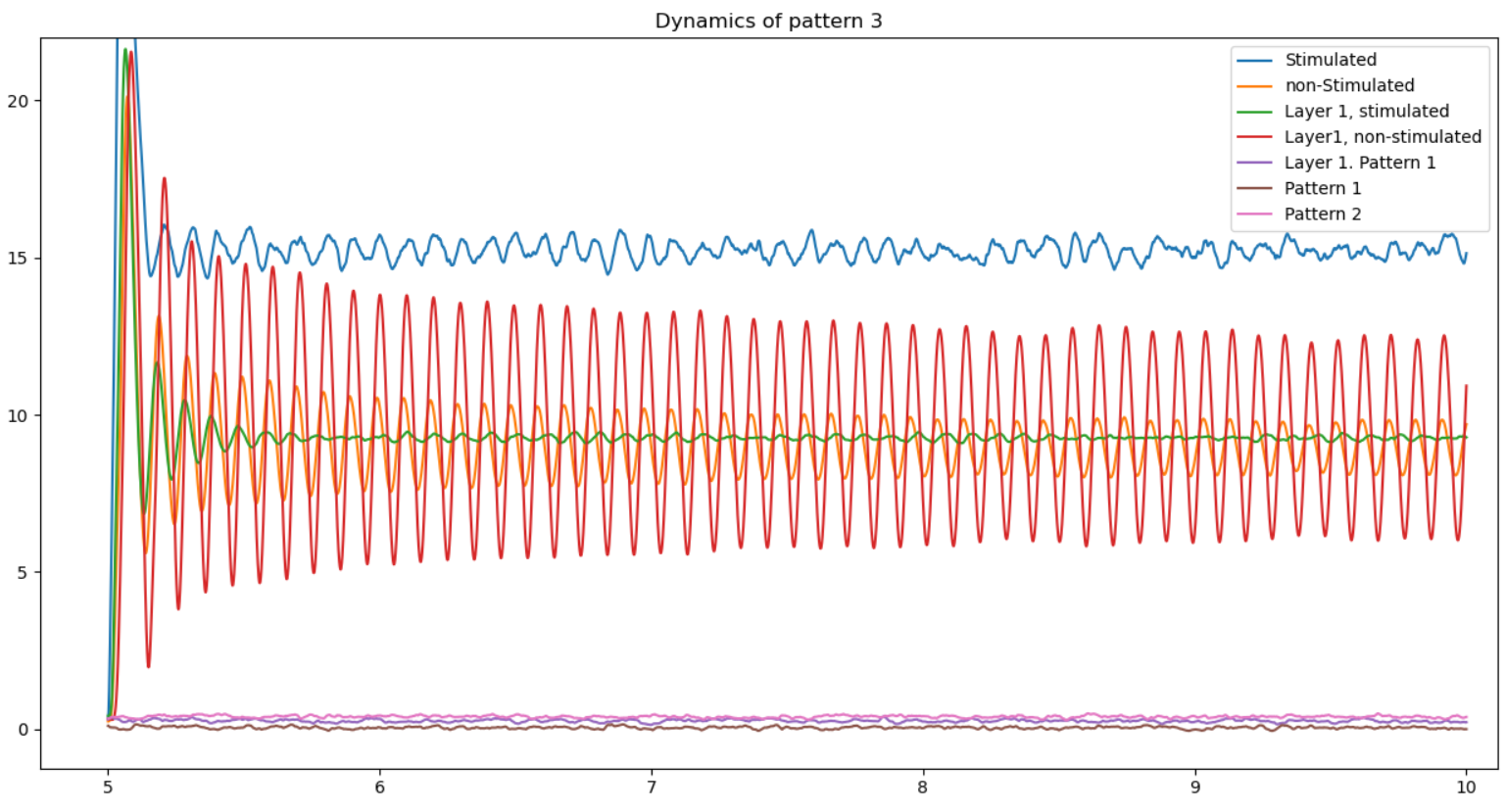
Nice achievement to obtain a non-error algorithm today. Still some theoretical aspects to be fully understood like what’s the difference in using S(y1-y2) and y0 in some of the places.

## 12/04/2021

Started doing some tests on the code that was left last day. Starting to see interesting things even though I still want to make sure that some of the code is written correctly (Last point of previous day). Some modifications of the interlayer inhibitory and excitatory parameters have been performed, trying to obtain plots with some kind of meaning (auto-associative memory to be passed to the subsequent layer).

Some of the interesting plots obtained (Inhibitory variable is wrong but whatever)



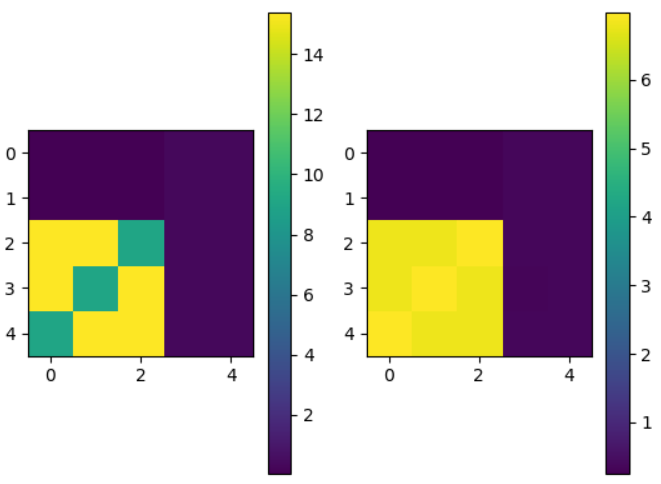


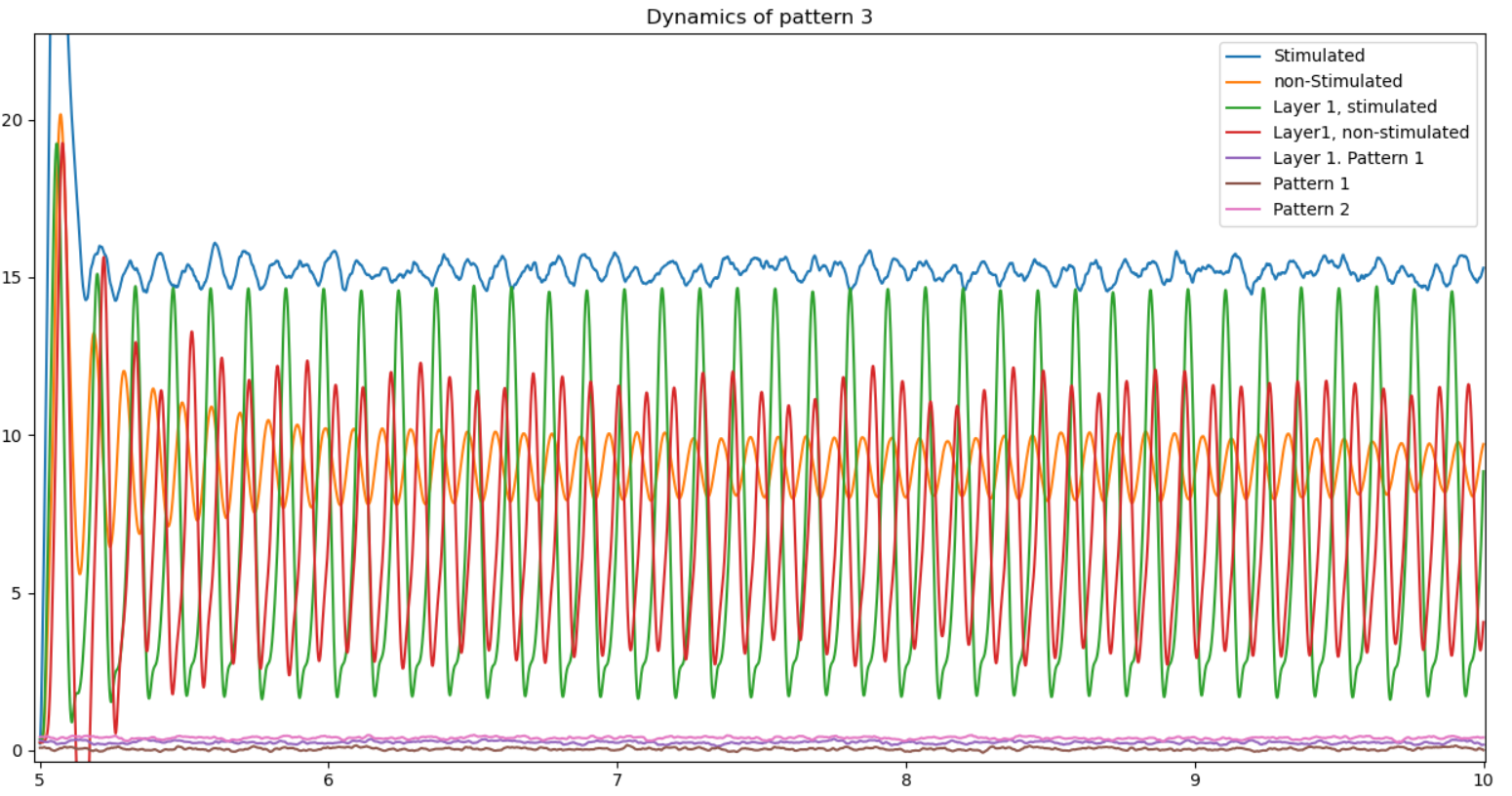
Partial answer to the previous doubt whether I had to use S(y1-y2) or y0 in the different coupling values. If the coupling value pa is introduced in the ode, then it has to have passed the transformation from potential to pulse density. When something enters the pyramidal population, it’s like adding a new input where the p(t) input is in the JR 1995 paper’s figure. There, the synaptic integration is performed in the he process. If the pyramidal population of one column *i* excites the population of column *j*, then we have to understand that the pulse density of pyramidal neurons in *i* arrives to the *j* excitatory synapses and is there integrated. Thus we have to write pa = K\*S(y1-y2).

If I want to excite the inhibitory population I think I will have to add an extra variable to the population, if not I don’t know how this can work… I will have to ask this tomorrow during the meeting to see if it’s worth it.

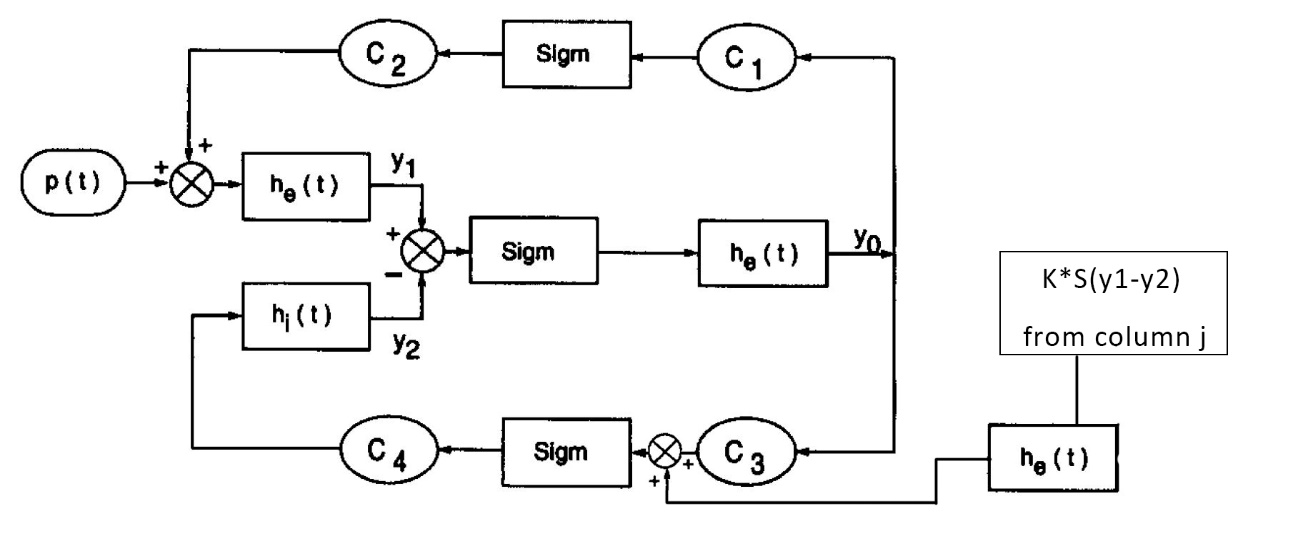
Code written, now ask Antonio if it’s okay or not.

Now, with the inhibitory modification added, let’s see how pattern 3 behaves, even though if matrix K is 0 it shouldn’t show any modifications.





Both the stimulated and not-stimulated nodes in layer1 are spiked and slightly desynchronized, even non-stimulated layer0/non-stimulated layer 1 are desynchronized. I don’t know which are the changes if K = 0.



Even if K is all 0 there will be slight modifications due to the fact that initial conditions are not 0. Trying to have IC = 0 for the inhibitory population still some desynchronization, makes me think that maybe it’s a random behavior.

Will have to make a thorough study on the parameters to see where it leads me. I should also modify some of the code to see how the inhibitory population variable behaves and also see what values the variables responsible for the K and O matrix training take in order to better tune the training params.