Available online at www.sciencedirect.com

ScienceDirect

ELSEVIER

Current Opinion in Neurobiology

What must a global theory of cortex explain?

Leslie G Valiant

At present there is no generally accepted theory of how cognitive phenomena arise from computations in cortex. Further, there is no consensus on how the search for one should be refocussed so as to make it more fruitful. In this short piece we observe that research in computer science over the last several decades has shown that significant computational phenomena need to circumvent significant inherent quantitative impediments, such as of computational complexity. We argue that computational neuroscience has to be informed by the same quantitative concerns for it to succeed. It is conceivable that the brain is the one computation that does not need to circumvent any such obstacles, but if that were the case then quantitatively plausible theories of cortex would now surely abound and be driving experimental investigations.

Address
School of Engineering and Applied Sciences, Harvard University, Cambridge, MA 02138, USA

Corresponding author: Valiant, Leslie G (valiant@seas.harvard.edu)

Current Opinion in Neurobiology 2014, **25**:15–19

This review comes from a themed issue on **Theoretical and computational neuroscience**

Edited by **Adrienne Fairhall** and **Haim Sompolinsky**

For a complete overview see the Issue and the Editorial

Available online 23rd November 2013

http://dx.doi.org/10.1016/j.conb.2013.10.006

Introduction

That computing is the right framework for understanding the brain became clear to many soon after the discovery of universal computing by Turing [1], who was himself motivated by the question of understanding the scope of human mental activity. McCulloch and Pitts [2] made a first attempt to formalize neural computation, pointing out that their networks were of equivalent expressive power to Turing machines. By the 1950s it was widely recognized that any science of cognition would have to be based on computation.

It would probably come as a shock to the earliest pioneers, were they to return today, that more progress has not been made towards a generally agreed computational theory of cortex. They may have expected, short of such a generally agreed theory, that today there would at least exist a variety of viable competing theories. Understanding cortex is surely among the most important questions ever posed by science. Astonishingly, the question of proposing general theories of cortex and subjecting them to experimental examination is currently not even a mainstream scientific activity.

It is not that the computational perspective was ever abandoned. It was well articulated by David Marr [3], who split the problem into three levels: '*Computational theory*: What is the goal of the computation, why is it appropriate, and what is the logic of the strategy by which it can be carried out? *Representation and algorithm*: How can this computational theory be implemented? In particular, what is the representation for the input and output, and what is the algorithm for the transformation? *Hardware implementation*: How can the representation and algorithm be realized physically?' This widely quoted passage is, of course, very general and could pass as a mission statement for computer science itself.

Our review here is informed by the observation that since Marr's time computer science has made very substantial progress in certain quantitative directions. The following four phenomena are clearly critical for the brain: communication, computation, learning and evolution. Over the last few decades all four have been subject to quantitative analysis, and are now known to be subject to *hard quantitative constraints* (see [4] for a general exposition). First there is the obvious cost of communication: if we desire to be able to communicate *any* n-bit message we will need to be able to send n bits. Second there is computational complexity: if we have some information and can define what processing we wish done on it, that processing may have an unaffordable cost in terms of operations even if we have at hand all the information and can precisely define the desired processing. A third level is learning — even if the desired processing can be achieved by an efficient computation, acquiring a program for it from examples or other behavior presents further impediments. Fourth, if we wish to acquire this program by Darwinian evolution then we encounter even more obstacles.

We do not believe that there can be any doubt that the theory sought has to be computational in the general sense of Turing. The question that arises is: In what way does Marr's articulation of the computational approach fall short? Our answer is that, exactly as in any other domains of computation, a successful theory will have to show additionally, *how the quantitative challenges that need to be faced are solved in cortex*. If these challenges were nonexistent or insignificant then plausible theories would now abound and the only task remaining for us would be to establish which one nature is using.

An augmented computational framework

If, as we believe, cortex is addressing this quartet (computation, learning, evolution and communication) with subtlety, then two additional requirements need to be added to those of Marr for any successful theory. First, *it has to incorporate some understanding of the quantitative constraints that are faced by cortex*. Second, as in other domains of computing, this quantitative understanding has to be articulated in terms of *models of computation appropriate to the problems at hand and the chosen levels of analysis*.

This augmented set of requirements is quite complex in that many issues have to be faced simultaneously. We suggest the following as a streamlined working formulation for the present:

(i) Specify a candidate set of quantitatively challenging cognitive tasks that cortex may be using as the primitives from which it builds cognition. At a minimum, this set has to include the task of memorization, and some additional tasks that *use* the memories created. The task set needs to encompass both the learning and the execution of the capabilities in question.
(ii) Explain how, on a model of computation that faithfully reflects the quantitative resources that cortex has available, instances of these tasks can be realized by explicit algorithms.
(iii) Provide some plausible experimental approach to confirming or falsifying the theory as it applies to cortex.
(iv) Explain how there may be an evolutionary path to the brain having acquired these capabilities.

To illustrate that this complex of requirements can be pursued systematically together we shall briefly describe the framework developed for this by the author [5]. It targets a particular class of tasks called *random access tasks*, to be executed on the *neuroidal* model of computation, using a *positive representation* and a particular style of algorithms called *vicinal algorithms*. Other researchers who have sought to understand cortex have generally not placed quantitative computational goals center stage. We shall make references to some recent examples [6•,7•,8•,9•] in order to contrast some of the currently pursued alternatives.

Positive representations

In order to specify computational tasks in terms of input-output behavior one needs to start with a representation for each task. It is necessary to ensure that for any pair of tasks where the input of one is the output of the other there is a common representation at that interface. Here we shall take the convenient course of having a *common representation* for all the tasks that will be considered, so that their composability will follow.

In a *positive representation* [5] a real world *item* (a concept, event, individual, etc.) is represented by a set S of r neurons. A concept being processed corresponds to the members of S firing in a distinct way. More precisely, as elaborated further in [10], if more than a fraction b (e.g. 88%) of S fire then the concept is definitely being processed, if fewer than fraction a (say 30%) then the concept is not being processed, and the system is so configured that the intermediate situation almost never occurs. We note that for any computational theory with specific algorithms one needs some definition of representation as specific as this.

Positive representations come in two varieties, *disjoint*, which means that the S's of distinct concepts are disjoint, and *shared*, which means that the S's can share neurons. Disjointness makes computation easier but requires small r (such as $r = 50$) if large numbers of concepts are to be represented. The shared representation allows for more concepts to be represented (especially necessary if r is very large, such as several percent of the total number of neurons) but can be expected to make computation, without interference among the task instances, more challenging.

Random access versus local tasks

We believe that cortex is communication bounded in the sense that: (i) each neuron is connected to a minute fraction of all the other neurons, (ii) each individual synapse typically has weak influence, in that a presynaptic action potential will make only a small contribution to the threshold potential needed to be overcome in the post-synaptic cell, and (iii) there is no global addressing mechanism as computers have. We call tasks that potentially require communication between arbitrary memorized concepts *random-access tasks*. Such tasks, for example, an association between an arbitrary pair of concepts, are the most demanding in communication and therefore quantitatively the most challenging for the brain to realize. The arbitrary knowledge structures in the world will have to be mapped, by the execution of a sequence of random access tasks that only change synaptic weights, to the available connections among the neurons that are largely fixed at birth.

We distinguish between two categories of tasks.

Tasks from the first category assign neurons to a new item. We have just one task of this type, which we call *Hierarchical Memorization* and define it as follows: For any stored items A, B, allocate neurons to new item C and make appropriate changes in the circuit so that in future A and B active will cause C to be active also.

The second category of tasks make modifications to the circuits so as to relate in a new way items to which neurons have been already assigned. We consider the following three. *Association*: For any stored items A, B, change the