Selmer Bringsjord
Selmer.Bringsjord@gmail.com

August 5, 2018

Logicist Remarks on Rapaport on Philosophy of Computer Science+

(in the context of his Barwise Prize)

Introductory Remarks

I shall restrict my brief remarks herein to William "Bill" Rapaport on philosophy of computer science (PCS) and some intimately related topics (which are gestured at by the superscripted + in my title), guided by his ever-expanding, online *Philosophy of Computer Science* (*PCS*); and I'll begin (in the next section) with some comments on this restriction itself. The present commentary is informed by a recent, sustained dialogue with Rapaport, one undertaken to inform my remarks (and, I confess, to allow me to somewhat selfishly enjoy some philosophical debate).[1] Unfortunately, and I wrap up the present essay by returning to this issue, our dialogue, at least by my lights, needs to continue, because important societal issues in the context of the philosophico-history of computer science and AI have been left unanalyzed, and more importantly (at least as I see things), because Rapaport (and his readers) would be well-served by having some errors that infect his *PCS*, beyond those touched upon herein, remedied. In particular, since — for reasons to be shortly seen — he views CS through the obfuscating lens of *algorithms* (first do *A*; now do *B*; if condition *C* holds, do *A* again; and so on), rather than as a part of reasoning in a well-defined logical system, it's especially important that Rapaport's account of PCS, which seems destined to be highly influential, be modified. I suppose it's possible that despite sustained discussion with

[1] The current version of *PCS*, as this sentence is written, is May 2018, and is available here: https://cse.buffalo.edu/~rapaport/Papers/phics.pdf. The reader should take account of the difference between PCS (the subject) and — note the italics — *PCS*, the Rapaportian book on that very subject.

him subsequent to what informs the present essay, he may resist such modification; but I hold out hope that he will engage in the discussion, and see the light.

The Vastness of Rapaport's Reach vs. What I Treat

As the reader will well know, much if not all of the field of philosophy is composed of sub-parts long traditionally designated by the phrase 'philosophy of *X*,' where instantiations of *X* include, for instance: `mind,' 'art,' 'economics,' 'religion,' and 'language.'[2] In cases where a sub-part of philosophy is designated without this syntax, as for example 'epistemology,' or 'metaphysics,' there can be little doubt that no accuracy is sacrificed if the Po*X* template is employed (though elegance, I concede, is threatened). Rapaport has made contributions in many a philosophy of *X* ≠ 'computer science' area, but my interest, in keeping with his recent Barwise Prize, and with the venue that the present discussion is bound for, is PCS, to which, arguably, Rapaport is *the* greatest contributor — and at any rate he certainly stands minimally as one of the five or so greatest authorities on PCS today, when the whole of CS, from theory to concrete practice, is considered. The restriction to PCS means, in particular, that very little will be said herein about philosophy of artificial intelligence (PAI)[3], another Po*X* subject on which Rapaport is a world-class authority, in no small part because of his being a long-time leader in a seminal team at the University at Buffalo devoted to AI and computational cognitive science. Multiple essays of the present sort could be written on the work of this group in connection with PCS and PAI, a group long led as well by AI pillar Stuart "Stu" Shapiro. Shapiro and Rapaport have long labored to advance the SNePS system, which can be used to build artificial agents that know, reason, plan, and act. Obviously, upon hearing my implicit claim (expressed by the previous sentence) that such artificial agents are on a planet that's lucky, AI-wise, to build self-driving cars that only *occasionally* kill people, readers who are philosophers will pay attention. Are such agents in fact with us

[2] I don't mean to imply that the sub-parts of philosophy to which I refer are self-contained. In point of fact, philosophy of language and philosophy of logic (in the Occidental case, anyway), are inseparably linked. Another inseparable link, one at the heart of any comprehensive analysis of Rapaport's PCS and his body of work, is that between PCS and PAI.

[3] PAI, and for that matter AI itself from a philosophical point of view, is covered in the SEP entry *Artificial Intelligence* (https://plato.stanford.edu/entries/artificial-intelligence).

already? it will be asked by such readers. I think an affirmative reply would come from Rapaport and Shapiro, and I suggest that philosophers of CS, AI, mind, and logic study the work in question, deeply.

So, the target is Rapaport on PCS. In our context, this target should strike the alert reader as pregnant. I'm writing for the Philosophy and *Computers* Committee (P&CC) of the APA; note my emphasis. It follows that I'm writing for a committee whose mission centers on the relationship between philosophy on the one hand, and "computers" on the other. But what is the meaning of 'computers' in this mission? This is very much like the question with which Rapaport has wrestled, when for instance he deliberated about what title to use for his *PCS* book. It turns out that he isn't particularly happy with the phrase 'philosophy of computer science.' He finds the continuous string 'computerscience' to be helpful, because (to brutally simplify the issue and his thinking) this neologism is easier to view as something that picks out a domain over which to philosophize that isn't in any way narrowly restricted to computers, or to what must be a science, and so on. One would think that a similar attitude is wise to adopt regarding the title and nature of the P&CC: Sure this committee's mission isn't any such narrow thing as exploring, sorting out, and charting for the APA the relationship between philosophy and, literally, computers, as in laptops and desktops. Surely 'computers' here is to mean that vast space of all philosophical things computational and computation-based, from all that Rapaport deals with in the bordering-on-1000-page *PCS* volume, to rigorously characterizing what privacy is by the standards of philosophy (which includes characterizations in its analytic side that at least aspire to jointly necessary and sufficient conditions) in an age of social media, where interaction on the shoulders of computation has led to philosophical problems as thorny as most longstanding ones, an issue to which I return when wrapping up.

Please note that in confining attention to Rapaport on PCS, the target remains enormous. This is true for the simple reason that PCS itself is gigantic. It's perhaps not uninteresting that in philosophy today, still, PCS is often thought of as some kind of Lilliputian curiosity off to the side, with the center proudly occupied by the venerable giants (ethics, epistemology, metaphysics, etc.) continuing to go merrily along as they have since Socrates. Those with this attitude should read *PCS*, and then think objectively about whether this traditional center-side conceptualization is accurate and/or sensible today. We have reached a time, now, when the prospect of artificial agents (which after all consist in things whose essence is *computing* over input to produce output)

that are ethical agents unto themselves, with radical forms of autonomy (e.g. the ability to write the very programs that power them), seem to many imminent. Understanding these creatures, and what they mean for us and the cosmos, will be impossible without a prior understanding of PCS.

Actually, Compuer Science is a (Small) Proper Part of Logic

In *PCS* Rapaport bravely gives a distilled answer to "What is computer science?"; the answer is given at the very end of the chapter whose title is the very question, and, verbatim, Rapaport's summative reply is this:

> Computer science is the scientific (or STEM) study of:
> what problems can be solved,
> what tasks can be accomplished,
> and what features of the world can be understood . . .
> . . . computationally, that is, using a language with only:
> 2 nouns ('0', '1'),
> 3 verbs ('move', 'print', 'halt'),
> 3 grammar rules (sequence, selection, repetition), and nothing else,
> and then to provide algorithms to show how this can be done:
> efficiently, practically, physically, and ethically.

This answer has a certain flair, I think. After all, by it, a great, big, daunting philosophical question is answered crisply and confidently in nothing more than flash. Unfortunately, this is an account of computer science ferociously biased in the procedural direction. (The account is very nicely elaborated in *PCS*, and is explicitly aligned with (similarly biased) accounts of so-called "computational thinking," the cultivation of which, at least in the U.S., is sought by its federal government, by many states as well, and by funders like the Gates Foundation.) Yet this is not *my* answer to the question, nor is it even *approximately* in line with my answer; and I doubt whether it's the answer that would be given by anyone who thinks of computation as a proper part of

reasoning and nothing more, not as a do-this-step-do that-step-do-this-step (DTS) process. Moreover, for philosophy and philosophers, I think DTS account of CS is particularly unwise. The reason is simply that philosophers, if they do nothing else, reason; and to teach philosophy is therefore naturally to in no small part teach how to reason. (Such pedagogy is of course self-evidently in operation in the case of logic as taught and pursued under philosophy.) In my experience, sometimes philosophers with little exposure to CS are surprised to learn that computation can be studied and mastered, without loss of formal generality or of practical functioning, as reasoning, but some illumination can be provided quickly by presenting the rudiments of standard logic programming. I personally have found that the instant a rigorously trained philosopher without any prior exposure to computer science/computation is shown the underlying theory of logic programming for Prolog (a programming language in the logic-programming fold), a light snaps on. (Wonderful introductory coverage of logic programming is provided in Ebbinghaus et al. 1994.) In fact, sometimes the coming on of that mental light is more akin to a sort of explosive eureka moment. "Wait, you mean a valid deduction by the machine from this set A of formulae expressed in something that looks quite like first-order logic, to that particular formula p, is what execution of my 'program' consists in?!?" That is correct. No need to write any DTS thingie here, at all. The traditional coverage of logic programming in mathematical logic isn't based on inference schemata that philosophers learn (e.g., *modus tollens*, universal elimination, etc.), but rather on inference schemata in the proof theories based on schemata conforming to **resolution**, but regardless, this is a far superior way to understand what computation is, in my opinion — yet this way is an utter alien in the DTS landscape of *PCS*.[4]

Semantics as Semantics, and Searle

[4] I would personally have preferred to use automated theorem proving rather than Prolog's basis in what I just wrote, but the need for economy at the moment rules. This is as good a place as any to report that in my interview of Rapaport, he indicated that he opted for DTS, and the encapsulation of it that I've quoted, for pedagogical purposes. However, even taking his expression of this strategy at face value, as I've explained, even from the perspective of pedagogy, reasoning is by my lights something much more valuable to teach than DTS. And besides, even after DTS is used, we are still left with the challenge of showing that the procedural artifact we have produced is correct; and showing this can only be accomplished via reasoning. Why not simply start and end with reasoning?

I have been intrigued for years by Rapaport's longstanding desire to portray semantics as syntax, and accordingly took up for the present project his 2016 "Semantics as Syntax" (which was wisely solicited by editor Boltuc) to study. Rapaport, as far as it goes, is entirely correct, at least spiritually speaking. (I'm limited to saying only that Rapaport is in *spirit* right, because were details discussed here, too much space would be consumed.) For my money, one major reason he's right is that the fundamental observations upon which proof-theoretic semantics (in any form thereof) is motivated by, and possibly even rests upon directly, can't be denied.[5] A simple example comes by way of considering the standard extensional semantics of a conditional with p as antecedent and q as consequent. We are standardly told in this case that the semantics for a material conditional $p \Rightarrow q$ consists in that such a conditional holds if and only if (iff), if p, then q. That is, expressed a bit more succinctly, $p \Rightarrow q$ iff if p then q. When you think about it, this is quite extraordinarily one-dimensional. Does it not directly give semantics via syntax? Consider the conditional $(p \& q) \Rightarrow q$. Does this conditional have the semantic value TRUE? Certainly. Why? Because it's TRUE iff if p and q, then q. Well, is it in turn TRUE that if p and q, then q? Absolutely:

> **Proof**: Suppose that p holds, along with q. We can deduce q directly. Hence our supposition implies q. **QED**

We are here using the standard textbook semantics for elementary extensional deductive logic, in use in classrooms across the globe, and what just happened? What happened is that we pinned down the meaning of the syntactic formula via a perfectly, indeed purely, syntactic process.[6] I view Rapaport as having found this phenomenon at work in a deep and intricate way, far and wide.

Yet why do I say that Rapaport's "sem-by-syn" view is correct only *as far as it goes*? The reason is that Rapaport is spot on with respect to *one* sense of "semantics," and dead wrong with regard to *another* sense of the term. The first sense aligns with proof-theoretic semantics, in

---

[5] Readers unfamiliar with proof-theoretic semantics could start with (Gentzen 1935). For what it's worth, nearly all my own work in intensional logic and philosophy is proof-theoretic in nature. See e.g. (Bringsjord et al. 2016).

[6] Die-hard Tarskians might accuse me of tendentiously and unfairly passing straightaway to a proof, rather than giving a truth-table or truth-tree (or in the first-order case a model/interpretation). Balderdash. We shall need for the skeptic a proof that the result of tabular or tree-based manipulation yields TRUE.

general; we have just seen this sense in operation on a simple specimen; *and* it aligns with any formal dyad covering syntax on the one hand and semantics on the other. Unfortunately, the second sense can't be separated from *understanding on the part of a mind*; this is the Searlean sense of semantics, and is what stands at the heart of Searle's justly famous Chinese Room Argument (CRA), whose kernel, as a slogan, is that syntax doesn't produce semantics. Rapaport believes that the sem-by-syn view can be extended in order to allow syntactic expressions (e.g., 'hamburger') to be "internalized," and hence CRA to be dodged. He writes:

> In the case of a real human being, [a] *representative* is the end result of, say, the visual process of seeing a hamburger … resulting in a "mental image" of a hamburger. … More precisely, the biological neural network in the human's brain has neurons whose firing represent the word 'hamburger', and it has neurons whose firings represent the actual hamburger. Both of these sets of neuron firings are in the same "language" — the same syntactic system. (Rapaport 2016, 12)

This quote does nothing beyond communicating the faith of computationalist materialists, and/or (with the 'neuron' here e.g. mapped to artificial neurons in artificial neural networks so in vogue again these days) Strong AIniks. Can't we imagine this more elaborate syntactic dance happening in the complete and utter absence of our understanding, bound up with subjective awareness as it is, of the shout by a grillmaster that our redolent burger is done? Of course we can. What Rapaport is in end doing is ingeniously (but to a degree unwittingly) working out the sem-by-syn paradigm in and for AI — but not for *us*.

Hypercomputation

Rapaport's *PCS* includes a chapter on hypercomputation (which is, harshly encapsulated, forms of information-processing more powerful than the operation of standard Turing machines); coverage of the topic therein is what most would no doubt classify as "steadfastly balanced." I somewhat less charitably classify this chapter as noncommittal, and in being so, well, irrational. However, the chapter is also, even in its present, not-fully-polished form, the absolute best overview of the topic available in one place, over one digestible-in-one-sitting stretch of content.

Indeed, I suspect that even most aggressive fans of hypercomputation will regard the chapter's wishy-washy maybe-maybe-not position on hypercomputation to be fully redeemed by its laconic erudition, right down to the lucid presentation of some key theorems. After all, *PCS* is intended to be a broad-coverage textbook, not a polemical position statement.

Nonetheless, I've declared the chapter to be irrational. Why? In short, because there can be no denying, in light of the relevant logico-mathematics, that hypercomputation is as real and robust as can be, in the context of the fact that even if (like me) we count Leibniz as having discovered general-purpose computation in the 17th century, the human race has really only been at this modern computation thing for about three centuries. The late 20th century, and the beginning of the third millennium, have revealed that computation absolutely, positively cannot be rationally restricted to what standard Turing machines and their equivalents (which Rapaport lists and often discusses in *PCS*) can compute. I can't here review in any detail my own writings on this subject, and will rest content to mention but two things. To wit:

One: Rapaport respectfully cites and discusses Martin Davis's "The Myth of Computation." While there can be no denying that Davis is the author of much brilliant work, this paper is far from his finest hour; it may in fact be his worst. Calling a spade a spade (and I did have the opportunity to do so orally, in debating the issue with Davis in person), joined by my colleague N.S. Govindarajulu, we wrote something I recommend to Rapaport, his readers, and readers of the present essay: "The Myth of 'The Myth of Hypercomputation' " (Govindarajulu & Bringsjord 2012), in which is shown that Davis's arguments are anemic at best, and stunningly fallacious at worst. I confess to being deeply surprised that Rapaport is content, at least at present, to leave the impression that Davis may have succeeded in revealing that hypercomputation is to be placed alongside, say, Hercules and Odin.

Two: It's a logico-mathematical fact that hypercomputation is as real as can be. In the logicist interpretation of computer science adumbrated above, we have only to consider, for a few minutes, any number of computing machines vastly more powerful than standard Turing machines and their equivalents, specified via the use of formal logic. Not wanting (again) to cite my own work in this connection, I can simply rely on infinite-time Turing machines (Hamkins & Lewis 2000); they provably exceed standard Turing machines, and yet *are* Turing machines; end of story. An even-more-direct route is simply to take note of the fact that formal logic includes infinitary logics, and some reasoning (e.g. proof discovery) in even the smallest of these (which

allow infinitely long formulae and infinitely long proofs) is logic-style hypercomputation. Of course, some myopic empiricists may deny the reality of hypercomputation because they affirm the dogma that what is *real* is only what is *physical*. But this position is not only at odds with such mathematical facts as that there is a natural number N too large to correspond to any physical entity whose components sum to N; it's also at odds with something that Rapaport leaves aside: Since we are coming to see that physics can be axiomatized (by, say, the axiom system **P**), absent a disproof of the proposition that **P** and a formal assertion of the *physical* existence of hypercomptuational machine is consistent, it's irrational to advance the claim that hypercomputation is only mathematically possible.[7]

Final Remarks

Any serious dialogue with Rapaport, and engagement with his writings, could clearly continue, profitably and enjoyably, for a very long time. Yet, as is always the case, in order for a piece to be delivered and published, we must end — with, if you'll allow, a final thought: viz., that we need to hear at some point soon from Rapaport-qua-philosopher on the *history* of computation, of the fields which centrally partake of it (e.g. AI, logic, mathematics, linguistics, and nowadays computing machines as ethical agents), and on the complex and philosophically charged turbulence that has now been catalyzed by so-called "social media." Rapaport's professional life shows no signs of slowing down (witness the ever-growing *PCS* book itself), which means his contributions will continue, but his professional life to this point has passed through the evolution of the computational sciences over a period of decades, during which time *a lot* has happened. Rapaport is one of only a handful of computationally informed philosophers who have seen firsthand the evolution (with an occasional spate of rapid change) of the many parts of philosophy intimately connected to computation (philosophy of mind, of language, etc.). Did he ever think for a moment, yesterday, that today's advocacy of the end of programming (in light of such phenomena as "Deep Learning") would ever arrive? That the concept of a machine which self-learns and thereby beats humans at their own games would become reality, as happened in the case of AlphaGo? Did he think, yesterday, that computation, first isolated in the minds and

[7] (Govindarajulu et al. 2015) isn't a bad place to start reading about such matters.

soon thereafter the simple, disconnected "pet" machines of Turing and von Neurmann et al., would come to mediate arguably all that Earth's technologized youth do, daily, via social-media technology? In all this, who are we? What is truth? What is fake? What is real? What control can computation be allowed to have over our interaction with each other, and over the analysis and presentation thereof? Philosophy, and anyone concerned with the intersection "philosophy and computers," is going to need to come to grips with these computation-infused questions, the lack of answers to which has already started to plague us.[8] Actually, truth be told, *I* need to come grips in this regard. Time to talk again to Rapaport …

Acknowledgments

Some research in AI and theoretical computer science that informs my commentary was made possible by support from ONR and AFOSR, and I'm very grateful for this support. I have an enormous debt to Piotr Boltuc for his guidance and supernatural patience as I (irrationally?) poured more and more time into thinking about Rapaportian work, all the while with the clock cranking beyond a series of promised delivery dates. Rapaport's body of work, as I've said, has only been quickly touched upon herein. That body of work is endlessly stimulating, and I'm grateful to Rapaport for creating it.

[8] I recommend as a quick, non-technical start to this side of PCS: (Wiesberg 2018).

References

Govindarajulu, N.S. & Bringsjord, S. (2012) "The Myth of 'The Myth of Hypercomputation' " *Parallel Processing Letters* **22.3**. URL: https://doi.org/10.1142/S0129626412400129.

Bringsjord, S., Licato, J., Arista, D., Govindarajulu, N.S. & Bello, P. (2016) "Introducing the Doxastically Centered Approach to Formalizing Relevance Bonds in Conditionals" in Müller, V., ed., Computing and Philosophy (Berlin, Germany: Springer), pp. 117–131. This volume is *Synthese Library 375*, in *Studies in Epistemology, Logic, Methodology, and Philosophy of Science*.

Davis, M. (2004) "The Myth of Computation," in Teuscher, C., ed., *Alan Turing: Life and Legacy of a Great Thinker* (Berlin, Germany: Springer), pp. 195–211.

Ebbinghaus, H., Flum, J. & Thomas, W. (1994) *Mathematical Logic* (2nd edition) (New York, NY: Springer-Verlag).

Gerhard G. (1935) "Investigations into Logical Deduction" in *The Collected Papers of Gerhard Gentzen*, edited by M.E. Szabo (Amsterdam, The Netherlands: North-Holland), pp. 68–131. (This is an English version of the well-known 1935 German version.)

Govindarajalulu, N.S., Bringsjord, S. & Taylor, J. (2015) "Proof Verification and Proof Discovery for Relativity" *Synthese* **192.7**: 2077–2094.

Hamkins, J. and Lewis, A. (2000) "Infinite Time Turing Machines" *Journal of Symbolic Logic* **65.2**: 567–604.

Weisberg, J. (2018) "The Digital Poorhouse" *The New York Review of Books*, June 7 2018, online.

Rapaport Resources:

Rapaport's PCS is currently available, in the May 2018 edition, here: https://cse.buffalo.edu/~rapaport/Papers/phics.pdf.

Rapaport, W. (2016) "Semantics as Sytax." https://cse.buffalo.edu/~rapaport/Papers/synsemapa.pdf