

Gödel's Theorem and Mechanism (*)

David Coder

University of Alberta

In [Minds, machines, and Gödel](#), (1) J. R. Lucas claims that Goedel's incompleteness theorem constitutes a proof "that Mechanism is false, that is, that minds cannot be explained as machines". (2) He claims further that "if the proof of the falsity of mechanism is valid, it is of the greatest consequence for the whole of philosophy". (3) It seems to me that both of these claims are exaggerated. It is true that no mind can be explained as a machine. But it is not true that Goedel's theorem proves this. At most, Goedel's theorem proves that not all minds can be explained as machines. Since this is so, Goedel's theorem cannot be expected to throw much light on why minds are different from machines. Lucas overestimates the importance of Goedel's theorem for the topic of mechanism, I believe, because he presumes falsely that being unable to follow any but mechanical procedures in mathematics makes something a machine. Lucas explains Goedel's theorem in this way:

Goedel's theorem states that in any consistent system which is strong enough to produce simple arithmetic there are formulae which cannot be proved-in- the system, but which we can see to be true. (4)

I shall not base my objection to Lucas's argument upon any exception one might take to this statement of Goedel's theorem.

Lucas's argument is this:

Goedel's theorem must apply to cybernetical machines, because it is of the essence of being a machine, that it should be a concrete instantiation of a formal system. It follows that given any machine which is consistent and capable of doing simple arithmetic, there is a formula which it is incapable of producing as being true - i.e., the formula is unprovable in-the-system - but which we can see to be true. It follows that no machine can be a complete or adequate model of the mind, that minds are essentially different from machines. (5)

If minds are essentially different from machines, then no mind is machine-like. The antecedent and the consequent of this proposition are true. But Lucas's argument fails to establish either. Consider a man whose intelligence is sufficient for him to learn to make deductions from the axioms of some system of number theory, but insufficient for him to follow Goedel's incompleteness proof. Try as we may, we cannot get him to see that Goedel's formula is true. So far as Lucas has shown, there is nothing that this man can do that a machine cannot do. Hence Lucas has not shown that a machine cannot provide an adequate model for this man's mind. He has not shown that minds are essentially different from machines. ("x is essentially different from y" is essentially different from "x is not essentially the same as y".)

We may, however, make use of the above counter-example to Lucas's argument to argue in a different way for his conclusion. Let us distinguish two cases. One man is smart enough to exercise some ingenuity in the deduction of theorems in number theory. Given a simple theorem, he may see how to prove it. But he is not smart enough to follow Goedel's proof.

Another man is not capable of even this man's limited ingenuity. However, he can recognise a proof of a theorem of number theory when he sees one. Given an enumeration of the sequence of well-formed formulas of the system, he can distinguish those sequences that constitute proofs of their last lines from those that do not.

If the mind of either of these men is machine-like, it is the mind of the latter. He finds proofs for theorems in the same way a machine does. The former exercises ingenuity.

But the mind of the latter is not so machine-like, either. He has to check over many sequences several times before he is sure whether they are proofs; one day he recognises straightway that a certain sequence is a proof, but on a subsequent trial has to check it several times before he is sure; he checks the sequences in a whimsical order; and so on. A machine might simulate such behaviour. It might e.g. scan the lines of each sequence twice before printing "proof" or "non-proof". But the machine will behave this way because we built it to behave this way. If the repetitions serve any purpose at all, it will be to accomplish something that could not, given the machine's construction, be accomplished in one scanning. It will not be to "make sure" of anything the machine has already done.

Lucas's argument gives the false impression that if no one could do arithmetical operations except by picking out the proofs in an enumeration from the non-proofs, a complete mechanical model of our actual behaviour when we do arithmetical operations could be constructed. We see here the common misconception that formal systems provide something approaching blueprints for machines. As Lucas says, "... it is of the essence of being a machine, that it should be a concrete instantiation of a formal system". (6) What makes something a machine, then, is most of all that it operates an algorithm. The way in which it operates the algorithm is, then, unimportant. Or, more accurately, there is but one way to operate an algorithm: mechanically. An algorithm is, after all, a mechanical procedure.

There is a misunderstanding here of how the term "mechanical" is used in mathematics. Perhaps we can see from my discussion of Turing machines how this misunderstanding arises. Mathematical definitions of "Turing machine" (7) are usually preceded by heuristic remarks which make it look as if the definitions were intended to specify the behaviour of an actual machine. But this appearance is deceptive. A Turing machine is first of all an algorithm. We may also call "Turing machine" anything that calculates according to a Turing algorithm. Let us teach the man who separates sequences of well-formed formulas of number theory into proofs and non-proofs to calculate according to Turing algorithms. Given any Turing algorithm, he can follow its instructions through. This is a completely mechanical procedure. But "mechanical" is here a mathematical concept. It means only: the instructions given by a Turing algorithm do not at any step leave open the next step *in the calculation*. But not everything a man does when he is operating an algorithm is a step in the calculation. So we should not infer that the man whose mathematical abilities are limited to operating Turing algorithms is, regarding his "mathematical faculty", a machine. He does not follow this mechanical procedure mechanically, as if driven by clockwork. Where, according to his instructions, he is supposed to erase one symbol and print another, he may erase the former, print the same one again, notice his slip, and finally print the right symbol. Or he may begin work on the wrong square of his tape and have to back up. He may get confused and have to start again. Perhaps he deliberately writes the wrong symbol in every square, but remembers the right one, and when he is through makes necessary corrections.

There is nothing in the mathematical definition of "Turing machine" which keeps this man from being a Turing machine in the sense of what calculates according to a Turing algorithm. But if you mistake this definition for the specification of an actual machine's behaviour, you might think otherwise. You might think that what does not behave in a completely mechanical fashion does not calculate according to a completely mechanical procedure which a Turing machine does.

It seems to me that Lucas misunderstands “mechanical” in this way. He thinks of being unable to calculate except according to mechanical procedures as a sufficient condition for being a machine. But either this leaves out of account the fact that one important part of something’s being a machine is that its construction determines its operation; or the mathematical definition of “Turing machine” requires that the behaviour of whatever operates a Turing algorithm is determined by its construction. The latter alternative is false, or so I have argued. Lucas presumes its truth. This comes out in a passage in which he argues that if we are to consider what the actual behaviour of a machine can be instead of what it must be, we must imagine that it calculates according to a semi-mechanical instead of a fully mechanical procedure.

Our idea of a machine is just this, that its behaviour is completely determined by the way it is made and the incoming “stimuli”: there is no possibility of its acting on its own: given a certain form of construction and a certain input of information, the machine must act in a certain specific way. We, however, shall be concerned not with what a machine *must* do, but with what it *can* do. That is, instead of considering the whole set of rules which together determine exactly what a machine will do in given circumstances, we shall consider only an outline of those rules, which will delimit the possible responses of the machine, but completely. The complete rules will determine the operations completely at every stage; at every stage there will be a definite instruction, e.g. “if the number is prime and greater than two add one and divide by two: if it is not prime, divide by its smallest factor”: we, however, will consider the possibility of there being alternative instructions, e.g. “In a fraction you may divide top and bottom by *any* number which is a factor of both numerator and denominator”. In relaxing the specification of our model,...it is no longer completely determinist, though still entirely mechanistic,... (8)

So the behaviour of a calculator is completely determined by its construction and “input” unless the instructions by which it calculates are not definite, i.e. unless it calculates by a non-mechanical procedure. It is not surprising, then, that Lucas thinks it important, for the refutation of mechanism, that “we” are not restricted, in doing arithmetic, to calculating according to mechanical procedures. But this fact loses all importance once we see that even if we could do arithmetic in no other way, it would not follow that our behaviour, in doing arithmetic, was determined by our “construction” and “input”.

Notes

(*) First published in *Philosophy*, 1969, 44, pp. 234-237. © Cambridge University Press.

Republished here by permission. [back](#)

(1) [Minds, Machines, and Godel](#), *Philosophy*, Vol. XXXVI, No. 137, pp. 112-27; reprinted in *Minds and machines*, A.R. Anderson, editor (Englewood Cliffs, N.J., 1964). My page references are to the original.). [back](#)

(2) P. 112. [back](#)

(3) P. 126. [back](#)

(4) P. 112. [back](#)

(5) P. 113. [back](#)

(6) P. 113. [back](#)

(7) See e.g. E. Mendelson, *Introduction to Mathematical Logic* (Princeton, N.J., 1964), pp. 229 and following. [back](#)

(8) Pp. 113-4. [back](#)