SKETCH OF EXPLICIT PROOFS OF THE INCOMPLETENESS THEOREMS WITH CONCATENATION THEORY AND ROSSER'S TRICK

ALEXANDER R. PRUSS

Abstract. I sketch an explicit proof of the first incompleteness theorem with concatenation theory and Rosser's trick.

1. The concatenation language

We will work in a first-order quantified language L with identity and logical connectives $\forall$, $\exists$, $\vee$, $\wedge$, $\rightarrow$, $\leftrightarrow$, and $\neg$. The language is designed to talk about strings of symbols. These symbols include the logical connectives, lower and uppercase Latin letters, digits, parentheses, brackets, $=$, $+$, $*$, comma, the quotation marks ‘ and ’, plus symbols needed to write down proofs, such as a vertical line, a newline and a horizontal line for Fitch systems. Suppose there are N symbols in the alphabet. Unindexed Greek letters early in the alphabet like α and β indicate arbitrary symbols, and σ_i indicates the ith symbol in the alphabet. I will allow substitution within quotation marks, and won't bother with corner quotes.

Then, the names are:

- ‘α’: the symbol α (for any of the symbols in L's alphabet)

and the functions are:

- $x + y$: $x + y$ is the concatenation where x is followed by y
- $R(x, y_1, ..., y_N)$: this takes the string of symbols x and replaces σ_i with the string y_i.

This comes with a system T that contains a recursively enumerable collection of intuitive axioms that is rich enough for the proofs. Note that T will say that there is an empty string ($\exists x \forall y (y = y + x \wedge y = x + y)$).

2. Useful abbreviated stuff

I will use abbreviations of complex expressions in the system. Parentheses are taken to be inserted in a consistent way whenever needed to disambiguate things, and variables will be switched as needed.

- Proves(x, y): an incredibly complex statement that x is a proof of y from the axioms of T
- "$\alpha\beta\gamma$...": shorthand for ('α' + 'β' + 'γ' + ...) for any symbols $\alpha\beta\gamma$...
- Quotes(x, y): shorthand for an expression that says that x is a quotation of y, i.e., that x is an expression of the form ('α' + 'β' + 'γ' + ...) where y is $\alpha\beta\gamma$.... This can be defined using the replacement function R:

$$\exists z(\text{`('} + z + \text{`)'} = x \wedge x + \text{`+'} = R(y, \text{"`}\sigma_1\text{'+"}, \text{"`}\sigma_2\text{'+"}, ...)).$$

- Contains(x, y): x contains substring y, i.e.,

$$\exists v \exists w(x = v + y + w).$$

- FirstQuotes(x, y): x has a quotation in it, and the first quotation in it is of y, i.e.,

$$\exists v \exists w \exists z(\text{Quotes}(z, y) \wedge x = v + z + w \wedge \neg\,\text{Contains}(v, \text{``'})).$$

- FQAsterisked(x, y): y is obtained by replacing the first quotation in x by an asterisk, i.e.:

$$\exists v \exists w \exists z \exists u(\text{Quotes}(z, u) \wedge x = v + z + w \wedge \neg\,\text{Contains}(v, \text{``'}) \wedge y = v + \text{`}*\text{'} + w).$$

- $x \leq y$: y is at least as long as x, i.e.,

$$\text{Contains}(R(y, \text{`}*\text{'}, \text{`}*\text{'}, ...), R(x, \text{`}*\text{'}, \text{`}*\text{'}, ...)).$$

- $n(x)$: the negation of x, i.e., "$\neg$(" + x + ')'.
- Refutes(x, y): x proves the negation of y or a negand of y, i.e.,

$$\text{Proves}(x, n(y)) \vee \exists z(y = n(z) \wedge \text{Proves}(x, z)).$$

- P(y): there is a proof of y, i.e., $\exists x\, \text{Proves}(x, y)$
- RP(y): there is a Rosser proof of y, i.e., there is a proof of y such that any refutation of y is longer:

$$\exists x(\text{Proves}(x, y) \wedge \neg \exists z(z \leq x \wedge \text{Refutes}(z, y))).$$

3. The Gödel and Rosser sentences

3.1. **Truth is not provability.** Assume the theory T is true.

Let Almost abbreviate:

$$\forall x \forall z((z = * \rightarrow (\text{FirstQuotes}(x, z) \wedge \text{FQAsterisked}(x, z))) \rightarrow \neg\,\text{P}(x)).$$

Let g be the Gödel sentence:

$$\forall x \forall z((z = \text{"Almost"} \rightarrow (\text{FirstQuotes}(x, z) \wedge \text{FQAsterisked}(x, z))) \rightarrow \neg\,\text{P}(x)).$$

Fact 1: The one and only string z that satisfies

$$\forall z((z = \text{"Almost"} \rightarrow (\text{FirstQuotes}(x, z) \wedge \text{FQAsterisked}(x, z)))$$

is g.

It follows that g is true if and only if $\neg\,\mathrm{P}(g)$. Now, either $\mathrm{P}(g)$ or $\neg\,\mathrm{P}(g)$. If $\neg\,\mathrm{P}(g)$, then g is true and not provable. If $\mathrm{P}(g)$, then g is not true, and hence $\neg\neg P(g)$, so g is provable in T. Thus, if all statements provable in T are true, it follows that g is true, a contradiction.

Hence:

Theorem 1. *Assume all statements provable in T are true. Then g is an unprovable truth.*

This is a consequence of Tarski's Undefinability of Truth.

Question: Haven't we just proved that there is an unprovable truth, and thus contradicted ourselves?!

Answer: We proved that if all statements provable in T are true, then g is unprovable. That all statements provable in T are true will not be provable in T. (After all, "true" is not in the language L.)

3.2. **The First Incompleteness Theorem.** Let r be defined like g except with RP in place of P.

Theorem 2 (First Incompleteness)**.** *If T is consistent, then neither r nor $\neg r$ is provable in T.*

Fact 2: In T, for any p we can prove that we cannot have both $\mathrm{RP}(p)$ and $\mathrm{RP}(n(p))$.

Proof. Reason in T. To get a contradiction, suppose $\mathrm{RP}(p)$ and $\mathrm{RP}(n(p))$. By $\mathrm{RP}(p)$, let x be a proof of p such that any refutation of p is longer than x, and by $\mathrm{RP}(n(p))$ let y be a proof of $\neg p$ such that any refutation of $\neg p$ is longer than y. Then y is a refutation of p, so y is longer than x, and x is a refutation of $\neg p$, so x is longer than y, a contradiction. □

Fact 3: In T, we can prove that $r \leftrightarrow \neg\,\mathrm{RP}(r)$.

This is an analogue of Fact 1 and it's a lot of fiddling with strings. Since I'm doing a sketch, I won't bother with any proof.

Now we can prove First Incompleteness.

Suppose T proves r with proof x. Then by consistency there is no refutation y of r with $y \leq x$. We can prove in T that x is a proof of r (just some syntactic checking). We can also prove in T that there is no refutation y of r with $y \leq x$ (only need to go through a finite number of strings of length not exceeding that of y, and for each one verify that it's not a refutation of r). Thus, T proves $\mathrm{RP}(r)$.

By Fact 3, T proves $\neg r$, and so T is inconsistent, a contradiction.

Thus, T cannot prove r.

Suppose T proves $\neg r$ with proof x. Then T has no refutation y of $\neg r$ with $y \leq x$. We can prove in T that x is a proof of $\neg r$ and that there is no such refutation. Thus, T proves $\mathrm{RP}(n(r))$. By Fact 2, in T can prove $\neg\,\mathrm{RP}(p)$. By Fact 3, T proves r. So T is inconsistent, a contradiction.

Thus, T cannot prove $\neg r$.

4. The Second Incompleteness Theorem

Use $\mathrm{Con}(T)$ to abbreviate the statement $\neg(\mathrm{P}(\phi) \wedge \mathrm{P}(n(\phi)))$ for whatever sentence ϕ you wish.

Suppose T proves $\mathrm{Con}(T)$. Then formalizing the proof of First Incompleteness inside T, we can prove $\neg\,\mathrm{RP}(r) \wedge \neg\,\mathrm{RP}(n(r))$ in T. Hence we can prove $\neg\,\mathrm{RP}(r)$ in T. By Fact 3, we can prove r in T. By First Incompleteness, it follows that T is inconsistent.

5. Arithmetic

We can encode strings as arithmetic (and vice versa). For instance, if $N < 1000$, we can encode the ith symbol σ_i as a three digit number between 001 and 999, and then string them together to get a number. All the appropriate axioms of our concatenation theory will have analogues in arithmetic. Consequently, First and Second Incompleteness hold in arithmetic (with the Second Incompleteness prsupposing a specific encoding of what $\mathrm{Con}(T)$ means).

6. *Appendix: Binary replacement

The $R(x, y_1, ..., y_N)$ function has rather big arity. One may prefer to work with a smaller arity function, like $\rho(x, y, z)$ where every occurrence of y in x is replaced by z. (The precise semantics are that the replacements are done left to right, and after each replacement of a y by z, the search and replace restarts after the end of the z.) To define $\mathrm{Quotes}(x, y)$ in terms of ρ, first apply ρ repeatedly, N times, replacing each symbol α in the alphabet by a string of two copies of α. Once this has been done, Then we apply ρ again N times to replace $\alpha\alpha$ with 'α'+ for each symbol α, with ++ replaced first, then ‘‘, and then all the other paired symbols. This gives us the equivalent of $R(y$, "'σ_1'+", "'σ_2'+", ...).

Email address: alexander_pruss@baylor.edu

Baylor University