Modelling Cell Assemblies

Christian R. Huyck

Middlesex University

c.huyck@mdx.ac.uk

Abstract

Neural Networks are very popular computational models that are generally said to be inspired by human neural functioning. However, neurons in most neural nets function quite differently from natural systems.

This paper describes a novel architecture, the CANT (Connections, Associations and Network Technology) model which is designed to function like natural neural systems. It first elucidates the important aspects of the model and how they relate to natural systems. The basic idea is derived from D. O. Hebb's idea of the Cell Assembly which is the neural equivalent of a concept.

The paper goes on to describe three instances of the model. In each case, stimulus is presented to the network by directly activating neurons in the system. In any given run of the network, this should lead to activation of the appropriate CA. When learning is involved, early runs on the system will not lead to activation of a CA. Instead, the early activation will lead to the system learning the CA.

CAs are initially determined by appropriate levels of activation in the network. Later they are determined by statistical analysis of the activation patterns. The first experiment shows that the CANT model generates CAs given a wide variety of parameter settings. Learning is essential in neural models; the second experiment shows how learning can develop a CA where mere parameter setting can not. In natural neural systems, multiple CAs must exist as natural systems must have multiple concepts. The third experiment shows how a simple net can contain two CAs.

The paper concludes with a discussion of future developments of the model. These include duplicating psychological and neurobiological data. This will require further analysis of CAs, learning algorithms and development of higher order formalisms.

1 Introduction and Background

Neural Networks are very popular computational models that are generally said to be inspired by human neural functioning. However, neurons or units in most neural nets function quite differently from human neurons.

An average human neuron fires for a brief time, and then fatigues. If such a neuron were to represent a concept, we could only think of that concept for a brief time. Since we can think of a given concept for at least several seconds, a single neuron cannot represent an entire concept. A concept must be represented by a collection of neurons. How might such a concept be represented?

D. O. Hebb proposed a solution to this problem [4], a reverberating circuit of cells. Hebb called this reverberating circuit a *Cell Assembly* (CA). Some physiological evidence exists; for instance, Abeles [1] has experimented with electrical probes in the brains of monkeys. There are 10 probes in a small ($\leq$ 5 cm) area of the brain. The electrodes show repeating patterns of activation when the monkey is sensing or performing a specific event.

While there has been a fair amount of work in psychology [8] and neurobiology to show the existence of CAs, there has been little computational modelling of CAs [5] [6]. The computational modeling has either been from a more abstract level than neurons [6], or has been of a very limited nature [5].

This paper describes several instances a novel architecture, the CANT (Connections, Associations and Network Technology) model. The next two sections describe the general CANT model. One section is on the basic neuron and one is on the net and CAs.

The neuron is the basis of the model. The neuron has axons (connections); a current activation; an activation threshold it must reach before firing; it is connected, via axons, to other neurons in a distance-biased fashion; the activation of a neuron decays; and firing neurons fatigue.

The section on the CANT net and the properties of its CAs is next. CAs should follow a standard course of activation. They should emerge from an initial net via training and a localized learning rule. This section also describes the structure of the net used in the experiments described in this paper.

The fourth section describes three experiments. The first experiment shows the robustness of the model. The second integrates learning. The third shows two CAs in a single net.

The fifth section further discusses the results of the experiments. It elaborates the problems encountered during the experiments and how they relate to general problems with CAs.

The concluding section discusses the long-term and short-term plans for the model. It concludes with the overall goals of the model

2 Basic Neural Model

The basis of the model is the neuron. Very complex models of neurons exist (eg. [3], [2]). However, these models tend to take quite some time to run and since many neurons are needed, a simple model has been employed. CANT attempts to make good trade offs between computational efficiency and neurological validity, capturing the important computational aspects of the neuron efficiently.

The CANT model has six neural properties:

1. Connection Strength
2. Activation
3. Activation Threshold
4. Variable Connectivity
5. Decay
6. Fatigue

The first three are quite common in Neural Network models and the last three are less common.

2.1 Connection Strength

A CANT neuron has connections to other neurons, which are similar to connections inside biological neural systems. Connections are unidirectional. Like most neural net simulations, the connection strength may vary based on a learning rule. The connection may have positive or negative strength. Continuous activation is simulated by time steps.

The average biological neuron is activated by about 1000 other neurons, and in turn activates about 1000 other neurons [7]. Each biological neuron has several axons, which have feet to send activation to other neurons. The CANT model simply has neurons and connections. In the experiments described in this paper, there are fewer connections due to the small number of neurons. In future experiments with larger networks, more neurons and connections will be used.

2.2 Activation

When the neuron crosses a threshold, it sends activation down each of its axons. The activation of a given neuron i at time t is:

$$h_{i_t} = \frac{h_{i_{t-1}}}{d} + \sum_{j \in V_i} w_{ij} s_j$$

Equation 1.

The current activation is the activation from the last time step divided by a decay factor d plus the new activation coming in. This new activation is the sum of the active inputs s_j of all neurons $j \in V_i$, V_i being the set of all neurons that are connected to i, weighted by the value of the connection from neuron j to neuron i.

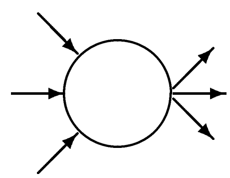

Figure 1.

There may also be external activation. Theoretically this comes from the environment, but in these experiments it comes from neurons being directly activated. Some experiments allow neurons to spontaneously activate when they have been inactive for a long time.

2.3 Activation Threshold

A neuron is active if and only if it has enough activation to surpass the activation threshold. Each neuron has the same activation threshold as all other neurons. So if the activation threshold is 5 and a given neuron has an activation of 4, it will not fire and thus will not propagate activation. Activation of an inactive neuron does decay but an inactive neuron does not fatigue.

2.4 Connectivity

Biological neural systems are connected in a distance-biased way. Each neuron is not connected to every other neuron; in the human brain, with 500 billion neurons, this would require each neuron to have 500 billion connections, and as we have seen the average neuron has approximately 1000 connections. If two neurons are closer together they are more likely to be connected. The CANT model adheres to this distance-bias connectivity.

2.5 Decay

At each step, the activation from the last step decays. Of course, new activation may lead to a net gain in activation. Decay is a constant and applies to active and inactive nodes.

2.6 Fatigue

The neuron fatigues. When a biological neuron is active for a long time it will fatigue and this will make it less likely to remain active. This is modelled by a fatigue factor, which increases the activation threshold. The threshold is increased by $f_c t_a$. f_c is the fatigue constant and t_a is the time that the neuron has been active. The longer that the neuron is active the larger the threshold becomes, and thus the less likely it is to remain active.

When a neuron becomes inactive, fatigue is reduced. Fatigue is $t_a - t_r R_c$. Fatigue is the time active minus the time recovering multiplied by the recovery constant. The higher the constant, the faster a neuron recovers from fatigue.

In the models described in this paper, a neuron tends to stay active for about 10 cycles; this varies with decay and fatigue. Even with one decay and fatigue setting this does of course vary. A neuron may remain active for as little as one cycle, and if given enough activation, may remain active indefinitely. In the experiments described in this paper, the longest a neuron has remained active is 13 cycles.

3 The Net and Cell Assemblies

How might a CA look at the cellular level? A CA should consist of a relatively large number of neurons; this could be hundreds or millions. The neurons should have a large connectivity to each other; that is, each neuron should have connections to other neurons in the assembly and the strengths of those connections should be high. Each neuron does not need to be connected to all the other neurons in the CA and may be connected to neurons outside the CA. This large connectivity should lead to mutual activation.

When several neurons in the CA are activated, they should activate other neurons in the CA. When the initial neurons fatigue and cease to be active, the newly activated neurons keep the activation in the CA. The initial neurons, after recovery, may later be reactivated. Thus the CA is a reverberating circuit, and can remain active much longer than a single neuron.

The time course of activation of a CA follows a simple curve. If enough stimulus is present a large number of cells will be activated. That is, if enough neurons are initially activated in the CA, they will cause a large fraction (1/3) of the neurons in the CA to become activated. After this initial burst of energy overall CA activation (number of active cells) will be reduced, but will remain high for a period. This activation will gradually decline. This gives the "snoopy" curve of activation [6] [1].

If insufficient evidence is present, a number of cells will still be activated. However, it will not be enough to activate the circuit, and overall activation in the circuit will quickly decline.

Learning in the CANT model is unsupervised. While different learning rules have been tested within the CANT model, all are basically Hebbian. That is, the learning rules are based on the activation of two adjacent nodes. If node A has a connection to node B and both are active, then the connection strength is increased. Strengths may also be reduced and may even become negative. This is either via a compensatory learning rule [2] or via Long-Term Depression [5]. Again, these are learning rules based on activity at two adjacent neurons. The learning rule for a particular model will be described along with the description of the appropriate experiment.

As Wittgenstein [11] has made clear, there usually are not necessary and sufficient conditions to say an entity is an example of a concept. Generally, dogs have four legs, but three-legged dogs exist. To be a bit more simplistic, a concept is a group of features that tend to travel around together. We have the concept (and thus the word) dog because the features tend to travel around together. We often find wagging tails, fur, wet noses, four legs and other dog features together. Rarely, if ever, do all of the features coexist, but they tend to be together. This type of concept relates to Rosch's [9] prototype theory.

An instance of a concept is present when enough of its features are present. In the CANT model a CA is activated when enough features in a given feature set are present. Neuronal activation is used as a rough equivalent to a feature being present. A concept is made up of a set of features (neurons in CANT terms). So if a sufficient number of those neurons are activated, then the CA should become active. The network will recognize that the concept is present in the environment.

As with most Neural Net models, the CANT model has the advantage of redundancy. A large number of the neurons or axons in a given CA (or a net) can be removed and the CA (or net) will function in almost exactly

[1] Kaplan cites peak activation in the range of 75% of neurons in a CA. It is not clear how this number is derived. An important question is how much of a CA is activated? Still the general shape of the curve remains uncontested.

[2] This rule sets a threshold of positive strength for all axons from a given neuron. Once this threshold is reached, any gain must come at the expense of other neurons.

the same way.

The main goal of the CANT model is that it is a good model of neural processes. At some point of development, the CANT model should start to exhibit simple psychological behavior. Of course, this will initially be very low-level behavior.

An example of such a behavior would be the presentation of ambiguous input. (The Necker cube is an example of this input.) An ambiguous input should be resolved as one of the possible concepts, but this should take longer than when an unambiguous input is presented. Activation of one CA competes with the activation of another until one wins. Since, each is suppressing the other, a peak of activation (recognition) will be delayed.

Distance-biased connections should lead to some localization in a CA; that is, the reverberating circuit should consist of neurons that are close together. When a neuron is activated it is likely to activate neurons near it, and via learning these connections will be strengthened. As the system learns more and more, nearby neurons will be recruited into the CA, and it is likely that these neurons will be nearby. The entire CA may be spread out over large areas of the brain, but any parts of a CA in a given area are likely to be close together.

A complete CANT system consists of a number of neurons laid out into a network. This net can be of any configuration. In the experiments described below the net is a two dimensional array of neurons (N by N neurons). Distance is determined by adjacency; the 0,0 neuron is 1 step away from the 0,1 neuron and from the 1,0 neuron. 0,0 is two steps away from the 0,2 neuron and from the 1,1 neuron. There is also wrap around so 0,0 is also one step away from 0,N and N,0.

CAs should emerge from the net via learning. Given multiple presentations of similar items, the net should develop a CA based on that item. Depending on the history of that net (what it has learned), it may contain multiple CAs.

A concept should be activated when there is external evidence that it is present. A concept is represented by a CA, and in this case, external means evidence outside of the CA. This evidence consists of activation. If the CA were connected to sensory devices, specific neurons would be sent activation. Similarly, if it were an internal CA, the stimulation would come from other neurons closer to the sensory interface. In CANT external stimulation is simulated by giving activation to a particular neuron or neurons.

Thus a particular input is presented to the system by activating certain neurons. The net runs. If a CA is activated, then the net has recognized the input as an instance of the concept that the CA represents.

4 Experiments

The basic CANT model is largely constant. It consists of a series of neurons, connected in a distance-biased fashion. When a neuron is activated, it spreads its activation through its connections. If a neuron gets enough activation, it fires and sends activation or inhibition to adjacent neurons. The activation of a neuron decays, but a neuron may remain active for many time steps. Though a neuron gets more activation, it will be more and more difficult for it to remain active due to fatigue.

These experiments are designed to test whether the CANT model can successfully generate CAs. The first experiment shows that the CANT model generates CAs given a wide variety of parameter settings. Learning is essential in neural models; the second experiment shows how learning can develop a CA where mere parameter setting can not. In natural neural systems, multiple CAs must exist as natural systems must have multiple concepts. The third experiment shows how a simple net can contain two CAs.

In each case, stimulus is presented to the network by directly activating neurons in the system. In any given run of the network, this should lead to activation of the appropriate CA. When learning is involved, early runs on the system will not lead to activation of a CA. Instead, the early activation will lead to the system learning the CA.

Three experiments using CANT are described below. The first does not take advantage of learning. The second does take advantage of learning, and thus builds a better CA. The third has multiple CAs in a given network.

4.1 One Cell Assembly in a Network

When an input is presented to the system, if the CA is activated, the input is categorised as an instance of the concept represented by that CA. In terms of the net, the CA is activated when a sufficient number of neurons in the CA are activated [6].

The general course of activation should follow the snoopy curve shown in figure 2. [6].

When the external stimulus is presented a few neurons become active. These in turn pass on activation to more neurons, and eventually a great number of neurons become activated. Activation should continue even if the stimulus is removed; after all, you can continue to think about a concept even after you have ceased to look at the instance of the concept. However, there should be less and less activation as time passes.

Two potential problems present themselves: activation explosion and lack of activation. Activation explosion occurs when too many nodes become active; at any given time many nodes are active. When a node fatigues,

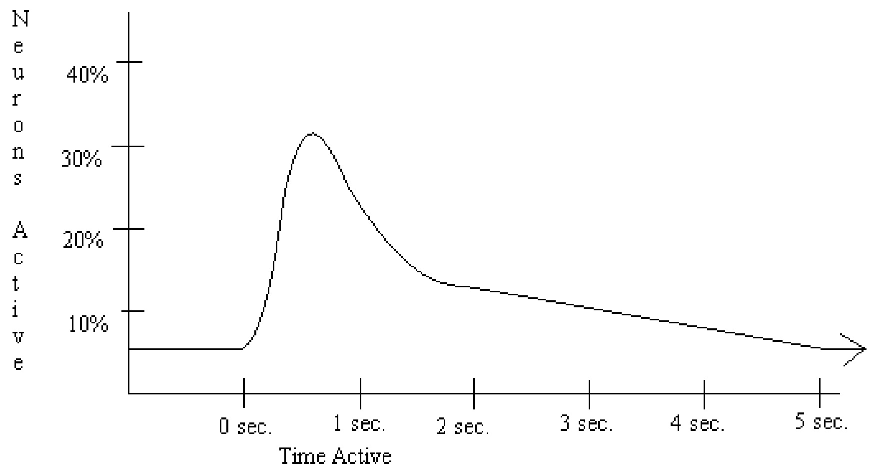

Figure 2.

activation from other nodes reactivates it. The problem with this is that the CA always remains active. Instead the CA should remain active for a period of several seconds after the stimulus has gone.

Lack of activation is the other problem. A stimulus is presented and this activates certain neurons. These neurons will remain active for a brief period after the stimulus is gone, but will decay and fatigue and quickly become inactive. The only way for the CA to persist is for the initial neurons to activate other neurons. If there are insufficient connections and connection strength, then no other neurons will become active.

The first question is how easy is it to create a CA? Can they be randomly generated or do they need to be learned?

A related question is what are the values of the initial parameters. The parameters are connectivity, connection strength, activation threshold, decay, and fatigue.

The first experiment was to test nets with a great number of different parameter settings and see which configuration led to the activation pattern most similar to the snoopy curve.

For a network to match the snoopy curve it had to pass three separate tests. Each test assured that the network never had too much activation, and that it had sufficient activation at its maximum. The first test initially activated 20 nodes, while running it could have no more than 1/3 of the nodes active, and had to have at least 5% active. The second test activated 10 nodes, could have no more than 25% of the nodes active and had to have at least 2% of the nodes active. In the final test two neurons were activated and no more than 2% of the nodes could become active.

Initially this was tried with a 20x20 network of neurons. Despite testing several thousand different parameter settings, no appropriate activation pattern emerged.

Another experiment was done with a 30x30 network of neurons. In this case 52 parameter settings led to the appropriate activation pattern.

	Low Act.	Med Act.	High Act.
Conn. Low & Str. Low	4	3	0
Conn. Low & Str. Med.	1	11	2
Conn. Low & Str. Hi	0	4	4
Conn. Med. & Str. Low	6	5	2
Conn. Med. & Str. Med.	0	2	3
Conn. Med. & Str. Hi	0	0	0
Conn. Hi & Str. Low	0	4	1
Conn. Hi & Str. Med.	0	0	0
Conn. Hi & Str. Hi	0	0	0

Table 1: CA occurrence via parameter settings.

Table 1 shows the number of parameter settings that lead to CA formation. Connectivity and connection strength along the left, and the activation threshold on top. For example, 4 parameter settings with low connectivity, low strength, and low activation lead to CA formation. A high connectivity means that a neuron was connected to more neurons than low connectivity. High connection strength means that each connection has a higher w_{ij} than low strength; so if the neuron is activated, it will send more activation to the neurons it is connected to. A high activation threshold means that a neuron needs more activation before it can fire, as opposed to a low activation threshold where the neuron will fire with a low amount of activation.

The table shows that high connectivity and strength were not very successful. This might have been compensated by an extremely high threshold or a large decay and fatigue factor.

The most successful groups were low connectivity and medium strength, and medium connectivity and low strength. Both were successful with low, medium and high decay constants.

All ranges of individual parameters led to an appropriate CA activation curve. A low activation along with a low connection strength and low connectivity could generate an appropriate curve. Similarly, a high activation along with a medium connectivity and strength would yield an appropriate curve.

High connectivity and strength tend to encourage activation. A high activation threshold tends to discourage activation. As long as the two are balanced a CA can form. This is shown by table 1. The only place low connectivity and strength fail is when there is a high activation threshold. Similarly high connectivity only succeeds with medium and high activation thresholds.

Decay rate and fatigue also tend to discourage activation though they are not shown in the table. The full range of decay and fatigue values lead to CAs. High fatigue values were more successful than low values but only marginally so.

There are two important lessons to draw from this experiment:

1. bigger nets are more flexible, and
2. the CANT model is robust.

The difference between the larger and the smaller net shows that larger nets are more flexible. No parameter settings were successful in generating a CA in a net with 400 neurons, but 52 settings were successful in generating a CA with 900 neurons. It is much easier to generate a snoopy curve of activation in a larger net. This implies that you can do more things with more neurons. To us this is obvious, but it is important that the model exhibits this behavior.

The variety of parameter settings that generate CAs show that the model is robust. High, medium and low values of all of the parameters led to CAs. These needed appropriate values from the other parameters, but the flexibility of the model is encouraging.

This is particularly encouraging as the parameters were set and no learning was used. One of the parameters, connection strength, would be largely set by learning. A second parameter, connectivity, could be reduced by learning [3].

This experiment shows that it is not easy to create a CA by randomly generating a network. It is easier to do so in a larger net. CAs were formed when the connectivity and strength of connections was balanced by a similar activation threshold.

A related question is what are the values of the initial parameters. The parameters are connectivity, connection strength, activation threshold, decay, and fatigue.

After these successes, parameters were set and it was hoped that a CA could be learned in a net with 400 neurons. This experiment is described next.

---

[3] Setting the connection strength to 0 is equivalent to removing a connection, thus reducing connectivity. As long as the connection strength remains at 0, it is as if there were no connection. Of course, future changes from zero would result in a different system than one that had no connection at all.

4.2 Learning One Cell Assembly in a Network

In Neural Networks, the topology of the network is important. In models of Hebbian Cell Assemblies, the topology is defined by the connections between nodes and the weights of those connections. The above experiment showed that several topologies could elicit the appropriate behavior in a larger net, but it was difficult (random searched failed) to find a topology which elicited an appropriate response in a smaller network.

Learning is a hallmark of both natural and artificial neural systems. Could the system learn an appropriate topology?

In CANT, learning is assumed to be unsupervised. The system is presented with stimuli and it needs to categorise those stimuli. If similar stimuli are presented, a similar group of cells (a CA) gets activated. What is learned is that categorization. In the CANT model, learning is implemented by changing the weights of the connections.

The CANT model is based around Hebbian learning; that is a localized model of learning. Within this model there are several alternatives. All are based on adjacent activations. If node A is connected to node B, and both are activated at a given time step then the strength of the connection is increased. This is known as Long Term Potentiation and is shown in Equation 2:

$$w_{ij} = w_{ij} + R$$

Equation 2.

The connection strength of the axon is increased by the learning rate R. Various other methods are used to handle decreasing strengths of connections.

In this experiment, loss of strength is handled by a simple compensatory learning mechanism. There is a maximum of the total connection strength of axons coming from a neuron. If learning increases the strength of a given axon, M_{ij}, so that the total strength exceeds the maximum connection strength of all axons M_{ix}, all of the other axons have their strength reduced.

This reducing rule can lead to inhibitory links. Thus some active neurons can reduce the activation of other neurons.

The initial parameters were set based on experiment one. The activation threshold was set to a medium value. Decay was set to a medium value though a small value was also attempted. Connection strength was set very low as this would be modified by the learning rule. Connectivity was set very high as to some extent this was modifiable downward by the learning rule.

In this experiment, a net of 400 neurons was used. The net was connected in a distance-biased fashion as shown in section 3. During training a random set of 20 neurons was selected and activated for each run. Some neurons were labelled as primary neurons, and the rest as secondary neurons. The primary neurons were more likely to be activated from the environment. The experiment varied the number of primary neurons from 0 to 20. Each activated neuron was activated at twice the threshold for the first time step. After the first step, there was no more external activation.

Initially the connection weights were set to a low random number. On the initial training run, no neurons aside form those externally activated surpassed the activation threshold.

For each test with different numbers of primary neurons, after 400 runs, a solid CA had been formed based on the criteria of experiment 1. This was also the case with both medium and low decay rates. In each case, if 20 neurons were externally activated for one time cycle, more than 40 neurons became active, and the activation persisted for over 30 cycles.

The best result was when there were a few primary neurons. In this case, 71 neurons became active, and the activation persisted for 62 cycles.

This system proved robust. If training was done with 10 neurons being activated instead of 20 neurons, a similar result was found. If the net generated through 400 runs of 20 neurons of activation were tested with 10 units of activation, it showed the snoopy curve but with a lower ceiling.

There were some problems with the system. As the network continued to train the values of connection strengths tended to diverge. Each connection would either have a low value or a high value. This was possible as the learning rule allowed the connection strength of all of the neurons to have a threshold, however there was no limit to the individual strengths.

This experiment shows that a simple learning algorithm can lead to a CA-like topology. Experiment one showed that this topology was difficult to generate on a net of 400 neurons. This experiment showed the appropriate activation curves on different types of external inputs.

Clearly the model of experiment two is very incomplete. The main problem is that the network contains only one CA. A network needs to contain more than one CA. It then becomes a recognizer of various objects. The next experiment discusses such a system.

4.3 Multiple Concepts in a Network

The first two experiments were based on the overall behavior of a net. They showed that neurons in a net could exhibit activation long after a stimulus had ceased to be presented. Moreover, this activation followed the desired snoopy curve.

As stated earlier, the goal is to have one CA activated when one type of pattern is presented and another activated when a second pattern is presented. Below we will show that the network of the third experiment exhibits this behavior.

This system is identical to the Experiment 2 system explained above except it had 900 neurons instead of 400 neurons. The major difference in the experiments is that two types of patterns were presented instead of 1. Each pattern had 10 primary activation points and 10 secondary activation points. The points of pattern A were between 0 and 100 while those for pattern B were between 500 and 600. At each training run, 6 primary and 4 secondary neurons from either A or B were activated. The type of activation (A or B) was randomly selected, as were the neurons in any given pattern.

Neurons may also spontaneously activate. If a neuron has not fired in many (hundreds of) cycles, it may spontaneously activate. This mimics auto-activation in the brain. An example of auto-activation is when a person hallucinates while in a sensory isolation tank. This has interesting effects on CA development discussed in the next section.

This system is similar to that described in [5]. Hetherington's system had 360 neurons and learned 4 patterns, but his patterns were 5 neurons and all were always turned on.

Following Hetherington, it is hoped that CAs are *unique*, *persist*, and are *reliably activated*. Statistical measurements of the entire network at certain stages can show these properties. Uniqueness can be measured by comparing activation patterns of different CAs; when an A pattern is presented to the network it should lead to the activation of different neurons than if a B pattern is activated [4]. Persistence can be measured by presenting a stimulus to a network, and then comparing the activation patterns in early cycles and in later cycles. If the two are correlated, then the CA persists. A CA is reliably activated if different inputs of a similar pattern lead to similar neurons being activated. For instance, one A pattern can be presented and the state of the network can be measured after a given number of cycles. When a second (but different) A pattern is presented, it should lead to the activation of a similar set of neurons.

The statistical measurement used in this evaluation 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 2. shows different correlations based on the number of training runs. These correlations show the development of CAs in the net.

The columns represent the number of training runs. A training run consists of a presentation of the stimuli followed by 1000 cycles of activation

[4] While there is nothing theoretically to stop a neuron from participating in two CAs, in the current experiment it should not participate. Each input pattern is entirely separate. It's like comparing As to Os; they have no overlapping features. A different pair of input patterns, say Es and Fs, might contain overlapping neurons in the CA.

propagation. The table shows the state of the initial net (0 Runs), the net after 200 training runs, and after 400 runs. In each case the system was presented with 4 A stimuli and 4 B stimuli, in different runs. The table represents the average results.

	0 Runs	200 Runs	400 Runs
Max. Neurons	10	30	31
A-A Corr.	-.0074	.7585	.7442
B-B Corr.	-.0067	.6456	.8077
A Self Corr.	-.0068	.1649	.2110
B Self Corr.	-.0057	.2175	.2332
A-B Corr.	-.0073	-.0068	-.0351

Table 2: Network Correlations

Max. Neurons represents the maximum number of neurons that are activated in any cycle of a run. Since the cell of the table represents eight runs, this number represents the average maximum number.

A-A Corr. represents the correlation of active neurons between different runs of type A. The measurement is taken at cycle 15. *B-B Corr.* represents the correlation between B runs. Since four runs occurred, the correlation is the average of the six comparisons. These measurements show reliable activation. If different but similar patterns are activated, they should send activation to the same neurons, especially in the initial phase of CA activation. The higher this number the more reliable the CA.

A Self Corr. shows the correlation between the activation states of one net in a given run when there is an A stimulus. The correlation is between the net after cycle 15 and after cycle 30. *B Self Corr.* is the version when there is a B stimulus. Since there are four patterns of these types, this number represents the average of four runs. This number is an indicator of persistence [5]. Similar neurons are active early in the run and later in the run. To some extent, the higher the number, the more persistent the CA.

Finally, *A-B Corr.* represents the correlation between runs of different types. Again, the state of the net was recorded after cycle 15, and again the average correlation is shown. This number shows uniqueness. If two network activation patterns derived from different types of inputs are compared, they should not have similar neurons activated. The lower this number, the more unique the CAs are.

[5] It really only shows similarity of activation at two different parts of a given run. Since neurons fatigue, the same CA may be active but different neurons are active. A better measurement might be to compare several states (eg. after cycle 15, 30, 45 and 100). Another measurement might be to see the reverberating activity of a particular neuron. Do the input neurons become active again after fatiguing? This would parallel [1]. See the discussion section for further elaboration.

In this experiment, the two CAs are in competition for neurons. Clearly, the neurons that are part of the initial patterns (both primary and secondary) will be part of their respective CAs. Additional neurons will be added to the CAs via learning. For instance, a neuron X that is being recruited by CA A will have connections to it from neurons in CA A. These connections will be strengthened by the learning rule. Connections from X to neurons in CA A will also be strengthened. Connections from X to neurons in CA B will have their strengths reduced and might become inhibitory.

Thus when CA A has evidence that supports it, it will become active, and it will suppress CA B. As CA A recruits more neurons it will be able to suppress CA B more readily, and will thus be able to recruit more neurons. Similarly CA B will be trying to recruit neurons from CA A.

5 Discussion

There has been much discussion about CAs; what is their size? What exactly do they do? However, there has not been much work in computationally modelling CAs from a neural basis. The goal of this work is to explore CA topologies and properties. If CAs are the basis of thought, as Hebb proposed [4], it is very important to be able to model them.

This paper proposes the CANT model and tries to show that it is a good starting point for modelling CAs. The three experiments have shown how CAs are built using the CANT model.

The first experiment made three main points. It was shown that the appropriate activation curve can be generated. It showed that different parameter setting could generate the appropriate curve and this was used to determine initial settings. Since a variety of parameter settings can be used, this also shows that model is robust. Finally, the experiment showed that it was easier to generate the appropriate curve with a larger network.

The second experiment introduced a simple learning rule and showed that it could be used to generate CAs. This was done on a smaller net because a net of this size had not previously been able to produce the appropriate activation curve. Thus a CA could be created via learning which is an important prerequisite of CAs. Moreover, learning developed a topology which could not be found via random search.

The third experiment dealt with two competing CAs in a net. If CAs occur in biological neural nets, then they must put many CAs into one net because the net (the brain) needs to have many concepts. The third experiment showed that multiple CAs could be put into a CANT net.

The statistics for uniqueness show that the CAs that have been constructed contain no overlapping neurons. That is, a neuron is a member of at most one CA.

Another problem is saturation. If the system is allowed to train for 1000 training runs, uniqueness is lost. The system becomes an A or B recognizer. It is hoped that this is a property of the rather simple learning rule. A more sophisticated learning rule, perhaps using post-not-pre Long Term Depression, might eliminate the problem [10].

A CA is formed by recruiting neurons. That is the connections between neurons are strengthened. As the interconnections increase, a reverberating circuit, a CA, is formed. As the CA grows larger, it should be able to fractionate into separate CAs that are more appropriate. For instance, at some point, most children have the concept of four-legged animal. All four legged animals get called "dog". Later the (concept) CA fractionates into dogs, cats and other animals. The problem with Experiment 3 is that the CAs grow and grow until they recruit each other (around training run 1000) The system become an *A or B* recognizer and uniqueness vanishes.

When a single activation pattern is repeatedly presented, a small tight CA develops. This CA quickly activates all of its neurons, but then fatigues and does not reverberate. The CA is loosened in two ways. First, a broader pattern may be presented. This leads to more neurons being recruited into the CA. Each neuron that is externally activated will recruit adjacent neurons. When more neurons are part of the pattern (on some training runs), more neurons will become part of the CA.

The second way to loosen a CA is random, spontaneous in this case, activation. Introducing random activation to the net leads to extra neurons being recruited into the CA. This will lead to lengthening the time of activation. This is valid from a real-world standpoint. No network is constantly presented a stimulus. Occasionally it is at rest, and this is when spontaneous activation occurs. Both of these loosening techniques can lead to the subassemblies Hebb describes [4].

A final problem with these experiments is the statistical analysis. The Pearson's product moment correlation coefficient is good for uniqueness and reliable activation as it is used to compare which neurons are activated early on in CA activation.

However, it is a poor mechanism for measuring persistence. A neuron should fatigue after a few cycles and for the CA to persist other neurons should be activated. The initial neuron may be reactivated later, but the net should not have a high Pearson correlation ($< .50$) with itself at vastly different cycle numbers.

Hetherington [5] reports persistence correlations of $r{=}.95$. Clearly this is not measuring persistence of the CA but persistence of individual cells. The measurement must be measuring the neurons remaining active, not new neurons being activated.

While Pearson correlation values within a particular run can be useful, the numbers should be lower. Moreover, measurements such as reverberating cell activity and overall net activation (similarity to the snoopy curve) should also be useful.

6 Conclusion and Future Work

CANT is a model in development. There are both long-term and short-term goals for the model.

The long-term goal of the model is to:

1. discover how CAs work.
2. discover what CAs can do.
3. duplicate psychological data with CAs.

All psychological processing can be done with a CANT model. This means that learning, forgetting, rule like and goal oriented behavior, even creativity should be possible with a CANT model. These things may not be possible; if not why not?

How can a CA be learned after seeing only one instance of an object? Can emotion and consciousness be integrated into the model?

These are the long-term goals of the CANT model. It is hoped that it will be the bases of a real, hard, Artificial Intelligence.

The short-term goals are much less lofty, though difficult (and one might say achievable) in themselves. All these goals can be reached within the current properties of the CANT model (described in section 2).

The current CANT systems do not duplicate the appropriate activation curves well. A better model needs to be developed. This may require a change in size of the net or in the learning rule, or in both.

The model can be made useful for real world tasks by adding "supervised" learning. "Supervised" means that the answer is presented as part of the input stimulus. A standard local Hebbian learning rule will still be used to associate the input patterns with the answer. When an input pattern is presented alone, the answer neuron (or neurons) will also be activated, and the system will actually have decided on a classification. This in effect is an effector neuron.

Experimentation with the model should include variations on the input patterns. Can the system build separate CAs from overlapping input patterns? (Can it be an E or an F recognizer?) What happens when three or more pattern types are presented to a net? How does the locality of the input pattern change the CA?

The evolution of a CA should also be studied. An initial net grows its CAs from experience. The CA should then be able to recruit new neurons

Competition between CAs for neurons should be studied. The CA should also be able to fractionate into new CAs.

The topology of the system can also be varied. Do configurations other than a toroidal net behave differently? Can multi-layer nets (many connections within a layer and few connections between layers) be used to form hierarchies? Can areas of specialization be useful as in human brain areas?

Of course, metrics to measure the behavior of CAs should also be generated. Pearson's product moment correlation coefficient is good but it is incomplete.

This paper describes the CANT model. It is hoped that CANT's simple premises will lead to the emergence of powerful behavior, and the model has lofty goals.

This paper also explains the first systems built with the CANT model. These systems are simple but show the ability of the CANT model to generate simple CAs that behave appropriately.

A large body of thought has emerged from the Cell Assembly concept. CANT hopes to capitalize on this body of thought by implementing a sound computational model of the CA.

References

[1] Abeles, M., H. Bergman, E. Margalit, and E. Vaadia. (1993) Spatiotemporal Firing Patterns in the Fronatal 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] 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*

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

[5] Phil. 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

[6] Kaplan, S., M. Sontang, 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

[7] Limppmann, R., P. (1987) An Introduction to Computing with Neural Nets. *IEEE ASSP Magazine* April 1987.

[8] Pulvermuller, F. (1999) Words in the brain's language. In *Behavioral and Brain Sciences* 22 pp. 253-336.

[9] Rosch, E. and C. Mervis. (1975) Family Resemblances: Studies in the Internal Structure of Categories. In *Cognitive Psychology* 7 pp. 573-605.

[10] White, G., W. B. Levy, and O. Stewart. (1988) Evidence that associative interactions between afferents during the induction of long-term potentiation occur within local dendritic domains. *Proc. Natl Acad. Sci.* 85:2368-72

[11] Wittgenstein L. (1953) *Philosphical Investigations* Blackwell, Oxford.