# Cell Assemblies and Neural Network Theory: From Correlators to Cell Assemblies

Christian R. Huyck

Middlesex University

c.huyck@mdx.ac.uk

**Abstract**

The Hebbian learning rule is the best model that we have for learning in biological neural systems. This paper points out that any reasonable variant of the Hebbian rule forces connections between neurons to become correlators of the firing of those neurons. That is, if neurons A and B fire together more frequently than neurons A and C, then the connection between A and B is greater than the connection between A and C. A new variant of the Hebbian rule is described and is shown to exhibit linear correlation values.

This learning rule is then used as a basis for forming Hebbian Cell Assemblies (CAs). It is shown how the learning rate and pattern presentation mechanism affects the variance of these weights. It is shown how spreading activation increases the correlation values, and how fatigue reduces the correlation values.

The compensatory learning rule is described. This variant of the Hebbian learning rule is used to enable CAs to form based on a variety of patterns. These simplified neurons, simplified axons, and idealised learning mechanisms are used to allow the formation of overlapping CAs. To the best of our knowledge, this is the first description of overlapping CAs being formed from valid physical neural models.

This information forms the basis of a formal explanation of how CAs are formed, and when they are formed. The paper concludes with a discussion of these issues and of future work.

## 1 Introduction and Background

The Cell Assembly (CA) is a central concept in computational neuropsychology and in cognitive science in general. The basic idea is that what we consider concepts are stored in the brain by reverberating neural circuits called CAs. CAs are made up of neurons that are connected by synapses with large strengths. The CA is activated by some of its neurons being fired. These neurons then cause other neurons in the CA to fire leading to a cascade. These neurons then form a reverberating circuit that enables the CA, and thus the concept, to remain active for longer than a single neuron could remain active. This report is based on work described in [11], further explanations of some aspects of CAs can be found in that report.

The CA is formed by strengthening existing connections. This is done by a form of Hebbian learning which may be manifested in the brain via long-term potentiation and depression.

There are a wide range of simulations of neurons (e.g. [3, 6, 2]). In this report we focus on a simple neural model, the spiking leaky integrator. This simple model may not be perfect [16], however it is clear that biological neurons can integrate activation from incoming synapses, that activation leaks away, and that they fire sending spikes to the synapses leading from it [5]. Consequently, any behaviour that can be generated from this type of model can be done in the brain.

Simulated networks of spiking leaky integrator neurons can learn CAs [11]. CAs are good because they have interesting computational properties and there is biological evidence that they exist [1, 7, 20]. Some computational properties include their ability to represent both short and long term memories. Short term memories are represented by a firing CA. A long term memory is represented by a change of synaptic weights that allows a stable state to form. This stable state or circuit is a CA.

Biological plausibility and five decades of theoretical exploration of CAs has lead to a relatively malleable theory. This theory started with [8], and has had major contributions from [17, 1, 20, 4] among many others. This report describes simulations based on the CANT model [12]; this model attempts to adhere closely to CA theory, and thus attempts to be biologically plausible.

In this report, we start with the simple assumption of firing neurons and Hebbian learning. We show how our particular variant of Hebbian learning leads to synapses being indicators of the likelihood the post-synaptic neuron fires when the pre-synaptic neuron fires. This is done both mathematically and via simulation evidence.

We then introduce the known neural properties of fatigue and spread of activation. We describe simulations that show how these properties effect the synaptic weights.

In the fourth section we discuss CA formation from these simple neurons. The compensatory learning rule is described and it is shown how this biologically plausible learning rule can be used to increase the range of patterns that can be classified. We introduce inhibitory neurons to allow CAs to compete. Prior work has shown how CAs can be formed with neurons participating in only one CA. The present work shows how neurons can participate in multiple CAs.

The fifth section is a discussion of the ramifications of these simulations. The final section is a conclusion and a discussion of future work.

## 2 Connections as Neuron Correlators

The Hebbian learning rule can be described as follows: if a node is connected to another node, and both are activated at a given time step then the strength of the connection is increased. This is shown in Equation 1:

$$if((F(i) and F(j)) then (w_{ij} = w_{ij} + R) \qquad Equation\ 1.$$

If nodes $i$ and $j$ fire, then the connection strength of the synapse is increased by some value $R$.

The logical extension of the Hebbian learning rule is that the weights correlate how often the post-synaptic neuron fires when the pre-synaptic neuron fires. If neuron $i$ is connected to neurons $j$, and $k$, then if $j$ fires more frequently than $k$ along with $i$, the synaptic weight $w_{ij}$ will be higher than $w_{ik}$.

### 2.1 Formal Correlation Values

The Hebbian learning rule can be used to correlate the firing of two neurons. In biological systems, the connections or synapses lead to the activation of other neurons. In simulations, this need not be the case. For the rest of this section, the connections merely state how likely it is that the post-connection neuron will fire if the pre-connection neuron fires. We will call this the correlation $L_{ij}$.

In our model, learning only occurs when the pre-connection neuron is fired. If the post-connection neuron also fires, the connection's weight is increased. If on the other hand it does not fire, the connection's weight is reduced using a parallel rule known as the anti-Hebbian learning rule. However, the amount that the weights change cannot be constants. If they were constant, the weight will either stay at or around zero (if the post-connection neuron fired less than half the time) or tend to infinity (if the post-connection neuron fired more than half the time) [13].

If we would like $w_{ij}$ to have the value of $L_{ij}$ it has to be between 0 and -1. So, the amount the weight decreases should keep the weight above 0. If decreasing never goes below 0, half of this constraint is met.

The weight can be kept above 0 by changing the weight by a constant multiplied by the weight as shown in Equation 2. The constant should be between 0 and -1 and is called the learning rate $R$. The equation is thus:

$$\Delta w_{ij} = w_{ij} * R \qquad Equation\ 2.$$

As the weight approaches 0, the amount it changes is also reduced. It will never go below 0.

The goal is to have the weight equal the correlation. If the weight is greater than correlation, it should tend to decrease, and if it is smaller it should tend to increase.

The weight will tend to increase if the amount reduced *f(x)*[1] times the frequency of reduction $L$ is less than the amount increased *g(x)* times the frequency of increase *1-L*. The weight will tend to decrease if $f(x) * L > g(x) * (1-L)$. If the correlation is high, there should be many small increases and few large decreases; if the correlation is low, there should be many small decreases and a few large increases.

The amount that the weight should be increased should balance the amount it is decreased. This balances when the increase is the amount of decrease *f(x)* times the number of reductions for each increase. The number of reductions per increase is $(1/L_{ij}) - 1$ yielding Equation 3:

$$\Delta w_{ij} = f(x) * ((1/L_{ij}) - 1) \qquad \textit{Equation } 3.$$

This all assumes that we have the correlation $L_{ij}$. However, we do have this value in the weight $w_{ij}$. Initially we do not know this value, but frequent applications of the learning rules makes $w_{ij}$ tend toward $L_{ij}$. If the $w_{ij}$ is greater than $L_{ij}$, it will be reduced because there will be fewer small increases than expected. Similarly if $w_{ij}$ is less than $L_{ij}$, it will be increased because there will be fewer small decreases than expected.

This rule leads to a function where $w_{ij} \simeq L_{ij}$. This rule could be modified to produce a range of monotonic functions. However, for this report we will continue using this linear learning rule.

For example, if a neuron starts with a weight of .3, and the learning rates are the high values of .1 for increase, and -.1 for decrease, Table 1 describes the results of a sample series of activations.

| *j* fires | $(1/L_{ij}) - 1$ | $\Delta w_{ij}$ | New Weight |
|---|---|---|---|
| fires | 2.3 | 0.23 | 0.53 |
| fires | 0.89 | 0.09 | 0.62 |
| no fire | not app. | 0.06 | 0.56 |
| fires | 0.79 | 0.08 | 0.64 |
| no fire | not app. | 0.06 | 0.58 |

**Table 1: Sample Connection Weights**

Note that the real correlational value from the data is 0.60, and the last four all are quite near to that.

---

[1] *f(x)* is the change $\Delta w_{ij}$ from Equation 2.

### 2.2 Actual Correlation Values

Using the CANT model, we have conducted several simulations. The topology of the CANT network is a 20 by 20 network of neurons. These neurons are connected in a distance biased fashion; i.e. a neuron is more likely to be connected to a neuron that is close to it than one that is far from it. The topology is toroidal meaning that the top row is adjacent to the bottom row as well as the second row, and the left column is adjacent to the right column as well as the second column. The toroidal topology is used to avoid boundary conditions.

All of the simulations in this report are presenting two types of patterns. The first type of pattern consists of the top eleven rows and the second pattern consists of the bottom eleven rows. This means that both patterns contain the middle two rows. An instance of a pattern is twenty of the possible 220 neurons.

This type of pattern allows a range of correlations and it allows neurons to participate in more than one CA. Indeed this overlapping pattern is more complex to learn than is a mutually exclusive pattern (see section 4).

#### 2.2.1 Correlation with .1 Learning Rate

In the first simulation, an instance of the top pattern is presented followed by an instance of the bottom pattern. Each presentation is one cycle. The connection weights are initially set to a random small weight. Table 2 describes the average connection weight of various types of connections.

The solid line refers to connections from a neuron in the top nine rows to a neuron in the top nine rows. The likelihood of the post-connection neuron being activated when the pre-connection neuron being activated is 19/219 or 0.0867. As predicted the average tends toward this. The chart only shows the top to top activation, but the bottom to bottom and middle to middle do the same thing. The dotted and dashed line shows the average weight of the connection from a neuron in the top to a neuron in the bottom. This quickly tends to 0.01 which is the minimum value of a connection in the current simulation. The actual correlation is zero because no top neuron fires simultaneously with a bottom neuron. Again, only top to bottom is shown, but bottom to top connections have a similar value. Finally, the dotted line shows the average weight of connections from the middle to the top; similar weights are obtained from the middle to the bottom. This weight should tend toward 19/399 or 0.0476 and it does tend toward this weight.

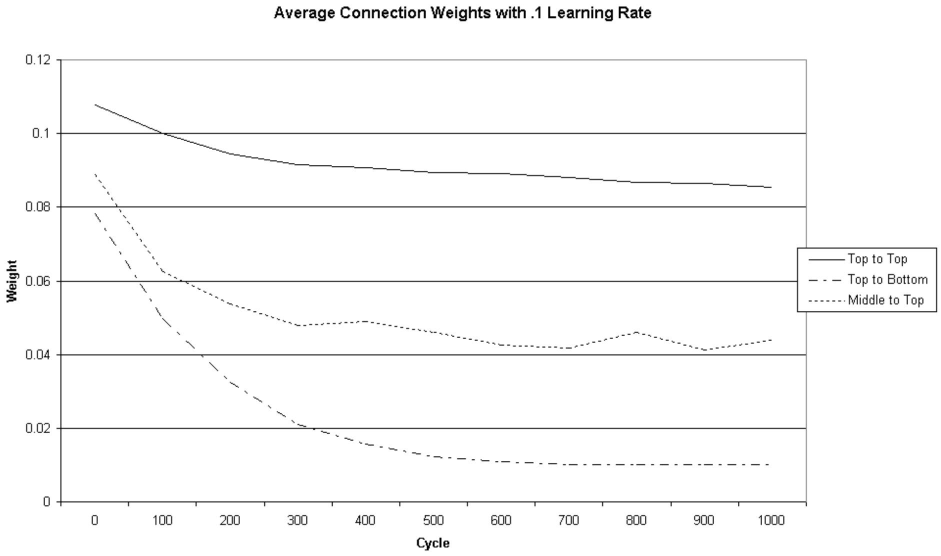

Table 2.

### 2.2.2 Correlation with .05 Learning Rate

The learning rate is important. There is stochastic variation within these runs. Note a brief rise in table 2 of the dotted line at cycle 800. This variance is also shown below in table 4. If the learning rate is low, the connection weights should converge more slowly, but there should be less variance. The above simulation used a learning rate of 0.1. The same simulation was rerun with a learning rate of 0.05. The average connection weights are described in table 3.

As predicted, the top to top connections tend toward 0.0867, the middle to top connections toward 0.0476 and the top to bottom connections toward 0.01. Again, the connections not described tend toward their appropriate values. Note however that the curve is much shallower with a learning rate of 0.05 than with 0.1. For instance, the top to bottom connections converge by 500 cycles with a higher learning rate, but have still not converged at 1000 cycles with the lower learning rate.

Table 4 describes the variance of connections from the top row to the top row. As can be seen, the variance is always higher for the higher learning rate. This is due to more of a change in weights with each application of the learning rule. Clearly there is a trade off between time to learn the weight and variance of the weight.

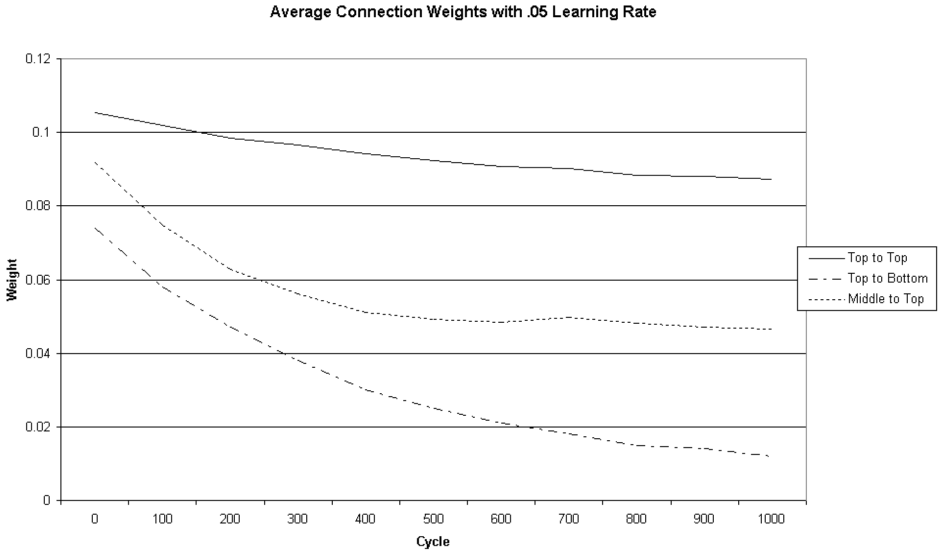

Table 3.

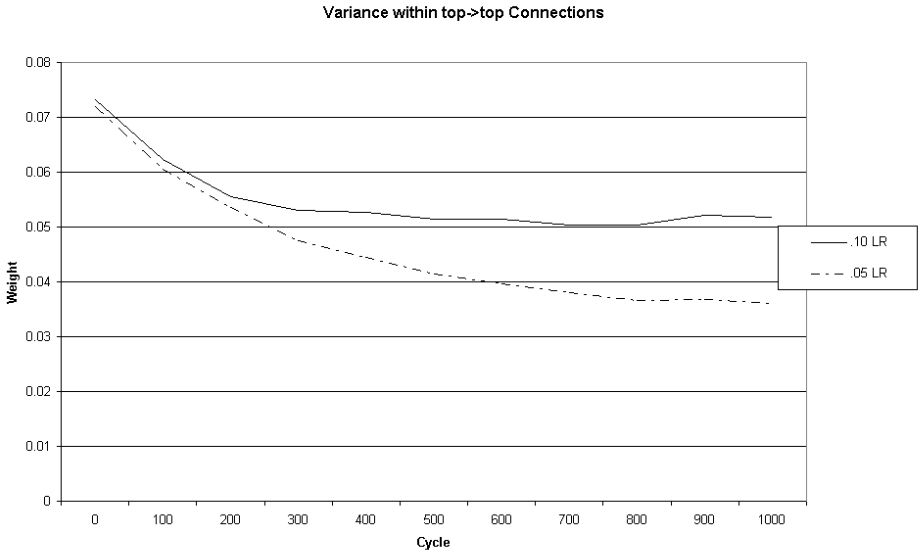

Table 4.

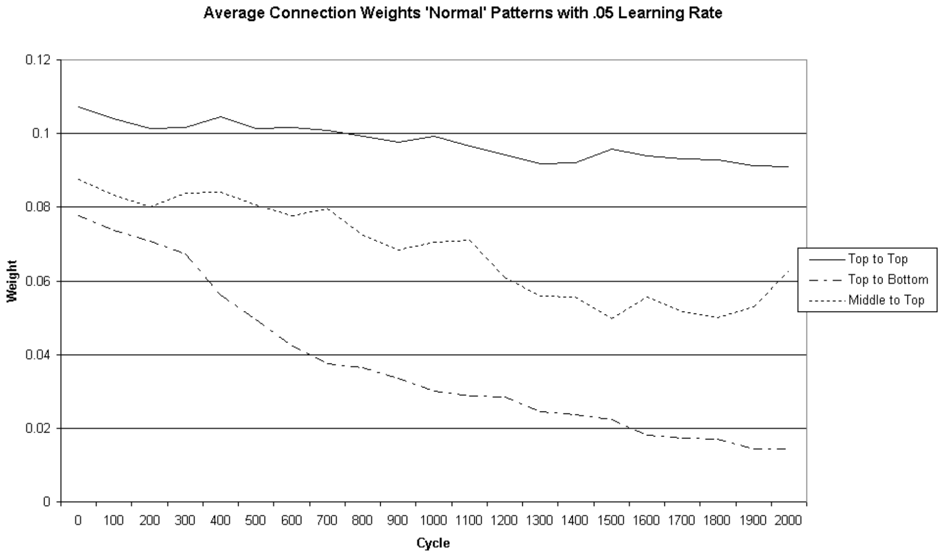

Table 5.

### 2.2.3 Correlation with 'Normal' Patterns

The first two simulations used a pattern that changed every cycle. Prior simulations with CANT [11, 12] have generally presented a single pattern for twenty cycles, then allowed the network to run with no external input for 30 or even 70 cycles. The general idea of this type of presentation is that a particular object is presented for a reasonable amount of time; each cycle represents 10 msec and thus 20 cycles are 200 msecs. Consequently, the system is learning to categorise objects from real objects.

Extra time with no input is allowed to see completion effects, allow competition between CAs, and to enable more CA solidification. However, running without input does not make sense in the simulations in this section because spread of activation is not yet introduced. Consequently, no neurons will be activated from internal stimulation.

Still the way the pattern is presented is important. Presenting one pattern for 20 cycles as opposed to 20 patterns for one cycle each will effect the variance of connection weights.

In this simulation, a pattern was presented for 20 cycles instead of one. The results are described in tables 5 and 6.

Table 5 shows the convergence of connection weights. Note, that there has been a change of scale from earlier tables and table 5 now represents 2000 cycles. There is still convergence, but it has taken more than twice as long. Additionally, the curves bounce up and down.

Table 6 shows the variance. When patterns alternate each cycle (the .10

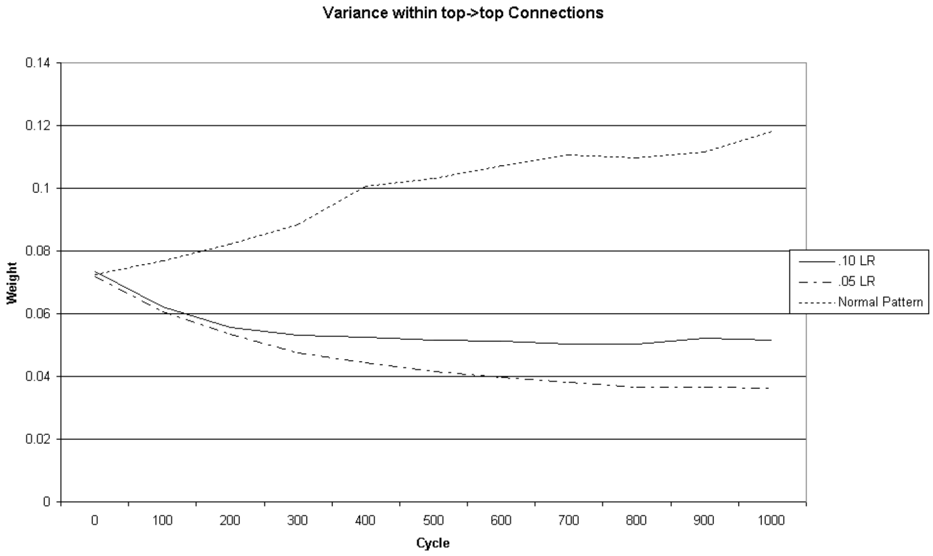

Table 6.

LR and .05 LR lines), the variance goes down to near point 0.04. However, with the 'normal' patterns variance rises to near 0.12. That is, a weight that is typically 0.086 varies on average by 0.12.

This variance is due to repeated changes in the weight of connections. The weight is relatively sensitive to recent behaviour due to the learning rule incorporating the existing strength (see section 2.1). Consequently, the weight shoots up when it is being strengthened, and gets to be very high.

Clearly, the superior mechanism for presentation is a new pattern at each cycle. However, a reduced learning rate would compensate for this variance, and larger nets may be less sensitive to this variance. Finally, spreading activation may lead to less wild fluctuations.

## 3 From Connections to Synapses

In the prior section, the weights between neurons were mere correlators. However biological neurons show that this model is not appropriate. Firstly, neurons fatigue. Secondly, connections spread activation based on their weight. For CA theory, both of these properties are extremely important. This section shows the effect of introducing these properties into the model.

### 3.1 Fatigue

Fatigue is a property of biological neurons [14]. If a neuron is repeatedly fired it will tire, and will be unable to fire or will fire at a lower rate. This

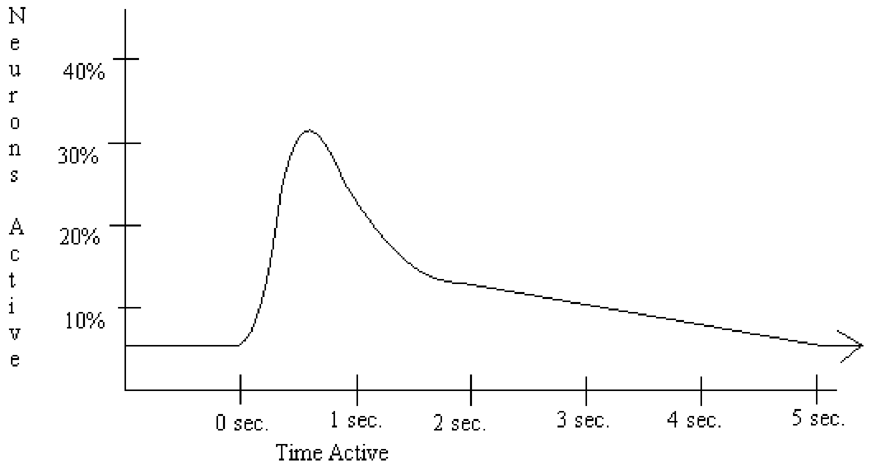

Figure 1.

is separate from a post-firing depression where neurons cannot fire twice within 10 ms. Fatigue is longer and prevents neurons from continually firing for much longer than a second.

For CAs, fatigue is important because it enables CAs to stop on their own accord. Hopfield nets reach a stable point where the same neurons will stay active infinitely. CAs are quite similar to Hopfield nets, but fatigue means that neurons will become less and less likely to fire. Without fatigue, an active CA would either keep the same neurons firing or would cycle through a pattern of neural activation. This same neural pattern occurs with Hopfield nets.

A CA has an activation curve where it is presented information. This can be done by directly stimulating neurons (as in our simulations) or by having neurons elsewhere in the network send activation to some neurons in the CA. If there is enough evidence (activation), the CA will ignite via intra-CA stimulation. After ignition, a large percentage of the neurons in the CA are active. This percentage is gradually reduced largely due to fatigue, until the recurrent activity is no longer supported and the CA becomes inactive.

Figure 1 is a description of this activation curve derived from [14]. Before presentation of the stimulus there may be background activity; some neurons are firing. Presentation of the stimulus causes extra neurons to fire. In the case of figure 1, a cascade of activation occurs, and a large proportion of the neurons in the CA begin to fire. This is CA ignition. After ignition the CA remains active even if external stimulus is removed. This activation continues, but eventually decays due to neural fatigue. At this point the CAs activation may be at or below the base rate.

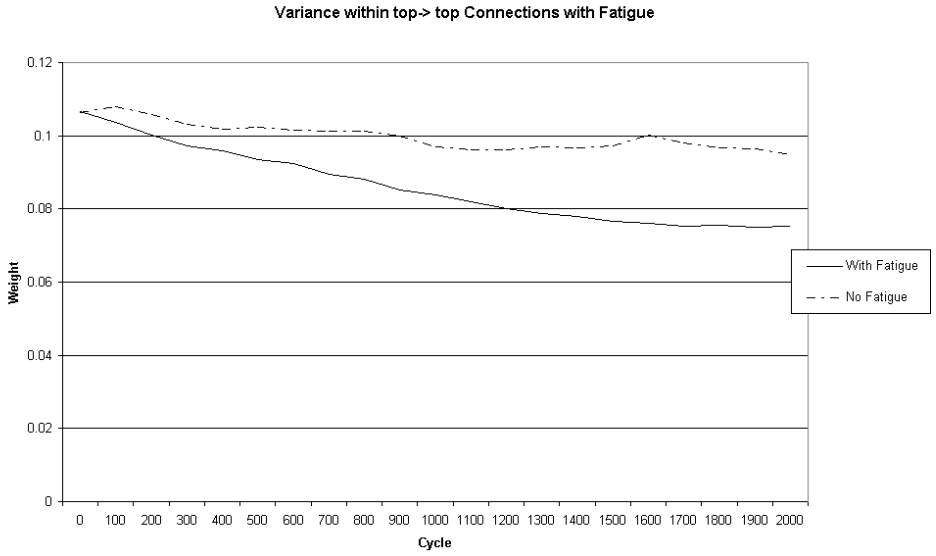

Table 7.

The introduction of fatigue reduces the likelihood of correlation. When twenty neurons are presented repeatedly, some of them will be fatigued and will be unable to fire. Consequently, one would expect the connection weights to be less than the earlier correlations.

A simple simulation is to run the system with fatigue on and off. The simulation from section 2.2.3 was modified. First the patterns were presented with neural fatigue off, then the simulation was rerun with fatigue on.

Table 7 shows the results of this simulation on connections from neurons in the top pattern to other neurons in the top. As expected connection weights are smaller when neurons fatigue.

## 3.2 Spreading Activation

If connections are used to spread activation, these connections are no longer correlators. Firstly, they are now much more like synapses in biological systems and they will be called synapses in the rest of this report. Secondly, since these excitatory synapses now increase the likelihood of the post-synaptic neuron firing, they influence the correlation measurement.

In this simulation we presented the patterns to the network. The patterns were from 2.2 and changed every cycle. The patterns were presented for 20 cycles, and the system was allowed to run for a further 30 cycles. In one condition the network spread activation, and in the other no activation was spread.

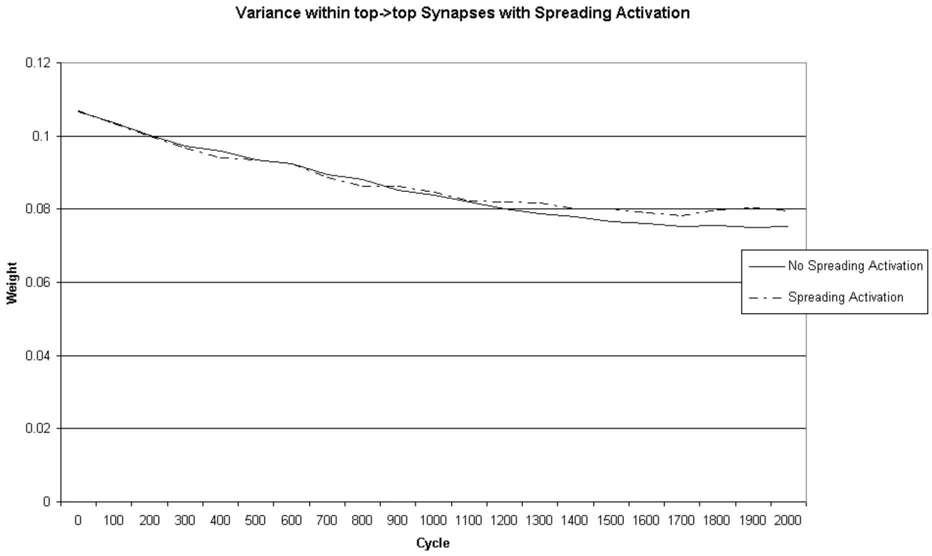

Table 8.

The average weights of synapses were higher in all cases in the spreading activation condition. This is because more neurons were firing in the spreading activation case. In table 8 we present only the synapse from neurons in the top part other neurons in the top, but again the data holds for the other conditions.

What is the use of this spreading activation? From a psychological perspective, CAs must have benefits for the system that uses them. These benefits are well documented and include completion effects, maintaining a concept in working memory and passing information on to other CAs. Completion effects enable a full CA to be activated even though part of it is present in the environment; one may see the front half of a dog, but assume that the dog's tail is there. This 'assumption' is a completion effect.

CAs maintain concepts in working memory by remaining active. Spread of activation allows neurons in the CA to remain active after the external stimuli has ceased. A CA can remain active for much longer than a single neuron because neurons fatigue.

CAs can also be used to activate other CAs. This can be used for memory of sequences [19] or possibly for semantic nets.

At a computational level, the primary function of CAs is categorisation. A stimulus pattern is presented, activation spreads from these externally stimulated neurons to other neurons. If there is enough activation the CAs is ignited, and the stimulus is categorised as that CA.

A CA is formed by Hebbian learning increasing the intra-CA synaptic strengths. This enables the CA to be a reverberating circuit. This has now changed the connections from correlation measures to pattern attractors. The stronger the connection, the more likely the neurons are to be in the same CA.

Reverberation leads to higher separation. Spreading activation above leads to a very small difference in synaptic weight (.02), but reverberation will amplify this difference. So, spreading activation causes a difference in synaptic weights, which enables CA formation. This in turn causes a further change in synaptic weights.

For a CA to form, the patterns must be strong enough to enable the synaptic strengths to grow enough to enable them to activate other neurons in the CA. As CAs are essential, there must be mechanisms to categorise sparse patterns, and mechanisms to break overlapping patterns into separate CAs.

The simulation described above in table 8 does not lead to CA formation as no reverberatory circuit is set up. The parameters of the simulation could be modified to allow CA formation on this particular pattern, but the CA learning mechanism needs to be more robust. CAs need to form when a network is presented with a wide range of input patterns.

The learning rule can be modified to encourage or discourage CA formation and thus categorise sparse patterns. For instance, the compensatory learning rule can be used to encourage CA formation; the compensatory learning rule is a post-hebbian learning rule [21]. See section 4.2 below.

## 4 CAs with Neurons Participating in Multiple CAs

A fundamental tenet of CA theory is that neurons can participate in more than one CA [20]. The fundamental reason behind this is that it enables there to be more CAs, but it also allows more co-operation between CAs. If each neuron participated in only one CA, there are 100,000 neurons per CA, and 500 billion neurons, then there can only be 5 million CAs. If neurons can participate in multiple CAs, it is possible to have more CAs than neurons [22].

### 4.1 Non-overlapping CAs Are Easily Created

There have been many simulations of CAs that have created non-overlapping CAs [15, 9, 12]. Non-overlapping CAs are equivalent to orthonogal inputs in other connectionist work. These networks use simplified neurons that are reasonable models of neurons. However, each of them has neurons participating in only one CA. That is, the CAs were non-overlapping.

For non-overlapping CAs to be learned, all that is needed is that the strength of connections between the inter-CA neurons is increased, and the strength of connections between neurons in different CAs is reduced or kept small. Inhibitory neurons complicate the description, but make CA formation even easier by allowing CAs to suppress each other[2].

Simple Hebbian learning rules can be used to learn non-overlapping CAs. There are advantages to using post-Hebbian learning rules [21]. These advantages include reducing the time to learn a CA, and changing the properties of the eventual CA. These properties such as CA ignition time and duration of CA activation are important, but are not the focus of this report.

It is more difficult to have neurons included in multiple CAs. The problem here is that CAs will ignite and the neurons that are included in other CAs will activate the other CAs. If multiple CAs are active simultaneously, the Hebbian learning rule will cause the CAs to recruit each other and merge into one CA.

## 4.2 The Compensatory Learning Rule

The compensatory learning rule is a variant of the Hebbian learning rule. It is still a localist rule based only on properties of the pre and post-synaptic neurons. The compensatory learning rule considers strength of all the synapses leaving the neuron. If the overall strength is low, the new strength is increased a large amount. This increases the likelihood that the neuron will fire later. Variants of the rule also consider the total strength of the post-synaptic neuron and a combination of both neurons. In this report, the simulations only use the pre-synaptic neuron.

The compensatory learning rule regulates the synaptic strength of all axons coming from a neuron. If there is low correlation on all synapses, the compensatory learning rule will raise the synaptic strengths relative to a non-compensatory learning rule. This will enable CAs to form where there are sparse patterns.

If there is high correlation on all synapses, the compensatory learning rule will cause the average synaptic strengths to decrease. This will reduce the likelihood of simulated epilepsy. It will be harder for all of the neurons to participate in one CA.

If some synapses have high correlation and some low, they will all be moderated. In combination with reverberating activity, this leads to unusual attractor dynamics. See the discussion section 5.

---

[2] Without inhibition, CAs can recruit each other too easily becoming one giant CA. This is simulated epilepsy.

## 4.3 Overlapping CAs Can Be Created

One of the central tenets of CA theory is that neurons can participate in more than one CA [20]. This enables more CAs to exist.

The participation of neurons in more than one CA allows many more CAs because the number of CAs is not linearly limited by the number of neurons. Wickelgren [22] has shown that in a given net there can be more CAs than neurons. A CA is roughly a stable atractor state.

If a neuron participates in more than one CA, it is still spreading activation to all of the CAs when it is firing. Only one of the CAs may actually be ignited at a given time, but other CAs will be primed.

The compensatory learning rule allows CAs to from based on a wider range of patterns. Patterns with low intra-neuron correlation are weak patterns. When there are weak patterns, the compensatory learning rules makes the strengths of synapses stronger. In effect, the learning rule heats up the system, and encourages CA formation.

Once there is enough strength in a neuron, the compensatory learning rule discourages synaptic strengthening. Thus the compensatory learning rule also encourages neurons to participate in only one or a few CAs.

Similarly, if a neuron participates in many highly correlated patterns, only the strongest will have high synaptic strengths. The compensatory learning rule will reduce the synaptic strengths of weaker patterns because it is devoting the neurons synaptic strength to the strongest patterns.

## 4.4 Overlapping CA Simulation

Clearly it is important that neurons participate in multiple CAs. Unfortunately, there has been little simulation of CAs when neurons participate in multiple CAs. Wickelgren [22] simulates this but does not use biologically plausible neurons.

A simulation was done to store overlapping CAs. The two patterns used in the simulation from section 2.2 were used again. The network had 400 neurons, and the patterns consisted of 220 neurons; consequently each shared 40 neurons. Again each training pattern had 20 of these 220 neurons randomly selected as external stimulus. The patterns were presented for 20 cycles then allowed to run for a further 30 cycles.

The first version of the simulation included no inhibitory neurons. Table 9 describes the results. Initially, synaptic weights within a pattern increase and cross pattern weights remain small. However, after about 2000 cycles, the cross pattern weights increase rapidly. This implies that one super-CA is formed. As we would like two CAs to form, this is a poor result. This might be repairable by adjusting learning rates, but we would like a more robust system.

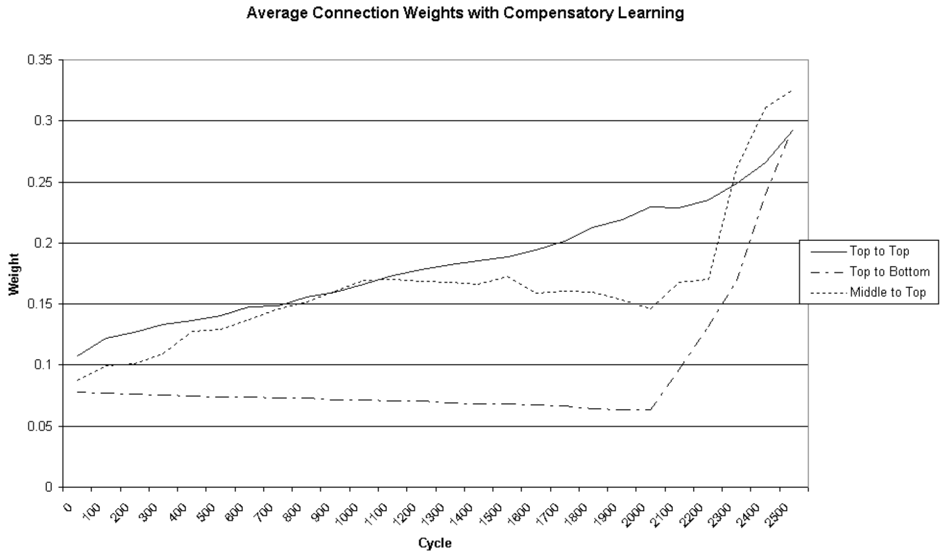

Table 9.

A final set of simulations was done but this time inhibitory neurons were included. Three versions of the simulation were run. In the first, patterns alternated every cycle. After 2000 cycles, two distinct but overlapping CAs formed.

In the second version of the simulation, 'normal' patterns were used. Patterns were presented for 20 cycles then the system was allowed to run without external input for 30 cycles. After 4000 cycles (80 presentations), one CA had formed. This is to some degree a negative result as both patterns were seen as the same.

In the second version of the simulation, the learning rate was set to .1, but in the third version it was set to .05. In this situation, after 8000 cycles 2 CAs formed in 7 of the 10 tests, and a single CA formed in 3.

Table 10 reflects the weights of one of the runs where two CAs form. Excitatory synaptic weights within a pattern are high, while those between patterns (top to bottom) are small. The weights from the middle are large, but smaller than those from the top to the top; this is because extra weights have to be used for the other pattern.

An additional measurement is Pearson's product moment correlation coefficient. The measurement that is used for the network is whether a given neuron is active or not. The Pearson's correlation is used because it ignores the inactive neurons, and in all cases, most of the neurons are inactive.

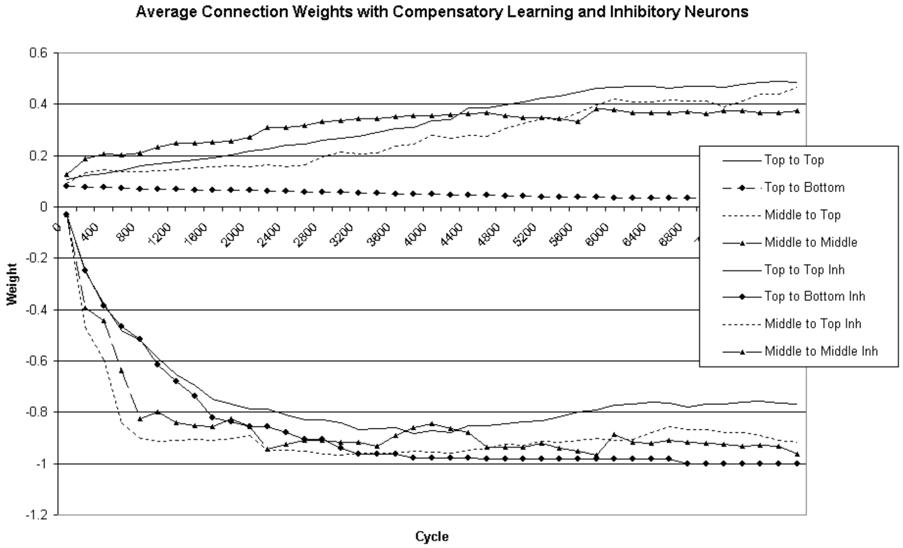

Table 10.

| | |
|---|---|
| Inter-Pattern Correlation | 0.5079 |
| Cross-Pattern Correlation | -0.1952 |
| Cross-Net Correlation | 0.3053 |
| Cross-Net and Pattern Correlation | -.1849 |

**Table 11: Network Correlations**

Table 11 shows the correlations between different runs of the network. After the network has been trained the Pearson's measurement can be used to indicate the presence of a CA. The Inter-Pattern correlation is a comparison between presentations of different patterns of the same type. For example, at cycle 8000 and 8100, the top pattern is presented. The neurons that fire at 10 cycles after the beginning of presentation are compared. As the value is 0.5079, on average over half of the neurons that fire in one presentation fire in the other. This shows that a CA is reliably activated.

The cross-pattern correlation value is a comparison between different types of patterns (top patterns vs. bottom patterns). The negative correlation shows that different neurons fire for the different types of patterns. This shows that the two CAs are unique. These values are based on two networks where two CAs have formed. A comparison of a network where only one CA has formed would have a positive cross-pattern correlation.

The Cross-Net correlation compares the similar patterns on different networks. Again in this case we are comparing two networks where two CAs have formed. This number is lower than the inter-network comparison, but still shows similar activity.

The final correlation compares different patterns from different nets. It shows they are unique.

The Pearson's measurements show that the two CAs really do form. These can be combined with the average synaptic weight to give a good picture of how the CAs develop.

The result of these simulations show that:

- Overlapping CAs can be formed.
- Inhibitory neurons are important in preventing the CAs recruiting each other.
- Oscillating patterns also help to prevent CAs recruiting each other by reducing the variance in synaptic weights.
- A low learning rate also reduces the synaptic variance and also helps prevent CAs recruiting each other.

# 5 Discussion

Earlier we stated that CAs are stable attractor states of the network. The stable state concept is similar to that of Hopfield nets [10]; indeed CANT nets and Hopfield nets are similar in many ways including the recurrent nature of connections.

If fatigue in a CANT net is ignored, a stable state can be reached by a set of neurons continuing to fire, or a cyclic pattern of neurons firing (e.g. neurons 1 to 10 fire in cycle 1, neurons 1 to 11 fire in cycle 2, then neurons 1 to 10 fire in cycle 3). The stable state can be thought of as a CA. There can be many stable states and neurons may participate in several states. This also occurs in Hopfield nets.

Fatigue modifies this because it causes the stability to eventually break down. As a firing CA represents a short term memory, fatigue is useful to cause short term memories to fade.

In addition to being stable states, CAs are also attractors. The completion effects are due to these attractor dynamics. Some evidence for a CA is presented when a few neurons fire. This is not a stable state because the activation will either die out or will lead to CA ignition. As many different initial neural patterns will lead to the same CA igniting, all these initial states are attracted to the same CA.

Inhibitory neurons are important because they encourage CAs to compete and in the long run to separate. If the stimulus is ambiguous, the CAs will compete. Both CAs will have evidence for them and will begin to be activated. Both CAs will contain inhibitory neurons and these neurons will

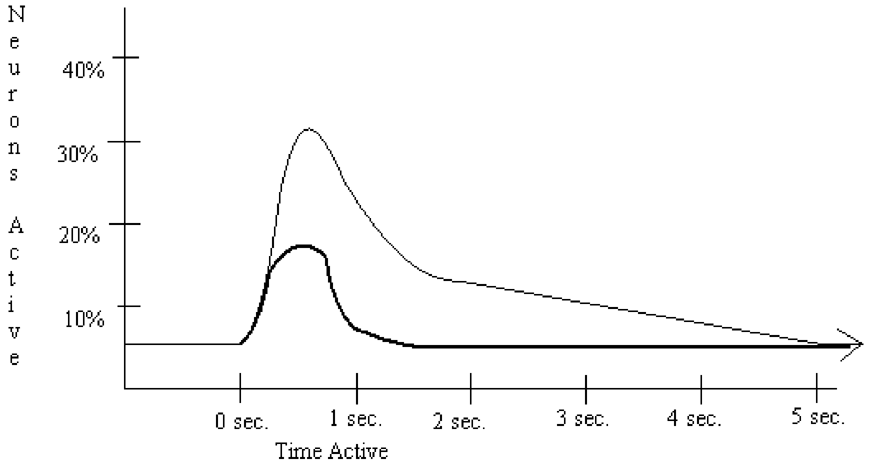

Figure 2.

inhibit neurons in the opposing CA. When one wins, the stimulus is categorised as the successful CA. This is described by figure 2. Both CAs start to ignite but inhibition from one suppresses the other.

Without inhibition, CAs would have to be quite separate sharing few or no neurons. Overlapping neurons would cause other CAs to ignite. Inhibition can prevent the ignition even though there is a large amount of firing neurons in a CA.

The compensatory learning rule enables a wider range of patterns to be the basis of a CA and thus to be categorised. If neurons are not involved in a CA but are externally stimulated by a sparse pattern, the compensatory learning rule will encourage the CA to form by making the synaptic weights grow more rapidly and larger.

If, on the other hand, a neuron participates in many CAs it is encouraged to choose one. When one CA is active, the synapse to the inactive CA will be reduced markedly. In essence, the CAs are battling for control of the neuron. This enables CAs to fractionate into different CAs.

The compensatory learning rule thus encourages recruitment of neurons into CAs and fractionation of CAs into multiple CAs. This can lead to a highly dynamic system where new concepts grow and are created.

This is the first study of simulations of neurons participating in more than one CA that we are aware of. Overlapping neurons are critical to enable a large number of CAs [20, 22].

Moreover, when neurons participate in more than one CA connections between CAs are increased. These increased inter-CA connections enable priming, associative memory, hierarchy and cognitive maps. For example, when neurons participate in only one CA, one CA can prime a second CA by direct synaptic connections. In contrast, when the CAs share neurons, the second CA is primed merely by activating the first. Determining neural membership in a CA is more complex, but this is only a problem of measurement, not a problem for the functioning neural network.

## 6 Conclusion and Future Work

This report has shown how the Hebbian learning rule leads to connections that are correlators. It has also provided a learning rule that enables the connections to act as nearly linear correlators showing the probability that the post-connection neuron fires when the pre-connection neuron fires.

It has then shown how the addition of biologically inspired constraints has effected these connection weights. Neural fatigue reduces the weights and spreading activation increases the weights. With the addition of these biological constraints, the connections can be renamed synapses.

These neurons and synapses can now be formed into networks that can learn CAs. A CA is a pattern categoriser. The pattern is categorised when the CA ignites.

Inhibitory neurons are added to the model. They reduce the ability of CAs to recruit each other and allow CAs to compete.

A compensatory learning rule is used to improve CA creation dynamics. It has the effect of increasing weights when a neuron has little synaptic strength and decreasing strength when it has a large amount. This has the additional benefit of preventing the entire network becoming active.

This network is used to learn CAs where neurons participate in multiple CAs. To the best of our knowledge, this is the first simulation of realistic neurons participating in multiple CAs.

CAs are not completely explained by this work. At a basic level, the CAs described in this work do not behave in the way that is expected from neurophysiological data. Firstly, the size and connectivity of the nets is too small. Secondly, CAs do not persist long enough after external stimulation stops. These CA basics need to be improved.

An obvious extension of this work is to see what types of overlapping patterns can be learned by CAs. Moreover, what types of learning rules and presentation mechanisms can encourage learning of these CAs.

Overall CA recruitment and fractionation need further study. CAs should be able to expand into areas that receive no external stimulation. Work on this has begun using the CANT model and spontaneous activation [13]. Additionally, CAs may break into separate CAs; this is known as fractionation. This needs to be studied. It is plausible that repeated presentation of overlapping patterns, where on CA has formed, may lead to fractionation into two CAs.

So far we have described CAs as categorisors. We hope that these CAs will work together to form more than categorisors. CAs should work together to form an Associative Memory. This memory may also form hierarchies and even cognitive maps.

CAs may also allow rule formation and application. For this to work, the variable binding problem needs to be solved [18]. This will probably require an extension of the CANT model that handles more refined timing data. This is a medium term goal of this research.

In general, we feel that CAs can form the basis of both a full cognitive model and a real artificial intelligence. This work is far from complete, but we are making progress on known problems, and specifying the remaining hurdles to be jumped before these lofty goals can be met.

## Acknowledgements

This work was supported by EPSRC grant GR/R13975/01.

## References

[1] Abeles, M., H. Bergman, E. Margalit, and E. Vaadia. (1993) Spatiotemporal Firing Patterns in the Frontal Cortex of Behaving Monkeys. *Journal of Neurophysiology* 70(4):1629-38

[2] Borman G. Brosens F. and DeSchutter E. (1998) Modeling Molecular Diffusion. Chapter 8. in *Computational Methods in Molecular and Cellular Biology* J.M Bower and H. Bolouri eds. MIT Press.

[3] Bower, J. and D. Beeman (1995) The Book of GENESIS. Springer-Verlag, Berlin. ISBN 3540940197

[4] Bratenberg, V. 1989. Some Arguments for a Theory of Cell Assemblies in the Cerebral Cortex. In *Neural Connections, Mental Computation* Nadel, Cooper, Culicover and Harnish eds. MIT Press.

[5] Churchland, P.S. and T.J. Sejnowski (1992) The Computational Brain. MIT Press.

[6] De Schutter E. Dhyrfjeld-Johnsen J. and Maex R. (1998) A realsistic cerebellar network simulation of mossy fiber induced Purkinje cell activity. *Society for Neuroscience*

[7] Fuster, J. (1995) Memory in the Cerebral Cortex: An Emperical Approach to Neural Networks in the Human and Non-Human Primate. MIT Press

[8] Hebb, D.O. (1949) The Organization of Behavior. John Wiley and Sons, New York.

[9] Hetherington, P. A., and M. Shapiro. (1993) Simulating Hebb cell assemblies: the necessity for partitioned dendritic trees and a post-net-pre LTD rule. *Network: Computation in Neural Systems* 4:135-153

[10] Hopfield, J. (1982) Neural Networks and Physical Systems with Emergent Collective Computational Abilities. In *Proceedings of the National Academy of Sciences, USA* 79 pp. 2554-8

[11] Huyck, Christian R. 1999. Modelling Cell Assemblies. Middlesex University Technical Report ISSN 1462-0871

[12] Huyck, C. 2000. Modelling Cell Assemblies. *Proceedings of the International Conference on Artificial Intelligence* ISBN: 1-892512-59-9 pp. 891-7

[13] Huyck, C. and R. Bowles (submitted). Competition in Cell Assemblies to Resolve Ambiguity *the Journal of Cognitive Systems*

[14] Kaplan, S., M. Sontag, and E. Chown. (1991) Tracing recurrent activity in cognitive elements(TRACE): A model of temporal dynamics in a cell assembly. *Connection Science* 3:179-206

[15] Lansner, A. and E. Fransen. (1992) Modelling Hebbian cell assemblies comprised of cortical neurons. In *Network* 3:105-119

[16] McKenna,T., J. Davis and S. Zorenetzer (eds) Single Neuron Computation (1992) Academic Press, San Diego CA. ISBN: 0-12-484815-X

[17] Milner, P. M. (1957) The Cell Assembly: Mark II. *Psychological Review* Vol 64, No 4.

[18] Palm, G. (2000) Robust identification of visual shapes enhanced by synchronisation of cortical activity. In *EmerNet: Third International Workshop on Current Computational Architectures Integrating Neural Networks and Neuroscience.* Wermter, S. ed.

[19] Rochester, N., J. H. Holland, L. H. Haibt, and W. L. Duda (1956) Tests on a Cell Assembly Theory of the Action of the Brain Using a Large Digital Computer. In *IRE Transaction on Information Theory* IT-2, pp. 80-93

[20] Sakurai, Yoshio. 1998. The search for cell assemblies in the working brain. In *Behavioural Brain Research* 91 pp. 1-13.

[21] Shouval, H. Z. and M. P. Perrone. (1995) Post-Hebbian Learning Rules In *The Handbook of Brain Theory and Neural Networks.* Arbib, M. ed. MIT Press pp. 745-8

[22] Wickelgren, W. A. (1999) Webs, Cell Assemblies, and Chunking in Neural Nets. *Canadian Journal of Experimental Psychology* 53:1 pp. 118-131