Kleene's proof of Gödel's Theorem

Peter Smith

Faculty of Philosophy, University of Cambridge

There is a familiar derivation of Gödel's Theorem from the proof by diagonalization of the unsolvability of the Halting Problem. That proof, though, still involves a kind of self-referential trick, as we in effect construct a sentence that says 'the algorithm searching for a proof of me doesn't halt'. It is worth showing, then, that some core results in the theory of partial recursive functions directly entail Gödel's First Incompleteness Theorem *without* any further self-referential trick.

We start with reminders about two theorems from the theory of partial recursive functions. First, Kleene's Normal Form theorem:

Theorem 1. *There is a three-place p.r. function C and a one-place p.r. function U such that any one-place partial recursive function can be given in the standard form*

$$f_e(n) =_{\text{def}} U(\mu z[C(e, n, z) = 0])$$

for some value of e.

(Read 'μz' as *the least z such that.*) This is a standard textbook result.

We next fix some not-so-familiar terminology:

Defn. 1. *The function g is a* completion *of a partial function f if g is total and for all n where $f(n)$ is defined, $f(n) = g(n)$.*

Defn. 2. *A partial function f is* potentially recursive *if it has a completion g which is recursive.*

Then we have another textbook result:

Theorem 2. *Not every partial recursive function is potentially recursive.*

In fact, this follows immediately from Theorem 1, by a diagonalization argument:

Proof. Put $f(n) \approx U(\mu z[C(n, n, z) = 0]) + 1$. So $f(n)$ is the function which for argument n takes the value $f_n(n) + 1$ when that is defined and is undefined otherwise. $f(n)$ is by construction a partial computable function. But there is no total recursive function g which completes it.

Suppose otherwise. For some e, then, g is the function f_e – remember, the partial recursive functions include the total recursive functions, and by Theorem 1 the f_e are all the partial recursive functions!

Since g is total, $g(e)$, i.e. $f_e(e)$, is defined. So $f(e)$, i.e. $f_e(e) + 1$, is defined. But g must agree with f when f is defined. So $f_e(e) = g(e) = f(e) = f_e(e) + 1$. Contradiction! □

Two more definitions just to fix more not-entirely-standard terminology:

Defn. 3. *The theory T represents the k-place function $f(\vec{x})$ just in case there is an $k+1$-place open wff $\varphi(\vec{\mathsf{x}}, \mathsf{y})$ such that for any $k+1$ numbers $\vec{m}, n$,*

i. if $f(\vec{m}) = n$, then $T \vdash \varphi(\vec{\mathsf{m}}, \mathsf{n})$,

ii. if $f(\vec{m}) \neq n$, then $T \vdash \neg\varphi(\vec{\mathsf{m}}, \mathsf{n})$.

Defn. 4. *A theory T is p.r. adequate just in case it represents all primitive recursive functions.*

And we can now state a core version of Gödel's First Incompleteness Theorem, and show it follows from our first two Theorems:

Theorem 3. *If theory T is (i) recursively axiomatized, (ii) p.r. adequate, and (iii) ω-consistent, then T is negation incomplete.*

Proof. Suppose for reductio that T is (i) recursively axiomatized, (ii) p.r. adequate, and (iii) ω-consistent (and hence consistent), yet is negation complete. We argue to a contradiction.

Since T is p.r. adequate, there will be a four-place wff C by which it can represent the p.r. function C that appears in Kleene's Normal Form theorem.

Now consider the following definition,

$$\overline{f}_e(n) = \begin{cases} U(\mu z[C(e,n,z) = 0]) & \text{if } \exists z[C(e,n,z) = 0] \\ 0 & \text{if } T \vdash \forall \mathsf{z} \neg \mathsf{C}(\mathsf{e},\mathsf{n},\mathsf{z},\mathsf{0}) \end{cases}$$

We show that – given our assumptions about T – this must well-define an effectively computable total function for any e.

Take this claim in stages. First, we need to show that the two conditions in our definition are exclusive and exhaustive:

1. The two conditions are mutually exclusive (so the double-barrelled definition is consistent). For assume that both (a) $C(e,n,k) = 0$ for some number k, and also (b) $T \vdash \forall \mathsf{z} \neg \mathsf{C}(\mathsf{e},\mathsf{n},\mathsf{z},\mathsf{0})$. Since the formal predicate C represents C, (a) implies $T \vdash \mathsf{C}(\mathsf{e},\mathsf{n},\mathsf{k},\mathsf{0})$. Which contradicts (b), given that T is consistent.

2. The two conditions are exhaustive. Suppose the first of them doesn't hold. Then for every k, it isn't the case that $C(e,n,k) = 0$. So since T represents C, for every k, $T \vdash \neg \mathsf{C}(\mathsf{e},\mathsf{n},\mathsf{k},\mathsf{0})$. But by hypothesis T is ω-consistent, so $T \nvdash \exists \mathsf{z}\, \mathsf{C}(\mathsf{e},\mathsf{n},\mathsf{z},\mathsf{0})$. Hence since T is negation complete and it can't prove $\exists \mathsf{z}\, \mathsf{C}(\mathsf{e},\mathsf{n},\mathsf{z},\mathsf{0})$, it proves its negation, i.e. $T \vdash \forall \mathsf{z}\, \neg \mathsf{C}(\mathsf{e},\mathsf{n},\mathsf{z},\mathsf{0})$.

Which proves that, given our initial assumptions, our conditions well-define a total function $\overline{f}_e$.

Now we check that, for given e, and with our our initial assumptions still in place, $\overline{f}_e$ is effectively computable.

3. For input n, do two searches (taking alternative steps through each). One search runs through numbers $k = 0, 1, 2, \ldots$ and looks to see if k such that $C(e,n,k) = 0$ (and if and when we first find one, then we put $\overline{f}_e(n) = U(\mu z[C(e,n,z) = 0])$, as instructed). The other search runs through the proofs of T, looking to see if $\forall \mathsf{z} \neg \mathsf{C}(\mathsf{e},\mathsf{n},\mathsf{z},\mathsf{0})$ gets proved (and then we put $\overline{f}_e(n) = 0$). Each of those searches can be effectively pursued – in the second case because T is recursively axiomatized so we can recursively enumerate proofs. And it follows from what we've just shown that eventually one of the searches must terminate, and give us a value for $\overline{f}_e(n)$.

What's the upshot? Our initial assumptions imply that for every partial recursive function f_e there is a *total* effectively computable function $\overline{f}_e$ which agrees with f_e when it is defined (i.e. when $\exists z[C(e,n,z) = 0]$) and is zero otherwise. Hence for partial recursive function f_e there is an effectively computable total function $\overline{f}_e$ which completes it. And now a labour-saving invocation of Church's Thesis allows us to replace 'effectively computable' by 'recursive'. So our initial assumptions imply that every partial recursive function f_e is potentially recursive. Which contradicts Theorem 2. $\square$

Now in 1931, Gödel proved not just negation incompleteness, but negation incompleteness for Π_1 sentences. That sharpening relies on showing that Σ_1 sentences are enough to do the work of representing primitive recursive functions. But if we help ourselves to the assumption that the wff C can be Σ_1, our argument similarly delivers Π_1 incompleteness too.

In sum: Kleene's Normal Form Theorem entails Gödel's First Theorem – a neat result first noticed by Kleene himself that ought to be better known. And it is revealing that we have to do no special construction *within* the theory T to get the incompleteness result.