Brains as computers: metaphor, analogy, theory or fact?

Romain Brette

Sorbonne Université, INSERM, CNRS, Institut de la Vision, 17 rue Moreau, F-75012 Paris, France

romain.brette@inserm.fr

Abstract

Whether electronic, analog or quantum, a computer is a programmable machine. Wilder Penfield held that the brain is literally a computer, because he was a dualist: the mind programs the brain. If this type of dualism is rejected, then identifying the brain to a computer requires defining what a brain "program" might mean and who gets to "program" the brain. If the brain "programs" itself when it learns, then this is a metaphor. If evolution "programs" the brain, then this is a metaphor. Indeed, in the neuroscience literature, the brain-computer is typically not used as an analogy, i.e., as an explicit comparison, but metaphorically, by importing terms from the field of computers into neuroscientific discourse: we assert that brains compute the location of sounds, we wonder how perceptual algorithms are implemented in the brain. Considerable difficulties arise when attempting to give a precise biological description of these terms, which is the sign that we are indeed dealing with a metaphor. Metaphors can be both useful and misleading. The appeal of the brain-computer metaphor is that it promises to bridge physiological and mental domains. But it is misleading because the basis of this promise is that computer terms are themselves imported from the mental domain (calculation, memory, information). In other words, the brain-computer metaphor offers a reductionist view of cognition (all cognition is calculation) rather than a naturalistic theory of cognition, hidden behind a metaphoric blanket.

Keywords: brain-computer metaphor; algorithms; programs; philosophy; metaphors.

What is a computer?

It is common to assert that the brain is a sort of computer. It goes without saying that no one believes that people have a hard drive and USB ports. More broadly, a computer is a machine that can be programmed. Computers can be programmed in many different ways: procedural programming (a series of elementary steps, as in a recipe or the C language), logic programming (using logical propositions as in the language Prolog), and so on. There can be such things as "non-conventional" computers, parallel computers, analog computers, quantum computers, and so on, which execute programs in different ways.

"Programmable machine" is both the common usage and the technical usage of "computer". Computer science offers no formal definition of computer: it is the concept of program that unifies much of theoretical computer science. In computability theory, a function f is said to be computable if there exists a program that can output $f(x)$ given x as an input. In computability theory, an undecidable problem is a decision problem for which no program gives a correct answer, such as the halting problem. Complexity theory examines the number of steps that a program takes before it stops, and classifies problems with respect to how this number scales with input size. Kolmogorov complexity is the size of the shortest program that produces a given object.

Richards and Lillicrap (2022) rightfully recommend to clarify the exact definition of computer we use, and they offer "some physical machinery that can in theory compute any computable function". Unfortunately, this definition hides the notion of a programmable machine behind the vagueness of the phrase "can in theory". What does it mean that an object *can* do certain things?

Consider a large (say, infinite) pile of electronic components. For any computable function, one "can in theory" assemble the elements into a circuit that computes that function. But this does not make the pile of components a computer. To make it a computer, one would need to add some machinery to build a particular circuit from instructions given by the user. Certainly, the electronic elements "can in theory" compute any computable function, but in the context of computers, what is meant by "can" is that the computer *will* compute the function *if* it is given the adequate instructions, in other words it is a programmable machine.

In the same way, the fact that any logical function can be decomposed into the operations of binary neuron models (McCulloch and Pitts, 1943) does not make the brain a computer, because the brain is not a machine to assemble neurons according to some instructions, as if neurons were construction blocks. Thus, it is fallacious to assert that the brain is literally a computer on the mere basis that formal neural networks can approximate any function (Richards and Lillicrap, 2022), for this would attribute computerness to a disorganized pile of electronic components or to any large enough group of atoms, and this is neither the common usage nor the technical usage in computer science.

A dualistic entity

As pointed out by Anthony Bell (Bell, 1999), the computer is a fundamentally dualistic entity, where some machinery ("hardware") executes instructions ("software") defined by an external agent. It is exactly in this sense that Wilder Penfield, who discovered the cortical homunculi (sensory and motor "maps" of the body on the cortex), claimed that the brain is literally a computer (Penfield, 1975). Penfield was a dualist: he considered that the brain is literally a computer, which gets programmed by the mind.

Although modern neuroscience is deeply influenced by Cartesian dualism, most neuroscientists do not embrace this type of dualism (Brette, 2019; Cisek, 1999; Mudrik and Maoz, 2015). Therefore, it is generally not believed that the brain gets *literally* programmed by some other entity. Perhaps the brain-computer is "programmed by evolution" or "self-programmed", but these are rather vague metaphorical uses. To give some substance to the statement "the brain is a computer", one needs to identify programs in the brain, and a way in which these programs can be changed arbitrarily.

For example, classical connectionism might propose that the program is the set of synaptic weights, and that some process may change these weights. This view, as any attempt to identify a program in the brain, assumes that the brain can be separated into a set of modifiable elements (software) and a fixed set of processes (hardware) that act on those elements, for otherwise the "program" would not unambiguously specify what it does, i.e., would not be a program at all. But synaptic weights are certainly not the only modifiable elements in the brain. This hardware/software distinction is precisely what Bell (1999) opposed because everything in the brain, or in a biological organism, is "soft": "*a computer is an intrinsically dualistic entity, with its physical set-up designed not to interfere with its logical set-up, which executes the computation. In empirical investigation, we find that the brain is not a dualistic entity*". A living organism does not simply adjust molecular knobs: it continuously produces its own structure, synapses and everything else (Kauffman, 1986; Montévil and Mossio, 2015; Rosen, 2005; Varela et al., 1974).

Furthermore, to make the case that the brain is a computer, one must demonstrate that there is a way in which the brain's programs can be changed arbitrarily. The problem with this claim is that it implies some form of agency. If not a distinct mind, then who decides to change the program? One might say that the brain is programmed by evolution to achieve some goals, but unless we believe in intelligent design, we know that evolution is not literally a case of programming but rather the natural selection of random structural changes. One might say that the brain "programs itself", but it is not straightforward to give substance to this claim either, beyond the trivial fact that the structure of the brain is plastic. If this plasticity follows some particular rules, then the "programs" that the brain produces are in fact not arbitrary. And indeed, it is not the case that a cat can "self-program" itself into playing chess. Perhaps it might "in theory" be able to play chess, that is, if we allow some fictional observer to rewire the cat's brain in certain ways, but this is not a case self-programming. In the idea that the cat's brain is a computer, there appears to be a confusion of Umwelts (Gomez-Marin, 2019): an observer might be able to "program" a cat's brain in some sense, but the cat itself cannot.

Theory, analogy or metaphor?

Therefore, it is not a fact that brains are computers. It might be a certain type of dualist theory, or a fundamentalist connectionist theory, but those theories are at odds with what we know about the biology of brains. However, in most cases, the statement is not taken literally in the neuroscience literature. Is it an analogy or a metaphor? The distinction is that an analogy is explicit while a metaphor is implicit. It might be occasionally stated that the brain is *like* a computer, but a much more common case in the neuroscience literature is that one speaks of sensory *computation*, *algorithms* of decision-making, *hardware* and *software*, *reading* and *writing* the brain (for measuring and stimulating), biological *implementation*, neural *codes*, and so on. These are clear cases of metaphorical writing, borrowing from the lexical field of computers without explicitly comparing the brain to a computer.

Metaphors can be powerful intellectual tools because they transport familiar concepts to an unfamiliar setting, and they have shaped the history of neuroscience (Cobb, 2020). The linguists Lakoff and Johnson (1980) have shown that metaphors pervade our language and shape the concepts with which we think, even though we usually do not notice it ("to shape" in this sentence and "to transport" in the previous one, both applied to concepts). As the authors emphasized: "*What metaphor does is limit what we notice, highlight what we do see, and provide part of the inferential structure that we reason with*". It is this inferential structure that deserves closer attention. The brain-computer metaphor might be a "semantic debate" (Richards and Lillicrap, 2022), but meaning is actually important. What do we mean when we say that the brain implements algorithms, and is it true?

A double metaphor

Before we discuss algorithms in the brain, it is useful to reflect on why the brain-computer metaphor is appealing. The brain-computer metaphor seems to offer a natural way to bridge mental and physiological domains. But it is important to realize that it does so precisely because computer words are themselves mental metaphors. In the 17th century, a "computer" was a person who did calculations. Later on, by analogy, devices built to perform calculations were called computers. We say for example that computers have "memory", but memory is a cognitive ability possessed by persons: it is people who remember, and then we metaphorically say that a computer "memorizes" some information; but when you open some text file, the computer does not literally remember what you wrote. This is why Wittgensteinian philosophers point out that "*taking the brain to be a computer [...] is doubly mistaken*" (Smit and Hacker, 2014).

No wonder computers offer a natural way to describe how the brain "implements" cognition: computers were designed with human cognition in mind in the first place. For this reason, there is a sense in which certain persons (but not brains, cats or young children) might literally and trivially be computers: an educated person can execute a series of instructions, for example the integer multiplication algorithm. This trivial sense exists precisely because the computer is modeled on a subset of human cognitive abilities, namely doing calculations. But of course, the relevant scientific question is whether all cognitive activity is of this kind, that is, is a sort of unconscious calculation. In other words, the brain-computer metaphor is a reductionist view of cognition, which claims that all cognitive activity in all animal kingdom (perception, decision, motor control, etc.) is actually composed of elementary cognitive steps, these steps being those displayed by educated humans when they calculate.

At the very least, this claim is not trivially true.

Algorithms of the brain

What do we mean when we say that the brain implements algorithms? The textbook definition of algorithm in computer science is: "*a sequence of computational steps that transform the input into the output*" (Cormen et al., 2009). There are different ways to define those steps, but it must be a procedure that is reducible to a finite set of elementary operations applied in a certain order.

What is *not* algorithmic is, for example, the solar system. The motion of planets follows some laws, but it cannot be decomposed into a finite set of operations. These laws constitute a *model* of planet motion, not an algorithm. In the same way, a feedback control system is not in general an algorithm (see e.g. van Gelder's example of Watt's centrifugal governor (van Gelder, 1995)). Of course, some algorithms can be feedback control systems, but the converse is not true.

In the same way, a model of brain function is not necessarily an algorithm. Of course, some are. For example, networks of formal binary neurons (McCulloch and Pitts, 1943) are algorithmic. Each "neuron" is defined as a binary function and a feedforward network transforms an input into an output by a composition of such functions. The same applies to deep learning models. Backpropagation is an algorithm too. But the Hodgkin-Huxley model (Hodgkin and Huxley, 1952) is not an algorithm. It is, as the name implies, a model: laws that a number of physical variables obey.

Of course, the Hodgkin-Huxley model can be *simulated* by an algorithm. But the membrane potential is not in reality changed by a sequence of Runge-Kutta steps. More generally, the fact that a relationship between two measurable variables is computable does not imply that the physical system actually implements an algorithm to map one variable to the other. It only means that *someone* can implement the mapping with an algorithm.

Biophysical models of the brain are typically dynamical systems. But dynamical systems are not generically algorithms, and therefore asserting that the brain runs algorithms is a particular commitment that deserves proper justification. To justify it, one needs to identify elementary operations in the brain. For example, the computational view of mind holds that cognition is the manipulation of symbols, that is, the elementary operations are symbolic operations. This leaves the issue of identifying symbols in the brain, which is generally done through the concept of "neural codes", but this concept is problematic both theoretically and empirically (Brette, 2019). Among other examples, Minsky (1988) attempted to describe cognition in terms of elementary cognitive operations, and Marr (1982) tried to describe vision as a sequence of well-identified signal processing operations, with limited success (Warren, 2012). More generally, it is not so obvious that behavior can be entirely captured by algorithms (Roli et al., 2022).

The word "algorithm" is sometimes used in a broader sense, to mean some kind of detailed quantitative description of brain function. But this metaphorical use is confusing: not everything lawful in the world is algorithmic. A quantitative description is a model, not an algorithm, and there are many kinds of model.

Computation in the brain

Perhaps a less misleading term is "computation". The brain might not be a computer, because it is not literally programmable, and it might not literally run algorithms, but it certainly computes: for example, it can transform sound waves captured at the ears into the spatial position of a sound source. But what do we mean by that exactly?

If what we mean is that we are able to locate sounds, look at their expected position and generally behave as a function of source position, then should we not just say that we can perceive the position of sound sources? The word "computation" certainly suggests something more than that. But if so, then this is not a trivial statement and it requires proper justification. Perhaps what is meant is that perception is the result of a series of small operations, that is, by an algorithm, but this is far from obvious.

Perhaps we mean something broader: the brain transforms the acoustic signals into some neural activity that can be identified to source position, and that then leads to appropriate behavior and percepts. But this assumes some form of separability between an encoding and a decoding brain, which can be questioned (Brette, 2019). Or perhaps "computation" is simply meant to designate a transformation from sensory signals to some mental entity that represents source position. The difference between a computation and a mere transformation is then the fact that the output is a representation, not just a value. But then we need to explain what "representation" means in this context, for example that a representation has a truth value (it is correct or not), and how representations relate to brain activity.

Thus, it is not at all obvious in what sense the brain "computes", if it does, and the metaphorical use of the word tends to bury the important questions.

Conclusion

Computers are programmable machines. Let us leave aside the concept of a "machine", which would deserve specific treatment (see e.g. (Bongard and Levin, 2021; Nicholson, 2019)), and allow for an even broader definition: a computer is a programmable thing. Brains are not programmable things - at least not literally.

Except in rare Cartesian views where the mind is seen to program the brain (Penfield, 1975), the brain-computer metaphor is indeed a metaphor. Explicit formal comparisons with computers are rare, but brain processes are often described using words borrowed from the lexical field of computers (algorithms, computation, hardware, software, and so on). It is in fact a double metaphor, because computers are themselves metaphorically described with mental terms (e.g. they memorize information). This circular metaphorical relationship explains why the metaphor is (misleadingly) appealing.

The brain-computer metaphor is a source of much confusion in the literature. "Computer" might be used metaphorically to mean something complicated and useful. But computers run programs: what programs are we referring to? Evolution? The connectome? Neither is actually a program, and it is misleading to suggest they are. "Algorithm" might be used metaphorically to mean "laws" or "model". But this is misleading: "algorithm" suggests elementary operations and codes, which are not found in all models, and certainly not obviously found in brains (Brette, 2019). "Computation" is used metaphorically, but what is meant exactly is generally undisclosed: is it a claim about the algorithmic nature of cognition? about representations? or simply about the fact that behavior is adequate?

Once the meanings of these computer terms are properly disclosed, the scientific debate might begin.

Acknowledgments

This work was supported by Agence Nationale de la Recherche (ANR-20-CE30-0025-01 and ANR-21-CE16-0013-02), Programme Investissements d'Avenir IHU FOReSIGHT (Grant ANR-18-IAHU-01), and Fondation Pour l'Audition (Grant FPA RD-2017-2).

References

Bell AJ. 1999. Levels and loops: the future of artificial intelligence and neuroscience. *Philos Trans R Soc B Biol Sci* **354**:2013–2020.

Bongard J, Levin M. 2021. Living Things Are Not (20th Century) Machines: Updating Mechanism Metaphors in Light of the Modern Science of Machine Behavior. *Front Ecol Evol* **9**.

Brette R. 2019. Is coding a relevant metaphor for the brain? *Behav Brain Sci* 1–44. doi:10.1017/S0140525X19000049

Cisek P. 1999. Beyond the computer metaphor: behaviour as interaction. *J Conscious Stud* **6**:125–142.

Cobb M. 2020. The Idea of the Brain: The Past and Future of Neuroscience. New York: Basic Books.

Cormen TH, Leiserson CE, Rivest RL, Stein C. 2009. Introduction to Algorithms, third edition, third edition. ed. Cambridge, Mass: The MIT Press.

Gomez-Marin A. 2019. A clash of Umwelts: Anthropomorphism in behavioral neuroscience. *Behav Brain Sci* **42**. doi:10.1017/S0140525X19001237

Hodgkin AL, Huxley AF. 1952. A quantitative description of membrane current and its application to conduction and excitation in nerve. *J Physiol* **117**:500–544.

Kauffman SA. 1986. Autocatalytic sets of proteins. *J Theor Biol* **119**:1–24. doi:10.1016/S0022-5193(86)80047-9

Lakoff G, Johnson M. 1980. Metaphors We Live By. University of Chicago Press.

Marr D. 1982. Vision, First Edition. ed. W.H.Freeman & Co Ltd.

McCulloch WS, Pitts W. 1943. A logical calculus of the ideas immanent in nervous activity. *Bull Math Biophys* **5**:115–133. doi:10.1007/BF02478259

Minsky M. 1988. Society Of Mind. New York: Simon & Schuster.

Montévil M, Mossio M. 2015. Biological organisation as closure of constraints. *J Theor Biol* **372**:179–191. doi:10.1016/j.jtbi.2015.02.029

Mudrik L, Maoz U. 2015. "Me & my brain": exposing neuroscience's closet dualism. *J Cogn Neurosci* **27**:211–221. doi:10.1162/jocn_a_00723

Nicholson DJ. 2019. Is the cell really a machine? *J Theor Biol* **477**:108–126. doi:10.1016/j.jtbi.2019.06.002

Penfield W. 1975. The Mystery of the Mind. Princeton University Press.

Richards BA, Lillicrap TP. 2022. The Brain-Computer Metaphor Debate Is Useless: A Matter of Semantics. *Front Comput Sci* **4**.

Roli A, Jaeger J, Kauffman SA. 2022. How Organisms Come to Know the World: Fundamental Limits on Artificial General Intelligence. *Front Ecol Evol* **9**.

Rosen R. 2005. Life Itself – A Comprehensive Inquiry into the Nature, Origin, and Fabrication of Life, New e. édition. ed. New York: Columbia University Press.

Smit H, Hacker PMS. 2014. Seven Misconceptions About the Mereological Fallacy: A Compilation for the Perplexed. *Erkenntnis* **79**:1077–1097. doi:10.1007/s10670-013-9594-5

van Gelder T. 1995. What Might Cognition Be, If Not Computation? *J Philos* **92**:345–381. doi:10.2307/2941061

Varela FG, Maturana HR, Uribe R. 1974. Autopoiesis: The organization of living systems, its characterization and a model. *Biosystems* **5**:187–196. doi:10.1016/0303-2647(74)90031-8

Warren WH. 2012. Does This Computational Theory Solve the Right Problem? Marr, Gibson, and the Goal of Vision. *Perception* **41**:1053–1060. doi:10.1068/p7327