Computational Cognitive Neuroscience
Assignment 1: Hopfield networks, Schizophrenia and the Izhikevich Neuron Model

Grigorios Sotiropoulos, 0563640

25th February 2010

1 Hopfield Attractor Network and Schizophrenia

General methods

The Hopfield network was implemented in MATLAB. The neurons in the network are indexed linearly, i.e. the states and energy levels of each neuron are stored in column vectors, as opposed to indexing by 2 variables that correspond to the x and y position of the neuron in the 10x10 arrangement. This was done for efficiency reasons, for example the various summations required are reduced to matrix multiplications. Code optimisation enabled a large number of repetitions of each simulation to complete within a reasonable time, leading to better estimates of average network performance under various conditions. In the next section, that deals with pruning, the linear index of each neuron is converted to a (x, y) pair prior to calculating distances between neurons.

Neuronal states are updated cyclically but asynchronously until convergence is achieved. Cyclical asynchronous update means that each neuron is updated in turn, ordered by its (linear) index in the network, with only one neuron being updated at each time step. This is not to be confused with what [3] refer to as cyclical update in a footnote: their version is (or seems to be) synchronous update, whereby the new state is calculated for all neurons (according to their current states) and then all neuronal states are updated "at once". This is quite different from asynchronous updating (whether cyclical or random). In contrast, the only difference between the update method I used and the asynchronous update originally described by Hopfield [4] is that neurons are picked in turn in the former case but randomly in the latter. A cyclical update may be less plausible biologically but does not affect the end results; it does, however, result in somewhat faster convergence in the computer model.

1.1 Hopfield network

Experiment 1: storing 3 memories

The 3 memories were generated randomly and stored in the network. Degraded versions of these memories, at 4 chosen degradation levels, 20% (low noise), 33% (moderate noise), 40% (high noise) and 45% (very high noise), were subsequently generated. Degradation was performed by flipping 20, 33, 40 or 45 randomly selected states in each input. At each degradation level, the simulation was run 5000 times. The network (i.e. the connection weights) remained the same across runs; what changed was the degraded inputs, which were randomly generated (at the specified degradation level) in each of the 5000 runs. The performance of the network was then averaged over all runs, for each degradation level. The performance metric here is recall, i.e. the proportion of degraded inputs that resulted in full recovery of the target memory.

Figure 1 shows the results. The network was able to recover all inputs degraded by 20% and the vast majority of inputs degraded by 33%. As degradation increases and approaches 50%[1], the ability of the network to recover full memories diminishes significantly, dropping to about 0.4 at 45% degradation.

[1] (50% is the maximum level of degradation, as anything beyond 50% is closer to the complement of the memory, which is also stored in the network due to the symmetric connections.

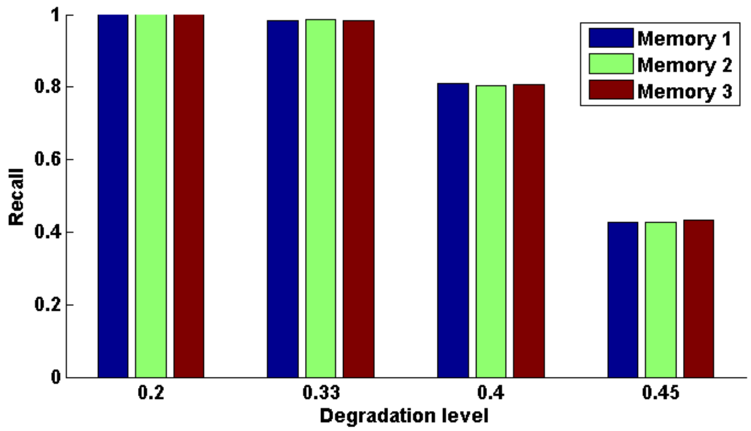

Figure 1: Performance of a 100-unit network in recovering each of the 3 stored memories from degraded inputs, for each degradation level tested.

Experiment 2: performance, capacity and size correlations

This experiment examines how the number of stored memories affects performance and how the size of the network affects its capacity.

For the former question, a network of 100 neurons was trained on varying numbers of memories and was subsequently tested with inputs of varying degrees of degradation, including zero degradation, which served as a "control": if a memory cannot be recalled from intact inputs (that is, if a network presented with an input identical to one of the stored memories does not remain in that state) then we can safely conclude that the network's capacity has been exceeded. In other words, testing with intact inputs provides an upper bound on the network's capacity. For each degradation level and for each number of memories (M) stored, 5000 simulations were performed to obtain reliable averages and reveal subtle effects.

For the second part of the experiment, capacity was defined as the maximum number of memories that can be stored in the network such that the network is able to retrieve near-full (up to 2 corrupted bits) memories from degraded inputs 75% of the times. That is, the network's capacity is exceeded once it becomes unable to successfully recall a memory on more than 25% of the inputs. Of course, for a given number of stored memories, the proportion of inputs that can be correctly recognised depends on how severely degraded those inputs are. Thus the capacity, as a function of network size, was measured for 4 different degradation levels, the first being the control condition of 0% degradation.

The methodology was similar to the previous experiment, with two differences: here, a different set of memories was randomly generated in each simulation (as opposed to fixing the M memories and only varying the degraded inputs, as in the previous section). This was done to eliminate the possibility of obtaining results that are specific to the chosen memories.

Figure 2, A shows that as the number of stored memories increases, the performance of the network decreases sublinearly at first and then approximately linearly (with a superlinear component at large numbers of memories). Overall the curves look sigmoid and symmetrical about the point of median performance, resembling psychometric curves. As expected, performance also drops when input degradation increases.

Figure 2, B shows that the capacity of the network is approximately linear in network size. A possible informal explanation of this observation is as follows: the size of each memory (number of elements) is exactly the network size N, thus the amount of information ("bits") contained in each memory is linear in N. However, information in the network is stored in its connection weights, and since there are $N(N-1)$ connections, the amount of information that can be stored in a network is quadratic in N. Therefore the number of memories that can be stored is linear in N. Hopfield has found through simulations [4] that the capacity of the network is approximately $0.15N$. Since then, it has been shown that the capacity of the Hopfield network is $\frac{N}{2\log N}$ if full recall is required (no corrupted bits) [7] and $0.15N$ if some corruption is permitted [1]. The results of my simulation for are fairly consistent with these findings when the network is evaluated against mildly corrupted inputs (0-20%).

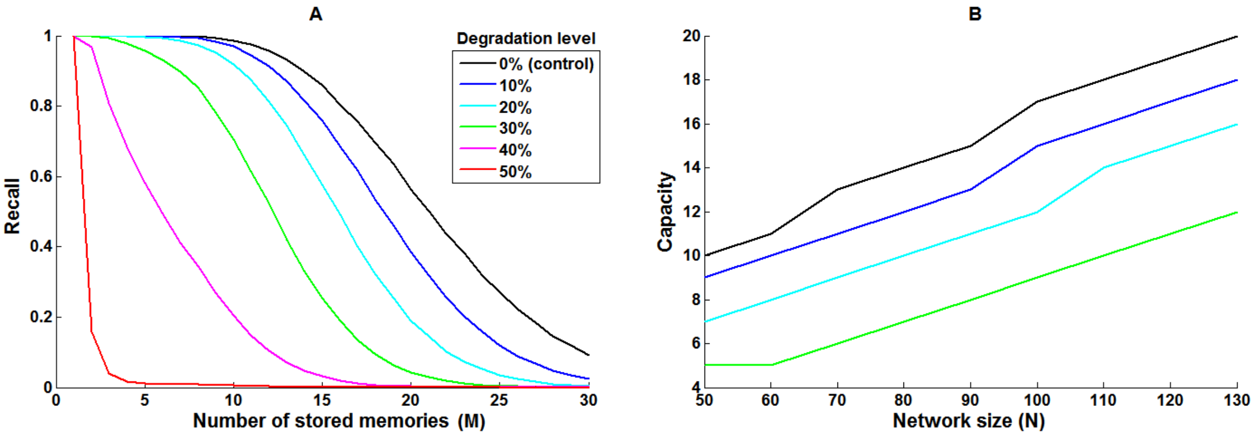

Figure 2: **A**, Network performance (recall, i.e. proportion of inputs that result in recovery of the correct memory) as function of stored memories in a network of 100 units. **B**, Network capacity (number of memories that can be stored and later retrieved correctly from partial inputs in 75% of the times) as a function of size (number of units).

1.2 Cortical Pruning and the development of Schizophrenia

For this section, the previously described network of 100 units was modified to include a stochastic element during state update, as described in [3]. During preliminary testing, a temperature of $T = 4$ resulted in network performance almost identical to the deterministic-update version of the previous subsection, as Figure 3 shows. As before, performance was averaged over 1000 runs of the experiment, each time with different, randomly generated memories.

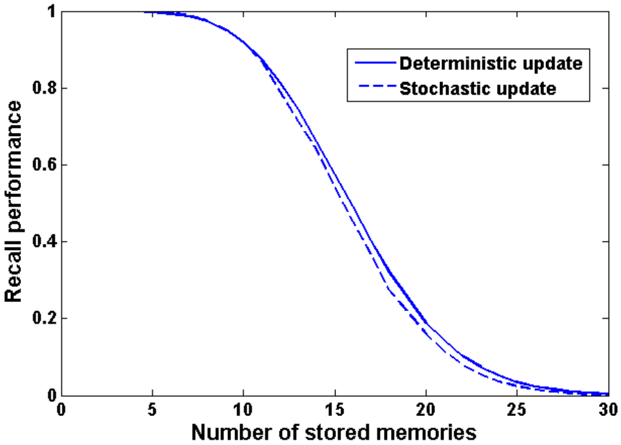

Figure 3: Preliminary testing of the introduction of stochasticity in updating of the network state. Performance (recall) is plotted as a function of number of stored memories at a degradation level of 20% (see Figure 2, A and text).

To quantify the effects of axonal pruning on the behaviour of the network, a series of simulations where performed; in each simulation, a different set of 9 memories were randomly generated and stored in the network. Axonal pruning was then performed, using values of pruning coefficient $\hat{p}$ ranging from 0.5 to 1 in steps of 0.1. Following pruning, the network was tested on degraded inputs, using two levels of degradation, 20% and 33%, as in [3]. Thus there was a total of $6 \times 2 = 12$ simulations, each repeated 1000 times to obtain averages. This is in contrast to the methodology in [3], where results are reported only for a single set of 9 memories and 18 degraded inputs (9 inputs for each of the 2 degradation levels)[2].

Figure 4, A shows the performance of the network as a function of axonal wiring reduction (AWR), which is defined as the combined length of pruned axons divided by the total axonal length in the (unpruned) network. AWR is a monotonic function of the pruning coefficient $\hat{p}$, with values of $\hat{p}$ in the range $[0.5, 1]$ corresponding to AWR values in the range $[0.67, 0.91]$. AWR was used instead of $\hat{p}$ in order to allow direct comparison with the results of Hoffman and Dobscha (Figure 4, B - adapted from [3]). Performance is quantified by the proportion of network responses that were successes (where success is defined as either an end state with a Hamming distance of no more than 3 from the target state or as a "generalisation") as well as the proportion of responses that were "loose associations" (defined as end states more than 10 Hamming units (HU) away from the target that were not generalisations). Generalisations are end states that are more than 10 HU away from the target but are closest to the two memories that the input state was closest.[3].

It can be seen that in both simulations, as AWR increases, the number of successful runs decreases whereas the number of loose associations increases. The overall levels of these two performances indices are different for the two degradation levels, as one would expect, however my simulations show that the rates of these changes are independent of the degradation level. Furthermore, Hoffman and Dobscha report a threshold-like behaviour, where network performance is quite good until an AWR of 80%, after which performance drops rapidly. In my simulation, no such threshold-like effects are evident, although the slopes of all curves become progressively steeper past the 80% mark. Finally, the authors note that for the moderately degraded inputs (20 HU), the number of loose associations remains low even past the threshold (even though the number of successful runs decreases). This is not the case in my simulation.

All of these differences likely stem from the fact that my simulations were repeated 1000 times and thus were able to reveal the average behaviour of the network, whereas the authors report results from a single set of simulations: it is possible that with another set of simulations (using different memories) the authors would have obtained quantitatively different results.

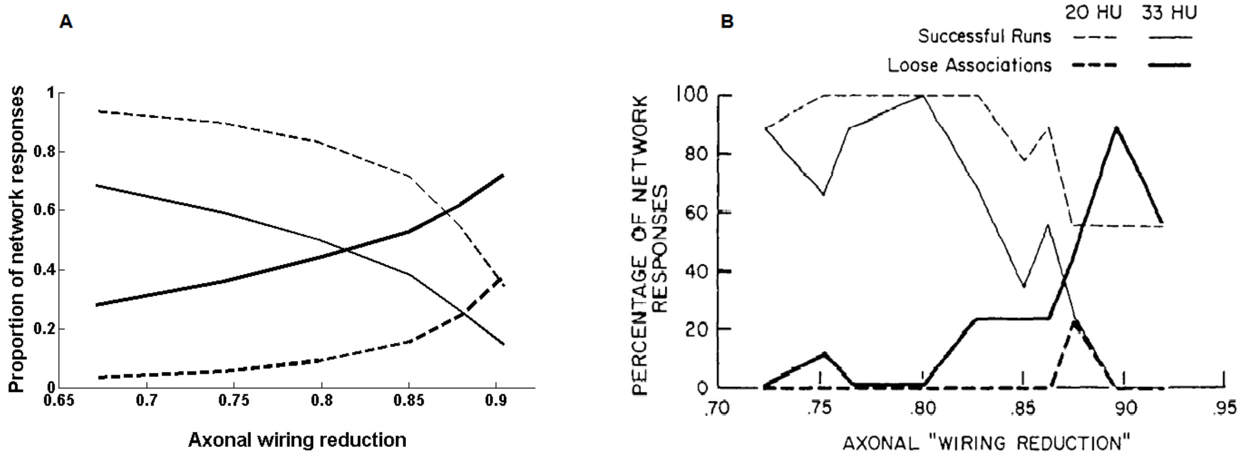

Figure 4: Network performance as a function of axonal wiring reduction (AWR). **A**, results from present simulations. **B**, results from a single simulation by Hoffman and Dobscha (adapted from [3]).

Discussion

Hoffman and Dobscha [3] argue that the formation of parasitic foci in overpruned regions of the network could account for the positive symptoms of schizophrenia. These parasitic foci are persistent and are not parts of any memory. If axonal pruning occurs in the frontal cortex, such parasitic foci could manifest as intruding thoughts that are out of the patient's control. While this could in principle account for experiences that are not based on reality and are incoherent with other thoughts, not all Schneiderian symptoms fit this description. For example, it is difficult to see how thought broadcasting and thought insertion could occur via

[2] The authors focused on a single case as they had to show specific patterns that developed due to pruning, such as parasitic foci. Studying this kind of effects is of course only possible on a case-by-case basis.

such a mechanism: these are primary (first-rank) symptoms of schizophrenia and not secondary explanations of delusional nature. While the model could account for the formation of bizarre thoughts, it cannot explain why people believe that these thoughts were inserted to them. A unified treatment of Schneiderian symptoms is therefore beyond the capabilities, and indeed the scope, of this model.

Hallucinations seem to be accounted for by the model in a more convincing way. Considering that each unit in the network codes for a feature, an overpruned cortical region could, according to the model, produce nonmemory parasitic foci, i.e. combinations of features. If these features happen to be sensory (e.g. visual), it is easy to see how hallucinations could occur. Such combinations of features do not correspond to an actual memory and may not even be plausible (realistic), which is typical of hallucinations.

As for delusions, the authors suggest that large parasitic foci control the state of most other neurons in the network by locking into activation patterns that are independent of input. Delusions could then be a result of the patterns taking control of belief orientations. It is not obvious to me how an associative memory population that serves as a model of working memory could interact with the populations dealing with beliefs. Such a mechanism, if it exists, is once again beyond the scope of the current model.

Pruning is not the only way to degrade the performance of the Hopfield network. It is well-known that by making the connections asymmetric (i.e. $w_{i \to j} \neq w_{j \to i}$) convergence of the network can no longer be guaranteed in general (although it has been proven than a Hopfield network with nonnegative asymmetric weights can be stable for any input [6]). As for dopamine neuromodulation and the effects of stress, these are not processes that can be directly incorporated to the model. The Hopfield network, even in its continuous form, is firing rate based. To model the effects of dopamine neuromodulation directly, spiking neuron models are necessary that are rich enough to model the dynamics of various synapses and the effects of the different receptors, such the D1 and D2 dopaminergic receptors, thought to play an important role in the symptomatology of schizophrenia (see [8] for a review).

2 Neuron Model of Izhikevich

The neuron model of Izhikevich was implemented as described in [5] and after studying the code samples available at www.izhikevich.com. Figure 5 shows the results of my simulations. All dynamic modes described in Part III of [5] were found, with the exception of the resonator (RZ); it proved impossible to produce anything similar to the pattern described in [5]. Incidentally, the resonator is used twice in code samples published by Izhikevich however with different parameters. None of the two published parameter sets was able to reproduce the reported pattern in my simulation. The parameters used in my simulations to produce all 8 patterns are listed in Table 2. In all cases except TC (both) and RZ, a step current of $10nA$ with a $10ms$ onset was applied (and held until the end of the trial). Between $0 - 10ms$ the current was zero. In the first case of TC, the step current was $3nA$ whereas in the second, the current between $0 - 10ms$ was $-30nA$ in order to simulate the hyperpolarisation ($-90mV$) that is required to induce the short burst when the current is set to 0 at $10ms$ (and for the duration of the trial). In the case of RZ, the present result was obtained by inducing two short current pulses on top of the step current applied in all other cases.

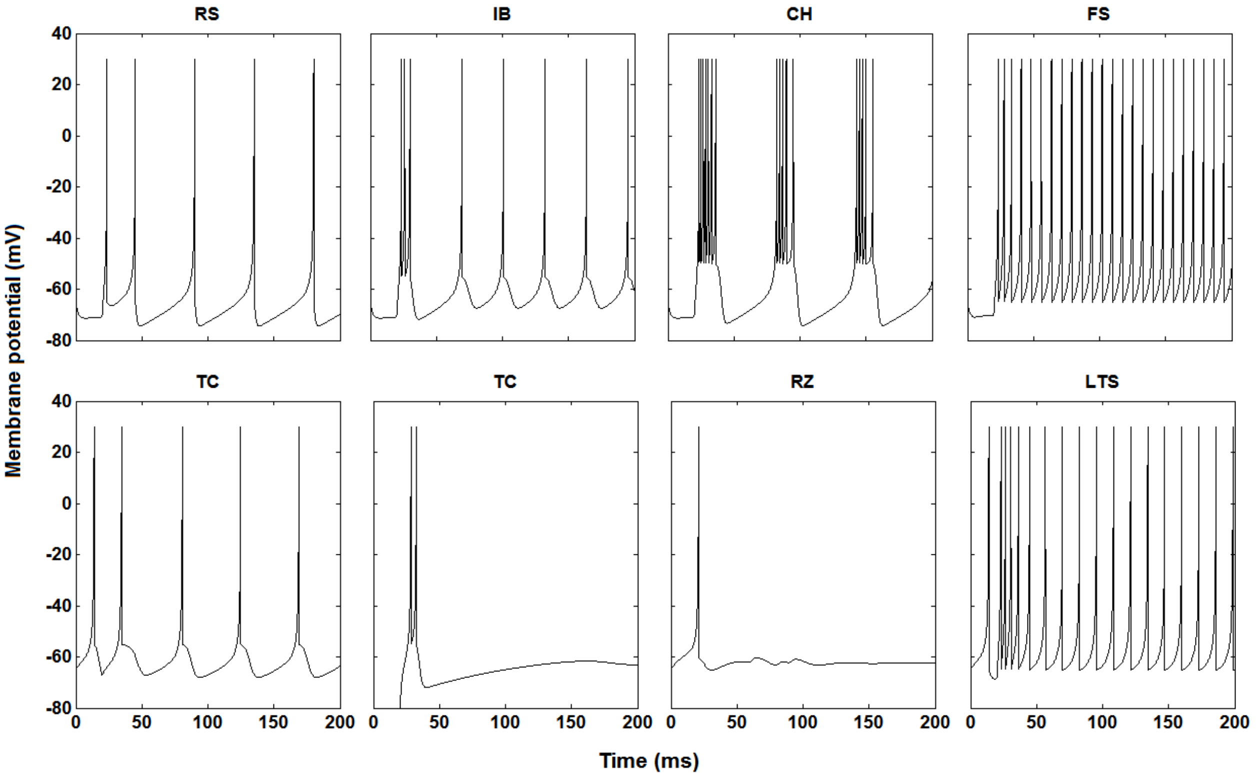

Figure 5: Dynamic modes of the Izhikevich model, corresponding to the ones described in part III of [5].

a	b	c	d	
0.02	0.20	-65	8	**RS**
0.02	0.20	-55	4	**IB**
0.02	0.20	-50	2	**CH**
0.10	0.20	-65	2	**FS**
0.02	0.26	-55	3	**TC**
0.10	0.26	-60	-1	**RZ**
0.02	0.26	-65	2	**LTS**

Table 2: Parameter values for the different dynamic modes described in Part III of [5].

3 Spikes, Attractors, and Schizophrenia

Izhikevich [5] states that by modifying the thalamic input and synaptic strenghts (connection weights) in the population of neurons, it is possible to obtain various dynamic modes. In the example code in , these parameters are hardcoded (i.e. they are not represented by variables).

The connection weights are represented by the following line of code (taken directly from [5]):

```
S=[0.5*rand(Ne+Ni,Ne),−rand(Ne+Ni,Ni)];
```

and the aforementioned hardcoded parameters are the factors 0.5 and -1 that control the mean of the random variables representing the excitatory and inhibitory (respectively) connection weights. These parameters are hereafter referred to as $\langle w_e \rangle$ and $\langle w_i \rangle$.

Likewise, the thalamic input is represented by the following line of code:

```
I=[5*randn(Ne,1);2*randn(Ni,1)];
```

The hardcoded parameters are the factors 5 and 2 that specify the variance of the zero-mean, normally distributed excitatory and inhibitory (respectively) input currents. These parameters are hereafter referred to as σ_e and σ_i.

Figure shows raster plots of the activity of a population consisting of 800 excitatory and 200 inhibitory neurons, for selected values of the 4 parameters.

The first row shows what happens when the variance of the excitatory input is increased 2, 3 or 4 times from the hardcoded value: in all 3 cases, the network exhibits high, synchronised activity that lasts about $20ms$, followed by a longer period of relative silence, after which the inhibitory population shows high-frequency rhythmic activity (from about 100 to $160Hz$). In the brain, such activity could account for gamma waves, at least for the first case of $\sigma_e = 10$, as for higher values the oscillation frequency is higher than then typical upper limit of gamma waves ($100Hz$). The excitatory population shows some synchronised rhythmic activity at first but as $\sigma_e$increases, the excitatory firings look more chaotic.

The second row shows what happens when the mean excitatory weight is increased to 1, becoming equal to the mean inhibitory weight. In this case, when σ_e is smaller than about 3.335 nothing interesting happens (population activity looks random). However, above that threshold, the network exhibits remarkable synchronisation, with very defined alternating periods of silence and activity. During the silent period, the excitatory population is almost completely inactive, while the inhibitory one shows moderate (unsynchronised) activity. Towards the end of the silent period, the excitatory activity starts to increase, which results in an explosive increase in activity of the entire network that lasts about $50ms$. Such activity could give rise to low-frequency ($3 - 5Hz$ in the present examples), high-amplitude waves, similar to delta waves.

The third row shows what happens when $\langle w_e \rangle$ and $\langle w_i \rangle$ are both increased. Unlike in the first two rows, the insets in the third row result from 3 runs of the same experiment (using the same parameters). The reason for this is that this mode was highly variable; raster plots of activity significantly across different runs of the same simulation, unlike in the previous two cases, where the raster plots were very similar across runs and thus are illustrated only once per parameter set. What we see in these plots are extended periods of very high global activity interrupted by periods where excitatory activity is absent except for a few (1-10) excitatory neurons that sustain activity during this period and sometimes even until the next silent period. During these silent periods, inhibitory neurons exhibit high-frequency synchronised activity, directly proportional to the number of excitatory neurons with sustained activity. Such a pattern is unstable and I am not aware of any similar pattern of activity, pathological or not, observed in the brain; it is interesting nonetheless.

Perhaps the most instructive result of this simulation was the fact that the discovered dynamic modes are quite robust: they depend on statistical properties of the network parameters and not on their exact values.

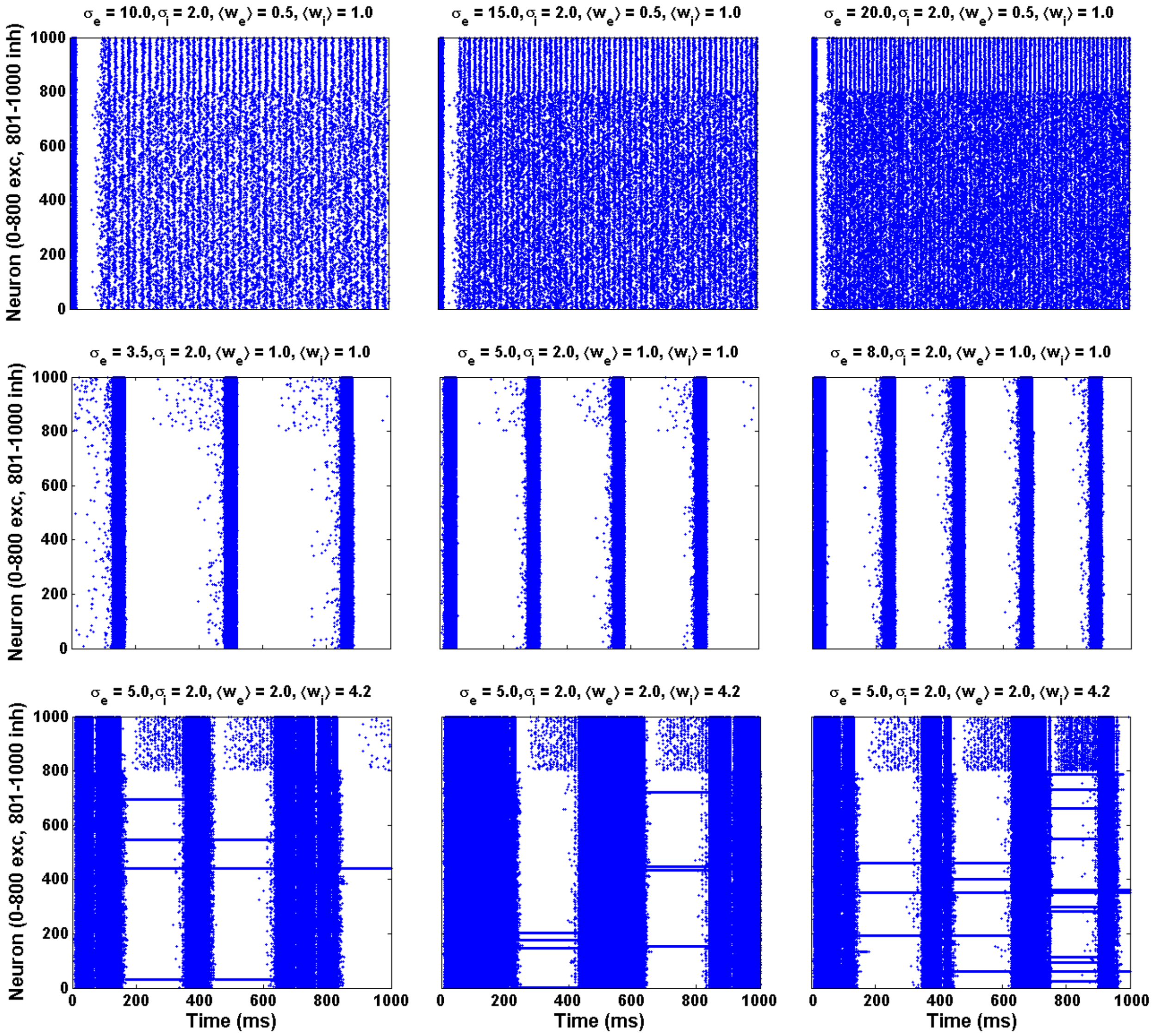

Figure 6: Raster plots of activity of 800 excitatory (bottom) and 200 inhibitory (top) neurons for selected values of mean connection weight and variance of the thalamic drive.

The model of Izhikevich could be used to implement more plausible attractor-based networks. Hopfield networks are point attractor networks; there are however (both theoretical and experimental) reasons to believe that other types of attractor networks are also present in the nervous system, such as line and plane attractors for oculomotor control [9], ring attractors as a model of the head direction system in rodents [10] and cyclic attractors as a model of oscillatory/repetitive movements, such as walking or swimming. A potentially interesting endeavour would be to incorporate Izhikevich's neuron model to the working memory model of Brunel and Wang [2], replacing their leaky integrate-and-fire neuronal model. Brunel and Wang study the effects of neuromodulation on the conductance of NMDA and GABA synapses but neuromodulation is added on top of the model, by simply changing the synaptic conductances in a similar manner as dopamine in the prefrontal cortex. As they themselves point out and as mentioned before, dopamine neuromodulation likely plays an important part in schizophrenia and thus directly simulating it in models may provide valuable insight into the related mechanisms.

References

[1] S. Amari and K. Maginu. Statistical neurodynamics of associative memory. *Neural Networks*, 1(1):63–73, 1988.

[2] N. Brunel and X.J. Wang. Effects of neuromodulation in a cortical network model of object working memory dominated by recurrent inhibition. *Journal of Computational Neuroscience*, 11(1):63–85, 2001.

[3] R.E. Hoffman and S.K. Dobscha. Cortical pruning and the development of schizophrenia: A computer model. *Schizophrenia Bulletin*, 15:477–490, 1989.

[4] J.J. Hopfield. Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the national academy of sciences*, 79(8):2554, 1982.

[5] E.M. Izhikevich. Simple model of spiking neurons. *IEEE Transactions on neural networks*, 14(6):1569–1572, 2003.

[6] J. Ma. The stability of the generalized Hopfield networks in randomly asynchronous mode. *Neural Networks*, 10(6):1109–1116, 1997.

[7] R. McEliece, E. Posner, E. Rodemich, and S. Venkatesh. The capacity of the Hopfield associative memory. *IEEE Transactions on Information Theory*, 33(4):461–482, 1987.

[8] E.T. Rolls, M. Loh, G. Deco, and G. Winterer. Computational models of schizophrenia and dopamine modulation in the prefrontal cortex. *Nature Reviews Neuroscience*, 9(9):696–709, 2008.

[9] H. Sebastian Seung. Continuous attractors and oculomotor control. *Neural Networks*, 11(7-8):1253–1258, 1998.

[10] K. Zhang. Representation of spatial orientation by the intrinsic dynamics of the head-direction cell ensemble: a theory. *Journal of Neuroscience*, 16(6):2112, 1996.