APA Newsletter on Philosophy and Computers 19 (1), 18-21 (2019)
https://cdn.ymaws.com/www.apaonline.org/resource/collection/EADE8D52-8D02-4136-9A2A-729368501E43/ComputersV19n1.pdf

Should Physical Computation be Understood Mechanistically?

John Symons

Commonsense tells us that a rock is not a computer but that your laptop is. While commonsense is untroubled by the straightforward cases, it is of little help with the exceptions and the exotic cases: Does a broken or faulty laptop count as a computer? And what if aliens used devices that looked like rocks to compute in ways we are unable to understand? Is it sometimes correct to think of rivers, forests, or strands of DNA as computers? Is the central nervous system of an animal a computer? Some people like to think of brains and minds as computers, but isn't this an odd claim given how unlike ordinary computers biological systems seem to be? We could begin to answer such questions in a principled way if we had a plausible theoretical account of the kinds of things that are physical computers. Currently we don't have such an account.

Our uncertainty with respect to the nature of physical computers contrasts sharply with our excellent understanding of the mathematics of computation; we have a well-articulated theory of computable functions, and we have a good mathematical model of systems that compute. The formalism of computability is elegant, clear, and relatively easy to understand. We say that a system that computes is a finite state automaton. A finite state automaton is a *mathematical object* that can be in one state at a time and that switches between states in a rule-governed way. We can reason clearly about the properties of such mathematical objects and have discovered some of their limits. Where things become less clear is when we ask what it means to say that some *physical object* computes.

One response is to simply dismiss questions about physical computation as being nothing more than a matter of interpretation. Arguably, one can construct an interpretation whereby *any* physical object or any arbitrary mereological sum of objects (your thumb, a lake, Jupiter, or the pile of crumbs on the counter) can be interpreted as being *any* finite-state automaton. Hilary Putnam defended a position roughly along these lines (See especially Putnam 1988). However, as a response to the problem of distinguishing physical computers from non-computers this strategy is philosophically unsatisfying and scientifically unhelpful (Buechner 2008).

Stating an adequate criterion for distinguishing physical computers from non-computers has proven difficult for philosophers. Clearly the problem is not solved simply in virtue of having a good mathematical theory of computability. More is needed. This is because a satisfactory account of physical computation, unlike a mathematical theory of computability should provide individuation conditions that distinguish objects and processes that compute from those that do not. In his recent book *Physical Computation: A Mechanistic Approach,* Gualtiero Piccinini defends a novel and appealing theory of physical computation. On his view, a physical computer is "a mechanism whose teleological function is to perform a physical computation and a physical computation is the manipulation of a medium-independent vehicle according to a rule" (2015, 10). In this paper, I will examine the strengths and limitations of this position concentrating on the question of whether the mechanistic approach has the resources to provide a satisfying account of the individuation of physical computers. Unfortunately, the concept of mechanism cannot provide an illuminating account of the metaphysics of physical computation. This is because the notion of mechanism is insufficiently fundamental and insufficiently general. Nevertheless, Piccinini's book provides an accurate map of the philosophical problems associated with the individuation of physical computers and is filled with important distinctions and insights. Piccinini offers the best attempt to date to answer these questions.

Piccinini shares a view of mechanism with the New Mechanists in philosophy of biology and philosophy of cognitive science. The New Mechanist approach began to take shape in the early 1990s with the work of Bill Bechtel and Bob Richardson, especially in their book *Discovering Complexity*. Today, the canonical reference for the New Mechanist position is Peter Machamer, Lindley Darden, and Carl Craver's "Thinking about Mechanisms". In his *Stanford Encyclopedia of Philosophy* entry 'Mechanism in Science', Carl Craver writes that all definitions of mechanism involve four characteristic features: (1) a phenomenon to be explained, (2) parts, (3) causings, and (4) organization (2015). According to Bechtel and Abrahamsen for example "[a] mechanism is a structure performing a function in virtue of its component parts, component operations, and their organization. The orchestrated functioning of the mechanism is responsible for one or more phenomena" (2005, 423).

Very broadly speaking, the New Mechanists provide an account of mechanism that they have gleaned from the manner in which some biologists and some neuroscientists give explanations. They notice that in many fields in the biological sciences explanations seem to work by showing how structured parts and processes are orchestrated so as to be responsible for the way things appear to happen. What is meant by orchestration and responsibility is left relatively loosely characterized. Nevertheless New Mechanists correctly point to the ubiquity and wide acceptance of such patterns of explanation in biology. On this view, the job of at least some biologists is to discover the mechanisms that produce some phenomena.

The New Mechanists are a group of pragmatically-inclined philosophers who are guided by what they see as the explanatory norms and practices of the scientific communities they know best. In this spirit, the New Mechanists sometimes seem to imply that their accounts capture a metaphysically agnostic core of ordinary explanatory practice in the sciences. While New Mechanists assume a modest posture in relation to scientific practice there is (or should be) more to New Mechanism than just a neutral way of parsing accepted styles of explanation in scientific communities. Mechanists regard the explanatory practices of scientists as indicating that there really are distinguishable parts of mechanisms with well-defined relations and activities or operations. In this sense, New Mechanism has non-trivial metaphysical commitments.

Piccinini puts this view of mechanism into action in his description of physical computers. On his view, physical computers can be understood in terms of their component parts, their *functions*, and their organization. Physical computers are objects whose function, according to Piccinini, is to perform physical computations. Physical computation is the manipulation of medium-independent vehicles according to rules. Rules and functions resist reduction to mechanism but on Piccinini's account the physical systems that follow those rules are ultimately nothing more than mechanisms.

In practice, physical computers are certainly picked out by reference to their functions and those functions are determined (at least in part) by reference to abstract rules. However, a description of what the physical computer *is* (as opposed to how we happen to identify it) need not include mention of those rules and functions. The role of rules and functions in his presentation is intended to serve as a way of distinguishing the subset of mechanisms that count as physical computers (namely those that serve the function of following rules) from those that do not. Thus, Piccinini does not claim that we rely exclusively on the notion of mechanism in the process of *identifying* some objects as physical computers. In fact, it might be a matter of epistemic necessity that our identification of physical computers depends on reference to rules and functions. Nevertheless, the way we happen to distinguish physical computers from other mechanisms is not ultimately relevant to what they are. Identification is not the same as individuation.

On Piccinini's account, mechanism plays a role in helping us to understand the metaphysics of physical computation. One important reason to give an account of physical computation in terms of mechanism is to sidestep Putnam's concerns about the challenge of determining a unique mapping from abstract computational characterizations to physical implementation. Putnam argued that every ordinary open physical system can be interpreted as implementing every finite-state automaton. This means that every physical object can justifiably be interpreted as, a computer (1988, pp. 120-125) but it would also render the mapping account useless as a means of individuating physical computers. Piccinini claims that the mapping approach poses the problem incorrectly and so he does not answer Putnam's challenge. His strategy is to abandon the question of how we should interpret the mapping between states of a physical system and states of an abstract computer. He suggests that even amended or strengthened mapping accounts will trivialize the claim that a physical system computes (2015 22). While Piccinini does not employ the distinction between individuation and identification in the manner discussed above, the mapping approach can be understood as addressing epistemic considerations involved in identifying physical computers whereas the correct approach would address the problem of individuating physical computers. In this sense, the mechanistic account can be understood as offering an alternative strategy for thinking about those individuation conditions. The activity of physical computation is a purely mechanistic matter on Piccinini's view.

This explanatory strategy allows him to claim that what it is to *be* a physical computer does not depend on representational or semantic concepts. If correct, this would mark an important step forward insofar as representation and semantics have been central to many previous accounts of computation and have presented deep conceptual difficulties familiar to the traditions of philosophy of mind and philosophy of language. Central to Piccinini's contention that the mechanistic approach can do without representation and semantics is his understanding of how physical computers perform *concrete computations*. "A physical system is a computing system" he writes "iff it's a system that performs concrete computations" (2015, 118). Concrete computations support the teleological function of the physical computer insofar as the rules characterizing the transformation of vehicles are fixed by the purpose of the machine. Nevertheless, that transformation itself can be described independently of abstract rules. Rather than individuating these vehicles by reference to their semantical roles, the non-semantic properties of strings of discrete states that figure in program-controlled computers can serve as the basis of the explanation of the operation of those systems. "Different bits and pieces of these strings of states have different effects on the machine" (2015, 45). Because of this, he argues, we can describe the operation of sub-strings and states without mentioning what their semantics are. Moreover, according to Piccinini the mechanistic account of vehicles can explain syntactical properties of digital computation. Since mechanism is a more basic notion than the syntax of a language, Piccinini can argue that a mechanistic approach has the additional advantage of providing an account of non-digital forms of computation whereas the syntactical approach fails to do so. The promise of the mechanistic approach is that only non-semantic, non-representational and even non-syntactic individuation conditions for the vehicles of computation will figure in mechanistic explanations.

The mechanistic approach seems more metaphysically basic than, for example, the semantic or representational level of analysis. However, mechanistic approaches have a hard time shedding light on the problem of individuation (Symons, 2010). To begin with, the metaphysical significance of mechanism is not clear. In part, this is because the mechanistic approach is primarily derived from reflections on scientific explanation as a practice rather than on those aspects of reality that ground the reliability of scientific explanations. It is obviously true that in many areas of biology explanations are given in mechanistic terms. But this is not an explanation of why mechanistic accounts are widely accepted in those areas or why they have been so successful. If anything, the success of mechanistic explanation in these areas itself demands explanation. Worth mentioning too is the need for an account of the notion of part, cause, and organization, that undergird the notion of mechanism itself.

A second reason for the difficulty of using mechanistic accounts to shed light on individuation and other metaphysical topics arises from the fact that mechanism is both an insufficiently general and an insufficiently fundamental notion. With respect to generality, notice, for example, that the mereological commitments of New Mechanism make it inapplicable to some kinds of objects and phenomena. The New Mechanists acknowledge that their view is only applicable to some regions of even more mundane domains of scientific explanation. Even in biology, as Skipper and Millstein 2005 point out, and as Craver and Tabery acknowledge (2017), the mechanistic notion of part is difficult to sustain both at the level of the very small in well-understood biochemical phenomena and at the level of the very large in natural selection. Our best understanding of nature does not support the idea that everything can be understood or explained in terms defended by the New Mechanists, therefore the mechanistic approach is unlikely to have the resources to answer general questions concerning individuation.

With respect to fundamentality, it is difficult to reconcile New Mechanism with basic physics. For example, it is difficult to understand field-theoretic explanations within a mechanistic framework. Furthermore, the ontological implications of quantum mechanics pose a challenge insofar as New Mechanism seems to rely on local causal interactions and the idea of isolable parts with definite properties.

In the case of physical computation, there is a sense in which such metaphysical concerns might seem irrelevant. Most physical computers are artifacts with very special characteristics. The purposes of most physical computers are dictated by the demands of the target markets of their manufacturers. Since the practice of manufacturers serves specific commercial or scientific purposes and since most computers are built from pre-fabricated parts according to a plan for organization, it is appropriate to characterize these physical computers as functional mechanisms.

The component parts of physical computers also have parts of course. However, the mereological ground floor for individuating physical computers (on Piccinini's view) is fixed by the logic of software (Horner 2019). At bottom, the primitive *computationally relevant* components for conventional computers are logic gates. In principle, any bistable system can serve as a primitive component for a computing technology if it allows us to reliably manipulate whether that system is in one or the other equilibrium state. There is nothing relevant to computational function *per se* at more fundamental levels than the level of such bistable systems.

The mechanistic approach to physical computation does not tell us why or how the primitive components of a conventional physical computer work as they do. As far as the mechanistic perspective is concerned, once one has reliably manipulable bistable systems to build on, everything underneath can be ignored. However, the promise of the mechanistic approach to physical computation had been the possibility of providing an explanation of the relevant primitive components: "The mechanistic explanation of a primitive computing component – say, an AND gate – explains how that component exhibits its specific input-output behavior. In our example, the components of a particular electrical circuit, their properties, and their configuration explains how it realizes an AND gate" (155) Unfortunately, even in relatively mundane engineering contexts, for example in the explanation of modern transistors and circuits, New Mechanist style explanations fall short. The New Mechanist approach will not get very far, for example, in the understanding of semi-conductors or in the understanding of field effect transistors more generally. The details of how fields behave are completely opaque if one's only explanatory resource is mechanism. Mechanisms are composed of objects or processes with definite properties. The quantum mechanical account of the behavior of fields does not assume local causal interactions of the mechanist kind nor does it assume objects with definite properties (See Kuhlman 2013 338). Mechanists might reject this concern given that once reliable bistable systems emerge at some level everything underneath can be ignored. However, the fundamental physical nature of these bistable systems turns out to be relevant to their operation and must be taken into account in engineering contexts. Consider, for instance, the consequences of miniaturization. As the miniaturization of transistors continues, quantum tunneling will make it difficult to insulate the relevant parts of the circuits. In order to build the primitive bistable components of computers, quantum-level behaviors would be unavoidable. In this context one solution that has been proposed involves exploiting quantum tunneling as part of the operation of the circuits themselves. There will be no explanation of such components and no light will be shed on practical solutions from a mechanistic perspective.

These challenges and details fall below the level of the primitive components of the mechanistic account of computation. The trouble here is that these primitive components are picked out by the logic of a particular kind of software. Thus the mereological ground floor for the mechanist's treatment of computation is established by reference to the abstract characteristics of computation rather than being explained by, or grounded in, mechanism. If we were concerned about questions like: What are the physical constraints on computation? How do bistable systems emerge in physical reality? Or any number of other questions about the physical instantiation of computation, the mechanistic approach will not satisfy.

ACKNOWLEDGMENTS

Many thanks to Mark Bickhard, Piotr Boltuc, Frances Egan, Jack Horner, Corey Maley, Marcin Miłkowski, Nico Orlandi, Gualtiero Piccinini, and Sarah Robins for helpful discussions of this material.

This work is supported by The National Security Agency through the Science of Security initiative contract #H98230-18-D-0009

REFERENCES

Bechtel, William, and Adele Abrahamsen. "Explanation: A mechanist alternative." *Studies in History and Philosophy of Science Part C: Studies in History and Philosophy of Biological and Biomedical Sciences* 36, no. 2 (2005): 421-441.

Bechtel, William, and Robert C. Richardson. *Discovering complexity: Decomposition and localization as strategies in scientific research*. MIT press, 1993.

Buechner, Jeff. *Gödel, Putnam, and Functionalism: A New Reading of 'Representation and Reality'*. MIT Press, 2008.

Craver, Carl and Tabery, James, "Mechanisms in Science", The Stanford Encyclopedia of Philosophy (Spring 2017 Edition), Edward N. Zalta (ed.), URL = https://plato.stanford.edu/archives/spr2017/entries/science-mechanisms/ (Last Accessed Jan 2, 2019).

Horner, Jack and John Symons. "Understanding Error Rates in Software Engineering: Conceptual, Empirical, and Experimental Approaches." *Philosophy & Technology* (2019): 1-16.

Kuhlmann, Meinard, and Stuart Glennan. "On the relation between quantum mechanical and neo-mechanistic ontologies and explanatory strategies." *European Journal for Philosophy of Science* 4, no. 3 (2014): 337-359.

Machamer, Peter, Lindley Darden, and Carl F. Craver. "Thinking about mechanisms." *Philosophy of science* 67, no. 1 (2000): 1-25.

Piccinini, Gualtiero. *Physical Computation: A Mechanistic Account.* Oxford: Oxford University Press, 2015.

Putnam, Hilary. *Representation and reality*. MIT press, 1988.

Skipper Jr, Robert A., and Roberta L. Millstein. "Thinking about evolutionary mechanisms: Natural selection." *Studies in History and Philosophy of Science Part C: Studies in History and Philosophy of Biological and Biomedical Sciences* 36, no. 2 (2005): 327-347.

Symons, John. "The Individuality of Artifacts and Organisms." *History and philosophy of the life sciences* (2010): 233-246.