BRAIN MECHANISMS

Linking Cognitive Phenomena to Neuron Activity

L. ANDREW COWARD

CONTENTS

CHAPTER 1

Introduction

In contrast with conventional belief, it is actually straightforward to understand higher cognition and human consciousness in terms of brain anatomy, physiology, and chemistry. It is no more difficult—perhaps it is even easier—to understand the cognitive capabilities of the brain in terms of neuron processes than to understand the features of the most complex electronic systems in terms of transistor processes. This book will explain how and why such understanding of the brain is possible.

Explanations of the brain are sometimes offered in terms of single anatomical or physiological structures being the key to some psychological phenomenon: the frontal cortex as the seat of higher cognition, the gamma band frequency observed in the EEG as the neural correlate of consciousness, the neurochemical dopamine as the basis of desire, the amygdala as the source of the fear emotion, and so on. Unfortunately, these types of simple correlations do not exist. It is necessary to think about the brain as a system in which many or all physiological processes and anatomical structures contribute to every cognitive ability. An analogous situation exists for computers: no one transistor, integrated circuit, or printed circuit assembly is the location where one particular application, like internet browsing, is carried out. Rather, almost all of the hardware components may contribute to every application.

A computer system with many billions of transistors and other components can be understood. In computer science, we have developed ways to organize the information about these very complex systems in such a way that they can be designed and modified. These same techniques can be applied to organizing the vast amount of information acquired by the neurosciences so that human beings can understand their own brains. Simple versions of these techniques are actually very familiar; we use them every day to understand the world around us. But we do not always realize that we are using these techniques or think about how and why they work.

It is important to point out that all this is not to imply that there is any resemblance between brains and computers; they are qualitatively different types of systems. Rather, it is to say that techniques used to understand computing systems by organizing information about billions of components can be applied to understanding the brain.

From 1969 to 1999, I participated in all sorts of different aspects of the design of some very complex telecommunications systems. The company where I was employed was called Northern Electric, and for many of those years I worked in a Northern Electric subsidiary called Bell Northern Research (BNR), where most of the product design took place. Later, Northern Electric was renamed Northern Telecom, then Nortel Networks, and late in first decade of the twenty-first century it encountered problems that caused it to go out of existence. But in the years when I was working for the company, it was a pioneer and perhaps the best in the world at designing extremely complex real-time electronic systems.

In the 1960s, transistor and software technologies were relatively new, and until the 1970s, telecommunications systems had all relied on simple mechanical switches called relays to support telephone dialling and speech connections. Telephone exchanges that connected all the thousands or tens of thousands of telephones in one geographical area contained vast numbers of these relays. Northern Electric/BNR was the first in the world to replace all the relays in these systems with integrated circuit and software technologies. These technologies were becoming more widely available, but their application to telecommunications was a great challenge for several reasons.

The first reason was that telephone exchanges were real-time control systems. A connection had to be created between two telephones within seconds of dialling, even if thousands of other telephones were also making connections at the same time. This real-time requirement was hard to meet with the electronic information processing speeds available in the 1970s.

Secondly, in telephone systems before the 1970s, voices were represented by electrical signals that varied in time just like the variations in air pressure that are the sounds of a voice. This is called an analogue representation. For electronic systems it is necessary to convert these voice signals into digital form, representing the sound of the voice at each point in time by a series of 1s and 0s. This is called a digital representation. The electrical analogue of a voice speaking into a telephone is converted into digital form before being processed through the exchange to its destination telephone. At the destination it is converted back to analogue form to generate the sound of the voice. Analogue-to-digital and digital-to-analogue conversions require a lot of processing power, which was again difficult to achieve at a reasonable cost with the technology of the 1970s.

Thirdly, the mechanical systems were highly reliable in the sense that although relay failures were not unusual, such a relay failure might cause one or two telephones to lose service, but it was extremely rare for all the telephones in a geographical area to lose service at the same time. You could always go to a neighbour to make an urgent call if your own telephone didn't work. However, failure of an electronic exchange might take all the telephones in a city out of service. The risk of this type of failure led to the requirement that one of the new electronic exchanges could not be totally out of service for more than two hours in forty years. These two hours had to include both hardware and software failures, plus any downtime resulting from a need to upgrade hardware or software. This stringent target was actually met by the Nortel electronic systems.

The design culture of Northern Electric/BNR was very self-reliant, and in any case the new technologies were in their early development stages and not available elsewhere. From the mid-1950s the company had been manufacturing its own transistors, and from the early 1970s it both designed and manufactured its own integrated circuits containing many transistors. As the complexity of the required integrated circuits increased, BNR created the tools needed to design those circuits, and also the design tools for the printed circuit assemblies to interconnect the integrated circuits. In response to the need for software to drive the new systems, BNR created software languages appropriate for real-time systems, the compilers to translate software into machine code to drive the hardware, and the user environments in which the software could be designed and tested. All this made it possible to introduce the first fully electronic telephone switching system, called SL-1. Available in the mid-1970s, it was designed for use within a business. Designing the first fully electronic public switching system in the late 1970s and early 1980s required of the order of five thousand designer years of effort. One such system contained of the order of five billion transistors, a huge number for that time.

It happened over the course of my career that at different times I had responsibilities in many of the key design areas, including system design, transistor and integrated circuit design, software design, compiler design, user-feature design, and design of the design environments for integrated circuits, printed circuit assemblies, and software. So I saw first-hand many of the problems encountered in the top-to-bottom design of an extremely complex system, from transistor design to user-feature design. I therefore worked on many different levels with the techniques that were used in combination to ensure that the design information about a very complex electronic system with billions of components was manageable by human designers.

A system design was first created at very high level. The system was conceived as perhaps half a dozen subsystems. At this initial point, the information processes performed by each subsystem and the way the subsystems interacted were defined. Then the way these subsystems would step by step perform a range of user functions was imagined. To allow the subsystems to operate effectively, the information processes in one subsystem needed to be carried out with relatively little ongoing interaction with processes in other subsystems. To put this another way, information exchange between the subsystems in the course of performing these functions needed to be minimized. This consideration led to changes in the way in which information processes were divided up between the subsystems.

For example, there are two different major types of information processes performed by a computer system: memory processes that store or retrieve data and instruction processes that execute commands. In the system design process, initially one subsystem might perform all the memory processes and a different subsystem will perform all the instruction processes. Once this division of processes between the subsystems was made, the subsystems could be designed in more detail by different groups of engineers. In the course of this more detailed design, sometimes it was necessary to move a few of the information processes originally assigned to one subsystem into a different subsystem. To keep information exchange relatively low, a small proportion of memory-type processes might need to be transferred to the instruction subsystem, and vice versa. This made the high-level design description only approximate, but it was still retained because for most purposes it was still good enough, and any other separation into subsystems was less accurate. The memory subsystem still contained the vast majority of memory processes and only a few instruction processes. Most of the time when modifying the design further, the approximation was a good guide, although it was important to be aware of situations when it could be misleading.

This use of approximate descriptions on a higher level to guide more detailed design was also used for even more detailed design, and so on. In other words, a manageable design process was heavily dependent on the creation of a hierarchy of descriptions. In this hierarchy the higher-level descriptions were more approximate, but it was known where they were approximate, and they could be mapped into more detailed descriptions when required. A key point was that the relationships between more detailed descriptions of different parts of the system could only be understood by means of higher-level descriptions.

In the early 1980s, I began to wonder if these techniques could be applied in any way to understanding the brain. In 1984, the idea of a high-level description of the brain crystallized in my mind. This description involved subsystems of a novel architecture I later called the *recommendation architecture*. Over the next few years, I developed this idea when I could spare the mental bandwidth mostly needed for my work at BNR/Nortel, eventually leading to my first book on understanding the brain, *Pattern Thinking*, published by Praeger in 1990. This book contained many of the architectural ideas and connections with neuroscience that I later mapped more precisely into anatomy, physiology, and neurochemistry.

While the book was in publication, I implemented a software model for the cortex within the architecture I had conceived, but I had little time to develop the ideas further. Then in 1999, I took early retirement from Nortel and put much more energy into developing these ideas. At a conference on neural networks a year or two before retiring I made a contact with Professor Tom Gedeon, then at the University of New South Wales and later at the Australian National University (ANU). This contact led to a long-term relationship with ANU that included both research and teaching.

One of the first steps I took after leaving Nortel was to develop more formally the concept that a range of practical considerations imposed major constraints on the architecture of any complex control system. Two of the key considerations are (1) the need to limit the amount of information processing hardware required to carry out all the system features and (2) the need to make changes to system features without introducing undesirable side effects on other features. I was able to demonstrate that these types of practical considerations show that the hardware of any complex control system that must perform large numbers of different features with limited resources can only take one of two general forms. If all the features of the system are designed, the architectural form is the familiar memory/processing architecture, or *instruction architecture*. This instruction architecture is ubiquitous in computer systems. However, if the features of the system must be learned from experience, the instruction architecture is impractical, and the only possible form is the *recommendation architecture* I had described in my earlier book.

In the instruction architecture, it is the general use of two types of information processes, the instruction and the data read/write, that makes it possible for designers to understand the system. A key insight in my 1990 book was that in the recommendation architecture there are two analogous but qualitatively different types of information processes, the condition define/detect and the behaviour recommendation define/integrate that make it possible to understand recommendation architecture systems like the brain.

After this I went on to demonstrate that the major anatomical structures of the brain correspond with the subsystems required in a system with the recommendation architecture to perform different types of information processes. In computer systems there is always a memory specializing in data read/write processes and a processor specializing in instruction processes. Analogously, in the brain there is a cortex specializing in condition define/detect processes and subcortical structures specializing in behavioural recommendation define/integrate processes.

From there I could specify the processes performed by different anatomical structures more precisely and demonstrate that the detailed anatomy of the brain is optimised to perform these processes effectively. For example, the role of the hippocampus in managing cortical resources efficiently, and how it performs that role, steadily became clearer.

Psychology has long recognized that there are a number of qualitatively different types of memory, finding differences between memory for facts, memory for events, and memory for skills, plus a couple of short-term memory types called *working memory* and *priming memory*. A key reason for viewing all these as different is the observation that different types of brain damage can affect some memory types but not others. A major step in refining the recommendation architecture understanding of the brain was being able to understand these differences in terms of differences between the information processes supporting the different memory types at the physiological level.

These insights were pulled together in a second book, *A System Architecture Approach to the Brain: from Neurons to Consciousness*, published in 2005. This book became the basis for a course offered to advanced undergraduate and graduate students at ANU. Teaching the course resulted in further clarification of the architectural ideas, especially the links to anatomy and physiology, and the mechanisms supporting cognitive tasks. These clarifications resulted in a third book, which greatly strengthened the detailed links between cognition, anatomy, physiology, and neurochemistry. This book, *Towards a Theoretical Neuroscience: from Cell Chemistry to Cognition* was published by Springer in 2013 and became the new textbook for the course.

In parallel with the books, I have published a number of academic papers. However, a major problem in gaining acceptance for these ideas has been that few neuroscientists and psychologists or even engineers working in industry have had experience with all the levels on which design of an extremely complex real-time system takes place. Many scientists implement software and even design electronic hardware to support their research, but the design process for these systems takes place within an already created system design framework and never has to come to grips with the full problem from transistor design to complex real-time system design. To put it another way, the five thousand man-years of effort required within BNR/Nortel for the first real-time electronic telecommunications systems included not just the design itself, but creation of all the design support systems. This effort is several hundred times greater than even the largest design project that is implemented in academia. Even in the commercial world, the design support systems have been well established, and it is rarely necessary to grapple with the issues of top-to-bottom system design from features to transistors. Furthermore, the costs of the hardware to perform a given computing problem have declined by many orders of magnitude since the 1970s. As discussed later in the book, resource constraints exert considerable pressures on system design and have also greatly affected the architecture of the brain. However, the effect of these pressures on electronic system architectures has become less familiar with the decrease in resource costs.

A couple of symptoms of this lack of appreciation for the implications of complex system design include attempts to perform huge simulations of physiologically realistic computer models of the brain and the use of formal mathematics, like differential equations, to try to understand brain functioning.

Some computer models invest large computing power in the simulation of anatomical structures, like cortical columns, with as much physiological realism as possible. The issue with this approach can be appreciated from the following somewhat caricatured example. Suppose you could create software models for neurons that precisely reproduced all their physiological processes. Then suppose that you could study one human brain and identify all the neurons and all the connections between them. You pull this together to create a software model for the complete brain. When you run this model, you find that it exhibits higher cognitive behaviours. The problem is that all you now have is another extremely complex system you don't understand!

Formal mathematics has played a critical role in creating the understanding of the physical world embodied in the hard sciences like physics and chemistry. These sciences view physical systems as being made up of huge numbers of relatively simple units, such as particles. When the behaviour of a system changes over time, differential equations are the key tools used. Examples include Schrödinger's equation in quantum mechanics, the Geodesic equations of general relativity, or the Navier-Stokes equations for the flow of fluids. Mathematical modelling is very effective for handling situations in which there are very small numbers of units, or large numbers of identical units. In practice this means that a physical system can only be modelled in one of two different ways. One way is that a very small number of particles is modelled. The other way is that if a system is made up of a large number of particles, all the particles must be viewed as identical, or perhaps there can be large numbers of just a very small number of different types. The problem in a complex control system is that there are large numbers of units (transistors, neurons), but these units can rarely be usefully regarded as identical.

In a computer system, there may be billions of transistors. In physical form many of these transistors are similar. However, each transistor is connected to a specific small set of other transistors, which in turn are connected to other specific sets of transistors. In general, any one transistor therefore has a unique connectivity environment within the system as a whole. The behaviour of the system in most cases depends on this detailed system connectivity. Hence transistors cannot be regarded as identical for the purposes of describing or understanding the features of the system, and differential equations are almost never used in system design. The only exceptions are when calculating global physical system parameters like the heat generated and flowing within the system or the electrical noise produced by the system. For such parameters, viewing every transistor as the same is a useful approximation, and mathematical modelling can be used.

In the brain there are billions of neurons. There are physical similarities between some neurons, but once again each neuron has a unique connectivity environment within the brain. Performance of cognitive tasks depends on this connectivity. Hence formal mathematical modelling will not be useful for understanding cognition, although it may have value for modelling global parameters like the overall electrical activity (such as EEG measurements) when neurons can be regarded as more or less identical.

As discussed in chapter 2, the only possible approach to understanding systems with this complexity involves a much more sophisticated and controlled use of approximation when describing the system. When a computer system performs some task, like accessing a web page, millions or even billions of transistors may contribute at different stages. It is not possible for a human brain to simultaneously imagine the ongoing activities of all those transistors. However, that activity precisely determines whether the task will be successfully completed or not. Hence that activity must ultimately be specified precisely by a designer. Computer science has therefore developed ways to describe such activity so that it is within the comprehension of a human designer, by creating hierarchies of description. High-level descriptions of system features are approximate, but to make design possible there are also ways to make small parts of such a description more precise, with the more precise but partial descriptions still within the comprehension of a designer. Small parts of this more precise description can be made even more precise, and so on. Such hierarchies of descriptions are needed to link transistor activities to the performance of system features.

The instruction and data read/write information models are critical to the construction of these hierarchies. At every level, descriptions are constructed from information processes of these two types. High-level instructions are made up of sequences and combinations of more detailed instructions, and high-level data elements that are read or written are made up of combinations of more detailed data elements. This consistency makes it possible to move easily between descriptions on different levels of detail.

When a brain performs some cognitive task, millions or even billions of neurons may contribute, and their activity determines whether the task is completed successfully or not. Human understanding therefore depends on the creation of analogous hierarchies of description. In the case of electronic systems these hierarchies have been imposed by human designers. There are no such designers in the case of the brain. However, natural selection pressures favouring brains that make economical use of resources and that can learn without damaging prior learning have resulted in the organization of brain resources in the recommendation architecture form. This form makes it possible to create such hierarchies of description.

It is again the existence of two information models that make hierarchies of description useful for the brain. However, as mentioned earlier, in the case of the brain the two models are recommendation definition/integration and condition definition/detection. In an electronic system, information detected in the environment is compared with data recorded by the designer, and if there is a match the instruction specified by the designer is executed. In the brain, information detected in the environment is compared with conditions defined by the brain from previous experience. In some cases, the comparison results in changes to the definition of a condition. If there is a match between the information derived from the environment and a condition, the condition is detected. Such a detection recommends a range of behaviours. To be carried out, a behaviour must have the largest total recommendation strength across all the currently detected conditions.

The activity of the brain can be described at high level in terms of major condition definitions/detections and recommendations. Each high-level condition detection is made up of large numbers of more detailed condition detections, and the total recommendation weight in favour of each recommended behaviour must be determined. A high-level condition is made up of many more detailed conditions, and a high-level behavioural recommendation is made up of many detailed recommendations. This consistency makes it possible to move easily between descriptions of brain activity on different levels of detail.

The techniques for organizing information about complex electronic systems have now been available for over fifty years. A puzzling question is why these techniques have not been applied more generally to understanding the brain. Part of the answer lies with how career rewards in academia are achieved. Numbers of published papers are a key measure of success. In the neurosciences this tends to result in early career researchers finding a physiological process that is somewhat different from already studied processes and performing many experiments on that process. Ideally, the choice of process is supported by an already established professor, making it easier to get research grants. Lots of papers can be published, each varying the parameters within which the process operates in a different way. Correlations between the process and some easily controlled behavioural experiments may be found. Although this approach is valuable (for example, it can result in identification of possible treatments for medical conditions), the understanding of how everything fits together in the brain is not rewarded. Soon after my first book was published, I visited a couple of neuroscientists to see if there was any interest in testing my system ideas. After spending several hours explaining my approach, I vividly remember one of them commenting, "Your ideas are fascinating, but if we work on them we will never get tenure."

On the computer science side, the design of learning systems uses a technology called artificial neural networks. Researchers early on encountered the problem that if such a system learned one type of task, subsequent learning of a different type generally eliminated the ability to perform the first type. As a result, artificial neural networks have focussed on separate systems, each learning one type of task. These networks have become very effective at learning groups of similar tasks, such as recognizing faces, recognizing tumours in medical images, or playing a particular game like Chess or Go. However, little progress has been made with designing artificial general intelligence (AGI) systems to learn a wide range of unrelated groups of tasks. The problem with AGI is, firstly, that artificial neural networks are not designed within the recommendation architecture framework, and secondly, that an AGI system within that framework would require a very large design effort. There has so far been no economic justification for investing in such an effort.

This book aims to describe how to understand higher cognition in terms of anatomy and physiology at a more accessible level than my earlier books. Such understanding is satisfying in its own right, and in addition it is a critical basis for understanding how human beings interact with their environment and with society. However, this understanding requires a paradigm shift from the way the vast majority of scientists think about the brain, from the conventional mathematical model-based approach to an approach based on how we understand complex electronic systems.

On a final note, there is a huge amount of experimental work on the brain that supports the understanding of cognition in terms of anatomy and physiology. In a book intended to be accessible to a more general audience, references to this work would be inappropriate. Extensive references to the original literature are provided in my more academic work, *Towards a Theoretical Neuroscience: from Cell Chemistry to Cognition*.

INDEX