Preprint**: Exploring Computational Theories of Mind, Algorithms, and Computations**[1]

Jordan Dopkins (jdopkins@gsu.edu)
Georgia State University, Perimeter College. Decatur, GA.

Review of *The Computational Theory of Mind.* Cambridge, 2023, iv + 75 pp, $64.99 HB

The Computational Theory of Mind, by Matteo Colombo and Gualtiero Piccinini, excels in providing a concise and authoritative introduction to computational theories of mind. The book offers a comprehensive history, articulation, and defense in just 75 pages.

Section 1 is brief and introduces readers to the central commitment of any computational theory of mind: the human central nervous system is a mind because it is a computing system. It also introduces us to an important caveat: what is often presented as a single theory is actually a family of theories. By the end of the chapter, readers should understand that the phrase "computational theory of mind" encompasses a family of theories aimed at explaining how our central nervous system carries out the computations relevant to minds.

Section 2 distinguishes between two perspectives on the history of Computational Theories of Mind (CTM). The traditional view traces CTM to the shortcomings of behaviorist psychology and the advent of digital computers in the 1940s and 50s. Colombo and Piccinini reject this, arguing that it distorts and oversimplifies CTM. Instead, they argue for a historical foundation rooted in more fundamental concepts like that of an algorithm or mechanism. For other examples, see Isaac 2018 and Uckelman 2018. I identify four virtues to their approach. First, it affords a nuanced understanding of seminal contributions to CTM. Take, for example, McCulloch and Pitt's 1943 paper on 'all or none' neural signals and the ways they can be represented by 1s and 0s. Colombo and Piccinini's way of thinking about the history of CTM allows us to see beyond analogies to digital computer signals and situate the paper in the longer history of attempts to formalize algorithms, dating back to Leibniz or Llull. Second, it captures aspects of CTM not linked to digital computers, like the brain's analog manipulation of continuous variables, as discussed by Turing (1950) and Von Neumann (1958). Third, Colombo and Piccinini's approach sharpens mind/computer analogies by highlighting specific points related to algorithms and mechanisms. Finally, it expands the historical narrative back to the Persian mathematician Al-Khwarizmi (c.780-850) and connects familiar figures like Descartes, Hobbes, and Lovelace.

One challenge I foresee is that the approach could trivialize the historical roots of CTM since appeals to algorithms or mechanisms can be found across the history of philosophy and science. Consider that Thales appealed to the mechanisms of seeds in his account of how moisture produced life or that Wittgenstein appealed to procedures/algorithms for returning truth values in *the Tractatus.* It would seem that histories of CTM that don't start with Thales or incorporate Wittgenstein are less comprehensive or are picked arbitrarily.

---

[1] This preprint has not undergone peer review (when applicable) or any post-submission improvements or corrections. The Version of Record of this article is published in *Metascience* and is available online at https://doi.org/10.1007/s11016-024-01009-0.

Section 3 analyzes the core notion of a computing system in CTM. The analysis proceeds by decomposition: A computing system is one that implements *algorithms*, or procedures (rules, steps, a guide) for solving *functions*. And functions are just mappings from one element in a domain to another. Put that together and you get the seemingly straightforward idea that a computing system is one that follows an algorithm(s); it is a system that follows procedures for returning a specific output from a specific input. Colombo and Piccinini discuss three, *digital* procedures in varying levels of detail: Turing machines, Church's lambda calculus, and Von Neuman's cellular automata. They also outline other, less conventional methods associated with analog, quantum, and even reservoir computing. The section ends with two lessons. First, tread lightly when thinking about CTM in terms of *information processing* rather than computation since those concepts are not the same. Second, CTM is not the view that minds/mental processes are equivalent to algorithm procedures, like Turing machines. CTM is the (slightly) more complicated view that minds/mental processes are specific, physical systems that *implement* algorithm procedures.

Section 4 is about theories of implementation. Colombo and Piccinini highlight their necessity by raising two triviality concerns. The first arises from the possibility that every physical system is a computing system, rendering CTM trivial because it would fail to identify anything distinctive about minds and central nervous systems. They would be no different than rocks or Walkmans with respect to their status as computational systems. The second concern stems from the possibility that every physical system implements every computation. This would render CTM trivial because it would be completely incapable of distinguishing the central nervous system or mind from every other physical system. Rocks, Walkmans, and central nervous systems would all have indistinguishable profiles from a computational point of view. Theories of implementation are meant to insulate CTM from triviality concerns by stipulating necessary or sufficient conditions a physical system must meet to properly implement an algorithm. That allows proponents of CTM to say things like "no, such and such is not a computing system, it does not implement any computation" or "sure, maybe such and such is a computing system, but it does not implement the computations relevant to having a mind".

Colombo and Piccinini discuss four theories of implementation. They begin with the simple mapping theory, which states that an isomorphism (mapping) between the states of a physical system and the steps in an algorithm is sufficient to establish that the physical system implements the algorithm. They then discuss three additional theories that emerged in response to criticisms: restrictive mapping theories, which limit what can be mapped from the physical system; semantic theories, which require the physical system to have semantic states; and mechanistic theories, which focus on functional properties in addition to the causal properties of the physical system. They also address anti-realist views about implementation, which recognize the benefits of treating systems as though they implemented computations despite the fact that they do not (as a matter of fact) do so.

Colombo and Piccinini effectively illustrate how theories of implementation can guard against the stated triviality concerns, which are about *all* physical systems. However, they leave a more plausible triviality concern on the table: consider a version of the second triviality concern that says "all *sufficiently complex* physical systems implement every computation". While the theories (as discussed by Colombo and Piccinini) may help us parse the computational differences between rocks and central nervous systems, they may not help us distinguish between Walkmans and central nervous systems or between my own central nervous system and the authors'. For more on this specific version of the triviality concern, see Sprevak (2018), and Williams (2023).

In section 5, Colombo and Piccinini present four abductive arguments for CTM. The biggest difference between them is the phenomena to be explained. Colombo and Piccinini present arguments that CTM offers the best account of the mind-body problem, cognition, the success of computational modeling, and the success of our abilities to build/understand non-biological, intelligent systems (AI). It is worth noting that some of the arguments discussed are not abductive arguments for CTM generally but for specific versions of CTM like *classical* CTM which also incorporates the Language of Thought Hypothesis. While Colombo and Piccinini were successful in supporting these specific versions of CTM, the section suggests the general commitments of CTM offer few standalone benefits and highlights significant internal disputes among CTM proponents.

Section 6 addresses problems for computational theories of mind. The problems can be divided into two kinds. The first has to do with the empirical evidence collected about the brain, consciousness, and the behaviors they are meant to underwrite. The evidence speaks against those things being computational and shows that they are better thought of as embedded, dynamic, architecturally different from computational systems, or even computationally intractable.

The second kind of problem is conceptual: computational theories of mind just don't have the conceptual tools to explain the apparent intentionality, semantic properties, phenomenal consciousness, or abilities for understanding that we associate with minds. Colombo and Piccinini clearly think these are the tougher problems for CTM and identify two ways to respond. The first is to bite the bullet and admit that CTM is a more limited hypothesis that must be augmented with another theory, like a theory of intentionality or consciousness. The second is to naturalize those properties. Colombo and Piccinini admit that there are varying degrees of success at this project and hint that understanding and semantic properties may be easier to naturalize than intentionality and consciousness.

In sum, the book is an excellent, advanced introduction to computational theories of mind. While it may not be suitable for undergraduates new to the philosophy of mind, it will be a valuable resource for more advanced students and faculty exploring CTM.

My thanks to Philip Groth and Daniel Williams for comments.

Works Cited

Isaac, A. M. C. (2018a). Computational thought from Descartes to Lovelace. In Sprevak, M., & Colombo, M. (eds.), The Routledge Handbook of the Computational Mind, 9–22. New York: Routledge.

McCulloch, W. S., & Pitts, W. H. (1943). A logical calculus of the ideas immanent in nervous activity. Bulletin of Mathematical Biophysics, 7: 115–33.

Sprevak, M. (2018). Triviality arguments about computational implementation. In Sprevak, M., & Colombo, M. (eds.), The Routledge Handbook of the Computational Mind, 175-91. New York: Routledge.

Turing, A. (1950). Computing machinery and intelligence. Mind, 49: 433–60.

Uckelman, S. (2018). Computation in mediaeval Western Europe. In Hansson, S. O. (ed.), Technology and Mathematics: Philosophical and Historical Investigations, 33–46. Berlin: Springer.

von Neumann, J. (1958). The computer and the brain. New Haven: Yale University Press.

Williams, Danielle. (2023). Implementation and Interpretation: A Unified Account of Physical Computation. UC Davis. ProQuest ID: WILLIAMS_ucdavis_0029D_22222.