The Recommendation Architecture Model for Human Cognition

L. Andrew Coward
Department of Computer Science
Australian National University
ACT 0200
Australia
landrewcoward@shaw.ca

Abstract

A model for human cognitive processing is described. The advantages of the model are that it is able to learn complex combinations of capabilities with limited information recording and processing resources; it can bootstrap its memory and cognitive capabilities from experience with very limited, genetically plausible a priori guidance; and modules in the model resemble physiological structures in the brain. In the model, all information recorded or activated in the cortex is perceptually, cognitively and behaviourally ambiguous. Cognitively complex processing occurs within populations of ambiguous information, and only achieves unambiguous meanings in subcortical structures. The model can account for a wide range of cognitive phenomena with a limited range of information recording and access mechanisms, as illustrated by a detailed discussion of working memory phenomena. However, the need to limit resources means that modules in the model are defined as collections of similar system operations and do not correspond with cognitive features or categories. Descriptions of cognitive phenomena are therefore more complex than "user manual" type models, but "user manual" type models are not capable of providing an understanding of these phenomena in terms of physiology. An electronic implementation of the model confirms its viability.

1 Introduction

An operational system is one which achieves objectives by controlling a physical system that interacts with an environment. A simple electronic example is a refrigerator thermostat. An operationally complex system is one in which the number of control features is large relative to the available information recording and processing resources. An example is the central office switch that controls an extensive telecommunications network in real time with no human intervention. Such a switch may have of the order of ten thousand separate but interacting features that provide high reliability telecommunications services to over 100 thousand telephones and computers [Nortel Networks, 2001].

An operationally complex system does not calculate mathematical functions in any useful sense. Rather, it detects conditions within the information space available to it and associates different combinations of conditions with different behaviours. Its available information space includes information derived from the past and present states of the environment, of the physical system being controlled, and of the control system itself. A condition is defined as a small set of information elements from the information space, each with a corresponding value. A condition occurs when all (or a significant subset) of the information set defining the condition has the values specified for the condition.

A feature is a consistent way in which a system responds to a set of similar environmental circumstances. The environmental circumstances and corresponding responses are similar from the point of view of an external observer, and are a useful way for such an observer to understand the system, but as discussed below may not reflect similarities in the way the system detects conditions in its environment and generates behaviours on a more detailed level.

Operational complexity results when there is tension between the high number of features needed and limitations to the information storage and processing resources available. This tension results in severe constraints on system architecture. As the ratio of features to resources increases, a system will be constrained more and more tightly within specific architectural bounds. There are two different sets of bounds, one set applying when features are defined under external intellectual control, the other when features are learned with limited a priori and external guidance [Coward 2000]. It is plausible that natural selection will have favoured brains that can perform the largest number of features with the least resources. Selection pressures will therefore have constrained brains which learn a complex combination of features within the appropriate architectural bounds, called the recommendation architecture.

One consequence of these constraints which applies in both sets of architectural bounds is that there are major differences between system architecture type descriptions and user manual type descriptions of the system.

A system architecture describes how system resources are organized so that features can be performed efficiently. A user manual describes how features work in a way which can easily be understood by an outside observer. To illustrate the origins of the major differences between the system architecture and the user manual, consider how information handling resources can be used efficiently. The system architecture separates operations into groups of similar operations. A module is defined for each group of operations. Such a module contains information storage and processing resources optimized for the type of operations in its group. A module can therefore use the same resources to efficiently perform a substantial proportion of its operations, economizing on the resources required by the system as a whole. Note that the definition of "similar" in this context depends upon the types of detailed units available to perform the operations, which for an electronic system includes transistors and machine instructions.

Because system modules are organized on this basis, any one feature will require operations by many modules, and any one module will provide operations for many features. Hence tracing the performance of a feature through the system architecture is a very complex process.

For example, in the central office switch there are no modules on any level which correspond exactly with user manual features like conferencing or call forwarding, or even with major groups of features like processing calls, performing self diagnosis, or collecting billing information. At a detailed level, the management of a telephone connection between two users, from originating a connection through answering the call to disconnection of both users, is managed by repeated invocation of modules performing functions with names like queue handler, task driver, event router, function processor, or arbitrator, and differences in the features invoked are only reflected in the information provided to these modules and/or the order in which they are invoked. Major changes to the system architecture of such a system driven by changes to the available memory and processing technologies may result in minimal or no changes to the user manual.

In an operationally complex system which learns, modules perform groups of operations concerned with the definition and detection of conditions. Some modules determine when and where changes will be made to the conditions detected. Other modules detect groups of similar conditions relevant to determining behaviour. Yet other modules improve the consistency between different conditions detected within the same input state. Any one condition will be relevant to many different behaviours, and the relationship between observed cognitive features and the system architecture of the brain can therefore be expected to be very complex.

Any attempt to implement a system in which system modules corresponded closely with cognitive categories and processes would therefore require a very high level of resources. Even if the attempt was not impractical, the result would have a system architecture with very little resemblance to that of the brain being modeled, even if cognitive behaviour was similar. Furthermore, although cognitive behaviour might be similar in general terms, at some level of detail mismatches would appear in the interactions between features unless the resources employed were extreme.

This paper first summarizes theoretical arguments that as the ratio of features to resources increases, a system will be constrained more and more tightly within specific architectural bounds. Secondly, an overview of the design of a system within the recommendation architecture bounds is provided, and reference made to an electronic system which has been implemented within those bounds.

Thirdly, a cognitive architecture within the recommendation architecture bounds with the capability to support complex cognitive processing is described. This system uses the mechanisms which have been demonstrated in the electronic implementation.

Fourthly, the ways in which various cognitive phenomena, and in particular working memory, result from the operations of the recommendation architecture model are described. It will be argued that many psychological models of cognition are actually "user manual" type descriptions, and descriptions of the same phenomenology in system architecture terms are much more complex. However, system architecture descriptions are the only way to relate cognitive features to the underlying physiology.

Finally, the way in which the recommendation architecture relates to some current models for the relationship between visual working memory and awareness is discussed.

2 Architectural constraints on operationally complex systems

This section provides an overview of theoretical arguments set out in more detail in Coward [2000; 2001]. As mentioned earlier, the need to economize on resources forces any sufficiently complex operational system into a modular form. More specifically, modules are organized into a modular hierarchy in which each module detects a set of similar conditions. Conditions are similar if a high proportion of the information defining them is drawn from a limited information set in which information sources and values are specified and each source often has its corresponding value at times when other members of the set have their corresponding values. Detailed modules detect sets of very similar conditions and can therefore share a high proportion of detection resources across many conditions, and generate a limited range of outputs which indicate the presence of a set but do not need to indicate the actual identities of the conditions within the set which are present. A higher level module is made up of a group of detailed modules and detects all the conditions detected by the group. These conditions have a somewhat lower overall degree of similarity and therefore a somewhat lower but still significant degree of resource sharing can occur.

Similarity of conditions in this information sense does not guarantee similarity of behavioural meaning. Modules on any level will therefore detect conditions which are relevant to a number of different features, and not correspond with individual features. One problem with this lack of correspondence is that changes to conditions needed to support some desirable feature modification may have undesirable side effects on other features. The modular hierarchy must therefore achieve a compromise between the conflicting needs of resource economy and modifiability. In this compromise, the number of features influenced by one module is limited as far as possible consistent with the constraints imposed by resource limitations. Such a compromise is implemented by limiting information exchange between modules as far as possible, where an information exchange occurs when a condition detected by one module is provided to and utilized by another module.

Such an information exchange between two modules also has an operational meaning to the recipient module. This operational meaning is that the currently appropriate system behaviour is limited to a specific subset of the behaviours influenced by the recipient module. Operational meanings can be of two types. One type is unambiguous: the appropriate behaviour is limited to the subset with 100% confidence. In this case, exchanges can be viewed as commands and the need to maintain such unambiguous meanings forces a system to separate into the memory, processing von Neumann form. However, when features must be changed on the basis of consequence feedback, sustaining unambiguous contexts is impractical. The other type of meaning is ambiguous: the appropriate behaviour is probably limited to the indicated subset. Such exchanges can only be viewed as recommendations, and in general many such recommendations will be required in support of any accepted behaviour. The need to sustain ambiguous but adequately meaningful contexts forces a system to separate into a clustering subsystem which defines and detects conditions and a competition subsystem which associates those conditions with appropriate behaviours. Consequence feedback can guide changes to features, but only by changing the associations between conditions and behaviours (or condition recommendation weights), not by changes to the conditions themselves. This primary separation required by the recommendation architecture bounds is illustrated in figure 1.

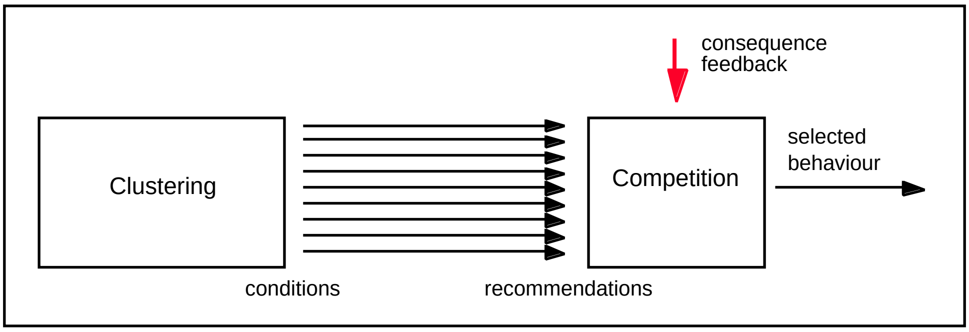

Figure 1 The primary separation in the recommendation architecture. Clustering is a modular hierarchy which defines and detects conditions within different ranges of complexity. Many of the detected conditions are used internally by clustering to manage when and where new conditions will be recorded. Some condition detections are communicated to competition. Any one such detection is interpreted by competition as a range of different behavioural recommendations with different weights. Competition selects and implements the behaviours with the strongest weights, and adjusts those weights on the basis of consequence feedback. Such consequence feedback does not change the definition of the detected conditions.

Another factor limiting architectural form is the need to detect conditions within sets of information with the appropriate temporal relationships. For example, if a system received input information from two environmental objects at different times, it must be possible to distinguish between conditions detected within the input information derived from the different objects, and not detect conditions with information derived from different objects in a confused fashion. Conditions detected by one module may be incorporated into conditions detected by other modules, and because modules may take different times to detect and communicate their conditions, temporal mismatches can occur.

There are two possible approaches to maintaining synchronization: global and local. In the global approach a snapshot of information at one time is stored in a reference memory and conditions detected by different modules within the information in this snapshot. Such conditions are then recorded in the reference memory. In the local approach, modules are arranged in layers with each layer detecting conditions only within conditions detected by the previous layer. Although the layering approach requires fewer resources than the global approach, the risk of differences in module processing times within the same layer means that it will in general only be appropriate if information is partially ambiguous.

3 The recommendation architecture

This section summarizes the detailed design descriptions in Coward [2001]. As discussed earlier, there is a major separation in the recommendation architecture between clustering and competition. Clustering defines and detects conditions within the available system information, and competition associates different combinations of conditions with different behaviours. A module in clustering detects a set of conditions. Module outputs indicate the detection of one or more conditions in its set, but do not indicate the exact identities of the conditions. A component in competition corresponds with one behaviour or type of behaviour, and interprets its inputs from clustering as recommendations for or against that behaviour. Competition components compete in a hierarchy of cross inhibition to determine the most strongly recommended behaviours.

When conditions must be defined heuristically, any change to a condition after definition will tend to result in uncontrollable side effects on any behaviours dependent on detection of the original condition. To limit these side effects, a module can add similar conditions to the set which it detects, but with some very limited exceptions cannot change or delete conditions. The effect of adding a condition which is similar in an information sense to existing conditions will generally be a slight increase in the behavioural ambiguity of module outputs, which can be tolerated because any accepted behaviour is recommended by the outputs of many modules.

The circumstances under which additional conditions are recorded must be tightly managed to avoid both excessive condition recording and excessive increase in behavioural ambiguity. This management is itself an operationally complex task which must be controlled by condition detection. As far as possible, new conditions which are identical with existing conditions should not be recorded, groups of conditions should overlap as little as possible, conditions should only be recorded if there is probable operational value in such recording, and modules should only record conditions which are not too different from previously recorded conditions. Much of the structure of clustering is concerned with achieving these objectives.

Within clustering, the most detailed modules are devices which record groups of very similar conditions. These devices are arranged in layers in which each layer derives its condition defining inputs from one preceding layer, although there are inputs from other layers for management purposes as discussed later. A key reason for this layering is to ensure that conditions are detected within temporally consistent sets of sensory inputs, such as inputs ultimately derived from one sensory object. In addition, the conditions detected in one layer will be within one range of complexity, where the complexity of a condition is the total number of sensory inputs that contribute to the condition, either directly or via intermediate conditions.

Conditions within different ranges of complexity may be more appropriate for recommending different types of behaviour such as externally directed behaviours or condition recording. "Appropriate" in this context means that conditions within one range of complexity have a lower behavioural ambiguity with respect to different behaviours of a particular type than conditions within other ranges of complexity. Differences in behavioural ambiguity between randomly defined conditions at different complexity levels has been demonstrated by electronic simulation [Coward 2001]. For example, conditions with complexities of the same order of magnitude as visual object perceptions will tend to be less ambiguous with respect to categorization of visual objects than conditions on other levels of complexity, even though no conditions on any level correlate unambiguously with such categories.

A higher level of module is the column, made up of groups of devices on each of a sequence of layers. A primary role of a column is to manage the circumstances in which condition recording can occur. A column detects all the conditions recorded on its constituent devices, but only a subset of those devices communicate detections outside the column. This subset is within ranges of complexity most likely to be behaviourally useful outside the column. Many of the other conditions are used to define when and where new conditions will be recorded within the column.

Groups of columns spanning the same sequence of device layers form array modules. A primary role of an array is to detect some conditions in every overall input state from a particular domain, and therefore generate some recommendations in response to every such input state. An array adds conditions and/or columns to generate such detections if necessary, by a process which uses conditions detected by columns to ensure that the overlap between conditions detected by different columns in the array is as small as possible. There are also mechanisms to add columns if an array has inadequate discrimination between similar but behaviourally different system input states.

An area module is made up of a series of array modules. A primary role of an area module is to manage the consistency within a population of activated conditions, by inactivating conditions which are not consistent with the majority of other active conditions as described in Coward [2000]. After consistency has been achieved, outputs from the area generated by the appropriate arrays are provided to competition.

3.1 Information conditions and portfolios

The substrate on which conditions are recorded in the recommendation architecture model is the pyramidal neuron in the cortex. The implementation of a condition is a group of inputs from the preceding layer which can cause the neuron to fire. Many similar conditions are recorded on one neuron, and other inputs to the neuron can excite or inhibit the recording of new conditions. Conditions are recorded permanently, in other words, once a condition is recorded, the neuron will always detect any repetition of exactly the same condition. If there are ~ 10^{10} such neurons, each with ~ 10^4 synapses, and ~ 50 active synapses cause a neuron to fire there could be well in excess of 2 x 10^{12} conditions. For a lifetime of about 70 years, ~ 1000 conditions per second could be recorded. The point of this very rough estimate is to bring out the contrast with artificial neural network models in which neuron responses are determined by continuous adjustment to input weights and a neuron will not necessarily respond to an exact repetition of an input combination to which it previously responded. However, algorithms more similar to classical artificial neural network algorithms are used in competition (i.e. subcortical structures) to define and adjust recommendation weights.

The criterion for condition recording is that a population of conditions above a minimum level must be detected at all times. This criterion is operationally equivalent to a requirement that a range of behavioural recommendations above a minimum level must be generated in response to every input state. Actual condition recording rates will therefore vary considerably depending on the degree of novelty in current experience.

The most useful conditions for behavioural purposes are not known a priori, and there is therefore a significant random element in the condition selection process. Although various biases can be applied to improve the probability that selected conditions are as behaviourally useful as possible, the presence of the random element means that conditions will not correlate unambiguously with, for example, cognitive categories. In other words, one condition may be present when a number of different categories are present, and its absence does not conclusively demonstrate the absence of any one category. However, the presence of a specific condition increases the probability that a specific, limited range of categories are present, and the presence of a range of conditions can be used to indicate the category most probably present.

Individual raw sensory inputs (such as individual retinal cell outputs) have very weak discrimination between the presence of different types of behaviourally relevant objects. These raw inputs are first combined and preprocessed into sensory inputs which discriminate somewhat more strongly between such objects. For example, retinal cell outputs may be processed into signals correlating with images on the retina independent of the size and position of the image on the retina and with separate information on distance to the object. A somewhat oversimplified way of understanding the definition of information conditions is that one condition corresponds with a specific set of these preprocessed sensory inputs, being present to a specified degree.

Such conditions are recorded on devices, and the presence of a relevant sensory input reaches a device either directly or via intermediate conditions recorded on other devices. The recording of a condition is indicated by activation of the device on which it is recorded. Any subsequent exact repetition of the condition will again activate the same device. Multiple similar conditions are recorded on the same device, and activation of the device therefore indicates the current presence of one or more of the conditions within the recorded set.

This simple view of conditions is made considerably more complex because a device can be activated not only by the presence of its conditions within sensory inputs, but also indirectly by two other types of mechanism. One mechanism is that a device can be activated if devices are already active in a number of other modules that have often been active in the past at the same time as the module in which the device is located. The other mechanism is that a device can be activated by the activity of a number of modules which recorded conditions at the same time in the past as the module in which the device is located. In both cases there is a strong tendency for the activity of one module to activate the other immediately after the simultaneous activity or recording, but unless such an indirect activation actually occurs, this tendency declines fairly rapidly with time. However, if an indirect activation occurs in the course of generating a useful behaviour, the decline is reduced, and if such useful indirect activations occur frequently the tendency stabilizes or increases.

These indirect activation mechanisms can be viewed as supplementing the conditions detected within current sensory inputs with other conditions which have a significant probability of being relevant to determining the most appropriate current behaviour. For example, conditions in modules which have been active in the past at the same time as modules containing currently present conditions may contain information about the current environment which cannot currently be observed [Coward & Sun 2004]. However, indirect activation is a behaviour which must be accepted by competition before it can occur. The recommendation strengths in favour of such indirect activations depend provisionally upon the frequency and recency of past simultaneous activity and longer term on the behavioural value derived from previous actual indirect activations.

A newly recorded condition is made up of a set of currently active component conditions. Some of these component conditions may be combinations of currently present sensory inputs, and some could have been activated by one of the two indirect mechanisms. Both the definition of conditions in terms of sensory inputs and the relationship between sensory inputs and the resultant pattern of condition activation can therefore become very complex.

A condition is physically defined by a set of inputs on to its device from other devices or from sensory inputs. These inputs in practice need to be established in advance of the occurrence of the condition to be recorded. Provisional conditions will therefore be physically defined by a group of provisional inputs. A new condition will be defined as the subset of these provisional inputs which is active at the point in time when other devices are encouraging the definition of a new condition on the device. To ensure adequate similarity to existing conditions on the same device, the selection of provisional inputs is biased in favour of inputs which have often been active in the past when the device has been active, or when other devices in a higher level module to which the device belongs have been active.

Simultaneous activity is the primary source of information available for definition of conditions without a priori knowledge. The definition of conditions in the recommendation architecture uses information on simultaneous activity in a number of ways. Firstly, provisional conditions are defined by random selection within a population of component conditions which have tended to occur at the same time in the past. Secondly, permanent conditions are defined by simultaneous occurrence of all their components. Thirdly, simultaneous activity of two condition devices creates a tendency for one device to activate the other at a later point in time. This third way is very similar to the mechanism suggested by Hebb [1949] in which the ability of one neuron to activate another increased when the two neurons were active at the same time. However, unlike Hebb, in the recommendation architecture the tendency decays unless it results in a behaviourally useful activation, and as discussed later, the activation cannot occur unless other active substrates also encourage secondary activations of the type. Fourthly, simultaneous condition recording on two devices creates a tendency for one device to activate the other at a later point in time.

Some devices within a column target competition. The set of conditions recorded on these devices will be called the portfolio of the column, and the activation of some of these devices will be referred to as the activation of the column portfolio. Portfolios are a useful intermediate concept for the description of cognitive processes, and will be employed extensively for this purpose in sections 5 and 6.

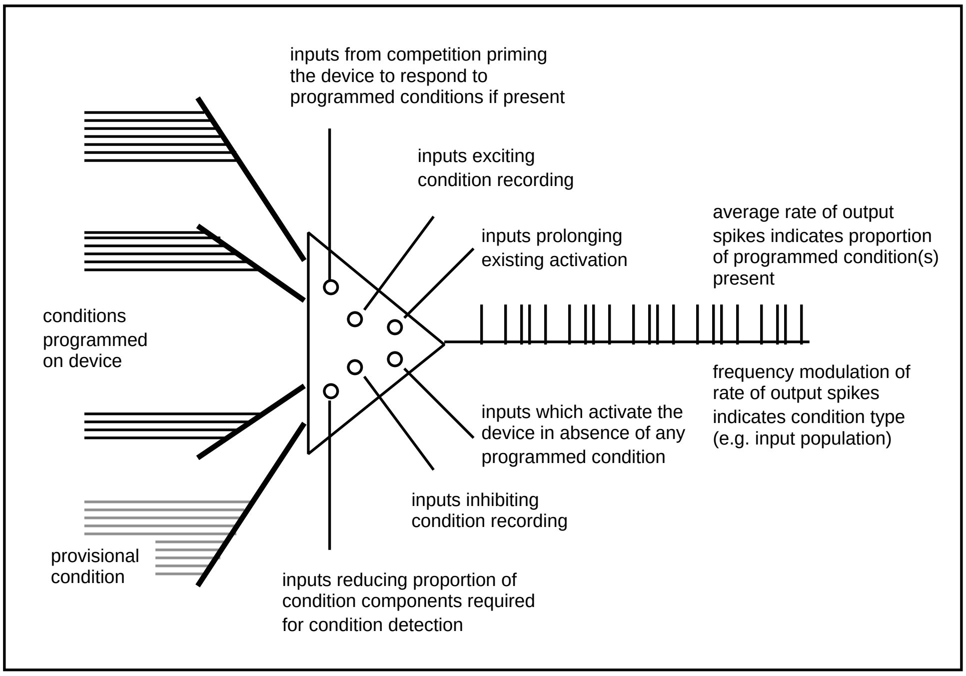

Figure 2. The condition recording device in the recommendation architecture. These devices have different groups of inputs defining different conditions or very similar sets of conditions. The presence of one or more recorded conditions activates the device. Other inputs determine whether or not a device will record an additional condition at any point in time, but these inputs do not form parts of conditions. Yet other inputs prolong the activity of an active device or activate it in the absence of any of its recorded conditions, or activate it if substantial proportions of conditions are present but not full conditions. An activated device generates a series of voltage spikes. The average spike rate indicates the number of recorded conditions currently present, and the phase of a frequency modulation imposed on the average rate indicates the input space within which the conditions have been detected.

3.2 Condition recording devices

A condition recording device within clustering therefore has the fairly complex structure illustrated in figure 2. Devices have inputs which define their programmed conditions, and also have inputs which determine when additional conditions will be recorded by the device. Devices also have inputs which activate them independent of the presence of any of their recorded conditions. Such inputs are required for activations on the basis of past simultaneous module activity as discussed in section 3.1. There are also inputs which can prolong the activity of the device beyond the time during which the condition is present. These inputs make the indication of the presence of the condition available longer when required for system processes.

Device outputs are sequences of voltage spikes with a structure which provides some additional information to their targets. The average rate at which spikes are generated indicates the number of conditions that are currently present. The frequency at which this spike rate is modulated indicates different input domains within which the conditions were detected [Coward 2000]. For example, sensory inputs derived from a visual domain corresponding with the current attention object could be tagged with a specific frequency modulation, and sensory inputs from outside that domain left unmodulated. As illustrated in figure 3, detection of conditions within that attention domain would be favoured and such conditions would also be tagged with the same modulation frequency.

Note that the modulation is not just a signal that neuron activations are derived from the same source as originally proposed for the function of synchronization across a neuron population by von der Malsburg [1981]. Rather, it is the mechanism which ensures that only information derived from one sensory object is used for condition detection in a neuron population.

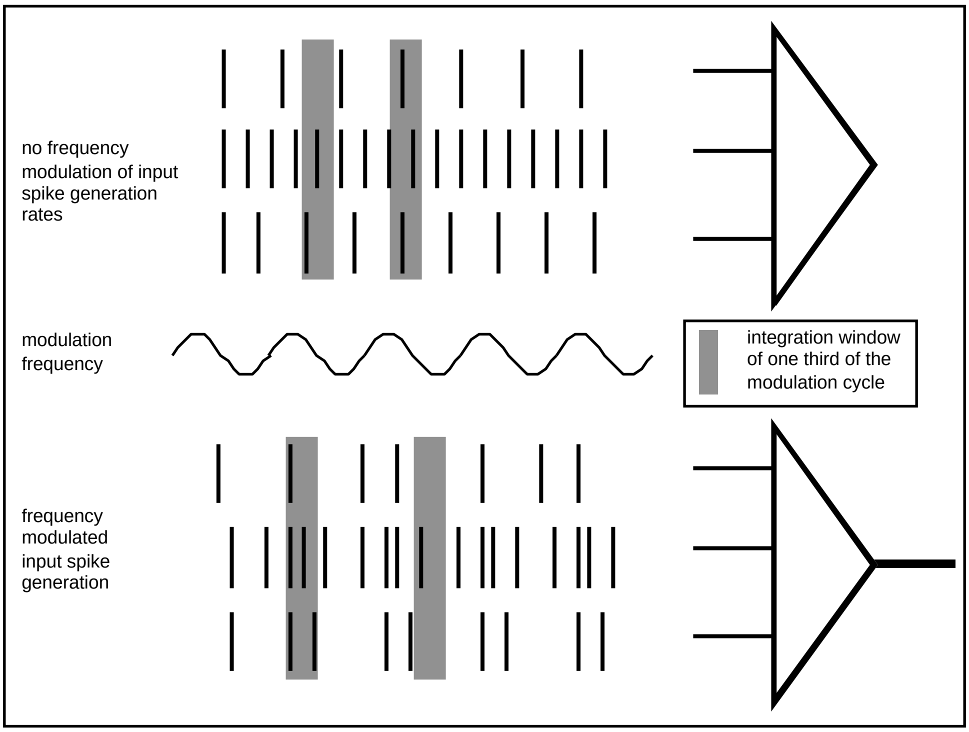

Figure 3 Effect of modulation on the size of the integrated signal in a neuron. The recipient neuron integrates its inputs over the (synaptic) integration window illustrated. The number of spikes within one integration window varies from two to four without modulation, but from one to five with modulation. Thus if the threshold for a neuron receiving the three inputs was at five peaks, only the modulated combination would generate any output.

If inputs to the same physical group of neurons from different domains are tagged with the same modulation frequency but different phases, separate active populations of neurons can be established with limited interaction between them. Separate processing of a number of different visual objects perceived at different times can occur, with conditions detected within all the objects retained active and independent until released simultaneously to devices detecting conditions within groups of objects. If conditions activated on the basis of past simultaneous module activity are tagged with a different frequency modulation phase from conditions within current sensory objects, secondary activations (corresponding with mental images) can be separated from direct sensory objects.

3.3 Columns

The function of a column is to manage the recording and detection of a portfolio of conditions and provide outputs to a range of external locations (other clustering modules and competition components) indicating the detection of the portfolio. Management of recording must ensure that the portfolio retains an adequate degree of internal similarity and as little as possible duplication of conditions with other column portfolios in the same array.

Different layers within a column perform different management functions, and these functions must all be managed by detection of conditions at some level of complexity. If the optimum level of complexity for performing two functions is similar, the functions could be performed by one layer. As the difference between the optimum levels of complexity for two functions increases, the more the functional advantages of two layers will tend to outweigh the resource costs of the additional layer. Hence the optimum number of layers is a compromise between the resource cost of and operational discrimination provided by additional layers.

Change management connectivity in the three layer column used in an electronic implementation of the recommendation architecture is illustrated in figure 4. With some additional connectivity to support prolonged and indirect activations, this column was used to demonstrate bootstrapping of memory, and activation of mental images on the basis of simultaneous past activity of portfolios [Coward 2001].

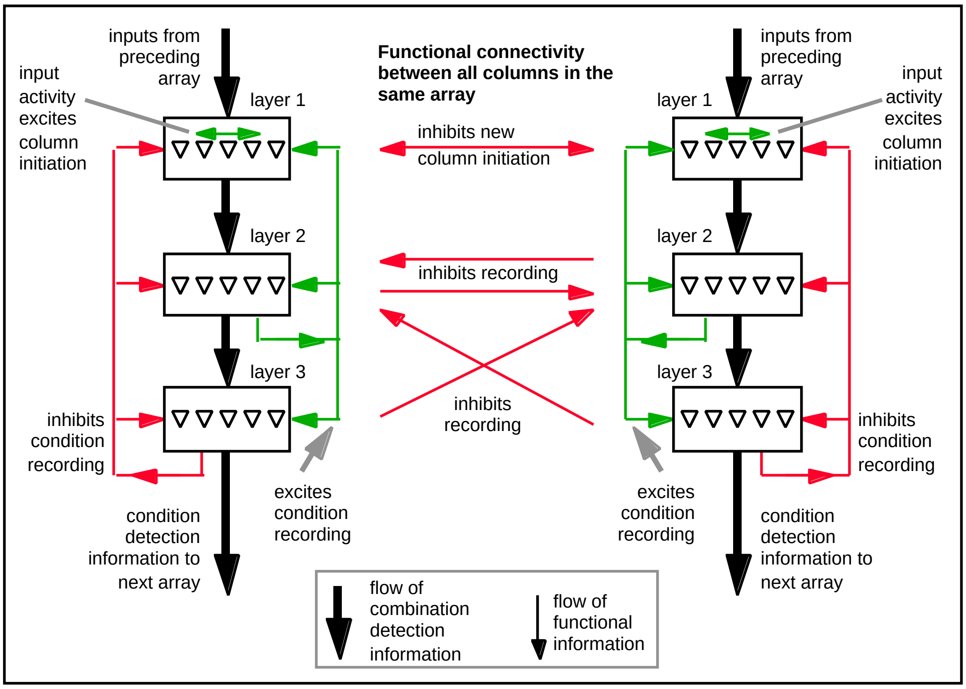

Figure 4 Condition defining and condition change management connectivity between and within two columns in the same array. The simple columns illustrated have three consecutive layers of condition recording devices. Condition defining connectivity enters layer 1 from a previous array, condition defining connectivity into layer 2 comes from layer 1 in the same column and so on. Conditions in one layer are therefore combinations of conditions in the preceding layer. Condition defining outputs go from layer 3 to columns in arrays detecting conditions in higher ranges of complexity. Non condition defining connectivity (functional connectivity) enters and leaves specific layers as shown. Much of this connectivity indicates the general level of activity in the source layer of the column, and may derive from special purpose devices which receive inputs from many condition recording devices in the same layer and column, and generate an output that averages those inputs. Inputs supporting extended activation and indirect activation of the column enter or leave specific layers but are not illustrated.

Layer 1 receives inputs from outside the column which indicate the presence or absence of relatively simple conditions detected in an earlier array. Conditions which are combinations of these inputs are defined and detected in layer 1 and form the inputs to layer 2. Layer 2 defines and detects conditions which are combinations of its inputs and these conditions are the inputs to layer 3. Combination conditions defined and detected in layer 3 are the outputs from the column indicating the presence of the column portfolio. The identity of the specific devices providing the output can provide some information on the type of conditions present within the portfolio.

An array begins with randomly defined provisional conditions programmed on devices in every layer of every column. The provisional conditions on layer 1 devices are random selections from the input space to the column, but biases are placed on the random selection process for these conditions to ensure that the provisional conditions in one column are relatively similar to each other and fairly different from those in other columns. The very early experience of the system does not record conditions in any column, but is used to identify different groups of array inputs which tend to be active at the same time. The random definition of provisional conditions in the first layer of different columns is biased in favour of inputs from different such groups.

To understand the development of an array, suppose that the array of which two columns are illustrated in figure 4 is exposed to sensory inputs derived from a series of different instances of a number of different categories of visual object. The system never experiences exactly the same object more than once. For clarity of explanation, suppose that the categories are different types of fruit: apples, pears, plums, peaches etc. Suppose also that the minimum level of condition detection which the system must achieve in response to every input state, as discussed in section 3.1, is outputs from two columns.

Initially the array is exposed, with no condition recording, to a series of sensory inputs derived from different instances of different types of fruit. Any one input will tend to be active more often when some inputs are also active, and less often when other inputs are active. Groups of inputs which tend to be active at the same time are defined, and provisional conditions for layer 1 in different columns are biased in favour of different groups. Condition recording in response to instances then begins.

Suppose the first instance is a red apple. At first no column produces an output, because no column has any regular conditions. As illustrated in figure 4, input activity in layer 1 excites initiation of columns. The two columns with the highest level of input activity will be triggered by that activity to record conditions on every level until a column output is present. Initiation of other columns will be inhibited by that activity. The initial portfolios of the two columns will correspond with two sets of similar conditions which were all present in the apple instance. One set might correspond with red, but such an exact correspondence with a cognitive condition is very unlikely given the random definition of the provisional conditions. More plausibly, one set might correspond with some mix of redness and presence of stalk, the other might correspond with smoothness and "spherical-ness". Even more plausibly, the portfolios might be sets of conditions which correlated even less precisely with well defined cognitive features. However, each portfolio corresponds with some set of similar conditions in the input space derived from the specific apple instance. Future apple instances would be somewhat more likely to contain conditions within these portfolios, future instances of other types of fruit would be somewhat less likely to contain such conditions.

When the next fruit instance is presented, there are a number of different possibilities. If this instance is almost identical with the first, there will be activity in layer 3 of both columns, and the condition recording inhibition generated by those layers will prevent further recording. If the next instance is different but sufficiently similar to the first apple, there will be activity in layer 2 of both of the columns but not layer 3, and condition recording will occur in all layers until there is activity in layer 3. This additional recording slightly expands the definition of condition similarity for the column portfolios. If the next instance is somewhat less similar, one or other columns will not have activity in layer 2, and a third column will be initiated with conditions somewhat different from those in the first two columns. A very dissimilar instance will result in initiation of two new columns.

Experience of more and more instances of different types of fruit gradually expands the number of portfolios and the similarity definition for each portfolio until there are enough portfolios to generate output (with or without condition recording) in response to every fruit instance. However, at any time a novel type of fruit could initiate new portfolios.

Any one portfolio will be activated (i.e. its column generate an output) in response to a number of different kinds of fruit, but not all types of fruit. Members of a small subset of the portfolios will each tend to be often (but not always) active in response to fruits of one type, and a different but sometimes partially overlapping subset in response to fruits of a different type.

The array definition process thus processes a large number of input portfolios which discriminate weakly between different categories into a (generally smaller) number of portfolios which discriminate more strongly between the categories. This increase in discrimination has been demonstrated by electronic simulation [Coward 2001]. The way in which competition can learn to associate such portfolios with appropriate behaviours using only consequence feedback and a genetically defined tendency towards imitation is described in section 5.1.

Note that a practical array of portfolios will generate responses to all possible categories of visual object, not just types of fruit. However, genetic information could result in partial physical separation between portfolios responding to different broad types of category. For example, if a physically separate array of portfolios was genetically biased in favour of the types of condition which tended to occur in human faces, the result could be faster learning of faces and partial separation between the columns responding to faces and the columns responding to other visual objects. The separation would only be partial because the biased random process for condition definition means that some portfolios in the "face" array could respond to other objects and vice versa.

3.4 Detailed Structures within Competition

Competition is made up of devices which total the excitatory and inhibitory weights of currently active inputs from a range of sources, and produce an output if the total exceeds a threshold. The devices adjust their input weights in response to consequence feedback. Unlike the device algorithms used in clustering, these algorithms are generally similar to the perceptron type algorithms used in artificial neural networks [see e.g. Ballard, 1997].

The competition system is made up of components corresponding on a one-for-one basis with all possible system behaviours. Each component is a device or group of devices. Some components correspond with behaviours which are “atomic” in the sense that the system can implement the behaviour or not implement it, can vary the speed and perhaps the degree with which the behaviour is performed, but cannot change the nature of the behaviour. An atomic behaviour could be the contraction of an individual muscle, or genetically programmed groups and sequences of such contractions. Other components correspond with higher level behaviours such as groups and sequences of atomic behaviours, and yet higher behaviours which are groups and sequences of such groups. At the highest cognitive levels, behaviour is achieved by outputs from clustering driving a sequence of competition components which in turn activate more specific competition components. Any very frequently occurring sequence or set of behaviours will tend to result in one component in competition which receives the inputs from clustering and drives behaviour directly into atomic behaviours.

The inputs to a component are the outputs of specific devices within a clustering module or another component. Because of the use of consequence feedback within competition, such inputs cannot have operationally complex meanings. Only two meanings are possible: a recommendation to perform the behaviour corresponding with the component; or a recommendation against performing that behaviour. The output from a component can only be interpreted as a recommendation not to perform any other behaviour (if it is directed to a component associated with another behaviour), as a recommendation to perform detailed behaviours within the general type recommended (if directed to a more detailed component), or as a command to perform the corresponding behaviour (if it exits competition).

3.5 Management Processes

To improve the probable operational value of permanent conditions, provisional inputs are derived from devices in modules which have often been active in the past at the same time as the module in which the device is located. To avoid the need to record very large volumes of information about the relative timing of past module activity, a periodic input driven rerun of a selection of past experience is performed. This process only requires storage of past relative occurrence frequency of input information. The frequency of past simultaneous module activity can be derived directly from this rerun and used to guide provisional condition definition.

The number of devices which will be required by a column and the number of columns which will be required by an array cannot be known a priori. There must therefore be a process for assigning resources as required. Such a process needs knowledge of which resources are available and which have been already assigned. A simple way to achieve this is to maintain a map of available resources, with an active boundary separating assigned from unassigned resources, and currently required resources being assigned from those corresponding with the current location of the boundary. Such a map is also an implicit map of the time sequence of experience, and damage to the map will prevent any future recording of conditions.

4 Physiological Resemblances

As discussed in detail in Coward [1990; 2001] there are strong resemblances between the structure of the brain and module separations and processes in the recommendation architecture. The organization of the cortex into layers, columns and areas resembles the modular hierarchy in clustering. The thalamus, basal ganglia and thalamus have functions which resemble the recommendation architecture competition subsystem [Coward 2001]. The REM sleep process resembles the required process for definition of provisional conditions and the effects of damage to the hippocampus system resemble the effects of damage to the condition recording resource management system [Coward 1990; 2000].

5 The recommendation architecture cognitive model

In the recommendation architecture, information can be recorded by two qualitatively different mechanisms. One is the permanent recording of conditions. The other is adjustment to recommendation strengths of recently active conditions into recently

performed behaviours in competition. Information can be accessed by four different mechanisms. Firstly, the actual presence of a condition within current sensory inputs activates the device on which the condition is recorded. If enough devices are activated within a column, the column portfolio is activated (i.e. column output devices are activated). Secondly, an activated portfolio can recommend activation of other portfolios which have often been active at the same time in the past, and the recommended portfolio will activate if adequate recommendation strength is present. Thirdly, an activated portfolio can recommend activation of other portfolios which recorded conditions at the same time in the past. A variant of this mechanism is activation of portfolios which recorded conditions somewhat before or somewhat after an episode of condition recording in the active portfolio. Indirectly activated portfolios can in turn recommend activation of yet other portfolios. The fourth mechanism is comparison of recommendation weights. The weights of all active behaviours into each recommended behaviour are totaled, and the behaviours with the strongest weights are implemented.
The simplest arrangement of clustering areas able to support complex cognitive processing is illustrated in figure 5. For the purposes of this description, an area module will be regarded as an array detecting portfolios on one level of complexity which are communicated to competition. The consistency management function of an area will be omitted. In figure 5, outputs from area 2a to competition indicate the detection of portfolios with a complexity comparable with visual features. Outputs from area 3a indicate portfolios comparable with visual objects. Area 4 detects conditions which combine portfolio outputs from area 3a generated by a sequence of perceptual objects. Individual area 4 portfolios will therefore correlate partially with groups of perceptual objects. Area 5 will detect conditions which combine portfolio outputs from area 4 generated by a sequence of such groups. Individual area 5 portfolios will therefore correlate with groups of groups of perceptual objects.

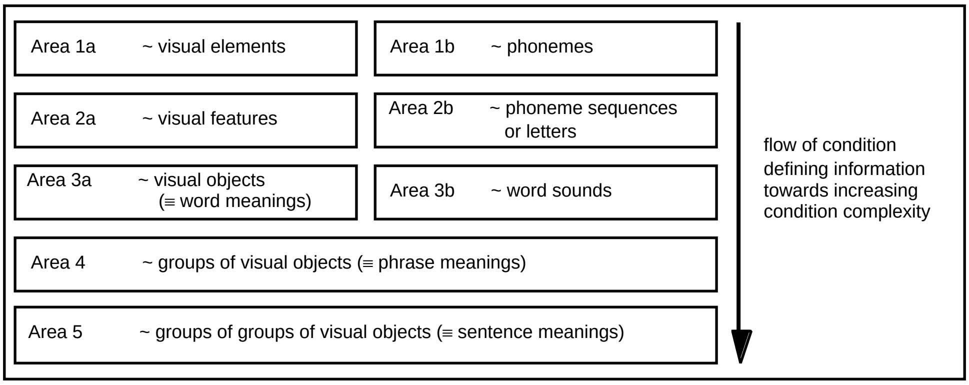

Figure 5. An architecture to support higher cognitive processes. The model has five major clustering levels Different clustering levels detect portfolios of conditions within different ranges of complexity, and output the detection of conditions with the same range of complexity as would be possessed by features, objects, groups of objects and groups of groups of objects respectively, but outputs do not correlate unambiguously with such cognitive conditions. The division of levels 1 through 3 into two areas reflects different input spaces within which conditions are detected

To give an example, consider the arithmetic processing of the visual object "3 + 5 = 12 - 6". As attention moves across the equation, area 3a detects portfolios within each of the sequence of visual objects 3, +, 5, =, 12, -, and 6. Area 4 detects portfolios within each of the groups 3 + 5, =, and 12 - 6. Such portfolios contain conditions that include simpler conditions detected within each separate member of the group. These portfolios would also correlate partially with 8, = and 6. Area 5 detects portfolios which contain conditions that include simpler conditions detected within each separate group. Individual portfolios in area 5 will therefore correlate partially with 8 = 6, and the group of portfolios would have recommendation strength in favour, for example, of saying "that is wrong". For more detailed discussion of this example, see Coward [2001].

The competition components which receive outputs from one area are illustrated in figure 6. Ten types of competition component corresponding with ten types of behaviour which could be recommended by a clustering area are as follows: prolong the activity of some currently active portfolios; activate portfolios active at the same time in the past as the currently active portfolio; activate portfolios containing conditions recorded at the same time in the past as some conditions in the currently active portfolio; activate portfolios containing conditions recorded just before or just after some conditions in the currently active portfolio; synchronize the activity of several different groups of currently active portfolios; perform a general sequence of attention behaviours; perform a specific sequences of attention behaviours; perform an individual attention behaviour; speak a word; and say a phrase.

Many of these types gate the release of a subset of clustering level outputs to the appropriate targets in the same or another clustering area. Some competition components determining the selection of general types of behaviour may also target competition components determining the selection of specific behaviours within their general type. Other competition components drive behaviours directly.

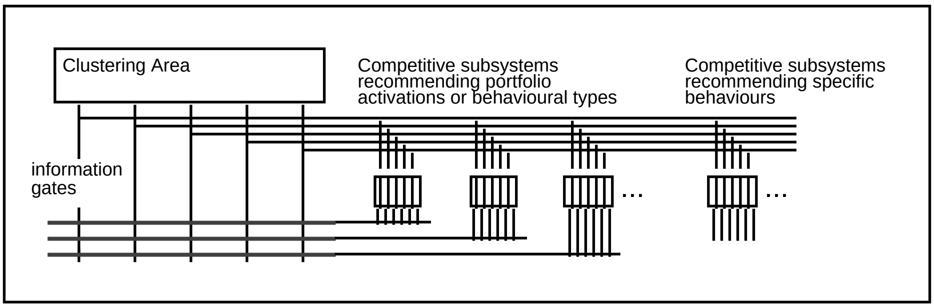

Figure 6 Competitive components receiving outputs from one clustering area module. Different behavioural interpretations are placed upon the same clustering outputs by different components. There is competitive inhibition between and within competitive components to limit selected behaviours to a small, consistent set. In some cases the behaviour accepted by a competitive subsystem is release of the outputs from clustering which correspond with the behaviour to either the next clustering level or to a more detailed competition subsystem. This release behaviour is indicated by the information gates. In other cases competition outputs drive their corresponding individual behaviours, either external (e.g. eye movements) or internal (e.g. prolonging the activity of clustering neurons in specific modules).

All outputs from a clustering area are potentially available to all competition components. One competition component corresponds with behaviours of one type, and smaller subcomponents within a component correspond with different behaviours of the same type. Inputs to competition from clustering are excitatory, and a component or subcomponent interprets any such inputs as a recommendation in favour of its corresponding behaviour type or behaviour. Inhibitive connectivity between and within competition components result in selection of one behaviour, or perhaps a small set of consistent behaviours, in response to currently active portfolios. Behaviours can be directed externally or can be further steps, such as indirect portfolio activations, towards determining external behaviour.

If a group of active portfolios recommended a sequence of behaviours of some kind, prolonging the activity of those portfolios will be part of ensuring the completion of the sequence. However, the output from the competition subsystem only indicates acceptance of this behaviour type, the identity of the clustering targets to remain active depends upon the identities of the specific combination of inputs from clustering which won the competition.

Synchronization of the frequency modulation phase of the outputs of different populations of portfolios in one level into another level is part of attention management. For example, as discussed earlier in this section, attention might have been focused sequentially on three different objects, and portfolio populations activated in level 3a of figure 5 in response to each object. To maintain separation between the populations in level 3a, device outputs detecting conditions within the different objects would be frequency modulated with a different phase as discussed in section 2.2. The combined portfolio outputs could then be interpreted as a recommendation to release all of those outputs to level 4, but synchronizing the phases of those outputs. The synchronization would mean that conditions containing information from all three objects would be detected in level 4, and the activated portfolio population would correspond with the group of objects and recommend behaviours in response to the presence of the group.

Competition receives outputs from portfolios currently being detected by clustering. If a portfolio has been present in the past at the same time as the performance of a number of different behaviours, it will have acquired recommendation weights in favour of or against those behaviours in the component corresponding with the behavior, depending on the consequence feedback from those behaviours. Competition adds the weights of all currently recommended behaviours and selects the behaviour with the largest weight. New portfolios are given an initial weight similar to the weights of the most similar previously existing portfolios, or genetically defined initial weights. If portfolios are new and no significant recommendation strength has been assigned in these ways, a behaviour can be selected randomly. Such random selection can be limited to behaviours within a behaviour type which has already been selected. Alternatively, a behaviour can be selected by imitation of an externally observed behaviour.

5.1 Bootstrapping of memory and behaviour

Behaviours can be defined heuristically with limited a priori guidance. Such definition will be illustrated by describing a possible process for acquisition of simple speech, with careful attention to the nature of the a priori (genetic) guidance needed. Learning goes through a series of steps generally consistent with observation of how humans learn to speak and understand words, but the purpose of this section is not to offer a formal model for speech acquisition but rather to demonstrate that a cognitive capability like speech can be acquired in the recommendation architecture model with only limited and plausible genetic guidance.

Genetic information specifies creation of a set of detailed competition components which drive muscle movements contributing to sound generation. Every possible such movement has a corresponding genetically specified competition component, and activation of the component results in the movement. Genetic information also specifies the existence of intermediate competition components which activate randomly selected sequences of detailed components and therefore generate sounds. Learning proceeds in a series of partially overlapping steps.

The first step is creation of an array of portfolios in clustering areas 1b and 2b of figure 5 in response to hearing sounds, at different levels of condition complexity. Because speech is somewhat different from other sounds, there will be a tendency for speech related portfolios to be somewhat separate from the portfolios created in response to other sounds. This tendency could be reinforced by genetically determined initial connectivity biases within clustering.

The next step is generation of sounds using intermediate competition components. Initially a component is randomly selected after any activation of a portfolio population in the array of sound related portfolios in levels 1b and 2b. A positive consequence feedback is genetically programmed to be generated if the portfolios activated in response to hearing an external sound (i.e. not self generated) are similar to the portfolios activated shortly afterwards in response to hearing a self generated sound. One effect of this consequence feedback is that the sequence of detailed components activated by the intermediate component is fixed long term. In other words, the presence of a sound in the environment results in production of that sound becoming instantiated in an intermediate component. If there is no feedback within some period of time, the intermediate component is reconfigured with a different randomly selected sequence of sound generating muscle movements, or deleted. The second effect of the consequence feedback is that the activated portfolios acquire recommendation strength in favour of activating the intermediate component. In other words, the behaviour of imitating sounds which are heard is acquired.

The next step in learning is that portfolios are created in area 3b of figure 5 in response to sequences of sequences of sounds which are heard. These portfolios will correlate partially (or ambiguously) with frequently heard sequences, and therefore with words which are heard. Higher level competition components are defined which activate randomly selected sequences of intermediate components. If a self generated sound sequence activates a portfolio population in area 3b similar to an immediately prior population activated by an external sound sequence, a genetically defined consequence feedback results in the higher level component being fixed and the active portfolios acquiring recommendation strength in favour of activating it. Thus the behaviour of imitating words which are heard is acquired.

The next step utilizes a genetically programmed tendency for the portfolios created in level 3b in response to sequences of sounds to have recommendation strength in favour of activation of portfolios created in level 3a in response to visual experiences, if the visual experience portfolios are often active at the same time as the sound sequence portfolios. The effect is that hearing the word will tend to activate a partial visual image of the type of object often seen when the word was present in the past. The portfolios making up a visual image will also recommend any other behaviours which have become associated with the object. In addition, the visual experience portfolios acquire recommendation strength in favour of activating the higher level component which tends to be activated by the sound sequence portfolios. The effect is that seeing the object will tend to result in speaking the corresponding word. Consequence feedback associated with the perceived behavior of adults in response to activating a higher level component (i.e. speaking a word) will affect the recommendation strengths of active portfolios in favour of the word just spoken.

Thus a set of genetically defined tendencies result in relatively efficient acquisition of simple speech behaviours. Learning does not require a priori internal definition of cognitive categories. Genetic information provided three types of information. Firstly, it indicates the available range of detailed muscle movements. Secondly, it biases initial connectivity in favour of the types of sensory inputs and portfolio condition complexity ranges which will most effectively drive those behaviours. Thirdly, it defines in general terms the circumstances in which consequence feedback will be generated and the effects of such feedback.

6 Accounting for Cognitive Phenomena

In this section the way in which memory results from the recommendation architecture mechanisms for information recording and access is described. Semantic and episodic memory and the relationship between attention and memory will be briefly discussed, and working memory will be discussed at more length.

6.1 Semantic Memory

In a typical semantic memory experiment, the speed with which a subject can confirm or contradict a proposed categorization of an object is measured. Examples might be pictures of a robin, an ostrich and then a cat with the word "bird". The speed of response to correct categorizations is higher if the instance is a more typical category member (like robin) than if it is less typical (like ostrich). However, speed of response to obvious non-members (like cat) is also high [Rips, Shoben & Smith 1973].

In the recommendation architecture model, each visual instance of a category directly activates a population of portfolios in level 3b of figure 5. Because of the similarity between different members of the same category, there are some portfolios which tend to be activated fairly often in response to different members. When instance name auditory portfolios (e.g. in response to hearing the word "robin") and visual portfolios (in response to seeing a robin) are active at the same time, the auditory portfolios acquire recommendation strength in favour of activating the visual portfolios activated in response to the corresponding bird.

The auditory portfolios activated in response to the word "bird" are active at the same time as the visual portfolios activated in response to many different types of bird (when the words "that is a bird" are heard). Hence these auditory portfolios will tend to indirectly activate the visual portfolios which are activated most frequently in response to different types of bird.

When a subject is show a picture of a robin and asked if it is a bird, visual portfolios are activated directly by the picture and indirectly by the word. Substantial overlap between the two populations exists because the robin is a typical bird, and many of the portfolios activated in response to a robin are also activated in response to many other birds. This overlap is interpreted as a recommendation to say "yes". For a cat, the overlap is minimal and the response is "no". For an atypical bird like an ostrich, the overlap will be moderate and inadequate to generate a clear recommendation. However, some of the portfolios activated in response to the ostrich will often have been active in the past at the same time in the response to other birds, and will therefore have acquired recommendation strength in favour of indirect activation of some of the other portfolios in such activations. Such indirect activations expand the portfolio population activated in response to the ostrich. Portfolios often active at the same time in the past as those activated in response to "bird" may be activated, expanding the "bird" activated population. The overlap between the two populations is then sufficient to generate the "yes" response. However, the need for an additional indirect activation step means that response is slower.

6.2 Episodic Memory

A typical episodic memory experiment is targeted recall in which subjects are asked to recall particular past events [Neisser & Libby 2000]. The starting point for targeted recall in the recommendation architecture model is hearing words which describe an event. Portfolios are activated in area 3b of figure 5 which contain conditions within the sounds of the words. Secondary populations of portfolios containing conditions which occurred within visual and other sensory inputs are activated in area 3b on the basis that the secondary portfolios have often been active in the past at the same time as the primary "auditory" portfolios. A significant proportion of the portfolios in these secondary populations were also active during the event, and a somewhat smaller proportion recorded conditions during that event. Because of the words used, the proportions are larger for the target event than for any other event.

All active portfolios have recommendation strengths in favour of activating other portfolios which recorded conditions at similar times in the past. Active populations derived from the presence of words like "recall" have recommendation strength in favour of accepting these types of recommendations. Because the target event has the highest proportion of activated portfolios in the active population, acceptance of such recommendations will tend to result in a tertiary active population with an even higher proportion of portfolios which recorded conditions during the target event. This process is self reinforcing, especially if a large number of conditions were recorded during the target event. The resultant population will contain a high proportion of the portfolios which were active during the original experience of the event, and will be experienced as a re-perception of the original event, although in general the portfolios closest to input from the senses (i.e. areas 1a and 2a in figure 5) are not reactivated.

Activated portfolios in this tertiary population have recommendation strength in favour of generating verbal descriptions of the event. Use of recommendation strength in favour of activating portfolios which recorded conditions somewhat before or somewhat after condition recording in currently active portfolios allows the re-perception to be set at the beginning of the event and moved through the event.

6.3 Attention, Awareness and Memory

Experiments have been performed which indicate that in some cases there is no memory for an experience created in the absence of attention, even though some awareness of the experience at the time can be demonstrated. An example is the dichotic listening experiments of Triesman [1960], which tested the ability to remember when inconsistent information was provided to left and right ears. The subjects were read two sections of different text, one to each ear, and were asked to repeat aloud (to shadow) the text heard by one specific ear. Part way through, the two texts were switched between ears. The subjects responded by (erroneously) switching to the text from the other ear (i.e. the meaningful continuation). Afterwards, some content could be recalled from the shadowed text but not from the other text. Thus although the unshadowed text did not result in later recall, the ability to immediately switch implies that information from both ears was being processed in some fashion.

In the recommendation architecture model, there are physically separate clustering areas 1b, 2b, and 3b in figure 5 for each ear, because the sources of information are physically distinct. Words heard by different ears therefore activate separate populations of portfolios. Both of these populations have recommendation strengths in favour of activating visual portfolios in area 3b which have often been active in the past at the same time. However, portfolios already active in areas 3a, 4a, and 5a also have such recommendation strengths, with an operational effect of favouring indirect activation of portfolios with syntactical and/or semantic connection with the currently active portfolios.

In the dichotic listening experiment, the original instruction to pay attention to one ear resulted in secondary activations recommended by portfolios activated by words to that ear being favoured. However, once the listening and shadowing were under way, recommendations were also present based on continuity of meaning. When the text shift between ears occurred, continuity of meaning recommendations generated the attention shift. However, later recall depends upon portfolio activity in levels 3b, 4b and 5b and associated condition recording. Because only the attended text generated activity in those levels, recall of the other text was not possible.

6.4 Working Memory

A range of experimental procedures are viewed as measurements of working memory. These procedures include list recall, short term forgetting and pattern matrix testing.

In list recall experiments, subjects are presented with a sequence of objects and asked to immediately report the objects on the list, in any order. The longest list length for which all objects can consistently be reported correctly is measured. Typical objects are numerical digits, letters, or words, which are presented verbally or visually. Normal subjects can fully recall lists of about seven (plus or minus two) random digits, but only four or five random words or letters [McCarthy & Warrington 1990]. In a variation of this test, subjects are asked to report as many items as possible on longer lists. Normal subjects show enhanced recall for the first few items on the list (the primacy effect) and enhanced recall for the last few items (the recency effect) [Baddeley 2000]. A brief delay occupied with another task eliminates the recency effect but has much less influence on the primacy effect [Glanzer 1972].

List recall capability is considerably increased if sequential groups of objects within a list have some semantic connection. Such organization is called "chunking". For example, although list recall for random words is limited to four or five, recall of meaningful sentences can extend to sixteen words or more [Baddeley, Vallar, & Wilson 1987].

In the short term forgetting experiments of Petersen & Petersen [1959], subjects were presented with three consonants and required to retain them over a delay of up to 18 seconds while counting backwards in threes. Performance indicates a short term memory component which declines over about five seconds [see also Baddeley & Scott 1971].

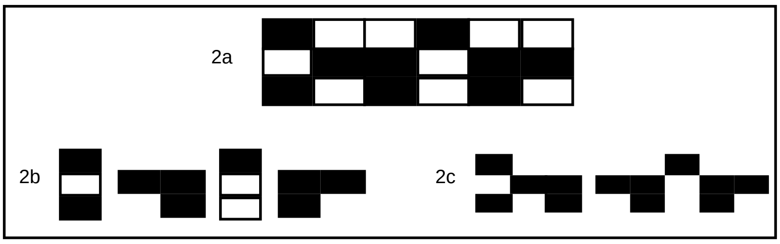

Figure 7. Pattern matrix test example. a. An example of a pattern matrix test with nine filled rectangles. b. Examples of visually objects perceived in level 4 of the recommendation architecture model. c. Examples of groups of objects perceived in level 5 of the recommendation architecture model.

Pattern matrix testing [Della Salla, Gray, Baddeley, Allamano, & Wilson 1999] investigates visual working memory, which is believed to be supported by a separate mechanism from working memory for objects with symbolic content, such as letters, words and numbers [Baddeley 2000]. In this test, subjects are presented for a few seconds with a matrix of rectangles in which half of the rectangles are coloured black, and are immediately asked to indicate which rectangles were filled on an identical but unfilled matrix. Matrices contain from four to thirty rectangles, filled rectangles therefore range from two to fifteen. An example of a matrix with nine filled rectangles is illustrated in figure 7a. The number of filled blocks on the most complex matrix accurately recalled is about nine for normal subjects.

6.4.1 General recommendation architecture mechanisms

In the recommendation architecture model, the function of the mechanisms which give rise to many of the phenomena labeled working memory is to ensure that portfolio populations activated in response to different information sources can be kept separate, can be combined as required to directly activate higher level populations, and can be kept active longer or reactivated when required.

For example, a number of different objects may be present in the visual environment, and the most appropriate behaviour may be determined by the presence of several (but not all) of these objects. As attention focuses on the different objects in turn, portfolio populations activated in response to one object must be kept active but separate from those activated in response to other objects until portfolios have been activated in response all relevant objects. Portfolios containing conditions made up of information derived from all the objects can then be activated. Several portfolio populations may be independently active within the same physical array.

As time passes, the probability of an active portfolio being still relevant to behaviour decreases. Furthermore, there are limits to the number of independent active portfolio populations which can be distinguished within the same physical group. Hence active populations will need to decay with some time constant. However, it may sometimes be necessary for behavioural reasons to extend the activation of a population or to (perhaps partially) reactivate a recently active population. Such extensions and reactivations are behaviours which must have an adequate recommendation strength across currently active populations.

For example, an existing active population will have recommendation strength in favour of prolonging its own activity. The set of portfolios in one region which could be activated in response to an object is large. A subset of this larger set will be actually activated in response to a perception of the object. The identity of this subset may change throughout the period in which the activated population exists. Self activation recommendation strength is thus recommendation strength to activate other portfolios active in the past at the same time, prolonging the activity of the population although not necessarily the activity of specific individual portfolios. Such a recommendation strength will in general not be adequate to keep a population active indefinitely without the support of recommendation strength from other active populations. Self activation strength will therefore determine the time constant for decay of the population in the absence of supporting recommendation strength. Supplementary recommendation strength from other populations might for example be in favour of prolonging the activity of any population in the level in which the population to be prolonged is located, and could extend activity beyond normal decay times.

When several active populations on one level generate a combination population at a higher level, the higher level population is active at a similar time as its constituent populations, and therefore acquires recommendation strength in favour of reactivating these populations based on recent simultaneous activity. Again, supplementary recommendation strengths will in general be required for such reactivations to become accepted behaviours.

Limits to working memory are therefore imposed by the decay time of active populations, the number of populations which can be active simultaneously in the same physical region, and the availability of recommendation strength in favour of prolonging population activity or of reactivating population activity on the basis of recent simultaneous activity.

6.4.2 List Recall

When a random verbal list of letters is presented, an activated population is generated in area 2b of figure 5 in response to each letter. Combinations of several letters may generate an activation in area 3b, but because the letters do not correspond with a real word this activation will be relatively weak and will not generate a secondary activation in area 3a. The ability to report the list therefore depends primarily on the number of independent populations which can be sustained in area 2b. For random words, strong active populations are activated in area 3b and secondary populations in area 3a. Secondary populations could be activated in level 4, but the absence of semantic relationship between the words means that active populations in level 4 will be weak. List recall therefore again is dependent on the number of independent populations which can be sustained at the same time, in this case in level 3b.

Population independence is based upon different phases of frequency modulation applied to different populations as discussed in section 3.2. The different phases limit the interaction between the populations. To preserve independence, the minimum phase separation between two populations must be of the order of the synaptic decay time for the neuron. The number of possible independent populations will therefore be of the order of the modulation interval divided by the synaptic decay time. Assuming that the observed 40 Hz cortex signal in the EEG gamma band [Llinas, Ribary, Joliot & Wang 1994] reflects frequency modulation and given a synaptic integration window of 8 milliseconds [Konig, Engel & Singer, 1996] it would be possible to support 3 - 4 independent active populations in the same physical group of neurons, each with a different frequency modulation phase, provided that the independent populations did not share more than a very small proportion of conditions. If words have a similar sound, there will tend to be greater overlap in the independent populations in level 3a because of shared portfolios, and accuracy will be reduced as reported by Baddeley [1966].

If there is a significant semantic relationship between sequential words on the list, the combination populations in levels 4 and 5 will be much stronger, and reactivation of populations originally generated in response to individual items becomes possible. For example, if the list is a meaningful sentence, each phrase will activate a combination population in level 4, and these populations will be maintained active simultaneously long enough for an overall combination population to be created in level 5. A "phrase" population in level 4 could reactivate its constituent "word" populations in level 3 on the basis of recent simultaneous activity. These reactivated "word" populations would have recommendation strengths in favour of speaking the individual words. If 3 - 4 "phrase" populations could each regenerate 3 - 4 "word" populations, and in addition the "sentence" activation could reactivate the "phrase" populations, the observed reporting of sentence lengths of sixteen words or more [Baddeley, Vallar & Wilson 1987] could be supported.

Numerical digits represent an intermediate case. Because pairs of digits are themselves meaningful and generally familiar numbers, a list of digits will tend to activate populations on level 3 in response to individual digits and level 4 in response to sequences of digits. The level 4 combination populations can in turn reactivate the individual populations in level 3 which can recommend speaking the digits. As observed [McCarthy & Warrington 1990], lists of digits will therefore be recalled better that random letters or words but not as well as meaningful sentences.

6.4.3 Short term forgetting

In short term forgetting experiments, a small group of objects is presented, such as three letters in the experiments of Petersen & Petersen [1959]. The recommendation architecture mechanism is that active portfolio populations corresponding with the items are constantly refreshed by acceptance of recommendations to perform such refreshing. As discussed under list recall, recommendation strength comes from the populations themselves but required additional strength from other active populations. Such active populations would be at higher levels and recommend extension of any active populations in the lower level. These higher level populations can therefore be interpreted as recommending the type of task, in this case recall of the items. Self refreshing would be experienced as mental rehearsal. A distracting task of a verbal type means that the populations in the relevant higher level area recommend a different task, and item populations will therefore decay with a time constant determined by self refreshment recommendation strength of the item populations alone.

6.4.4 Visuo-spatial working memory

To understand the mechanisms supporting pattern matrix testing, consider the example matrix illustrated in figure 7a. Although the individual rectangles are conceptually objects, it is more plausible that the component objects perceived, or in other words the objects that generate active populations in level 3a of figure 5, are made up of several blocks as illustrated in figure 7b. Groups of such objects as illustrated in figure 7c will result in active populations in level 4, and the activation in level 5 includes information derived from the entire matrix. The recall mechanism is therefore analogous with recall for sentences as discussed in section 6.1.2. The ability to recall block shading is a composite of the ability to maintain several independent active populations corresponding with different figure 7b objects in level 3a, several independent active populations corresponding with different figure 7c groups of objects in level 4, and recommendation strength to reactivate (recently simultaneously active) component populations from populations in levels 4 and 5. Active populations on any level will have some drawing behaviour recommendation strengths. If three to four independent populations can be supported in level 4, and each such population can reactivate a level 3 population corresponding with two to three filled rectangles, the result would be the observed ability to accurately recall matrices with up to nine filled rectangles [Della Salla, Gray, Baddeley, Allamano & Wilson 1999].

6.4.5 Working Memory Deficits

Observations of the cognitive deficits associated with brain damage indicates a wide range of dissociations between working memory and semantic and episodic memory and between different types of working memory. In the recommendation architecture model, there are two types of problem which could be caused by local damage and which would result in working memory deficits.

One type of problem is loss of recommendation strength in favour of continuation of self activity for some populations. Such a problem would result from damage affecting the connectivity between the physical substrate for a group of portfolios and the competitive subsystem managing the behaviour type. The damage would affect only populations containing portfolios located in the damaged region. For example, populations in just one area in figure 5 might be affected. The result of damage would be faster decay of active populations containing affected portfolios.

A second type of problem is loss of recommendation strength of populations at higher levels in favour of reactivation of their source populations at lower levels. Again, such a problem would result from damage affecting the connectivity between the physical substrates for the portfolios on the different levels and the relevant competitive subsystem. For example, the ability of some types of activated populations in level 4 of figure 5 to activate source populations in level 3 on the basis of recent simultaneous activity could be affected.

Unlike semantic or episodic memory, working memory does not depend upon recording of new conditions. Loss of the capability to record new conditions would therefore remove the ability to create new semantic and episodic memories but leave working memory unaffected. This dissociation is consistent with the observations that working memory generally operates normally in amnesic patients [Butters 1984; McCarthy & Warrington 1990; Corkin 2002]. A specific example of this dissociation is the disappearance of the primacy effect but preservation of the recency effect in recall of long lists [Glanzer 1972].

A range of cases in which list recall memory is severely affected have been reported [Caplan & Waters 1990]. In subject KF [Warrington & Shallice 1969; Shallice & Warrington 1970; Shallice & Warrington 1977], verbal recall for single numbers, letters or words presented verbally was good, but deteriorated considerably for two item lists. This combination of deficits would be generated in the recommendation architecture model by local damage affecting self prolongation recommendation strengths limited to area 2b and perhaps 3b. As a result, decay time of populations in area 3b with recommendation strength in favour of speaking different words would be faster than in normal subjects. Secondary populations in area 4 can reactivate populations in area 3b, so the deficit would be less severe if such reactivation capability is still present. However, during list recall experiments the strength of such secondary populations will be lower than when listening to meaningful sentences, because higher level populations are recommending repetition behaviours rather than comprehension behaviours. Consistent with this interpretation of list recall deficits being generated by damage affecting population decay times is the observation that list recall deficits generally occur together with short term forgetting deficits [Warrington, Longue & Pratt 1971; Basso, Spinnler, Vallar & Zanobia 1982; McCarthy & Warrington 1987].

In list recall experiments, lists are typically presented at one item per second. In normal speech, words are presented two or three times faster and higher level populations will encourage generation of strong secondary populations in area 3a corresponding with the meaning of the words. Hence the length of time required for word generated populations in area 3a to remain active in order to generate "phrase" and "sentence" populations in areas 4 and 5 is much shorter than the time such populations must remain active in list recall experiments. The ability to repeat sentences is dependent upon combination populations in areas 4 and 5 reactivating "word" populations in level 3, and will not be affected by damage to the capability of populations in areas 2 and 3 to prolong their own activity. Thus patient IL [Saffran & Marin 1975] could repeat sentences much longer than her reduces digit span. Errors in sentence repetition were close approximations which retained the meaning of the original sentence. The model in this case is that loss of self prolongation strength in areas 2 and 3 resulted in greater drift of such populations but the drift was constrained to similar meaning by reactivation recommendations generated by level 4 "phrase" populations into level 3. Information on the sound of words is lost but meaning is preserved.

6.4.6 Relationship with alternative models for working memory

An influential approach to modeling working memory phenomena and their role within cognition has been that of Baddeley and his coworkers [Baddeley 1986]. The approach was developed on the basis of an earlier model for the flow of information from perception of the environment to generation of behaviour proposed by Atkinson and Shiffrin [1968]. In the Atkinson & Shiffrin model, illustrated in figure 8a, the sensory systems provide information derived from the environment to a short term store (or working memory). Working memory uses this information plus information drawn from long term memory to generate behaviour. New information only reaches long term memory via working memory, by a process called consolidation.

The Baddeley approach models working memory in more detail within the Atkinson and Shiffrin framework. There are three major components to the Baddeley model as illustrated in figure 8b. These components are a phonological loop, a visuo-spatial sketch pad, and a central executive. Information is kept active within the phonological loop by a subvocal rehearsal process. The content of the phonological loop and of the visuo-spatial sketch pad is managed by the central executive.

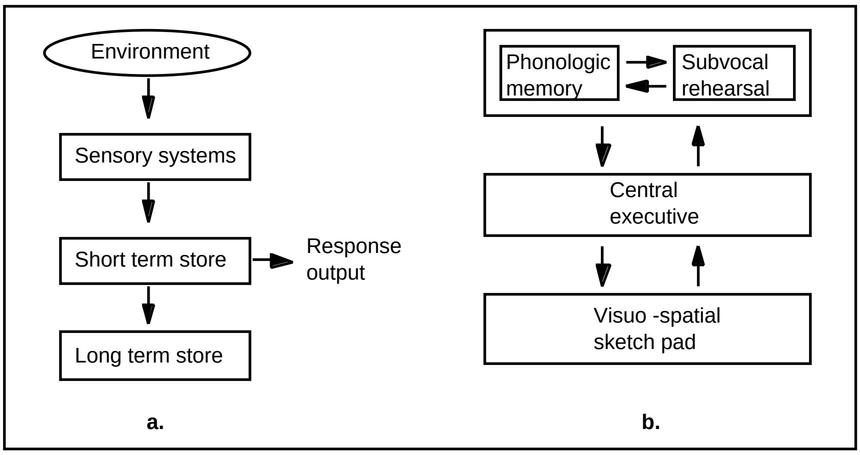

Figure 8. Prior models for the operation of working memory. a. Atkinson & Shiffrin [1968] proposed a model for the information flow from perception to response, and b. Baddeley & Hitch [1974] proposed a model for the internal operations of working memory within the Atkinson & Shiffrin model.

There is a fundamental difference in modeling approach between these models and the recommendation architecture. This fundamental difference has its roots in the way in which modules can be defined in the system architecture of an operationally complex system as discussed in section 1. System architecture modules will not in general correspond with features, and modules will typically not be an easy way to describe the features of the system. A description which focuses on simple descriptions of how a system behaves will therefore not in general be a good explanation of how that behaviour is generated by the system information handling resources. This is the difference between the user manual and the system architecture for an electronic system as discussed in section 1.

Although the Baddeley model provides a concise description for many of the phenomena of working memory, there is a major issue with the detailed definition of the central executive. As Baddeley [2000] has pointed out:

> "One danger with the concept of a central executive is that of postulating a homunculus that is simply assumed to have whatever capacities are necessary to account for the data. One response to this charge is to argue for the value of homunculi as a means of allowing the investigator to set aside some of the more intractable problems. The danger occurs when the theorist treats the homunculus as a solution rather than as a problem to be solved. The question of how to analyze the central executive remains a difficult one."

The more fundamental problem, of which the difficulty of analyzing the central executive is a symptom, is that the concept of "central executive" does not collect together a group of system operations which are supported by one module in the system, and that regardless of the descriptive value of the model, system operations supporting the phenomena of working memory cannot be separated into modules corresponding with the components of the model.

In the recommendation architecture, there is no requirement for a homunculus like function. Prolongation or reactivation of the activity of a population corresponding with one information set is determined by the current activity of all other populations. There is no physical separation between working memory storage resources and long term memory storage resources: the content of working memory is simply the currently activated information in long term memory. The different capabilities exhibited in working memory are the different capabilities by which one active population can prolong the activity of or reactivate another population. There is no consolidation process: any active information which is novel is made part of the long term store by its activation process. The limits to working memory are not defined by resource limitations to a separate working memory component but by limits to the number of different activations which can be active simultaneously in long term memory storage and by the decay time of such activations. Access to information recorded in long term memory once the portfolio population within which the information was recorded has decayed depends upon achieving a configuration of activity which can in turn reactivate the population.

7 Visual Working Memory and Awareness

Coward and Sun [2004] have proposed that there are five types of cognitive phenomena for which any model of conscious awareness must provide an account. Firstly, a sensory input can cause a neural activation which generates consciousness, an unconscious activation, or both. Secondly, an unconscious activation can cause physical behaviour but not complex verbal reports. Thirdly, a conscious activation can cause physical behaviour or complex verbal reports. Fourthly, a conscious activation can generate other conscious activations of associated objects including self. Finally, there is a qualitative difference between unconscious and conscious activations in response to the same perceived object, and between conscious activations in response to the same object at different times. The complexity of a conscious activation and the differences between different conscious activations in response to the same object can cause complex verbal reports, but the details of the complexity are difficult to describe, or in other words cannot cause complex verbal reports.

In the recommendation architecture model, unconscious activations are activations of portfolios within clustering which contain conditions actually present in the sensory information derived from the currently perceived object. Conscious activations also include an additional population of portfolios activated on the basis that they contain conditions recorded in the past or often active in the past at the same time as conditions in the portfolios activated by the currently perceived object. Directly activated portfolios are distinguished from indirectly activated portfolios by different frequency modulation phases of the neuron outputs indicating portfolio activations. In the expanded portfolio population, the information derived from any object other than the currently perceived object will in general be a small subset of the information which would be activated in response to a direct perception of such an object. However, one such small subset could generate a more complete set corresponding with the object by activation of portfolios often active at the same time in the past.

Because every activated portfolio is also a set of behavioural recommendations, the effect of a conscious activation is a considerable increase in the range of available behavioural recommendations, drawing on information associated to some degree with the information within the current input state.

Both conscious and unconscious activations can generate physical behaviour via competition, but complex verbal reports require extensive activation of associated information. The conscious activation in response to a perceived object can contain enough information combinations derived from other objects to cause further conscious activations which in turn generate verbal reports of those objects. The qualitative difference between unconscious and conscious activations derives from the much larger volume of activated portfolios. In a given conscious activation there will be large numbers of subsets of the activation which are also subsets of an activation which could generate naming behaviour for a different object, but which are too small to generate such behaviour alone. The presence of many such subsets which are unable to generate verbal behaviour corresponds with the perception of complexity which cannot be expressed verbally in conscious experiences [Coward and Sun 2004]. However, any one subset could potentially generate a more complete set corresponding with an object which is not present by further indirect activation.

Baars [1988, 1997] developed the classical theatre metaphor for conscious awareness into a specific theory. In this global workspace theory there are many processes active within a working memory "stage", and an attention "spotlight" highlights one of these processes. The highlighted process is the current content of consciousness. Working memory processes compete to enter the "spotlight", and information generated by the successful process is provided to a wide range of unconscious processes for learning and behaviour generation. Franklin and his collaborators [Franklin and Graesser 1999] have implemented a system with a global workspace architecture, and demonstrated that the system generates a wide range of behaviour analogous with consciousness.

From the perspective of this paper, the global workspace model is a "user manual" type description. For example, in the Franklin and Graesser implementation, there are modules corresponding with cognitive features like perception, learning, action selection, associative memory, consciousness, and emotion. Cognitive objects are represented by symbols. Although such a model may match some aspects of the phenomenology of conscious awareness, it will not reflect the way in which the physical system architecture of the brain achieves this phenomenology.

Another approach is the Neural Representation Model (NRM) developed by Aleksander and his collaborators [Aleksander, Morton and Dunmall, 2001]. This model has also been implemented in electronic form. At a conceptual level, any sensory activity which has the potential for being verbally reported is supported by a rich neural activity which correlates on a one-to-one basis with events of which the organism is aware. Such neural activity is called a "depiction", and the link between different information in the same depiction is provided by "gaze-locking" information, or neural activity which correlates with specific positions of the eyes. Attention is therefore interpreted as an internalization of the eye movement process.

The recommendation architecture differs from this NRM approach in that conscious awareness always involves an indirectly activated neural activity over and above the activity correlating with the currently perceived event. However, "gaze-locking" could well be one source of information used to manage the modulation of neuron outputs which distinguishes between those derived from the visual object in the current attention focus and other objects in the visual field.

At the implementation level there are some critical differences between the NRM and the recommendation architecture. The neuron model in NRM has binary input weights as in the case of the condition recording neuron in the recommendation architecture. A list of input vectors or conditions is maintained in both models. However, in the NRM neuron, there is a different output state associated with every input vector, and the associations between inputs and outputs are learned in a training period during which input vectors and output targets are presented together. After the training period, the neuron generates the output associated with the input vector on its list which is most similar to its current input. In the recommendation architecture condition recording neuron there is no separate training period, conditions can be added to its list at any time if they are sufficiently similar to those already on its list, provided that the need to add conditions is indicated by signals from specific other neurons. Outputs only indicate the number of conditions currently present, not the identity of the conditions. Learning can therefore occur continuously throughout the life of the system. From the perspective of this paper, the absence of learning beyond an initial training period limits the operational complexity of the features which a system like NRM can learn, and makes it a less plausible model for the relationship between neural activity and cognition.

O'Regan and Noe [2001] have argued that visual consciousness is not the activation of an internal representation of what is seen, but is generated when the organism has mastered the relationships between visual inputs and the ways in which the organism can act upon the source of those inputs. They argue that visual awareness is a matter of degree, depending on the extent to which mastery has been achieved with respect to a particular source. In the recommendation architecture, the activation of a portfolio simultaneously indicates both the detection of conditions within sensory inputs and a range of behavioural recommendations [Coward 1990]. There is therefore no distinction between "representation" and "ways of acting", although individual portfolios in the recommendation architecture are always perceptually and behaviourally ambiguous, so they are neither absolute representations nor "mastered" ways of acting. For unconscious awareness in the recommendation architecture, the activated portfolio population is strictly generated by the presence of portfolio conditions within sensory inputs and the spectrum of corresponding behavioural recommendations is correspondingly limited. In conscious awareness, a secondary population of activated portfolios is generated which expands the range of available behavioural recommendations. On the phenomenological level, O'Regan and Noe's position is therefore to some degree in agreement with the recommendation architecture approach, but the recommendation architecture provides a way in which this phenomenology results from neural processes.

8 Conclusions

The recommendation architecture is the set of bounds within which any system which must learn a complex combination of control features with limited resources will be constrained. A cognitive model within the recommendation architecture bounds, using only a limited range of information recording and access mechanisms, can provide an account for human cognitive phenomena, including semantic, episodic and working memory. The modules within the recommendation architecture resemble physiological structures in the brain. Because these modules are defined on the basis of similar system operations and therefore do not correspond with behavioural features or cognitive categories, descriptions of how such features and categories are supported are more complex than in "user manual" type descriptions. However, although "user manual" type descriptions such as Baddeley's model for working memory are helpful for organizing descriptions of cognitive phenomena, they are not capable of providing an understanding of these phenomena in terms of physiology. Such an understanding is possible with the recommendation architecture cognitive model.

Glossary

Behaviour: A motor action, combination of motor actions, or sequence of different combinations of motor actions. Also includes indirect activations of *portfolios* within *clustering*.

Clustering: A subsystem of the *recommendation architecture* which defines, permanently records and detects repetitions of *conditions*.

Competition: A subsystem of the *recommendation architecture* which receives inputs indicating the detections of groups of *conditions* (*portfolios*) detected by *clustering*, interprets the detection of a *portfolio* as a range of behavioural recommendations, adds the recommendations of all detected *portfolios* to select the strongest recommendations, and uses consequence feedback to adjust the weights which recommended the selected *behaviour.*

Component: a set of system resources customized to manage the selection decision for one system *behaviour* or behaviour type. Components are separations within *competition.*

Condition: a small set of information elements from the *information space* available to the system, each with a corresponding value. A condition occurs when all (or a significant subset) of the information set defining the condition has the values specified for the condition.

Condition similarity: Conditions are similar if a high proportion of the information defining them is drawn from a limited information set in which information sources and values are specified and each source often has its corresponding value at times when other members of the set have their corresponding values.

Condition complexity: the complexity of a condition is the total number of *sensory inputs* that contribute to the condition, either directly or via intermediate conditions

Feature: A feature is a consistent way in which a system responds to a set of similar environmental circumstances. The environmental circumstances and corresponding responses are similar from the point of view of an external observer, and are a useful way for such an observer to understand the system, but may not reflect similarities in the way the system detects *conditions* in its environment and generates *behaviours* on a more detailed level.

Information space: all the information available to a control system, derived from the past and present states of the control system itself, of the physical system being controlled, and of the environment in which the physical system operates.

Input state: The ensemble of *sensory inputs* available at one point in time.

Module: a set of system resources which have been customized to perform a group of system operations. The system operations are similar in the sense that they can all be performed efficiently by the module resources. Modules are separations within *clustering*.

Operational complexity: An operationally complex system is one in which the number of control *features* is large relative to the available information recording and processing resources.

Operational meaning: the interpretation by one *module* of the detection of a *condition* by another *module.* This interpretation is that the currently appropriate system *behaviour* is limited to a specific subset of the *behaviours* influenced by the recipient *module.* If the confidence in this interpretation is 100%, the meaning is unambiguous, otherwise it is ambiguous.

Portfolio: a set of *conditions* which are similar and programmed on one physical substrate within *clustering*. The substrate is activated if one or more of its *conditions* are detected and generates an output to *competition.* Activation of a portfolio therefore indicates both the detection of *conditions* and a range of behavioural recommendations.

Raw sensory input: A raw sensory input is the output from a sensor detecting something about the environment. The activity of one individual input in general carries no information discriminating between different behaviourally relevant states of the environment. Only combinations of inputs can carry such information.

Recommendation architecture: A set of architectural bounds within which any operationally complex system which learns will tend to be constrained. A major characteristic of the recommendation architecture bounds is separation between a modular hierarchy called *clustering* which defines and detects *conditions*, and a *competition* subsystem which interprets *condition* detections as behavioural recommendations.

Sensory input: A result of preprocessing of *raw sensory inputs* to create combinations which carry some information which can discriminate between different behaviourally relevant states of the environment.

System architecture: a description of how system information storage and processing resources are organized so that *features* can be performed efficiently.

User manual: a description of how *features* work in a way which can easily be understood by an outside observer.

References

Aleksander, I., Morton, H. and Dunmall, B. (2001). Seeing is Believing: Depictive Neuromodelling of Visual Awareness. In J. Mira and A. Prieto (Eds.). Connectionist Models of Neurons, Learning Processes and Artificial Intelligence. Lecture Notes in Computer Science 2084, 765 - 771. Berlin: Springer.

Atkinson, R. C. & Shiffrin, R. M. (1968). Human memory: A proposed system and its control processes. In K. W. Spence (Ed.). *The psychology of learning and motivation: Advances in research and theory.* New York: Academic Press.

Baars, B. (1988). *A Cognitive Theory of Consciousness*. Cambridge University Press.

Baars, B. (1997). In the Theatre of Consciousness. *Journal of Consciousness Studies,* 4, 4, 292 - 309.

Baddeley, A. D. (1986). *Working Memory.* Oxford: Oxford University Press.

Baddeley, A. D. & Hitch, G. J. (1974). Working memory. In G. Bower (Ed.). *Recent Advances in Learning and Motivation VIII*. New York: Academic Press.

Baddeley, A. D. (1966). Short-term memory for word sequences as a function of acoustic, semantic and formal similarity. *Quarterly Journal of Experimental Psychology* 18, 362 - 365.

Baddeley, A. D. (2000). Short-term and working memory. In E. Tulving & F. I. M. Craik (Eds.). *The Oxford Handbook of Memory.* Oxford: Oxford University Press.

Baddeley, A. D. & Scott, D. (1971). Short-term forgetting in the absence of proactive interference. *Quarterly Journal of Experimental Psychology* 23, 275 - 283.

Baddeley, A. D., Vallar, G. & Wilson, B. A. (1987). Sentence comprehension and phonological memory: some neuropsychological evidence. *Attention and Performance XII: The psychology of reading.* London: Erlbaum.

Ballard, D. H. (1997). *Introduction to Natural Computing*. MIT Press.

Basso, A., Spinnler, H., Vallar, G. & Zanobia, E. (1982). Left hemisphere damage and selective impairment of audio-verbal short term memory. *Neuropsychologia* 20, 263 - 274.

Butters, N. (1984). Alcoholic Korsakov's syndrome: an update, *Seminars in Neurology* 4, 2, 226-244.

Caplan, D. & Waters, G. S. (1990). Short-term memory and language comprehension: a critical review of the neuropsychological literature. In G. Vallar & T. Shallice (Eds.). *Neuropsychological Impairments to Short-Term Memory*. Cambridge: Cambridge University Press.

Corkin, S. (2002). What's new with the amnesic patient H.M.? *Nature Reviews (Neuroscience)* 3, 153 - 159.

Coward, L. A. (1990). *Pattern Thinking*. New York: Praeger.

Coward, L. A. (2000). A Functional Architecture Approach to Neural Systems. *International Journal of Systems Research and Information Systems* 9, 69 - 120.

Coward, L. A. (2001). The recommendation architecture: lessons from the design of large scale electronic systems for cognitive science. *Journal of Cognitive Systems Research*, 2, 2, 111-156.

Coward, L. A., Gedeon, T. & Kenworthy, W. (2001). Application of the recommendation architecture to telecommunications network management. *International Journal of Neural Systems* 11(4), 323-327.

Coward, L. A. & Sun, R. (2004). Some criteria for an effective scientific theory of consciousness and examples of preliminary attempts at such a theory. *Consciousness and Cognition*. In press.

Della Sala, S., Gray, C. Baddeley, A., Allamano, N. & Wilson, L. (1999). Pattern span: a tool for unwelding visuo-spatial memory. *Neuropsychologia* 37, 1189 - 1199.

Franklin, S. and Graesser, A. (1999). A Software Agent Model of Consciousness. *Consciousness and Cognition* 8, 285 - 305.

Glanzer, M. (1972). Storage mechanisms in recall. In G. H. Bower (Ed.). *The Psychology of Learning and Motivation: Advances in Research and Theory*. New York: Academic Press.

Hebb, D. C. (1949). *The Organization of Behaviour*. New York: Wiley

Konig, P., Engel, A. K., & Singer, W. (1996). Integrator or coincidence detector ? The role of the cortical neuron revisited. *Trends in Neurosciences* 19(4), 130 - 137.

Llinas, R., Ribary, U., Joliot, M., & Wang, X.-J. (1994). Content and context in temporal thalamocortical binding. In Buzsaki G., Llinas, R., Singer, W., Berthoz, A. & Christen, Y. (Eds.). *Temporal Coding in the Brain*. Berlin: Springer.

McCarthy, R. A. & Warrington, E. K. (1987). The double dissociation of short term memory for lists and sentences: evidence from aphasia. *Brain* 110, 1545 - 1563.

McCarthy, R. A. & Warrington, E. K. (1990). *Cognitive Neuropsychology: A Clinical Introduction, (Chapter 13: Short-Term Memory)*. San Diego: Academic Press.

Neisser, U., & Libby, L. K. (2000). Remembering life experiences. In E. Tulving and F. I. M. Craik (Eds.).*The Oxford Handbook of Memory*. Oxford: Oxford University Press.

Nortel Networks (2001). DMS-100/500 Feature Planning Guide. http://www.nortelnetworks.com/products/01/dms100w/doclib.html

O'Regan, J. K. and Noe, A. (2001). A sensorimotor account of vision and visual consciousness. Behavioral and Brain Sciences 24, 939 - 973.

Peterson, L. R. & Peterson, M. J. (1959). Short-term retention of individual verbal items. *Journal of Experimental Psychology* 58, 193 – 198.

Rips, L., Shoben, J. & Smith, E. Semantic Distance and Verification of Semantic Relations. *Journal of Verbal Learning and Verbal Behaviour* 12, 1 – 20.

Saffran, E. M. & Marin, E. S. M. (1975). Immediate memory for word lists and sentences in a patient with a deficient auditory short term memory. *Brain and Language* 2, 420 - 433.

Shallice, T. & Warrington, E. K. (1970). Independent functioning of the verbal memory stores: a neuropsychological study. *Quarterly Journal of Experimental Psychology* 22, 261 - 273.

Shallice, T. & Warrington, E. K. (1977). Auditory-verbal short term memory impairment and spontaneous speech. *Brain and Language* 4, 479 - 491.

Von der Malsburg, C. (1981). The Correlation Theory of Brain Function. Internal Report 81-2, Max Planck Institute for Biophysical Chemistry, Gottingen, Germany.

Warrington, E. K., Longue, V. & Pratt R. T. C. (1971). The anatomical localization of selective impairment of auditory verbal short-term memory. *Neuropsychologia* 9, 377 - 387.

Warrington, E. K. & Shallice, T. (1969). The selective impairment of auditory verbal short-term memory. *Brain* 92, 885 - 896.