Can Computers (Really) Think?

Mike Arnautov

To answer a the question posed in my title, we need to be sure we understand the meaning of its individual terms. What do we mean by 'a computer'? A glorified abacus? A present day digital computer? No, neither kind can think. But what about a computer as an information-processing artefact?

What do we mean by 'thinking'? The familiar verbal stream of consciousness? This is sometimes argued to be a very recent Western innovation! Or do we mean some wider class of mental activity, conscious or not? Is 'information processing' a sufficient qualifier for such activity? There is a conspicuous lack of consensus in this whole area.

In any case, could any artefact actually think or would it merely simulate thought? Opponents of Artificial Intelligence often ask: 'is simulated water wet?'. This way lie Philosophical Zombies, and I don't think I can do any better than point you at Dennett's "The Unimagined Preposterousness of Zombies"[1].

Perhaps a better way to ask the same question is: could an artefact exhibit sentient behaviour? Note my use of 'sentient' to avoid all the baggage carried these day by the notion of 'intelligence'. The core assumption of AI (Artificial Intelligence) is that the answer is 'yes'. But it is merely a working assumption. It must be admitted that we do not known enough to be sure of the answer. That's why most arguments on this topic rest (explicitly or implicitly) on some proposition being considered conceivable or inconceivable. They appeal to intuition rather than to rationality.

Analogies are powerful tools for shaping intuitions. For example, one does not have to think that electricity is wet, to be impressed with the degree to which water analogies can correctly guide our intuition in understanding electrical circuits. I believe computer analogies are similarly useful in guiding our intuition on the subject of minds and thoughts, without us having to buy into the computing paradigm of the mental.

With that in mind, let us turn to the topic of today's gathering. There are a few things we can say about the nature of thoughts. Firstly, thinking, as we know it, is in some way connected with brains. Secondly, we experience thoughts as mental states or, more plausibly, as a sequence of mental events – a static mind is a dead mind. Thirdly, we know that a lot of ceaseless neural and biochemical (i.e. at bottom physical) activity goes on in brains. A static brain is a dead brain.

Hence a question inevitably arises: what is the relationship, if any, between mental events of the mind and physical events of the brain? Historically, a number different answers have been offered to this question. Here is a list of the main ones, with no attempt to do justice to any of them:

1. Dualism – mental events of the mind are ontologically separate from physical events in the brain. This view has the well rehearsed problem of explaining how minds connect to bodies.
2. Epiphenominalism – mental events are an irrelevant epiphenomenon of the physical proceeding – they do no real work. That strikes me as sensible as saying that kinetic energy is an epiphenomenon of motion and does no real work. In the standard example of engine noise being an epiphenomenon of the engine's function, the noise *does* do work – of vibrating our eardrums. It is just not the work declared to be useful by the phrasing of the example.
3. Identity theory – there is a complete identity between mental events and physical events. This seems to be more on the right lines, in that it acknowledges that the mental and the

1 Dennett, Daniel *Brainchildren, Penguin 1998*

physical are simply different aspects of what is going on in the brain. But the theory is too simplistic and unsurprisingly got shot down in short order by Kripke[2].

4. Functionalism – acknowledges that mental events are multiply-realisable by physical events. I.e. that the same mental state can correspond to a number of distinct physical states. However, to my mind it still does not do justice to the complexity of the relationship between the mental and the physical aspects of brains.
5. Perhaps the most intriguing proposal is Davidson's Anomalous Monism and this is what I want to talk to you about.

As formulated by the Stanford site's entry on Davidson[3], Anomalous Monism makes the following three assertions:

1. At least some mental events interact causally with physical events -- *The Principle of Causal Interaction*
2. Events related as cause and effect fall under strict laws (that is, laws that are 'precise, explicit and as exceptionless as possible') -- *The Principle of the Nomological Character of Causality*
3. There are no strict laws (as opposed to mere generalisations) relating mental and physical events — *The Anomalism of the Mental*

The combination of the 3rd assertion with the preceding two strikes most people as incongruent, or at least surprising. Surely if there are causal interactions involved and if causality implies strict laws, the anomalism assertion simply cannot be true, can it?

And if it is true, if both causes and effects have physical descriptions, then there is no causal work left for the mental to perform. This leads critics to conclude that Anomalous Monism collapses into epiphenomenalism – the mental is just a useless side-effect of actual physical proceedings.

Furthermore, perhaps there is another problem. If the charge of epiphenomenalism is avoided and there are no strict laws relating physical and mental events, how can mental events have reliably predictable physical effects? E.g. if I wish to touch my nose, I have no problem touching my nose.

Yet Anomalous Monism has great appeal. To quote (with her kind permission) from the slides of Marianne Talbot's 2011 Romp though the Philosophy of the Mind[4]:

> "In virtue of the physical description of the states, according to AM, the state is governed by a physical law, whilst in virtue of its mental description, the state can underwrite reason explanations of behaviour."
>
> "Anomalous Monism is physicalist insofar as it insists that every causally efficacious mental state token has a physical description"
>
> "Anomalous Monism is non-reductive insofar as each token of a mental state type might have a different physical description."

Davidson is in effect proposing that while there is a token-token identity between between mind events and brain events, there is no type-type identity, which is what makes his monism an anomalous one. This fits very naturally with Quine's thesis of radical indeterminacy of translation[5].

---

2 Kripke, Saul *Naming and Necessity*, Harvard University Press 1980

3 http://plato.stanford.edu/entries/davidson/#Anomalism

4 http://media.podcasts.ox.ac.uk/conted/romp_mind/2011-11-26_mind_02.pdf

5 A connection accepted by Quine himself – see http://putnamphil.blogspot.co.uk/2014/07/a-letter-1988-from-quine-to-chris.html

While there is a token-token identity, there is a mismatch between extensions of types to which these tokens belong respectively in our first-hand experience and in the impersonal scientific view from the outside.

While we may readily grant the possibility of such a mismatch between different human languages, is it credible for it to arise without an appeal to some external intentional viewpoint?

Let's see how all of this pans out in a computer context. Following Einstein's maxim of making things as simple as possible, but no simpler, I shall very crudely translate 'physical' as 'hardware' and 'mental' as 'software'. This is not to suggest that we should accept such a crude analogy as representing anything real. I am merely considering the *structure* of Davidson's argument in a modified context. My translation gives:

1. At least some software events interact with hardware events.

I very much doubt anybody would serious argue that software events cannot cause hardware events. For instance, as is well known, a Windows operating system crash (a software event) can result in 'the blue screen of death' (a hardware event). The software error causing this phenomenon exists whether or not the software is instantiated in a computer memory.

2. Events related as cause and effect fall under strict laws (that is, laws that are ‘precise, explicit and as exceptionless as possible’) .

Computers are strictly causal machines. Again, I do not expect anybody to disagree.

3. There are no strict laws (as opposed to mere generalisations) relating software and hardware events.

I expect it will come as a surprise to most people not familiar with modern computer architectures, that this is in fact the case. To see how this works, we need the concept of a virtual pattern. Here is a simple example of such a pattern:

JANet was quite ill one day.
FEBrile trouble came her way.
MARtyr-like, she lay in bed;
APRoned nurses softly sped.
MAYbe, said the leech judicial
JUNket would be beneficial.
JULeps, too, though freely tried,
AUGured ill, for Janet died.
SEPulchre was sadly made.
OCTaves pealed and prayers were said.
NOVices with ma'y a tear
DECorated Janet's bier.

The list of month names stands out from the background of the poem's text only if we apply the simple rule of separating out the first three letters of each line. This is generally how virtual patterns are defined: a rule, arbitrarily complex or even just plain arbitrary, is provided for extracting the pattern from its background. In textual context, one could, for example simply provide a list of letter positions on a page to be considered in order. So for example the instruction 'bomb london' potentially exists as a virtual textual pattern on this page.

This is how programs are organised in computer memory. From the perspective of a particular program it has a contiguous and stable storage/memory/stack space, but in fact that space, and the program itself is scattered across physical memory in a thoroughly contingent, ever changing manner. What is known as 'copy-on-write' is a simple illustration of this. When a running process is forked (i.e. an identical copy of the original one is create -- a fairly common procedure), initially the newly created instance of the program physically shares its memory space with the original instance. But if either of them modifies memory contents in any way, the affected chunk of the memory space is relocated (for that process only) to some other physical location. Where? Wherever there currently happens to be some spare space in physical memory. This change is undetectable to either instance of the program. To both it appears that they have contiguous and stable private memory space.

Even this is an over-simplification. Things get more complicated in NUMA (Non-Uniform Memory Architecture) set-ups, where overall physical memory is apportioned between a number of CPUs. Add to that that physical memory is itself virtualised through multiple levels of caching, all the way to being paged out to disk storage, and the result is that virtual patterns of software simply cannot be mapped in a law-like way onto physical hardware.

You may reasonably object that virtual patterns only exist by virtue of some additional, external information. In the case of software that information, tying it all together, is provided by the operating system. Surely it follows that some general laws, perhaps very complex ones, can be still constructed on the more holistic level of the computer as a whole.

This is true, though one could argue in return that such a law would not satisfy Davidson's criteria, since it would effectively consist of a horrendously long ist of exceptions. There is, however a stronger response. Operating systems themselves are evolving towards being self-describing logical patterns. In the extreme case an Operating system (OS) loader can arrange the OS in a random fashion in physical memory, adding to it the necessary information to provide the appearance of a contiguous memory space. The OS entry point would be known only to that loader and having kicked things off via that entry point the loader would be deleted. This would make the OS a self-describing dynamic virtual pattern, the description of which does not exist anywhere else. The driver for this development is security. It is very much harder to subvert an operating system if it is not loaded into memory in a predictable manner.

To summarise... If we substitute 'hardware' for 'brain' and 'software' for 'mind', Davidson's Anomalous Monism appears to lose all mystery, and is in fact applicable to the hardware/software relationship. I have no intention of claiming that it therefore follows that Anomalous Monism applies equally well in the brain/mind context. I do, however, claim that the IT analogy makes Davidson's proposal intuitively much more acceptable.

The computer analogy demonstrates that despite initial impressions Davidson's definition of Anomalous Monism

1. Is not internally contradictory – if it were it could not apply in any context.
2. Avoids the charge of epiphenominalism – I see no credible way to argue that software is a mere epiphenomenon when its properties (e.g. 'bugs') can be examined without instantiating software in hardware.
3. Has no problem with software events reliably causing hardware events – e.g. displaying on the screen the result of data analysis.

There is one other point worth mentioning. In her 2011 lectures, Marianne Talbot concluded that Anomalous Monism could work, provided one accepted an unusual view of causation in which

relata of causation were token events rather than properties. As a scientist, I find nothing strange with such a view. It is the notion that properties could be causal relata that seems strange. Properties are labels we invent to account for different ways objects empirically interact. Property-based account of causation also does not seem to fit with computers – what are the causative properties involved when a Windows bug crashes a machine, resulting in 'the blue screen of death'?

Suppose we accept that Anomalous Monism makes sense in the IT context – does this have any impact on our consideration of the mind/brain relationship? I think it does.

Firstly, it shows that Anomalous Monism as such is a perfectly coherent position. To deny its applicability to mind/brain requires some additional arguments specific to that context.

Secondly, without claiming it as a fact, I feel the analogy may to some extent carry across to the mind/brain context. Hence returning to our theme of the nature of thought, I propose what seems to me a fairly natural suggestion. Perhaps one's thoughts are dynamic virtual patterns on neural activity of the brain, sustained by the self-describing grand dynamic virtual pattern of the mind. Perhaps our difficulties with the mind/body relationship lie in our looking for physical patterning of the mind, rather than allowing for self-describing virtual patterning.