Comments on Bringsjord's "Logicist Remarks"

William J. Rapaport

Department of Computer Science and Engineering,
Department of Philosophy, Department of Linguistics,
and Center for Cognitive Science
University at Buffalo, The State University of New York,
Buffalo, NY 14260-2500

rapaport@buffalo.edu
http://www.cse.buffalo.edu/~rapaport/

June 11, 2018

Abstract

A response to three comments by Selmer Bringsjord on my philosophy of computer science and my *Philosophy of Computer Science* (Rapaport, 2018a).

1 Introduction

I am grateful to my long-time friend and debating partner Selmer Bringsjord for the email interview that was the source of his "Logicist Remarks" (Bringsjord, 2018) and for his generous comments on my research and my textbook-in-draft (Rapaport, 2018a). In this brief response, I simply wish to clarify three of my positions.

2 A Procedural-Logical Controversy

Whereas I argue that computer science is fundamentally concerned with algorithms (Rapaport 2017b, pp. 13–16; Rapaport 2018a, §3.15), Bringsjord argues that "computation ... [is] a proper part of reasoning *and nothing more*" (Bringsjord, 2018, §"Actually, Computer Science Is a (Small) Proper Part of Logic", my italics). It's the "and nothing more" clause that I disagree with.

I agree that computation as a subject of study can be viewed "as a proper part of reasoning" or logic. But it can *just as well* be viewed as the study of (what Bringsjord somewhat dismissively characterizes as "do-this-step-do-that-step-do-this-step") procedures (loc. cit.). These are equivalent viewpoints from different perspectives. But I find the procedural perspective more perspicuous.[1]

This is exactly the same situation that we find in the theory of computation: Computation as a mathematical enterprise can be understood functionally, in terms of recursive functions or the lambda calculus (as well as in other ways, and by other formalisms), *as well as* procedurally, in terms of Turing machines or register machines (etc.). The multiple views (in both cases) are not rivals, but equivalent alternatives, each with its own advantages. Gödel found the Turing-machine analysis more convincing as a model of computability than even his own recursive functions (Gödel 1938, p. 168; Shagrir 2006; Sieg 2006; Soare 2009, §2; Copeland and Shagrir 2013). Similarly, I would argue, the procedural view is more compelling (for me, as well as for my students) than the logical view with respect to what is unique and interesting about computer science and computation.

3 Syntactic Semantics

I have long advocated for the position that syntax suffices for semantics—that the semantic enterprise of understanding is fundamentally a syntactic one. Briefly, I take syntax as the study of the properties of, and relations among, the members of a set of objects, and I take semantics as the study of the relations *between* two sets of objects—one studied syntactically, and other providing its semantic interpretation. (The latter set can also be studies syntactically, and its syntax is its "ontology".) But when you take the union of those two sets, the formerly semantic relations become syntactic ones of the union (Rapaport, 1986, 1988, 1995, 2000, 2002, 2003, 2006, 2011, 2012, 2017a, 2018b). A real-life, biological (and not merely "Strong AI") example of such a union is the neuron firings in our brain, some of which represent the objects in the external world and some of which represent the concepts (and language) that we use to understand them. But they all form one neural network.

Bringsjord says that "this ... does nothing beyond communicating the faith of computationalist materialists, and/or ... Strong AIniks" (Bringsjord, 2018, §"Semantics as Semantics, and Searle").[2] But it *does* do more than that: It shows that our subjective sense of understanding—the kind involved in Bringsjord's example of "the

[1]As I note in Rapaport 2018a, §§2.3, 2.7, on the question of what philosophy is, I take philosophy to be the *personal* search for truth, in any field, by rational means, following Hector-Neri Castañeda, who said that philosophy should be done "in the first person, for the first person" (Rapaport, 2005).

[2]I suspect that this section title is a typo for "Semantics as *Syntax*, and Searle".

shout by a grillmaster that our redolent burger is done" (loc. cit.)—is accomplished by a single system (a single, unioned set) that is understood syntactically, not by two separate systems (a syntactically understood one and its semantic interpretation).

4 Hypercomputation

Bringsjord's discussion of my position on hypercomputation was based on an earlier version of Rapaport 2018a, Ch. 11, than the one currently available. That earler version was, indeed, somewhat "noncommittal" (Bringsjord, 2018, §"Hypercomputation").

Rather than distinguishing between Turing-machine computation and hypercomputation, I prefer to think of there being three categories:[3]

Sub-Turing Computation:
Finite-state automata, pushdown automata, primitive recursive functions, etc.

Turing-Machine Computation:
Turing machines and their equivalents (partial recursive functions, lambda calculus, etc.)

Super-Turing Computation:
Oracle machines, Zeus machines, Malament-Hogarth machines, analog recurrent neural networks, interactive computing, trial-and-error machines, etc.

To my mind, the only interesting kinds of super-Turing computation are not the "newer physics" kind (Zeus machines, etc.),[4] but the ones that can be modeled by Turing's own theory of oracle machines. These include interactive and trial-and-error computing. But oracle computation, studied under the rubric 'relative computability', is well-understood and not something that computer scientists have ignored (as some hypercomputationalists have suggested). Nor is it typically understood as a counterexample to the Church-Turing Computability Thesis. (See Davis 1958, pp. 20–24; Soare 2009, 2012; and Fortnow 2010 for this point of view.)

5 Conclusion

Bringsjord raised a number of important questions in his "Final Remarks" (some of which I touch on in my book), observing that it was "time to talk again to

[3]I am limiting myself here to digital computing, so analog computation is another story, told best, I think, in Piccinini 2015.

[4]The term 'newer physics' is from Copeland and Sylvan 2000, p. 190

Rapaport". I look forward to continuing our conversation!

References

Bringsjord, S. (2018, forthcoming). Logicist remarks on Rapaport on philosophy of computer science. *American Philosophical Association Newsletter on Philosophy and Computers*.

Copeland, B. J. and O. Shagrir (2013). Turing versus Gödel on computability and the mind. In B. J. Copeland, C. J. Posy, and O. Shagrir (Eds.), *Computability: Turing, Gödel, Church, and Beyond*, pp. 1–33. Cambridge, MA: MIT Press.

Copeland, B. J. and R. Sylvan (2000). Computability is logic-relative. In D. Hyde and G. Priest (Eds.), *Sociative Logics and Their Applications: Essays by the Late Richard Sylvan*, pp. 189–199. Ashgate.

Davis, M. D. (1958). *Computability & Unsolvability*. New York: McGraw-Hill.

Fortnow, L. (2010, December). What is computation? *Ubiquity 2010*. Article 5, http://ubiquity.acm.org/article.cfm?id=1921573.

Gödel, K. (perhaps about 1938). Undecidable Diophantine propositions. In S. Feferman et al. (Eds.), *Collected Works, Vol. III*, pp. 164–175. Oxford: Oxford University Press. http://tinyurl.com/ybw84oa6.

Moor, J. H. (Ed.) (2003). *The Turing Test: The Elusive Standard of Artificial Intelligence*. Dordrecht, The Netherlands: Kluwer Academic.

Piccinini, G. (2015). *Physical Computation: A Mechanistic Account*. Oxford: Oxford University Press.

Rapaport, W. J. (1986). Searle's experiments with thought. *Philosophy of Science 53*, 271–279. http://www.cse.buffalo.edu/~rapaport/Papers/philsci.pdf.

Rapaport, W. J. (1988). Syntactic semantics: Foundations of computational natural-language understanding. In J. H. Fetzer (Ed.), *Aspects of Artificial Intelligence*, pp. 81–131. Dordrecht, The Netherlands: Kluwer Academic Publishers. http://www.cse.buffalo.edu/~rapaport/Papers/synsem.pdf; reprinted with numerous errors in Eric Dietrich (ed.) (1994), *Thinking Machines and Virtual Persons: Essays on the Intentionality of Machines* (San Diego: Academic Press): 225–273.

Rapaport, W. J. (1995). Understanding understanding: Syntactic semantics and computational cognition. In J. E. Tomberlin (Ed.), *AI, Connectionism, and Philosophical Psychology (Philosophical Perspectives, Vol. 9)*, pp. 49–88. Atascadero, CA: Ridgeview. http://www.cse.buffalo.edu/~rapaport/Papers/rapaport95-uu.pdf. Reprinted in Toribio, Josefa, & Clark, Andy (eds.) (1998), *Language and Meaning in Cognitive Science: Cognitive Issues and Semantic Theory (Artificial Intelligence and Cognitive Science: Conceptual Issues)*, Vol. 4 (New York: Garland).

Rapaport, W. J. (2000, October). How to pass a Turing test: Syntactic semantics, natural-language understanding, and first-person cognition. *Journal of Logic, Language, and Information 9*(4), 467–490. http://www.cse.buffalo.edu/~rapaport/Papers/TURING.pdf. Reprinted in Moor 2003, 161–184.

Rapaport, W. J. (2002). Holism, conceptual-role semantics, and syntactic semantics. *Minds and Machines 12*(1), 3–59. http://www.cse.buffalo.edu/~rapaport/Papers/crs.pdf.

Rapaport, W. J. (2003). What did you mean by that? Misunderstanding, negotiation, and syntactic semantics. *Minds and Machines 13*(3), 397–427. http://www.cse.buffalo.edu/~rapaport/Papers/negotiation-mandm.pdf.

Rapaport, W. J. (2005). Castañeda, Hector-Neri. In J. R. Shook (Ed.), *The Dictionary of Modern American Philosophers, 1860–1960*, pp. 452–412. Bristol, UK: Thoemmes Press. http://www.cse.buffalo.edu/~rapaport/Papers/hncdict.tr.pdf.

Rapaport, W. J. (2006). How Helen Keller used syntactic semantics to escape from a Chinese room. *Minds and Machines 16*, 381–436. http://www.cse.buffalo.edu/~rapaport/Papers/helenkeller.pdf. See reply to comments, in Rapaport 2011.

Rapaport, W. J. (2011, Spring). Yes, she was! Reply to Ford's 'Helen Keller was never in a Chinese room'. *Minds and Machines 21*(1), 3–17. http://www.cse.buffalo.edu/~rapaport/Papers/Papers.by.Others/rapaport11-YesSheWas-MM.pdf.

Rapaport, W. J. (2012, January-June). Semiotic systems, computers, and the mind: How cognition could be computing. *International Journal of Signs and Semiotic Systems 2*(1), 32–71. http://www.cse.buffalo.edu/~rapaport/Papers/Semiotic_Systems,_Computers,_and_the_Mind.pdf. Revised version published as Rapaport 2018b.

Rapaport, W. J. (2017a, Fall). Semantics as syntax. *American Philosophical Association Newsletter on Philosophy and Computers 17*(1), 2–11. http://c.ymcdn.com/sites/www.apaonline.org/resource/collection/EADE8D52-8D02-4136-9A2A-729368501E43/ComputersV17n1.pdf.

Rapaport, W. J. (2017b, Spring). What is computer science? *American Philosophical Association Newsletter on Philosophy and Computers 16*(2), 2–22. Paper based on 2017 APA Barwise Award talk; http://c.ymcdn.com/sites/www.apaonline.org/resource/collection/EADE8D52-8D02-4136-9A2A-729368501E43/ComputersV16n2.pdf.

Rapaport, W. J. (2018a). Philosophy of computer science. Current draft in progress at http://www.cse.buffalo.edu/~rapaport/Papers/phics.pdf.

Rapaport, W. J. (2018b). Syntactic semantics and the proper treatment of computationalism. In M. Danesi (Ed.), *Empirical Research on Semiotics and Visual Rhetoric*, pp. 128–176. Hershey, PA: IGI Global. References on pp. 273–307; http://www.cse.buffalo.edu/~rapaport/Papers/SynSemProperTrtmtCompnlism.pdf. Revised version of Rapaport 2012.

Shagrir, O. (2006). Gödel on Turing on computability. In A. Olszewski, J. Wołenski, and R. Janusz (Eds.), *Church's Thesis after 70 Years*, pp. 393–419. Ontos-Verlag. Page references to online version at http://edelstein.huji.ac.il/staff/shagrir/papers/Goedel_on_Turing_on_Computability.pdf.

Sieg, W. (2006, June). Gödel on computability. *Philosophia Mathematica 14*, 189–207. Page references to preprint at http://repository.cmu.edu/cgi/viewcontent.cgi?article=1112&context=philosophy.

Soare, R. I. (2009). Turing oracle machines, online computing, and three displacements in computability theory. *Annals of Pure and Applied Logic 160*, 368–399. Preprint at http://www.people.cs.uchicago.edu/~soare/History/turing.pdf; published version at http://ac.els-cdn.com/S0168007209000128/1-s2.0-S0168007209000128-main.pdf?_tid=8258a7e2-01ef-11e4-9636-00000aab0f6b&acdnat=1404309072_f745d1632bb6fdd95f711397fda63ee2. A slightly different version appears as Soare 2013.

Soare, R. I. (2012). Formalism and intuition in computability. *Philosophical Transactions of the Royal Society A 370*, 3277–3304. doi:10.1098/rsta.2011.0335.

Soare, R. I. (2013). Interactive computing and relativized computability. In B. J. Copeland, C. J. Posy, and O. Shagrir (Eds.), *Computability: Turing, Gödel, Church, and Beyond*, pp. 203–260. Cambridge, MA: MIT Press. A slightly different version appeared as Soare 2009.