

# TENNENBAUM'S THEOREM

PATRICK STEVENS

<https://www.patrickstevens.co.uk/misc/Tennenbaum/Tennenbaum.pdf>

## 1. INTRODUCTION

**Theorem 1.1** (Tennenbaum's Theorem). *Let  $\mathfrak{M}$  be a countable non-standard model of Peano arithmetic, whose carrier set is  $\mathbb{N}$ . Then it is not the case that  $+$  and  $\times$  have decidable graphs in the model.*

*Notation.* We will use the notation  $\{e\}$  to represent the  $e$ th Turing machine.  $e$  is considered only to be a standard integer here. For example, we might view the Gödel numbering scheme as being “convert from ASCII and then interpret as a Python program”.

*Remark.* How might our standard Turing machine refer to a nonstandard integer? The ground set of our nonstandard model is  $\mathbb{N}$ : every nonstandard integer has a standard one which represents it in  $\mathbb{N}$ . Perhaps  $4 \in \mathbb{N}$  is the object that the nonstandard model  $\mathfrak{M}$  thinks is the number 7, for instance. So the way a Turing machine would refer to the number 7-in-the-model is to use 4 in its source code.

What does it mean for  $+$  to have a decidable graph? Simply that there is some (standard) natural  $n$  such that, when we unpack  $n$  into instructions for running a Turing machine, we obtain a machine that takes three naturals (that is, standard naturals)  $a, b, c$  and outputs 1 iff, when we take the referents  $a', b', c'$  of  $a, b, c$  in the model  $\mathfrak{M}$ , it is true that  $a' +_{\mathfrak{M}} b' = c'$ .

*Example.* A strictly standard-length program may halt in nonstandard time, when interpreted in a nonstandard model. Indeed, fix some nonstandard “infinite”  $n$  (i.e.  $n$  is not a standard natural). Then the following program halts after  $n$  steps.

```
ans = 0;
for i = 1 to n:
  ans := ans + 1;
end
HALT with output ans;
```

## 2. OVERVIEW OF THE PROOF

The proof *est omnis divisa in partes tres*.

- (1) In any model, there is some pair of semidecidable but recursively inseparable sets.

---

*Date:* 27th April 2016.

- (2) We can use these to create an undecidable set of true standard naturals which can, in some sense, be coded up into a (nonstandard) natural in our model.
- (3) If  $+$  and  $\times$  were decidable, then the coding process would produce an object which would let us decide the undecidable set; contradiction.

### 3. EXISTENCE OF RECURSIVELY INSEPARABLE SETS

This is fairly easy. Take  $A = \{e : \{e\}(e) \downarrow = 0\}$  and  $B = \{e : \{e\}(e) \downarrow > 0\}$ , where  $\downarrow =$  means “halts and is equal to”, and  $\downarrow >$  means “halts and is greater than”. Recall that  $e$  must be standard.

Now, suppose there were a (standard) integer  $n$  such that  $\{n\}$  were the indicator function on set  $X$ , where  $X \cap B = \emptyset$  and  $A \subseteq X$ . Then what is  $\{n\}(n)$ ? If it were 0, then  $n$  is not in  $X$ , so  $n$  is not in  $A$  and so  $\{n\}(n)$  doesn't halt at 0. That's a contradiction. If it were 1, then  $n$  is in  $X$  and hence is not in  $B$ , so  $\{n\}(n)$  doesn't halt at something bigger than 0; again a contradiction.

So we have produced a pair of sets which are both semidecidable but are recursively inseparable, in the sense that no standard integer  $n$  has  $\{n\}$  deciding a superset  $X$  of  $A$  where  $X \cap B = \emptyset$ . (This is independent of the model of PA we were considering; it's purely happening over the ground set.)

### 4. CODING SETS OF NATURALS AS NATURALS

We can take any set of (possibly nonstandard) naturals and code it as a (possibly nonstandard) natural, as follows. Given  $\{n_i : i \in I\}$ , code it as  $\sum_{i \in I} 2^{n_i}$ . If  $+$  and  $\times$  are decidable, then this is a decidable coding scheme. (The preceding line is going to be where our contradiction arises, right at the end of the proof!)

Notice that if  $I$  is “standard-infinite” (that is, it contains nonstandardly-many elements) then the resulting code is nonstandard. Additionally if any  $n_i$  is strictly nonstandard.

### 5. UNDECIDABLE SET IN $\mathfrak{M}$

Take our pair of recursively inseparable semidecidable sets:  $\mathfrak{A}$  and  $\mathfrak{B}$ . (We constructed them explicitly earlier, but now we don't care what they are.) Recalling a theorem that being semidecidable is equivalent to being a projection of a decidable set, write  $A$  for a decidable set such that  $(\exists y)[(n, y) \in A]$  if and only if  $n \in \mathfrak{A}$ , and similarly for  $B$ . (The quantifiers range over  $\mathbb{N}$ , because  $A$  and  $B$  consist only of standard naturals, being subsets of the ground set.)

By their recursive-inseparability, they are in particular disjoint, so we have

$$(\forall n)[(\exists x)(\langle n, x \rangle \in A) \rightarrow \neg(\exists y)(\langle n, y \rangle \in B)]$$

where the quantifiers all range over  $\mathbb{N}$ . Equivalently,

$$(\forall n)(\forall x)(\forall y)(\neg\langle n, x \rangle \in A \vee \neg\langle n, y \rangle \in B)$$

If we bound the quantifiers by any standard  $m = SS \dots S(0)$  (which we explicitly write out, so it's absolute between all models of PA), we obtain an expression which our nonstandard model believes, because the expression is absolute for PA:

$$(\forall n < m)(\forall x < m)(\forall y < m)(\neg\langle n, x \rangle \in A \vee \neg\langle n, y \rangle \in B)$$

This is true for every standard  $m$ , and so it must be true for some nonstandard  $m$  by overspill, since  $\mathfrak{M}$  doesn't know how to distinguish between standard and nonstandard elements. If the property were only ever true for standard  $m$ , then  $\mathfrak{M}$  could identify nonstandard  $m$  by checking whether that property held for  $m$ .

Let  $e$  be strictly nonstandard such that

$$(1) \quad \mathfrak{M} \models (\forall n < e)(\forall x < e)(\forall y < e)(\langle n, x \rangle \notin A \vee \langle n, y \rangle \notin B)$$

where we note that this time  $e$  is not written out explicitly as  $SS\dots S(0)$  because it's too big to do that with.

Finally, we define our undecidable set  $X \subseteq \mathbb{N}$  of *standard* naturals to be those standard naturals  $x$  such that

$$\mathfrak{M} \models (\exists y < e)(\langle x, y \rangle \in A)$$

This is undecidable in the standard sense: there are no standard  $m$  such that  $\{m\}$  is the indicator function of  $X$ . Indeed, I claim that  $X$  separates  $\mathfrak{A}$  and  $\mathfrak{B}$ . (Recall that all members of  $X$ ,  $\mathfrak{A}$  and  $\mathfrak{B}$  are standard.)

- If  $a \in \mathfrak{A}$  then there is some standard natural  $n$  such that  $\langle a, n \rangle \in A$ ; and  $n$  is certainly less than the nonstandard  $e$ . Hence  $a \in X$ .
- If  $b \in \mathfrak{B}$ , then there is standard  $n$  such that  $\langle b, n \rangle \in B$ . Then  $n < e$ , so by (1) we have  $\langle b, x \rangle \notin A$  for all  $x < e$ . That is,  $b \notin X$ .

## 6. CODING UP $X$

Now if we code up  $X$ , which is undecidable, using our coding scheme

$$\{n_i : i \in I\} \mapsto \sum_{i \in I} 2^{n_i}$$

we obtain some nonstandard natural; say  $p = \sum_{x \in X} 2^x$ . Supposing the  $+$  and  $\times$  relations to be decidable, this coding is decidable. Remember that  $X$  is a set of standard naturals which is undecidable: no standard Turing machine decides  $X$ .

But here is a procedure to determine whether a standard element  $i \in \mathbb{N}$  is in  $X$  or not:

- (1) Take the  $i$ th bit of  $p$ . (This is decidable because  $+$  and  $\times$  are.)
- (2) Return “not in  $X$ ” if the  $i$ th bit is 0.
- (3) Otherwise return “is in  $X$ ”.

This contradicts the undecidability of  $X$ .

## 7. ACKNOWLEDGEMENTS

The structure of the proof is from Dr Thomas Forster's lecture notes on Computability and Logic from Part III of the Cambridge Maths Tripos, lectured in 2016.