https://doi.org/10.18778/1689-4286.38.02

HYBRIS nr 38 (2017)
ISSN: 1689-4286

OPEN ACCESS

MARCIN MIŁKOWSKI
INSTITUTE OF PHILOSOPHY AND SOCIOLOGY
POLISH ACADEMY OF SCIENCES

THE FALSE DICHOTOMY BETWEEN CAUSAL REALIZATION AND SEMANTIC COMPUTATION

It has been argued that there is a tension between the semantic characterization of neural computation and the causal account of computation (Shagrir 2010). Just because the heuristic role of semantic entities in cognitive science is difficult to deny (Bechtel 2016), it might be thought that that the causal account is descriptively inadequate for our current scientific practices. Moreover, others have claimed that there is a role for content-involving computation (Rescorla 2013) in computational explanations. If there is, it means that the mechanistic account of computational explanations misses an essential aspect of the scientific practice.

However, I will argue that semantic computation and the causal account of neural computation are not mutually exclusive, and they both have important explanatory, descriptive, and heuristic roles. One does not have to decide to embrace the mechanistic account on pain of rejecting all semantic considerations; this is a false dichotomy. In particular, semantic notions usually require rich interactions with the environment and appropriate internal orchestration of the mechanism; purely computational modeling is usually limited to the internal functioning of a mechanism, while there are complex inter-level and intra-level relationships between computational, semantic, and, more broadly speaking, causal posits in explanatory models in neuroscience.

In this paper, I will show how semantic factors constrain the understanding of the phenomena to be explained so that they naturally help build better mechanistic models. In section 1, I will elucidate why one could think that there is a tension between mechanistic accounts of physical computation and semantic computation in general. Next, in Section 2, it will be argued that understanding of what cognitive systems may refer to is important in building better models of their cognitive processes by specifying the function of cognitive mechanisms *partially* in content-involving ways. For this purpose, a recent study of some phenomena in rats that are capable of 'entertaining' future paths (Pfeiffer and Foster 2013) will be analyzed in Section 3. The researchers stress that the hippocampus 'generates brief sequences encoding spatial trajectories', which is a clearly semantic way of framing the phenomenon. The above case shows that computational modeling is not just about 'turning inside'. It requires looking up, down, and around (Bechtel 2009). Looking around requires one to understand the environmental structure. In short, computation and representation, considered in an externalist fashion, do not screen off each other. Why should they? Representing requires physical information, and functional physical information processing amounts to physical computing.

1. The tension between causal realization and semantic computation

The purpose of mechanistic accounts of physical computation is to deliver a normatively and descriptively adequate list of necessary and sufficient conditions that physical systems must satisfy to qualify as computers. There are some differences between these accounts (Miłkowski 2013; Piccinini 2015), yet they may be summarized jointly in the following way. The necessary condition for candidate physical systems is that they be mechanisms (in the sense of the new mechanistic philosophy, cf. (Machamer, Darden, and Craver 2000; Bechtel 2008; Craver 2007)) whose function is to compute. The mechanism's causal structure should correspond strictly to a mathematical model of computation over physical vehicles specified in a substrate-neutral way. Moreover, the computational explanation should essentially involve processing of information (as Miłkowski states the condition) or be usable as information (as Piccinini has framed it). The rest of conditions spelled out by Miłkowski and Piccinini simply follow from the general methodological norms of mechanistic explanation.

One striking feature of the mechanistic account is that it does not require vehicles of computation to be semantic in any rich sense. In other words, mechanists explicitly reject the claim that only physical systems whose parts are semantic can be computers (Piccinini 2008; cf. Fresco 2010). They assume that there may be computers that operate on symbols without any denotation or intrinsic meaning. But this is not because they share the conviction that semantic notions are disposable altogether. Rather, they think that semantic notions are more difficult to specify than the conditions of physical computation. David Chalmers has long argued in the same vein:

> If we build semantic considerations into the conditions for implementation, any role that computation can play in providing a foundation for AI and cognitive science will be endangered, as the notion of semantic content is so ill-understood that it desperately needs a foundation itself (Chalmers 2011, 336).

As such, mechanistic and causal accounts refrain from semantic considerations. For this reason, however, they can be criticized. First of all, there is an important role of cognitive representations in cognitive explanations. For example, the whole history of research on the cognitive maps in rats was based on a strong assumption that they refer in various ways to their environment, and it has resulted in a very promising research program (Bechtel 2016). But this role seems to be irrelevant to the mechanistic account.

Second, it has been argued that mechanists cast their net too wide which results in limited pancomputationalism: they would have to admit, as Chalmers does, that a rock implements a trivial computation – or even worse, a class of trivial computations specified as *any* constant function. Namely, the rock's position may be considered to encode the result of the computation. Of course, the rock does not implement *all possible* computational functions, but still a lot of them (Shagrir 2006, 398, 2010, 272). But Miłkowski (2013, 79), for example, denies that a rock is a computer: a computational explanation of the rock's behavior is not any more predictive nor has any more explanatory power than a physical one in terms of gravity, which explains why the rock does not fly away etc. Furthermore, the rock's function is not to compute; no parts of the rock were selected according to any design as types to perform the constant functions (Miłkowski 2013, 62). Piccinini also requires that the result of the computation be usable: "the important point is that we are interested in computation because of what we (finite observers) can learn from it"

(Piccinini 2015, 256). So, while it could still be argued that the semantic constraints do not restrict the class of the candidate physical computers, other constraints allow mechanists to avoid the charge of drawing the boundary between computational and non-computational systems in a wrong way.

A third objection is much more difficult to handle *prima facie* (cf. Shagrir 2006, 409; the example has been simplified). Imagine two electrical circuits, CIRC1 and CIRC2. The first responds with voltage v_2 whenever it receives v_2 and v_2 on its input, otherwise it responds with v_1; whereas the second responds with v_1 whenever it receives voltage v_1 and v_1 on input, and otherwise with v_2. Which one of these is the OR gate that corresponds to inclusive disjunction, and which is the AND gate, the device for computing conjunction? If we treat v_1 as true, and v_2 as false, then CIRC1 is an OR gate, and CIRC2 implements an AND gate. But we might switch the logical interpretation, and then CIRC1 is an AND gate, and CIRC2 an OR gate. In other words, it seems that there are two empirically adequate but inconsistent mechanistic explanations of CIRC1 and CIRC2. This would mean that the mechanistic account is deficient and clearly worse than the semantic account. The semantic account, after all, can constrain the interpretation of voltages by taking into account the use of the circuit in its environment and possibly in a larger computational context.

Note, however, that if we have no further information about how the circuit is used, the semantic account fares no better. There is no fact of the matter that could restrict possible interpretations. So what kind of information could restrict explanations in this case? For example, there could be also one-input circuits that respond with v_2 to v_1, and vice versa. These are probably NOT gates, but we still have no way to say how to assign truth and false to voltages. But there are frequent combinations of NOT gates and CIRC1 gates. As this combination in a disjunctive normal form for propositional calculus corresponds to a material implication realized as NOT + OR, we could settle for the interpretation of v_1 as true, and v_2 as false. This is a purely syntactic hypothesis. We could also see that a device responds to two input data (for example, from its receptor devices) by using CIRC2 gate, and then v_2 triggers some response. A semantic hypothesis could be that these inputs need to be both present for the whole system to respond; so the system uses a conjunction of two receptor values. This is again a semantic hypothesis, which seems to confirm the first one. But it's definitely not sufficient in itself, as it does not allow us to reject the hypothesis that the receptors are actually silent and that what one sees is the false disjunction. In short, it takes a lot of experimentation and careful consideration to decide such issues (and it may be impossible to decide which logical connectives are at play as based merely on stimuli and responses also in the human case, cf. (Berger 1980)). It does not seem, therefore, that one account fares better than another in this case; the case is indeed difficult. However, it may motivate the claim that the mechanistic account should not restrict itself to purely formal considerations. How the mechanism responds to the environment may be essential for explaining it.

The difficult case above is similar to the one sketched in the argument put forward by Michael Rescorla (2013, 686). While Rescorla does not endorse the semantic view on computation, he claims that there are content-involving instructions in computer programs. This claim is defended against all structuralist accounts of physical computation, not only against the mechanistic view. Content-involving instructions depend in their causal efficacy on the wide social context of the use of computers; an example of this may be the dependence of the numerical notation of numbers in a programming language Scheme. It is executed on two machines in two different societies: one uses base-10 notation, and another base-13 notation, so the program to compute the greatest common divisor of 115 and 20:

```
(gcd 115 20)
```

correctly yields '5' in the base-10 society, but incorrectly in the base-13 society because '5' "is not a divisor of the base-13 denotation of '20' (namely, the number twenty-six)" (Rescorla 2013, 688).

However, the example does not fully prove the point. The problem is that the type of numerical notation is explicitly defined *syntactically* in Scheme. Specifically, it is defined in Backus-Naur Form (BNF), which is a syntactical tool used (usually with numerous extensions) to define programming languages. The format numbers is defined in the section 4.2.8, which is a part of Chapter 4 "Lexical syntax and datum syntax" of the official language specification (Flatt et al. 2009). Here are the definitions of decimal digits and hexadecimal digits:

```
<digit> → 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9
```

<hex digit> → <digit> | a | A | b | B | c | C | d | D | e | E | f | F

It would be difficult to encode base-13 using <digit> as defined above, as one would not be able to write out A, B, or C in 13-base notation (which correspond to 10, 11, and or 12 in the decimal notation). There are missing symbols, at least according to standard encoding conventions used in programming (note: one could have a non-standard notation that would treat one of the digits as special, and not use a simplistic positional encoding). In other words, while the base-13 society wrongly thinks that Scheme assumes the base-13 notation, it makes no difference as to what program is physically implemented. If there are any facts about programming languages such as Scheme, the base-13 society got them wrong.

To see that they could be shown to be wrong, it is useful to remind how language compilers or interpreters are evaluated. A series of tests, called *regression tests*, are devised in a given programming language. The execution of such tests triggers a number of assertions embedded in the test. For example, one can assert that (gcd 115 20) yields '5'. A failure of the assertion means that the compiler does not conform to the language specification.

Similarly, the fact that a user thinks that Microsoft PowerPoint is a word processing program it does not make PowerPoint a word processing program. The user is simply wrong. Of course, it might be objected that if a society had used PowerPoint for its word-processing needs, PowerPoint *would become* a word-processing application. In other words, the intention of the software application developers may not determine the function of the application, just like the intentions of designers of technological artifacts do not fully determine their functions (amulets do not really have their functions). While the issue of technical functions of artifacts is vexed, the general consensus is that one of the determining factors of technical function is also the users' intention, rather than the designer intention (see, e.g. Vermaas and Houkes 2006).

While the simple numerical notation example introduced by Rescorla does not satisfactorily show that the mechanistic approach is deficient, there is a deeper point there. The point can be easily proven by adapting the example and using some notation that would use, say 8-base, as there would be no symbols missing for the BNF specification (Rescorla, personal communication). In such a situation, it would be impossible to determine the interpretation of ‘100’: one society would understand it to stand for decimal 64, and another for decimal 100. And there is, potentially, an infinite number of similar ambiguities inherent in programming languages.

To sum up, the function of mechanisms may depend on their social and widespread use, and the use may involve semantic factors. People frequently use computers to manipulate their external representations. Indeed, the rest of the paper will argue that the proper focus on the function of mechanisms shows that semantic considerations may play a serious role in computational explanations considered mechanistically. Mechanistic explanation should, at least for an important class of computational mechanisms, include semantic considerations.

2. Building mechanistic models by including semantic constraints

In this section, the notion of function used in mechanistic accounts of computation will be made more explicit. Then it will be shown that some but not all computational mechanisms have semantic functions (in a sense to be elucidated below). These functions will be only partially explained computationally. However, they will constrain the space of plausible computational mechanisms posited in mechanistic explanations.

Mechanistic accounts of physical computation focus predominantly on functional mechanisms (cf. Garson 2013). However, there is a debate over the notion of function appropriate for mechanistic explanations. Most defenders of mechanistic explanations rely on a fairly weak account that equates function with a capacity of a given physical system — its capacity to perform some causal role owning to its internal organization (Cummins 1975)— that is of epistemic interest (cf. Craver 2013, 2001). While it is a fact of the matter whether the system has such function or not, the ascription is based on the perspective taken by a beholder. But defenders of the mechanistic account of computation do *not* embrace the perspectivalist view: they argue that the mechanistic account of physical computation should avoid, if possible, any appeal to epistemic interest of beholders, since numerous objections against the possibility of an objective account of physical computation rely on the possibility of arbitrary ascriptions of computations to physical systems. Moreover, they want to account for malfunction of computational systems or the failure of physical mechanisms to perform their function. One of the major objections against the perspectivalist view is that the same capacity of a physical system may count as functional and dysfunctional at the same time (Millikan 2002). Instead, Piccinini and Miłkowski have both argued for teleological accounts. While there are some notable differences between their accounts, they both seem to embrace a unified view on a function that includes technical functions of artifacts and teleological functions of natural computing mechanisms. For example, Piccinini defines the notion in the following way:

> A teleological function (generalized) is a stable contribution to a goal (either objective or subjective) of organisms by either a trait or an artifact of the organisms (Piccinini 2015, 116).

The upshot of this definition is that there cannot be any computers without organisms: either as their users or as physical mechanisms whose goals are satisfied by the existence of such computers. Quite clearly, before there were organisms, there were rocks, and they were not implementing any functions. So far, so good. But couldn't it be possible in principle that there could exist computational physical systems other than organisms or artifacts produced by organisms? For example, one could imagine naturally evolved robots that have their goals fulfilled thanks to computation. But Piccinini rejects this possibility by saying that these would count as organisms in a broader sense (Piccinini 2015, 113).

The approach of Miłkowski is partially similar to the one proposed by Craver and Cummins but also relies on the teleological view defended at length by Ulrich Krohs (2004, 2007): "the functional role of a component is one of its causal roles, such that it contributes to the system behavior of the mechanism (as in the classical analytical account in Cummins 1975; for a mechanistic variant of this account, see Craver 2001), but the organization of the mechanism is based on the process of selection of its parts as types" (Miłkowski 2013, 62). This requires a bit more elucidation. Krohs defends a design-based notion of function where design is understood as a type fixation of a complex entity. The type-fixed entity is defined thus:

> (COM) A complex entity is type-fixed iff its components are type-fixed.

> (TF) A component of an entity is type-fixed iff it is part of the entity because of its type and not merely because of its properties (Krohs 2007, 77).

Again, the components of rocks are not selected as types: there is no assembly process that generates them for the purpose of computing constant functions. Yet, in contrast to Piccinini, no appeal is made to the existence of organisms.

It's beyond the scope of this paper to compare both accounts in detail, and see how they address the main objections in the debate over teleological function. Still, it's instructive to discuss shortly an alternative view on technical functions. For example, a sophisticated ICE theory (Intentional-Causal role-Evolutionist) is defended by Pieter Vermaas and Wybo Houkes:

> An agent *a* ascribes the capacity to ϕ as a function to an artefact *x*, relative to a use plan *p* for *x* and relative to an account *A*, iff:
>
> I. the agent *a* has the capacity belief that x has the capacity to ϕ, when manipulated in the execution of *p*, and the agent *a* has the contribution belief that if this execution of *p* leads successfully to its goals, this success is due, in part, to *x*'s capacity to ϕ;
>
> C. the agent *a* can justify these two beliefs on the basis of *A*; and
>
> E. the agents *d* who developed *p* have intentionally selected *x* for the capacity to ϕ and have intentionally communicated *p* to other agents *u* (Vermaas and Houkes 2006, 9).

Note that this account rules out ascriptions of computational functions to biological brains, as they were not selected by any intelligent agent.[1] However, it can be easily used to ascribe functions to a computer running a Scheme interpreter or to a pair of logical gates. One can consult the agents who have developed the Scheme interpreter and determine that base-13 society is indeed wrong in assuming that '5' is given in base-13 notation (see section 1). In other words, under ICE account, semantic considerations may be framed in terms of the developers' intentions. And

[1] At least most of them, except for direct genetic modifications, such as the ones used in optogenetics (Deisseroth et al. 2006)

these considerations may constrain the hypotheses about the function of computational artifacts.

A similar move is possible under Miłkowski's account, as long as the type fixation process is sensitive to semantic values of computations performed. For example, one may analyze the compiler or interpreter of Scheme programming language to see whether the results of defined numerical functions turn out to be systematically correct and coincide with the BNF specification. The BNF specification, after all, was most probably used to design the compiler or interpreter (it makes no difference to this account whether it was this particular specification or some other). And the same can be done using Piccinini's account: the goals of organisms using Scheme on their computers will be achieved if the Scheme interpreter or compiler is executed, so the computer may be ascribed a function to run Scheme programs (interpreted or compiled), and thus to execute any function the user might want to execute. So, while mechanistic accounts of function are more general, in terms of semantic considerations, they do not fall behind sophisticated accounts of technical functions.

The upshot of this short discussion is that the gist of considerations cited in favor of semantic accounts of computation can be preserved in the mechanistic account. For example, Jerry Fodor has claimed that it's characteristic for (some) mental processes to preserve semantic properties such as truth. In his opinion, what makes computational psychology so compelling is the fact that one may build a computer that does the same:

> if you have a device whose operations are transformations of symbols, and whose state changes are driven by the syntactic properties of the symbols that it transforms, it is possible to arrange things so that, in a pretty striking variety of cases, the device reliably transforms true input symbols into output symbols that are also true. I don't know of any other remotely serious proposal for a mechanism that would explain how the processes that implement psychological laws could reliably preserve truth (Fodor 1995, 9).

While mechanists have pointed out that there could be computational processes that do not preserve the constraint of truth preservation — a trivial counterexample is a single NOT gate – there are plenty that do. So while preservation of semantic properties is not an essential property of computational mechanisms, it is a property that can be partially explained computationally in terms of reliable processes of computation over vehicles that were arranged in a manner that preserves semantic constraints. Simply, one cannot explain truth preservation unless there are also appropriate syntactic processes. This is what can be explained computationally about representation; so even if intentionality cannot be reduced to computation, some regularities in intentional processes can be explained computationally.

In semantic computation, the vehicles over which the computations are performed are bearers of semantic information. Notice that a vehicle *cannot* have semantic properties if it is not a bearer of structural information (data): the data needs to be well-formed to have semantic content. The condition of well-formedness of data is always satisfied for computational mechanisms according to the mechanistic account of physical computation. But computational mechanisms need not operate on meaningful data. They may as well process gibberish.

In general, two kinds of semantic information may be distinguished: instructional and factual (Floridi 2010, 34). The first conveys the need for a specific action, and the latter states the facts. While it is not controversial that in programmable computers there are programs full of instructional information (Fresco and Wolf 2013) it is far from obvious that one can build computers whose symbols are genuinely or intrinsically meaningful in the factual sense (Harnad 1990). The mechanistic account of physical computation does not presuppose, therefore, that all computation is over meaningful data. However, it does not exclude the possibility of computation over meaningful data. In this, it clearly differs from the semantic view defended by Shagrir, and at the same time, it can include semantic constraints in mechanistic explanations. This also means that the mechanistic account is not *merely* structural: it may appeal to content-involving facts, such as the ones invoked by Rescorla.

While the account of what makes well-formed data semantic goes beyond the scope of this paper (but see Floridi 2010; Dretske 1982; MacKay 1969), there are mechanistic explanations of representational phenomena. Mechanists presuppose that intentionality or semantic properties may be explained in terms of semantic information and teleological function, and some have already proposed accounts of representational or intentional mechanisms (Miłkowski 2015; Plebe and

De La Cruz 2016). Representational mechanisms are an important proper subset of computational mechanisms.

The assumption that a given mechanism is representational constrains computational hypotheses about the system; here, the mechanistic account follows Shagrir's (2001) analysis. Let's take the example of ambiguous circuits, CIRC1 and CIRC2. If we know how these circuits are supposed to work – what their representational function is, i.e., what kind of characteristics of entities are represented by computational vehicles – we can settle for one interpretation of the voltages in the circuits. To wit, the mechanistic account, thanks to the notion of the representational function of computational mechanisms, can make use of the considerations cited by Shagrir and Rescorla. In the next section, one case will be studied in detail to show how.

3. Semantic constraints at work

Cognitive maps are paradigmatic examples of genuine mental representations cited by neuroscience. The representational hypothesis, put forward by Edward Tolman (1948), has inspired a particularly rich research program (Bechtel 2016). Such maps are structured but not reducible to language-like media (Rescorla 2009); they are also prime examples of structural representations (Cummins 1996). While there are multiple different mechanisms involved in the functioning of cognitive maps – different kinds of cells are responsible for representing distinct features of the environment in quite complex ways, a recent finding of representing future paths as trajectories to a goal will be analyzed here. The finding concerns a neural code discovered in the rat's hippocampus.

The rat's hippocampus generates brief sequences encoding spatial trajectories strongly biased to progress from the subject's current location to a known goal location. Pfeiffer and Forster (2013) were able to find direct evidence for the existence of future-focused navigational activity of place cells in a realistic two-dimensional environment. They have elegantly shown that it is related to sharp-wave-ripple (SWR) events; SWRs are irregular bursts of brief (100–200 ms) large-amplitude and high-frequency (140–200 Hz) neuronal activity in the hippocampus. In other words, there is direct evidence that place cells are involved in planning future routes. To find this evidence, Pfeiffer and Foster used a 40-tetrode microdrive that permitted synchronous electrophysiological activity recording from 250 place cells. Using sophisticated mathematical methods, they were able to decode the locations represented by this cell ensemble in SWRs.

However, the finding is all the more exciting because it can be integrated with previous work on cognitive maps (Schmidt and Redish 2013). This previous work is also computational. A number of computer simulation studies were designed to study cognitive maps and their possible neural encodings (see e.g. McNaughton et al. 2006; Conklin and Eliasmith 2005). Simulations take inspiration from experimental results and often go beyond available evidence, and experiments are then designed to test for plausible computational schemes. Neuroscientists understand that there are neural structures that have special computational roles, but that doesn't mean that a single anatomical structure plays just one role; as it turns out, it may play multiple roles in multiple neural systems, which is evidenced in the work on the hippocampus (Redish 1999, xiii). The neural code used to plan future routes is yet another code among the ones already discovered in navigation computations performed by the rat.

From the mechanistic point of view, current computational models, impressive as they are, remain incomplete because of the intrinsic complexity of the navigational subsystems and difficulties involved in their study. What is notable here is that Pfeiffer and Foster assume a representational point of view and explore the electrophysiological activity of neurons as related to the features of the external environment in the rat subject in various experimental conditions. In the discussed experiment, rats foraged for food distributed in random locations. Every day, they would start from the same home location, which remained constant for the day, and would change the next day. This way, rats could try novel routes. In other experiments, rats may learn the topology of the maze and then they are transferred to similar mazes to discover how they remember the topology (Alme et al. 2014). In other words, what is studied is the relationship between the activity of the organism and its environment. Only in such a context does a computational model make sense; and the overarching hypothesis is that neural processes are involved in various representational tasks.

The discovery of encoding requires researchers to understand what features of the environment could be encoded by neural events, and then to study (statistically) the results of electrophysiological recordings as related to these features. In the study under analysis, the researchers have found that there are two kinds of trajectory events: ones that were initiated when the rat was at the Home location ('home events'), and the ones initiated elsewhere ('away events'). Interestingly, it turns out that the Home location was over-represented in away-events relative to other locations in the open field. This means that researchers need not presuppose that representation in the brain is absolutely veridical; it may be biased for some reason (one may speculate, for example, that the Home location is particularly important because the rat started its exploration there). So how can they be sure that these trajectory events really represent future routes? The confirmation of this representational hypothesis is that the rat simply takes one of the future routes immediately after planning it.

The trajectory events discovered by Pfeiffer and Foster are consistent with the number of previous hypotheses and allow researchers to make them more precise by offering an experimental method:

> trajectory events relate to hippocampal function in multiple conceptual contexts: as a cognitive map in which routes to goals might be explored flexibly before behaviour, as an episodic memory system engaging in what has been termed 'mental time travel', and as a substrate for the recall of imaginary events. These conceptualizations reflect a continuity with earlier speculations on animals' capacities for inference (Pfeiffer and Foster 2013, 78).

In other words, understanding the context in which a given mechanism works helps the modelers to analyze its internal structure that is supposed to perform inferential computations, especially those related to mental time travel, route planning, and the recall of imaginary events. The experimental method yields semantic constraints on computational models of these inferential processes: plausible models should conform to neural encoding schemes discovered experimentally. Otherwise, computational models of the hippocampus might diverge from what is known about the behavioral functioning of the rat, and this is precisely what researchers want to avoid. In terms of the mechanistic approach to explanation, one may state it in the following way: The phenomenon to be explained is described as the function of place cells to represent future paths, and the causal explanation (currently somewhat incomplete, as precipitating conditions of the mechanism are not clear) shows the orchestrated activity of place cells that contributes to the realization of this function.

It needs to be noted that computational models are in general difficult to confirm or disconfirm experimentally; one may usually produce a number of *different* models consistent with experimental findings. Including more constraints allows researchers to reject at least some models. This way modeling is less arbitrary. In some sense, modelers need to practically solve the ambiguities such as the ones mentioned by Shagrir in his example of experimentally ambiguous logical gates, or by Rescorla in his example of ambiguous numerical encoding. They may do it by including semantic constraints in the specification of the explanandum phenomenon.

To sum up, it is only natural to assume that the function of neural mechanisms involved in solving representational tasks is to represent. There is no particular reason to abstain from representational hypotheses, which are extremely helpful from the mechanistic perspective to make models explanatorily more plausible.

4. Conclusion

Successful cognitive modeling is a question of satisfying multiple constraints from multiple fields of inquiry, levels of organization, and theories. Semantic and ecological considerations are not just heuristics of discovery of mechanisms. They are constraints over the space of possible mechanism representations. By a *constraint* I understand a representation that shapes the boundaries of the space of plausible representations of mechanisms or the probability distribution over that space (Miłkowski 2017). The more constraints are satisfied, the more integrated the model of a mechanism becomes. Ideally, all constraints should be satisfied to produce an explanatorily plausible mechanism model.

The mechanistic view on physical computation does not assume that all computation makes sense. There may be plenty of computation without any representational role. However, there are computations over representations, and these are extremely important for cognitive (neuro)science. For this reason, to remain descriptively and normatively adequate, the mechanistic view has to assume that representational constraints are important, and they can be naturally included in descriptions of functions of computational / representational mechanisms.

Hence, the dichotomy between the causal realization and semantic computation is false. Semantic computations are realized causally, and they can be studied mechanistically. For the mechanistic account of explanation, there is no reason to abstain from representational hypotheses in science. The proponents of the mechanistic account of physical computation only stress that not all computers operate on semantic information. But computation and representation do not screen off each other.

REFERENCES

Alme, Charlotte B., Chenglin Miao, Karel Jezek, Alessandro Treves, Edvard I. Moser, and May-Britt Moser. 2014. "Place Cells in the Hippocampus: Eleven Maps for Eleven Rooms." *Proceedings of the National Academy of Sciences* 111 (52):18428–35. https://doi.org/10.1073/pnas.1421056111.

Bechtel, William. 2008. *Mental Mechanisms*. New York: Routledge (Taylor & Francis Group).

———. 2009. "Looking Down, Around, and up: Mechanistic Explanation in Psychology." *Philosophical Psychology* 22 (5):543–64. https://doi.org/10.1080/09515080903238948.

———. 2016. "Investigating Neural Representations: The Tale of Place Cells." *Synthese* 193 (5):1287–1321. https://doi.org/10.1007/s11229-014-0480-8.

Berger, Alan. 1980. "Quine on 'Alternative Logics' and Verdict Tables." *The Journal of Philosophy* 77 (5):259–77. https://doi.org/10.2307/2025755.

Chalmers, David J. 2011. "A Computational Foundation for the Study of Cognition." *Journal of Cognitive Science*, no. 12:325–59.

Conklin, John, and Chris Eliasmith. 2005. "A Controlled Attractor Network Model of Path Integration in the Rat." *Journal of Computational Neuroscience* 18 (2):183–203. https://doi.org/10.1007/s10827-005-6558-z.

Craver, Carl F. 2001. "Role Functions, Mechanisms, and Hierarchy." *Philosophy of Science* 68 (1):53–74.

———. 2007. *Explaining the Brain. Mechanisms and the Mosaic Unity of Neuroscience*. Oxford: Oxford University Press.

———. 2013. "Functions and Mechanisms: A Perspectivalist View." In *Functions: Selection and Mechanisms*, edited by Philippe Hunemann, 133–58. Dordrecht: Springer.

Cummins, Robert. 1975. "Functional Analysis." *The Journal of Philosophy* 72 (20):741–65.

———. 1996. *Representations, Targets, and Attitudes*. Cambridge, Mass.: MIT Press.

Deisseroth, Karl, Guoping Feng, Ania K Majewska, Gero Miesenböck, Alice Ting, and Mark J Schnitzer. 2006. "Next-Generation Optical

Technologies for Illuminating Genetically Targeted Brain Circuits." *The Journal of Neuroscience : The Official Journal of the Society for Neuroscience* 26 (41):10380–86. https://doi.org/10.1523/JNEUROSCI.3863-06.2006.

Dretske, Fred I. 1982. *Knowledge and the Flow of Information*. 2nded. Cambridge, Mass.: MIT Press.

Flatt, Matthew, Anton van Straaten, Robby Findler, and Jacob Matthews. 2009. *Revised[6] Report on the Algorithmic Language Scheme*. Edited by Michael Sperber. Cambridge; New York: Cambridge University Press. http://www.r6rs.org.

Floridi, Luciano. 2010. *Information: A Very Short Introduction*. Oxford: Oxford University Press.

Fodor, Jerry A. 1995. *The Elm and the Expert: Mentalese and Its Semantics*. Cambridge, Mass.: MIT Press.

Fresco, Nir. 2010. "Explaining Computation Without Semantics: Keeping It Simple." *Minds and Machines* 20 (2):165–81. https://doi.org/10.1007/s11023-010-9199-6.

Fresco, Nir, and Marty J. Wolf. 2013. "The Instructional Information Processing Account of Digital Computation." *Synthese* 191 (7):1469–92. https://doi.org/10.1007/s11229-013-0338-5.

Garson, Justin. 2013. "The Functional Sense of Mechanism." *Philosophy of Science* 80 (3):317–33. https://doi.org/10.1086/671173.

Harnad, Stevan. 1990. "The Symbol Grounding Problem." *Physica D* 42:335–46.

Krohs, Ulrich. 2004. "Der Begriff Des Designs." In *Eine Theorie Biologischer Theorien. Status Und Gehalt von Funktionsaussagen Und Informationstheoretischen Modellen*, 59:70–119. Berlin: Springer.

———. 2009. "Functions as Based on a Concept of General Design." *Synthese* 166 (1):69–89. https://doi.org/10.1007/s11229-007-9258-6.

Machamer, Peter, Lindley Darden, and Carl F. Craver. 2000. "Thinking about Mechanisms." *Philosophy of Science* 67 (1):1–25.

MacKay, Donald MacCrimmon. 1969. *Information, Mechanism and Meaning*. Cambridge: M.I.T. Press.

McNaughton, Bruce L, Francesco P Battaglia, Ole Jensen, Edvard I Moser, and May-Britt Moser. 2006. "Path Integration and the Neural

Basis of the 'Cognitive Map'." *Nature Reviews. Neuroscience* 7 (8):663–78. https://doi.org/10.1038/nrn1932.

Miłkowski, Marcin. 2013. *Explaining the Computational Mind*. Cambridge, Mass.: MIT Press.

———. 2015. "Satisfaction Conditions in Anticipatory Mechanisms." *Biology & Philosophy* 30 (5):709–28. https://doi.org/10.1007/s10539-015-9481-3.

———. 2016. "Unification Strategies in Cognitive Science." *Studies in Logic, Grammar and Rhetoric* 48 (1):13–33. https://doi.org/10.1515/slgr-2016-0053.

Millikan, Ruth Garrett. 2002. "Biofunctions: Two Paradigms." In *Functions: New Essays in the Philosophy of Psychology and Biology*, edited by Andrew Ariew, Robert Cummins, and Mark Perlman. New York: Oxford University Press, USA.

Pfeiffer, Brad E, and David J Foster. 2013. "Hippocampal Place-Cell Sequences Depict Future Paths to Remembered Goals." *Nature* 497 (7447). Nature Publishing Group:74–79. https://doi.org/10.1038/nature12112.

Piccinini, Gualtiero. 2008. "Computation without Representation." *Philosophical Studies* 137 (2):205–41. https://doi.org/10.1007/s11098-005-5385-4.

———. 2015. *Physical Computation: A Mechanistic Account*. Oxford: Oxford University Press.

Plebe, Alessio, and Vivian M. De La Cruz. 2016. *Neurosemantics*. Vol. 10. Studies in Brain and Mind. Cham: Springer International Publishing. http://link.springer.com/10.1007/978-3-319-28552-8.

Redish, A. David. 1999. *Beyond the Cognitive Map: From Place Cells to Episodic Memory*. Cambridge, Mass.: The MIT Press.

Rescorla, Michael. 2009. "Cognitive Maps and the Language of Thought." *The British Journal for the Philosophy of Science* 60 (2):377–407. https://doi.org/10.1093/bjps/axp012.

———. 2013. "Against Structuralist Theories of Computational Implementation." *The British Journal for the Philosophy of Science* 64 (4):681–707. https://doi.org/10.1093/bjps/axs017.

Schmidt, Brandy, and A. David Redish. 2013. "Neuroscience: Navigation with a Cognitive Map." *Nature* 497 (7447):42–43. https://doi.org/10.1038/nature12095.

Shagrir, Oron. 2001. “Content, Computation and Externalism.” *Mind* 110 (438):369–400.

———. 2006. “Why We View the Brain as a Computer.” *Synthese* 153 (3):393–416. https://doi.org/10.1007/s11229-006-9099-8.

———. 2010. “Brains as Analog-Model Computers.” *Studies In History and Philosophy of Science Part A* 41 (3). Elsevier Ltd:271–79. https://doi.org/10.1016/j.shpsa.2010.07.007.

Tolman, Edward Chace. 1948. “Cognitive Maps in Rats and Men.” *Psychological Review* 55 (4):189–208.

Vermaas, Pieter E., and Wybo Houkes. 2006. “Technical Functions: A Drawbridge between the Intentional and Structural Natures of Technical Artefacts.” *Studies in History and Philosophy of Science Part A*, The dual nature of technical artefacts, 37 (1):5–18. https://doi.org/10.1016/j.shpsa.2005.12.002.

ABSTRACT

THE FALSE DICHOTOMY BETWEEN CAUSAL REALIZATION AND SEMANTIC COMPUTATION

In this paper, I show how semantic factors constrain the understanding of the computational phenomena to be explained so that they help build better mechanistic models. In particular, understanding what cognitive systems may refer to is important in building better models of cognitive processes. For that purpose, a recent study of some phenomena in rats that are capable of 'entertaining' future paths (Pfeiffer and Foster 2013) is analyzed. The case shows that the mechanistic account of physical computation may be complemented with semantic considerations, and in many cases, it actually should.

KEYWORDS: physical computation; semantic account of computation; mechanistic account of computation; mechanistic explanation; causal realization