CANADIAN JOURNAL OF PHILOSOPHY
Volume 34, Number 3, September 2004, pp. 375-410

Functionalism, Computationalism, and Mental Contents[1]

GUALTIERO PICCININI
Washington University
St. Louis, MO 63130-4899
USA

> Almost no one cites Sellars, while reinventing his wheels with gratifying regularity. (Dennett 1987, 349)

In philosophy of mind, there is functionalism about mental states and functionalism about mental contents. The former — mental state functionalism — says that mental *states* are individuated by their functional relations with mental inputs, outputs, and other mental states. The latter — usually called functional or conceptual or inferential role semantics — says that mental *contents* are constituted by their functional relations with mental inputs, outputs, and other mental contents (and in some versions of the theory, with things in the environment). If we add to mental state functionalism the popular view that mental states have their content essentially, then mental state functionalism may be seen as a form of functional role semantics and a solution to the *problem of mental content*, namely, the problem of giving a naturalistic explanation of mental content. According to this solution, the functional relations that constitute contents are physically realized — in a metaphysically unmysterious way — by the functional relations between mental inputs, outputs, and the mental states bearing those contents. But for this solution

---

[1] I'd like to thank those who commented on previous versions of this paper, especially Eric Brown, Robert Cummins, Frances Egan, Peter Machamer, Susan Schneider, and two anonymous referees.

to be noncircular, the functional relations between mental inputs, outputs, and states must be specified in a way that does not appeal to the contents of mental states.

Philosophers who endorse mental state functionalism typically also endorse computationalism, or the Computational Theory of Mind (CTM), according to which the functional relations between mental inputs, outputs, and states are computational.[2] Unfortunately, most of the same philosophers also endorse what I call the *semantic view of computation*, according to which computational relations are individuated by the contents of the computational inputs, outputs, and internal states.[3]

To distinguish it from mental content, I will call the content of computational states computational content. The problem of how computational states have content, analogous to the problem of mental content, may be called the *problem of computational content*. Since CTM ascribes computations to the mind and the semantic view of computation holds that computational states have their content essentially, the conjunction of CTM and the semantic view of computation entails that CTM is a representational theory, i.e. a theory that ascribes content to the mind. Because of this, many computationalists have supposed that solving the problem of computational content may be (perhaps part of) the solution to the problem of mental content. Finally, the fact that CTM plus a theory of computational content is seen as a theory of mental content is used as a reason in favor of CTM: we should believe CTM because it offers (a step towards) a naturalistic explanation of mental content.[4]

---

[2] For an extended discussion of the relationship between these two doctrines, see Piccinini forthcoming a.

[3] Cf. Fodor: 'I've introduced the notion of computation by reference to such semantic notions as content and representation: a computation is some kind of content-respecting causal relation among symbols' (Fodor 1998, 11). Cf. also Pylyshyn 1984, 30.

[4] This attitude is well expressed in the following passage, where the author seems to see no relevant distinctions between semantic, computational, informational, and intentional descriptions:

> It is widely recognized that computation is in one way or another a symbolic or representational or information-based or semantical — i.e., as philosophers would say, intentional — phenomenon. Somehow or other, though in ways we do not yet understand, the states of a computer can model or simulate or represent or stand for or carry information about or signify other states in the world.... The *only* compelling reason to suppose that we (or minds or intelligence) might be computers stems from the fact that we, too, deal with representations, symbols, meanings, and the like (Smith 1996, 9-11; emphasis added).

But the semantic view of computation generates a circularity between functional role semantics, mental state functionalism, and computationalism. According to mental state functionalism plus the view that mental states have their content essentially, mental contents are constituted by the functional relations of the mental states that carry the contents. According to computationalism, those functional relations are computational. And according to the semantic view of computation, computational relations are individuated by the contents of the states that enter those relations. We are back to explaining contents, which is what we were hoping to explain in the first place.

This paper tells the story of how contemporary philosophers of mind entangled themselves in this circularity, so that computation and content got almost inextricably intertwined, and how the same philosophers tried to untangle themselves. The purpose is twofold: on one hand, to reconstruct an episode of recent and influential philosophy; on the other hand, to diagnose what went wrong. The upshot is that the semantic view of computation should be replaced by a functional view of computation, and that the problem of mental content should be solved independently of the question of whether mental states are computational. I hope this will make room for a better understanding of both computation and content.

I Content in Early Computationalism

The modern history of computationalism, and the connection between computation and content, goes back to the origin of computability theory. Alan Turing formulated his theory of computation in the context of investigations on the foundations of mathematics.[5] For instance, he wanted to prove that the decision problem for first order logic had no algorithmic solution. He formulated a theory of computation in terms of what are now called Turing Machines (TMs). Turing argued that some TM could carry out any computation that a human being could carry out. In his argument, Turing described TMs anthropomorphically as 'scanning' their tape, 'seeing symbols,' having 'memory' or 'mental states,' etc., although he introduced all these terms in quotation marks, presumably to underline their metaphorical use (Turing 1936-7, 117-18). Moreover, in using TMs for his mathematical purposes, Turing assigned interpretations to the inputs and outputs of TMs, usually as encoding

---

[5] For more details, see Piccinini 2003b and 2003a, chs. 1 and 3.

real numbers, but sometimes as encoding TM programs or formulae in a logical calculus. So far, there was nothing methodologically problematic with what Turing did. TMs were to be understood as mechanisms for deriving strings of symbols, and the theorist was free to assign interpretations to the strings (within the relevant methodological constraints).

In the 1940s, a few years after the publication of Turing's theory, stored-program digital computers were built. In an article that became very influential, Turing argued that digital computers could be programmed to carry out conversations indistinguishable from conversations with humans (Turing 1950). In explaining digital computers to an audience who was likely to know little about them, Turing used intentional (and hence semantic) language again. He drew an analogy between digital computers and computing humans. The rules followed by a human were analogous to the instructions stored by the computer, and the human process of applying the rules was analogous to the computer's process of executing its instructions: 'It is the duty of the [computer's] control to see that these instructions are obeyed correctly and in the right order' (ibid., 437). Turing's analogy helped explain succinctly what digital computers did, and for that purpose, there was nothing objectionable to it. But it had the potential to suggest that computers somehow understood and obeyed instructions similarly to how people understood and obeyed instructions. This analogy, by no means limited to Turing's writings, was a likely source of the semantic view of computation.[6] When combined with CTM, the semantic view of computation would be used to generate theories that seemed to explain mental contents.

The modern form of CTM was formulated by Warren McCulloch in the 1930s and published by him in the 1940s.[7] McCulloch held that the

---

[6] For example, later we will examine writings by Jerry Fodor, who was an important early proponent of the semantic view of computation. In his early work on this matter, Fodor maintained that computational descriptions are semantic, and the primary reason he gave was that computers 'understood' and 'executed' instructions (Fodor 1968a; 1968b, 638). Only later did he add that computers operated on representations (Fodor 1975).

[7] CTM is often attributed to Turing (e.g., by Fodor 1998). Although Turing occasionally wrote that the brain was a computer (see the essays collected in Ince 1992), his statements to that effect were made after McCulloch's theory, which Turing knew about, had been published (McCulloch and Pitts 1943). I don't know of any place where Turing stated that thinking is computation. In fact, Turing denied that 'intelligence' and 'thinking' were theoretically useful concepts (Turing 1950). McCulloch, however, explicitly held the view that thinking was computation (see

brain was a computing mechanism and that thinking was computation. He also argued that CTM explained the possibility of human knowledge and solved the mind-body problem. During the 1940s, CTM was adopted and elaborated by Norbert Wiener, John von Neumann, and other members of the newly forming cybernetics community, one of whose goals was to explain the mind by building computational models of the brain. During the 1950s, students and younger colleagues of McCulloch, Wiener, and von Neumann turned CTM into the foundation of the new discipline of artificial intelligence, whose goal was to explain the mind by programming computers to be intelligent.[8]

Early computationalists described computers and neural mechanisms using semantic language. For instance, they said that computers (either neural or artificial) 'manipulated numbers,' suggesting that something in the computer meant or represented numbers:

> Computing machines are essentially machines for recording numbers, operating with numbers, and giving the result in numerical form (Wiener 1948, 137).
> Existing computing machines fall into two broad classes: ''analog'' and ''digital.'' This subdivision arises according to the way in which the numbers, on which the machine operates, are *represented* in it (von Neumann 1958, 3; emphasis added).
> Thus the nervous system appears to be using a radically different system of notation from the ones we are familiar with in ordinary arithmetics [sic] and mathematics: instead of the precise systems of markers where the position — and the presence or absence — of every marker counts decisively in determining the *meaning* of the message, we have here a system of notations in which the *meaning* is conveyed by the statistical properties of the message (von Neumann 1958, 79; emphasis added).[9]

Early computationalists talked about computers having content, but they did not discuss how this was possible. They were concerned with building machines that exhibited intelligent behavior, not with philosophical issues about content. Accordingly, they did not address the problem of computational content explicitly. They did not even say explicitly whether they thought computers and minds had content in the same sense, although their writings give the impression that they thought so. McCulloch, for example, argued that CTM explained human

---

e.g. the essays collected in McCulloch 1965). For more on Turing's views on intelligence, see Piccinini 2003b.

[8] A detailed analysis of McCulloch and Pitts's theory is in Piccinini, forthcoming b. For more details on the early history of computationalism, see Piccinini 2003a, chs. 2 through 6.

[9] For a more comprehensive discussion of von Neumann's views on the computer and the brain, see Piccinini 2002a.

knowledge, which suggests that the content of the computational states postulated by CTM explained the content of knowledge states. But he did not discuss how a computational state acquired its content or how this related to the content of mental states.

Part of the reason for this lack of interest in the problem of content may be due to a certain operationalist spirit that early computationalists shared:

> The science of today is operational: that is, it considers every statement as essentially concerned with possible experiments or observable processes. According to this, the study of logic must reduce to the study of the logical machine, whether nervous or mechanical, with all its non-removable limitations and imperfections. (Wiener 1948, 147)
>
> The problem of giving a precise definition to the concept of ''thinking'' and of deciding whether or not a given machine is capable of thinking has aroused a great deal of heated discussion. One interesting definition has been proposed by A.M. Turing: a machine is termed capable of thinking if it can, under certain prescribed conditions, imitate a human being by answering questions sufficiently well to deceive a human questioner for a reasonable period of time. A definition of this type has the advantages of being operational or, in the psychologists' term, behavioristic. No metaphysical notions of consciousness, ego and the like are involved. (Shannon and McCarthy 1956, v)[10]

This operationalism may have led early computationalists to discount questions about computational or mental content on the grounds that either content could not be operationalized, or it had to be operationalized in non-semantic terms. Be that as it may, early computationalists formulated their CTM using semantic language, but they had no theory of content, and they gave no indication that they thought they needed a theory of content. This is probably the origin of the view that CTM ascribes content to the mind, and that it has something to contribute towards solving the problem of mental content.[11]

---

[10] Notice that Turing 1950 did not offer a definition of intelligence, let alone an operational one (cf. Moor 2001, Piccinini 2000). In reading Turing as offering an operational definition of intelligence, Shannon and McCarthy showed how strong their own operationalist leanings were.

[11] I believe that the cybernetic form of CTM, formulated using semantic language, later spread (in the same semantic form) into AI, psychology, and philosophy. I will not document this thesis here (but see Piccinini, forthcoming a, for its influence on Hilary Putnam).

II Functional Role Semantics

Before discussing computationalism in philosophy, I should mention the theme of content in the philosophy of language. In the 1950s through the 1970s, many philosophers were led to think about mental content by thinking about linguistic content and attempting to formulate a semantic theory of language. By semantic theory, I mean both an assignment of content to language (a semantics) and an account of how language acquires its content. For present purposes, the second question is the important one.[12]

A natural way to account for linguistic content is to say that it comes from the content of the language users' minds. In other words, linguistic content can be explained by postulating that certain mental states of the language users have appropriate contents that are transferred to the speakers' utterances. But postulating mental content as an explanation for linguistic content calls, in turn, for a theory of mental content.

A possible solution to the problem of mental content is a theory that goes back Wilfrid Sellars (1954, 1956, 1961, 1967, 1974). In Sellars's theory, mental states — more specifically, thoughts — were construed by analogy with linguistic sentences, and mental contents were constituted by the relations of individual thoughts to stimuli (linguistic and behavioral), responses (linguistic and behavioral), and other thoughts. Each thought was individuated by its content, and its content was constituted by on the one hand the inputs and other thoughts that elicited that thought, and on the other hand the outputs and other thoughts that were elicited by it.[13] Because of its reliance on inner linguistic episodes, Sellars's theory was said to postulate a 'language of thought' (Harman 1970, 404). Different authors describe the role played by mental states in this sort of theory as either functional, or inferential, or conceptual, and the resulting theory of content is correspondingly called functional, or inferential, or conceptual role semantics.

---

[12] On the history of philosophical discussions on content in American philosophy from the 1950s on, including many themes that I have no room to mention here, see Harman 1968, 1988; and especially Dennett 1987, ch. 10.

[13] Ludwig Wittgenstein (1953) offered a similar theory of linguistic content, without extending it to the content of mental states. Sellars's view appears to have originated independently of Wittgenstein's. In his 'Autobiographical Reflections,' Sellars traced his functionalism about content back to reflections that he made in the 1930s and the subsequent influence of Immanuel Kant on him (Sellars 1975, 285-6; see also Sellars 1974, 463).

With functional role semantics in place, we can go back to computationalism.

III Computationalism and the Philosophy of Mind

Computation became an important notion in contemporary philosophy of mind through work by Hilary Putnam and his student Jerry Fodor in the 1960s (Putnam 1960, 1963, 1964, 1967a, 1967b; Fodor 1965, 1968a, 1968b).[14]

Putnam was familiar with some of the cybernetics literature, including McCulloch's CTM,[15] and like the cyberneticians, he did not seem concerned with formulating a theory of mental content. In the first paper where he drew an analogy between minds and TMs (Putnam 1960), Putnam introduced computational descriptions to *dissolve* the mind-body problem. He argued that a problem analogous to the mind-body problem arose for TMs, and that this showed the mind-body problem to be a purely 'verbal' or 'linguistic' problem. In the same paper, Putnam said that internal states of TMs were individuated by their functional relations to inputs, outputs, and other internal states. Two years later (Putnam 1967a, delivered to the Wayne State University Symposium in the Philosophy of Mind in 1962), Putnam argued that mental states were individuated in the same (functional) way that TM states were, but left open the question of whether minds were TMs or something more complicated. In this occasion, he offered his analogy between minds and TMs as a solution to (as opposed to a dissolution of) the mind-body problem. Putnam's mind-body doctrine was going to be called (mental state) functionalism. In its canonical formulation, called *computational* functionalism, it stated that minds *were*, in fact, a kind of TM (Putnam 1967b).

---

[14] The relationship between Putnam and Fodor's computationalism and their mental state functionalism, which pertains only indirectly to the present topic, is explored in more detail in Piccinini, forthcoming a.

[15] McCulloch and Pitts's theory is cited both in Oppenheim and Putnam 1958 and in Putnam 1964. So Putnam knew McCulloch and Pitts's theory before moving to MIT (where McCulloch was) in 1961. During the early 1960s, both Putnam and Fodor were at MIT, which was perhaps the main center of cybernetics research as well as the home institution of Noam Chomsky, who was proposing to explain the human ability to manipulate language by postulating innate knowledge of a recursive (i.e. computational) grammar (Chomsky 1957, 1965). At that time, both Putnam and Fodor were close to Chomsky and his views (Putnam 1997).

Fodor was influenced by Putnam and by psychologists J.A. Deutsch (1960) and Stuart Sutherland (e.g., Sutherland 1960; Fodor 1965, 161, and personal correspondence). One of his goals was an account of psychological explanation. In his first paper on this subject, Fodor wrote that psychological theories were functional analyses, i.e. theories that described a system in terms of internal states and state transitions that were specified by their functional role (Fodor 1965). Later, Fodor added that psychological functional analyses were lists of instructions, or programs, which explained behaviors by stating which internally stored instructions were executed by people engaged in those behaviors (Fodor 1968b).[16]

Although in these writings Putnam and Fodor did not seem concerned directly with the problem of mental content, Putnam and Fodor's 1960s papers on the metaphysics of the mind overlapped considerably with Sellars's theory of mental content. The main common theme — expressed by different authors using different terminologies — was that mental states were to be individuated by their functional relations within a network of inputs, outputs, and other mental states. Another theme was the use of functionalism to dispense with arguments from the privacy of the mental to some special ontological status of the mental (Sellars 1954; Putnam 1960). Finally, there was the analogy between minds and computers. In this respect, Sellars wrote:

> [The] learning of a language or conceptual frame involves the following logically (but not chronologically) distinguishable phases:
> (a) the acquisition of S[timulus]-R[esponse] connections pertaining to the arranging of sounds and visual marks into patterns and sequences of patterns. (The acquisition of these ''habits'' can be compared to the setting up of that part of the wiring of a calculating machine which takes over once the ''problem'' and the relevant ''information'' have been punched in.)
> (b) The acquisition of thing-word connections. (This can be compared to the setting up of that part of the wiring of the machine which enables the punching in of ''information.'') (Sellars 1954, 333)

Despite the overlap between Sellars's functionalism in the 1950s and Putnam and Fodor's functionalism in the 1960s, there is little evidence that Sellars influenced Putnam and Fodor.

Sellars was already well known and one of his early papers, 'Empiricism and the Philosophy of Mind' (Sellars 1956), was widely read and

---

[16] According to Harman (personal correspondence), this view of psychological theories was influenced by the view of some psychologists, who proposed a similar vision of psychological theories to replace the behaviorist stimulus-response view of psychological theories (Miller, Galanter, and Pribram 1960).

discussed when it came out (Harman, personal correspondence). By the early 1960s, Putnam knew Sellars's paper (personal correspondence), and later named it 'one of the most important papers on [its] topic in recent decades' (Putnam 1974, 445).[17] But Putnam does not recall knowing Sellars's 1954 essay, in which Sellars explicitly defended his functionalism about content (personal correspondence). Although Sellars 1956 did present his view that thoughts were analogous to inner linguistic episodes as well as his functionalist theory of content (e.g., Sellars 1956, 180), functionalism about content was not the primary focus of that essay. So although Putnam eventually learned about Sellars's theory, initially he may not have seen it as a theory of content. As to Fodor, according to him at that time he was not acquainted with Sellars's work (personal correspondence). At any rate, neither Putnam nor Fodor cited Sellars in their 1960s papers.

---

[17] Dennett wrote that '[it] is clear that Putnam['s functionalism has] ... been quite directly influenced by Sellars' (Dennett 1987, 341). Dennett told me he got his sense of Putnam's debt to Sellars during a discussion of Sellars's views with Putnam, a discussion that took place in March 1973 (personal correspondence). Putnam, however, told me he 'arrived at functionalism quite naturally, being at the time both a philosopher and a recursion theorist,' and Sellars 1956 'didn't inspire my functionalism' (personal correspondence). Speaking of his work in the late 1950s, he also wrote as follows (recall that, as we saw in section I, Turing had used mentalistic language to describe his TMs):

> I was in the habit of explaining the idea of a 'Turing machine' [n. omitted] in my mathematical logic courses in those days. It struck me that in Turing's work, as in the theory of computation today, the 'states' of the imagined computer (the Turing machine) were described in a very different way than is customary in physical science. The state of a Turing machine — one may call such states *computational* states — is identified by its role in certain computational processes, *independently* of how it is physically realized. A human computer working with paper and pencil, a mechanical calculating engine of the kind that was built in the nineteenth century, and a modern electronic computer can be in the *same* computational state, without being in the same physical state. I began to apply images suggested by the theory of computation to the philosophy of mind, and in a lecture delivered in 1960 [n. omitted; the lecture was published as Putnam 1960] I suggested a hypothesis that was to become influential under the name *functionalism*: that the mental states of a human being are computational states of the brain (Putnam 1997, 180-1).

Curiously, here Putnam followed a common pattern in the literature on functionalism, which attributes computational functionalism to Putnam 1960 even though Putnam didn't formulate it until Putnam 1967b. Putnam 1960 did formulate an *analogy* between minds and TMs, but also denied that minds could be characterized in functional terms in the way TMs were. Perhaps reading 'Empiricism and the Philosophy of Mind' contributed to Putnam's shift from an anti-functionalist position in his 1960 paper to his later functionalist position, but there is no direct evidence of this.

The development of Putnam and Fodor's functionalism appears to have been largely independent of Sellars's functionalism. If this is true, it helps explain why Putnam and Fodor did not discuss the problem of mental content, did not distinguish between the classical mind-body problem and the problem of mental content, and did not seem to think that after giving their (computational) functionalist solution to the mind-body problem, the problem of mental content remained to be solved.[18]

IV The Semantic View of computation in the Philosophy of Mind

Another reason for ignoring the problem of mental content may be that Putnam and Fodor formulated their functionalist theory of mental states using computational descriptions, and — like the cyberneticians — Putnam and Fodor individuated computational states using semantic idioms.

In the papers cited above, Putnam was ambivalent on computational states and content. In his 1960 paper, he drew an analogy between mental states and TM states, and between introspective reports and TM self-descriptions, but he added that TMs could not be properly said to use a language. Later (Putnam 1967b), he stated that mental states were TM states. Together with the generally shared premise that some mental states have content, which Putnam never rejected, this entailed that at least some TM states had content.

Fodor argued that computational explanations in psychology were intellectualist in Ryle's sense (Ryle 1949). An intellectualist explanation accounted for an intentionally characterized overt behavior by postulating an intentionally characterized internal process. Ryle criticized intellectualist explanations; roughly speaking, he argued that they required the postulation of an internal homunculus to explain the intentionally characterized internal process, thereby generating an infinite regress of homunculi inside homunculi (Ryle 1949). Fodor (1968b) rejected Ryle's criticism of intellectualist explanations on the grounds that computers' activities were characterized in intellectualist terms but involved no

---

[18] A fortiori, they did not discuss whether mental states have their content essentially, a view that is popular nowadays. If one believes that mental states have their content essentially, then one will automatically see Putnam's and Fodor's early functionalism as providing a theory of content. Otherwise, one will interpret their work as offering a theory of the identity conditions of mental states that is neutral about their content (cf. Jackson and Pettit 1988, 388).

infinite regress. Fodor built an explicit view about computation ascription around the idea that computational descriptions ascribed semantic properties. According to Fodor, for something to be a computing mechanism, there must be a mapping between its physical states and certain intellectualistic (hence semantic) descriptions of what it does:

> [A] programming language can be thought of as establishing a mapping of the physical states of a machine onto sentences of English such that the English sentence assigned to a given state *expresses the instruction* the machine is said to be executing when it is in that state. (Fodor 1968b, 638, emphasis added)
>
> Every computational device is a complex system which changes physical state in some way determined by physical laws. It is feasible to think of such a system as a computer just insofar as it is possible to devise some mapping which pairs physical states of the device with formulae in a computing language in such a fashion as to preserve desired *semantic relations* among the formulae. (Fodor 1975, 73; emphasis added)

These passages show that for Fodor, computational descriptions ascribed semantic properties to a mechanism and individuated its states by reference to those semantic properties. This was the semantic view of computation.

The semantic view is not a theory of computational content. A theory of content explains how something acquires its content. The semantic view of computation is simply the view that computational states do have content (essentially) — that describing something as a computing mechanism is a way of ascribing content to it. The semantic view of computation does not specify by virtue of which properties or conditions computational states acquire their putative content.[19]

The blending of computational functionalism, CTM, and the semantic view of computation culminated in Fodor's Language of Thought (LOT) hypothesis and his famous slogan 'no computation without representation' (Fodor 1975). According to Fodor 1975, learning what a predicate in a public language meant required representing the semantic properties of that predicate (e.g., the predicate's extension) in some previously understood language — LOT. Now, if understanding LOT

---

[19] In order to have a theory of content, at the very least one needs to add to the semantic view of computation an interpretational semantic theory, according to which all there is to having content is being 'appropriately' described as having content. As an anonymous referee has pointed out to me, the above passages from Fodor 1968b and 1975 could be read as implicitly suggesting such an interpretational semantics. Given Fodor's realism about content, he would probably reject such a reading, but this is beside the point. Interpretational semantics is discussed in section VII below, and Fodor's alternative to it is discussed in section VIII.

required representing the semantic properties of *its* predicates in some previously understood language, this would lead to infinite regress. Fodor blocked this infinite regress by appealing to stored-program computers and the way they responded to their inputs and instructions.[20] Modern computers received data and instructions as inputs written in some programming language and then transformed those inputs into machine language code that they could execute. But executing code did not require a new transformation of internal code into another language, or to possess a representation of how to carry out the execution. Computers were hardwired to carry out certain elementary operations in response to certain lines of internal code, so the regress stopped at those hardwired processes (Fodor 1975, 65ff.).

Fodor likened human public languages to high level programming languages, and the human LOT to a computer's machine language. He argued that LOT was our best explanation for human cognition, and specifically for the human ability to manipulate language and make inferences in a way that respected the semantic properties of thoughts.[21] As we saw above, Fodor described computers and their languages using semantic idioms, perhaps in part because his appeal to stored-program computers was intended to render LOT mechanistically intelligible. So although LOT explained the content of language and mental states in terms of the content of LOT expressions, it did not include a theory of the content of the LOT expressions themselves. To be hardwired to execute certain lines of code was a very interesting property that stored-

---

[20] Fodor also argued that LOT was innate. This component of Fodor's version of LOT, which was rejected by many who accepted LOT, is irrelevant to the present discussion.

[21] This was one more similarity with Sellars's ideas, which included the postulation of inner linguistic episodes as an explanation for thought (esp. Sellars 1956). Sellars's ideas were turned into a systematic theory by Harman (1973, discussed below). Fodor told me he learned about Sellars's and Harman's theory of thought only after writing his 1975 book (personal correspondence). He added that around the same time Zeno Vendler also wrote a book that proposed a similar theory of thought (Vendler 1972). Although Vendler preferred not to talk of an inner 'language' (ibid., 42, 51), his theory postulated an innate neural 'code' that could be scientifically deciphered (ibid., 142). According to Vendler, he developed his theory by trying to improve on Austin's theory of illocutionary acts under the influence of Chomskian linguistics (ibid., viii, 4). Vendler did not refer to Sellars's theory of thought. Although Fodor told me he recalls no mutual influences on their respective versions of LOT between himself, Harman, and Vendler (personal correspondence), Fodor did cite Vendler 1972's arguments as relevant to Fodor's own argument for LOT (Fodor 1975, 58, n. 4), and Harman remembers that Vendler attended a presentation of Harman's version of LOT in 1968 (personal correspondence).

program computers had, but it could not be identified with having content, much less having content corresponding to the content of human language and thought, without argument. Fodor 1975 did not address how the process of translation between public language and LOT respected the semantics of the public language, namely how LOT acquired *its* semantics and managed to match it with the semantics of the public language. Unlike Sellars's LOT, Fodor's LOT offered no solution to the problem of content, nor did he purport to offer such a solution.[22]

During the 1970s, CTM became very influential in philosophy of mind. Many philosophers accepted some version of CTM, even though some of them rejected one or another tenet of Fodor's LOT. At the same time, we will see that the main authors who discussed CTM sympathetically subscribed to the semantic view of computation. They thought the mind was computational, and computational states had content, so they thought the problem of mental content might be reducible to the problem of computational content. And since computing mechanisms were mechanisms built by humans, the problem of computational content may have seemed less philosophically pressing than the problem of mental content. Nevertheless, solving the problem of mental content requires combining CTM with a theory of content.

V Computationalism and Theories of Content

In the rest of this paper, I will argue that computation ascription alone is insufficient for the ascription of mental content. If there were consensus about what mental content is, I could run a general argument of the following form:

> Premise 1. Having mental content is the same as satisfying condition C.
> Premise 2. Being a computational state does not entail satisfying condition C.

---

[22] Nor did he make explicit that his theory presupposed such a solution. I insist on this because LOT has been sometimes mistaken for a theory of content. For example, Putnam criticized Fodor's LOT as if it were a theory of content (Putnam 1988, 21, 40-1). Perhaps this says something about how Putnam was thinking about computation and content. I took this reference to Putnam from Loewer and Rey, who also point out that LOT is not a theory of content (Loewer and Rey 1991, xix). The reason why Fodor's LOT is misread as a theory of content may be partially due to how Fodor blended it with the semantic view of computation.

> Conclusion: Being a computational state does not entail having mental content.

Since there is no consensus on what mental content is, I will go through the main theoretical approaches to content and argue that in each case, ascribing any kind of content to computational states presupposes a non-semantic individuation of the computational states.

A consequence is the rejection of the thesis that CTM contributes to solving the problem of mental content, which eliminates this as a reason for believing CTM. My conclusion has no consequences on whether minds or computing mechanisms have content, whether mental and computational content are the same, and the project of reducing mental content to computational content. All I argue is that those questions must be answered by a theory of content, not by a theory of computation or a CTM.

VI CTM meets Functional Role Semantics

Among computationalist philosophers, the first who took the problem of mental content seriously was probably Gilbert Harman. Harman was familiar both with Putnam and Fodor's computational functionalist writings and with Sellars's theory of content (Harman 1968, 1970).[23] His idea was to explicitly combine computational functionalism about mental states with a Functional Role Semantics (FRS) about their content, so as to have a naturalistic theory of semantically individuated mental states. According to FRS, the content of mental states was constituted by their functional relations. But FRS, as Sellars left it, did not specify a mechanism that could physically realize those functional relations. From a naturalistic perspective, a FRS for mental states called for a theory of the mechanism realizing the functional relations. Computational functionalism had a mechanism that seemed to have the properties needed to realize the relevant functional relations — after all, at least under the semantic view of computation, computing mechanisms draw *inferences* — while at the same time lacking a theory of the mechanism's content.

By combining the two, Harman (1973) could use the one theory to solve the problem left open by the other, and *vice versa*. On one hand, he appealed to the roles of computational states within a computing mechanism as appropriate realizers of the functional relations that ascribed

---

[23] In personal correspondence, Harman has added to these the influences of Miller, Galanter, and Pribram 1960; and Geach 1956.

content to mental states (ibid., 43-8). On the other hand, he appealed to those functional relations to ascribe content to the states of the mechanism (ibid., 60). The result was a CTM in which the computational states and processes that constituted the mind were also the physical realizers of the functional relations that gave the mind its content. In the same theory, Harman also maintained Sellars's construal of thoughts as analogous to linguistic sentences, i.e., as internal representations.

After Harman's revival of it, FRS found many advocates. Some accepted Harman's combination of FRS and a representational CTM (e.g., Field 1978). In a similar vein, Ned Block argued that FRS was the best available theory of mental content to combine with LOT (Block 1986). Others took from Harman only FRS and a functional individuation of mental states, while discarding LOT (e.g., Loar 1981). And Paul Churchland, a student of Sellars, developed a version of FRS that was not conjoined with any form of CTM (Churchland 1979), although later Churchland embraced a connectionist version of CTM (Churchland 1989).

For present purposes, it is important to understand the division of labor in the combined CTM-FRS theory. The content of a mental state comes from its functional relations, so whether the component of the theory that accounts for mental content is successful or not depends on whether functional relations are adequate to provide the semantics of mental states. The functional relations of a mental state are (partially) individuated by the computational relations that the state bears to other states, inputs, and outputs.[24] So, the computational states and relations must be individuated in a way that does not presuppose their content, otherwise the theory of content becomes circular. In other words, a combination of CTM and FRS individuates content by functional relations, and functional relations (at least in part) by computational relations. If the computational relations are individuated by appeal to content, as the semantic view of computation would have it, then CTM-FRS is running in a small circle. If CTM-FRS wants to avoid circularity, it needs a non-semantic view of computation, i.e. a way to individuate

---

[24] CTM-FRS theorists do not individuate *content* itself with computational roles, however. Some, like Harman, postulate that the relations between inputs and outputs on one hand, and the environment on the other, also contribute to the relevant functional relations and hence to content (broad FRS); others, like Block, postulate that functional relations are only one of the factors determining content, the other being reference (narrow FRS). These subtleties make no difference for our purposes; see Block 1986 for discussion.

computational states and their relations without appeal to their content.[25]

As natural as it may seem to combine CTM and FRS, it is not mandatory. The two are logically independent. Even if one believes that content is constituted by functional relations, the physical realization of the functional relations need not be computational.[26] And even if one believes that the mind is computational and that computational states have content, one need not believe that content is constituted by the functional relations of the computational states. For a first important alternative theory of content for minds and computing mechanisms, we will look at the tradition that follows Daniel Dennett.

VII CTM meets Interpretational Semantics

Like Harman, Dennett was familiar with both computational functionalism and some of Sellars's work, and he was interested in the problem of mental content. Dennett also continued the tradition of his mentor Gilbert Ryle, whose analytical behaviorism rejected the postulation of semantically characterized mental states (Ryle 1949). Following Sellars and Putnam, Dennett did develop a version of functionalism about content, according to which mental content was constituted by the mutual relations between contentful states, inputs, and outputs (Dennett 1969, ch. 4).[27] But Dennett's functional relations between contents, unlike

---

[25] I don't know of any place where CTM-FRS theorists have offered such an account. Block (1986) says he doesn't know how to individuate functional relations. To a direct question, Harman answered thus: 'I don't think I have ever tried to provide a noncircular theory of content in that sense' (personal correspondence).

[26] To say more about what a non-computational realization of functional relations would be would take us too far afield. Given my account of computing mechanisms in Piccinini 2003a, ch. 10, FRS could be combined with a non-computational functional analysis of the mind-brain, i.e. a functional analysis that explains cognitive processes without ascribing computations to the mind-brain. For versions of FRS that are not committed to computational roles as physical realizers of functional relations, see Peacocke 1992, and Brandom 1994.

[27] It seems that Dennett's functionalism about content was more indebted to Putnam's computational functionalism than to Sellars's FRS. Dennett put it as follows:

> I had read some Sellars when I finished Content and Consciousness [Dennett 1969], but I hadn't thought I understood it very well. Several of his students had been in Oxford with me, and had enthused over his work, but in spite of their urging, I didn't 'become a Sellarsian.' I'd read all of Putnam's consciousness papers (to date), and was definitely influenced strongly by Putnam. One of Sellars'

Harman's, were not realized by computational roles (Dennett 1969, 1971).

Following Ryle, Dennett argued that explaining content by postulating contentful mental states was tantamount to postulating a homunculus who understood the content of the mental states. This, however, either begged the question of how the homunculus understands content, or led to the infinite regress of postulating homunculi inside homunculi. Dennett argued that mental states and their contents came from the external observer of a system; they were ascribed to people, not discovered in them, by describing and predicting the behavior of a system from what Dennett called the *intentional stance*. According to Dennett, people ascribed contentful states, e.g. beliefs and desires, to each other in order to predict each other's behavior, but inside people's brains there was nothing that realized those contentful states in an exact way, as Harman's computational states were supposed to do (Dennett 1987, 53, 71). If a system's behavior was sufficiently complex and adaptive and an external observer did not know its internal mechanisms well enough to derive its behavior mechanistically, then the observer interpreted the system as possessing beliefs and desires about its environment — hence as having contentful states — so as to explain the system's adaptive behavior as the satisfaction of appropriate desires given mostly true beliefs. In summary, an observer ascribed content to the system. So mental content came from the interpretation of external observers.

A few years after Dennett formulated his theory of content, Fodor's LOT hypothesis convinced Dennett that content could be ascribed to internal states of a system, and specifically to states of stored-program computers, without begging the question of how content was understood (Dennett 1978). Dennett explained why the question was not begged by applying his interpretational semantics to the internal states, in a version of his theory that was later called *homuncular functionalism* (ibid.). According to homuncular functionalism, content (and the intelligence needed to understand it) could be ascribed to the internal states

---

students, Peter Woodruff, was a colleague of mine at UC Irvine, and it was he who showed me how my work was consonant with, and no doubt somewhat inspired by, Sellars. But that was after he read C&C. I thereupon sent Sellars one of the first copies of C&C, and he wrote back enthusiastically....

I would think that my sketchy functionalist theory of meaning was more influenced by Putnam's 'Minds and Machines' paper [Putnam 1960] than anything I'd read in Sellars while at Oxford, but I can't be sure. I have sometimes discovered telltale underlinings in my copy of a book years later and recognized that I had been influenced by an author and utterly forgotten it (Dennett, personal correspondence).

of a system without begging the question of how it was understood as long as the ascribed content was ultimately discharged by a completely mechanistic explanation of the behavior of the system. Discharging content ascription mechanistically consisted of decomposing the system into parts and explaining the content and intelligent behavior of the system in terms of the content and intelligent behavior of the parts. In doing this, the parts and their behavior could still be ascribed content, but understanding their content must require less intelligence than the intelligence required by the content ascribed to the whole. The same process could be repeated for the content ascribed to the parts, explaining their content and intelligence in terms of the content and intelligence of their parts, but the parts' parts' intentional descriptions must ascribe less and less content and intelligence to them. The process ended with components whose behavior was so simple and obviously mechanical that it warranted *no* content ascription. Dennett offered his homuncular functionalism as a way to cash out the intentional descriptions commonly used for computers and their programs, especially in the context of artificial intelligence research. Thus, homuncular functionalism explicated the semantic view of computation in a metaphysically non-mysterious way. It dissolved, rather than solve, the problem of computational content.[28]

Dennett's theory went through elaborations and revisions (Dennett 1978, 1987), but its core remained: content, whether mental or computational, came from the interpretations of external observers. Dennett's theory had the great advantage of being equally applicable to organisms and artifacts like computers, giving a common treatment of computational and mental content. This was because both organisms and artifacts were equally subject to intentional interpretation by external observers. However, interpretational semantics also denied the reality of mental content: for Dennett, mental content was an instrument of prediction; as much as the predicted behavioral patterns were objective (Dennett 1987, 15, 25), the states posited by the predictive tool did not correspond to any of the internal states posited by a correct mechanistic explanation of the system whose behavior was being predicted. Put another way, Dennett did not believe in original or intrinsic intentionality (e.g., Dennett 1987, 288). A similar point applied to computational content: when

---

[28] In formulating homuncular functionalism, Dennett also argued that the Church-Turing thesis entailed that a mechanistic theory had to be computational, hence that 'AI is the study of all possible modes of intelligence' (Dennett 1978, 83). I discuss this further component of Dennett's homuncular functionalism, which makes no difference to the present discussion, in Piccinini 2003a, ch. 7.

computational content was discharged by the decomposition of the system into the activity and internal states of purely mechanical (non-contentful) components, computational content was explained away. Furthermore, interpretation was somewhat indeterminate: in principle the same behavior by the same system could be interpreted in different and equally adequate ways (e.g., Dennett 1987, 40, ch. 8). Dennett's theory explained content at the cost of deeming it unreal.

A corollary of Dennett's conjunction of the semantic view of computation and interpretational semantics is interpretationism about computation: whether something is a computing mechanism is a matter of interpretation not fact. From the perspective of Dennett's intentional stance, there is no principled difference between a desktop computer and a refrigerator, except that applying the intentional stance to the computer is more useful than applying it to a refrigerator. This leads to the paradoxical effect that Dennett's intentional stance, which seems to account so well for computational content, does not account for our practice of applying the term 'computer' only to some machines, which seem to belong to a special class distinct from other machines. Given Dennett's theory, in order to explain the difference between computing mechanisms and other machines, one must abandon the intentional stance and take some other stance (perhaps the design stance) towards the mechanisms. A similar problem arises in comparing different interpretations of the same computation. Given that for Dennett interpretation is partially indeterminate, there may be two equally adequate computational interpretations of a process. What they have in common, then, cannot be expressed from within the intentional stance: one needs to leave the intentional stance and resort to some other stance. It turns out that if Dennett wants to explain the difference between computing mechanisms and other mechanisms, or explain what two adequate interpretations of a computation have in common, he needs to individuate computing mechanisms and their states using non-semantic language.

Dennett's theory of content was very successful among philosophers interested in CTM. For example, John Haugeland used a version of Dennett's homuncular functionalism as an explication of the research program, pursued by many in psychology and AI, of developing a CTM (Haugeland 1978, 1985, 1997). For Haugeland, like for Dennett, the content ascribed by a computational theory of a system, including a CTM, came from the theorist's interpretation of the system.

The author who elaborated Dennett's theory of content in the most sophisticated and systematic way was perhaps Robert Cummins. Cummins built a theory of computational explanation (1983), which included a theory of computational content (1983, 1989), drawing from both Dennett and Haugeland's writings. Like Dennett and Haugeland's theories, Cummins's theory was squarely based on the semantic view of computation and explained content in terms of how an external observer interpreted a system.

Unlike Dennett, who was frankly anti-realist about mental content, Cummins took a more realist position (1983, 74-5). Cummins sharply distinguished between mental content and computational content. He argued that in formulating a CTM, psychologists and AI researchers postulated (contentful) computational states as explanations for people's cognitive capacities, and offered his interpretational theory as an account of the content of computational states postulated by CTM theorists. But Cummins also argued that CTM fell short of explaining genuine mental content, or as he put it, the intentionality of mental capacities. He discussed five strategies to explain mental content in terms of computational content and argued that they all failed (1983, 91-100). Cummins still stated that intentionality was somehow going to be accounted for in terms of computation, but he added that he had no idea how this could be done (1983, 89-90). He denied that he had a theory of mental content (1989). So, Cummins's interpretational theory of content did nothing to solve the philosophical problem of mental content with which we are presently concerned (nor was it intended to do so). Moreover, Cummins's theory entailed that the same system could be interpreted in different ways that ascribed different computations.

Another author in the interpretational semantics tradition is Patricia Churchland. In her book written with neuroscientist Terrence Sejnowski, she offered a detailed account of the brain as a computing mechanism, predicated on the semantic view of computation. In explicating what it meant to be a computer, Churchland and Sejnowski stated an informal version of Cummins's theory (with a reference to a paper by Cummins for more details; Churchland and Sejnowski 1992, 65). Unlike Cummins, Churchland and Sejnowski did not discuss explicitly what notion of content was at stake in their theory. Perhaps because of this, some authors have accused Churchland and Sejnowski of being ambiguous as to whether or not they ascribed genuine mental content to brains (Grush 2001). In light of the preceding discussion of interpretational semantics, I offer a more charitable reading. Churchland and Sejnowski were working with an interpretational semantics, which had the same advantages and disadvantages of all interpretational semantics. On the one hand, it applied equally well to organisms and artifacts. On the other hand, it did not solve the problem of mental content. Churchland and Sejnowski may choose whether to side with Dennett or Cummins: either they accept Dennett's anti-realism about mental content, so that their interpretational semantics explains mental content away (with Dennett), or they endorse Cummins's realism about mental content, but then they must defer to some other theory as far as mental content is concerned.

Given that at least Churchland is in print denying the existence of intrinsic intentionality (Churchland and Churchland 1983), presumably she would side with Dennett's anti-realism about content.

In conclusion, interpretational semantics is a natural and attractive way to cash out the semantic view of computation, but it comes at the cost of losing the ability to explain how computing mechanisms differ from other mechanisms, and what two distinct but equally adequate computational interpretations of a process have in common. In order to regain these abilities, an interpretational semanticist needs a non-semantic way to individuate computational states. In addition, interpretational semantics makes computational descriptions, even construed as semantic descriptions, insufficient to characterize mental content as a real property of minds. This may not trouble those who don't believe in mental content to begin with, but it leads others to look elsewhere for a theory of content.

VIII CTM meets Informational and Teleological Semantics

Until the mid-1970s, Fodor freely appealed to the semantic view of computation in formulating his version of CTM, without discussing the need for a theory of content. In a series of papers in the late 1970s (collected in Fodor 1981), he became more explicit about the relationship between CTM and mental content. He argued that folk psychology formulated its generalizations by quantifying over the *semantic* properties of propositional attitudes, whereas cognitive (computational) psychology formulated its generalizations by quantifying over the *syntactic* properties of mental representations. He added that the strength of a computational psychology, postulating LOT, was that computational psychology had the resources to reduce the generalizations of folk psychology to scientific generalizations, so as to underwrite our intuitive discourse about contentful propositional attitudes. This was because computations, albeit being causally driven by the syntactic properties of representations, were processes that (could) respect the semantic properties of representations. So, given a computational psychology, the two stories about the mind, cognitive and folk, would eventually match, in the sense that under the appropriate ascription of content to the states posited by cognitive psychology, the relations between semantically individuated folk psychological states would match the causal relationships between cognitive psychological states. What was needed to complete this picture was a theory ascribing the right contents to the computational states: a theory of content (Fodor 1981; see also Fodor 1987, ch. 1).

A theory purporting to do this was FRS, but Fodor rejected it. He wrote a critique of procedural semantics, a theory of content that was popular in AI and psychology. According to procedural semantics, the content of a computational instruction was given by the computational procedures that executed that instruction. Since the relations between an instruction and the computational procedures that operated on it were functional relations, procedural semantics was a version of FRS. Fodor argued that, since the procedures that executed computer instructions were entirely internal to the machine and, when transformed into machine language, were naturally interpreted as referring to the shifting of bits from one register of the machine to another, procedural semantics reduced the content of computational descriptions to content about shifting bits from one register to another, without ever involving anything external to the machine (Fodor 1978). Besides procedural semantics, Fodor also rejected FRS in general, in part because he saw that, given a semantic view of computation, FRS was circular (e.g., see Fodor 1990, ch. 1). Fodor maintained the semantic view of computation and treated content ascription as prior to computation ascription. For him, computations were defined over representations, which were individuated by their content, so computations were individuated by the semantic properties of the representations over which they were defined. Because of this — which was the semantic view of computation — a theory of content could not individuate contents by appealing to the notion of computation, on pain of circularity.[29]

By the end of the 1970s, any computationalist philosopher of mind who — like Fodor — took mental content as a real property of minds but rejected FRS, needed an alternative (naturalistic) theory of content. This demand was soon met by two new approaches, Informational Semantics and Teleological Semantics (ITS). According to ITS, the content of a

---

[29] The most explicit discussion of this point that I know of is in Fodor 1998. He wrote that since his notion of computation presupposed the notion of content, he could not account for content in terms of computation (ibid., esp. 13). He also said explicitly that because of this, his theory of content (unlike his theory of thought) was not computational (ibid., 11). Notice that at least *prima facie*, the semantic view of computation is consistent with the 'formality condition' (Fodor 1980), according to which computational processes are sensitive only to the formal (i.e., non-semantic) properties of representations. The semantic view of computation is about how to individuate computational states, whereas the formality condition is about which properties of computational states are causally efficacious. So, according to Fodor, '[that] taxonomy in respect to content *is* compatible with the formality condition, plus of minus a bit, is perhaps *the* basic idea of modern cognitive theory' (Fodor 1980, 240, emphasis in original).

mental state came from natural relations between that state and the mind's environment. Informational Semantics said that the crucial relations for individuating content were informational, namely relations determining what information was carried by a state (Dretske 1981, 1986). Teleological Semantics said that the crucial relations involved the evolutionary history of the state, namely what the mechanism generating that state was selected for (Millikan 1984, 1993).[30] Following Dretske and Millikan, Fodor developed his own version of ITS (1987, 1990, 1998) with the explicit goal of finding a theory of content for the representations postulated by LOT.

The main difference between FRS and ITS is that while the former is holistic and (partially) internalist, specifying (part of) the content of a state by its relation to other contentful states, ITS is externalist and atomistic, specifying the content of a state independently of its relation to other contentful states. So, ITS specifies the content of a state independently of what computations it enters.

According to Fodor, the combination of ITS and CTM offered our best hope for a scientific theory of mind that would respect folk intuitions about mental content, so that the stories told by cognitive and folk psychology would match. This was because CTM accounted for how mental processes could be causally efficacious while respecting the semantic properties of mental representations, and ITS accounted for how representations got their content (Fodor 1987, 1990, 1998).

Given a theory of content of the ITS form, it should be obvious that being a computational state is insufficient for having mental content. I can run an instantiation of the argument schema described in section V:

> Premise 1. Having mental content is the same as entering certain informational or teleological relations with the environment.
> Premise 2. Being a computational state does not entail entering the relations mentioned in premise 1.
> Conclusion: Being a computational state does not entail having mental content.

Premise 1 is just ITS. Premise 2 expresses the familiar fact that what we ordinarily call computers, and use as computers, are rarely if ever hooked up to the environment in the complicated ways postulated by ITS. Some computing mechanisms, which computer scientists call *em-*

---

[30] Incidentally, Millikan was another student of Sellars. At the beginning of her first book (Millikan 1984), she stated that her theory of content was inspired by some of Sellars's remarks.

bedded systems (e.g., cars' computers and digital thermostats), *are* connected to the environment in ways that resemble those postulated by some versions of ITS, but they are not the typical case. Ordinarily, whether something is a computing mechanism is independent of the ITS relations it bears to the environment.

This conclusion would not come as a surprise to ITS theorists: ITS theorists generally don't ascribe mental content to ordinary computers.[31] This has the important consequence that, to the extent that they accept ordinary computation ascription, ITS theorists who believe in CTM are committed to there being something in common between computing mechanisms that satisfy the demands of ITS for mental content and ordinary (non-embedded) computing mechanisms: although they are all computing mechanisms, some have mental content while others don't. That is, a consequence of conjoining CTM and ITS is that minds and ordinary computing mechanisms have something in common that cannot be specified by ITS. The way to specify it, I submit, is by a non-semantic account of computational states.

To summarize, given ITS the search for a theory of mental content is not by itself a motivation to endorse CTM, because the solution to the problem of mental content is independent of CTM. If anything, it is the search for a theory of contentful mental states that still motivates computationalist philosophers who want to save the semantic view of computation to match CTM with ITS. If they succeed, they find themselves in a position from which they cannot tell what minds have in common with ordinary computing mechanisms. In order to tell, they need a non-semantic way to individuate computing mechanisms and their states.

IX CTM meets Intentional Eliminativism

All the computationalist philosophers discussed until now shared the semantic view of computation. The first who questioned their assumption was Stephen Stich (1983). Like other computationalist philosophers, Stich's primary goal was not to give a philosophical account of computation but a theory of mind; his rejection of the semantic view of computation was an implicit consequence of his theory of mind. Stich was motivated by the belief that folk psychology, including the mental contents it postulated, would be eliminated in favor of a cognitive

---

[31] For a critical discussion of this feature of Dretske and Fodor's view, see Dennett 1987, ch. 8.

psychological theory of mind. *Contra* Fodor, he argued that the generalizations of cognitive psychology would not match those of folk psychology (Stich 1983, chs. 4, 7, and 9).

Stich formulated a version of CTM that did not require mental states to have content, a theory that he called syntactic theory of mind. In order to have a CTM without mental content, Stich implicitly rejected the semantic view of computation. According to him, the mind was computational, but computational descriptions did not ascribe semantic properties. For something to be computational, its physical states must be mappable onto syntactically defined states, without presupposing any semantics. A system was a computing mechanism if and only if there was a mapping between its behaviorally relevant physical states and a class of syntactic types, specified by a grammar that defined how complex types could be formed out of primitive types. According to Stich, the mind was a computing mechanism in this sense (Stich 1983).

By formulating his version of CTM in terms of his syntactic view of computation and doing away with mental content, Stich renounced what Fodor considered the crucial consideration in favor of CTM: the hope that cognitive psychological generalizations ranging over syntactically individuated states would correspond to folk psychological generalizations ranging over contentful propositional attitudes. According to Fodor, the point of CTM was to explain how a mere mechanism could mirror semantic relations by invoking the match between mechanical processes that responded only to syntactic properties (computations) and processes individuated by semantic properties (inferences). From Fodor's point of view, if one followed Stich in denying that mental states had content, it was unclear why and how mental states and their functional relations should be construed as being computational. Perhaps because of this, Stich's syntactic theory of mind won few converts. But Stich argued that a syntactic theory of mind offered the best explanation of mental phenomena: for instance, Stich said that beliefs in the folk sense would likely be identified with some non-contentful, syntactically individuated states that had a functional role similar to the one played by beliefs in folk psychology (Stich 1983, ch. 11). Moreover, Stich argued that his syntactic theory of mind was the best construal of the computational theories of mental phenomena offered by cognitive psychologists.

If Stich's proposal of a syntactic theory of mind is coherent, it shows how CTM can be formulated without the semantic view of computation. This point is independent of Stich's intentional eliminativism. There is an important sense in which Stich's syntactic theory of mind is compatible with the existence of mental content and the Fodorian argument for CTM. Stich's syntactic criterion for mental states can be taken as a way to individuate mental states and processes as computational in a non-semantic way, while leaving open the question of whether they also have content and whether there are generalizations ranging over contentful states that match those formulated over syntactically individuated states.[32]

This is an open possibility so long as we abandon the semantic view of computation, which maintains that computational states are individuated by their content. For if computational states are individuated by their content, it would be impossible to individuate them non-semantically, as Stich's theory requires, and then ask whether they have content and what content they have. From the point of view of the semantic view of computation, Stich's coupling of CTM and intentional eliminativism, according to which mental states are computational but contentless, is incoherent. And from the point of view of Stich's combination of CTM and intentional eliminativism, the semantic view of computation, and any version of CTM that is formulated using the semantic view of computation, begs the question of whether the computational mind has content.

Indeed, Stich's notion of syntax has been challenged on the grounds that it makes no sense to speak of syntax without semantics. According to this line of thought, something can be a token of a syntactic type only relative to a language in which that token has content (e.g., Crane 1990; Jacquette 1991). If this is right, then Stich's proposal is incoherent, but I don't think it is.

The coherence of Stich's proposal is easy to see when we reflect on the functional properties of stored-program computers.[33] Some special mechanisms, namely stored-program computers, have the ability to respond to (non-semantically individuated) strings of tokens stored in their memory by executing sequences of primitive operations, which in turn generate new strings of tokens that get stored in memory. Different bits and pieces of these strings of tokens have different effects on the machine. Because of this, the strings of tokens can be analyzed into sub-strings. An accurate description of how tokens can be compounded into sub-strings, and sub-strings can be compounded into strings, which does not presuppose that the strings of tokens have any content, may be called the syntax of the system of strings manipulated by the computer. Some strings, called instructions, have the function of determining, at any given time, which operations are to be performed by the computer

---

[32] In fact, Frances Egan has advocated the conjunction of a non-semantically formulated CTM, *a la* Stich, with the view that 'computational states have content' (Egan 1999, 181).

[33] For a more detailed account of stored-program computers along these lines, see Piccinini 2003a, ch. 10.

on the input strings. Because of how computers are designed, the global effect of an instruction on the machine can be reduced to the effects of its sub-strings on the machine. Then, the effect of sub-strings on the computer can be assigned to them as their content, and the way in which the content of the whole string depends on the content of its sub-strings can be specified by recursive clauses, with the result that the global effect of a string on the computer is assigned to it as its content. This assignment constitutes an *internal* semantics of a computer. An internal semantics assigns as contents to a system its own internal components and activities, whereas an ordinary (external) semantics assigns as contents to a system objects and properties in the system's environment.[34] Given that the strings manipulated by a computer may have a syntax (which determines how they are manipulated), and some of them have an internal semantics, they may be called a language, and indeed that is what computer scientists call them. None of this entails that computer languages have any external semantics, i.e. any content in the sense used by Stich's critics, although it is compatible with their having one. Stich may be construed as arguing that the functional organization of the mind is similar to that of a stored-program computer, so that the mind contains a system of strings of tokens with a syntax analogous to that of the strings manipulated by stored-program computers. Stich would probably have no difficulty in accepting that if the brain is capable of storing and executing its own instructions, then some of the mental strings also have an internal semantics.

The above account of syntax is functional, specified in terms of the components of a stored-program computer, their states, and their interactions. From the vantage point of this functional view of computation, not only do we see the coherence of Stich's proposal, but we can also give a functional account of his notion of syntax without presupposing any external semantics.

Stich's proposal shows that one can be a computationalist without having a theory of content and while rejecting the semantic view of computation, because one can be a computationalist without believing in mental content at all. A computationalist who wishes not to beg the question against the intentional eliminativist should formulate CTM without the semantic view of computation, and independently of any theory of content.

---

[34] For more on internal vs. external semantics, see Fodor 1978; Dennett 1987; and Piccinini 2003a, chs. 9 and 10.

X CTM With or Without Semantics

The first moral of this paper is that CTM was originally conceived in tandem with the semantic view of computation, and this convinced a lot of philosophers that CTM necessarily ascribed content to the mind. Between the 1940s and the late 1960s, both in science and philosophy, computationalists promoted CTM not only as offering a mechanistic explanation of the mind but also, by construing computational description semantically, as offering the beginning of a naturalistic account of content.

In the 1970s, it became clear that CTM *per se* offered no solution to the problem of mental content. The ensuing investigations of a theory of content revealed four main ways to combine CTM with a theory of content. The first combines CTM with Functional Role Semantics (FRS), which sees mental content as (partially) reducible to the computational relations among mental states. This option cannot construe computational relations semantically on pain of circularity, and hence it presupposes a non-semantic way of individuating computational states. The second combines CTM with interpretational semantics, which sees mental content as a tool for predicting behavior. This option maintains the semantic view of computation at the cost of denying the reality of mental content. The third combines CTM with Informational or Teleological Semantics (ITS), which sees content as reducible to a combination of causal and counterfactual relations between mental states and the environment. This option maintains the semantic view of computation at the cost of being inapplicable to ordinary computing mechanisms, because most ordinary computing mechanisms don't enter the causal and counterfactual relations postulated by ITS. The fourth combines CTM with intentional eliminativism, which abandons the semantic view of computation in favor of a non-semantic construal of computation.

These options seem to exhaust the possibilities that are open to the computationalist: either mental content comes from the computations themselves (FRS), or it comes from some non-computational natural properties of the content-bearing states (ITS), or it is in the eye of the beholder (interpretational semantics), or there is no mental content at all (eliminativism). Under any of these options, either the problem of mental content is solved by something other than computation ascription, or computation ascription must be construed non-semantically, or mental content is unreal. Usually more than one of the above is true of each option.

For each of the above theories of content, neither the theory of content entails CTM nor CTM entails the theory of content. None of the existing theories of mental content offers a reason to endorse CTM. Whether the mind is computational and whether the mind has content (and how it manages to have content) are different problems that need to be solved independently of each other. The semantic view of computation also begs the question of the FRS and intentional eliminativist theorists, who need a non-semantic individuation of computational states in order to formulate their views. In order to keep the two problems separate, we should avoid formulating CTM as a theory that ascribes content to the mind, as it is often done (e.g., by Fodor 1998 and Horst 2003). Even those who are skeptical about full-blown mental content but believe in some form of computational content (e.g., Churchland and Sejnowski 1992) should avoid formulating CTM as a theory that ascribes content to the mind. CTM should be formulated in a way that is neutral about content, leaving it to considerations about content to determine which theory of content is correct. CTM is a theory of the internal mechanisms of the mind or brain, which may or may not explain some mental phenomena. So, the semantic view of computation should be abandoned in favor of the functional view of computation, and CTM should be formulated without using semantic language. Fortunately, as I argue on independent grounds in Piccinini 2003a, chs. 9 and 10, this is also the best way to understand computing mechanisms in their own right. Stich is right in one important respect: in order to understand computing mechanisms and how they work (as opposed to why they are built and how they are used), there is no need to invoke content; it's actually misleading to do so.

Construing CTM without the semantic view of computation leaves the field entirely open for different positions about content. Perhaps some computational states have content and some don't; perhaps all do or none do. Perhaps some have content in one sense and not in others. CTM should not be seen as answering any of these questions. If the mind is computational but has no content, then CTM will explain the mind without requiring a theory of content. If the mind does have content, then this is going to be explained by a theory of content. If mental states are both contentful and computational, then the true version of CTM and the true theory of content will be compatible with each other. One example of compatibility is offered by combining a non-semantically formulated version of LOT with ITS; this is analogous to Fodor's view minus his semantic view of computation. Another example is the combination of a non-semantically formulated version of CTM and interpretational semantics; this is analogous to Dennett's view minus his semantic view of computation. A third example is the conjunction of a non-semantically formulated version of CTM with FRS.

XI Two Consequences

If questions of content are independent of questions of computation, there are some consequences that deserve to be explored. I will briefly mention two:

1. During the last two decades, it has become common to hear criticisms of CTM based on the rejection of representationalism (Brooks 1997; Thelen and Smith 1994; van Gelder 1995; and certain passages in Clark 1997). According to these criticisms, some or all mental phenomena can be explained without postulating contentful mental states, and therefore CTM should be rejected. As I've tried to show, many computationalists have endorsed the semantic view of computation, and therefore their position is vulnerable to this criticism. But I also argued that the semantic view of computation should be rejected. If this is done, then the anti-representationalist critique of CTM turns out to be confused in the same way that the semantic view of computation is. Even if we don't need representations to explain cognition (which I doubt), this would do nothing to undermine CTM *per se*, but only the combination of CTM with representationalism. CTM can and should be formulated independently of any theory of content, which makes it invulnerable to anti-representationalist critiques.

2. Above, I mentioned Fodor's argument according to which CTM is our best theory of mind because, by postulating causal processes that can mirror semantic relations between representations, it offers the hope to generate a scientific theory of mind close to our folk theory of mind. (Fodor has forcefully made this argument in conjunction with ITS, but a version of it could be run in conjunction with FRS.) However, if we accept that the question of computation is independent of the question of content, it becomes clear that this argument is missing a crucial premise. Before we accept that CTM has the potential to match contentful relations with computational processes, we should ask by what mechanism this match is achieved. In other words, we need to conjoin CTM not only with a theory of content, but also with a theory of how the computational relations get to match the semantic properties of the internal states. Notice that the mechanism that accomplishes the matching cannot be computational on pain of circularity. For if it were a computing mechanism, we should ask whether *its* computational processes match *its* semantic properties. If they don't, then it is unclear how such a mechanism could achieve the syntax-semantics match in the first mechanism. If they do, we need to answer the question of how they do, and we are back where we started. So the matching must be done by a non-computational mechanism. What mechanism is it and how does it work? At some point, Fodor formulated a problem very similar to this, which he called the 'coordination problem,' and argued that it's solvable (Fodor

1994, 12ff., 86). More recently, he has come close to admitting that he doesn't know how to solve this problem (see Fodor 2000, esp. 71-8). Without a solution, his argument for CTM doesn't go through. This may be one of the reasons for his skepticism that CTM is going to offer a complete explanation for the (non-conscious aspect of the) mind (Fodor 2000).

Received: October 2003
Revised: March 2004

References

Block, N. 1986. 'Advertisement for a Semantics for Psychology,' in *Midwest Studies in Philosophy X: Studies in the Philosophy of Mind*, P. French, T.E. Uehling, jr. and H.K. Wettstein, eds. Minneapolis: University of Minnesota Press.

Brandom, R.B. 1994. *Making it Explicit: Reasoning, Representing, and Discursive Commitment*. Cambridge, MA: Harvard University Press.

Brooks, R.A. 1997. 'Intelligence without Representation,' in *Mind Design II*, J. Haugeland, ed. Cambridge, MA: MIT Press.

Chomsky, N. 1957. *Syntactic Structures*. The Hague: Mouton.

______. 1965. *Aspects of a Theory of Syntax*. Cambridge, MA: MIT Press.

Churchland, P.M. 1979. *Scientific Realism and the Plasticity of Mind*. Cambridge: Cambridge University Press.

______. 1989. *A Neurocomputational Perspective*. Cambridge, MA: MIT Press.

Churchland, P.M. and P.S. Churchland. 1983. 'Stalking the Wild Epistemic Engine.' *Noûs* **17** 5-18.

Churchland, P.S. and T.J. Sejnowski. 1992. *The Computational Brain*. Cambridge, MA: MIT Press.

Clark, A. 1997. *Being There*. Cambridge, MA: MIT Press.

Crane, T. 1990. 'The Language of Thought: No Syntax Without Semantics.' *Mind and Language* **5** 187-212.

Cummins, R. 1983. *The Nature of Psychological Explanation*. Cambridge, MA: MIT Press.

______. 1989. *Meaning and Mental Representation*. Cambridge, MA: MIT Press.

Dennett, D.C. 1969. *Content and Consciousness*. London: Routledge and Kegan Paul.

______. 1971. 'Intentional Systems.' *Journal of Philosophy* **68** 87-106.

______. 1978. *Brainstorms*. Cambridge, MA: MIT Press.

______. 1987. *The Intentional Stance*. Cambridge, MA: MIT Press.

Deutsch, J.A. 1960. *The Structural Basis of Behavior*. Chicago: University of Chicago Press.

Dretske, F.I. 1981. *Knowledge and the Flow of Information*. Cambridge, MA: MIT Press.

______. 1986. 'Misrepresentation,' in *Belief: Form, Content, and Function*, R. Bogdan, ed. New York: Oxford University Press.

Egan, F. 1999. 'In Defence of Narrow Mindedness.' *Mind and Language* **14** 177-194.

Field, H. 1978. 'Mental Representation.' *Erkenntnis* **13** 9-61.

Fodor, J.A. 1965. 'Explanations in Psychology,' in *Philosophy in America*, M. Black, ed. London: Routledge and Kegan Paul.

______. 1968a. *Psychological Explanation*. New York: Random House.

______. 1968b. 'The Appeal to Tacit Knowledge in Psychological Explanation.' *Journal of Philosophy* **65** 627-40.

______. 1975. *The Language of Thought*. Cambridge, MA: Harvard University Press.

______. 1978. 'Tom Swift and His Procedural Grandmother.' *Cognition* **6**. Reprinted in Fodor 1981.

______. 1980. 'Methodological Solipsism Considered as a Research Strategy in Cognitive Psychology.' *Behavioral and Brain Sciences* **3**. Reprinted in Fodor 1981.

______. 1981. *Representations*. Cambridge, MA: MIT Press.

______. 1987. *Psychosemantics*. Cambridge, MA: MIT Press.

______. 1990. *A Theory of Content and Other Essays*. Cambridge, MA: MIT Press.

______. 1994. *The Elm and the Expert: Mentalese and Its Semantics*. Cambridge, MA: MIT Press.

______. 1998. *Concepts*. Oxford: Clarendon Press.

______. 2000. *The Mind Doesn't Work That Way*. MIT Press: Cambridge, MA.

Geach, P.T. 1956. *Mental Acts*. London: Routledge & Paul.

Grush, R. 2001. 'The Semantic Challenge to Computational Neuroscience,' in *Theory and Method in the Neurosciences*, P. Machamer, R. Grush and P. McLaughlin, eds. Pittsburgh, PA: University of Pittsburgh Press.

Harman, G. 1968. 'Three Levels of Meaning.' *Journal of Philosophy* **65** 590-602.

______. 1970. 'Sellars' Semantics.' *The Philosophical Review* **79** 404-419.

______. 1973. *Thought*. Princeton: Princeton University Press.

______. 1988. 'Wide Functionalism,' in *Cognition and Representation*, S. Schiffer and S. Steele, eds. Boulder: Westview.

Haugeland, J. 1978. 'The Nature and Plausibility of Cognitivism.' *Behavioral and Brain Sciences* **2** 215-60.

______. 1985. *Artificial Intelligence: The Very Idea*. Cambridge, MA: MIT Press.

______. 1997. *Mind Design II*. Cambridge, MA: MIT Press.

Horst, S. 2003. 'The Computational Theory of Mind,' in *The Stanford Encyclopedia of Philosophy (Fall 2003 Edition)*, ed. E. N. Zalta. URL = <http://plato.stanford.edu/archives/fall2003/entries/computational-mind/>.

Ince, D., ed. 1992. *Mechanical Intelligence*. The Collected Works of Alan Turing. Amsterdam: North-Holland.

Jackson, F. and P. Pettit 1988. 'Functionalism and Broad Content.' *Mind* **97** 381-400.

Jacquette, D. 1991. 'The Myth of Pure Syntax,' in *Topics in Philosophy and Artificial Intelligence*, L. Albertazzi and R. Poli, eds. Bozen: Istituto Mitteleuropeo di Cultura.

Loar, B. 1981. *Mind and Meaning*. Cambridge: Cambridge University Press.

Loewer, B. and G. Rey, eds. 1991. *Meaning in Mind: Fodor and his Critics*. Oxford: Blackwell.

Marr, D. 1982. *Vision*. New York: Freeman.

McCorduck, P. 1979. *Machines Who Think: A Personal Inquiry into the History and Prospects of Artificial Intelligence*. San Francisco, CA: Freeman.

McCulloch, W.S. 1965. *Embodiments of Mind*. Cambridge, MA: MIT Press.

McCulloch, W.S. and W.H. Pitts. 1943. 'A Logical Calculus of the Ideas Immanent in Nervous Activity.' *Bulletin of Mathematical Biophysics* **7** 115-133.

Miller, G.A., E.H. Galanter, and K.H. Pribram. 1960. *Plans and the Structure of Behavior*. New York: Holt.

Millikan, R.G. 1984. *Language, Thought, and Other Biological Categories: New Foundations for Realism*. Cambridge, MA: MIT Press.

______. 1993. *White Queen Psychology and Other Essays for Alice*. Cambridge, MA: MIT Press.

Oppenheim, P. and H. Putnam 1958. 'Unity of Science as a Working Hypothesis,' in *Minnesota Studies in the Philosophy of Science, Volume II Concepts, Theories, and the Mind-Body Problem*, H. Feigl, M. Scriven and G. Maxwell, eds. Minneapolis: University of Minnesota Press.

Peacocke, C. 1992. *A Study of Concepts*. Cambridge, MA: MIT Press.

Piccinini, G. 2000. 'Turing's Rules for the Imitation Game.' *Minds and Machines* **10** 573-82.

______. 2002. 'Review of John von Neumann's The Computer and the Brain.' *Minds and Machines* **12** 449-53.

______. 2003a. *Computations and Computers in the Sciences of Mind and Brain*. Doctoral Dissertation, Pittsburgh, PA: University of Pittsburgh. URL = <http://etd.library.pitt.edu/ETD/available/etd-08132003-155121/>

______. 2003b. 'Alan Turing and the Mathematical Objection.' *Minds and Machines* **12** 23-48.

______. forthcoming a. 'Functionalism, Computationalism, and Mental States.' *Studies in the History and Philosophy of Science*.

______. forthcoming b. 'The First Computational Theory of Mind and Brain: A Close Look at McCulloch and Pitts's ''Calculus of Ideas Immanent in Nervous Activity''.' *Synthese*.

Putnam, H. 1960. 'Minds and Machines,' in Dimensions of Mind: A Symposium, S. Hook, ed. New York: Collier.

______. 1963. 'Brains and Behavior,' in *Analytical Philosophy*, R.J. Butler, ed. New York: Barnes and Noble. Reprinted in N. Block, ed., *Readings in Philosophy of Psychology*, Volume 1: 24-36. London: Methuen 1980.

______. 1964. 'Robots: Machines or Artificially Created Life?' *Journal of Philosophy* **61** 668-91. Reprinted in H. Putnam, *Mind, Language and Reality: Philosophical Papers, Volume 2*. Cambridge: Cambridge University Press 1975.

______. 1967a. 'The Mental Life of Some Machines,' in *Intentionality, Minds, and Perception*, H.-N. Castañeda, ed. Detroit: Wayne State University Press.

______. 1967b. 'Psychological Predicates,' in *Art, Philosophy, and Religion*. Pittsburgh, PA: University of Pittsburgh Press. Reprinted as 'The Nature of Mental States' in W. Lycan, ed., *Mind and Cognition: An Anthology*. Second Edition, Malden: Blackwell 1999.

______. 1974. 'Comments on Wilfrid Sellars.' *Synthese* **27** 445-55.

______. 1988. *Representation and Reality*. Cambridge, MA: MIT Press.

______. 1997. 'A Half Century of Philosophy, Viewed from Within.' *Daedalus* (Winter 1997) 175-208.

Pylyshyn, Z.W. 1984. *Computation and Cognition*. Cambridge, MA: MIT Press.

Ryle, G. 1949. *The Concept of Mind*. London: Hutchinson.

Sellars, W. 1954. 'Some Reflections on Language Games.' *Philosophy of Science* **21** 204-28. Reprinted in Sellars 1963.

______. 1956. 'Empiricism and the Philosophy of Mind,' in *Minnesota Studies in the Philosophy of Science, Vol. I, The Foundations of Science and the Concepts of Psychology and Psychoanalysis*, H. Feigl and M. Scriven, eds. Minneapolis: University of Minnesota Press. Reprinted in Sellars 1963.

______. 1961. 'The Language of Theories,' in *Current Issues in the Philosophy of Science*, H. Feigl and G. Maxwell, eds. New York: Holt, Rinehart, and Winston. Reprinted in Sellars 1963.

______. 1963. *Science, Perception, and Reality*. Atascadero: Ridgeview.

______. 1967 *Science and Metaphysics: Variations on Kantian Themes*. London: Routledge and Kegan Paul.

______. 1974. 'Meaning as Functional Classification.' *Synthese* **27** 417-37.

______. 1975. 'Autobiographical Reflections,' in *Action, Knowledge, and Reality: Studies in Honor of Wilfrid Sellars*, H.-N. Castañeda, ed. Indianapolis: Bobbs-Merrill.

Shannon, C.E. and J. McCarthy. 1956. *Automata Studies*. Princeton, NJ: Princeton University Press.

Smith, B.C. 1996. *On the Origin of Objects*. Cambridge, MA: MIT Press.

Stich, S. 1983. *From Folk Psychology to Cognitive Science*. Cambridge, MA: MIT Press.

Sutherland, N.S. 1960. 'Theories of Shape Discrimination in Octopus.' *Nature* **186** 840-4.

Thelen, E. and L. Smith 1994. *A Dynamic Systems Approach to the Development of Cognition and Action*. Cambridge, MA: MIT Press.

Turing, A.M. (1936-7 [1965]). 'On Computable Numbers, with an Application to the Entscheidungsproblem. In *The Undecidable*, M. Davis, ed. Ewlett: Raven.

______. 1950. 'Computing Machinery and Intelligence.' *Mind* **59** 433-60.

van Gelder, T. 1995. 'What Might Cognition Be, if not Computation?' *The Journal of Philosophy* **92** 345-81.

Vendler, Z. 1972. *Res Cogitans: An Essay in Rational Psychology*. Ithaca: Cornell University Press.

von Neumann, J. 1958. *The Computer and the Brain*. New Haven: Yale University Press.

Wiener, N. 1948. *Cybernetics or Control and Communication in the Animal and the Machine*. Cambridge, MA: MIT Press.

Wittgenstein, L. 1953. *Philosophical Investigations*. New York: Macmillan.