Biologically Inspired Cognitive Architectures 2011
A.V. Samsonovich and K.R. Jóhannsdóttir (Eds.)
IOS Press, 2011
© 2011 The authors and IOS Press. All rights reserved.
doi:10.3233/978-1-60750-959-2-79

Architectures of Complex Learning Systems

L. Andrew COWARD and Tamas O. GEDEON

Australian National University, Canberra, ACT 0200, Australia

Abstract. A system which performs a complex combination of behaviours has two superficially independent architectures. One is the functional architecture, which separates the behavioural features of the system into feature modules made up of groups of similar behaviours, and defines the interactions between features. The other is the system architecture (alternatively called the physical or information process architecture) which separates the physical information handling resources of the system into modules that perform different types of information processes, each module optimized to perform a different type of process. Any one feature module will employ information processes performed by many or all resource modules. Many different functional architectures are possible, but the need to limit the resources supporting large numbers of different behaviours tends to constrain the form of the system architecture. In the limiting case as the ratio of the number of behaviours learned to the available resources becomes very large, the system architecture is constrained into a very specific form. In the case of a complex learning system this form is called the recommendation architecture. Because there are natural selection advantages for species that require fewer neural resources to learn a given set of behaviours, there is a tendency for the recommendation architecture form to appear in biological brains including human, mammal and avian brains. A system designed to perform a complex combination of behaviours will be much more effective if designed within this form..

1 Introduction

In this paper, a complex system will be defined as one which can perform a very large number of different purposeful behaviours in a way that is appropriate for a complex environment. Purposeful means that it is possible to identify system objectives which motivate behaviour selection. Such systems could also be called complex functional systems, but in this paper the simpler term will be used. Purely physical complex systems such as the weather are not purposeful as defined and are therefore not included. The behaviours of complex electronic systems like flight control computers or telecommunications network managers are specified in advance under external intellectual control. A complex learning system must learn a large combination of behaviours appropriate to achieve objectives in a very complex environment.

The behaviours performed by a complex electronic system are typically organized into different types of functions, called features or applications. Each function is a collection of similar or closely related behaviours. The definition of the different functions and the interactions between them is called the functional architecture, and can be very different for different systems. For example, a personal computer has separate applications for word processing, web access etc. with specific, limited ways in which they can exchange information. However, it is striking that any electronic

Copyright © 2011. IOS Press, Incorporated. All rights reserved.

Biologically Inspired Cognitive Architectures 2011 : Proceedings of the Second Annual Meeting of the BICA Society, edited by A.V. Samsonovich, and K.R. Jóhannsdóttir, IOS Press, Incorporated, 2011. ProQuest Ebook Central, http://ebookcentral.proquest.com/lib/anu/detail.action?docID=836
Created from anu on 2021-01-26 20:47:41.

system that performs a complex combination of behaviours always has the same physical (or information process) architecture at the highest level. This physical architecture separates memory and processing subsystems, with a common bus linking these subsystems together and with subsystems receiving inputs from the environment and generating outputs driving behaviours. All applications make use of the same memory and processing resources, and, conversely, any one memory or processing resource will contribute to many or all different applications.

Mammal brains of different species all have objectives to survive and reproduce, but are able to learn to perform very different combinations of behaviours, depending on the environmental niche occupied. However, all mammal brains have a very similar physical architecture at the highest level, with cortex, hippocampus, basal ganglia, thalamus, amygdala and cerebellum. Close examination of the avian brain reveals a very similar physical architecture with the same major subsystems [1]. In the case of the human brain, one "feature" like episodic memory uses many different physical resources, and another feature like imagination uses many of the same resources [2].

Both designed and learning systems must obtain information from their environment, and use this information to determine an appropriate behaviour at each point in time, selecting within a wide range of options. Such systems must therefore detect conditions within their input information, and associate different behaviours with different combinations of conditions. In systems that are designed, both conditions and associations between conditions and behaviours are specified by the designer. In systems that learn, most of the conditions and the associations between these conditions and appropriate behaviours must be defined heuristically on the basis of experience.

2 Practical constraints on complex electronic systems

Any complex system requires information handling resources in order to perform its functions. In the case of an electronic system these include the transistors and other components that constitute memory and processing subsystems. A system architecture that can perform a given set of functions with fewer resources will have an advantage over an architecture that requires more resources, although other considerations will interact with this resource constraint.

For an electronic system, these other practical considerations include modifiability, repairability, constructability and synchronicity. Modifiability means that it must be possible to add or change features without interfering with the operation of other features. Constructability means that it must be possible to build many copies of the system from blueprints by a process that minimizes the risk of errors and is therefore not too complex. Repairability means that it must be possible to diagnose and correct construction errors and later component failures or damage. Synchronicity means that it must be possible to handle a constant sequence of inputs from a continuously changing environment without confusing information derived from the environment at different times.

These different considerations are often in conflict. For example, if every feature had completely separate information handling resources, one feature could always be modified with no effect on other features. However, this would have a very high cost in resources. If information derived from the environment at different times was processed by different resources, synchronicity would be guaranteed at a very high cost

Copyright © 2011. IOS Press, Incorporated. All rights reserved.

Biologically Inspired Cognitive Architectures 2011 : Proceedings of the Second Annual Meeting of the BICA Society, edited by A.V. Samsonovich, and K.R. Jóhannsdóttir, IOS Press, Incorporated, 2011. ProQuest Ebook Central, http://ebookcentral.proquest.com/lib/anu/detail.action?docID=836
Created from anu on 2021-01-26 20:47:41.

in resources. The need to find an adequate compromise between conflicting practical considerations places strong constraints on the physical architecture [3].

2.1 Condition Detection and Modules

Any one condition detected within environmental inputs will be a list of relevant inputs and a specified state for each input. The condition occurs if each relevant input (or a high proportion) is in the specified state. In practice a condition may be a very complex combination of system inputs and states. Conditions must therefore be specified (by the system designer) and the system inputs at each point in time must be tested for the presence of each condition by comparison between the inputs and the condition specification. Both the specification of a condition and the testing for the presence of a condition will require information handling resources. A condition can be viewed as a group of smaller conditions (or subconditions). Two conditions are similar if some of their subconditions are the same.

Resource requirements can be considerably reduced if similar conditions are collected into modules, within which the resources to specify and detect any overlaps (or identical subconditions) are shared. This resource advantage drives a hierarchy of modules, with the most detailed modules detecting groups of very similar conditions, the similarity making it possible to share resources within a module. Intermediate modules are made up of the resources of a group of detailed modules, and detect a range of conditions with somewhat less similarity. Higher level modules are made up of groups of intermediate modules with lower similarity but still enough to achieve resource economies. Similarity between submodules of a higher module means that the submodules require some of the same subconditions. Sharing means that there is an information exchange between the modules, with each shared subcondition being detected by one module and detections communicated to other modules.

This type of modular hierarchy has considerable resource advantages, but creates problems with modifiability. Any one module will be required to support many different features. Change to a feature will often require changes to the condition definitions used by the feature, but such changes could have undesirable side effects on other features using the same condition. Furthermore, the undesirable effects of a change can propagate to other modules via information exchange. Modifiability therefore requires minimization of information exchange, and the modular hierarchy must be a compromise between these conflicting requirements.

One further constraint on a modular hierarchy is constructability. If every module were completely different, then the specification of the construction process would be very complex and the probability of errors correspondingly high. However, even on one level modules must be different in order to detect different conditions. Hence the compromise with constructability will result in modules on one level that are generally similar but differ in detail.

2.2 Handling of sequences of input states

Inputs to a complex system from the environment are continuous, and conditions must be detected in these inputs. Some conditions will be made up of input states which must all be present at the same time, others of input states which occur in a

Copyright © 2011. IOS Press, Incorporated. All rights reserved.

Biologically Inspired Cognitive Architectures 2011 : Proceedings of the Second Annual Meeting of the BICA Society, edited by A.V. Samsonovich, and K.R. Jóhannsdóttir, IOS Press, Incorporated, 2011. ProQuest Ebook Central, http://ebookcentral.proquest.com/lib/anu/detail.action?docID=836
Created from anu on 2021-01-26 20:47:41.

defined fashion over time. Hence the temporal relationships between condition detections must be recorded in some way. The only alternative would be to have multiple duplicates of resources to specify and detect conditions at different points in time. This would place impractical demands on resources.

The most precise practical way to support such recording is to use the same condition specification and detection resources, but record detections in a separate memory along with tags indicating detection time. Because all conditions are exactly specified and precisely detected, it is possible for condition detections to have unambiguous behavioural meanings. In other words, they can be interpreted as commands. There is therefore a major separation in such an architecture between a subsystem which records the occurrence of conditions (a memory) and a subsystem which executes commands (a processor). In other words, the combination of practical considerations results in the ubiquitous von Neumann architecture.

2.3 Practical constraints on a complex learning system

The information available to a complex learning system is made up of inputs from the environment and inputs from within the system itself. The system must detect conditions within its inputs which are effective for discriminating between circumstances in which different behaviours are appropriate. Conditions detected within internal inputs will be important for guiding a special class of behaviours, called reward behaviours. Such conditions discriminate between satisfactory and unsatisfactory internal situations, and can be used to reward or punish recent behaviours, which have probably played some role in reaching the current situation. Reward behaviours increase or decrease the probability that behaviours similar to those recently performed will be performed in similar circumstances in the future. Given that conditions within the environment must largely be defined heuristically, reward behaviours are critical for associating such conditions with appropriate behaviours.

A complex learning system is also subject to the same practical constraints: resource limits, modifiability, constructability, repairability and synchronicity. Modifiability in this context means the ability to learn without undesirable side effects on past learning. One effect will be organization of condition detection into a modular hierarchy as discussed earlier for complex electronic systems. A module on any level has a receptive field, defined by the group of conditions it contains. This receptive field is detected if a high proportion of these conditions is detected. However, the requirement to heuristically define the conditions which will be detected and also to define the associations between conditions and behaviours results in some qualitatively different architectural constraints from complex electronic systems designed under external intellectual control. Enough different conditions must be defined to permit discrimination between circumstances in which different behaviours are appropriate. The problem is the source of guidance to making changes to conditions. Consequence feedback following a behaviour is a possible source, but because any one condition will support many different behaviours, changes to the definition of a condition based on consequence feedback following one behaviour will have unpredictable effects on all the other behaviours dependent on the condition. As a result, consequence feedback cannot be used directly to guide changes to conditions or receptive fields. Modules cannot therefore be evolved to correspond exactly with the circumstances in which one behaviour is always appropriate. In contrast with the von Neumann architecture,

Copyright © 2011. IOS Press, Incorporated. All rights reserved.

Biologically Inspired Cognitive Architectures 2011 : Proceedings of the Second Annual Meeting of the BICA Society, edited by A.V. Samsonovich, and K.R. Jóhannsdóttir, IOS Press, Incorporated, 2011. ProQuest Ebook Central, http://ebookcentral.proquest.com/lib/anu/detail.action?docID=836
Created from anu on 2021-01-26 20:47:41.

condition detections cannot correspond with instructions, only with recommendations in favour of a range of different behaviours.

Recommendations must be interpreted into a behaviour that is actually implemented, and the weights of such recommendations must be adjusted by consequence feedback. Because such consequence feedback cannot be applied to condition definitions, there must be a separate subsystem which receives condition detections, interprets such detections as recommendations, determines the strongest recommendation across all current condition detections, implements that behaviour, and later adjusts recommendation strengths on the basis of consequence feedback.

As the ratio of the number of behaviours to be learned to the available resources increases, a complex learning system will therefore tend to be constrained into an architecture with two major subsystems. One subsystem (called clustering) defines and detects conditions within the information available to the system, the other subsystem (called competition) interprets condition detections as behavioural recommendations. This architecture is called the recommendation architecture, and is analogous with but qualitatively different from the memory, processing separation in complex electronic systems designed under external intellectual control.

There are four types of behaviour requiring special handling. These are condition change behaviours, recommendation weight change behaviours, general behaviour type recommendation selection, and behaviour sequence management.

Changes to modules always risk undesirable side effects on behaviours influenced by the changed conditions, and the behaviours of changing the conditions that define a module must therefore be tightly managed. This management requires detection of conditions able to discriminate between circumstances in which module changes are appropriate and inappropriate, and interpretation of such condition detections as recommendations in favour of changes to specific modules. The risk of undesirable side effects means that in general, changes to a module can only be by addition of a few extra conditions that are similar to those already detected by the module. The receptive field of the module is expanded slightly, but will continue to be detected in all the circumstances in which it was previously detected. Such receptive field expansions must only occur if the range of recommendations otherwise available is too low to achieve a high integrity behaviour selection. In other words, if the total number of receptive field detections is less than some minimum, receptive field expansions must occur to bring the number up to the minimum. A resource management subsystem is therefore required which detects receptive fields that are appropriate to be interpreted as recommendations in favour of expansions of other receptive fields. The resource management subsystem will therefore have the same clustering/competition separation.

Changes to recommendation weights in competition are also behaviours which must be tightly managed. These reward type behaviours will therefore also require a clustering/competition separation. Similarly, it may sometimes be appropriate to bias behaviour selection in favour of different general types of behaviour. The circumstances in which such biases are appropriate must be discriminated by receptive fields in clustering and implemented by interpretation of receptive field detections in competition. Finally, there may be sequences of actions which are often required to be implemented in the same order. Higher speed and accuracy can be achieved by recording such sequences in a separate subsystem, and implementing them in response to an initial trigger rather than by receptive field detection and interpretation after each individual action.

The architectural form which results from these considerations is illustrated in

Copyright © 2011. IOS Press, Incorporated. All rights reserved.

Biologically Inspired Cognitive Architectures 2011 : Proceedings of the Second Annual Meeting of the BICA Society, edited by A.V. Samsonovich, and K.R. Jóhannsdóttir, IOS Press, Incorporated, 2011. ProQuest Ebook Central, http://ebookcentral.proquest.com/lib/anu/detail.action?docID=836
Created from anu on 2021-01-26 20:47:41.

figure 1. As the ratio of behaviours which must be learned to available information handling resources increases, a complex learning system will tend to be constrained more tightly into this form. In an analogous fashion, systems that are designed to perform a complex combination of behaviours tend to be constrained into the von Neumann architectural form.

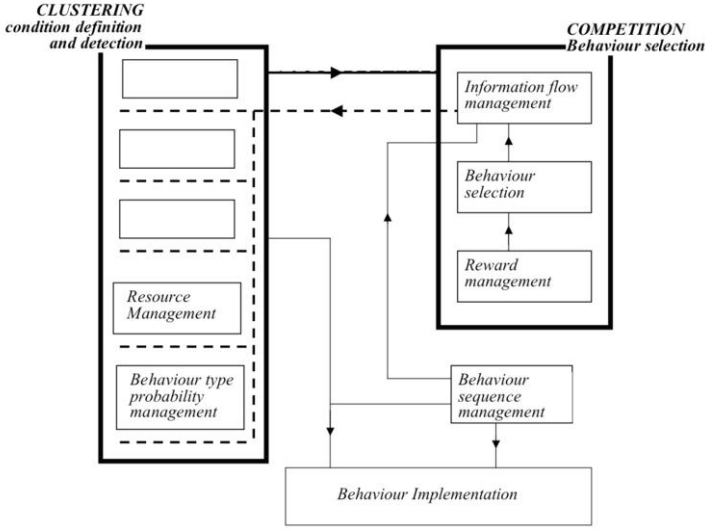

Figure. 1. The recommendation architecture. This architecture is the limit towards which a learning system will tend to be constrained as the ratio of behaviour to available information handling resources increases. Different subsystems perform different types of information processes. Any one behaviour will require many different types of information process performed by many different subsystems. Conversely, any one subsystem will perform information processes in support of many different behaviours. *Clustering* defines and detects conditions within the information available to the system , including both external environmental and internal state information. Conditions are organized into groups (called receptive fields), and if a high proportion of the conditions are present, the detection of the receptive field is signalled to competition. Withing *competition*, each receptive field detection is interpreted as a range of behavioural recommendations, each with an individual weight. *Behaviour selection* determines and implements the most strongly recommended behaviour . *Reward management* determines the total recommendation weight in favour of reward behaviours. If implemented, reward behaviours adjust the weights of recently implemented behaviours. Many behaviours are implemented by releasing receptive field detections from one module of clustering to another, or outside clustering to drive externally directed behaviours. *Information flow management* manages these information releases in detail, with guidance from the behaviour selection component. Note that information flow management gates the release of information but does not change the content. The *resource management* module uses special purpose receptive fields to determine when and where changes will be made to receptive fields throughout clustering. Behaviour type probability management detects special purpose receptive fields that recommend favouring different general types of behaviour are currently appropriate. Selection of the most favoured type is implemented by reducing the threshold for detection of receptive fields elsewhere in clustering that strongly recommend behaviours of that type. Many often utilized behaviours are sequences of actions.. Such sequences could be implemented by receptive field detection and behaviour selection prior to each action, but more rapid and accurate execution can be achieved by recording the sequence in the behaviour sequence management component, which executes the sequence on the basis of initial receptive field detections.

There is one further important architectural consideration imposed by practical considerations. Conditions must be detected within environmental inputs at a single point in time (e.g. within a single visual object, guiding appropriate behaviour in response to the object). More complex conditions must be detected that incorporate

Copyright © 2011. IOS Press, Incorporated. All rights reserved.

Biologically Inspired Cognitive Architectures 2011 : Proceedings of the Second Annual Meeting of the BICA Society, edited by A.V. Samsonovich, and K.R. Jóhannsdóttir, IOS Press, Incorporated, 2011. ProQuest Ebook Central, http://ebookcentral.proquest.com/lib/anu/detail.action?docID=836
Created from anu on 2021-01-26 20:47:41.

conditions detected at multiple points in time (e.g. within a group of objects, guiding appropriate behaviour in response to the group). To detect the more complex conditions, condition detections derived from information at different points in time must be simultaneously active, but without confusion of the information from different points in time. Avoidance of confusion requires either spatial or temporal separation. Spatial separation would require duplicate resources for condition detection. Temporal separation can be achieved by separating condition detections within different inputs into different time slots in the same physical condition detection resources. This solution is more resource effective and appears to be used in the human brain [4].

The cognitive deficits that result from damage to different anatomical structures in the brain [5] indicate that the human and other mammal brains have been constrained by natural selection pressures into the recommendation architecture form.

3 Indirect receptive field activation and cognitive processing

A module is activated by detection of its receptive field in current environmental inputs. Such activations make the behavioural recommendations corresponding with the detections available to guide immediate behaviour. However, complex learning requires that behavioural guidance has access to much more information than just these current inputs. If a receptive field of a module is currently not detected, but has often been detected in the past at the same time as many of the currently active receptive field modules, then that inactive module may have relevant recommendation strengths to contribute to current behaviour. Hence there can be behavioural value in the capability to indirectly activate modules on the basis of past temporally correlated activity. Different types of temporally correlated activity could be relevant: recent or frequent past simultaneous activity, or past simultaneous receptive field changes. In each of these types, past activity of the inactive module could be simultaneous with, shortly before or shortly after the past activity of the active modules.

If uncontrolled, such indirect activations would result in chaotic patterns of module activation. Indirect activations must therefore be behaviours that are recommended by module activations and only implemented if there is sufficient total recommendation strength into competition. Indirectly activated modules may themselves have recommendation strengths in favour of indirectly activating yet other modules, resulting in a sequence of indirectly activated module populations with a considerable degree of independence from current inputs from the external environment. It can be demonstrated that the three types indirect activation (on the basis of recent, frequent past, and past change activity) can support, respectively, priming, semantic and episodic memory in human beings [5]. More complex cognitive processes can be supported by ordered sequences of activations, including both direct and indirect activations [6;7].

4 Simulations of the recommendation architecture

The recommendation architecture can be simulated on a von Neumann machine [3]. Such simulations have demonstrated that the information processes of the

Copyright © 2011. IOS Press, Incorporated. All rights reserved.

Biologically Inspired Cognitive Architectures 2011 : Proceedings of the Second Annual Meeting of the BICA Society, edited by A.V. Samsonovich, and K.R. Jóhannsdóttir, IOS Press, Incorporated, 2011. ProQuest Ebook Central, http://ebookcentral.proquest.com/lib/anu/detail.action?docID=836
Created from anu on 2021-01-26 20:47:41.

recommendation architecture can organize experience heuristically into receptive fields able to discriminate between situations with behaviourally different implications [3;8]. Simple reward feedback applied to a separate competition subsystem receiving the receptive field detections results in learning of new behaviours with minimal interference with past behavioural learning [9]. Receptive field detections within different input states can be separated in different time slots [4]. Indirect activation of receptive fields can support more complex cognitive processing [3].

Organization of experience into arrays of receptive fields has some general resemblances with neural network approaches like Kohonen maps [10] and adaptive resonance [11]. However, as discussed more fully in Coward [3;5] there are major differences from these approaches. In particular, the concept of receptive fields that in most circumstances only expand, with detailed management of when such expansions can occur, is qualitatively different from alternative network algorithms. The approach also avoids the problem of catastrophic interference between new and prior learning found in many neural network algorithms [12].

5 Conclusions

If a system is to be designed that can learn to perform many different behaviours, it is important to focus on the information processes that are supported and how those processes are organized into a system architecture, not just on the functional architecture.

If many different conditions must be detectable in order to be able to discriminate between circumstances in which different behaviours are appropriate, and if most of those conditions must be defined heuristically, then the information processes required will be those of the recommendation architecture, and it will be necessary to organize system resources as in figure 1 to optimize the performance of those processes. A system that addresses a limited domain, with a significant amount of preprogrammed knowledge, will not experience these architectural constraints to the same degree.

The individual information processes are present in the brain, and have individually been tested by electronic simulation. The next step is development of a system which must learn a complex environment with minimal guidance, and perform a complex combination of appropriate behaviours in response to that environment.

References

1. Jarvis ED, Gunturkun O, Bruce L et al. (2005). The avian brain and a new understanding of vertebrate brain evolution. Nature Reviews Neuroscience 6, 151 – 159.
2. Addis DA, Wong AT, Schacter DL (2007) Remembering the past and imagining the future: Common and distinct neural substrates during event construction and elaboration. Neuropsychologia 45:1363–1377.
3. Coward, LA (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.
4. Coward LA (2004). Simulation of a Proposed Binding Model. *Brain Inspired Cognitive Systems 2004*, L. S. Smith, A. Hussain and I. Aleksander, (editors), University of Stirling: Stirling.
5. Coward LA (2005). *A System Architecture Approach to the Brain: from Neurons to Consciousness*. New York: Nova Science Publishers.
6. Coward LA, Gedeon TO (2009). Implications of Resource Limitations for a Conscious Machine. *Neurocomputing* 72, 767 - 788.

Copyright © 2011. IOS Press, Incorporated. All rights reserved.

Biologically Inspired Cognitive Architectures 2011 : Proceedings of the Second Annual Meeting of the BICA Society, edited by A.V. Samsonovich, and K.R. Jóhannsdóttir, IOS Press, Incorporated, 2011. ProQuest Ebook Central, http://ebookcentral.proquest.com/lib/anu/detail.action?docID=836
Created from anu on 2021-01-26 20:47:41.

7. Coward LA (2011). Modelling Memory and Learning Consistently from Psychology to Physiology. In V Cutsuridis et al. (eds.), *Perception-Action Cycle: Models, Architectures, and Hardware*, pp 52 – 123. Springer Series in Cognitive and Neural Systems.
8. Coward, LA (2009). The Hippocampal System as the Manager of Neocortical Declarative Memory Resources. in *Connectionist Models of Behaviour and Cognition II*, J. Mayor, N. Ruh and K. Plunkett (eds), 67 - 78.
9. Coward LA, Gedeon TO, Ratanayake, U. (2004). Managing Interference between Prior and Later learning. ICONIP 2004, Calcutta. Lecture Notes in Computer Science 3316, 458-464.
10. Kohonen, T. (1995). Self Organizing Maps. Springer Series in Information Sciences 30.
11. Carpenter, GA, Grossberg, S. (1988). The ART of Adaptive Pattern Recognition by a Self-Organizing Neural Network, IEEE Computer, 3, 77-88.
12. French, RM (1999). Catastrophic Forgetting in Connectionist Networks. Trends in Cognitive Science 3(4), 128-135.

Copyright © 2011. IOS Press, Incorporated. All rights reserved.

Biologically Inspired Cognitive Architectures 2011 : Proceedings of the Second Annual Meeting of the BICA Society, edited by A.V. Samsonovich, and K.R. Jóhannsdóttir, IOS Press, Incorporated, 2011. ProQuest Ebook Central, http://ebookcentral.proquest.com/lib/anu/detail.action?docID=836
Created from anu on 2021-01-26 20:47:41.