24 Gödel, Tarski, Church

The predicate $H(m,n,r,s)$ is primitive recursive. It is just a primitive recursive function producing truth-values (0/1).

It has the following meaning. Let y be the reverse length-lex encoding of m, a bitstring which we assume belongs to TM (the other case is easy). This means that as a bitstring, y encodes a Turing machine T_y as discussed early this term.

Let z be the reverse length-lex encoding of n.

Let w be the reverse length-lex encoding of r.

Let v be the reverse length-lex encoding of s.

The meaning of $H(m,n,r,s)$ is that v is a halting computation of T_y, its initial configuration is $q_0 z$, and its final configuration has the string w on the tape, surrounded by blanks with the read/write head positioned at its left end. This is a witness to

$$\phi_m(n) \downarrow r.$$

There is a formula $A(x_1,x_2,x_3,x_4)$ of PA such that for every $m,n,r,s \in \mathbb{N}$,

$$H(m,n,r,s) \iff \vdash_{\mathrm{PA}} A(\overline{m},\overline{n},\overline{r},\overline{s}).$$

Now define

$$S(x_1,x_2,x_3) \equiv \exists x_4 A(x_1,x_2,x_3,x_4)$$

'Obviously' this is equivalent to $\phi_m(n) \downarrow r$ but there are pitfalls, namely, that if $S(\overline{m},\overline{n},\overline{r})$ is provable, so there exists an x_4 such that, etcetera, we are cannot assume x_4 is a numeral $\overline{s}$.

(24.1) Lemma *For any m,n,r, if $\phi_m(n) \downarrow r$, then*

$$\vdash_{\mathrm{PA}} S(\overline{m},\overline{n},\overline{r}).$$

Proof. Let y be the reverse length-lex encoding of m, Since $\phi_m(n)$, $y \in$ TM and there exists an $s \in \mathbb{N}$ such that the reverse encoding u of s encodes a halting computation of T_y on input z with output w where n and r encode z and w respectively.

That is, $H(m,n,r,s)$.

Therefore $\vdash_{\mathrm{PA}} A(\overline{m},\overline{n},\overline{r},\overline{s})$.

Therefore $\vdash_{\mathrm{PA}} \exists x_4 A(\overline{m},\overline{n},\overline{r},x_4)$, i.e.,

$$\vdash_{\mathrm{PA}} S(\overline{m},\overline{n},\overline{r}). \quad \blacksquare$$

(24.2) Lemma *If $\phi_m(n) \downarrow r$ then* (i) *$S(\overline{m},\overline{n},\overline{r})$ is true in $\mathbb{N}$, and for any $r' \neq r$,* (ii) *$S(\overline{m},\overline{n},\overline{r'})$ is false in $\mathbb{N}$.*

Part (ii) *is tricky — both are omitted.* $\blacksquare$

Look very carefully at (ii). It is about ***truth in*** $\mathbb{N}$, ***not*** about provability in PA.

24.1 First-order formulae and Turing machines

We have seen how to encode Turing machines and Turing machine computations as bitstrings and, via length-lex, as numbers (in $\mathbb{N}$).

In this section, theorem-proving is studied as a computational problem. This requires an encoding of formulae and proofs as numbers too. To design such an encoding would be straightforward but time-consuming.

We assume that that has been done, and we can freely discuss computational problems about terms and formulae in PA, assuming they have been translated into problems about numbers.

If A is a formula, then "A" is the encoding of A as a natural number.

(24.3) Proposition *Assuming a reasonable encoding, the map*

$$j: \; m \mapsto \text{“}S(\overline{m}, \overline{0}, \overline{0})\text{”}$$

is recursive. ∎

(24.4) Theorem *Assuming suitable encodings of the formulae of PA as natural numbers, the set X' of theorems of Peano Arithmetic (encoded) and the set Y' of formulae which are* false *in $\mathbb{N}$ (encoded) are recursively inseparable.*

$$X' = \{ \text{“}A\text{”}: \; \vdash_{\text{PA}} A\}$$
$$Y' = \{ \text{“}A\text{”}: \; \textbf{not } \mathbb{N} \models A\}.$$

Proof. First claim that the sets

$$X = \{m \in \mathbb{N}: \; \phi_m(0) \downarrow 0\} \quad \text{and}$$
$$Y = \{m \in \mathbb{N}: \; \phi_m(0) \downarrow 1\}$$

are recursively inseparable: use the Fixed Point Theorem, as follows. If $X \subseteq C$ and $Y \cap C = \varnothing$, choose $a \in X$ and $b \in Y$ and let f map C to b and $\mathbb{N} \backslash C$ to a; f has no fixed point so it is not recursive and C is not recursive.

If $m \in X$, then

$$\vdash_{\text{PA}} S(\overline{m}, \overline{0}, \overline{0})$$

so "$S(\overline{m}, \overline{0}, \overline{0})$" $\in X'$.

If $m \in Y$, then

$$\phi_m(n) \downarrow 1$$

so

$$\textbf{not } \mathbb{N} \models S(\overline{m}, \overline{n}, \overline{0})$$

(Lemma 24.2): the encoding "$S(\overline{m}, \overline{n}, \overline{0})$" is in Y'.

Therefore $X \subseteq X'$ and $Y \subseteq Y'$. Since X and Y are recursively inseparable, so are X' and Y'. ∎

Remark. If X and Y are recursively inseparable sets, and they are disjoint, then neither X nor Y is recursive.

(24.5) Corollary Tarksi's Theorem. *The set of formulae true in* $\mathbb{N}$ *is not recursive.*

Proof. If the set of true formulae were recursive, then (it can be shown that) the set of true closed formulae would be recursive. But a closed formula F is true in $\mathbb{N}$ if and only if $\neg F$ is false in $\mathbb{N}$, so the set of false formulae would be recursive; and it isn't. $\blacksquare$

(24.6) Corollary *The set of theorems of PA is not recursive.* $\blacksquare$

(24.7) Proposition *The set of theorems of PA is recursively enumerable.*

(In other words there is a Turing machine which, given as input a formula A of PA, suitably encoded, will halt if A is provable in PA and loop otherwise.) $\blacksquare$

(24.8) Corollary Gödel-Rosser theorem. *PA is incomplete.*[1]

Proof. Otherwise, for every closed formula F, either F or $\neg F$ would be a theorem.

Set aside the possibility that PA is inconsistent, because then every formula is a theorem and the set of theorems is recursive.

Construct a Turing machine which, given a closed formula F, 'simultaneously' attempts to prove F and to prove $\neg F$. Given that PA is complete and consistent, exactly one of these attempts will succeed, so the Turing machine can decide whether or not F is a theorem, and halts on all inputs.

So the set of theorems would be recursive, which is false.

Therefore PA is incomplete. $\blacksquare$

(24.9) Corollary Church's Theorem. *Let P be the predicate calculus (no proper axioms) with the same language as Peano Arithmetic. Then the set of theorems of P is not recursive.*

Sketch proof. There is a formula $A(x_1, x_2, x_3, x_4)$ of PA such that for every $m, n, r, s \in \mathbb{N}$,

$$H(m, n, r, s) \iff \vdash_{\mathrm{PA}} A(\overline{m}, \overline{n}, \overline{r}, \overline{s}).$$

This development used only a finite list $Q_1, \ldots, Q_k$ of proper axioms (in closed form) of PA. Let K be the first-order theory with the language of PA, but whose only proper axioms are $Q_1, \ldots, Q_k$. It can be shown that m, n, r, s are in the relation H, i.e., $H(m, n, r, s)$, if and only if

$$\vdash_K A(\overline{m}, \overline{n}, \overline{r}, \overline{s}).$$

and if $\phi_m(n) \downarrow r$ then

$$\vdash_K S(\overline{m}, \overline{n}, \overline{r})$$

It will follow that the set X' of theorems of K and the set Y' of (closed) formulae false in $\mathbb{N}$ are recursively inseparable, and that the set X' is not recursive.

That is, the set of formulae, encoded,

$$\{\text{“}A\text{”} : \ \vdash_K A\}$$

[1] Gödel's original theorem gave a closed formula with certain properties. This is stronger, but non-constructive since it cannot say what the formula is.

is not recursive. But $\vdash_K A$ if and only if

$$Q_1, \ldots, Q_k \vdash_{\mathrm{PC}} A$$

which is equivalent to

$$\vdash_{\mathrm{PC}} (Q_1 \wedge \ldots \wedge Q_k) \rightarrow A$$

Therefore the set of encoded formulae of the restricted kind

$$\text{“}(Q_1 \wedge \ldots \wedge Q_k) \rightarrow A\text{”}$$

which are theorems of PC[2] is not recursive, and it would follow that the set of theorems of PC is not recursive. $\blacksquare$

[2] PC means predicate calculus, with no proper axioms. It depends uniquely on its language $\mathcal{L}(\mathrm{PC})$, which is $\mathcal{L}(\mathrm{PA})$.