MECHANISMS IN NEURAL LAYERS

Bita Moharrami
University of Tehran

Supervised by: Mohammad Ganjtabesh

May 2024

Contents

Chapter 1

Lateral Inhibition

1.1 Lateral Inhibition with Completely Distinct Neurons in Various Patterns

First of all, we are going to build a neural network which is using STDP learning rule, and we will use the Lateral Inhibition mechanism in the output layer. Note that in these experiments active neurons for each pattern are totally distinct.
We give two different input patterns to our network:

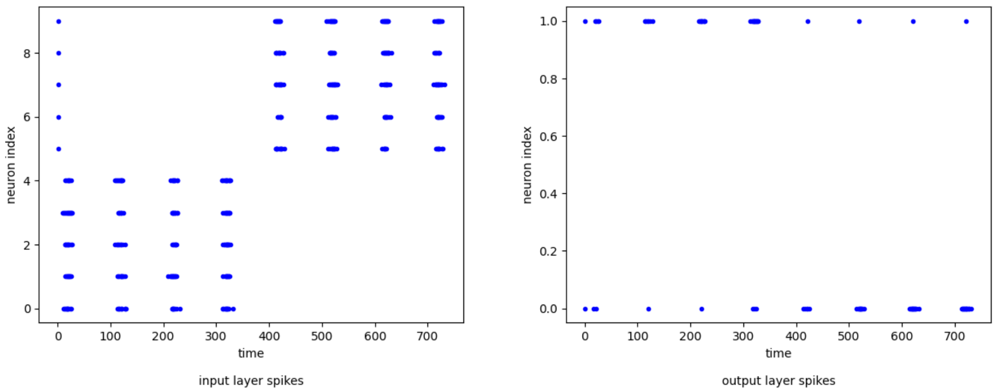

Figure 1.1: Given are two distinct input patterns, and we can see that each output neuron learns one of them.

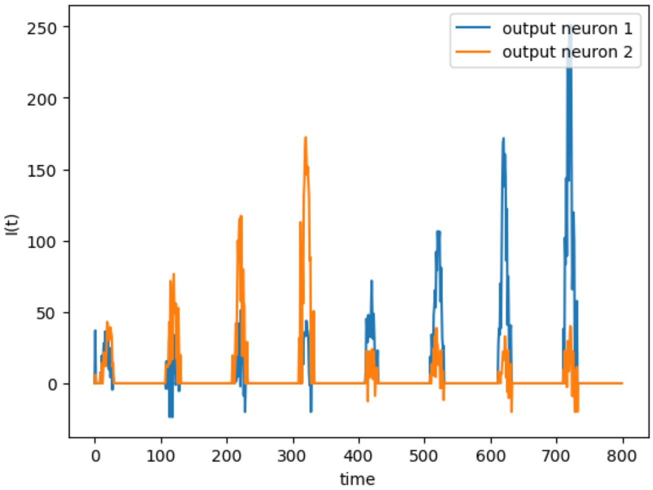

Figure 1.2: We can see that for each input pattern one of the output neurons is learning it while the other one is getting inhibited.

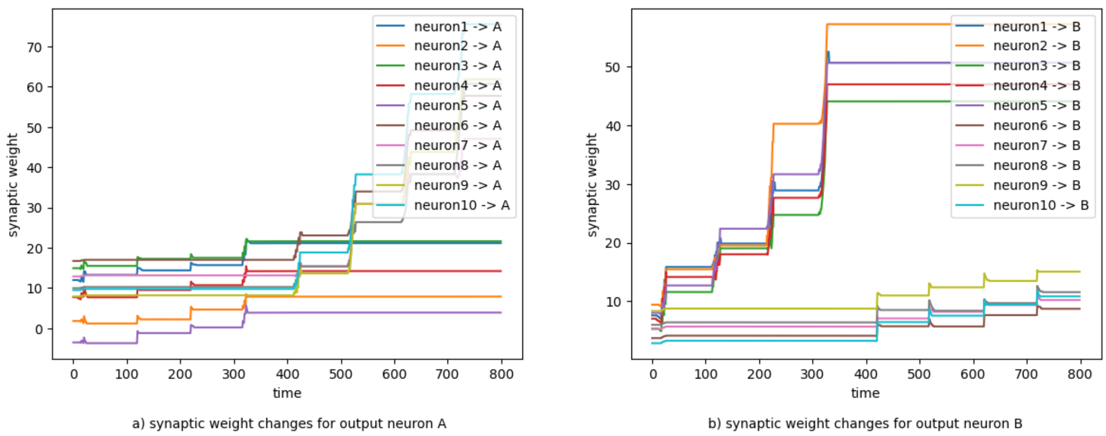

Figure 1.3: Synaptic weight changes for each output neuron

Now we are going to increase the current-coefficient value:

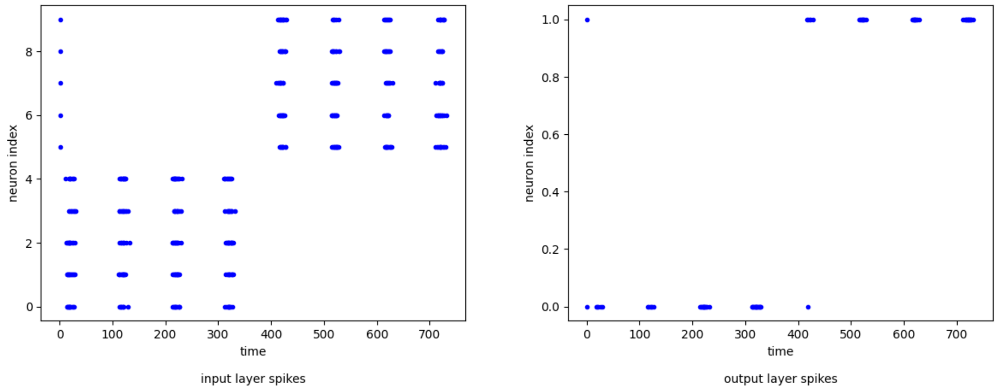

Figure 1.4: Given are two distinct input patterns, and we can see that each output neuron learns one of them.

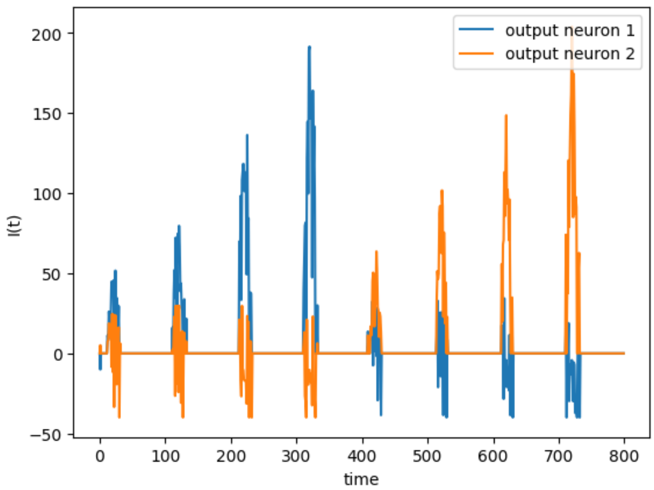

Figure 1.5: We can see that for each input pattern one of the output neurons is learning it while the other one is getting inhibited.

As it's clear in figs. 1.4 and 1.5, by increasing the current-coefficient value, each output neuron is learning one of the given patterns, this time in a more accurate way. Because by increasing the current-coefficient value, the inhibition of lateral activities gets stronger. So this leads to a sharper learning process.

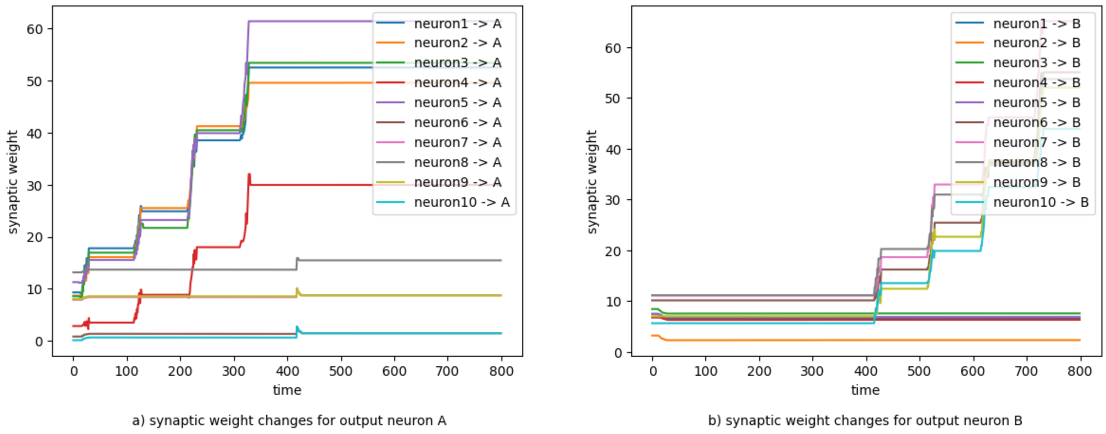

Figure 1.6: Synaptic weight changes for each output neuron (related to Figure 1.4)

1.2 Lateral Inhibition with Intersecting Neurons in Various Patterns

In this section, we have some shared neurons in our patterns to get closer patterns. We start with 2 common neurons in patterns:

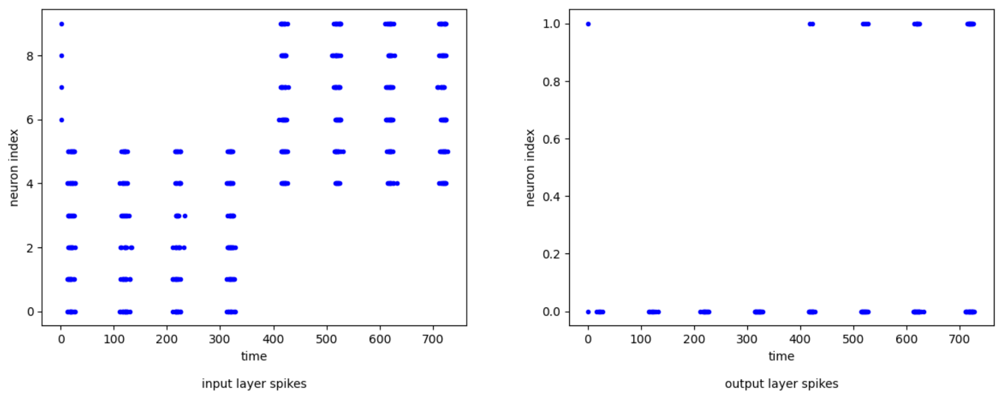

Figure 1.7: Given are two input patterns which have common neurons, and we can see that both output neurons are getting likely to learn both of the patterns!

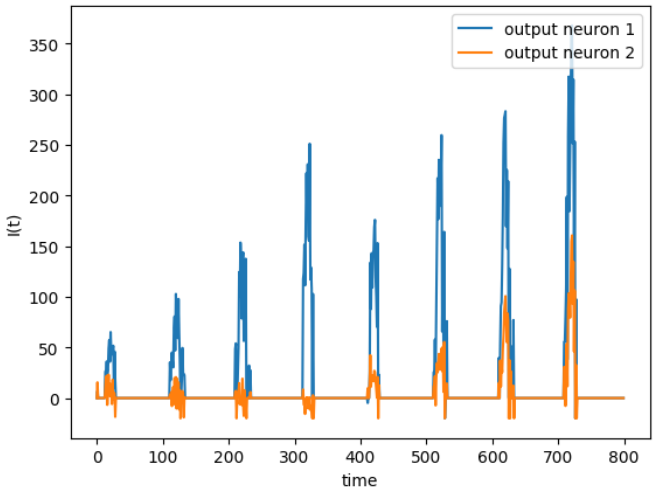

Figure 1.8: We can see that by adding intersecting neurons to our patterns, the effect of lateral inhibition intensely decreases.

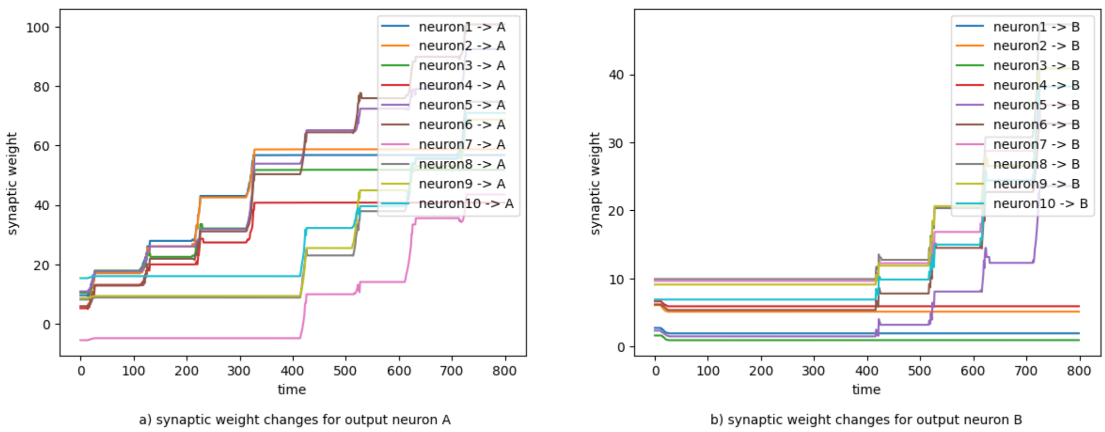

Figure 1.9: Synaptic weight changes for each output neuron

Now let's increase the current-coefficient value and analyse what happens:

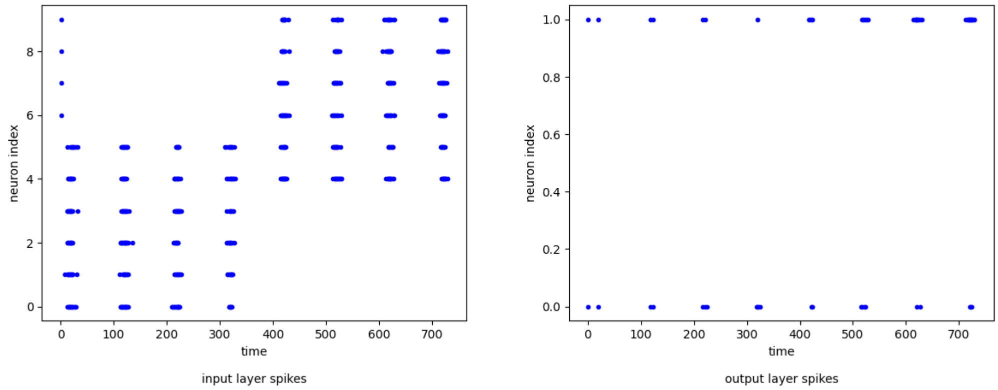

Figure 1.10: Given are two input patterns which have intersecting neurons; This time we increased the current-coefficient value. This time each output neuron is likely to learn exactly one of the patterns!

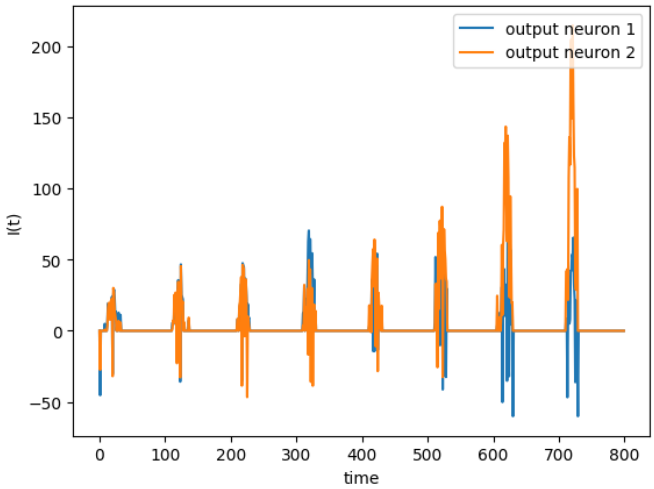

Figure 1.11: We can see that by increasing the current-coefficient value while we have shared neurons in our patterns, each output neuron tries to learn one of the patterns; But still existence of common neurons between patterns leads to the loss of sharpness in learning.

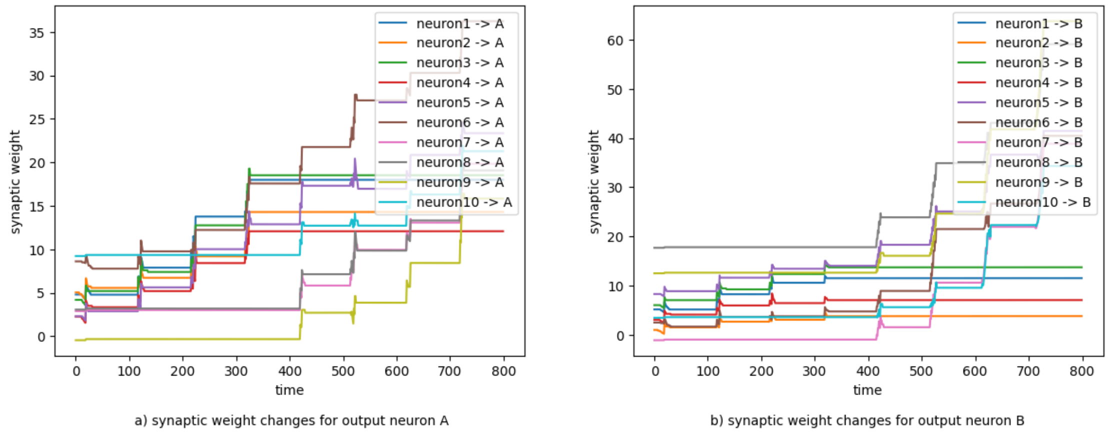

Figure 1.12: Synaptic weight changes for each output neuron (related to Figure 1.10)

Now, we are going to increase the number of common neurons between our patterns. Here we will have two states:

1. If lateral inhibition is weak, then clearly both output neurons will approximately learn both patterns. (see figs. 2.3 to 2.5)

2. If we use a strong lateral inhibition, then once one of the output neurons starts to learn a pattern, other activities start to get inhibited; But since we have lots of common neurons between our patterns, the other output neuron also starts to learn the given pattern over time (The inhibited parts are much less than the activated pattern parts.). And finally, we note that the first activated output neuron keeps learning all the time as its relevant pattern doesn't get inhibited anyway. (see figs. 2.6 to 2.8)

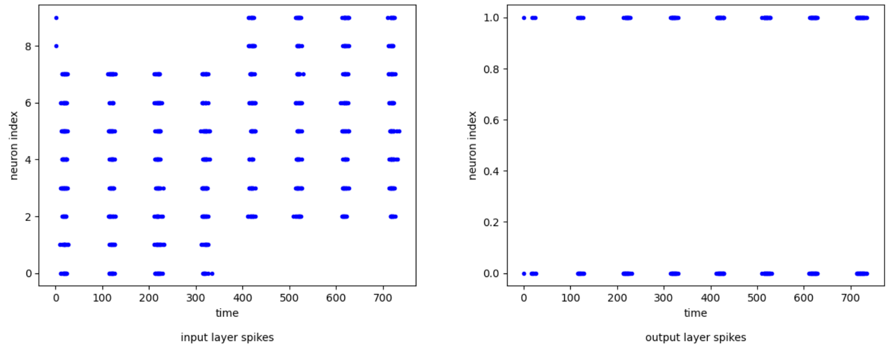

Figure 1.13: Here we have increased the number of intersecting neurons in the given patterns, but we have a low current-coefficient value.

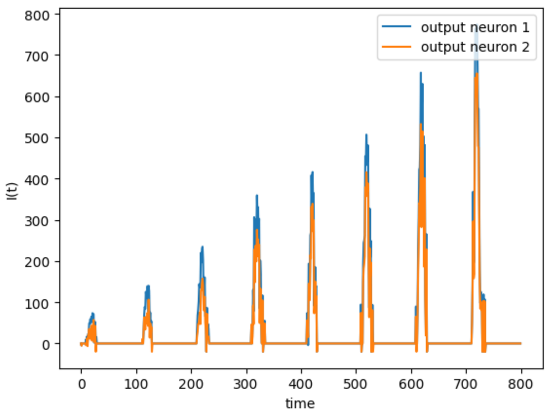

Figure 1.14: Both output neurons learn anyway!

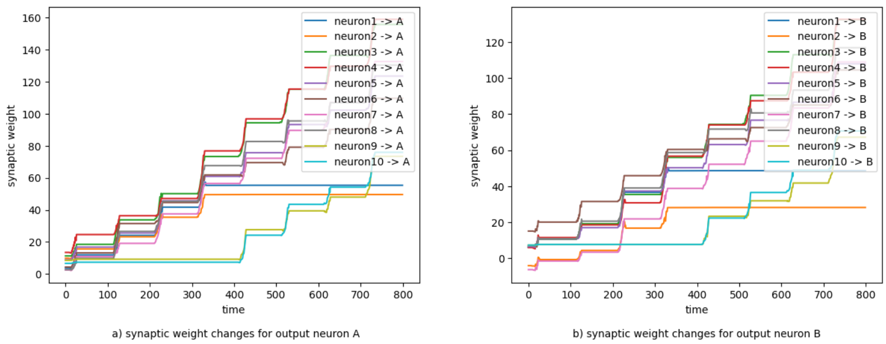

Figure 1.15: Synaptic weight changes for each output neuron (related to Figure 2.3)

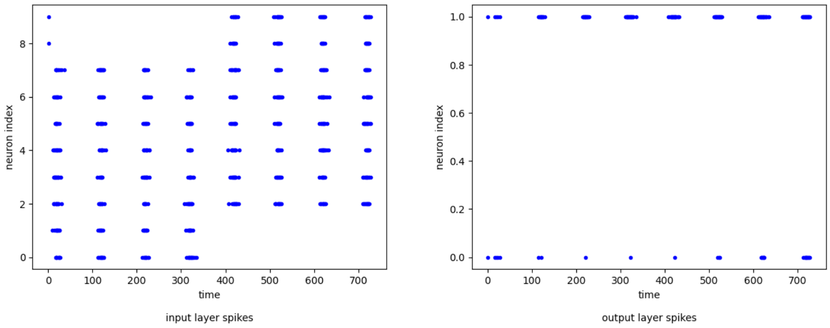

Figure 1.16: Here we have increased the number of intersecting neurons in the given patterns, and we have a high current-coefficient value.

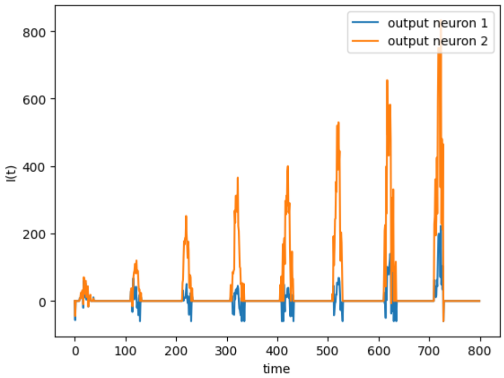

Figure 1.17: We can see the results which we discussed in the second case.

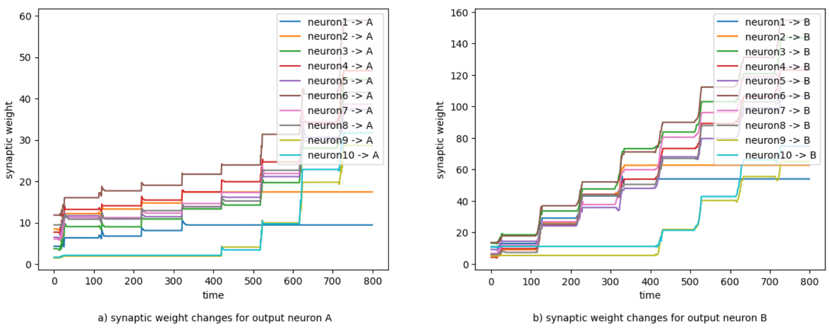

Figure 1.18: Synaptic weight changes for each output neuron (related to Figure 2.6)

Chapter 2

k-Winners-Take-All

*Note: In this report we always consider $k = 1$.

2.1 Experiments and Analyses Adding k-Winners-Take-All Mechanism

In this section, we are going to add k-Winner-Take-All mechanism to the output layer and analyse what happens (We also keep the lateral inhibition mechanism in the output layer; And we use the STDP learning rule just as we used in the previous chapter.). By adding k-Winners-Take-All mechanism to our output layer, we expect our network to learn faster and sharper and we can figure this out due to the number of times that we repeat the experiments to get the expected results.

First, we give two totally distinct patterns to our network:

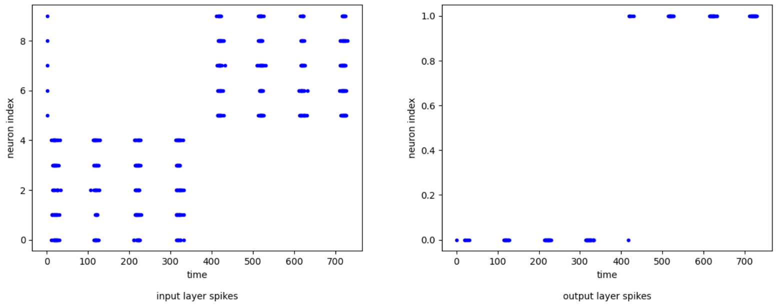

Figure 2.1: Given are two distinct input patterns, and we can see that each output neuron learns one of them. We can see that while one pattern is getting learned by one of the output neurons, the other output neuron's activity does NOT get updated anymore and it doesn't get activated anymore. This is a result of ONE-Winner-Takes-All mechanism that we used in this experiment.

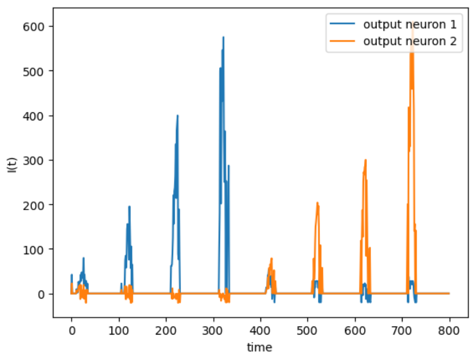

Figure 2.2: We can see that for each input pattern one of the output neurons is learning it while the other one is getting so strongly inhibited.

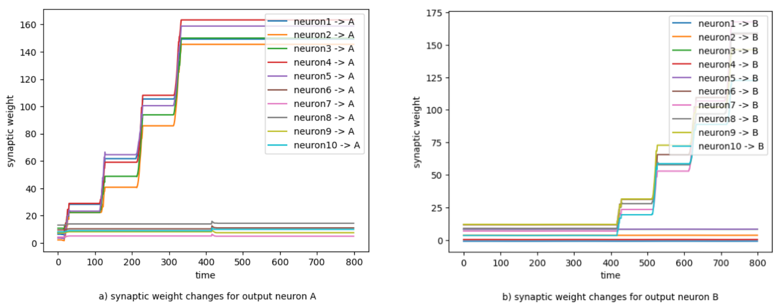

Figure 2.3: Synaptic weight changes for each output neuron

Now we increase the current-coefficient value and analyse the results:

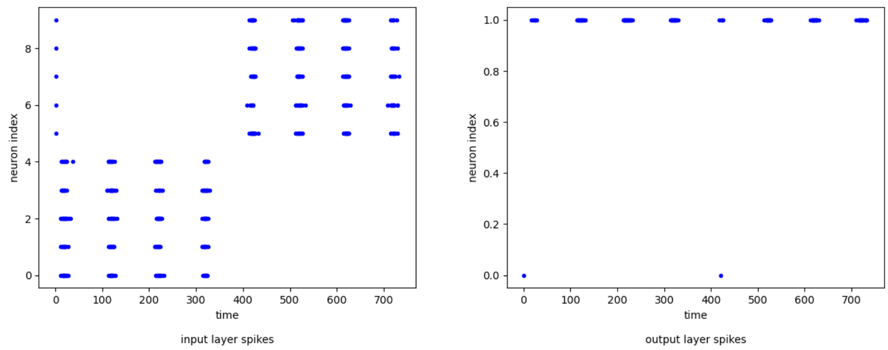

Figure 2.4: Given are two distinct input patterns, and this time we increased the current-coefficient value of the lateral inhibition. We can see that while the winning neurons are retained, losing neurons are getting extremely inhibited so that they can't get activated anymore and the winner can not ever get changed. So the first winner pattern is winning all the time and getting learned by one of the output neurons, and the other output neuron shuts down permanently.

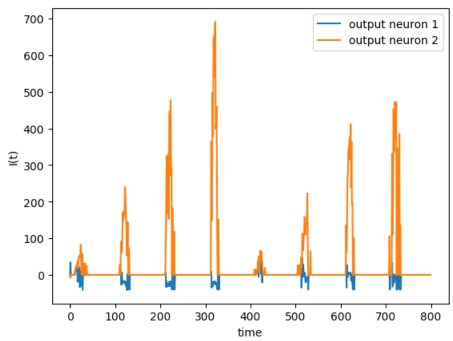

Figure 2.5: We can see the results we discussed in Figure 2.4

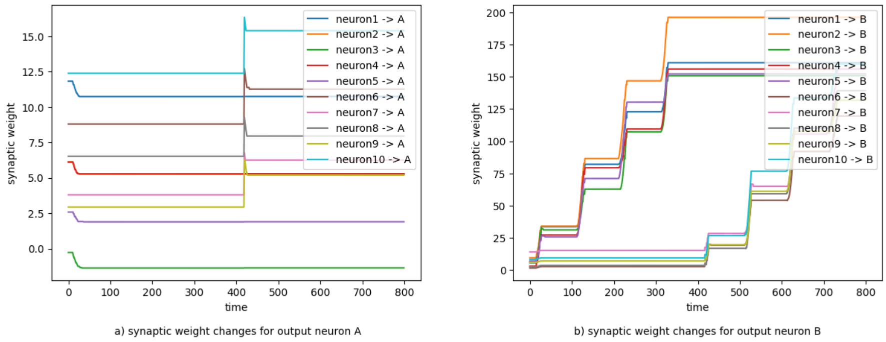

Figure 2.6: Synaptic weight changes for each output neuron (related to Figure 2.4); We can see that synaptic weights related to the loser output neuron (output neuron A) doesn't get changed/updated.

2.2 Adding Intersecting Neurons to Patterns

Now, let's hold some common neurons in our patterns and analyse what happens in such case:

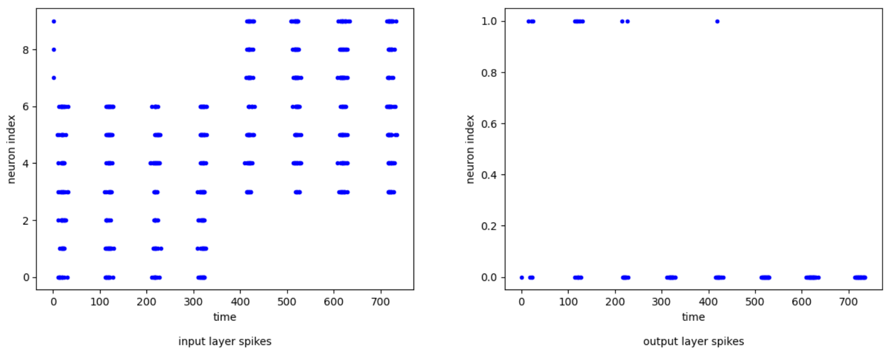

Figure 2.7: Given are two input patterns which have intersecting neurons, and we can see that once the winner stands out, first the other output neuron is also likely to start the learning process (because of the existence of intersecting neurons in patterns); But then as a result of One-Winner-Take-All (and also lateral inhibition), its activity starts to get inhibited over time.

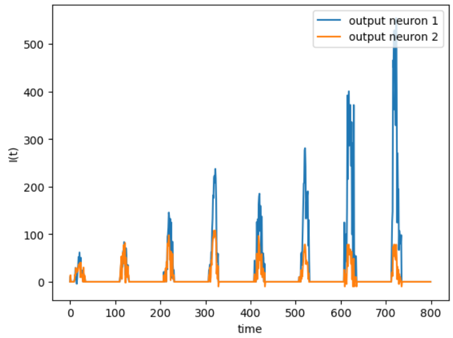

Figure 2.8: We can see the results we discussed in Figure 2.7

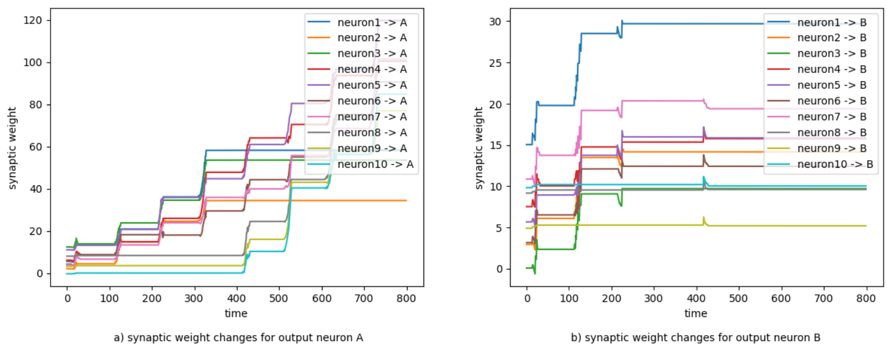

Figure 2.9: Synaptic weight changes for each output neuron (related to Figure 2.7)

Chapter 3

Homeostasis

3.1 Experiments and Analyses Adding Voltage-Base Homeostasis Mechanism

In this section, in addition to previous mechanisms, we add the voltage-base homeostasis mechanism to the output layer.
In our first experiment, we give 5 different patterns to our network while having 5 neurons in the output layer; We expect each pattern to get learned by exactly one output neuron. Note that since we use all lateral inhibition, k-Winner-Take-All and homeostasis mechanisms in our network, it's expected to be a strong network which is able to learn multiple patterns.

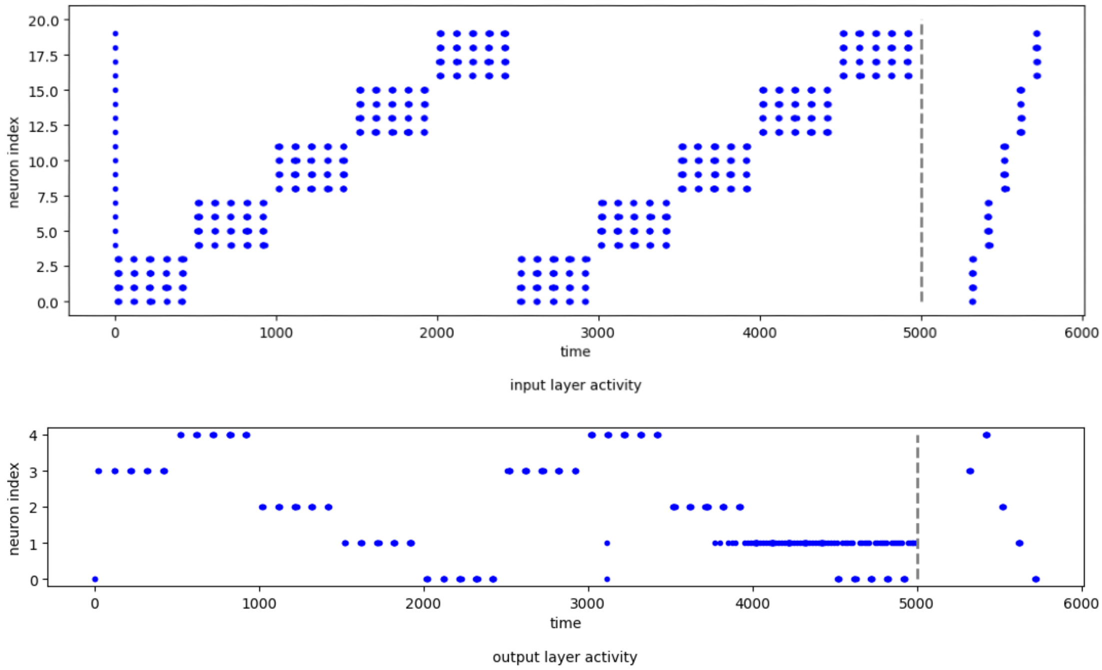

Figure 3.1: In this figure we can see the input patterns and the activity of output neurons; Clearly, each output neuron has learned exactly one pattern after training.

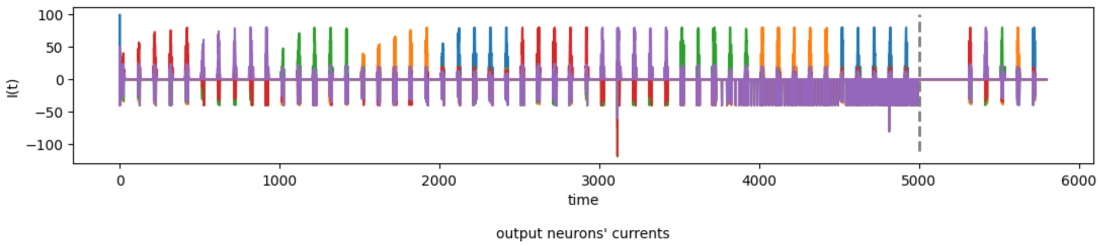

Figure 3.2: We can see that each output neuron has learned one pattern at the end of training.

Now, we are going to put 3 neurons in the output layer while giving 5 various patterns to our network! (Figure 3.4)

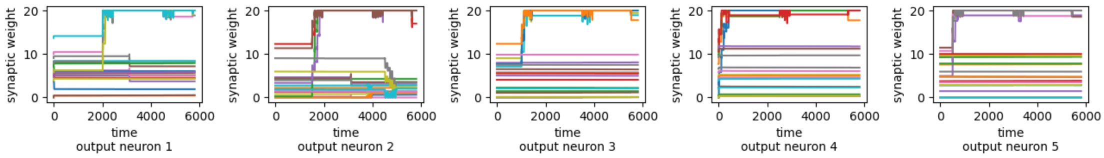

Figure 3.3: Synaptic weight changes for each output neuron (related to Figure 3.1)

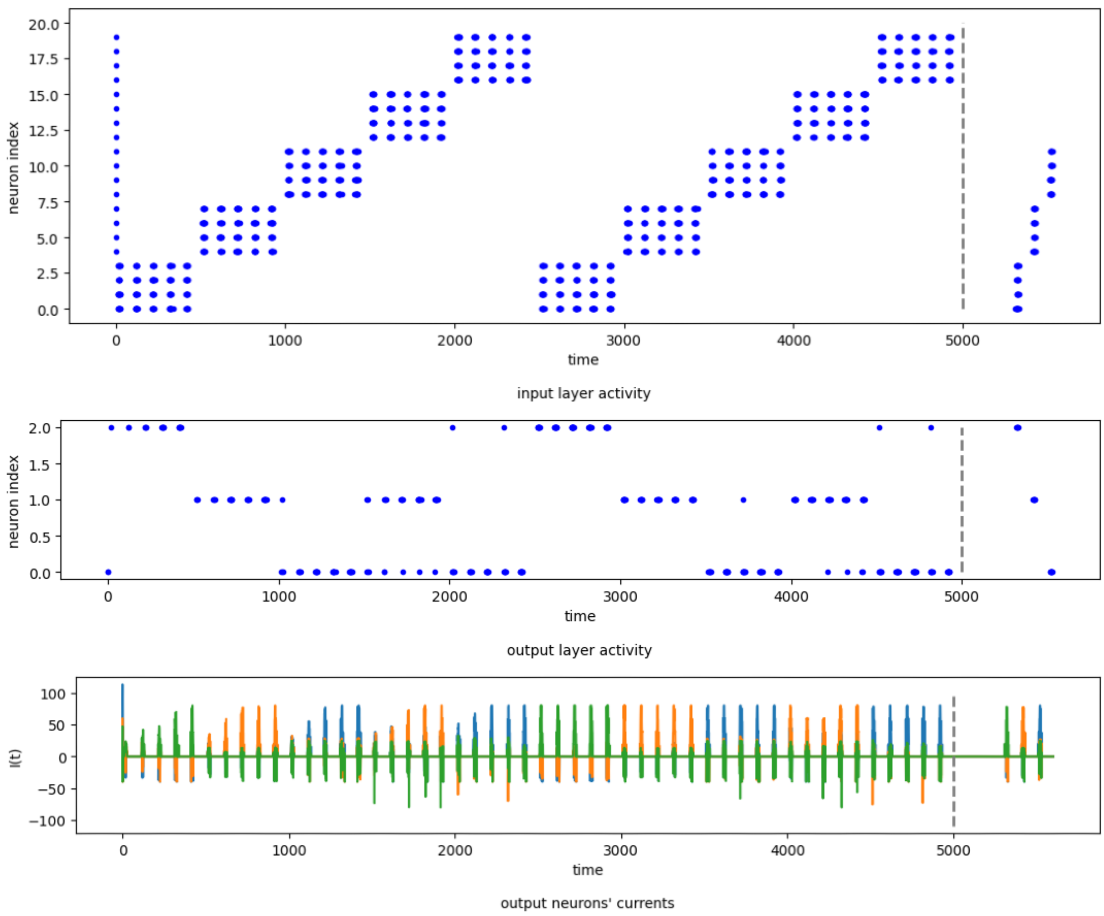

Figure 3.4: We can see when the number of patterns is more than the output layer neurons, some output neurons learn multiple patterns as a unit pattern which is a combination of that multiple patterns! This is amazing!

This time, let's put 7 neurons in the output layer while giving 5 various patterns to our network to see what happens in that case:

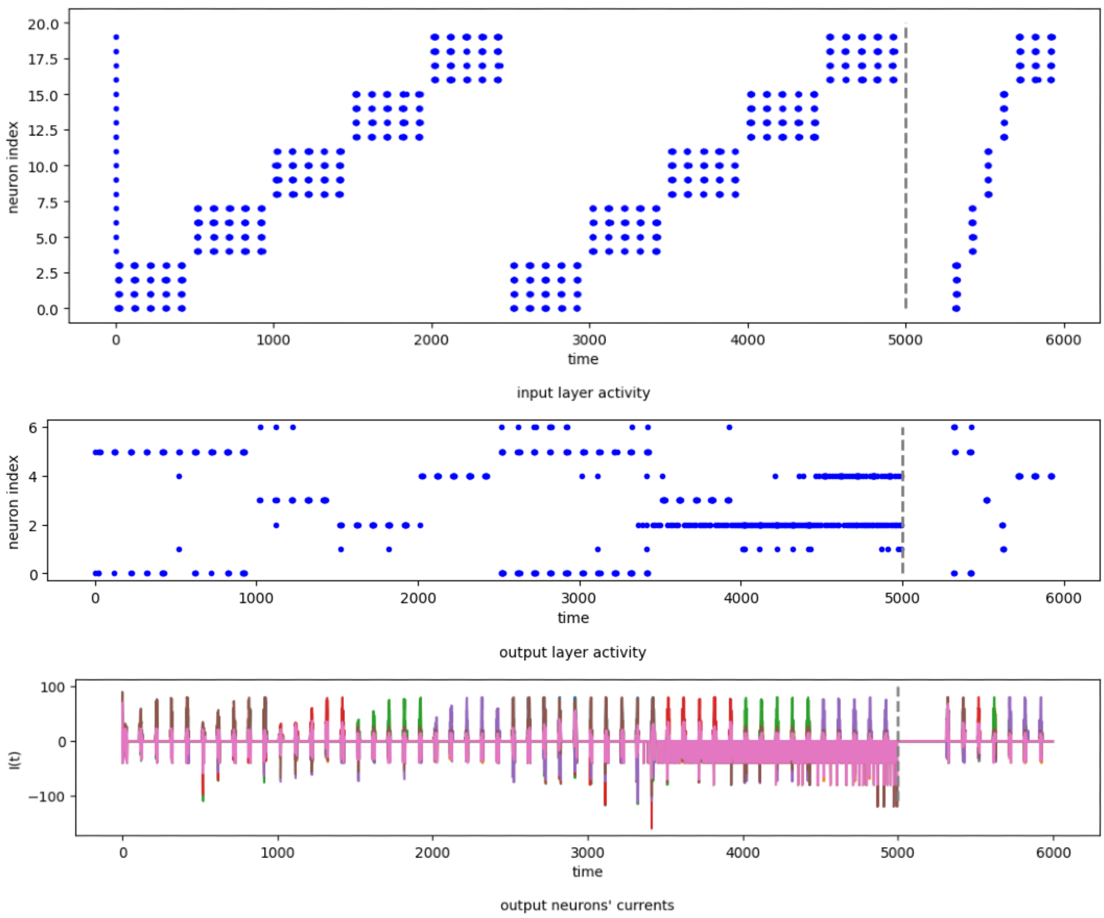

Figure 3.5: We can see when the number of output layer neurons is more than the patterns, some output neurons learn the same pattern and this leads to a higher overall activity in the network negating the effect of lateral inhibition and one-winner-take-all mechanisms.

3.2 Voltage-Base versus Activity-Base Homeostasis, and Experiments

*In the following experiments, we have 5 input patterns and 5 output neurons. Also : $u_{rest} = -65$, $u_{reset} = -73$, $threshold = -13$.

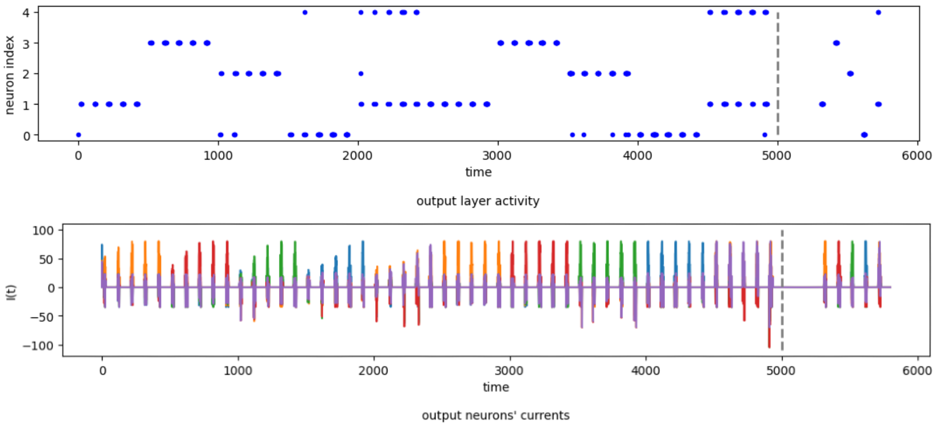

Figure 3.6: Experiment 1: voltage-base homeostasis, min voltage target=-50, max voltage target=-15, updating rate (eta-ip)=0.0003

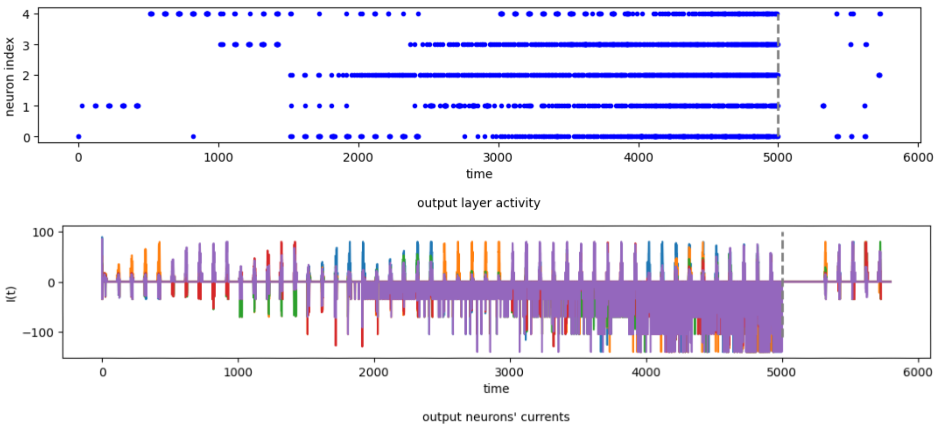

Figure 3.7: Experiment 2: voltage-base homeostasis, min voltage target=-30, max voltage target=-15, updating rate (eta-ip)=0.0003

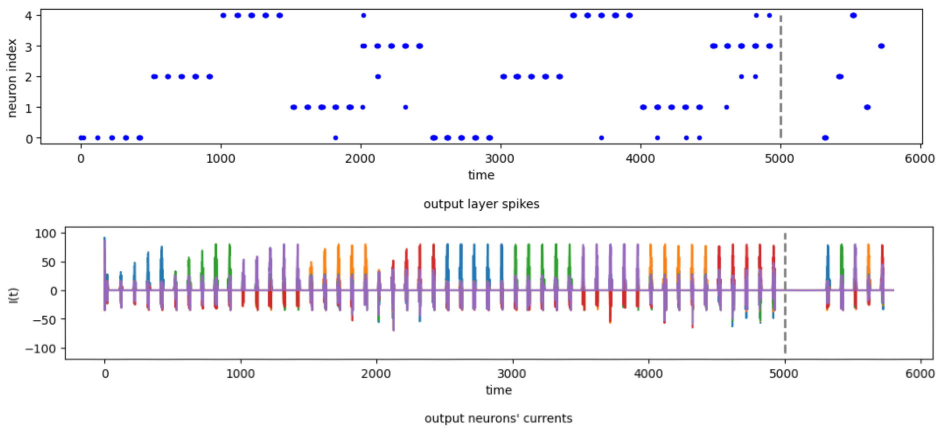

Figure 3.8: Experiment 3: voltage-base homeostasis, min voltage target=-30, max voltage target=-15, updating rate (eta-ip)=0.00003

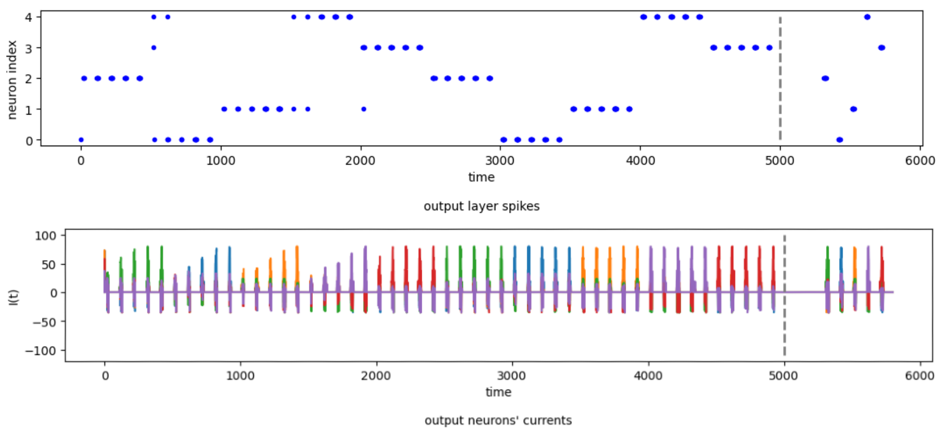

Figure 3.9: Experiment 4: activity-base homeostasis, window size=20, activity rate=10, updating rate=0.001

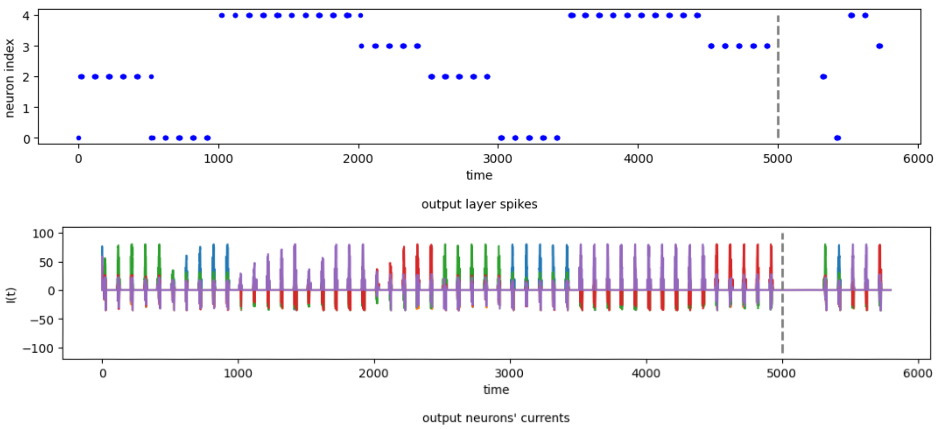

Figure 3.10: Experiment 5: activity-base homeostasis, window size=40, activity rate=10, updating rate=0.001

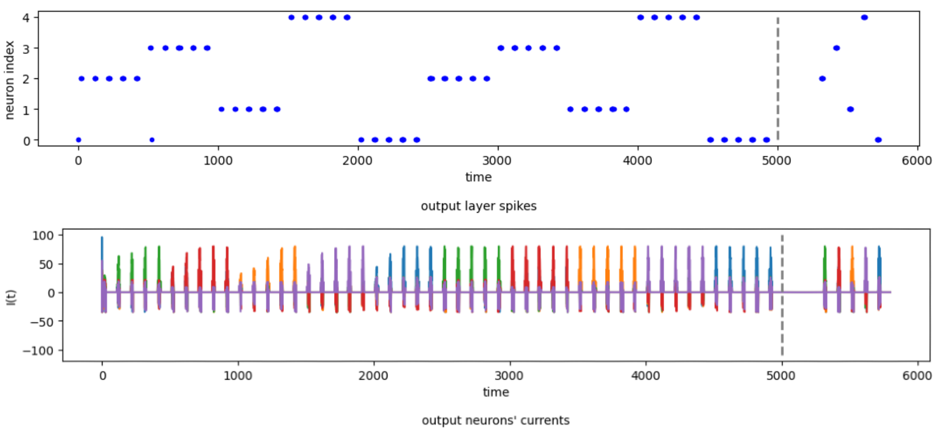

Figure 3.11: Experiment 6: activity-base homeostasis, window size=40, activity rate=10, updating rate=0.0001

At the end of the day, both voltage-base and activity-base homeostasis are demonstrating the same theme, but each of them using a different path. For example, we can see that if we have a situation that each output neuron does not learn one and only one pattern, like figs. 3.7 and 3.10, it can easily get fixed by adjusting the updating rate (figs. 3.8 and 3.11). However, in a brief overview, we can see that the activity-base homeostasis is more reasonable with handling the process of learning several patterns. It can be because of the exhaustion value which is used in the voltage-base homeostasis, which seems to be overreacting against tiny changes we make in the process.