Undecidable problem

In computability theory and computational complexity theory, an **undecidable problem** is a decision problem for which it is known to be impossible to construct a single algorithm that always leads to a correct yes-or-no answer. The halting problem is an example: there is no algorithm that correctly determines whether arbitrary programs eventually halt when run.

1 Background

A decision problem is any arbitrary yes-or-no question on an infinite set of inputs. Because of this, it is traditional to define the decision problem equivalently as the set of inputs for which the problem returns *yes*. These inputs can be natural numbers, but also other values of some other kind, such as strings of a formal language. Using some encoding, such as a Gödel numbering, the strings can be encoded as natural numbers. Thus, a decision problem informally phrased in terms of a formal language is also equivalent to a set of natural numbers. To keep the formal definition simple, it is phrased in terms of subsets of the natural numbers.

Formally, a decision problem is a subset of the natural numbers. The corresponding informal problem is that of deciding whether a given number is in the set. A decision problem A is called decidable or effectively solvable if A is a recursive set. A problem is called partially decidable, **semi-decidable**, solvable, or provable if A is a recursively enumerable set. This means that there exists an algorithm that halts eventually when the answer is *yes* but may run for ever if the answer is *no*. Partially decidable problems and any other problems that are not decidable are called undecidable.

2 Example: the halting problem in computability theory

In computability theory, the halting problem is a decision problem which can be stated as follows:

> Given the description of an arbitrary program and a finite input, decide whether the program finishes running or will run forever.

Alan Turing proved in 1936 that a general algorithm running on a Turing machine that solves the halting problem for *all* possible program-input pairs necessarily cannot exist. Hence, the halting problem is *undecidable* for Turing machines.

3 Relationship with Gödel's incompleteness theorem

The concepts raised by Gödel's incompleteness theorems are very similar to those raised by the halting problem, and the proofs are quite similar. In fact, a weaker form of the First Incompleteness Theorem is an easy consequence of the undecidability of the halting problem. This weaker form differs from the standard statement of the incompleteness theorem by asserting that a complete, consistent and sound axiomatization of all statements about natural numbers is unachievable. The “sound” part is the weakening: it means that we require the axiomatic system in question to prove only *true* statements about natural numbers. It is important to observe that the statement of the standard form of Gödel's First Incompleteness Theorem is completely unconcerned with the question of truth, but only concerns the issue of whether it can be proven.

The weaker form of the theorem can be proved from the undecidability of the halting problem as follows. Assume that we have a consistent and complete axiomatization of all true first-order logic statements about natural numbers. Then we can build an algorithm that enumerates all these statements. This means that there is an algorithm $N(n)$ that, given a natural number n, computes a true first-order logic statement about natural numbers such that, for all the true statements, there is at least one n such that $N(n)$ yields that statement. Now suppose we want to decide if the algorithm with representation a halts on input i. We know that this statement can be expressed with a first-order logic statement, say $H(a, i)$. Since the axiomatization is complete it follows that either there is an n such that $N(n) = H(a, i)$ or there is an n' such that $N(n') = \neg H(a, i)$. So if we iterate over all n until we either find $H(a, i)$ or its negation, we will always halt. This means that this gives us an algorithm to decide the halting problem. Since we know that there cannot be such an algorithm, it follows that the assumption that there is a consistent and complete axiomatization of all true first-order logic statements about natural numbers must be false.

4 Examples of undecidable problems

Main article: List of undecidable problems

Undecidable problems can be related to different topics, such as logic, abstract machines or topology. Note that since there are uncountably many undecidable problems, any list, even one of infinite length, is necessarily incomplete.

5 Examples of undecidable statements

See also: List of statements independent of ZFC and Independence (mathematical logic)

There are two distinct senses of the word "undecidable" in contemporary use. The first of these is the sense used in relation to Gödel's theorems, that of a statement being neither provable nor refutable in a specified deductive system. The second sense is used in relation to computability theory and applies not to statements but to decision problems, which are countably infinite sets of questions each requiring a yes or no answer. Such a problem is said to be undecidable if there is no computable function that correctly answers every question in the problem set. The connection between these two is that if a decision problem is undecidable (in the recursion theoretical sense) then there is no consistent, effective formal system which proves for every question *A* in the problem either "the answer to *A* is yes" or "the answer to *A* is no".

Because of the two meanings of the word undecidable, the term independent is sometimes used instead of undecidable for the "neither provable nor refutable" sense. The usage of "independent" is also ambiguous, however. It can mean just "not provable", leaving open whether an independent statement might be refuted.

Undecidability of a statement in a particular deductive system does not, in and of itself, address the question of whether the truth value of the statement is well-defined, or whether it can be determined by other means. Undecidability only implies that the particular deductive system being considered does not prove the truth or falsity of the statement. Whether there exist so-called "absolutely undecidable" statements, whose truth value can never be known or is ill-specified, is a controversial point among various philosophical schools.

One of the first problems suspected to be undecidable, in the second sense of the term, was the word problem for groups, first posed by Max Dehn in 1911, which asks if there is a finitely presented group for which no algorithm exists to determine whether two words are equivalent. This was shown to be the case in 1952.

The combined work of Gödel and Paul Cohen has given two concrete examples of undecidable statements (in the first sense of the term): The continuum hypothesis can neither be proved nor refuted in ZFC (the standard axiomatization of set theory), and the axiom of choice can neither be proved nor refuted in ZF (which is all the ZFC axioms *except* the axiom of choice). These results do not require the incompleteness theorem. Gödel proved in 1940 that neither of these statements could be disproved in ZF or ZFC set theory. In the 1960s, Cohen proved that neither is provable from ZF, and the continuum hypothesis cannot be proven from ZFC.

In 1970, Russian mathematician Yuri Matiyasevich showed that Hilbert's Tenth Problem, posed in 1900 as a challenge to the next century of mathematicians, cannot be solved. Hilbert's challenge sought an algorithm which finds all solutions of a Diophantine equation. A Diophantine equation is a more general case of Fermat's Last Theorem; we seek the integer roots of a polynomial in any number of variables with integer coefficients. Since we have only one equation but *n* variables, infinitely many solutions exist (and are easy to find) in the complex plane; however, the problem becomes impossible if solutions are constrained to integer values only. Matiyasevich showed this problem to be unsolvable by mapping a Diophantine equation to a recursively enumerable set and invoking Gödel's Incompleteness Theorem.[1]

In 1936, Alan Turing proved that the halting problem—the question of whether or not a Turing machine halts on a given program—is undecidable, in the second sense of the term. This result was later generalized by Rice's theorem.

In 1973, the Whitehead problem in group theory was shown to be undecidable, in the first sense of the term, in standard set theory.

In 1977, Paris and Harrington proved that the Paris-Harrington principle, a version of the Ramsey theorem, is undecidable in the axiomatization of arithmetic given by the Peano axioms but can be proven to be true in the larger system of second-order arithmetic.

Kruskal's tree theorem, which has applications in computer science, is also undecidable from the Peano axioms but provable in set theory. In fact Kruskal's tree theorem (or its finite form) is undecidable in a much stronger system codifying the principles acceptable on basis of a philosophy of mathematics called predicativism.

Goodstein's theorem is a statement about the Ramsey theory of the natural numbers that Kirby and Paris showed is undecidable in Peano arithmetic.

Gregory Chaitin produced undecidable statements in algorithmic information theory and proved another incompleteness theorem in that setting. Chaitin's theorem states that for any theory that can represent enough arithmetic, there is an upper bound *c* such that no specific number can be proven in that theory to have Kolmogorov complexity greater than c. While Gödel's theorem is related to the liar paradox, Chaitin's result is related to Berry's paradox.

In 2007, researchers Kurtz and Simon, building on earlier work by J.H. Conway in the 1970s, proved that a natural generalization of the Collatz problem is undecidable.[2]

6 See also

- Decidability (logic)
- Entscheidungsproblem
- Proof of impossibility
- Wicked problem

7 References

[1] Matiyasevich, Yuri (1970). Диофантовость перечислимых множеств [Enumerable sets are Diophantine]. *Doklady Akademii Nauk SSSR* (in Russian). **191**: 279–282.

[2] Kurtz, Stuart A.; Simon, Janos, "The Undecidability of the. Generalized Collatz Problem", in Proceedings of the 4th International Conference on Theory and Applications of Models of Computation, TAMC 2007, held in Shanghai, China in May 2007. ISBN 3-540-72503-2. doi:10.1007/978-3-540-72504-6_49

8 Text and image sources, contributors, and licenses

8.1 Text

8.2 Images

8.3 Content license