Physical Instantiation and the Propositional Attitudes

Citation for published version:
Schweizer, P 2012, 'Physical Instantiation and the Propositional Attitudes' Cognitive Computation, vol 4, no. 3, pp. 226-235. DOI: 10.1007/s12559-012-9134-7

Digital Object Identifier (DOI):
10.1007/s12559-012-9134-7

Link:
Link to publication record in Edinburgh Research Explorer

Document Version:
Peer reviewed version

Published In:
Cognitive Computation

[Published in *Cognitive Computation*, 2012, DOI 10.1007/s12559-012-9134-7]

Physical Instantiation and the Propositional Attitudes

Paul Schweizer
Institute for Language, Cognition and Computation
School of Informatics, University of Edinburgh
3 Charles Street, Edinburgh EH8 9AD, UK
Email: paul@inf.ed.ac.uk, t: +44(0)131 650 2704, f: +44(0)131 651 3190

Abstract. The paper addresses a standard line of criticism of the Computational Theory of Mind (CTM), based on the claim that the notion of realizing a computational formalism is overly liberal to the point of vacuity. I argue that even for interesting and powerful cases, realization is essentially a matter of approximation and degree, and interpreting a physical device as performing a computation is always relative to our purposes and potential epistemic gains. However, while this may fatally undermine a computational explanation of conscious experience, I contend that, contra Putnam and Searle, it does not rule out the possibility of a scientifically defensible account of propositional attitude states in computational terms.

Keywords: computational theory of mind; multiple realizability; consciousness; propositional attitudes.

1. Introduction

Central to the theory of computation is the intuitive notion of an effective or ‘mechanical’ procedure, which is simply a finite set of instructions for syntactic manipulations that can be followed by a machine, or by a human being who is capable of carrying out only very elementary operations on symbols. A key constraint is that the machine or the human can follow the rules without knowing what the symbols *mean*. The notion of an effective procedure is obviously quite general – it doesn’t specify what form the instructions should take, what the manipulated symbols should look like, nor precisely what manipulations are involved. The underlying restriction is simply that they are finitary and can proceed ‘mindlessly’ i.e. without any additional interpretation or understanding. So there are any number of different possible frameworks for filling in the details and making the notion rigorous and precise. Turing’s ‘automatic computing machines’ [1] (TMs), supply a very intuitive and elegant rendition of the notion of an effective procedure, and in the ensuing discussion TMs will be taken as the conceptual archetype. But there is a variety of well known alternative frameworks, including Church’s Lambda Calculus, Gödel’s Recursive Function Theory, Lambek’s Infinite Abacus Machines, etc.

Turing machines and other types of computational formalisms are *mathematical abstractions*. Like equations, sets, Euclid’s perfectly straight lines, etc., TMs don’t exist in physical time or space, and they have no causal powers. In order to perform *actual* computations, an abstract Turing machine, thought of as a formal program of instructions, must be realized or instantiated by a suitable arrangement of matter and energy. And as Turing observed long ago [2], there is no privileged or unique way to do this. Like other abstract structures, such as chess games and isosceles triangles, Turing machines are *multiply realizable* - what unites different types of physical implementation of the same abstract TM is nothing that they have in common as physical systems, but rather a structural isomorphism characterized at a higher level of description. Hence it’s possible to implement the very same computational formalism using modern electronic circuitry, a human being executing the instructions by hand with paper and pencil, a Victorian system of gears and levers, as well as more atypical arrangements of matter and energy including toilet paper and beer cans. Let us call this ‘downward’ multiple realizability, wherein, for any given formal procedure, this *same* abstract computational formalism can be implemented via an arbitrarily large number of *distinct* physical systems. And let us denote this type of downward multiple realizability as ‘↓MR’.

After the essential foundations of the mathematical theory of computation were laid, the vital issue then became one of engineering – how best to utilize state of the art technology to construct rapid and powerful physical implementations of the abstract mathematical blueprints, and hence perform actual high speed computations *automatically*. This is a clear and deliberate ↓MR endeavour, involving the intentional construction of artefacts, painstakingly devised to instantiate the algorithms that we have created. From this top-down perspective, there is an obvious and pragmatically indispensible sense in which the hardware that we have designed and built can be said to perform genuine computations in physical space-time.

2. The Computational Theory of Mind (CTM)

According to the widely embraced ‘computational paradigm’, which underpins cognitive science, Strong AI and various allied positions in the philosophy of mind, computation (of one sort or another) is held to provide the scientific key to explaining mentality and, ultimately, to reproducing it artificially. The paradigm maintains that cognitive processes are essentially computational processes, and hence that intelligence in the physical world arises when a material system implements the appropriate kind of computational formalism. In terms of the classical model of computation as rule governed symbol manipulation, the relation between the abstract program level and its realization in physical hardware then yields an elegant solution to the traditional mind-body problem in philosophy: the *mind* is to the *brain* as a *program* is to the *hardware* of a digital computer.

On the CTM view, mental states and properties are seen as complex internal processing states, which computationally interact within a formal structure of internal state transitions, thereby mediating the inputs and outputs of intelligent behaviour. Hence any mental process leading to an action will have to be embodied as a physical brain process that realizes the underlying computational formalism. A perceived virtue of this approach is that it can potentially provide a *universal* theory of cognition, a theory which is not limited by the details and idiosyncrasies of the human organism. Since mentality is explained in computational terms, and, as above, computational formalisms are multiply realizable, it follows that the mind-program analogy can be applied to any number of different types of creatures and agents.

Combining CTM with ↓MR, it follows that a human, a Martian and a robot could all be in exactly the *same* mental state, where this sameness is captured in terms of implementing the same cognitive computation, albeit via radically different forms of physical hardware. So on this view, computation is seen as providing the scientific paradigm for explaining mentality in general – all cognition is to be literally described and understood in computational terms.

But rather than welcoming multiple realizability as a theoretical virtue promising a universal account of mentality, various opponents of CTM target this feature as its Achilles heel. In *Representation and Reality*, Hilary Putnam [3] argues that implementing a computational formalism cannot serve as the theoretical criterion of mentality, because such a standard is overly liberal to the point of vacuity. As a case in point he offers a proof of the thesis that *every* open physical system can be interpreted as the realization of *every* finite state automaton. In a related vein, John Searle [4] argues that computation is not an intrinsic property of physical systems. Instead, it is an observer relative interpretation that we project on to various physical systems according to our interests and goals. And on such a view, computation *per se* is too weak to offer a theoretical criterion of mentality, since it fails to uniquely characterize and isolate the phenomenon in question.

Searle contends that multiple realizability makes CTM conceptually bankrupt, since virtually any physical system can be interpreted as following virtually any program. Thus hurricanes, our digestive system, the motion of the planets, even an apparently inert lecture stand, all possess a level of description at which they instantiate any number of different programs – but it is absurd to attribute mental states or intelligence to them on that basis. Even though the stomach has inputs, internal processing states and outputs, it isn't a cognitive system. Yet if one wanted to, one could interpret the inputs and outputs as code for any number of symbolic processes. And in his article 'Is the Brain a Digital Computer' [5] Searle attempts to illustrate the extreme conceptual looseness of the notion of implementing an abstract formalism by famously claiming that the molecules in his wall could be interpreted as running the Wordstar program.

Let us label multiple realizability in this direction, wherein any given *physical system* can be interpreted as implementing an arbitrarily large number of different *computational formalisms* 'upward MR' and denote it as '↑MR'. The basic import of ↑MR is the non-uniqueness of computational ascriptions to particular arrangements of matter and energy. In the extreme versions suggested by Putnam and Searle, there are apparently no significant constraints whatever – it is possible in principle to interpret every open physical system as realizing every computational procedure. Let us call this extreme version '*universal* upward MR' and denote it as '↑MR*'. If every physical system can be construed as implementing every computational formalism, then clearly every computational formalism is realized by every physical system, and the corresponding position in the reverse direction, i.e. universal downward MR (↓MR*), is also true. So in this sense the two positions are equivalent and ↑MR* = ↓MR*.

But mere ↑MR is clearly weaker than ↑MR*, since the former does not assert that there are *no* salient constraints, and hence ↑MR would be consistent with the denial that, e.g., the molecules in Searle's wall can in fact be interpreted as implementing the

Wordstar program (although every physical system might still be interpretable as implementing some exceedingly large set of distinct computations). What ↑MR denies is simply that any particular computational description that *can* be legitimately applied is somehow privileged or unique, and hence the relation in the direction from physical system to computational ascription is held to be irreducibly one-to-many.

3. Defending CTM

In response to the Putnam/Searle universal realizability objection, various defenders of CTM attempt to block ↑MR* and/or its theoretical impact with two immediate and natural tactics. One (i) is to narrow the set of computations relevant, since only very complex and advanced procedures will be of any interest to CTM as candidates for mental architecture. Putnam's initial proof involves *inputless* finite state automata, and these are commonly dismissed as too primitive. Full input/output capabilities are required, as well as rich internal processing structure, which calls for something on a par with, say, Jerry Fodor's [6] Language of Thought (LOT) model of cognition. Hence in what follows it will be assumed that only formalisms comparable to TMs in general strength and complexity are under consideration.

The second (ii) tactic is to place greater constraints on what counts as a legitimate physical realization. In line with this approach, David Chalmers [7] advocates what he takes to be two essential restrictions in distinguishing many of the 'false' cases of implementation required by Putnam's argument, from 'true' cases consistent with a non-trivial reading of CTM. The first (ii_a) is an appropriate *causal* structure relating the state transitions in the physical implementation of the computational formalism (this is also proposed by, e.g. Ronald Chrisley [8]) , and the second (ii_b) is the ability of the mapping to support *counterfactual* sequences of transitions on inputs not actually given (which is also considered by Tim Maudlin [9]). Both of these are quite significant features inviting extended analysis, which unfortunately is not possible within the confines of the current discussion. However, selected points regarding each of these proffered constraints will be touched on below.

Regarding point (ii_a), Chalmers argues that it is a necessary condition (for counting as a legitimate implementation) that the pattern of abstract state transitions constituting a particular run of the computational procedure on a particular input, must map to an appropriate transition of physical states of the machine, where the relation between succeeding states in this sequence is governed by proper causal regularities. This suggestion constitutes quite a natural and immediate corrective measure in response to the extreme laxity that might seem to underwrite ↑MR*, since the physical states in the chronological progression exploited by Putnam's method have no nomological connection. Nevertheless, I would argue that the constraint is too strong in general and rules out cases which should not be excluded. For example, in the Chinese room scenario, or indeed *any* situation where a human being is following an abstract computational procedure, the transition from one state to the next is not causal in any straightforward physical or mechanical sense. When I take a machine table set of instructions specifying a particular TM and then perform a given computation with pencil and paper by sketching the configuration of the tape at each step in the computation, the transitions sketched on the piece of paper are *not themselves* causally connected: one sketch in the sequence in no way causes the next. It is only through my understanding and intentional choice to execute the procedure that the next state appears on the paper. Clear-cut physical causation of the sort required by Chalmers comes in only very indirectly, as in light rays illuminating the page and allowing me to see the symbols, and at an elementary and extraneous level, as in the friction between the pencil lead and the paper's surface causing various marks to appear.

Yet this is a perfectly legitimate and indeed paradigmatic case of implementing a Turing machine. And similarly in the Chinese room, it is merely through Searle's *understanding* of English, his voluntary choice to behave in a certain manner, and a number of highly disjointed physical processes (finding bits of paper in a certain location, turning the pages in the instruction manual, all mediated by the human agent) that the implementation takes place. Searle, as an intentional agent, is choosing to cause various things to happen in accordance with a set of rules that he chooses to follow. And Searle's intentionally characterized behavior is not something that we currently have any hope of ever being able to recast in terms of causal regularities at the purely *physical level of description*.

One might rejoin that, at least in principle, it's still theoretically *possible* to characterize the overall system purely in terms of natural laws and causal regularities, *a la* Dennett's [10] Martian superscientist, who doesn't require the intentional stance to predict human behavior. And while this may well be true in principle, I don't think it really helps, since *we* can't do so, and we're the ones interpreting Searle as performing a computation. Furthermore, we can let chance and randomness into the scenario. Suppose at each step in the computation Searle flips a coin, and will only follow the rule if the coin comes up heads. And suppose further that, for a particular run on an input question, the coin comes up heads every time and Searle successfully outputs the answer. He has still implemented the formalism, even though this outcome was not predictable on the basis of causal regularities or natural law.

In this case it counts as an implementation simply because what can be interpreted as the appropriate states in the procedure *occur* in the correct linear order. Questions regarding the mechanics of *how* they happen to occur are not relevant to answering the question of whether or not the procedure has been implemented. In the Chinese Room we can know that the procedure has been implemented without knowing how Searle himself (or his brain) manages to do the requisite internal processing and control his limbs in order to make the correct marks on the slips of paper. The physical how is a *different* question, and is not on the same level of analysis as that invoked when determining whether or not the desired mapping from formalism to physical configuration obtains. But this then critically loosens the requirements for counting a physical system as instantiating a program. As long as what can be described or interpreted as the correct sequence of states actually occurs, then the underlying mechanics of how this takes place are not strictly relevant.

The right sort of causal connections and regularities are needed if the instantiation in question is to be *fully automatic*, and if we want to be able to rely on the automatic device to perform systematically correct computations yielding outputs with the potential to supply us with new information. And although this is the engineering norm when constructing and interpreting computational artefacts, it does not exhaust the general space of possibilities. The causal requirements advocated by Chalmers constitute a sufficient but not a necessary condition – in the general case we must still allow for chance and human agency to play a role, as well as chronological sequences of states that are not themselves governed by overarching causal regularities. Hence strategy (ii$_a$) does not successfully block the argument for ↑MR*.

Chalmers' proposed *counterfactual* requirement (ii$_b$) is aimed at another apparently 'slack' feature incorporated by Putnam, *viz*. the mapping from formalism to physical system is defined for only a single run, and says nothing about what *would* have happened *if* a different input had been given. And it is objected that this is too weak to satisfy the more rigorous operational notion of being a 'genuine' realization. However, in response to Chalmers' (again quite natural) proposal, it is worth noting that for a physical system to realize a rich computational formalism with proper input and output capacities, such as an abstract TM, this will always be a matter of *mere approximation*. For example, any given physical device will have a finite upper bound on the size of input strings it is able to process, its storage capacities will likewise be severely limited, and so will its actual running time. In principle there are computations that formal TMs can perform which, even given the fastest and most powerful physical devices we could imagine, would take longer than the lifespan of our galaxy to execute. Hence even the fastest and most powerful physical devices we could envision will still fail to support all the salient counterfactuals.

It will never be possible to construct a complete physical realization of an abstract TM – the extent to which the concrete device can execute the full range of state transitions of which the abstraction is capable will always be a matter of *degree*. So in turn, the class of counterfactual cases on alternative inputs with which the realization can cope is by necessity limited – not all counterfactual cases will be supported by *any* physical device implementing a TM. And this renders the appeal to counterfactuals unavoidably *ad hoc*. The restrictive strategy demands that the mapping be able to support counterfactual sequences of transitions on inputs not actually given - but precisely *how many* inputs not actually given? One, two, twenty million? There is no clear or principled cut off point demarking 'genuine' implementations from 'false' ones in terms of counterfactuals. Consider a standard pocket calculator that can intake numbers up to, say, 6 digits in decimal notion. Is this a 'false' realization of the corresponding algorithm for addition, since it can't calculate $10^6 + 10^6$? It's an approximate instantiation which is nonetheless exceedingly useful for everyday sums. It will always be a matter of degree how many counterfactuals can be supported, where a single run on one input is the degenerate case. Where in principle can the line be drawn after that? It's a matter of our purposes and goals as interpreters and epistemic agents, and is not an objective question about the 'true' nature of the physical device as an implementation. In some cases we might only be interested in the answer for a single input, a single run.

Hence for a physical device to successfully 'perform a computation' is distinct from 'fully instantiating a computational formalism'**.** Performing a computation is an occurrent event, an actual sequence of physical state transitions yielding an output value, whereas instantiating a complete computational formalism is much more stringent and hypothetical, requiring appeal to counterfactuals, and as above, this will only obtain as a matter of degree. In light of this distinction, it is clearly possible for a physical device to successfully perform a computation *without* instantiating a complete computational formalism. And so again, tactic (ii$_b$) does not successfully rule out ↑MR* (nor weaker but extremely wide ranging versions of ↑MR).

4. Observer Relativity

One of Searle’s central negative claims is the allied notion that computation is not an ‘intrinsic’ property of physical systems – instead it’s founded on an observer relative act of interpretation. This basic point has been objected to in different ways, and is itself in need of some clarification. The latter part of Searle’s claim may seem to suggest that it is a purely capricious and subjective matter, and Ned Block [11] objects by pointing out that it’s simply not the case that anything goes. As an illustration, he notes that, although it’s possible to reinterpret an inclusive OR gate as an AND gate by flipping our interpretations of the values of ‘0’ and ‘1’, it is still not possible to reinterpret an *inclusive* OR gate as an *exclusive* OR gate. So although we have a great deal of latitude about how we interpret a material device, there are also very important restrictions on this freedom, and according to Block, this makes it a substantive claim that, e.g., the human brain is a computer of a certain sort.

Block’s position suggests that there are two important strands here that need to be separated. ‘Observer relative’ could mean that it’s totally subjective and anything goes, which is the claim he wants to deny. But it could also mean something more curtailed, *viz*., that computation is not an objective, observer independent feature of any given arrangement of matter and energy, and hence that no such description follows from nor is implicated by a purely physical account. Instead, the *attribution* of computational activity requires an observer to project the interpretation onto the system in question, and in this sense it is *observer dependent*. This doesn’t mean that the interpretation doesn’t have to satisfy various objective constraints supplied by the given characterization of the system. It simply means that, as Searle also says, it’s *not intrinsic* to the system itself, *qua* physical mechanism, and must be provided by the observer as an additional, outside ascription.

At this point an objector might reply that there are many levels of description that are not ‘intrinsic’ from the perspective of fundamental physics, but are nonetheless perfectly legitimate and scientifically respectable. For example, various arrangements of matter and energy configured in such a way as to perform some clear biological function. ‘Being a kidney’ is not an intrinsic property of the collection of molecules comprising a given instance of an organ of this kind, but this is still an objective and scientifically rigorous categorization. In response, I would argue that the attribution of computational structure is crucially disanalogous to cases such as this, which still trade on characteristics which are themselves essentially physical in nature. In order to be a kidney, a particular assemblage of material stuff must *do things* with other instances of material stuff that are characterized in terms of, e.g. the chemical composition of blood, waste products, filtering, etc. There is an objective, observer independent fact of the matter regarding whether or not a given configuration of matter performs the chemically specified functions required of kidneys, because biological functions are defined in terms of cause and effect relations in the physical world, and in stark contrast, computational realizations are *not*. There is a pronounced difference here between *actual* versus *abstract* characteristics which makes attributions of computational structure entirely observer dependent in a manner not shared by biological functions. The inputs to a computational system are essentially 'symbolic' rather than physical, where the material implementations of the symbolic or formal inputs must be *interpreted* as such by an outside agent, and where this symbolic interpretation is entirely *conventional* in nature. This marks a pronounced discontinuity in levels of description.

At the abstract, formal level, computation is essentially a syntactic phenomenon, and how we choose to interpret arrangements of matter and energy as constituting, say, tokens of an abstract syntactic type, and thus specifying an implementation of the basic computational vocabulary, is entirely independent of physical composition. For example, there is a more or less limitless diversity in the ways that material patterns and arrangements can be viewed as implementing the binary notation of '0' and '1', from ink marks on a piece of paper, stones placed in wooden boxes, patterns on old fashioned punch cards, electric voltages, beer cans positioned on rolls of toilet paper,... And as we've already seen, this scales up in the reverse ↑MR direction wherein the same stones placed in wooden boxes can be interpreted as implementing any number of distinct computational structures. Hence it's easy to *reinterpret* an inclusive OR gate as an AND gate – there is no objective fact to the matter as to which truth function is being computed. In light of Block's objection, some interpretations appear to be excluded (on the very pivotal assumption that the physical system itself is characterized as an 'inclusive OR gate' and not as something more fundamental), which seems to cast some doubt on ↑MR*. In the ensuing discussion I will not argue for or against ↑MR* (see Mark Bishop [12-13] for an interesting version of the claim) but instead confine my attention to the more modest, but nonetheless still vastly permissive ↑MR.

The non-intrinsic nature of computation would seem to follow as a direct consequence of the comparatively weak ↑MR, since ↑MR alone critically undermines the notion that any given computational interpretation of a physical device is somehow privileged or unique. As long as there are always at least two distinct interpretations, then there is no objective fact of the matter regarding *which* computation is 'really' being performed. And indeed, even if ↑MR* were to turn out false and *some* computational interpretations are excluded for a particular physical system, it remains the case that, as opposed to merely two, there are yet *arbitrarily many* distinct interpretations which are *not* excluded. Computation is not an intrinsic property of the physical device, but instead is founded on an act of human interpretation, and is usually tethered to issues involving design and engineering, relative to our purposes and interests. Indeed, discrete states themselves are idealizations, since the physical processes that we interpret as performing computations are in fact continuous, and this fundamental building block of digital procedures must be projected on to the natural world from the start. Thus implementation is always a matter of both interpretation and degree of approximation, and its usefulness will depend on our interests and epistemic needs (e.g. as above - how big a set of counterfactual inputs we want it to be able to compute).

It's certainly true that there is no pragmatic value in most interpretive exercises compatible with ↑MR and ↑MR*, e.g. *post hoc* attributions of single runs, or any case where we know the outcomes in advance of the interpretation. Physically instantiated computation is *useful* to us only insofar as it supplies informative outputs, which in most cases will come down to new data acquired as a result of the implemented calculation. Interesting observer relative computation takes place when we can directly read-off something that *follows from* the formalism, but which we didn't already know in advance and explicitly incorporate into the mapping from the start. That's the incredible value of our computational artefacts, and it's the only *practical* motivation for playing the interpretation game in the first place

Of course, this doesn’t mean that we cannot ascribe other interpretations to the same artefact – the difference is that in most cases the outputs will then be of no pragmatic or epistemic value to us. But this is still something relative to our human interests, practices and goals – the success of the strategy is based on objective features of the system (typically that we have designed and built), but this does not make computation itself intrinsic – it is still a purely conventional interpretation, an *abstract* level of description, and as such is neither canonical nor unique. Indeed, computation is no more an intrinsic property of a physical systems than is ‘being a sequence of inscriptions constituting a formal derivation of a theorem in first-order logic’.

In line with this logic/formal proof example, when I execute a particular TM computation by drawing the initial tape configuration on a piece of paper, then write down the succeeding tape configuration for each step in the computation according to the instructions in the machine table until I reach a halting configuration and stop, the physical states realizing the computation are a sequence of scratch marks on a two dimensional sheet of paper. There is nothing *physical* about these scratched in patterns that is intrinsically computational – indeed, the shapes could be interpreted in any manner one likes, or not at all. The computational interpretation of the physical scratch mark is purely *extrinsic*. And this is the same for syntactic interpretations in general – e.g. being an instance of the spoken English sentence ‘The cat is on the mat’ is not an intrinsic property of the sound waves constituting an instantiating utterance. Classical computation is rule governed syntax manipulation, and it is no more intrinsic to physical configurations than is syntax itself. And again, this was explicitly noted by Turing [2] long ago.

Physical systems, as such, are governed by *physical laws*, while formal systems are intrinsically *rule governed*. In the case of our computational artefacts, a system governed by physical laws must be deliberately engineered so that it can be interpreted as isomorphic in the relevant sense to a chosen rule governed formal system. 'Obedience' to physical law is an essentially *descriptive* matter and there is no sense in which mistakes or error can be involved – natural laws cannot be broken, and the time evolution of material systems is wholly determined (in the classical case at least) by the laws in question. On the other hand, 'obedience' to formal rules is an essentially *normativ*e matter, and there is a vital sense in which error and malfunction can occur. If my desk top machine is dosed with petrol and set on fire while still in operation, the time evolution of the hardware will remain in perfect descriptive accord with natural law. However, it will very soon fail to comply with the normative requirements of implementing Microsoft Word, and serious computational malfunctions will ensue. Being an implementation of Microsoft Word is a normative and *provisional* interpretation of the hardware system, which can be withdrawn when something goes ‘wrong’ or when the system is disrupted by non-design intended forces - being an implementation of Microsoft Word is not intrinsic to the physical structure itself.

5. Computation and Consciousness

Many versions of CTM focus solely on the functional analysis of propositional attitude states such as belief and desire, and simply ignore other aspects of the mind, most notably consciousness and qualitative experience – Fodor’s LOT is a classic case in point. However others, such as William Lycan [14], try to extend the reach of Strong AI and the computational paradigm, and contend that *conscious states* arise via the implementation of the appropriate computational formalism. This then invites reapplication of the Putnam/Searle line in the ↓MR* direction, with the rejoinder that every open physical system implements the 'appropriate computational formalism', so that consciousness is everywhere. According to this polemical strategy, rampant panpsychism follows as a consequence of CTM extended to the explanation of consciousness (which will be dubbed 'CTM+'), and this is taken as a *reductio ad absurdum* refutation of such views.

A natural line of defense for CTM+ is to invoke the counterfactual constraint discussed in section 3. to try and rule out unwanted implementations. Only highly sophisticated physical systems (such as brains, presumably) are able to support all the counterfactuals required to count as an implementation of the appropriate computational formalism, and hence the attempted panpsychic *reductio* is blocked. But as Maudlin and Bishop have argued, this is a highly dubious strategy in the case of conscious states, since these are essentially *occurrent* phenomena, and the invocation of non-occurrent process seems tantamount to summoning occult forces. While it's true that our *conceptual analysis* of 'causation', 'natural law', etc., invokes the notion of counterfactuals, this is an entirely different issue to the question at hand. Regardless of the abstract modal and other conceptual machinery required for philosophical analysis, it is still not the case that what is said to happen in a relevant counterfactually possible world has any *causal efficacy in this world*. So even though consciousness may involve causation, and counterfactuals are invoked in the philosophical analysis of the concept of causation, it still does not follow that conscious states can be affected in any way by things that *might* have happened *but didn't*. As Bishop rightly observes, the appeal to counterfactuals apparently requires a non-physical link between non-entered states and the resulting conscious experiences of the system.

Hence I would agree that for conscious states counterfactuals don't matter – it's only the *actual* run that could have any bearing, so that the foregoing attempted defense of CTM+ is unsuccessful. However, at this juncture a critic might reply that the occurrent character of consciousness should not in itself present a problem that wouldn't apply to any mental process, e.g. thinking (I would like to thank an anonymous reviewer for raising this and several other points pertaining to consciousness). And my view is that if 'thinking' is treated as an occurrent mental process, then the same observations as above *do* hold - no progression of an actual, temporally extended thought process can be causally influenced by what might have taken place but didn't. Counterfactual states of affairs still have no causal efficacy in the actual world.

Additionally, I would argue that the computational account of consciousness is fundamentally wrong in any case, and that even given the implementation of all purportedly relevant counterfactuals, this would still not constitute a sufficient condition for the presence of conscious experience. And this is because, as argued above, computation is not an intrinsic property of physical systems, and so is inherently unsuited to serve as the foundation for consciousness, which should instead be based on intrinsic properties of the brain as a physical mechanism. So to return to the example of thinking, there's an important distinction to be drawn between the conscious and the non-conscious aspects of thought. If by 'thought process' we're talking in CTM terms about the occurrent physical realization of a computational procedure, then this is an *abstract* level of description of the physical process realizing the computation, while it's the underlying causal powers of the physical medium that are responsible for the actual progression from state to state. And as just argued above, the intrinsic powers/properties of the physical medium should also be held responsible for the conscious aspect of thinking, and, contra CTM+, it is *not* the abstract computational procedure which sustains consciousness.

Unlike computational formalisms, conscious states are inherently *non-abstract*; they are *actual*, occurrent phenomena extended in physical time (whereas only the physical implementation of formal procedures is temporally extended). The computational camp makes a critical error by espousing ↓MR as a hallmark of their theory, while at the same time contending that qualitatively identical conscious states are maintained across wildly different kinds of physical realization. The latter is the claim that an actual, substantive and *invariant* phenomenon is preserved over radically diverse real systems, while the former entails the claim that *no* internal physical regularities need to be preserved. And this is because, as noted at the start of the paper when the notion of ↓MR was introduced, what unites different types of physical implementation of the same abstract formalism is nothing that they have in common as physical systems, but rather a structural isomorphism characterized at a higher level of description. Hence it's possible to implement the very same computational formalism using modern electronic circuitry, a human being executing the instructions by hand with paper and pencil, a Victorian system of gears and levers, as well as more atypical arrangements of matter and energy including toilet paper and beer cans. There are no internal physical regularities preserved over electronic circuitry, gears and levers, and toilet paper and beer cans. And hence there is no actual, occurrent factor which could serve as the causal substrate or supervenience base for the substantive and invariant phenomenon of internal conscious experience. The advocate of CTM+ cannot rejoin that it is *formal role* which supplies this basis, since formal role is abstract, and such abstract features can only be *instantiated* via actual properties, but they do not have the power to *produce* them.

The only (possible) non-abstract effects that instantiated formalisms are required to preserve must be specified in terms of their input/output profiles, and thus *internal* experiences, qua actual events, are in principle omitted. Hence (as has also been argued elsewhere: see Schweizer [15-16]) it would appear that the non-abstract, occurrent nature of conscious states entails that they must depend upon intrinsic properties of the brain as a proper subsystem of the actual world. It is worth noting that from this it *does not follow* that other types of physical subsystem could not share the relevant intrinsic properties and hence also support conscious states. It only follows that they would have this power in virtue of their intrinsic physical properties and *not* in virtue of being interpretable as implementing the same abstract computational procedure.

6. Observer Dependency and CTM

In the remaining discussion I propose that we restrict CTM to the schematic belief-desire framework commonly assumed to characterize intentional systems, and leave conscious experience out of its purview. Within this restricted context, I argue that it is possible to give an account of how this type of approach could, at least in principle, offer us an effective theoretical handle on the mind, even if we accept Searle's view that computation is not an intrinsic property of physical systems. The classical paradigm in cognitive science derives from Turing’s basic model of computation as rule governed transformations on a set of syntactical elements, and it has taken perhaps its most literal form of expression in terms of Fodor’s aforementioned Language of Thought hypothesis, wherein mental processes are explicitly viewed as formal operations on a linguistically structured system of internal symbols. So in the present discussion I will use the LOT as a very clear illustration of the classical approach, although the basic points made do not depend on the specific details of the LOT *per se*. According to the LOT, propositional attitude states, such as belief and desire, are treated as computational relations to sentences in an internal processing language, and where the LOT sentence serves to represent or encode the propositional content of the intentional state. Symbolic representations are thus posited as the internal structures that carry the information utilized by intelligent systems, and they also comprise the formal elements over which cognitive computations are performed. According to the traditional and widely accepted belief-desire framework of psychological explanation, an agent’s actions are both *caused* and explained by intentional states such as belief and desire. And on the LOT model, these states are sustained via sentences in the head that are formally manipulated by the cognitive processes which lead to actions.

Fodor plausibly notes that particular tokens of these LOT sentences could well turn out to be specific neuronal configurations or brain states. The formal syntax of LOT thus plays a crucial triad of roles: it can represent meaning, it’s the medium of cognitive computation, and it can be physically realized. So the syntax of LOT can in principle supply a link between the high level intentional description of a cognitive agent, and the actual neuronal process that enjoy causal efficacy. This triad of roles allows content bearing states, such as propositional attitudes, to explain salient pieces of behavior, such as bodily motions, if the intermediary syntax is seen as realized in neurophysiological configurations of the brain. Because the tokens of LOT are semantically interpretable and physically realizable, they form a key theoretical bridge between content and causation. In this manner, a very elegant (possible) answer is supplied to the longstanding theoretical question of how mental states individuated in terms of their *content* could be viewed as causes of actual behaviour, without violating fundamental conservation laws in physics. This is a specialized instance of the general solution to the problem of mental causation supplied by the computational paradigm which was noted in section 2. In this respect the LOT constitutes a very fine grained version of the general approach, wherein the internal processing structure involved explicitly reflects the standard belief-desire framework of traditional psychological explanation.

If we take something like Fodor’s LOT (for the sake of illustration), this is at least the basic type of highly sophisticated and complex computational structure relevant to CTM. Propositional attitudes themselves are abstract, dispositional states, and their functional/computational rendition could in principle be interpreted as a computational level of description of the activities of the human brain. Unlike occurrent conscious states viewed in purely qualitative terms, content laden propositional attitudes *are* highly dispositional in character, and for such abstract, dispositional states, the relevant counterfactuals pertaining to formal processing structure *do* matter. For example, if some agent is purported to instantiate the propositional attitude of *believing that snow is white*, then if given as input the question 'Is snow white?' an affirmative response such as 'yes' is required as output. But this is clearly not sufficient for implementing the belief that snow is white, and a host of counterfactual input/output patterns would also need to be supported, such as, if the agent had been asked 'Is snow green?' the output would have been 'No', if asked 'What color is snow?' the answer would have been 'White', etc., etc. So a version of Chalmer's counterfactual constraint is applicable in this more specialized case, but the reason is due to the prior conceptual requirements of instantiating a *propositional attitude* itself, rather than a computational formalism *per se*. These counterfactual constraints must be satisfied by any system held to properly sustain a belief or desire, and this is independent of the choice of CTM as the particular theory used to account for the underlying mechanics of *how* this takes place.

In line with the discussion in previous sections, even if, for the sake of argument, we grant that the brain can be interpreted as implementing Fodor's LOT, still, this would *not* be an intrinsic property of the brain as a biochemical mechanism. Obviously, there would be no scientific interest in a mere *ad hoc* mapping from LOT onto the brain, although in principle this may be possible, *a la* ↓MR*. Instead, for a theoretically substantive approach, there would be a myriad of pre-existing and empirically intransigent 'wet-ware' constraints that the mapping would have to satisfy, in order to respect the salient causal structure of brain activity as discovered by neuroscience. The largely independent body of functional and anatomical data from neuroscience would supply a host of highly non-trivial restrictions on how the physical system itself is characterized and what the material state transitions should look like that are interpreted as implementations of the abstract computational procedures. A scientifically significant mapping is *not* free to view the arrangement of matter and energy comprising the human brain in terms of brain-irrelevant aspects such as cosmic ray bombardment, gravitational fields, arbitrary molecular kinetics, etc. Instead, it must restrict itself to salient *causal* factors pertaining to the physical system's time-evolution when viewed *as a brain*. So a version of Chalmers' causal regularities between states would in fact obtain in this more regimented and specialized case, because, like a standard computational artefact, the brain must perform the implemented computational procedures automatically and reliably.

If a physical system when viewed as a brain were methodically interpretable as implementing the LOT, this would entail that the transitions between the various neurological states instantiating respective tokens of mentalese symbols, obeyed a causal progression in accord with the transformation of these symbols as prescribed by the abstract computational formalism. *If* this could be done, it would provide a scientifically fruitful and explanatorily powerful key to organic cognition, because it would constitute a unifying perspective tying together actual brain function and the standard belief-desire framework of intentional explanation. As above, the abstract computational level of description has the potential of supplying a bridge between content bearing propositional attitude states and causally efficacious physical mechanisms. And if a rigorous and explicit mapping could be specified, then it would appear that such a bridge had been found.

This abstract computational interpretation of brain activity would also need to mesh with the salient input and output capabilities that we want to explain via the attribution of internal cognitive structure, to explain, e.g. intelligent linguistic performance as in a Turing test. So from a purely physical perspective, the inputs and outputs are various forms of energy bombarding the organism's surface and emanating from it, and are not intrinsically computational either. But on the non-intrinsic cognitive level, these would be viewed as instances of written and spoken language, for example. And when interpreted as such, this non-intrinsic syntactic level would correspond to the internal processing activity triggered by the incoming energy pulse, interpreted as, say, a sentence in an English conversation. And this would have to conform with observable input and output patterns interpreted symbolically, to yield successful *predictions* of both new outputs given novel inputs, and predictions correctly describing new brain configurations entailed by the theory as realizations of the appropriate formal transformations required to produce the predicted output.

If successful, this would indeed be a case of real science, with at least two primary levels of empirical constraint satisfaction and experimental testing, to substantiate or refute the accuracy of the proposed mapping between formalism and brain structure. First there is the level of brute input and output profiles, which can be experimentally scrutinized in terms of outputs predicted by the formalism given new inputs. In this manner, a very wide range of counterfactual capabilities can become actualized over time. Second, the internal brain processes mediating input and output must preserve the interpretation of computational state transitions in accord with the formalism, and again, experimentation can allow many counterfactuals to be probed. Additionally, the linguistic *interpretation* of input and output signals would have to mesh with corresponding objects and states of affairs in the agent's environment, since in the human LOT case, we are studying and explaining an environmentally embedded system, and not a solipsistic syntax manipulator (such as a chatbot, for example). So if this CTM project were to turn out successful, then the LOT would be as powerful and well confirmed as a scientific venture could hope to be, and the objection that computation is still not an 'intrinsic' property of the brain would fade into irrelevance. It is in virtue of all of these factors considered together that human cognition could plausibly be accounted for in computational terms, and not simply in virtue of the brain being (in-principle) interpretable as realizing the LOT, by appeal to a mapping that ignores these crucial factors.

7. Conclusion

In accord with Searle, computation should be viewed as an extrinsic, observer dependent feature of physical systems. As such, it does not constitute a stable or independent natural kind. Various natural phenomena can be modelled or simulated using computational techniques, but this is to be distinguished from the notion that the system *itself* spontaneously instantiates and executes a computational procedure. Physical systems can be interpreted as realizing various abstract formalisms, but such realization is essentially a matter of approximation and degree, and interpreting a physical device as performing a computation is always relative to our purposes and potential epistemic gains. Configurations of matter and energy are governed by natural law, and computational *modelling* simulates this in a fundamentally descriptive manner. In contrast, formal procedures are essentially normative, rule governed structures, and in principle this interpretation can be projected onto natural systems in an almost limitless variety of ways. However, *interesting and illuminating* cases of computation realized in the physical world will come down to a question of engineering, either artificial or perhaps biological, in order to attain a robust, informative, non *post hoc* constraint satisfying *degree* of fit as a level of description for a physical system.

It is conceivable that the human brain has been biologically engineered such that there exist interesting, informative and predictively successful levels of computational description in the above sense. Propositional attitudes are at least *potentially* explainable in terms of functional/computational structure, which is abstract and multiply realizable. In contrast, conscious states, if they occur in a given implementation, should be explained in terms of the intrinsic physical properties of the medium of instantiation. In this manner, conscious experiences are properly seen as *hardware* states that may play an abstract functional role. This abstract role remains a legitimate software concern, and it must be preserved across divergent realizations. But the purely qualitative aspects of temporally extended conscious states should be seen as features of the particular material substrate that implements this role on a given occasion, and these features are not guaranteed to be preserved across divergent types of physical realization. Hence I would conclude that Searle's basic point against CTM is not well taken. Although CTM+ and a purely computational theory of consciousness are ruled out, in the case of propositional attitude states, the non-intrinsic status of computation does not trivialize predictively successful ascriptions of formal structure, and multiple realizability on its own, even in the extreme case of ↑MR*, does not render CTM empirically vacuous.

REFERENCES

[1] Turing, A. 'On Computable Numbers, with an Application to the Entscheidungsproblem, *Proceeding of the London Mathematical Society*, (series 2), 42, 230-265, (1936).

[2] Turing, A., 'Computing Machinery and Intelligence', *Mind* 59: 433- 460 (1950).

[3] Putnam, H., *Representation and Reality*, MIT Press, (1988).

[4] Searle, J., 'Minds, Brains and Programs', *Behavioral and Brain Sciences* 3: 417-424, (1980).

[5] Searle, J., 'Is the Brain a Digital Computer?', *Proceedings of the American Philosophical Association*, 64, 21-37, (1990).

[6] Fodor, J., *The Language of Thought*, Harvard University Press, (1975).

[7] Chalmers, D. J., 'Does a Rock Implement Every Finite-State Automaton?', *Synthese*, 108, 309-333, (1996).

[8] Chrisley, R. L., 'Why Everything Doesn't Realize Every Computation', *Minds and Machines*, 4, 403-420, (1994).

[9] Maudlin, T., 'Computation and Consciousness', *Journal of Philosophy*, 86, 407-432, (1989).

[10] Dennett, D. 'True Believers: the Intentional Strategy and Why it Works'. In A. F. Heath *Scientific Explanation: Papers Based on Herbert Spencer Lectures given in the University of Oxford*, Oxford University Press, (1981).

[11] Block, N., 'Searle's Arguments against Cognitive Science'. In J. Preston and J. M. Bishop *Views into the Chinese Room*, Oxford University Press, (2002).

[12] Bishop, J. M., 'Dancing with Pixies'. In J. Preston and J. M. Bishop *Views into the Chinese Room*, Oxford University Press, (2002).

[13] Bishop, J. M., 'Why Computers Can't Feel Pain', *Minds and Machines*, 19, 507-516, (2009).

[14] Lycan, W. G., *Consciousness*, MIT Press, (1987).

[15] Schweizer, P., 'Physicalism, Functionalism and Conscious Thought.' *Minds and Machines*, 6**,** 61-87 (1996).

[16] Schweizer, P., 'Consciousness and Computation.' *Minds and Machines*, 12, 143-144, (2002)