Witold Marciszewski

On going beyond the first-order logic in testing the validity of its formulas
A case study

1. Let us fancy for a moment that the first-order logic (FOL) is decidable. And let the famous Gödelian sentence (G) about its own unprovability, after its encoding in the language of arithmetic, be put as the consequent in the conditional whose antecedent is obtained as the conjunction of Peano axioms (P); thus we have: $P \Rightarrow G$ — a sentence recorded in the language of first-order logic. According to the above supposition regarding decidability, it can be decided whether the formula $P \Rightarrow G$ is valid or not. Is it valid, then G is provable in arithmetic, contrary to Gödel's (1931) result. Is it not valid, then with the same method one can check whether validity attaches to the formula $P \Rightarrow \neg G$. If so, then $\neg G$ is provable on the basis of P, again contrary to Gödel's assertion that neither G nor its negation is provable in Peano arithmetic.

Thus, when denying undecidability of first-order logic, one rejects incompletability of formalized arithmetic of natural numbers. Such a rebellion has been staged by Leon Gumański in his paper titled "The Decidability of the First-Order Functional Calculus". (in this issue). There would be no reason to bother about that claim for anyone convinced by the arguments of Gödel. Church, Turing and Post. However, there is a kinship between Gumański's claim and the current strong AI, called *computationalism*, that may be worth considering. This kinship is not to mean that computationalism challenges those limitative results like Gumański does; this only means a trend to extending the field of decidability. The computationalists do not pretend to use the first-order logic algorithms in every artificial reasoning. What they do pretend is that for any reasoning there must be possible to device a suitable algorithm, which in some cases has to be stronger than the means of FOL.

Such a project might have as its motto the following statement by Turing (1950, Section "The Mathematical Objection"); it alludes to the Gödelian sentence when used as evidence of human superiority over machines.

> "Our superiority can be only felt on such an occasion in relation to the one machine over which we scored our petty triumph. There would be no question of triumphing simultaneously over *all* machines. In short, then, there might be men cleverer than any given machine, but then again might be other machines cleverer again, and so on."

The italicized (by Turing) "*all*" refers not to all possible machines but those actually given. Turing seems to have in mind his study of 1939 where he considers an infinite series of ever stronger machines. In what follows, a case will be considered in which a problem not being solvable in FOL becomes solvable with additional resources, namely a piece of knowledge which makes it possible to construct a concrete counterexample. Should we regard such a new device, when put in the form of algorithmic procedure, as a cleverer machine in Turing's sense? The present author does not intend to answer the question but just to produce a case to make the problem more conspicuous.

In such a framework, Gumański's research may be considered as one in which a system has been devised clever enough to solve problems which in FOL itself are not solvable. Then the rejection of Gödel results as intended by the Author would fail but the unintended result might become a useful contribution to AI research.[1]

[1] A simpler explanation of the disagreement between Gumański's contention and the commonly acknowledged limitative results might consist in supposing an error in his reasoning. However, as far as I know,

Though the strong AI followers do not question the undecidability of logic, they believe that whatever occurs in human reasoning is due to an algorithm. Thus, according to that view, when one reasons in a way which he or she regards as intuitive, in fact, one is steered by a hidden algorithm without being aware of that process. This is not inconsistent with the undecidability results concerning FOL, since the algorithm in question may belong to a stronger logic. Anyway – it is argued by computationalists – no proceses in the real world, including mental processes, can be uncomputable.

2. Let us examine a certain formula of FOL for which there fails the strongest method of testing validity from among those available in FOL. However, the same formula can be rejected by constructing a counterexample that resorts to a piece of extralogical knowledge. The case to be studied is the following formula (called "CC" for "Case Considered"):

[CC] $\forall_x \exists_y Rxy \Rightarrow \forall_x Rax.$

It is no theorem of FOL, but one can neither prove it nor disprove because of an infinite sequence of loops which must appear in the process of testing. The process, using the method of analytic tableaux (Beth's semantic tableaux would also do), will be referred to as PIL (the Process of Infinite Looping). It runs as follows.

[1] $\forall_x \exists_y Ryx$

[2] $\neg\forall_x Rax$

[3] $\neg Rab$ 2

[4] $\exists_y Rya$ 1

[5] $\exists_y Ryb$ 1

[6] Rca 4

[7] Rdb 5

[8] $\exists_y Ryc$ 1

[9] $\exists_y Ryd$ 1

....

And so we go on, each elimination of the existential quantifier, as in 6 and 7, leading back to formula 1 to be satisfied with newly introduced individuals (as, say, c and d), and this produces ever new individuals due to the repeated procedure of the existential quantifier elimination.

This situation will be interpreted in different ways by a computationalist and by someone believing that in some cases human intuition alone is able to solve a problem unsolvable for algorithms. Let the latter be called *intuitionist* (in a special, ad hoc coined, meaning).

The intuitionist's comment runs as follows. The process will never stop, hence the problem is unsolvable for the algorithm, while intutively we can be certain of two things. First, that the formula is not valid, since there is a lot of counteramples supplied by our knowledge, both mathematical and empirical. Second, that the process never stops. The

many attempts were made by the Author to obtain a critical examination of his claim, but it proved difficult to find anybody who would scrutinize the course of reasoning. Some friendly critical remarks were by him obtained in private communications but these, he believes, did not destroy the argument, just helped to improve it technically.

latter judgment derives from the observation that the loops must infinitely be generated by the structure of the formulas in question. When a new individual is being introduced by the lastly occurring existential quantifier, the universal formula has to be once more tested against the existence of that individual, and according to the same formula, its existence generates the next individual, and so on in infinity. The clause ‘and so on, in infinity’ is what no algorithm can arrive at, while with humans it expresses a simple observation of regular recurrence for which there is no reason to halt.

A computationalist, on the other hand, would argue as follows. The observation concerning the infinite series of recurring loops may be wrong. We cannot be sure of it, because only an algorithm can grant us certainty. Should it be true, it would result from a hidden algorithm of which the person reasoning is not aware of. Such an algorithm is, presumably, encoded in a language of neural system.

The nature of such hidden algorithms and their relation to human consciousness may be grasped through the following parable. In the operating system Unix there are programs called *demons* which are active without being noticed by the user. They control the functionig of the system and pay useful services when not being asked for that. Thus they behave very differently from a software whose functioning in each step is controlled by a user. A computationalist would claim that when performing a reasoning like that discussed above, the reasoner consciously follows a program involved in the rules of logic up to a certain moment. It is up to that moment in which he perceives that the loops should infinitely recur. And this means that another software must have switched on, to wit a demon, which performed the hidden unconscious inference leading to the conclusion about PIL. And so, intuition is just a hidden algorithmic process performed by a logical demon.

3. However, a more cautious computationalist might prefer avoiding such a recourse to occult forces, and suggest the following reasoning instead. Let us pick a finite domain to check whether the denial of CC is satisfiable in it. If so, then CC is not universally valid, hence it is not a theorem of FOL.

Consider a finite domain with linear ordering, including at least four individuals in which the relation of *being a neighbour* holds, defined as follows: y is said to be a neighbour of x, $N(y,x)$ for short, if either x is the successor of y or y is the successor of x. In this domain the denial of CC is true since there holds both the assertion of the antecedent and the negation of the consequent of CC:

AA: $\forall x \exists y N(y,x)$ and
NC: $\neg \exists y \forall x N(y,x)$.

Let the domain in question consist of the first four natural numbers <1, 2, 3, 4>. It is easily seen that every element has a neighbour and none is a neighbour of every element.

However, it is not necessary to mention definite objects, as numbers or other ones. The counterexample can be produced in a more abstract way — as concerned with any objects linked with one another by relations whose formal properies are defined in logical and set-theoretical terms alone.

The relation to order the set in question is defined as transitive, asymmetric and connected in that set, while the neighbourhood relation as symmetric and non-transitive in the same set. The other assumption is to the effect that the domain consists of exactly four individuals. Its wording requires no more than the language of FOL with identity. When defining the formal properties of relations, one has to use the concept of set and membership, hence a set-theoretical language. Since set theory can be replaced by higher-order logics, the counterexample in question may be regarded as stated in sole logical terms (without any extralogical concepts) but going beyond the limits of first-order logic.

Once having such a counterexample, we draw the metalogical conclusion that there is a domain in which the denial of CC holds, hence CC is no universally valid formula, And so the case is solved, owing to that small step towards a stronger system.

4. There are immense possibilities of obtaining ever stronger systems so that problems insoluble so far become solved owing to new devices. The issue was dealt by Gödel 1936 and Turing 1939. With Gödel a set of more and more powerful systems is ordered according to the hierarchy of types. Here is his statement (ad hoc translation by WM).

> "Now, let S_i be the system of logic if i-th order in which the natural numbers are regarded as individuals. That is, more precisely: S_i should include variables and quantifiers for natural numbers, for classes of natural numbers, for classes of classes of natural numbers, and so on, up to classes of i-th type, together with appropriate logical axioms, but it does not include any variables of higher type. Then, certainly, there are sentences in S_i which are provable in S_{i+1} but not in S_i."

The proving means here making formalized proofs, and this is a mechanical procedure. Thus the systems of which Gödel speaks can be understood as machines. This provides a link between the above statement by Gödel and Turing's 1950 text (as quoted in Section 1). In both there occurs an infinite sequence of ever cleverer machines, with first-order logic at the very beginning of that sequence. A systematic and thought-provoking treatment of such a hierarchy is found in Turing 1939. Let me introduce this study with quoting Hodges (WWW text).

> "Turing's 1938 Princeton Ph.D. thesis, work conducted in close cooperation with Church, was entitled Systems of logic defined by ordinals, and published as (Turing 1939). Predominantly the work consisted of highly technical developments within mathematical logic. However the driving force lay in the question: what is the consequence of supplementing a formal system with uncomputable deductive steps? In pursuit of this question, Turing introduced the definition of an 'oracle' which can supply on demand the answer to the halting problem for every Turing machine. Turing gave his subject-matter an interpretation which described the mathematician's 'intuition' in theorem-proving, and Newman (1955) effectively identified the uncomputable 'oracle' with intuition. This was perhaps going too far since the 'oracle' is capable of far more than any human being; nevertheless Newman had a unique status as Turing's collaborator at this period and must reflect the tenor of Turing's discussions. In any case, Turing makes it clear that the 'intuition' being discussed is related to the human act of seeing the truth of a formally unprovable Gödel statement. To summarise, it is notable that Turing's 1938 work focussed on the same issue as Penrose now raises: the interpretation of uncomputable deductions."

There is a dramatic split between Turing 1939 and Turing 1950. The former stresses the necessity of intuition conceived as non-mechanical process to handle uncomputable deductions, while the letter thinks of reasoning as being totally reducible to mechanical procedures. Nevertheless they have something in common. This is the vision of a potentially endless sequence of reasoning systems ordered according to the increasing deductive power, like in Gödel 1936.

This is why we are not helpless in the face of more involved problems which exceed the capabilities of elementary logic. However, there is a price to be paid for that intellectual comfort, as can be best seen from Gödel's approach combined with Quine's idea of ontological commitment. According to Quine's criterion of existence "to be is be value of a bound variable", logics of higher and higher order, so abundantly offered by Gödel, imply the existence of abstract Platonian entities. This makes also evident a division of labour between humans and machines. A computing machine can solve very complex problems owing to some software and data based on strong assumptions due to the bold Platonian approach. To opt for such an approach, going very far beyond the mundane realm of first-order logic, it is a human affair and human responsibility.

References

A. Church, An unsolvable problem of elementary number theory. *Am. J. Math.* **58**, 1936, 345-363.

A. Church, A note on the Entscheidungsproblem. *J. of Symbolic Logic* **1**, 1936, 40-41, 101-102.

K. Gödel, Über formal unentscheibare Sätze der *Principia Mathematica* und verwandter Systeme – I. *Monatshefte für Mathematik und Physik*, **38**, 1931, 173-198.

K. Gödel, Über die Länge der Beweisen. *Ergebnisse eines mathematischen Kolloquiums*, Heft 7 [papers read in 1934-35], Franz Deuticke, Leipzig und Wien 1936.

A. Hodges. Uncomputability in the work of Alan Turing and Roger Penrose. http://www.turing.org.uk/philosophy/lecture1.html

A. Turing, On computable numbers, with an application to the Entscheidungsproblem. *Proc. of the London Math. Society*, Series 2, **42**, 1936, 230-265.

A. Turing, Systems of logic based on ordinals. *Proc. of the London Math. Society*, Series 2, **45**, 1939, 161-228. This was also Turing's Princeton Ph.D. thesis (1938).

A. Turing, Computing machinery and intelligence. *Mind* **59**, 1950, 433-460.