Cell Assemblies as an Intermediate Level Model of Cognition

Christian R. Huyck

Middlesex University, London, UK,
C.Huyck@mdx.ac.uk,
WWW home page: http://www.cwa.mdx.ac.uk/chris/chrisroot.html

Abstract. This chapter discusses reverberating circuits of neurons or Cell Assemblies (CAs) derived from Hebb's [9] proposal. It shows how CAs can quickly categorise an input and make a quick decision when presented with ambiguous data. A categorisation experiment with a computational model of CAs shows that CAs categorise a broad range of patterns.
This chapter then describes how CAs might be used to implement the primitives of an symbolic cognitive architecture. It also shows how a system based on CAs is theoretically capable of fast learning, variable binding, rule application, integration with emotion and integration with the external environment.
CAs are thus an ideal mechanism for further research into both computational and cognitive neural models. Our medium to long-term plan for exploration of thought via CAs is described.
If humans use CAs as a basis of thought, then studying how biological systems use CAs will provide information for computational models. The reverse is also true; computational modelling can direct our research activity in biological neural systems.

1 Background and Introduction

I want to start this chapter with an apology. This work is about how humans (and other creatures) think. The claims often strike me as arrogant. Discovering how people think has been a major problem of philosophy for thousands of years. For me to say, "we think because of Cell Assemblies (CAs)" is arrogant, and thus we apologise. The statement is also currently backed by little evidence. I once heard Herb Simon say "if you work on an interesting problem, you will get somewhere if you answer it." Understanding thought is important, and CAs are crucial to our understanding of thought. So, I will continue with the claim that we think because of CAs, show how they can provide a sound model of thought, and provide some evidence.

Thought is extremely complex and perhaps the most complex thing that we study. One way to decompose the problem of modelling thought is to model it at different levels of granularity. For instance, Newell divides cognition into levels by time [19]. The biological band is fastest and its primitive operations take place between 10^{-4} and 10^{-2} seconds; the primitives in the cognitive band function from 10^{-1} seconds to 10^{1} seconds, the rational band from 10^{2} to 10^{4} seconds and the social band above that. Models can be built at any level.

Each band is broken into three levels. The neural level, functioning near 10^{-3}, and the neural circuit level, functioning near 10^{-2} seconds, are in the biological band. The deliberate act level near 10^{-1} is in the cognitive band.

While models can be built at different levels, each level depends on those below it. A model at the deliberate act level functions via certain assumed primitive operations. It is assumed that these primitives can be implemented in the lower levels.

Newell and other symbolicists have focused on the cognitive and rational bands [16]. These are the symbol processing levels. Anderson's ACT* [2] system works at this level but also integrates neural models as a means of showing human reaction and performance times.

For instance, Newell bases his architecture on the deliberate act level. The primitives are to assemble operators and operands, apply the operator to the operands, and to store results. Additionally, the system must support structures, input and output. Higher levels are then built upon the deliberate act level.

Why does Newell select these primitives? Can we be sure that these are the right primitives? While Newell has reasons for selecting these primitives, there is no firm basis in the neural circuit level (the level directly below the deliberate act level) that shows these things working. If some basis could be found, then it would strengthen the argument for these primitives. Moreover, it might have further ramifications on higher levels. Currently, Newell's model is a floating island; it is not firmly based on lower level structures.

Aside from the neural level, we have only indirect evidence of the behaviour of all of the other levels. We can look at the firing and activation of neurons, but we cannot see a rule being selected in the operations level. Indirect evidence is useful; reaction times, recognition times and other psychological evidence gives important direction to studies of cognitive models. If a model does not match the psychological evidence it is flawed. Still the size of the space of possible models is huge (and probably infinite).

We are still working on models of neural behaviour but we do know a great deal. Since we have this knowledge, we can use simplified models of neurons to build neural circuits. This can inform higher level models and reduce the size of the search space.

An interesting and somewhat neglected level is the neural circuit level. The neural circuit level can bridge the gap between the neural level and the cognitive band. If neural circuits can implement a model from the cognitive band, then all of the models will be more firmly based.

While we have a good understanding of how neurons work, we have a rather poor understanding of how neural circuits work. Direct analysis of neural circuits is difficult because functioning circuits (in animals) are very complex and our scanning techniques, including fMRI, PET and electrodes are not very good at examining these circuits.

D. O. Hebb proposed a neural circuit model [9], a reverberating circuit of neural cells. Hebb called this reverberating circuit a *Cell Assembly* (CA). Some physiological evidence for CAs exists (e.g. [1, 26]).

While there has been a fair amount of work in psychology [21] and neurobiology to show the existence of CAs, there has been little computational modelling of CAs [6, 10, 11, 14, 20]. The computational modelling has either been from a more abstract level than neurons [14], has been of a very limited nature focusing on associative memory [6, 10, 11, 20].

The basic CA model is a network of neurons; each neuron connects to other neurons, which in turn may come back (directly or indirectly) to the original neuron. Some neurons (e.g. rods in the eyes) are not parts of circuits; however, physiological studies show that most neurons are in some sort of circuit [24]. That is, the human brain is made mostly of circuits of neurons. Studying these circuits is crucial to our understanding of thoughts.

While there is a considerable theoretical and physical body of evidence for CAs, what exactly do CAs do? CAs have the following (theoretical) properties.

1. CAs recognise concepts.
2. CAs are composed of neurons.
3. Neurons may exist in more than one CA.
4. CAs are in working memory if and only if they are active.
5. A CA is a long-term memory item, and is formed by change in synaptic strength.
6. A CA remains in working memory for a short period ($<$ 5 sec.).
7. CAs interact with other CAs.

This chapter will describe an early model of CAs called CANT. This model is not completed and is not near to being the basis for a model at the deliberate act level; however it is hoped that the model will one day form the basis of such deliberate act level model.

The next section of this chapter explains the CANT model. This is followed by some experiments in categorisation with the model; this should be similar to item 1 in the above list of theoretical properties of CAs. This section is followed by a section showing how CAs can be activated quickly.

At this point this chapter shifts from a functioning program as a model to a theoretical model (we have not written the program yet). The sixth section shows how CAs can quickly resolve ambiguity. The seventh section shows how CAs relate to the other areas of interest for this book. This is followed by a section on CAs as an intermediate level model. The chapter concludes with a section on future work and a section of discussion and conclusions.

2 The CANT Model

A given instance of a CANT model will be a network of neurons. This network may have many CAs. That is, different input patterns will be classified in different ways, and different output patterns may be generated given input patterns.

2.1 Connection Strength and Connectivity

A CANT neuron has connections to other neurons, which are similar to connections inside biological neural systems. Connections are unidirectional (Figure 1). Like most neural net simulations, the connection strength may vary based on a local (Hebbian) 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 [17]. Current experiments work with small numbers of neurons, so they have a smaller number of connections.

Complex models of neural behaviour (e.g. [4]) consider spread of depolarization, K^{+} gating and many many other factors. The CANT model simply has neurons and connections. This simplification makes CANT computationally efficient and hopefully maintains the essential properties of the brain.

Biological neural systems are connected in a distance-biased way [24]. 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-biased connectivity.

2.2 Activation and Activation Threshold

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

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

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 weight of the connections of all the active neurons that are connected to i. This weight is the value of the connection from neuron j to neuron i. If a neuron has fired, it loses all activation from the prior time step (d=∞).

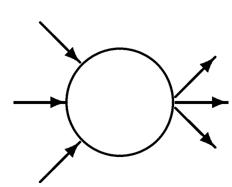

Figure 1.

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.

A neuron fires if and only if it has enough activation to surpass the activation threshold. Each neuron has the same base 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.

2.3 Decay and Fatigue

At each step, the activation of non-firing neurons decays. Of course, new activation may lead to a net gain in activation. Decay is a constant and applies only to non-firing neurons.

Active neurons fatigue. 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 $f_c t_a - t_r R_c$. 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.

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 [1]. 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.

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 neurons.

[1] Our current thoughts are that human CAs have on the order of 10^5 neurons.

4 Experimental Results

As Wittgenstein [29] 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 [23] 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 "recognise" that the concept is present in the environment.

The network in this experiment used 400 neurons in a 20x20 matrix with a toroidal topology. The network is presented with input patterns. Each series of runs started from a randomly generated net and used two patterns. The patterns were exhaustive; each neuron was a member of one of the two patterns.

The most basic type of pattern divided the network in half. Pattern A consisted of the neurons 0-199 and pattern B the neurons 200-399. These patterns were entirely local. This is the Local column in Table I.

Fully interleaved patterns had pattern A having all of the even neurons, and pattern B having all of the odd neurons; this is the Interleaved column in Table I. Intermediate patterns combined these; for example pattern A consisted of neurons 0-99 and the even neurons from 100-299, pattern B consisted of neurons 300-399 and the odd neurons from 100-299. This is the Half column in Table II.

The stimulus pattern is presented for 10 cycles. The pattern consists of 20 neurons randomly chosen from the pattern type. Two different A patterns might share no common neurons.

Table I: Network Correlations Exp. 1

	Local	Half	Interleaved
Number Runs	300	350	1500
A-A Corr.	1	.9286	.9900
B-B Corr.	1	.9492	.9917
A Self Corr.	.9970	.8821	.8621
B Self Corr.	.9734	.9562	.9386
A-B Corr.	-1	-.4973	-.9826

The measurements are Pearson's product correlation coefficient. This shows that the neurons activated when one A pattern is presented is highly correlated with a different A pattern being presented (the A-A row) and when two B patterns are presented (the B-B row). Both are measured 5 cycles after the stimulus stops being presented. Similarly, the A and B patterns are negatively correlated (the A-B row). The pattern of activation is also maintained. The A and B self correlation rows show the correlation between active neurons at cycle 5 and 20 cycles after the end of stimulus presentation.

This experiment shows that this model is capable of learning CAs over the full range of exhaustive patterns. Local patterns, interleaved patterns, and patterns which combined local groups and interleaved groups all formed CAs. These patterns are unique, persist and are reliably activated.

It takes longer to learn interleaved patterns, but they are eventually learned. This is because a distance-biased network helps localised CAs form. Distance-biasing acts as an attractor and when a localised pattern is placed into that attractor, it is easily learned. Patterns that fight the attractor can still be learned but take longer, and are less successful.

Other experiments have been done, but this experiment shows that this model is capable of recognising and categorising different types of patterns. This is the primary function of CAs. From this, more complex behaviour can be generated.

5 Quick Activation of CAs

The theory of CAs states that a CA is quickly activated [14]. This shows that slow neurons can quickly recognise an object.

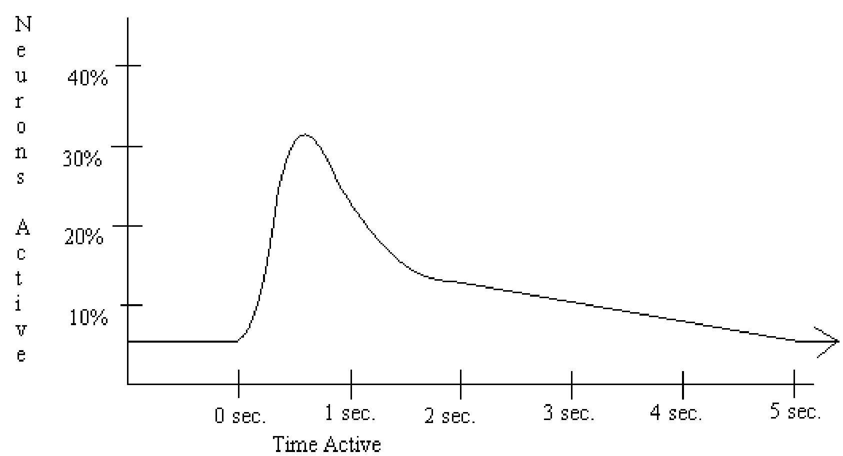

Figure 2.

Experiments using simple computational models of CAs shows this to be the case [11]. This model shows that activation of 10 neurons can lead to activation of the entire CA of 200 neurons in the next time step. A time step is meant to model 10 ms. of real neural activity. This particular experiment is probably not viable from a neurological standpoint as the size of the CA is very small. However, it does show how quickly a few neurons can lead to activation of many more neurons.

A real CA would probably cross several brain areas. However, it would only take a few time steps (tens to hundreds of milliseconds) to activate the entire assembly. Highly parallel activation compensates for slow neural speed. Practically, it will be easy to get large numbers of neurons activated within the speeds that are needed for normal real-time performance.

Figure 2 shows the activation curve of a CA. At time 0, a stimulus is presented to the network. Rapidly, a large percentage of the neurons in the CA become activated and this is associated with recognition. Stimulus may be removed or remain present. Some neurons in the CA are active for quite some time meaning the CA can spread information, strengthen itself (via the Hebbian learning rule) and remain in short-term memory. Note that no neuron remains active for the whole period; they become active, fatigue, recover and are reactivated by other neurons in the CA.

6 Quick Decision on Ambiguous Data

A human brain must have millions of CAs, but our model networks are much smaller. A given net may have several CAs in it, but only one or a few should be active at a given time. What happens when a net is presented with ambiguous data?

When ambiguous data is presented to a net, some of the neurons from two separate CAs will be activated. These will tend to activate other neurons in their own CAs. Since the two CAs have never been active together and reside in the same area of a distance-biased network, they will tend to inhibit each other. Thus a competition ensues between the two ambiguous CAs. One CA wins, and the data is recognised as an instance of that type of object.

When the net is presented with ambiguous data it does take slightly longer to recognise the object. This is due to the extra time needed for competition. However, it should only take a few more cycles to recognise the object, and thus is almost as fast as recognising an object from unambiguous data.

Figure 3 shows the activation pattern of two CAs. Both get an initial burst of activating neurons. Some of these neurons are inhibitory neurons and inhibit neurons from other CAs, thus inhibiting the other CA. The CA represented by the bold line loses the competition and returns to its base firing rate. Of course this is all theory. We do not know of any model of a CA that handles ambiguous data [2].

[2] We are currently working on such an experiment based on the CANT model and the results are promising.

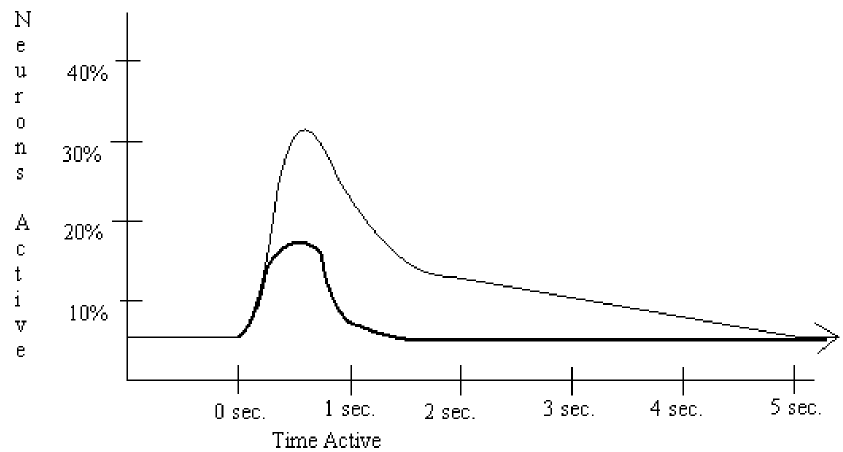

Figure 3.

7 The Other Areas

CAs categorise and they do it quickly. This chapter has described how the CANT model does this. CAs also provide many other properties that are very useful for a computational model of intelligence. They are robust, they learn in context and they enable synchronisation of neural firing. Additionally, they provide an excellent landmark for asking questions about modularity and timing.

Like most ANN architectures, CAs are robust. Loss of neurons might even strengthen a given CA. Since a CA is composed of a suite of neurons, it can handle the loss of neurons; new neurons can be added and incorporated into CAs. The dynamics of CAs recruiting new neurons and fractionating into multiple CAs is not well understood, but the process must be robust.

CAs learn in context. The CA learning mechanism is unsupervised. A given network will spontaneously create CAs based on the input that is seen. Thus all learning is in context. This may lead to unforeseen connections between concepts, but those connections are based on the input that has been presented.

CAs are self synchronising. Neurons in a CA tend to fire in a similar pattern [20]. Thus CAs actually act as a synchronising mechanism. The mechanism also leads to a number of questions about how CAs could work together. There are questions about variable binding [7]; co-operating CAs may be bound together by synchronisation. How is this done? How do CAs combine into larger structures like sequences and cognitive maps [13]?

CAs also provide a landmark for questions of modularity. Braitenberg [5] notes that the brain is largely a uniform mass of neurons. CAs span brain areas [21]. This enables cross modular co-operation, yet intra-modular specialisation. A working CA model could explore lots of questions about modular communication.

Finally, animals have to do things by a certain time. CAs must be able to function under deadline. CAs function rapidly but they still show no way to solve problems by a certain time. CAs provide a mechanism for building complex systems to solve problems. If CAs can be used to generate a goal mechanism with interrupts, then the deadline problem can be addressed.

8 An Intermediate Level Model

A good intermediate level model[3] would support the primitives of Newell's deliberate act level [19]:

1. Get Input
2. Select Operators and Operands
3. Apply the Operator
4. Store the Results
5. Generate Output
6. Support Structures

The CANT model already has demonstrated support for input, operator and operand selection, and storing results. It holds promise for operator application, output generation and more complex structures.

The CANT model supports direct input. In biological systems, sensory organs activate neurons. In a CANT network, we could connect sensory organs directly to the neurons. The sensory system and underlying neural system are very complex and well studied, and the neural system is largely compatible with the CANT model.

Operators and operands can be selected based on current input and prior input. CA activation can be thought of as operator and operand selection; an operator and operands are selected if and only if their CAs are activated.

Results are stored by either putting them in working memory or in long-term memory. To put them in working memory an existing CA has to be activated. To put them in long-term memory, a new CA must formed. This is done by changing synaptic weights.

Output can be generated by effectors. Like biological input, biological output is done by the body. Muscles move and vocal chords vibrate. The underlying neural control mechanisms, though not entirely understood, are well studied. Again, these are compatible with CA theory and CA based controllers.

Structures are more complex for CAs to generate and have many of the questions in which we are currently interested. How do CAs build, for instance, a verb frame? Weak connections between CAs must be quickly learned via the Hebbian learning mechanism. We do not currently have a functioning computational model of this process, but this is in our medium term plans.

Operators are more complex in a CA model. Simple operators like *store* have already been implemented. More complex operators like move forward or

[3] a neural circuit model

multiply two numbers are more difficult. These complex functions may be implemented by decomposition into primitives. Primitive operators are done by the output mechanisms and storage mechanisms described above. Still the functions need to compose these primitives via CAs. This composition is done via complex structures like sequences and cognitive maps. As before, these are learned by creating weak links to combine CA primitives.

We like Newell's division into levels and are particularly interested in Soar; thus we have focused our argument in that direction. However, there is nothing sacred about the Soar model. Newell's deliberate act primitives may turn out to be wrong. What is important is that the higher level behaviours that can be built on the deliberate act primitives can be built from CAs. If it turns out that humans build the behaviours differently, CAs should be flexible enough to account for that behaviour. If we can implement Newell's system than CAs will have implemented a strong system.

Additionally, the CANT model may support other aspects of behaviour including emotion, attention, fast learning and forgetting. Emotion can be incorporated using a pleasure/pain mechanism. This interacts particularly well with a Soar-like goal mechanism. Unappealing and dangerous CAs are associated with (connected to) pain, a pleasurable CAs to pleasure. This can direct behaviour.

Attention can be explained with a more global mechanism. CA theory, as presented so far, is entirely localist. There is no global mechanism. However, CAs are connected to other CAs. An attention area or cognitive map could be connected to all other areas. It could focus attention via a global inhibitory mechanism. This would allow only the most active CA to remain active.

Fast learning may be accomplished by reverberation, support from cognitive maps, and attention. These aids can support more neural pair coactivations and thus quick strengthening of connections.

We also forget. CAs can account for this. Forgetting is done by a change in synaptic strengths. Core concepts, CAs, are rarely forgotten (you do not forget what a dog is). This is because the CAs occasionally become active and their synaptic strength is reinforced. Connections between concepts are more likely to be forgotten because they are initially weaker and these connection are also used to connect other CAs.

Finally, the CANT model supports a blackboard like architecture [8, 22]. CAs consist of neurons that are closely connected but connections to other CAs exist. This means that information from a wide range of CAs may be brought to bare on a single problem.

A well known example of a blackboard system is the Hearsay speech understanding system [22]. Hearsay consists of several subsystems. Each subsystem can communicate to other subsystems via a blackboard. A special subsystem chooses which subsystem will function next based on which has sufficient information to function. In a CANT blackboard, the architecture would decide which subsystem or subsystems would function based on enough information being available. Similarly information is exchanged via neural activation.

In traditional software systems we break the system into subsystems or into components. This facilitates engineering. In the brain, this really does not happen. Despite physiologists division into brain areas, each area is closely connected to other areas. This whole system is tightly coupled. This tight coupling may not facilitate engineering, but the brain was not engineered.

Existing CA systems are associative memory systems; this includes the experiments described in this chapter. What is needed is a model that can combine CAs in an interesting way [12].

Von der Malsberg [18] proposes dynamical connections as a variant that is much more flexible than standard CA theory. CAs are bound together via synchronous firing. This enables new concepts like *blue square* to be formed quickly and temporarily from primitive concepts like *blue* and *square*.

An unappealing alternative to this is that variable binding is done by the change of synaptic strength. This is more tenable when overlapping set coding [28] is considered. A given neuron participates in more than one CA. So, a given neuron might participate in the *blue* CA and the *square* CA. When the two CAs are coactivated, a new *blue square* CA is formed. This is unappealing because of learning time constraints and forgetting.

However, a middle ground exists. CAs are formed in the traditional way and represent a "strong force" (e.g. *square* and *blue*). Variable binding is done via synchrony and is a short term effect just like CA activation is a short term effect. If the binding is important or repeated, Hebbian learning leads to an association between CAs; this is a "weak force". These weak forces are available for sequences, hierarchies and cognitive maps of CAs. This "weak force" is facilitated by overlapping set coding. This may be similar to the SMRITI system [25].

The important point is that both synchronous firing and overlapping set coding are compatible with CAs in general and the CANT model specifically. Synchronous firing emerges from the activation of neurons [20]. Overlapping set coding also emerges from the input and learning mechanisms.

9 Future Work

The long-term goal of this work is to move from neural models of categorisation to neural models of processing. Currently, we are working on a model of categorisation and associative memory. This work duplicates, and hopefully extends, existing computational CA models.

Next we hope to model interactions of CAs. This includes competition amongst CAs, hierarchies of CAs, sequences of CAs, and maps of CAs. This will be done via overlapping set encoding and synchronous firing patterns.

Variable binding can lead to rule activation. If A and B are active, then complete the pattern and activate C. Rule activation provides the basis for traditional symbolic architectures. One popular symbolic cognitive architecture is Soar. It can be described as an expert system, with goals, operators, operator subgoaling and chunking. If all of these can be implemented in a CA based system, then CAs will have provided an intermediate level for cognitive modelling.

A system like CA-Soar would escape from the symbol grounding criticism of Fodor and Pylyshyn [3]. It would direct psycho-neurological research. It would provide a plausible model for thought and thus for real Artificial Intelligence.

Implementing this set of performances is not straight forward, but one can easily imagine that it is possible. We suspect that this CA-Soar model would be more dependent on categorisation than current Soar programs. This reflects the associative nature of the brain.

10 Discussion and Conclusion

This chapter has shown that CAs can be used to classify a wide range of patterns. It has also argued that CAs can be expanded to account for variable binding, hierarchy and cognitive maps. This functionality will enable CAs to provide the primitives for Newell's deliberate act level. Thus CAs can form a bridge between the relatively solid knowledge that we have of neural behaviour to the highly developed work we have on symbolic cognitive architectures.

One may ask why CAs have not yet been used to build a symbol system. It is likely that the recent interest in connectionist systems [27] has laid new foundations for this work. Additionally, computational speed has only recently enabled large neural models to be run.

CAs can be used to store memories that are based in reality. They can solve the symbol grounding problem. Current computer memory can store the sentence "the cat chased the mouse". When asked "what was being chased" it can respond, but it can not respond to "why was he being chased".

Traditional computer memory is great for number crunching. CA models may be faster than biological CAs but will still be slower than Von Neumann architectures. CAs however function in context and can generate statistically reasonable answers from little data. They will function better than Von Neumann architectures in AI tasks like Natural Language Processing and Decision Support Systems.

This book is derived from the Third International Workshop on Current Computational Architectures Integrating Neural Networks and Neuroscience. The key question for the workshop was *What can we learn from cognitive neuroscience and the brain for building new computational neural architectures?* In relation to this chapter, we would like to break that question into two questions:
1. *What can we learn from cognitive neuroscience and the brain for building CAs?*
2. *What can we learn from CAs for building new computational neural architectures?*

It is simple to say what the brain can tell us about CAs. The way each neuron behaves, and the way they are connected in the brain is the way CAs should behave. Any CA model will be simplified as modelling a single neuron is very complex. Perhaps we can simplify the model by simply modelling changes in neural activation in discrete steps and learning by changes of strength. This is not as robust as modelling neural activation continuously with ionic transfer between neurons, and modelling changes of strength by changes in axonal radius, myelination and geometric properties of the synaptic cleft. This simplification makes the model run faster which should enable us to model the large number of neurons needed for CAs. Still it is a simplification and, as such, may miss important data.

It is more difficult to say what CAs can tell us about building new computational neural architectures. CAs are models for concept storage. They can solve the symbol grounding problem. Perhaps the most interesting question is in what way can CAs work together.

At heart, CAs are a model for data storage. They have several advantages that could be transferred to other models. They remain active for a certain period of time. They are a model for both long and short-term memory. They can be used to choose between equally likely solutions. It is the reverberating activity that makes this model different from most existing ANN models.

Further studies to show how CAs can store more complex structures and process data are important. The class of reverberating circuits is large and we have done little in studying this class. Clearly, the brain is not just a bunch of CAs. For instance, non-local effects [15] probably are important. However, CAs are crucial and we need to understand them and how they work.

Using CAs as a basis for more complex phenomena puts us on reasonably solid ground for future exploration of how the brain works at more complex levels. This in turn leaves us with a better understanding of intelligence and thought.

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. Anderson, John R. (1983). *The Architecture of Cognition* Cambridge, MA: Harvard University Press.
3. Bechtel, W. and A. Abrhamsen. (1991). *Connectionism and the Mind* Blackwell Publishers.
4. Bower, J. and D. Beeman (1994). *The Book of Genesis* Springer-Verlag.
5. 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.
6. Fransen, E., A. Lanser, and H. Liljenstrom. (1992) A Model of Cortical Associative Memory Based on Hebbian Cell Assemblies. In *Connectionism in a Broad Perspective.* Niklasson, L. and M. Boden eds. Ellis Horwood Springer-Verlag pp. 165-71
7. Fujii, Hiroshi, I. Hiroyuke, K. Aihara, N. Ichinose and M. Tsukada. 1996. Dynamical Cell Assembly Hypothesis - Theoretical Possibility of Spatio-temporal Coding in the Cortex. In Neural Networks Vol 9:8 pp. 1303-50
8. Hayes-Roth, B. (1985) A Blackboard Architecture for Control. In *Artificial Intelligence* 26 pp. 251-321.
9. Hebb, D.O. (1949) The Organization of Behavior. John Wiley and Sons, New York.

10. 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
11. Huyck, Christian R. (2000). Modelling Cell Assemblies. In *Proceedings of the International Conference of Artificial Intelligence* Las Vegas, Nevada
12. Ivancich, J. E., C. Huyck and S. Kaplan. 1999. Cell Assemblies as Building Blocks of Larger Cognitive Structures. In *Behaviour and Brain Science.* Cambridge University Press
13. Kaplan, S., M. Weaver, and R. French (1990) Active Symbols and Internal Models: Towards a Cognitive Connectionism. In *AI and Society* 4:51-71
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. Katz, P. S. (1999) What are we talking about? Modes of neuronal communication. In *Beyond Neurotransmission: Neuromodulation and its Importance for Information Processing* Paul Katz ed. Oxford Press. pp. 3-30
16. Laird, John E., A. Newell, and P. Rosenbloom. (1987) Soar: An Architecture for General Cognition. In *Artificial Intelligence* 33,1
17. Lippmann, R. P. (1987) An Introduction to Computing with Neural Nets. *IEEE ASSP Magazine* April 1987.
18. Malsberg, C. von der. (1986) Am I Thinking Assemblies? In *Brain Theory.* Palm, G. and A. Aersten eds. Springer-Verlag pp. 161-76
19. Newell, Allen. 1990. *Unified Theories of Cognition* Cambridge, MA: Harvard University Press.
20. Palm, G. (2000) Robust indentification 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.
21. Pulvermuller, F. (1999) Words in the brain's language. In *Behavioral and Brain Sciences* 22 pp. 253-336.
22. Reddy, R., L. Erman, R. Fennell, and R. Neely. (1976) The HEARSAY speech understanding system: An example of the recognition process. In *EIII Transactions on Computers* C-25: 427-431
23. Rosch, Eleanor and C. Mervis. 1975. Family Resemblances: Studies in the Internal Structure of Categories. In *Cognitive Psychology* 7 pp. 573-605.
24. Schuz, A. (1995) Neuroanatomy in a Computational Perspective. In *The Handbook of Brain Theory and Neural Networks.* Arbib, M. ed. MIT Press pp. 622-626.
25. Shastri, L. (2000) SMRITI: a computational model of episodic memory formation inspired by the hippocampal system. In *EmerNet: Third International Workshop on Current Computational Architectures Integrating Neural Networks and Neuroscience.* Wermter, S. ed.
26. Spatz, H. (1996) Hebb's concept of synaptic plasticity of neuronal cell assemblies. In *Behavioural Brain Research* 78 pp. 3-7.
27. Sun, R. (1995) Robust reasoning; integrating rule-based and similarity-based reasoning. In *Artificial Intelligence* 75 pp. 241-95.
28. Wickelgren, W. A. (1999) Webs, Cell Assemblies, and Chunking in Neural Nets. *Canadian Journal of Experimental Psychology* 53:1 pp. 118-131
29. Wittgenstein L. (1953) *Philosophical Investigations* Blackwell, Oxford.