Does the brain encode information?

Jakub Mácha[1]

Abstract. Our common sense intuition says that when remembering something we store a piece of information in our memory, that is, in our brain. We can go further by claiming that the brain computes a program by processing information. The computational theory of the mind treats minds as information processing systems. These claims are the main tenets of contemporary cognitive science and neuroscience. Drawing on Searle's famous Wall argument, I argue, counter to these claims, that the brain cannot carry information by any reasonable definition of information (Shannon, semantic, algorithmic, quantum). Possessing information cannot be an *intrinsic* feature of any material object, including the brain. If the brain cannot store any information, it cannot compute any function or run any program, for non-trivial functions and programs presuppose the input information. Hence, the analogy between hardware/software and brain/mind is flawed in this respect.

1 INTRODUCTION

There has been always a human desire to explain the mind with the help of an analogy to the most developed technology of the time, and a desire to build a machine that would be (like) a human mind. Once it was a wax tablet, then it was a hydraulic system, a mechanic system, the steam engine, the phonograph, a camera, a hologram. And since the middle of the 20th century, it has been the computer.[2]

The analogy between the human mind and the computer is appealing. Some of its parts are clearly correct and worth further research. The central tenet of this analogy is that the human brain (or body) is analogous to the hardware and the human mind is analogous to software. There is some truth in this. For instance, there is the fact that the causal mechanical processes in the hardware, which cause programs to run, is analogous to the fact that certain (biological, molecular, causal) processes in the brain give rise to the appearance of the human mind. To put this in negative terms, not having software without hardware is analogous to no mind without a brain. This relationship is nowadays called *supervenience*: any change in a mental state implies a change in a physical state. What kind of processes and on which level are these responsible for the appearance of the human mind is one of the big questions facing cognitive science and philosophy today.

The core of the analogy is: Since computers process information, i.e. compute, brains must be able to compute as well. The analogy, thus, depends on the notions of computer and computation. In computer science, computers are usually modelled as Turing machines, the notion of computation remains somehow vague and ambiguous (cf. Searle's "standard textbook definition" of computation below). I shall focus on one component of the Turing machine: the *tape* which is the machine's memory. In Turing's words: "The machine is supplied with a 'tape' (the analogue of paper) running through it, and divided into sections (called 'squares') each capable of bearing a 'symbol.'" [2] In our analogy, the tape is a model of human memory. Here is an outline of the first part of my argument: If brains are computers, they must be able to process information. If they are able to process information, they must be able to store and retrieve it in their memory. In short, the brain must be able to encode information. Before we focus on the notion of information, which is the main topic of this paper, let us discuss a stronger instance of our analogy, which is the Computational Theory of Mind (CTM).

Although there are different versions of CTM, its core can be summed up as follows: the brain is *essentially* a computer (hardware) and the mind is *essentially* a program (software). Mental states and processes are essential computational states and processes. So, for instance, Jerry Fodor says: "Insofar as we think of mental processes as computational (hence as formal operations defined on representations) it will be natural to take the mind to be, *inter alia*, a kind of computer. That is, we will think of the mind as carrying out whatever symbol manipulations are constitutive of the hypothesized computational processes. […] we may thus construe mental operations as pretty directly analogous to those of a Turing machine. There is, for example, a working memory (corresponding to a tape)" [3, p.65]. David Chalmers claims "that computational descriptions of physical systems serve to specify their causal organization (or causal topology), and that cognitive processes and properties are organizational invariants that are determined by the abstract causal organization of the system." [4, p.3]

John Searle, in a series of thought experiments and arguments, argued that the brain-computer analogy and especially CTM are misguided. He came out with the following structure of his argument (which captures his Chinese room argument, although not in the form of a thought experiment):

– Axiom 1. Computer programs are formal (syntactic).
– Axiom 2. Human minds have mental contents (semantics).
– Axiom 3. Syntax by itself is neither constitutive of nor sufficient for semantic content.
– Conclusion 1. Programs are neither constitutive of nor sufficient for minds. [5, p.27]

By this argument, Searle maintained that a computer can produce a perfect simulation of the human mind's cognitive capacities like understanding a language without having these capacities. I take Searle's Chinese room argument to show, *inter alia*, that semantics cannot be ascribed to an agent on the basis of observing its behavior (e.g. on the basis of the Turing test).

---

[1] Department of Philosophy, Masaryk University, Arna Nováka 1, 602 00 Brno, Czech Republic, email: **macha@mail.muni.cz**.

[2] See [1] as an exemplary article: "The Memory Code" with the subtitle: "Researchers are closing in on the rules that the brain uses to lay down memories. Discovery of this memory code could lead to the design of smarter computers and robots and even to new ways to peer into the human mind".

As we have seen for Fodor, computational states are defined on representations, i.e. semantically. There are, however, versions of CTM that do not rest on semantics. That is, it doesn't matter what computational states represent or what they refer to. Chalmers clearly endorses this view: "Semantic notions such as that of representation played no role in that account." [4, p.6] The Chinese room argument therefore does no harm for such a version of CTM. What all versions of CTM have in common is that it is essential for the computation, and ergo for the mind, that the computer manipulates symbols, i.e. that computers are syntactical machines. Hence, Turing, Fodor and Chalmers, all accept Searle's Axiom 1.[3]

In 1992, Searle [7] published an even more radical argument: Any material object can compute any program; for instance, the wall behind me can be *interpreted* as computing the Wordstar program (a word processing program from the 1980s). If this were so, the fact that an object computes a program would be trivial. This argument is sometimes called a triviality argument against functionalism or computationalism about the mind.[4] Why can Searle claim that a wall is implementing the Wordstar program? He presupposes, what he calls the standard textbook definition of computation, which is equivalent to Axiom 1 above: that computation is defined syntactically as a manipulation of symbols. Now, given a syntactical description of an object, we can say "there is some pattern of molecule movements which is isomorphic with the formal structure of Wordstar".

One may wonder, however, what the formal structure of Wordstar actually is. Is it a particular run of the program when it is editing a particular text? I think we can assume that there exists such a description of the wall that is isomorphic to this instance of Wordstar. Or there is a single description of the wall that is isomorphic with the formal structure of any run of Wordstar? Not the actual run of the program, but also every possible run. Chalmers [9] criticized such triviality arguments (especially Putnam's version of them) against CTM that they cannot accommodate possible, but non-actual (counterfactual) situations. Definitely, there does exist a description of the wall where it is isomorphic with a particular run of Wordstar. However, given Chalmers' argument, there is no single description of the wall that would cover every possible run of Wordstar. There is, of course, a single description of any desktop computer that covers every run of Wordstar. Or in today's terms, there is a natural description of a PC so that it is isomorphic to every possible run of MS Word 2016. This amounts to the almost trivial fact that MS Word 2016 can be run on every contemporary PC, but not on every wall.

Did, then, Chalmers provide a knock-down counter-argument against triviality arguments? In my view he didn't, because the notion of computation is not the main focus of Searle's argument. I think the central claims of the argument are: "Syntax […] is not intrinsic to physics. […] Syntax and symbols are observer relative." [10, p.35]

In short, the Chinese room argument says that semantics is observer relative; and Searle's Wall argument says that syntax is observer relative as well. Both aren't intrinsic to a physical system like a brain. Because the notion of computation involves either syntax alone or both syntax and semantics, computation cannot capture the intrinsic features of a brain.

I'm going to argue that we can arrive at these claims without invoking the notion of computation or implementation. I will employ the notion of information instead. As I have argued above, computation is in fact information processing. The notion of information is thus a part of the notion of computation in the following sense: If information is not intrinsic to a system, then computation is neither. The advantages of this approach are that information doesn't involve any temporal or modal dimension that is liable to Chalmers' argument (i.e., that storing or encoding information means encoding any possible piece of information at any time, given physical limits) and that the notion of information is clearly defined. I'm going to consider every relevant notion of information in order to show that they involve syntax or semantics.

There are many conceptions of information. Primarily, we have to focus on those conceptions that are relevant in computer science. These are, no doubt, Shannon information, algorithmic (or Kolmogorov) information and more recently, quantum information. Moreover, I will consider semantic accounts of information as well.

2 SHANNON INFORMATION

Shannon [11] defines the information of a message or symbol S as $-\log_2 p_S$ where p_S is the probability of the occurrence of S. If the probability of the occurrence of S is 0.5, i.e. that it is equally probable that S occurs and that S does not occur, then S generates information of 1 bit ($= -\log_2 0.5$).

First, we have to make clear what counts as a message or symbol. It can be a letter, a binary code, a continuous function, a vector of several functions or any combination of these [11]. In fact, any object can generate information. Let us take for example a toaster that can be in two states: on and off. Let the probability of being on be p and the probability of being off be 1–p. The average amount of information the toaster generates is then $-p.\log_2 p - (1-p).\log_2(1-p)$. If a toaster can generate information, then a neuron can do so as well. Let us assume that a neuron can fire an action potential, and that the probability of its firing is p. Then the neuron can generate information of $-\log_2 p$.

Two interrelated points are our focus now:

(1) Every phenomenon can be interpreted as a symbol and as a bearer of information. I'll call this fact a *syntactic* point.[5]

(2) It does not matter what this symbol means or refers to. This is a *semantic* point.

Let us now focus on the syntactic point. Every phenomenon (situation, state of affairs, event) can be interpreted as a symbol, i.e.

---

[3] Objection (by J. Haugeland [6]): Computers run programs which are sequences of instructions. They are able to interpret these programs according to an encoding. They do something more than just manipulating symbols. Hence, computers have a syntax as well as semantics. My reply: When a computer runs a program, it gets the program on its input. In abstract terms, a universal Turing machine gets a program on its input tape. There must be an encoding of this input data, i.e. information. Then, however, the computer or the universal Turing only manipulates these symbols. Computational semantics can be, thus, accounted for in syntactic terms.

[4] Putnam [8] came up with a similar and more detailed argument. Nothing in my account rests however on Putnam's version of the argument.

[5] Cf. Dretske [12]: "Any situation may be taken, in isolation, as a generator of information." Or Searle [7]: "The ascription of syntactical properties is always relative to an agent or observer who treats certain physical phenomena as syntactical."

as a bearer of information. This means that there must be an external agent that decides which phenomena are taken or recognized as symbols. "S" may by a symbol, or a random shape on the screen. Lower case "s" may be the same symbol or another symbol. We can consider the toaster as capable of being in two states: on and off. The toaster can encode any number as its temperature. It can be a natural number or even a real number (up to a certain accuracy of measurement). Finally, we can say that a neuron is either firing (emitting its action potential) or not firing. Then a neuron would encode one bit of information (like the toaster capable of being either on or off). There are, however, many ways how to interpret the state of a neuron, i.e. there are many neural encoding schemes.[6] We can look at the duration, the amplitude, or the shape of actions potentials. Or, we can consider the frequency of neuronal firings (spike firings). More importantly, we can, instead of a single neuron, take blocks of neurons as the basic unit of information encoding.

As I have argued above, every computation presupposes an information encoding. In fact, even a stronger claim can be put forward here: that different information encodings lead to different computations on the same physical states. Here is an example of such a case taken from Shagrir [14]:

> Consider a physical system **P** that works as follows: It emits 5-10 volts if it receives voltages greater than 5 from each of the two input channels, and 0-5 volts otherwise. Assigning '0' to emission/reception of 0-5 volts and '1' to emission/reception of 5-10 volts, the physical gate implements the logical *AND-gate*: '0','0' → '0'; '0','1' → '0'; '1','0' → '0'; '1','1' → '1'.
>
> Let us suppose it turns out that flip detectors of **P** are actually tri-stable. Imagine, for example, that **P** emits 5-10 volts if it receives voltages greater than 5 from each of the two input channels; 0-2.5 volts if it receives under 2.5 volts from each input channel; and 2.5-5 volts otherwise. Let us now assign the symbol '0' to emission/reception of under 2.5 volts and '1' to emission/reception of 2.5-10 volts. Under this assignment, **P** is now implementing the *OR-gate*: '0','0' → '0'; '0','1' → '1'; '1','0' → '1'; '1','1' → '1'.

Two different encodings are considered in this example. On the first encoding, the emission/reception of 0-5 volts encodes '0' and the emission/reception of 5-10 volts encodes '1'. On the second encoding, the emission/reception of 0-2.5 volts encodes '0' and the emission/reception of 2.5-10 volts encodes '1'. Let us assume that voltages are intrinsic properties of the physical system (they don't need to be, see below). These encoding are, however, *assigned* by an external agent. Thus, they are not intrinsic to the physical system.[7] A mere choice of the encoding can determine what function is computed by the system, whereas the underlying physical states are the same in both cases.

Shagrir concludes from his example that "the very same physical system **P** simultaneously implements two distinct logical gates". But why simultaneously? The two encodings are mutually exclusive. The system implements either the *AND-gate*, or the *OR-gate*, depending on the choice the encoding. Blackmon [16] proposes to treat the implementation of a computation as "a three-place relation over causal structures of physical things, programs, and mappings". This mapping which maps causally related physical states to symbols is our information encoding. Given this terminological clarification, no physical system can simultaneously implement different computations.

All these considerations lead us to the following: Does information encoding depend on the encoding agent, i.e. is encoding observer-relative? Advocates of CTM have to come up with an account of information that is independent of the encoding agent. It is clear now, however, that Shannon-information is not. This leads to the following conclusion: If one is interpreting a physical state (S, a toaster, a neuron) as a bearer of (Shannon-)information, the agent has to come up with an encoding, i.e. a method of recognizing physical states as symbols.

This is, however, precisely Searle's point: "For any program and for any sufficiently complex object, there is some description of the object under which it is implementing the program." [6, p.208] Now amending this point to the present case: For any information content and for any sufficiently complex object, there is some encoding under which the object encodes this information content.

3 SEMANTIC INFORMATION

The semantic point has sometimes been seen as the reason why Shannon's theory has been accused of neglecting the semantic aspect of information [17, p.1]. It does not matter whether symbol S means "It is a day" or "The number of my hairs is even". And it does not matter whether the fact that the toaster is on means that it is heating or that the sky is blue. The firing of a neuron may mean that this neuron has been stimulated (for example, by another neuron or, in the causal chain, by an object in the visual field). The connection between a symbol and what it means or refers to, can be natural or conventional. A natural connection is a law-like connection between two states (without any intermediate agent), the most common being a causal regularity (but other regularities are also conceivable, e.g. visual resemblance). A conventional connection between two states is a semantic function that mediates between these states and is set up from outside by an agent (or agents). There is a natural connection, for example, between the number of growth rings on a stump and the tree's age (this is an example of environmental information). There are also connections that are partly natural and partly conventional, e.g. visual resemblance may depend on a perspective which presupposes an external agent.

There are various accounts of *semantic* information, e.g. Bar-Hillel and Carnap [17], Dretske [12], Floridi [18], which compute the amount of information according to the probability of obtaining situations, states of affair, events, possible worlds. In contrast to these accounts, what affects the amount of (Shannon-)information are the relative probabilities of the occurrence of a symbol. We may learn about these probabilities by considering what these symbols mean (e.g. we can research the relative probabilities of letters, words, expressions in a language). The semantic accounts of information are, however, not relevant in computer science. When we say that a hard disk has the capacity of 80 GB, all that matters are the probabilities of its states (which are all equal in

---

[6] See [13] for an overview.

[7] This point is indicated in Dennett's 1969 dissertation: "The electrical characteristics of an impulse sequence, or the molecular characteristics of a nerve fibre could not independently determine what the impulses *mean* [i.e. the semantic point], or what *message* [i.e. the syntactic point] the nerve fibre carries" [15].

this case). It doesn't matter what is stored on the hard disk. Whatever content is stored there, its capacity remains unchanged. In sum, computer science works neither with environmental semantic information, nor with conventional semantic information.

4 ALGORITHMIC AND QUANTUM INFORMATION

There are two other accounts of information that are relevant in computer science: algorithmic information and quantum information. I am now going to argue that neither of these accounts can do without encoding.

The motivation for the algorithmic information theory (Solomonoff [19], Kolmogorov [20], Chaitin [21]) was the problem of assigning a priori probabilities to symbols (in Shannon) or to situations (in Bar-Hillel and Carnap). The problem is that there is no universal method of assigning these probabilities. The main idea is as follows: The *information content* of a string x is the length of the shortest program that gives x on its output on a given reference universal Turing machine. Formally: $\mathrm{AI}(x) = \min_p\{l(\mathrm{p})\colon U(p) = x\}$, where $l(p)$ gives the length of p in bits, U is a universal Turing machine and $U(p) = x$ means that program p, when run on U outputs x. Intuitively, this definition gives us the length of the shortest description of a given string, i.e. the length of the best compression of it.[8]

There might be the hope that algorithmic information would overcome the problem with arbitrary encodings which fuels the core of Searle's Wall argument. Algorithmic information strives after the best encoding of a given string. Instead of choosing an arbitrary encoding, algorithmic information takes the best one. This move allows us to overcome the arbitrariness of the encoding of semantic information. In other words, it allows us to cope with the arbitrariness of encoding within the semantic point (2) above. The present argument is, however, based on the arbitrariness of the encoding of symbols (the syntactic point (1) above). The algorithmic information is the amount of information of a *string*. It is left open here what actually counts as a string, however. Hence, there must be an encoding that says which phenomena can count as symbols. This encoding is however as arbitrary as in the case of Shannon-information. And there is another additional problem with algorithmic information: and that is that it is always relative to a referential universal Turing machine.

Let us turn now to quantum information which is another account of information that has been successfully applied in computer science as well as in the brain sciences. Quantum mind is the idea—made famous by Roger Penrose—that considers that genuine quantum processes can (at least partly) explain the relation between mind/consciousness and the brain (cf. Penrose [22]; Beck and Eccles [23]; see Atmanspacher [24] for an overview). There have already been numerous ways proposed for how quantum processes might cause or correlate consciousness. They are all very intricate, but also controversial. The present perspective is slightly different. Can the mind/brain problem be explained with the help of an analogy to quantum processing and computers? If so, we have to ask how quantum information can be stored or encoded in a brain while being considered as a quantum system.

Quantum information is much closer to Shannon-information than to algorithmic or semantic information. The basic idea is that information is represented by quantum states which can be in a *superposition* (i.e. more states can be added – superposed together). There are many peculiarities of such states that go beyond classical mechanics, e.g. quantum states cannot be cloned (duplicated), destroyed, without the loss converted into classical states; they can be, on the other hand, teleported, i.e. transported into other underlying physical substratum. It seems, hence, that quantum mechanics allows what classical physics does not: to transfer a quantum state into another underlying medium. My response to this is that even if this were possible (and we are far from knowing it), there is no need to invoke the notion of information. There is no added value from speaking of quantum information instead of speaking of quantum states in explaining quantum processes in the brain.

In the end, we can reemploy the argument from section 2: A quantum state must be recognized or taken as a bearer of quantum information (the syntactic point (1) above). For example: quantum states A and B are realized in the spin of certain electrons. This is, however, encoding similar to the encoding of classical Shannon-information. Without such an encoding, there would be no use of invoking the concept of quantum information.

5 POSSIBLE OBJECTIONS

The Homunculus objection: The encoding agent is in the brain or in the mind. It's called a homunculus. Such a homunculus is a mind-like entity and we would need an independent explanation of its nature and skills in order to avoid a potential infinite regress. The other problem is that even if there were a homunculus that could come with the encoding, we (as external observers) wouldn't have any epistemic access to this encoding. Dennett [25] proposed that there is a way of getting rid of the homunculi regress. As complex computational operations can be analyzed or decomposed into simpler operations so the homunculus in the mind is a substantially simpler (or stupider) entity. The regress reaches its bottom when the homunculus performs simple operations. Then there can be a mechanistic account of such a way of functioning. Dennett says explicitly that this machine may be "a neurone, or a flip-flop in a computer that only has to remember 0 or 1 as its only expertise" [26, pp.87–88]. However, as we see from this brief quotation, even the machine-like homunculus operates with symbols, i.e. with bearers of information. Even if this homunculus were as simple as the neural gate discussed above, there would have to be another agent that would supply an information encoding. Converting the simplest computer into a machine thus fails. The homunculi regress is, hence, not blocked. As Searle puts it: "the only way to get the syntax intrinsic to the physics is to put a homunculus in the physics" [10, p.29].

Let us move now to another attempt of putting syntax and semantics into the physics, or better into say, in nature – to the evolutionary and teleological theories of mental content, sometimes called teleosemantics. Dretske's semantic account of information may be seen as a kind of teleosemantics. Let us first look at Ruth Millikan's influential account. Millikan claims that human brains (as well as other organs and parts of bodies) have biological purposes which are "functions that they have historically performed".

[8] The aim of compression programs is to be close to this best rate.

[27, p.330] Such functions are, in Millikan's account, the results of meme selections (there are other accounts of selections and other accounts of the origins of biological functions such as learning or conditioning).
If evolutionary design is responsible for the selection of a computational function, then the same evolutionary mechanism must be responsible for the selection of the right information encoding. If the encoding were undetermined, then it could be undetermined likewise whether the *AND-gate* or the *OR-gate* has been selected. The underlying physical states and causal structures implementing the one gate or the other are thus identical. An advocate of a teleological explanation of computational functions is, thus, committed to the claim that the evolutionary mechanism is able to select between two different features that have physical and causal powers. This means that the evolutionary mechanism could be able to distinguish non-naturalistic properties.
This initial objection against my account from a teleosemantic position can be turned into an objection against teleosemantics itself. This objection has been discussed under the label "functional indeterminacy problem", cf. Dennett [28] or Fodor [29].

Simon Blackmon recently objected to Searle's Wall argument that on the three-place notion of computation (see above), the universal realization is blocked, and the conditioning on the existence of an encoding possesses no real problem to still seeing a scientific character of the computational investigation of the mind. This argument is by an analogy to motion and location in physics and to meaning in linguistics. In physics, motion and location is assigned relative to frames of reference. In linguistics, meaning are assigned relative to an interpretation. Physics and linguistics both hold nevertheless the status of sciences. Computation is assigned relative to an encoding. Computational cognitive science holds a scientific status too. Information encoding is, thus, like a frame of reference or an interpretation. If physics is able to discover the intrinsic features of the world, then cognitive science is able to discover the intrinsic features of the brain.
This is a serious objection. I accept most of Blackmon's presuppositions: On his analysis of the notion of computation, the universal realization is, indeed, blocked and information encoding is like a frame of reference in physics or a linguistic interpretation. However, motion or location are not intrinsic features of a brain. Suppose that brain research discovers the brain centrum A is located 9 cm from the brain centrum B. This doesn't imply that the number 9 is an essential feature of the brain. We can use another unit of length and say that the distance between these centrums is 3.54 inches. By the same argument, neither voltages are intrinsic features of the brain. Arguably, different units of length don't bother physics so much, for there are simple methods of converting one unit into another one; there is no room for any indeterminacy.[9] Things get more complicated when we consider meanings and interpretations. As we know from Quine's and Davidson's well known arguments on linguistic indeterminacy, to translate one language into another or to interpret one language in another can be at times a very complicated affair. Blackmon writes:

> Significant attributions of program implementation, like those of motion and meaning, are comparative and contrastive acts. What we seek empirically is some default standard mapping, that mapping by which we can make objective discoveries of relevant similarities and differences holding not only among the brains of various biological organisms but also among artificial systems. [16]

According to this suggestion, there is a default interpretation by which we can make objective discoveries of linguistic behavior. But this idea is futile, as to interpret one's linguistic behavior is a one-time affair, cf. for example Davidson's essay "A Nice Derangement of Epitaphs" [30] for a detailed argument. The same indeterminacy concerns information encoding (except that Davidson argued for a semantic indeterminacy, while I am arguing for a syntactic indeterminacy). Information encoding is therefore more like linguistic interpretation rather than like a frame of reference in physics.

The Intentionality objection: The present argument rests on Searle's notion of intentionality. The advantage of my argument is, however, that it doesn't rest on Searle's conception of intentionality, esp. his distinction between the original intentionality, which is the feature of genuine mental states, and the derived intentionality, which can be applied onto the states of computers. This is, of course, due to the fact that I don't provide any positive account of the mind/brain relation.

6 WHAT ARE INFORMATIONAL EXPLANATIONS FOR: TWO POOL CUE BALLS

Before we proceed to the concluding section, allow me to illustrate my point that describing a system in terms of information processing is an epistemological or a pragmatic choice.
Let us have two balls on a pool table, and call them A and B. Let us assume ball A is moving and directly hits ball B. The law of the conservation of linear momentum says that the linear momentum (i.e. the product of mass and velocity) of A is transferred to the linear momentum of B (disregarding the influence of the environment). Now, assume an agent α stipulates that the linear momentum of an object encodes information (as a real number up to a certain accuracy, in gram meter per second). A ball that is not moving encodes 0, a ball weighing 10 g that is moving at a velocity of 1 m/s encodes number 10, the same ball when moving at a velocity of 0.5 m/s encodes 5 etc. Given this encoding, *we* can say that there is an information transfer from ball A to ball B after the collision. Or on colliding, ball A informs ball B about its momentum. However, an agent β could bring up another encoding by stipulating that the weight of an object encodes information. Given this encoding, *we* cannot then say that ball A informs ball B on collision. There is no information transfer, because the weights of the balls remain unchanged after a typical collision on a pool table.
On α's encoding, a collision between A and B counts as a transfer of information, but on β's encoding the same event doesn't count as a transfer of information. The crucial question is now: What description of the collision is preferable? We can describe the situation by Newton Laws (which include the law of conservation of linear momentum) or in terms of information transfer, either by α's encoding or by β's encoding (and there are infinitely many encodings). From an ontological point of view, a mechanical description is fundamental, because it is observer independent.

[9] This holds for classical (Newtonian) physics only.

There might be epistemological reasons, however, to prefer a description in terms of information transfer and furthermore practical reasons to prefer α's encoding or β's encoding.
It is convenient to describe computers in terms of information transfer, because they were designed for information processing. It might be useful to describe the brain (or its parts) also in terms of information processing. But to say that the mind/consciousness is not intrinsically biological, but rather that it is computational, is conceptually flawed.

7 CONCLUSION AND IMPLICATIONS

Let me now recap my argument: My starting point is the analogy between hardware/software and brain/mind. The brain is analogous to a computer. As computers have memory in which information can be stored, brains also must be capable of storing information. The analysis of every account of information that is relevant for this analogy shows, however, that storing information presupposes its encoding. Encoding means taking or recognizing certain physical properties (e.g. voltages in the brain) as symbols or messages. Information encoding is analogous to the syntax of a language (i.e. recognizing certain shapes as letters and words).
I argued that every encoding presupposes an encoding agent. The notion of storing information in a computer or a brain is relative to this agent (it's an observer-relative notion in Searle's words). The brain *itself* doesn't encode any information (where "itself" means "in an absolute / intrinsic / ontological / observer-independent sense"). Information encoding as well as syntax is not intrinsic to physics. Dennett proposed that what we need are "systems which store, transmit and process information in non-syntactic form" [15, p.87]. I hope to have shown that all relevant accounts of information cannot do without encoding, i.e. without syntax and that syntax cannot be accounted for in physicalist terms.
An immediate implication of this is: If the brain cannot store any information, it cannot compute any function or run any program, for non-trivial functions and programs presuppose some input information. Hence, the analogy between hardware/software and brain/mind is flawed (or it is "the most far-fetched and least useful" in Dennett's words). Nevertheless, the computational theory of the mind might be still true. The burden upon an advocate of CTM is however—following Dennett's suggestion—to provide an account of the information encoding/syntax that is observer-independent (or to get rid of the syntactic indeterminacy discussed in sect. 5). That would, in my view, involve accounting for the independent ontological reality of abstract entities (for syntax is an abstraction, i.e. recognizing particular features out of phenomena as symbols). It seems, thus, that CTM is committed to some kind of metaphysical realism in the end.
In the broad perspective, information is an epistemic notion. It is a syntactic notion (in Shannon's, the algorithmic or quantum sense) and it is a semantic notion (in Bar-Hillel–Carnap's and Dretske's sense). Using the concept of information in a non-epistemic, i.e. in an ontological sense is inconsistent with these accounts.

ACKNOWLEDGEMENTS

I'm thankful to Oron Shagrir and Marek Picha for insightful comments.

REFERENCES

[1] J. Z. Tsien. The Memory Code. *Scientific American*, July 1, 2007.
[2] A. M. Turing. On computable numbers, with an application to the Entscheidungsproblem. *Proceedings of the London Mathematical Society*, series 2, 42: 230–265 (1936–37).
[3] J. A. Fodor. Methodological solipsism considered as a research strategy in cognitive psychology. *The Behavioral and Brain Sciences*, 3(1):63–109 (1980).
[4] D. Chalmers. The Varieties of Computation: A Reply. *The Journal of Cognitive Science*, 13:213–248 (2012).
[5] J. Searle. Is the Brain's Mind a Computer Program? *Scientific American*, 262(1): 26–31 (1990).
[6] J. Haugeland. Syntax, Semantics, Physics. In J. Preston and M. Bishop (Eds.). *Views into the Chinese Room: New Essays on Searle and Artificial Intelligence*. Oxford University Press, New York, 379–392 (2002).
[7] J. Searle. *The Rediscovery of Mind*. MIT Press, Cambridge, MA (1992).
[8] H. Putnam. *Representation and Reality*. MIT Press, Cambridge (1988).
[9] D. Chalmers. Does a Rock Implement Every Finite-State Automaton? *Synthese*, 108: 309–333 (1996).
[10] J. Searle. Is the Brain a Digital Computer? *Proceedings and Addresses of the American Philosophical Association*, 64(3):21–37 (1990).
[11] C. Shannon. A Mathematical Theory of Communication, *Bell System Technical Journal*, 27: 379–423, 623–656 (1948).
[12] F. Dretske. *Knowledge and the Flow of Information*. The MIT Press, Cambridge, MA (1981).
[13] L. Kostal, P. Lansky, J.P. Rospars. Neuronal coding and spiking randomness. *Eur. J. Neurosci*. 26(10): 2693–701 (2007).
[14] O. Shagrir. Can a brain possess two minds? *Journal of Cognitive Science*, 13:145–165 (2012).
[15] D. Dennett. *Content and Consciousness*. Routledge, London (1969).
[16] J. Blackmon. Searle's Wall. *Erkenntnis*, 78 (1):109–117 (2013).
[17] Y. Bar-Hillel and R. Carnap. Semantic Information. *The British Journal for the Philosophy of Science*, 4(14): 147–157 (1953).
[18] L. Floridi. *The Philosophy of Information*. Oxford University Press, Oxford (2011).
[19] R. J. Solomonoff. A preliminary report on a general theory of inductive inference. *Techical Report ZTB-138*, Zator (1960).
[20] A. N. Kolmogorov. Three Approaches to the Quantitative Definition of Information. *Problems Inform. Transmission*, 1(1): 1–7 (1965).
[21] G. J. Chaitin. On the length of programs for computing finite binary sequences: statistical considerations. *J. Assoc. Comput. Mach*., 16: 145–159 (1969).
[22] R. Penrose. *The Emperor's New Mind*. Oxford University Press, Oxford (1989).
[23] F. Beck and J. Eccles. Quantum aspects of brain activity and the role of consciousness. *Proceedings of the National Academy of Sciences of the USA*, 89:11357–11361 (1992).
[24] H. Atmanspacher. Quantum Approaches to Consciousness. *The Stanford Encyclopedia of Philosophy* https://plato.stanford.edu/archives/sum2015/entries/qt-consciousness/ (2015).
[25] D. Dennett. *Brainstorms: Philosophical Essays on Mind and Psychology*. MIT Press, Cambridge, Mass (1978).
[26] A. Pyle (ed.). *Key Philosophers in Conversation: The Cogito Interviews*. Routledge, London (1999). Dan Dennett, 85–100.
[27] R. G. Millikan. Truth Rules, Hoverflies, and the Kripke-Wittgenstein Paradox. *The Philosophical Review*, 99(3): 323–353 (1990).
[28] D. Dennett. Evolution, Error and Intentionality. In Y. Wilks and D. Partidge (eds.), *Sourcebook on the Foundations of Artificial Intelligence*. New Mexico University Press, New Mexico (1988).
[29] J. A. Fodor. Deconstructing Dennett's Darwin. *Mind and Language*, 11: 246–262 (1996).
[30] D. Davidson. A Nice Derangement of Epitaphs. In: E. Lepore, (ed.), *Truth and Interpretation: Perspectives on the Philosophy of Donald Davidson*. Basil Blackwell, Oxford (1986).

[31] J. Searle. Minds, Brains and Programs. *Behavioral and Brain Sciences*, 3: 417–57 (1980).
[32] J. Searle. *Minds*, *Brains and Science*. Harvard University Press, Cambridge, MA (1984).