Noûs

NOÛS 00:0 (2016) 1–23
doi: 10.1111/nous.12142

Measurement and Computational Skepticism

ROBERT J. MATTHEWS
Rutgers University

ELI DRESNER
Tel Aviv University

Abstract

Putnam and Searle famously argue against computational theories of mind on the skeptical ground that there is no fact of the matter as to what mathematical function a physical system is computing: both conclude (albeit for somewhat different reasons) that virtually any physical object computes every computable function, implements every program or automaton. There has been considerable discussion of Putnam's and Searle's arguments, though as yet there is little consensus as to what, if anything, is wrong with these arguments. In the present paper we show that an analogous line of reasoning can be raised against the numerical measurement (i.e., numerical representation) of physical magnitudes, and we argue that this result is a reductio ad absurdum of the challenge to computational skepticism. We then use this reductio to get clearer about both (i) what's wrong with Putnam's and Searle's arguments against computationalism, and (ii) what can be learned about both computational implementation and numerical measurement from the shortcomings of both sorts of skeptical argument.

Introduction

For the past several decades, at least since the early 1960s, the concept of computation has played a central role both in philosophical accounts of the mind and in scientific research into cognition and brain function. Proponents of these accounts—'computationalists', as they have come to be called—claim not simply that mental states and processes are usefully modeled in computational terms (in the way, e.g., that weather systems are modeled computationally), but also, and more importantly, that such states and processes are in fact computational. Cognition, on their view, simply is computation, and as such the scientific challenge is to discern the particular functions computed by the computational brain.

A number of substantive questions and worries have attended the emergence and continued development of this computational conception of mind, questions and worries that a committed computationalist can hardly afford to neglect. Some of these questions concern the general nature of computation: What, exactly, are computational processes? Are they mathematical entities of some sort, and, if so, what distinguishes them from other mathematical constructions? What are the conditions under which we can justifiably say that a certain physical object computes a certain function, or a certain physical process implements (i.e. instances, realizes) a certain program or automaton? Does computation require representation, as Fodor (1975:34) famously maintains, and, if so, how do computationally defined states and entities come to have the representational contents they do? Other questions have to do more directly with the computational conception of mind itself: What sort of computational architecture does the brain employ? How are the various computations that cognitive scientists attribute to individuals implemented in the brain? Many of these questions and worries have the aspect of the usual growing pains that attend the development of any new theoretical framework, but the skeptical challenges to computationalism posed by Putnam (1988) and Searle (1990, 1992) seem altogether more serious, because they seem to challenge the very coherence of a computational conception of mind and brain. Both philosophers argue (albeit on somewhat different grounds) that there is no fact of the matter as to what mathematical function a given physical system is computing, that virtually any physical system can be construed as computing every computable function, and that as a consequence computationalism, as regards both mind and brain, is an utterly trivial thesis, completely devoid of empirical content.

Not surprisingly, the skeptical challenges raised by Putnam and Searle have been, and continue to be, the focus of considerable philosophical attention, with numerous philosophers undertaking to defend computationalism from these challenges, often in the context of a more general theory as to the nature and bounds of computation (Blackmon 2013, Chalmers 1993/2011, Chalmers 2012, Copeland 1996, Chrisley 1994, Godfrey-Smith 2009, Scheutz 1999, Piccinini 2007, Egan 2012). The proposed theories vary widely, and there seems to be little consensus as to what is wrong with Putnam's and Searle's skeptical arguments, even if critics all agree that something is indeed wrong with them.

In this paper we address the Putnam/Searle challenge through an appeal to a theoretical framework that has not until now been invoked in this debate, namely, numerical measurement theory, viz., the theory of numerical representation of quantities of physical magnitudes. Instead of attempting to defeat the challenge by presenting an elaborate theory of computation that insulates computationalism from the Putnam/Searle challenge, we offer a reductio argument that both defeats the skeptical challenge and also serves to diagnose the problem with the computationalist's favored conception of computation, a conception that makes it vulnerable to this challenge. Our key claim, based on the formal similarity between computational implementation and numerical measurement, is that if computational skepticism is warranted, then so too is measurement skepticism: if virtually any arbitrarily chosen physical object computes/implements every computable function, then by parity of argument virtually every arbitrarily chosen physical object has every quantity of every physical magnitude that it possesses. This conclusion is absurd, even by Searle's and Putnam's lights: the arguments of both presuppose the non-triviality and objectivity of numerical measurement. Thus we conclude that their skeptical arguments are defective. Furthermore, we use our analysis of the shortcomings of the skeptical arguments as applied to measurement to diagnose just what is wrong with the original arguments for computational skepticism.

The structure of our paper is as follows: In the first section we present a short review of the Putnam/Searle objections against computationalism and the main rejoinders put forward by critics to these objections. In the second section we outline the representational theory of measurement and explain the formal similarity between computational implementation and measurement-theoretic representation. In the third section of the paper we construct a measurement-theoretic analogue of the Putnam/Searle arguments, concluding that there must be something wrong with the latter. In the fourth section of the paper we diagnose the failure of these arguments as applied to measurement. Finally, in the fifth section we apply this diagnosis to the computational case.

1. The Arguments for Universal Implementation

Putnam (1988) and Searle (1990, 1992) argue that virtually every object computes every computable function, implements every program or automaton, and hence the claim that a particular object computes any particular function or implements any particular program/automaton is trivial. Let us dub their arguments to this effect 'arguments for universal implementation'.[1] In presenting these skeptical arguments, we begin with Searle's argument and then, following Sprevak (2012), present Putnam's argument as a generalized, more abstract version of Searle's argument.

Searle presents two different arguments for universal implementation, one an argument to the effect that if a physical system has sufficient complexity in physical states and processes, then it will be capable of computing any computable function (call this the 'sufficient complexity' argument), and the other an argument to the effect that irrespective of its physical complexity, any physical system can compute any computable function because the syntactic structure that realizes the computation is observer-relative (call this the 'observer-relativity of syntax' argument). Let's begin with the first of these arguments, since it has been the focus of most of the discussion of Searle's claims for universal implementation (Chalmers 1993/2011). Searle presents his 'sufficient complexity' argument as follows:

> On the standard textbook definition of computation, it is hard to see how to avoid the following results:
>
> (1) For any object there is some description of that object such that under that description the object is a digital computer.
> (2) For any program and for any sufficiently complex object, there is some description of the object under which it is implementing that program. Thus for example the wall behind my back is right now implementing the Wordstar program, because there is some pattern of molecule movement that is isomorphic with the formal structure of Wordstar. But if the wall is implementing Wordstar, then it is implementing any program, including any program implemented in the brain (Searle 1992:208–9).

Implicit in this argument are the following premises: (i) a physical system S implements a certain formally specified computational process C, if there is a structure-preserving mapping between (certain physical processes in) this system and the computational process, and (ii) for any sufficiently complex physical system S and any computational process C there can be found a physical process in S which can be mapped in a structure preserving way into C. Searle concludes that any sufficiently complex physical system can be assigned any computational characterization whatsoever, and therefore that computational characterizations—specifically, of the brain—are empty.

Searle (1992:209f) anticipates two sorts of challenge to the foregoing argument. Some critics will attempt to block his skeptical conclusion by (i) challenging the standard definition of computation, and by extension, his definition of computational implementation. They will propose ways of, as he puts it, 'tightening up the definition of computation' that will serve to rule out certain physical systems as computers and hence as computing one or another computable function (e.g., by requiring causal relations among program states, programmability and controllability, and 'situatedness in the real world'). Other critics, notably those inclined to agree with Searle that isomorphism is sufficient for computational implementation, will attempt to block, or at least raise doubts about, his skeptical conclusion by (ii) challenging the claim that for any sufficiently complex physical system there will be a structure-preserving mapping between the system and any computational process. These critics will ask how we can be sure that within any physical object there can be found realizations of any formal structure, such that there is guaranteed to be a structure-preserving isomorphism from the physical system to computational structure. Put another way, how can we be sure that there is sufficient complexity, indeed complexity of the right sort, to guarantee that there will be such an isomorphism? Searle dismisses both challenges by deploying his second argument for computational skepticism, what we're calling his 'observer-relativity of syntax' argument. Here is what he says:

> . . . But these further restrictions on the definition of computation are no help in the present discussion *because the really deep problem is that syntax is essentially an observer-relative notion. [. . .] [Computational processes] are not intrinsic to the system at all. They depend on an interpretation from the outside.* We were looking for some facts of the matter which would make the brain processes computational; but given the way we have defined computation, there never could be such facts of the matter. We can't, on the one hand, say that anything is a digital computer if we can assign a syntax to it, and then suppose there is a factual question intrinsic to its physical operation whether or not a natural system such as the brain is a digital computer (1992:209–10).

Searle's argument here seems to be this: the actual physical properties of the system that implements a computation don't dictate how we choose to individuate the particular physical states, properties, and processes that we take to implement the computation; this is something we impose on the system. But if this is so, then we are always free to individuate the implementing states, properties, and processes in such a way as to guarantee the existence of any given isomorphism; hence, on the assumed definition of computational implementation as a structure-preserving isomorphism, any physical object computes any computable function.

We see, then, that despite the attention that has been given to Searle's 'sufficient complexity' argument, it is his 'observer-relativity of syntax' argument that serves as the ultimate grounds for his claim that implementation is universal, and hence trivial. This observer-relative arbitrariness is never fully elaborated by Searle, and it is shown only in Copeland (1996) just how it can be achieved if we loosen some of the implicit constraints commonly imposed in standard implementations. Specifically, Copeland argues that if we think of Searle's argument in terms of register machines (the sort of computational architecture he chooses to illustrate Searle's argument), and we allow that the registers can 'change physical place' over the course of a computation, i.e., each register can be realized by different parts of the (would be) computing object at different stages of the computation, then there will indeed be a structure-preserving mapping of the required sort.

Somewhat earlier than Searle, Putnam (1988) offered a version of this same skeptical argument, but developed it for finite state automata (FSAs): he first argued that every physical object is affected by its ever-changing, boundless environment, and hence that at no two points in time can an object be said to be in the very same physical state. It follows from this that the object's states can in principle be grouped into types any way we want—in particular, they can be grouped in accord with the state-transition regularity of any FSA. Thus, suppose that a given FSA goes through a series of n formal steps while performing a certain computation. Considering any given object and its state transitions through a certain period of time, we can break this time-period into n segments, and then stipulate that any state reached by the object during one of the time-intervals, say the i-th, is an implementation of the FSA's i-th formal state. All these physical states of the object can be grouped into a single, disjunctive state that completely matches the corresponding formal state of the automaton. The non-repeatability of the object's physical states ensures us that this mapping relation created in this way is well defined. Furthermore, Putnam argues that the causal relations among the object's states match the formally defined transitions of the FSA: when the formal regularity of the FSA is such that one of its states brings about the next, the physical implementation of the first state will causally give rise to that of the other.

Putnam's argument differs from Searle's 'sufficient complexity argument' in the following way: The complexity in physical state types necessary to provide an empirical structure that can stand in a structure-preserving relation to computational states is introduced, not through an appeal to the inner structure of the computing object, but through a state-individuation scheme that individuates the states of the computing object in part by the effects of their ever changing environment. Thus Putnam's argument can dispense altogether with any reference to the inner physical structure of computational implementations. Rather, his argument rests on the non-repeatability of states of the whole physical universe, which induces a similar non-repeatability in the states of every object contained in it.

There have been a number of attempted rebuttals of these arguments, rebuttals that impose further constraints on bona-fide implementations with the aim of ruling out the Putnam/Searle deviant cases. It is not possible in the brief compass of this paper to discuss these attempted rebuttals in any detail. At best we can only present in outline several of the main challenges to Putnam's and Searle's skeptical arguments, in order to support our claims that (i) these challenges are quite various, (ii) there is thus no consensus regarding what is wrong with these arguments, and therefore (iii) our own contribution to the discussion, which is to follow, fulfills an existing need for clarity on this important foundational problem.

One line of argument invokes modality, arguing that Putnam and Searle's non-standard implementations of computational processes do not support any of the relevant counterfactuals required for computational implementation (Copeland 1996). In particular, Chalmers (1996, 1993/2011, 2012) invokes the modal constraint in order to argue that, as opposed to Finite State Automata, his internally structured Combinatorial State Automata (CSAs) are immune to trivialization.

Another line of argument focuses on causality, arguing that legitimate realizations of computational transitions must be implemented by causal processes in the computing device. As mentioned above, Putnam anticipates this objection, saying that the states of his would-be computing object are causally related. Chrisley (1994) challenges this claim, arguing that in fact the state transitions of Putnam's objects fail to satisfy a desideratum standardly required of causal processes (namely, the necessitation of one state by its predecessor).

A further line of attack against Putnam is that he does not 'take the physical seriously' (Schuetz 1999:170). Thus, according to Schuetz (ibid.), the grouping of physical states (as implementations of the same formal computational state) must be constrained by physical theory (rather than being arbitrary, as in Putnam), and similarly for transitions between states. Yet another, pragmatic line of argument focuses on intended function, arguing that a given object can be rightly said to implement a certain computation only if it is its function to do so (Piccinini 2007). Again, neither Searle's nor Putnam's implementations satisfy this requirement.

Some critics are prepared to concede an unavoidable degree of latitude in just what functions an object computes. Chalmers (1993/2011) is ready to concede that all objects implement some (usually simple) computations, and some objects may implement several highly complex ones. Shagrir (2012) argues that none of the proposed constraints on implementation is sufficient to rule out that, in some cases, an object may simultaneously implement different computations, even computations that are as complex as those that according to computationalists underlie human cognition. Godfrey-Smith (2009) concludes that implementations cannot be strictly divided into those that are legitimate and those that are not (like Putnam's), but are rather graded in their naturalness. Like Shagrir and Chalmers (as well as many of the other authors mentioned above), Godfrey-Smith is interested in the ramifications of his conclusion to the computationalist outlook in the philosophy of mind, ramifications which we cannot pursue here.

These different lines of argument are not mutually exclusive—indeed, attempted rebuttals of Putnam's and Searle's skeptical challenges often put forward more than one of them (Chalmers 1993/2011, Copeland 1996, Chrisley 1994, Piccinini 2007). Nevertheless, each of these argumentative lines proposes a distinct diagnosis of what is wrong with the challenges and how our conception of computational implementation (and also computation), should be reformulated in order to address them. This state of affairs leads to an impasse: As there is no agreement on the analysis of implementation, there is no common ground on the basis of which to reject the Putnam/Searle skeptical challenge.

The argument presented in this paper avoids this difficulty. On the basis of just the mapping requirement, as it appears both in the context of computational implementation and in measurement, we present a reductio ad absurdum of the skeptical claims against computation, and this without putting further constraints on implementation. Once the Putnam/Searle challenge has been met in this way, we turn to examine what these skeptical arguments can teach us regarding the nature of both computational implementation and numerical measurement.

2. Numerical Measurement and Computational Implementation

Discussions of computational implementation have generally paid little attention to other cases where formal entities, relations, and processes are, for various purposes, put in systematic structural relation to empirical (physical world) entities, relations, and processes. One central such case is numerical measurement, where real numbers are assigned to objects as measures of, e.g., their length, mass and temperature. Numerical measurement is particularly instructive in the present context, because it is a case in which the relation of formal structures to physical entities and systems is well understood. For over a century now, beginning in the late 19$^{\text{th}}$ century with Hölder and Helmholtz, theorists have studied the foundations of numerical measurement with an eye to explaining in precise terms just what numerical measurement consists in, what conditions, both formal and empirical, a valid system of measurement must satisfy, how different systems for measuring the same physical magnitude are formally related, and so on. In the second half of the 20$^{\text{th}}$ century, the pioneering work of Krantz et al. (1971–1989) brought together under the 'Representational Theory of Measurement' (RTM) several different threads of theorizing about the foundations of measurement.[2] In this section we present a brief description of this theory, identifying those essential features that it shares with computational implementation.

According to the RTM, numerical measurement consists in a structure-preserving mapping between a represented empirical relational structure and a representing numerical relational structure:[3] The former, represented structure is a domain of physical objects and certain qualitatively defined physical relations and operations on these objects; the latter, representing structure consists of the real numbers and certain relations and operations defined on them.[4] In the case of length measurement, for example, the empirical structure involves (i) a binary qualitative physical relation, 'longer than', that can be empirically verified by putting any two objects one next to the other and determining which extends beyond the other, and (ii) an appropriate physical concatenation operation for putting together any two lengthy objects to form a third. The counterparts of this relation and this operation on the reals are the real number relation 'greater than' and the real number operation 'addition', respectively. Crucial to the RTM's account of any measurement practice such as the measurement of length is the proof of a so-called 'representation theorem'—a proof to the effect that if the empirical domain satisfies a certain list of axioms, there exists a homomorphism (i.e., a structure preserving mapping) from this domain into the real numbers—that is, a mapping that assigns numbers to objects and numerical relations/operations to empirical relations/operations such that an empirical relation obtains among certain objects if and only if the corresponding numerical relation obtains among their numerical representatives. Proof of a representation theorem establishes that we can reason reliably and 'surrogatively', as Swoyer (1991) puts it, about objects and the quantities of physical magnitudes they possess by reasoning about their numerical representatives and their relations in the reals, something that is arguably the very point of numerical measurement.

It is easily provable, as one might expect, that for any measurable empirical structure there will be any number of structure-preserving mappings into the reals, each of which satisfies the conditions necessary to prove a representation theorem and thus, by definition, preserves the empirical content being measured. (Thus, for example, the Centigrade, Fahrenheit, and Kelvin scales are different mappings of one and the same empirical structure [temperature], as indeed are more exotic mappings, like the mapping of extensive magnitudes such as length into the multiplicative reals.) The existence of these different mappings raises the obvious question: how are they related? This question is answered by the proof of a so-called 'uniqueness theorem', which provides a characterization of the class of mappings, all of which satisfy a representation theorem for the empirical domain in question. On the basis of the uniqueness theorem, we know what transformations of scale preserve the empirical content of our measurements, knowing, e.g., that the empirical content of measurement scales for mass and length are invariant under multiplicative transformations, that the empirical content of measurement scales for temperature are invariant under linear transformations, and so on. Scales so related by a uniqueness theorem are equivalent in empirical content, although there may of course be practical reasons for preferring one scale over another.

This very brief overview of the RTM suffices for our purposes here, in that it is sufficient to establish the following fundamental similarity between numerical measurement and computational implementation: Both cases involve a structure-preserving mapping between a physical domain and a formal one. According to the RTM, numerical measurement (i.e., numerical representation) consists, indeed consists essentially, in such a mapping, such that whenever the existence of a homomorphism of the right kind is established, no further justification for numerical measurement is required. The foundations of computational implementation are not as well studied as the foundations of measurement—there is as yet nothing approaching the RTM in this domain—but, as was shown in the previous section, Putnam and Searle, as well as most of their adversaries, agree that implementation must involve a structure-preserving mapping between a physical system and a formal computation. Searle and Putnam assume that such a mapping is sufficient for implementation, whereby they derive their skeptical challenges, while those who oppose them add further restrictions that supposedly block these challenges.

Someone skeptical of the relevance of measurement-theoretic representation for our understanding of computational implementation might concede that both involve a structure-preserving mapping, but insist that there are nonetheless several important ways in which the two cases differ, specifically in the way they invoke a homomorphism between the physical and the formal. Dresner (2010) considers and dismisses these differences as irrelevant or merely apparent. Thus, for example, it is indeed the case that the formal structures involved in the two cases are different (numerical in standard measurement, symbolic in typical cases of computation), but this is of no substantial theoretical significance. (Measurement theorists such as Narnes [1985] consider measurement schemes where the representing formal structure is not numerical, but rather of a more abstract mathematical nature.) Also, contrary to what may seem to be implied by the term 'implementation', the direction of the mapping in both cases is the same—from the physical to the formal. (That the direction of the mapping is the same is suggested by Copeland's [1996] description of implementation as assigning computational *labels* to physical states, and attested to by the fact that while within a given implementation identical computational labels may be, and indeed in Copeland's example are, assigned to different physical entities and states, a physical entity or state cannot be assigned multiple labels within that implementation.) Further, it may seem that the two cases are importantly disanalogous inasmuch as in the case of numerical measurement the empirical domain is that of a class of objects, whereas in the case of computational implementation the empirical domain is a class of states, or properties of physical objects. But this is an insignificant difference inasmuch as on a second-order version of the RTM (see fn. 4 above) the empirical domain is a class of properties of physical objects. Finally, the phenomenon of scale-change may be thought to be a distinctive aspect of the measurement-theoretic case, but scale-change is in fact a mirror image of computational multiple-realizability: In measurement the objective is to keep track of the physical using the formal, and hence all homomorphic mappings from a given physical domain to the numbers will do, whereas in the case of computation the goal is to get formal results through physical processes, and hence given a specific formal process, any physical process to which it can be mapped in the right way is as good, formally speaking, as any other.

There does seem to be this difference: in the case of computation, our talk of implementation or realization seems to suggest an intimate metaphysical connection between a given physical system and the computation it realizes, while in measurement no one is (at least shouldn't be) tempted to suppose that there is an intimate metaphysical connection between the numbers and what they are used to measure. However, we should be skeptical, or at least wary, of assuming that there exists an intimate metaphysical connection between the implementing physical system and the computations that system implements. We would arguably be better off to think of implementation in just the sort of way we think of numerical measurement, viz., of computational descriptions as a way of *representing* certain objectively real physical properties (events, processes, etc.), though without reading back into the physical domain the formal computational properties of their computational representations.

3. Nonstandard Measurement Schemes

We are now in a position to raise the following key question: Can a case be made for universal, trivial measurability, on a par with the argument for universal, trivial implementation presented by Putnam and Searle? The rationale for raising this question should be clear: If the construal of computational implementation in terms of structure-preserving mapping opens the door for Putnam's and Searle's skeptical challenge, and if numerical measurement is also construed in such terms, then it is reasonable to expect that measurement is equally vulnerable to the sort of skeptical challenge that Putnam and Searle raise against implementation, and hence to the analogous skeptical conclusion that numerical measurement is equally universal and trivial. To put the matter succinctly, if Putnam and Searle are right about computational implementation, then contrary to what commonsense would suggest, it would seem that virtually *every* physical object, e.g., the desk before me, has every quantity of *every* physical magnitude it possesses. A surprising and disconcerting discovery to say the least!

It may seem, at least at first glance, that the RTM incorporates the conceptual resources to avoid this consequence. As noted in the previous section, at the core of the RTM stand representation and uniqueness theorems: The proof of such theorems both provides necessary and sufficient grounds for the application of measurement in any given context and characterizes the legitimate scale changes for a given empirical domain. This seems to be exactly what is required in order to distinguish between physical systems that satisfy the relevant axioms, and are therefore amenable to numerical representation, and those that do not. However, satisfaction of a representation theorem cannot by itself rule out the possibility that numerical measurement, like computational implementation, is universal and hence trivial. The reason is exactly the same as in the computational case: through deviant, Putnam/Searle-style regimentation, any physical domain can be molded into a measurement theoretic empirical relational structure of any kind whatsoever, and thus will be amenable to numerical measurement-theoretic representation. Put in somewhat different terms, the RTM takes empirical relational structures as given and then articulates what requirements they must satisfy in order to be mapped to the real numbers, thereby allowing for measurement. But the RTM does not address an earlier step, the step in which a certain 'chunk' of physical reality is conceptualized as an empirical relational structure of a certain sort—that is, as involving various objects, relations and operations satisfying a certain set of axioms—and it is at this step that the door can be opened for deviant cases of the Putnam/Searle type. Let us illustrate this point with an example, very close in spirit to Putnam's own example.

Our aim in developing this example is to present a Putnam-style argument for the universality—and hence triviality—of measurement. By 'universality' we do not mean merely that any physical object can be measured, or that for any object and any number there is a scale in which the latter is the measure of the former (for some measureable property, say length)—both of these (trivially true) universality claims are disanalogous to what Putnam shows, in that they do not involve mappings between whole structures.[5] Rather, choosing length as our example of a measurable property, we will show that, given any finite series of objects, each object in the series can be assigned any number whatsoever as a measure of its length (or its so called 'deviant length'—see below). This kind of universality (as opposed to those mentioned above) renders trivial any specific claim to the effect that a given array of objects can be numerically measured, on a par with Putnam's argument, where universality of implementation renders it trivial.

In order to establish this conclusion we cannot stick to the standard relational structure underlying length measurement. Rather, we will show how each arbitrary assignment of numbers to objects as length measures can induce a construction of a deviant empirical relational structure that can be mapped homomorphically to the reals. This is analogous to Putnam's line of reasoning, where an arbitrary assignment of formal computational states to the physical states of a given object induces a deviant grouping of these physical states, and this in a way that allows for a homomorphic mapping between these physical states and the computational states.

Thus consider a series of objects $x_1..x_n$, and a series of real numbers $a_1..a_n$, from which we derive the desired empirical relational structure. First, we get from $x_1..x_n$ all their (recursively generated) concatenations in the standard way (for length measurement). The resulting set of objects will be the domain of the empirical relational structure. Next, we assign numbers to all elements of the domain, again in a recursive way—$x_1..x_n$ are assigned, respectively, to $a_1..a_n$, and any concatenation of two objects that have already been assigned numbers is assigned the sum of these numbers. Finally, we define our deviant length comparison: For any two objects **a** and **b** in the domain, **a** is longer than **b** or equally long to it if the number assigned to **a** is greater or equal to the number assigned to **b**. (This is as opposed to standard length measurement, where an object is longer than another if, when both are put next to each other in the right way, the former extends beyond the latter.) Let us label the deviant comparison relation 'dlonger than' (as opposed to the standard 'longer than').

Having defined the deviant empirical relational structure we show that there exists a homomorphism h from this structure to the real numbers. The desired homomorphism is easy to define: For any object x in the empirical domain, h(x) is the number assigned to it in the construction of this domain, as described in the previous paragraph. The proof that this function, h, is indeed a homomorphism from the empirical structure to the reals is straightforward, as the empirical relational structure was constructed so as to allow h to be a homomorphism—for all relevant purposes the objects in the domain are related to each other exactly like the numbers assigned to them. Thus, for example, for any two objects x and y, x is dlonger than y iff, by definition, h(x) is greater or equal to h(y), as required, and similarly if z is the concatenation of x and y then, by definition, $h(z) = h(x) + h(y)$. (Another way to ascertain that this is indeed a homomorphism is the following:

Suppose we replace each object x_i with another, y_i, the standard length of which is a_i according to some given—say, the metric—scale. Surely the map from the y_i-s to the a_i-s can be expanded into a homomorphism—the standard, metric one. But the deviant comparison relation makes the x_i behave as if they were the y_i, and hence the map from the x_i [and their concatenations] to the a_i [and their sums] via the y_i is a homomorphism as well.)

Recall now that according to RTM the existence of a homomorphic mapping from an empirical domain to the real numbers is both necessary and sufficient for measurement. The deviant mapping we have found therefore consists in a measurement scheme for length, inasmuch as the initial n objects are assigned n arbitrary numbers, where the assignment meets the required necessary and sufficient conditions. This construction can be repeated for any series of objects and any series of numbers; hence length measurement is indeed universal in the desired sense. Similar constructions can be carried out for other measureable properties (e.g., temperature, mass).

It might be objected here that the above construction leads to absurd results. In particular, objects that have the same length could be assigned different numbers as their length measures, and objects that have distinct lengths could be assigned the same measure. For example, suppose our initial series consist of three objects, the first two of which are 1 meter long and the third is 2 meters long, and the numbers assigned to them (respectively) are 1, 2 and 2. The first two are assigned distinct measures while having the same length, and the last two are assigned the same measure while having distinct lengths. Thus, the objection goes, the homomorphism requirement is not satisfied after all. This objection, however, ignores the fact that according to RTM the relation 'having the same length as' is derived from the empirical relational structure that underlies length measurement—in particular, from the comparison relation that is part of this structure: for any two objects **a** and **b** to have the same length is simply for each of them to be longer or equal to the other. In the deviant cases we replaced 'longer than' with one of the 'dlonger than' relations, which yields different classes of similarity among the objects measured—induced by the arbitrary numerical assignments we started with. From the formal point of view of RTM, and this is the crucial point here, there is no difference between the standard and deviant cases. As an example, consider again the three objects mentioned above: These objects were grouped in one way according to standard empirical structures for length and differently according to deviant empirical structures, and the measures assigned to them in each case reflect this difference. Recall that the situation in Putnam's construction was similar: There the (allegedly non-repeating) states of an object were grouped in a non-standard way in order to allow for a homomorphic mapping to an arbitrary formal structure, and here distinct objects are grouped in a non-standard way for a similar purpose. But, it might be insisted, what is the use of such a measurement scheme, if it does not allow us to keep track of those length relations among objects that interest us (e.g., when a given object can be inserted into a given space), or to relate in physical laws length measures to measures of other properties? The answer, of course, is that the scheme is of no use whatever, exactly like Putnam's computation. However, the formal mapping requirement elaborated by RTM does not distinguish between useless and useful relational structures[6].

It might be objected that the deviant measurement schemes may be formally adequate, but there is no reason to say they measure *length*. What, it may be asked, ties the proposed arbitrary numerical assignments to length more than to other measureable properties? There are two ways to address this worry. One way is to weaken it—by arguing that *some* connection to standard length measurement is retained—and then to embrace it. Thus, note that the concatenation operation in the deviant schemes is the same as in the standard schemes, and it creates series of objects where the numerical assignments consist in *ordinal* measures of length. (The fact that the measure of a concatenation of two objects is greater than each of their measures represents the fact that the concatenation is longer, in the standard sense.) Also, the nearer such assignments are to the measures of the said objects on a given scale, the more similar the resulting scheme is to standard measurement. (In the limiting case we get a scheme of assignments that can be embedded in a standard scale.) Thus we have a host of deviant measurement schemes, each close to some degree to one of the standard measurement schemes of length (or any other measureable property, for that matter). The fact that RTM allows this zoo of measurement-schemes is perfectly in tune with our claim for measurement universality.

Another possible way to answer the challenge is to view numerical measures as assigned not directly to the objects in question, but rather to the instantiations of a certain class of properties in them—in our example, the class of lengths. (This is second-order measurement theory—see footnote 4 above.) According to this approach the deviant schemes clearly measure length and not anything else—the numbers are assigned to tropes of length, not to those of any other property. These tropes, though, are grouped differently than in standard cases, whence the absurd result that objects that are usually taken to have the same length can turn out to be dissimilar. (Here we take Putnam's idea one step further—an arbitrary classification not of states, but of tropes.)

Finally, it may be noted that in Putnam's construction the artificial connections between object-states and abstract computational states share with standard, bona-fide realizations an important feature, namely causality: In both standard and Putnam-style realizations the sequential aspect of the formal process is allegedly matched with a temporally couched, physical causal sequence. In measurement causality does not play such a direct role, and hence it could be argued that Putnam gets the universality of computational implementation while retaining more features of standard implementations in his non-standard ones. As an answer to this challenge we note that (i) there is disagreement whether causality is operative in Putnam's scenario as he claims it does (see section 3 above), and that (ii) our own non-standard schemes *does* retain some degree of modal extendibility, which Putnam's construction does not. (That is, the use of the concatenation operation in our non-standard schemes allows for counterfactual assignments of formal entities—numbers—to physical objects, while this is not possible for Putnam.) Therefore there is arguably no significant dissimilarity here between the two contexts in the degree to which the non-standard cases mimic the standard ones.[7]

We conclude, then, that the introduction of deviant relational structures allows for the construction of a plethora of measurement schemes. All of these schemes are sanctioned by RTM as formally adequate: like a purely formal theory of computational implementation, RTM does not preclude deviant (and hence trivial) measures of physical magnitudes. If such are to be precluded, then it has to be on other grounds.

4. Why the Skeptical Challenge to Measurement Fails: The Demands of Practicability

What should we make of these deviant measurement cases? Do they provide the basis for a skeptical challenge to numerical measurement? By parity with Putnam's and Searle's skeptical challenge to computational implementation, it would seem that they do. For if (i) non-standard cases of computational implementation provide the basis for a skeptical challenge to computational implementation, and if (ii) non-standard measurement schemes stand to standard ones in the same way that non-standard computational implementations stand to standard ones, then by parity of argument (iii) non-standard measurement schemes provide the basis for a skeptical challenge to numerical measurement. The Putnam-Searle slingshot, if sound, is therefore much more powerful than earlier realized: It undermines not only the notion of computational implementation, but also the much older and seemingly much more robust notion of numerical measurement.

So the question here is this: Should this skeptical conclusion be accepted? Commonsense says it should not. The practice of numerical measurement is so well entrenched in our everyday lives and scientific practices that it seems reasonable to expect (and hence to try to show) that there is something wrong with the parity argument for measurement skepticism. Rather than undermining numerical measurement, the argument seems to be a *reductio-ad absurdum* of the skeptical challenge to computational implementation: If this seemingly credible threat to computation implementation turns out to have such dramatic implications, then its plausibility is substantially weakened.

Furthermore, Putnam's and Searle's arguments themselves presuppose the objectivity of numerical measurement for physical magnitudes such as length and time: In their characterizations of the physical systems that are would-be computational implementations they rely on measurement. Thus by their own lights, if measurement is vulnerable to a skeptical challenge analogous to the one they raise against computational implementation, then their challenge, at least the way they present it (i.e., as relying on the robust objectivity of scientific measurement) is untenable. Thus we claim to have reached the first main goal of this paper—a reductio ad absurdum of the Putnam/Searle skeptical challenge that does not rely on any hypothesis regarding computational implementation beyond those that they themselves endorse.

But even if Putnam's and Searle's skeptical arguments fail to undermine both computational implementation and numerical measurement, they do point to a lacuna in the theoretical underpinnings of these notions. It is not enough simply to reject measurement skepticism and hence (by parity of argument) computational skepticism, and leave it at that. Instead, we need to diagnose what it is about the formal conception of numerical measurement, as articulated by the RTM, that leaves it vulnerable to skeptical challenge.

The skeptical conclusion can be avoided if the non-standard measurement schemes that lead to it can somehow be ruled out. But on what basis might this be done? The first step towards answering this question was made in the previous section, in the construction of the non-standard measurement schemes. It was noted there that the anomaly of these schemes was not due any special characteristic of the mappings each involved (between an empirical relational structure and the numbers). Rather, what was non-standard in these schemes was the very nature of the empirical relational structures that were to be numerically represented. Fundamentally, our formal conception of measurement, as articulated by the RTM, fails to preclude non-standard empirical relational structures. In particular, satisfying the formal axioms sufficient for proof of a representation theorem doesn't preclude non-standard empirical relational structures. So again our question, though now a bit more focused: On what basis are we to preclude non-standard empirical relational structures?

There are various possible answers to this question, some of which invoke notions already mentioned in the first section of this paper, where we canvased some of the arguments against the Putnam-Searle challenge to computational implementation. For example, one might note that like non-standard computational implementations, it is not clear how, if at all, non-standard empirical relational structures are to be extrapolated to the measurement of new objects or to counterfactual situations. In a word, it is unclear how non-standard empirical relational structures could support the robust numerical measurement practices that the RTM proposes to account for. But if this is all we can say, we have gained little ground by moving to consider measurement skepticism instead of computational skepticism. Fortunately, this is not the case. Although there are close ties and similarities between the computational and measurement-theoretic cases—indeed, such ties and similarities provide the underlying rationale for this paper—the context of measurement offers a clearer insight into just what it is about computational implementation and numerical measurement that gives rise to the skeptical challenge.

The very point of numerical measurement, we suggested in our exposition of the RTM in section 2, is to enable what Swoyer (1991) called 'surrogative reasoning', where one reasons using the real numbers and their numerical relations as surrogates for empirical objects and their qualitative relations (as specified by an empirical relational structure). The representation in the reals of these objects and their relations makes possible a practical precision and more importantly an ease of reasoning (for the arithmetically adept) that would otherwise be difficult if not impossible. Try to imagine, for example, how we could keep track of a non-numerical property such as weight except in numerical terms. It would be exceedingly difficult, and this difficulty points to yet a further virtue of numerical measurement: it also enables what might be described as a kind of *surrogative conceptualization*, whereby the arithmetically adept not only reason surrogatively but also conceptualize the objects and their relations in numerical terms, a practice that greatly facilitates our dealings with such objects, even if it runs the risk of blinding us to the fact that the numerical representatives of the relations are only the *images* of the real properties of things. Of course, non-numerical comparisons such as one object being 'heavier than', 'lighter than', or 'the same weight as' another could support a rudimentary form of comparative reasoning, one that would support judgments to the effect that, e.g., one object was heavier than another because the first was heavier than a third object that was in turn heavier than the second, but such reasoning could hardly support the cognitive and social practices in business, commerce and science that numerical measurement enables.

These considerations suggest that when we ask ourselves how we might rule out the deviant empirical relational structures that give rise to measurement skepticism, we should focus on what a successful numerical measurement *practice* (i.e., a successful practice of numerical representation) requires if it is to make possible the particular cognitive and social practices in which numerical measurement figures prominently. Such requirements need not figure among the formal axioms the satisfaction of which is a necessary condition for the proof of a representation theorem for a particular scheme of numerical measurement. Rather they are conditions that numerical measurement practice must satisfy if it is to support the sort of surrogative conceptualization and reasoning enabled by numerical measurement, something that is not at all guaranteed by the proof of a representation theorem.

If we ask just what such a successful measurement practice requires, we can distinguish at least two different sorts of requirement: first, those requirements that have to do with the nature of the objects being measured, and second, those requirements that have to do with the measurement practice itself. As an illustration of the first of these, consider our extant practices of measuring and reasoning surrogatively about lengthy objects. These practices would be extremely difficult, if not impossible, if the relative lengths of objects varied dramatically with changes of temperature, pressure, location, and so on.[8] It's not just that we can't think of some things, e.g., gases, in length terms, it's rather that if length were variable in these ways we would have no way of reasoning about those objects that we in fact think of as lengthy. It might seem that we could think about lengthy objects along the lines of the way we think about the volume of a gas, i.e., as relative to a specified standard temperature and pressure. But arguably we can do this for gases only because the lengths of most objects, at least in the range of environmental variables in which we normally conduct our length measurements, don't exhibit such variability. Even in the case of a gas, the variability we have to control for is relatively limited. Now the point here is that the demands of a successful measurement practice require that the objects measured themselves satisfy certain requirements, but these requirements are reflected only implicitly, if at all, in the formal axioms of the proof of a particular representation theorem.

A similar point can be made with respect to the conduct of measurement practice itself: If our practice of measuring lengthy objects didn't presume a certain fairly specific procedure for comparing and concatenating lengthy objects, e.g., that this be done by placing the objects to be compared or concatenated along a single straight line, then this measurement practice and the particular social practices it enables would never have gotten off the ground. Again, these are matters that aren't mentioned in the axioms, but this is hardly a failing of the RTM, because the objective of that formal theory is to give an account of the possibility of numerical measurement *as we in fact practice it*, and such procedures and conventions as make this practice successful as a social practice with certain implicit goals are simply presupposed by the formal theory. Of course, the RTM could undertake to codify these 'practical axioms' that insure successful measurement and include them among the formal axioms that figure in the proof of a representation theorem. Thus, for example, we might require that measurements be 'projectable', by which we mean (in a sense similar to what Nelson Goodman had in mind) that numerical measures of the same qualitative property or relation should be suitably invariant across a range of appropriately defined conditions and practices of measurement. For example, the length of most middle-sized objects should not change with local displacements across time and space, or as measured by one person or another. In so codifying these 'axioms' of practicability, the RTM would preclude certain deviant empirical relational structures. But it seems certain that with a bit of cleverness one could discover yet other possible deviant empirical relational structures satisfying such 'axioms', which is just to say that it is probably not possible to spell out all the ways in which an imagined measurement practice might fail to promote the surrogativity that we associate with actual measurement practice. If this is so, then we are forced to the conclusion that the RTM is concerned to provide a formal account (and justification) of numerical measurement *both as we in fact practice it and in the context of the cognitive and social practices in which such measurement in fact figures*. The RTM is simply not concerned with some practice superficially similar to numerical measurement, one in which numbers would be assigned willy-nilly to objects, but in a way that wouldn't permit the sort of surrogative reasoning that we associate with numerical measurement and which would not therefore support the particular cognitive and social practices that numerical measurement makes possible.

The deviant cases considered in the previous section, then, pose no threat to bona fide cases of measurement. The fact that there can be such deviant cases is fully consistent with the practicability of measurement *as we in fact practice it*. Such cases do not provide the basis for a general skepticism about measurement, even if they satisfy the RTM's formal requirements for measurability. Nor do deviant cases impugn the adequacy of the formal axioms, adverted to in the proof of a representation theorem, that guarantee the existence of a structure-preserving mapping of empirical relational structures into representing relational structures in the reals. The intent of the RTM is to characterize the necessary formal properties of measurement, not to characterize exhaustively the requirements that any successful measurement practice must satisfy, or to provide full-fledged conceptual analysis of each measurable property in terms of its underlying empirical relational structure.[9]

In focusing here on practicability as a way to rule out deviant cases of measurement, we are not committed to there being a sharp line between standard, practicable cases of measurement and deviant, non-practicable such cases. (Also, it is not our claim that surrogativity always fails in deviant measurement cases.) The distinction is surely vague and context sensitive. However, as is the case with other vague distinctions and concepts, this is fully consistent with the distinction we have drawn being coherent and productive, and with the cases we have considered being distinctively standard or deviant. (This is in tune with Godfrey-Smith's [2009] claim that the naturalness of computational implementations is a matter of degree.)

We argue, then, that the Putnam/Searle challenge to computationalism turns out not to undermine the theory and practice of measurement, but rather to bring to the fore an important aspect of measurement that is typically left unnoticed in philosophical discussions and scientific practice alike. Under the bedrock of the formal constraints that allow for measurement and justify it, there is a more pliable layer of pragmatic, non-axiomatized (and possibly non-axiomatizable) requirements that are necessary for successful measurement practice and yet are often left unmentioned.

5. Application to Computational Implementation

We can now return to computational implementation and ask to what degree considerations of practicability arise in this domain as well. Our aim will be to show both that computational implementation must similarly be practicable and furthermore that deviant cases of implementation such as Searle's and Putnam's are precisely cases that fail one or another practicability requirement while at the same time maintaining the sort of structure-preserving mapping from physical system to formal computational system that many have assumed to be sufficient for implementation.

As a first step, recall that what is non-standard in the Putnam/Searle deviant cases is not the structure-preserving mappings. It is rather the conceptualization of the physical systems themselves. The mappings themselves were fine—otherwise the Putnam/Searle constructions would not have served the skeptical purpose for which they were devised. Rather, the entities and states in the systems considered were either identified non-standardly (e.g., as switching location at different points in time, in Copeland's rendering of Searle), or grouped into types in a non-standard fashion (as in Putnam's example).

Computational implementation standardly makes possible a kind of surrogative reasoning and conceptualization about physical systems (very much in the way that numerical measurement standardly makes possible a kind of surrogative reasoning and conceptualization about physical magnitudes): just as in the measurement case, we reason surrogatively about the implementing physical system in formal computational terms, and, as in the measurement case, the structure-preserving mapping guarantees the validity of our reasoning in these terms. Of course, we are not reasoning here about qualitative relations among objects as in the measurement case. Rather we are reasoning surrogatively about certain state changes in the physical system, changes we are able to keep track of by thinking about them in computable function-theoretic terms with which at least some of us are familiar and comfortable. For example, to the question 'What is going on at this stage of early visual processing?', the computational theorist such as David Marr (1982) might answer, 'The visual system is computing the Laplacian of the Gaussian', a mathematical operation with which those adept with computer scientist's toolbox of computable functions are familiar and comfortable. And the computational implementation enables as well a kind of surrogative conceptualization, according to which the computationally adept can conceive of, and hence describe, the implementing physical system as computing a certain function, a conceptualization that enables the construal of what the physical system is doing in a way that makes sense of its operations from the particular rational epistemic perspective that the cognitive scientist (and her audience) brings to her research. Thus, returning to the same example, to conceptualize the visual system as computing the Laplacian of the Gaussian is to conceive of it as carrying out a familiar kind of smoothing operation on visual input. Here, too, then, practical considerations play a similarly crucial role as they do in measurement: the association between the physical and the formal is made for a certain practical purpose, sometimes to enable us to use a given object to make certain calculations that are of interest to us, and at other times to use the construal of an object as computing a function to make sense, in terms we understand, of the object's physical operations. It is some measure of the striking success of this surrogative reasoning and conceptualization that, as in the case of numerical measurement where we come to think of the object as actually possessing the numerical properties and relations by which we represent quantities of various physical magnitudes, so too in the case of computational implementation we come to think of a computing device as actually possessing the function-theoretic properties that the implementation attributes to it—e.g., as when we say 'The device is an adder.'

Similar points could be made about the point of conceptualizing a particular physical system as implementing a certain program, say a program for determining the shortest path between two points in a graph structure (think of navigation programs for cars) or an abstract machine, say a finite state machine or a universal machine of one sort or another. Conceptualizing a physical system in such terms can enable us to reason surrogatively about the state transitions of the system, enabling us to see operations of the system again in rational epistemic terms as, e.g., performing certain smoothing operations over a data array, making certain inferences, choosing among available options on the basis of certain desiderata, and so on. Such conceptualizing is in all these cases metaphysically harmless so long as we bear in mind that the relation of computational description to physical system is one of structural homomorphism: the device does not literally have the computational properties that we commonly ascribe to it, but of course it *does* literally have the physical properties of which the computational properties are surrogatively faithful images. And given this fact it is perfectly reasonable to describe a device as computing the Laplacian of the Gaussian, or being an adder, or implementing a program or abstract machine of some sort, or drawing certain inferences, since these are things that the physical system in question actually does *under the computational interpretation sanctioned by the implementation relation*.

Note that by acknowledging this last point we are not giving the game away to Putnam and Searle. That is, bona-fide computational implementation is presented here as depending on practical surrogativity, but, contrary to Putnam and Searle, we argue that it is not distinctive in having this feature—numerical measures have exactly the same property. Thus we are in the same boat with Chalmers (2012), for example, in opposing Putnam and Searle's view that computational properties of objects are subjective in a distinctive fashion: According to the above presented argument they are as objective as other physical properties, specifically those that allow for numerical measurement. However, in contrast with Chalmers we argue that standard numerical and computational depictions of physical reality alike have practical characteristics that distinguish them from non-standard ones. This distinctive point of view is arguably sufficient to serve as a sound basis for computationalist accounts of the mind, while acknowledging at the same time the practicability-relatedness of these accounts, which they share with the whole of science.

There are, of course, differences between the measurement and the computational implementation cases. For one thing, in the measurement case, the empirical relational structure was chosen for pragmatic considerations that are independent of its subsequent association with the representing relational structure in the reals: we are interested here in specific qualitative relations among objects or property tropes (depending on how you think of measurement); their numerical representation in the reals is simply a useful way of representing and conceptualizing these relations. In the computational implementation case, the situation is more complex. In the case of an artifactual computer (such as a laptop or even a very simple device like an adder or an *and*-gate), a particular physical system is chosen precisely because it is amenable to a particular computational construal, viz., as computing a particular function or implementing a particular program or abstract machine. In the case of a 'natural' computer of the sort computationalists take us (or our brains) to be, the difference is not so sharply drawn: as in the measurement case, the physical system is given, and the chosen formal computational description reveals what we, from the rational epistemic perspective that we bring to computational description, take to be the cognitively relevant operations of the system. Put another way, in the case of measurement the choice of both empirical relational structure and representing relational structure are grounded in the interests and purposes of those devising the measurement scheme (e.g., we are interested in the relative lengths of lengthy objects, and we devise a numerical scheme for conceptualizing and representing that property), whereas in cases of computational implementation of the sort that interest computational theorists, the empirical relational structure is constrained both by the system under study (viz., the brain) and by the formal computational representation scheme that we employ to conceptualize and reason about that system's cognitive operations, a scheme that both reflects and is constrained by the rational epistemic perspective that the computationalist brings to her research. But both sorts of case involve surrogative reasoning and conceptualization: in the measurement case, we want to be able to reason about, and conceptualize, qualitative relations and operations by reasoning about numerical relations and operations on the reals that preserve the qualitative relations and operations. In the computational case, we want to be able to reason about and conceptualize cognitive operations by reasoning about computational relations and operations that preserve explanatorily relevant cognitive relations and operations.

So the crucial requirement for practicability in both the measurement case and the computational implementation case is that of supporting and enabling the sort of surrogative reasoning and conceptualization characteristic of both. And this is precisely where the imagined computational implementations proposed by Putnam and Searle fail, despite the fact that they both manage to establish a structure-preserving mapping from physical system to formal computation. We can neither reason nor conceptualize surrogatively with either. This failure manifests itself in the shortcomings diagnosed in the Putnam/Searle constructions by extant rebuttals to their challenge: These constructions do not support counterfactuals, for example, nor do they rely on groupings of physical states that arise in a natural way from extant physical theory. However, according to the picture presented here these shortcomings are manifestations of the more basic failure, i.e. lack of surrogativity. (It should be stressed, though, that we do not claim to have presented here a fully worked-out practicability-based account of implementation, nor, ipso facto, to have pursued the relations of such an account with others presented in the literature. Such a further step is beyond the scope of this paper.)

Thus we conclude that the Putnam-Searle constructions no more threaten computational implementation than do deviant measurement schemes threaten numerical measurement. The fact that there can be artificially described physical systems that are unable to support the relevant sort of surrogativity is fully consistent with there existing other systems that are able to do so. The discovery, e.g., that a certain area of the brain can be construed and represented computationally in a way that supports this surrogativity is a non-trivial discovery of a *real* property of that area of the brain, just as the discovery that certain qualitative relations and operations within a class of objects can be construed and represented in a way that supports the kind of surrogativity that we associate with numerical measurement is a *real* discovery about the measured magnitudes.

Finally, the very close parallels between the role and purpose of representation schemes in both numerical measurement and computational implementation suggests that just as we need to be cautious when thinking in numerical terms about our weight and height, not allowing ourselves to be seduced into imagining that such of our properties are intrinsically numerical in character, whatever that might mean, computationalists need to be cautious in the metaphysical interpretation that they impute to their computational descriptions. Specifically, they need to be careful not to confuse represented properties with their formal images, yet all the while remaining thoroughly realist both about the specific relations and operations that these images are images of and about the amenability of these relations and operations to certain computational characterizations.[10]

Notes

[1] In what follows (and following both Putnam and Searle), we will not distinguish between the claim that every physical object computes every computable function and the claim that every physical object implements every program/automaton. Whatever difference there may be between the two claims, and this turns on just how one understands the notion of implementation, our arguments against the one apply equally to the other.

[2] For a history of the development of the theory of measurement, see Díez (1997).

[3] It is crucial to bear in mind that by ‘measurement’ here, one means not the actual practice of numerical measurement, i.e., measuring one thing or another and then perhaps reporting the results of these measurings in numerical terms, but rather just the possibility of faithfully *representing* in numerical terms certain properties of objects. RTM is not a theory of measurement practice, but a theory of the formal relations that underlie and make possible such practice.

[4] Second-order versions of RTM treat the domain of the represented structure as properties of objects, namely the magnitudes themselves, and the (second-order) relations and operations defined on these properties. Mundy (1987) and others have argued that this second-order version is in fact much more intuitive than the first-order version.

[5] It is for this reason that Blackmon’s (2013) argument against computational skepticism does not seem to meet Putnam’s challenge. Blackmon points out that the same object may have different velocities with respect to different frames of reference, but this falls short of what Putnam purports to show, i.e. that the same system may be homomorphically mapped as a whole to any formal system of a certain kind, viz., an FSA.

[6] Note that, according to RTM, the physical laws the relate measurable properties to each other are not part of the axiomatic requirements that are necessary for the measurement of each property on its own. Therefore the fact that these laws do not obtain in the case of our deviant measurement schemes does not render these schemes illegitimate.

[7] Similarly, it can be conceded that the class of numerical representing domains invoked by our deviant measurement schemes is not as rich as the class of computations that can be associated with any object according to Putnam’s deviant implementations, without such a concession being detrimental to the analogy between the two cases that we aim to establish.

[8] See Law (2005) for a discussion of hypothetical cases where the lengths of all objects (or all objects made of the same material) change concomitantly, in the same ratio, and how such scenarios bear on the foundations of measurement.

[9] See Bozin (1998) for a discussion of mass measurement that seems to be motivated by the latter of these two outlooks (that we find misguided).

[10] This paper has benefited greatly from discussions with a number of people, especially Jack Copeland, Frances Egan, Oron Shagrir, and the anonymous referees for this journal. Early versions were presented at a workshop on computability sponsored by the Israeli Institute for Advanced Study, Hebrew University, Jerusalem, and at a conference on ‘Inter-Level Relations in Cognitive Neuroscience’ at the University of Köln, Cologne. This research was supported by the Israeli Science Foundation (Grant No. 111/2009).

References

Blackmon, J. 2013. Searle’s Wall, *Erkenntnis* 78(1), 109–117.

Bozin, D. 1998. Alternative Combining Operations in Extensive Measurement, *Philosophy of Science* 65(1), 136–150.

Chalmers, D. 1993/2011. A Computational Foundation for the Study of Cognition, *Journal of Cognitive Science* 12(4), 323–357.

———. 1996. Does a Rock Implement Every Finite-State Automaton? *Synthese* 108, 309–333.

———. 2012. The Varieties of Computation: A Reply, *Journal of Cognitive Science* 13(3), 211–248.
Chrisley, R. 1994. Why Everything does not Realize Every Computation, *Minds and Machines* 4, 403–420.
Copeland, J. 1996. What is Computation?, *Synthese* 108, 335–359.
Díez, J.A. 1997. A Hundred Years of Numbers: An Historical Introduction to Measurement Theory 1887–1990: Part I: The Formation Period: Two Lines of Research: Axiomatics and Real Morphisms, Scales and Invariance, *Studies in History and Philosophy of Science*, 28(1), 167–18.
Dresner, E. 2010. Measurement-Theoretic Representation and Computation-Theoretic Realization, *The Journal of Philosophy* 107(6), 272–292.
Egan, F. 2012. Metaphysics and Computational Cognitive Science: Let's Not Let the Tail Wag the Dog, *Journal of Cognitive Science* 13(1), 39–49.
Godfrey-Smith, P. 2009. Triviality Arguments against Functionalism, *Philosophical Studies* 145, 273–295.
Fodor, J. 1975. *The Language of Thought*. New York: Thomas Y. Crowell.
Krantz, D., Luce, D., Suppes, P. and Tversky, A. 1971–1989. *Foundations of Measurement (Vol. 1–3)*, New York: Academic Press.
Law, S. 2005. Systems of Measurement, *Ratio* 18, 145–164.
Marr, D. 1982. *Vision*. Cambridge: MIT Press.
Mundy, B. 1987. The Metaphysics of Quantity, *Philosophical Studies*, 51, 29–54.
Narnes, L. 1985. *Abstract Measurement Theory*. Cambridge: MIT Press.
Piccinini, G. 2007. Computing Mechanisms, *Philosophy of Science* 74(4), 501–526.
Putnam, H. 1988. *Representation and Reality*. Cambridge, MA: MIT Press.
Scheutz, M. 1999. When Physical Systems Realize Function, *Minds and Machines* 9(2), 161–196.
Searle, J. 1990. Is the Brain a Digital Computer? *Proceedings and Addresses of the American Philosophical Association* 64, 21–27.
———. 1992. *The Rediscovery of Mind*. Cambridge, MA: MIT Press.
Shagrir, O. 2012. Computation, Implementation, Cognition, *Minds and Machines* 22, 137–148.
Sprevak, M. 2012. Three Challenges to Chalmers on Computational Implementation, *Journal of Cognitive Science* 13(2), 107–143.
Swoyer, C. 1991. Structural Representation and Surrogative Reasoning, *Synthese* 87, 449–508.