GÖDEL'S INCOMPLETENESS AND TARSKI'S UNDEFINABILITY THEOREMS

James T. Smith
San Francisco State University

This unit is incomplete itself, still in outline form.

1. The famous theorems of Gödel on incompleteness and on unprovability of consistency, and of Tarski on undefinability of truth, are based on the same techniques of argument. They're usually proved in the context of first-order arithmetic, then extended to apply to any formalizable theory powerful enough to incorporate the appropriate techniques of formal syntax. That approach requires first realizing, then proving, that first-order arithmetic is sufficiently powerful. These tasks are especially troublesome because the familiar arithmetic theories based on the Peano postulates are not at all convenient for handling formal syntax. To avoid that problem, I'll present these theorems in the context of set theory. This approach doesn't yield Gödel's and Tarski's much stronger results, but does display their logical structure, at the expense of ignoring the techniques for analyzing syntax with weak arithmetic tools.

2. Mathematicians have found that virtually all present-day mathematics can be based solely on the principles of set theory introduced in this course. You've been exercising them here in some areas of mathematics. We usually shortcut the development, tacitly relying on your familiarity with the fundamentals of the main parts of our subject. Perhaps unrealistically, we expect you to recognize that everything done in courses in those areas can be regarded as part of set theory.

3. One of these parts of mathematics includes the specification and use of formal languages. You've seen how to handle symbols, strings of symbols, finite sequences of strings, and so on. You've seen how certain strings can be classified and related to develop and apply first-order languages. The techniques that require the most work to incorporate into set theory are recursive definition and recursive proof.

4. Formal-language techniques were originally developed during 1875–1950 by logicians for studying foundations of mathematics and linguistics. During the next decades they were used to design languages for communication with machines. The earlier formal-language studies applied recursive techniques almost exclusively in proving results about properties of natural numbers. The later researchers developed more specific recursive techniques for handling formal linguistic objects, often representing those by trees. I feel that their discussions are much easier to understand than the earlier ones. However, most logic texts have not employed the newer techniques. I am quite familiar with their use in software design, but have not yet

adapted them for use in the context of this course. Therefore, you may have to simply trust that standard computer-science techniques are sufficient for handling the formal language used here, and can be based on the set theory introduced in this course.

5. Formal-language tools for set theory, introduced next, have been designed to take advantage of techniques you've met already, both in the basic set theory part of this course, and in the part devoted to syntax.
   a. You've seen already an extremely simple first-order formal language E that is adequate for the set theory developed in the first part of this course: its only nonlogical symbol $\in$ stands for the membership relation. Strings of symbols of E are used to *represent* informal set-theoretic expressions.
   b. Using only four *auxiliary* symbols— , ϕ { } —we can construct names for objects related to E, as follows.
      i. You've seen definitions of this sort:
         (1) $<x,y> = \{\{x\},\{x,y\}\}$;
         (2) $<x,y,z> = <x,<y,z>>$, etc.;
         (3) $0 = \phi$, $1 = \{0\}$, $2 = \{0,1\}$, etc.;
         (4) n is a *natural number* $\Leftrightarrow$ (a certain expression formalizable in E);
         (5) f is a *function* on a set X $\Leftrightarrow$ f is a certain type of set of ordered pairs, one for each element of X;
         (6) the *symbols* of E are certain ordered pairs of natural numbers (let the first entry indicate the symbol type, and the second enumerate the symbols of that type);
         (7) a *string* of symbols, of length n, is a function from n to the set of symbols;
         (8) a finite sequence of strings, of length n, is a function from n to the set of strings.
      ii. Since functions with finite domains are finite, we can use strings of auxiliary symbols in a familiar way as *formal names* of the symbols, strings, and sequences of strings of E. These names are not unique, since we can change the order in which the elements of a set such as $\{x,y\}$ are listed. But we need to take some care that the sets of formal names of symbols, of strings, and of finite sequences of strings are disjoint and easy to characterize.
      iii. In an earlier part of the course, on virtual classes, you saw how to eliminate the auxiliary symbols from an informal set-theoretic expression by repeated use of extensionality and writing $x \neq x$ for $x \in \phi$.

6. It's possible to find formulas of E that represent the following (informally expressed) syntax conditions. (Many others are required as intermediate steps.)
   a. S is a symbol of E;
   b. S is a string of symbols of E;

   The words "of E" are usually omitted in the ensuing discussion.
   c. V is a variable, F is a formula, and V occurs free as the nth entry of F;

d. S is a sentence;
e. A is a sentence representing an axiom of first-order logic or of set theory;
f. A,B,C are formulas and C can be inferred from A,B by *modus ponens*;
g. P is a string of formulas, each of which is an axiom or can be inferred from one or two of its predecessors by *modus ponens*;
h. P is a provable formula;
i. F is a formula, V is a variable, N is a formal name, and S is the formula that results from F by substituting N for all free occurrences of V in F, then eliminating all auxiliary symbols by the virtual-class method.

7. To avoid confusion,
   a. the syntax terminology in the previous item will *not* be used to describe informal reasoning about sets.
   b. Formulas and sentences of E represent *conditions* and *propositions* about sets.
   c. A proposition about sets is called *formally provable* if it is represented by a provable sentence.
   d. In (informal) set theory all axioms are regarded as true, and the inference rules are recognized as truth-preserving. Thus *all formally provable propositions are true.*

8. Consider the following condition $\Gamma(x,y)$ on two objects x and y:
   a. x is the formal name of a formula with exactly one free variable,
   b. y is a formal name, and
   c. when y is substituted for all free occurrences of the free variable in the formula whose formal name is x, and the auxiliary symbols are all eliminated, the resulting sentence is provable.

9. Consider the condition $\neg\,\Gamma(x,x)$.
   a. It can be represented by a formula with exactly one free variable.
   b. It has a formal name N.
   c. $\neg\,\Gamma(N,N)$ is the proposition represented by the sentence that results when N is substituted for all free occurrences of x in the formula that represents $\neg\,\Gamma(x,x)$.
   d. Suppose $\Gamma(N,N)$ —that is, substituting N for all free occurrences of x in the formula that represents $\neg\,\Gamma(x,x)$ and eliminating all auxiliary symbols would yield a provable sentence that represents $\neg\,\Gamma(N,N)$. Given this assumption, $\neg\,\Gamma(N,N)$ would be formally provable.
   e. Since $\Gamma(N,N)$ implies formal provability of its own negation, and formally provable propositions are true, $\Gamma(N,N)$ implies its own negation. By propositional logic, it must be false. Since formally provable propositions are true, $\Gamma(N,N)$ must not be formally provable.
   f. If $\neg\,\Gamma(N,N)$ were formally provable, then $\Gamma(N,N)$ would be true merely by definition of $\Gamma(x,y)$. That would contradict the previous result, that $\Gamma(N,N)$ must be false. Therefore, $\neg\,\Gamma(N,N)$ must not be formally provable.

10. In sum,
   a. $\Gamma(N,N)$ *is false,* $\neg\Gamma(N,N)$ *true, and neither is formally provable*.
   b. Thus, the consequences of the sentences representing the axioms of set theory form an incomplete theory, which I'll call Z. (A theory is called *incomplete* if there is a sentence S in its language, such that neither S nor $\neg S$ is formally provable.)
   c. A proposition is often called *formally unprovable* if it is not formally provable, and *formally refutable* if its negation is formally provable. $\Gamma(N,N)$ asserts its own formal refutability; $\neg\Gamma(N,N)$ asserts its own formal unprovability.

11. The argument in steps 8–10 was carried out in informal set theory.
   a. Twice in step 9e it used the assumption that formally provable theorems are true.
   b. That amounts to assuming that Z is a consistent theory: on that assumption, if Z contained some sentence and its negation, both would represent true propositions, which is not possible. Of course, that assumption in turn is based on our informal belief that no proposition about sets can be simultaneously true and false—that is, informal set theory must be consistent.
   c. *The results in step 10 depend on the assumption that set theory is consistent*.
   d. Step 9e uses the words *true* and *false* in a way that is possibly vague: those attributes are not listed in step 6 as representable by formulas of E. Moreover, this usage is possibly confusing because we are often tempted to regard *true* as equivalent to *(informally) provable in set theory*.

12. Assume the condition "P is a sentence representing a true proposition" were representable by a formula of E.
   a. Then you would be able to consider the following condition $\mathrm{T}(x,y)$ on two objects x and y:
      i. x is the formal name of a formula with exactly one free variable,
      ii. y is a formal name, and
      iii. when y is substituted for all free occurrences of the free variable in the formula whose formal name is x, and the auxiliary symbols are all eliminated, the resulting sentence represents a true proposition.
   b. And you could consider the condition $\neg\mathrm{T}(x,x)$.
      i. It can be represented by a formula with exactly one free variable.
      ii. It has a formal name N.
      iii. $\neg\mathrm{T}(N,N)$ would be the proposition represented by the sentence that results when N is substituted for all free occurrences of x in the formula that represents $\neg\mathrm{T}(x,x)$.
   c. Suppose $\mathrm{T}(N,N)$ —that is, substituting N for all free occurrences of x in the formula that represents $\neg\mathrm{T}(x,x)$ and eliminating all auxiliary symbols would yield a sentence that represents the true proposition $\neg\mathrm{T}(N,N)$. That conclusion contradicts the supposition.
   d. If $\neg\mathrm{T}(N,N)$, then $\mathrm{T}(N,N)$ would be true merely by definition of $\mathrm{T}(x,y)$. That is contradictory, also.

e. The previous two steps show that the assumption is untenable: *truth of propositions in set theory is not representable in the language E*.

13. Thus, the consistency assumption used in step 9e, that formally provable theorems are true, cannot be represented directly by a sentence of E. But the informal argument used in steps 8–10 can be modified to use an assumption that can be.
   a. Assume consistency in this form: *no sentence and its negation should be simultaneously provable*. This proposition, which could be called *formal consistency*, is indeed represented by some sentence of E.
   b. Recast the argument in steps 9d–9f as follows.
      i. Suppose $\Gamma(N,N)$ were formally provable: that I should have at hand a finite sequence of formulas that constitute a proof of the sentence S that represents $\Gamma(N,N)$. But $\Gamma(N,N)$ is the proposition that the sentence T that represents $\neg\Gamma(N,N)$ should be provable. In short, I would have a proof of a sentence S that represents the proposition that a sentence T should be provable.
      ii. It can be shown that in such a situation, I can construct a proof of T.
      iii. (This is a special, verifiable case of the assumption that a formally provable sentence S is true: the case where S itself asserts the provability of some sentence.)
      iv. In this case, T is the sentence $\neg S$.
   c. Since that conclusion contradicts the consistency assumption, the supposition that $\Gamma(N,N)$ should be formally provable is untenable: given that assumption, it must be formally unprovable.
   d. Suppose $\neg\Gamma(N,N)$ were formally provable. Then then I would have a proof of the sentence S in step 12d that represents $\Gamma(N,N)$, which again contradicts the consistency assumption. Given that assumption, $\neg\Gamma(N,N)$ must be formally unprovable.

14. Is the consistency assumption formally provable?
   a. Steps 13a–13c presented an informal proof that the formal consistency can be represented by a sentence C of E and does (informally) imply the formal unprovability of $\Gamma(N,N)$. That (informal) argument did not employ the notion of truth of propositions; the sequence of formulas representing the steps in that argument could be appended to a proof of C to construct a proof of the sentence that represents the proposition that $\Gamma(N,N)$ should be formally unprovable. That is a proof of the proposition $\neg\Gamma(N,N)$; however, according to step 13d, given the consistency assumption, that is impossible.
   b. Thus, *if set theory is formally consistent, formal consistency is formally unprovable*.

15. Now I need
   a. some paragraphs on the original form of these results.
   b. And to satisfy myself where and why the need for an assumption like ω-consistency disappeared.

c. I should, moreover, point out that analogous results could be obtained by showing that Z cannot prove there is a model of Z, and certain sentences can be identified as true and false and formally unprovable.
d. Question. It is important for applications to find a finitely axiomatizable theory that is incomplete. Z is not finitely axiomatizable. But is a finitely axiomatizable fragment of Z sufficient for doing these arguments?

16. Bibliography

a. Franzén 2005. This is a wonderful informal exposition that's as much devoted to what Gödel did *not* do.
b. Gödel [1931] 1967. Gödel's original paper.
c. Jech 1994. This seems comparable to the present treatment.
d. Smullyan 1961. An SFSU graduate student successfully completed an independent study recently, based on this book.
e. Smullyan 1992. I learned early from this presentation, which incorporates the syntax into arithmetic, in a nonstandard way that is rather clean.
f. Stoll [1963] 1979, section 9.10. This is helpful but not complete.
g. Tarski [1936] 1983. Tarski's original paper.
h. Tarski 1969. A masterful paper for a scientific lay audience.