COGNITIVE SCIENCE **13**, 139–144 (1989)

Parallelism and Functionalism

WILLIAM M. RAMSEY
University of California, San Diego

It has recently been argued by Paul Thagard (1986) that parallel computational models of cognition demonstrate the falsity of the popular theory of mind known as *functionalism*. It is my contention that his argument is seriously mistaken and rests on a misunderstanding of the functionalist position. While my primary aim is to defend functionalism from Thagard's attack, in the process I hope to provide some much needed clarification of matters both philosophical and computational. Since I intend to untangle issues that are often troublesome in cognitive science, the paper should prove useful even for those unfamiliar with Thagard's original piece.

I. INTRODUCTION

In his "Parallel Computation and the Mind-body Problem," Paul Thagard (1986) has suggested that recent work in cognitive modeling involving various forms of parallel computation undermines the philosophical position know as *functionalism*. Thagard rests his claim on two closely related arguments designed to demonstrate how parallelism reveals the falsity of functionalist assertions. The first argument focuses on functionalism's alleged dependence upon an explicit hardware-software distinction. Since parallel systems often blur this distinction, goes the charge, they provide a model of cognition that is not amenable to functionalist assumptions. Thagard's second argument focuses on the functionalist assertion that considerations of physical implementation are largely irrelevant to the characterization of psychological states. He maintains that parallel systems provide a counterexample to this claim, and demonstrate how hardware plays an important role in determining the sort of processes which can be implemented.

Below, my aim is to take issue with Thagard's position and argue that it rests upon an inaccurate account of functionalism's central tenets, abetted by an equivocation on ambiguous terminology. After criticizing Thagard's interpretation of functionalism and offering what I take to be a correct

---

I am grateful to Mike Bishop, Paul Churchland, Gary Cottrell, Robert Cummins, John Haugeland, Patricia Kitcher, Paul Thagard, and especially Stephen Stich for helpful comments and suggestions.

Correspondence and requests for reprints should be sent to William M. Ramsey, Department of Philosophy, B-002, University of California, San Diego, La Jolla, CA 92093.

reading of this position, I will turn my attention to his arguments. My claim will be that, once properly construed, functionalism can be seen not only to survive parallel and connectionist theories of cognition, but, indeed, to thrive with them.

II. WHAT FUNCTIONALISM ASSERTS

While there are many different forms functionalism can take (Block, 1980), it is nonetheless possible to provide a relatively generic account that will serve our purposes here. The best way to do this is to consider the central question which functionalism is designed to answer. Traditionally, a key question in philosophy of mind has been something like, *What are psychological states such as pains and beliefs?* Unfortunately, such *What is it?* questions often leave ambiguous exactly what would count as a suitable answer. A more useful way of asking the fundamental question which both functionalism and competing views in philosophy of mind are designed to answer might be put as follows: In virtue of what, exactly, can we say that the states of two different organisms (or artifacts) are tokens of the same psychological type (Dennett, 1978a)? The functionalist answer to this question is that the relevant properties for characterizing and classifying psychological states are not neural or physical properties, but functional properties that states have by virtue of their role in causal network. P.M. Churchland (1984) nicely captures the general idea when he states, "According to functionalism, the essential or defining feature of any type of mental state is the set of causal relations it bears to (1) environmental effects on the body, (2) other types of mental states, and (3) bodily behavior" (p. 36). In other words, what makes something a particular psychological state is not intrinsically tied to the substance or *stuff* it is composed of, but rather, is dependent upon the sort of causal role it plays in a given behavior-controlling system. The functionalist position makes mentality dependent upon causal systematicity, not upon being composed of a particular substance.

Thus, functionalism in psychology is simply the claim that psychological states are determined by functional or causal properties, and not by first-order physical properties. Yet, while this position my seem straightforward enough, it is all too easy to wander from this legitimate reading of functionalism to a construal that is off the mark. A clear example of this is provided by Thagard (1986), who at first adopts an account of functionalism quite similar to our own: "Functionalism...says that mental states are to be understood in terms of their functional relations to other mental states, not in terms of any particular material instantiation" (p. 301).[1] However, like

[1] Actually, the position also invokes relations to environmental stimulus and behavioral output as well, but this definition captures the basic idea.

others who sometimes embrace the computational perspective much too closely, Thagard slides from this conventional construal of the functionalist position to the view that "[i]n computational terms, functionalism is the claim that only software matters to the mental" (p. 302).

At the outset, the two readings of functionalism may appear similar enough to make the shift seem perfectly natural, and Thagard is certainly not the first to talk of functionalism in terms of hardware and software (see, e.g., Dennett, 1978b, pp. xiv–xii; Fodor, 1981). Nonetheless, invoking a notion of software to characterize functionalism is problematic for the following reason. Software is commonly construed as the program which a computer executes; however, the term *program* is itself ambiguous. On the one hand, it is sometimes used to refer to an algorithm or flowchart which merely *describes* the workings of the machine (the pattern of states leading to other states) in terms that abstract from physical details. On the other hand, *program* also commonly refers to a list of commands in a computer language which actually *causes* the machine to function in a certain fashion (similar points have been made by Cummins, 1977; and Stabler, 1983). Now the crucial point to note is that functionalism has affinities with only the former, much weaker notion (as a descriptive but causally inert flowchart), and is not in any way tied to the latter, software-oriented notion of program. Hence, while it may be tempting to regard functionalism as the view that the mind is the brain's program, one must be careful not to construe *program* in this context as anything more than a descriptive flowchart.

Given that there are really two distinct notions of program, one entailing something not required by functionalism—namely, causally efficacious software—it becomes clear where Thagard's (1986) account is erroneous. For example, his claim that functionalism "assumes a sharp distinction between hardware and software" (p. 306) is mistaken insofar as it adopts the wrong interpretation of program. In other words, even "in computational terms" (p. 302), the conventional software-hardware distinction is not the same as the functional state–physical state distinction. The former is the distinction between a set of explicit procedural rules and the machine which follows these rules. The latter is the distinction between different ways of describing and classifying various processes, by focusing either on physical properties or the more abstract functional properties. While it might be plausibly argued that the set of rules embodied in the software may itself be employed in a functional characterization of computational processes (i.e., serve as a guideline for functional description), it is certainly not the case that a functional analysis *requires* such a set of rules. Instead, functionalism merely requires that a system's behavior can be explained by appeal to psychological states that have their identity determined by their pattern of causal interactions (and not by their physical ontology). With this clarification made, we are prepared to assess the validity of Thagard's arguments against functionalism.

III. DOES PARALLELISM UNDERMINE FUNCTIONALISM?

Thagard's first main claim is that recent work on parallel computation reveals how "software and hardware are much more intertwined than the functionalist allows" (1986, p. 301). The central point of his argument is that with parallel systems, the hardware-software distinction is tenuous (or even nonexistent) and, since functionalism depends upon such a distinction, the position is thereby challenged. However, given our clarification of functionalist commitments, this criticism is clearly misguided. While Thagard is certainly correct in pointing out that the hardware-software distinction appears inappropriate for connectionist and other parallel models, functionalism does not presuppose the sort of von Neumann computer architecture which originally inspired this distinction. What matters from a functionalist perspective is that the psychological description of a state in a given system be determined by the pattern of causal relations between the various states of a system. In fact, this is exactly what we find to be the case with connectionist and parallel architecture. These models operate at a level of abstraction that makes no commitment to physical implementation, but rather, exploit the causal relations of the system's components.

Thagard's second main point is closely related to his first, but takes more direct aim at functionalism's central tenets. Thagard (1986) points out that parallel architectures lead to "qualitatively different kinds of algorithms" (p. 302) than standard serial von Neumann architectures. From this, he argues that parallelism shows how considerations of physical implementation are indeed important with regard to the mental, contrary to functionalist claims. The argument rests upon the obsevation that parallel architectures implement significantly different processes and functions than are executed on traditional computers, thereby demonstrating the close relationship between computer hardware and the sort of programs they can run. Hence, functionalism "errs in abstracting from hardware" (p. 302) and claims of multiple instantiability appear "computationally naive" (p. 301).

In order to see where Thagard has gone astray, some explication of what multiple instantiability amounts to is required; that is, it must be made clear just what can be multiple in type-identical instantiations. Once again, to accomplish this it is necessary to clear away some ambiguity in terminology. Traditionally, functionalists have claimed that in computational systems it is the hardware that can vary without altering the psychological characterization of the system's states. However, like *software,* the term *hardware* is problematic and can lead to equivocation. On the one hand, it is often used to refer to particular physical instantiations (the stuff a given system is made out of, be it neurons or silicon chips). On the other hand, it can also denote the computational architecture of a given system (i.e., the way it is hooked up). Given our clarification of functionalism, it should be clear that the former notion of hardware—the *stuff* that the apparatus is made out of —is what functionalists maintain is interchangeable. Insofar as the organization or arrangement of a system (the latter notion of hardware) determines the causal relations among its components, functionalism explicitly denies that this configuration can vary without being relevant to the psychological characterization of the system. Indeed, an embarrassment for functionalists is that a strict reading of their position suggests that only systems which are functionally isomorphic can be said to have the same cognitive states (Block, 1978; Stich, 1983). While I do not wish to address such difficulties here, they are worth noting since they help bring into focus just what functionalism amounts to. Claims of multiple instantiability entail that type-identical psychological states can be implemented by different hardware in the sense of different *kind of stuff,* not in the sense of different *computational architecture* or *causal arrangement* of such stuff.[2]

Given this clarification, it can readily be seen where Thagard's account is misguided. His case against functionalism rests on the entirely correct observation that connectionist models employ quite different computational architectures than traditional systems. However, this in no way undermines the multiple instantiability argument, which asserts not that computational architecture may deviate, but simply that there need be no commitment to particular ontological types, such as neurons. Indeed, connectionist models can be composed of neuronal arrays in the brain or lattices of copper wires and transistors or optical elements such as beams of light directed by crystals and mirrors (Abu-Mostafa & Psaltis, 1987) or groups of interacting people, or whatever else can embody the necessary causal properties required by the psychological theory. What matters to the functionalist is the way a system does whatever it does. Hence, Thagard's claim that computational architecture is relevent to psychology is certainly a claim no clear headed functionalist would wish to deny. Indeed, since any real difference between computational architecture and functional organization is hard to discern, I suspect Thagard's claim is really another way of putting the functionalist position!

Thagard is not completely misguided in arguing against those who, misconstruing functionalism, view multiple instantiability arguments as a license for ignoring neuroscience. However, in the functionalist account, considerations of material instantiation are relevant only insofar as they il-

---

[2] It is important to keep in mind that functionalism is a response to the identity theory, which held that mental states are to be identified with brain states. The functionalist response is that mentality need not be identified with neuronal events in the brain, but could be instantiated in any physical system (or even nonphysical system, if one happens to be a dualist) that embodies the proper causal interrelations. Of course, there is the remote empirical possibility that neurons may turn out to be the only thing in the universe capable of implementing the required causal interactions in reasonable time (Dennett, 1988). But note that even if neurons did turn out to be the only "right stuff," this would not undermine functionalism. Rather, functionalism is presupposed in such a scenario, since here the criteria whereby neurons gain their lone eligibility for cognitive implementation appeal to the *causal* properties (viz., fast interaction) neurons have.

luminate or provide clues to the *functional* nature of the cognitive system. While considerations of physical implementation are certainly important to cognitive science, these considerations do not contradict functionalist claims regarding the proper way to construe mental states.

IV. CONCLUSION

My goal above has been two-fold. First, to give an accurate construal of the functionalist position, thereby rescuing it from Thagard's attack. While functionalism may prove faulty for other reasons, I hope to have demonstrated that it is not heavily dependent upon traditional serial computer architecture and thus has nothing to fear from even the more unconventional parallel systems. My second aim has been to unpack some of the tricky and trouble-making concepts endemic to cognitive science. One of the virtues of the newer connectionist cognitive models is that they bring into sharp relief many sources of confusion (such as two different notions of program) that are difficult to discern from a more traditional outlook. Another important virtue of the newer models is that they point to intriguing new ways to regard cognitive mechanisms and processes. The possibility that connectionist and parallel systems will force us to rethink many issues and notions in the philosophy of mind is very real, but first we must have a firm grasp of just what these matters (including functionalism) amount to.

■ Original Submission Date: March 10, 1988.

REFERENCES

Abu-Mostafa, Y., & Psaltis, D. (1987). Optical neural computers. *Scientific American, 256 (3),* 88–95.

Block, N. (1978). Troubles with functionalism. In C.W. Savage (Ed.), *Perception and Cognition.* Minneapolis: University of Minnesota Press.

Block, N. (1980). Part three, introduction: What is functionalism? In N. Block (Ed.), *Readings in Philosophy of Psychology: Vol 1.* Cambridge, MA: Harvard University Press.

Churchland, P.M. (1984). *Matter and Consciousness.* Cambridge, MA: Bradford Books/MIT Press.

Cummins, R. (1977). Programs in the explanation of behavior. *Philosophy of Science, 44,* 269–287.

Dennett, D. (1978a). Current issues in the philosophy of mind. *American Philosophical Quarterly 15 (4),* 249–261.

Dennett, D. (1978b). *Brainstorms.* Cambridge, MA: Bradford Books/MIT Press.

Dennett D. (1988). Fast thinking. In *The intentional stance.* Cambridge, MA: Bradford Books/MIT Press.

Fodor, J. (1981). The mind-body problem. *Scientific American, 244, (1),* 124–133.

Stabler, E. (1983). How are grammars represented? *Behavioral and Brain Sciences, 6,* 391–420.

Stich, S. (1983). *From folk psychology to cognitive science.* Cambridge, MA: Bradford Books/MIT Press.

Thagard, P. (1986). Parallel computation and the mind-body problem *Cognitive Science, 10,* 301–318.