Mathematical Logic 2016

Lecture 22: Gödel's incompleteness theorem II

Instructor: Ashutosh Gupta

TIFR, India

Compile date: 2016-11-11

Where are we and where are we going?

We have seen

- Representable
- Numeralwise determined

We will see

- Gödel numbers
- encoding proofs using Gödel numbers
- recursive relations
- the incompleteness theorem

Topic 22.1

Road to Gödel numbering

Representable functions for numbering

We need to assign a unique number

to each variable, term, and formula

such that the set of proofs is representable.

Divisibility is representable

Theorem 22.1

$Div = \{(a, b) \in \mathbb{N}^2 | b \mod a = 0\}$ is representable.

Proof.

We can define *div* as follows.

$$div = \{(a, b) | \text{there is } x \text{ s.t. } a \cdot x = b\}$$

The above definition is representable. □

Exercise 22.1

Show the set of primes P is representable.

Consecutive primes

Theorem 22.2

The set of consecutive primes is representable.

Proof.

The following relation defines the set of consecutive primes.

$$Pair = \{(x, y) | x, y \in P \text{ and } x < y \text{ and for each } x < z < y \text{ s.t. } z \notin P\}$$

□

$(a+1)$th prime

Theorem 22.3

Let function $p(a)$ returns $a+1th$ prime. p is representable.

Proof.

We use the following property of natural numbers.

$p(a) = b$ iff

$b \in P$ and $\exists z < b^{a^2}$ s.t.

1. $(2, z) \notin div$
2. for each q, r, if $q < r \leq b$ then $(q, r) \in Pair$ and
   for each j, if $j < z$ then $(q^j, z) \in div$ iff $(r^{s(j)}, z) \in div$
3. $(b^a, z) \in div$ and $(b^{a+1}, z) \notin div$

We need to show that the above encoding indeed finds $a+1$th

$(a+1)$th prime

Proof.

Let b be $(a+1)th$ prime then Let $z = 2^0 \cdot 3^1 \cdot 5^2 ... \cdot b^a$.
$z < b^{a^2}$

1. 2 does not divide z
2. every next prime divides one extra times
3. b^a divides z and b^{a+1} does not divide z

Other direction:
Let $p(a) = b$.
Due to condition 1-2, $i+1$th prime will divide z upto ith power.
Therefore $z = 2^0 \cdot .. \cdot c^a \cdot .. \cdot d^n$.
Due to the 3rd condition, b^a must divide z but not b^{a+1}.
Therefore, $b = c$. Hence, b is $a+1$th prime. □

Sequence encoding

Definition 22.1

A sequence encoding $en : \mathbb{N}^* \to \mathbb{N}$ *maps strings of numbers to numbers as follows.*

$$en(a_0, .., a_n) = p(0)^{a_0+1} \cdot .. \cdot p(n)^{a_n+1}$$

Theorem 22.4

For each n, $en(a_0, .., a_n)$ *is representable*

Proof.

the previous theorem and function composition. □

Note: the theorem is parameterized by n. The whole en is not claimed to be representable. For each n, there is a representing formula.

Sequence decoder

Definition 22.2

A *sequence decoder de* $: \mathbb{N} \times \mathbb{N} \rightarrow \mathbb{N}$ *is defined as follows.*

$$de(en(a_0, .., a_n), i) = a_i$$

Theorem 22.5

de is representable.

Proof.

Let $R = \{(a, i, n) | (a \mod p(i)^{n+2}) \neq 0 \text{ or } a = 0\}$

Let $de(a, i) = \mu n(K_{\bar{R}}(a, i, n) = 0)$ □

Exercise 22.2

What is the output of $de(2^2 \cdot 5^2, 2)$*?*

Note: If the first parameter of *de* is not a sequence encoding for some sequence then it gives an arbitrary answer, which is allowed by the definition.

Sequence numbers

Definition 22.3

The sequence numbers set contains the numbers that are sequence encoding of some sequence.

Theorem 22.6

Sequence numbers set is representable.

Proof.

Let $R = \{(a,n)|(a \mod p(n)) = 0 \text{ and } a \neq 0\}$
Let $R' = \{(a,n,n')|n' \leq n \text{ or } (a,n') \notin R\}$
Let $R'' = \{(a,n)|(a,n) \in R_\forall \text{ and } (a,n,a) \in R'_\forall\}$
$\text{sq} = \{a|\text{ there is an } n < a \text{ s.t. } (a,n) \in R''\}$ □

Encoding length

Definition 22.4

Let lh be a function that takes sequence number and returns its length, i.e.,

$$lh(en(a_0, .., a_n)) = n$$

Theorem 22.7

lh is representable.

Proof.

$lh(a) = \mu n.((a \mod p(n)) \neq 0)$ □

Restriction function

Definition 22.5

Let re be a restriction function that is defined as follows.

$$re(en(a_0, .., a_n), i) = en(a_0, .., a_i)$$

Theorem 22.8

re is representable.

Proof.

Let $R = \{(a, i, n, k) |$ if $(a \mod p(i)^k = 0)$ then $(n \mod p(i)^k = 0)\}$
Let $R' = \{(a, i, n) | a = 0$ or, $n \neq 0$ and $(a, i, n, a) \in R_\forall\}$
$re(a, i) = \mu n.\ ((a, i, n) \in R')$ □

Encoded Recursion

Definition 22.6

Let $\bar{f}(a, \vec{b}) = en(f(0, \vec{b}), .., f(a-1, \vec{b}))$

Theorem 22.9

For a function $g : \mathbb{N}^{n+2} \to \mathbb{N}$, *there is a unique function* $f : \mathbb{N}^{n+1} \to \mathbb{N}$ *s.t.*

$$f(a, \vec{b}) = g(\bar{f}(a, \vec{b}), a, \vec{b})$$

f *is called encoded recursion function. If* g *is representable then so is* f.

Proof.

Since f is recursively constructed therefore unique.

Here is a definition of f in $m_{\mathbb{N}}$.

$\bar{f}(a, \vec{b}) = \mu s.(\text{for each } i < a, de(s, i) = g(re(s, i), i, \vec{b}))$

Therefore, f is representable (why?). □

Primitive recursion

Theorem 22.10

If g and h are representable then so is f that is defined as follows

$$f(0,\vec{b}) = g(\vec{b}) \qquad f(a,\vec{b}) = h(f(a-1,\vec{b}),a,\vec{b})$$

Proof.

We need to show that f is well defined, which is straightforward(why?).

Here is a definition of f in $m_{\mathbb{N}}$ with the help of g'.

$$g'(a,i,b) = \begin{cases} g(\vec{b}) & i = 0 \\ h(de(a,i-1),i,\vec{b}) & \text{otherwise} \end{cases}$$

$$f(a,\vec{b}) = g'(\bar{f}(a,\vec{b}),a,\vec{b})$$

The constructions are numerically determined, therefore f is representable. □

Exercise 22.3

Show if f is representable the so is $f'(a,\vec{b}) = \prod_{i<a} f(i,\vec{b})$

Concatenation

Definition 22.7

Let $a * b$ concatenates two sequence numbers, i.e.,

$$en(a_1, .., a_n) * en(b_1, .., b_n) = en(a_1, .., a_n b_1, .., b_n).$$

Theorem 22.11

$*$ is representable.

Proof.

Let us define

$$f(i, a, b) = p(i + lh(a))^{de(b,i)+1}.$$

Here is a definition of $*$ in $m_{\mathbb{N}}$ with the help of f.

$$a * b = a \cdot \prod_{i<lh(b)} f(i, a, b)$$

Again, due to the construction $*$ is representable. □

Exercise 22.4

Show $*_{i<a} f(i) = f(0) * .. * f(a-1)$ is representable.

Topic 22.2

Gödel number

Show A_D is powerful

Our goal is to show that A_D has enough reasoning power for making claims about FOL reasoning over natural numbers.

For that we need to represent various objects of FOL reasoning within the language of A_D.

The object of concern are

- symbols in the signature
- variables
- terms,atoms,formulas
- proof steps
- proofs

Converting the above objects into numbers is called Gödel numbering.

Naturally, we want to number them in a way such that they are representable.

Numbering Logical connectives

We will assign a number to each symbol.

h	symbol	h	symbol
0	$\neg$	9	0
1	$\wedge$	10	s
2	$\vee$	11	$<$
3	$\Rightarrow$	12	$+$
4	$\approx$	13	$\cdot$
5	$\exists$	14	e
6	$\forall$	15	x_1
7	(	17	x_2
8	)		$\vdots$

$$h(x_i) = 13 + 2i$$

repesentable symbols

The following are general definitions wrt any signature.

Definition 22.8

$funcs = \{(k, n) | h(f) = k \text{ and } f/n \in \mathbf{F}\}$

We assume funcs is representable.

Definition 22.9

$pds = \{(k, n) | h(p) = k \text{ and } p/n \in \mathbf{R}\}$

We assume pds is representable.

In our setting, *funcs* and *pds* are finite, therefore representable.

Gödel number of expressions

We will assign a Gödel number to every expression.

Definition 22.10

For an expression $e = s_1 ... s_n$, a Gödel number $\#e$ is defined a follows.

$$\#e = en(h(s_1), .., h(s_n))$$

Example 22.1

1. $\#0 = en(9) = 2^{9+1}$
2. $\#s(0) = en(10, 9) = 2^{10+1} \cdot 3^{9+1}$
3. $\# \approx (0, x_1) = en(4, 9, 15) = 2^{4+1} \cdot 3^{9+1} \cdot 5^{15+1}$

Note that we do not count parenthesis within terms.

Example 22.2

1. $\#\forall x_1. (\exists x_2. \neg \approx (s(x_1), x_2)) = en(6, 15, 7, 5, 17, 0, 4, 10, 15, 17, 8)$

Note that we count parenthesis separating parts of formula because they play a meaning full role.

Gödel numbers for set and sequence of expressions

Definition 22.11

For a set of expressions Σ, we assign as set of Gödel numbers.

$$\#\Sigma = \{\#e | e \in \Sigma\}$$

Definition 22.12

For a sequence of expressions $e_1, ., e_n$, we assign a single Gödel number.

$$\#(e_1, ., e_n) = en(\#e_1, .., \#e_n)$$

Gödel number: variables

Theorem 22.12

The set of Gödel numbers of variables are representable.

Proof.

$V = \{a | \exists b < a.\ a = en(15 + 2b)\}$ □

First time we are using ∃ symbol in a proof of a metatheorem! This ∃ is not same as the formal ∃

Theorem 22.13

Consider function $f : \mathbb{N} \to \mathbb{N}$ s.t. $f(n) = \#s^n(0)$. f is representable.

Proof.

We may define the function using primitive recursion.

$$f(0) = en(h(0))$$
$$f(n) = en(h(s)) * f(n-1)$$

Hence, f is representable. □

Gödel number: terms

Theorem 22.14

The set of Gödel numbers of terms Trs is representable.

Proof.

Let us define the characteristic function for Trs as follows.

$$K_{Trs}(a) = \begin{cases} 1 & \text{if } a \in V \\ 1 & \text{if } \exists i < a^{a \cdot lh(a)}, k < a \text{ s.t. } sq(i) \text{ and} \\ & \forall j < lh(i). K_{Trs}(de(i,j)) = 1 \text{ and} \\ & (k, lh(i)) \in funcs \text{ and } a = en(k) **_{j<lh(i)} de(i,j) \\ 0 & otherwise \end{cases}$$

claim: search for i upto $a^{a \cdot lh(a)}$ finds a satisfying i if $a \in Trs$.

Let us suppose $\#s(t_1, ..., t_n) = a$

Then $i = 2^{\#t_1} \cdot .. \cdot p(n-1)^{\#t_n}$

$$\leq 2^a \cdot .. \cdot p(n-1)^a \leq 2^a \cdot .. \cdot p(lh(a)-1)^a \leq \underbrace{a^a \cdot .. \cdot a^a}_{lh(a) \text{ times}} \leq a^{a \cdot ln(a)}$$

□

Exercise 22.5

Translate the above definition into the encoded recursion. *Hint: Find a proper g*

Gödel number: atoms

Theorem 22.15

The set of Gödel numbers of atoms Ats is representable.

Proof.

Let us define the characteristic function for *Ats* as follows.

$$K_{Ats}(a) = \begin{cases} 1 & \text{if } \exists i < a^{a \cdot lh(a)}, k < a \text{ s.t. } sq(i), \\ & \forall j < lh(i).de(i,j) \in Trs, \\ & (k, lh(i)) \in prds, \text{ and } a = en(k) **_{j<lh(i)} de(i,j) \\ 0 & otherwise \end{cases}$$

Rest is similar argument as the previous theorem. However, there is no recursion here. □

Gödel number: formulas

Theorem 22.16

The set of Gödel numbers of formulas Frms is representable.

Proof.

Let us define the characteristic function for *Frms* as follows.

$$K_{Frs}(a) = \begin{cases} 1 & \text{if } a \in Ats \\ 1 & \text{if } \exists i < a, \text{ s.t. } i \in Frs \text{ and } a = en(h(\neg)) * op * i * cl \\ 1 & \text{if } \exists i, j < a, \text{ s.t. } i, j \in Frs \text{ and } a = op * i * en(h(\circ)) * j * cl \\ 1 & \text{if } \exists i, j < a, \text{ s.t. } i \in V \text{ and } j \in Frs \text{ and } a = en(h(\forall)) * i * op * j * cl \\ 1 & \text{if } \exists i, j < a, \text{ s.t. } i \in V \text{ and } j \in Frs \text{ and } a = en(h(\exists)) * i * op * j * cl \\ 0 & \textit{otherwise} \end{cases}$$

where $\circ$ is some boolean binary operator, $op = en(h(())$ and $cl = en(h()))$ □

Topic 22.3

Encoding proofs

Substitution

Theorem 22.17

$$sub(\#F(x), \#x, \#t) = \#F(t)$$

Proof.

sub is recursively defined.

$sub(a, b, c) =$

1. c if $a \in V$ and $a = b$
2. $en(k) * \ast_{j<lh(i)} sub(de(i,j), b, c)$ if $i < a^{a \cdot lh(a)}$, $k < a$, for each $j < lh(i)$, $de(i,j) \in Trs$ and $(k, lh(i)) \in funcs \cup prds$
3. $en(h(\forall)) * i * op * sub(j, b, c) * cl$ if $i, j < a$, $i \in V$, $j \in Frms$, and $i \neq b$
4. ... similarly for boolean operators and existential quantifier...
5. a, otherwise

□

Gödel number: variable occurs

Definition 22.13

Let $Oc = \{(\#F, \#x) | x \textit{ occurs in } F\}$

Theorem 22.18

Oc is representable.

Proof.

$(a, b) \in Oc$ iff $Sb(a, b, \#0) \neq a$ □

Theorem 22.19

Let snts *is the set of Gödel numbers of sentences. snts is representable.*

Proof.

$snts = \{a | a \in frms \text{ and } \forall b < a. \text{ if } b \in V \text{ then } (a, b) \notin Oc\}$ □

Recall : Resolution proofs

Definition 22.14

*A resolution derivation R for a set of **S**-sentences Σ is a finite sequence of clauses that are generated by the following resolution expansion rules.*

$$\textsc{Intro}\frac{}{\{F\}}F \in \Sigma \quad \textsc{Db-Neg}\frac{\{\neg\neg F\} \cup C}{\{F\} \cup C} \quad \alpha\textsc{-Rule}\frac{\{\alpha\} \cup C}{\begin{array}{c}\{\alpha_1\} \cup C\\ \{\alpha_2\} \cup C\end{array}}$$

$$\beta\textsc{-Rule}\frac{\{\beta\} \cup C}{\{\beta_1, \beta_2\} \cup C} \quad \textsc{Res}\frac{\{\neg F\} \cup C \quad \{F\} \cup D}{C \cup D}$$

$$\gamma\textsc{-Rule}\frac{\{\gamma\} \cup C}{\{\gamma(t)\} \cup C} t \in \hat{T}_{\mathbf{S}^{\mathbf{par}}} \quad \delta\textsc{-Rule}\frac{\{\delta\} \cup C}{\{\delta(c)\} \cup C} \textit{fresh } c \in \mathbf{par}$$

$$\textsc{Ref}\frac{}{\{t \approx t\}} t \in \hat{T}_{\mathbf{S}^{\mathbf{par}}} \quad \textsc{Replace}\frac{\{t \approx u\} \cup C \quad \{F(t)\} \cup D}{\{F(u)\} \cup C \cup D}$$

Some changes in resolution derivation

To enable encoding of the derivation, we need to make the following changes in resolution proof system

- A clause is viewed as a sequence not a set
- Due to the above change, we need a factoring rule.

$$\textsc{Factor}\frac{C \vee F \vee D \vee F \vee E}{C \vee F \vee D \vee E}$$

- We assume each derivation is for some theorem $\Sigma \vdash_r F$ and $\neg F$ is introduced first in the derivation.

Recognizing proof steps

Definition 22.15

For each resolution proof $\textsc{Rule}$*. Let* $\#\textsc{Rule}$ *be a relation s.t.*

$$\textsc{Rule}\frac{C_1..C_k}{C} \qquad \textit{iff} \qquad (\#C_1, .., \#C_k, \#C) \in \#\textsc{Rule}.$$

Theorem 22.20

$\#\textsc{Rule}$ *is representable.*

Proof.

We show a couple of examples. Rest should follow similarly.

case $(a, b) \in \#\textsc{DB-Neg}$:

$lh(a) = lh(b)$ for some $i < lh(a)$, $de(a, i) = en(\neg) * en(\neg) * de(b, i)$ and

for each $i \neq j < lh(a)$, $de(a, j) = de(b, j)$

case $()\#\delta\text{-}\textsc{Neg}$ $lh(a) = lh(b)$ for some $i < lh(a)$, ... □

Exercise 22.6

Finish the above case

Gödel number: proofs

Theorem 22.21

For a finite set of sentences Σ the set of resolution proofs are representable

$$proofs(\Sigma) = \{\#Pr \mid \text{There is a } F \text{ s.t. } Pr \text{ is a resolution proof for } \Sigma \vdash_r F\}$$

Proof.

Our goal is to check proofs. Let $r \in proofs(\Sigma)$.
We need to show

1. $de(r, 0) \in stncs$
2. $last(r) = 1$ encoding empty clause

For each $0 < i < lh(r)$, $j = de(r, i)$, we need to show either of the following

3. $j \in \#\Sigma = \#\textsc{Intro}$
4. $(de(r, i_1), .., de(r, i_k), j) \in \#\textsc{Rule}$, for some $\textsc{Rule}$ and $i_1, .., i_k < i$

□

cc by-nc-sa

Topic 22.4

Recursive Relations

Recursive relations

Definition 22.16

A relation $R \subseteq \mathbb{N}^n$ is recursive *if it is representable in some consistent finitely axiomatizable theory.*

Theorem 22.22

Let R be a relation. If R is recursive then R is decidable.

Proof.

The members of axiomatizable theory are enumerable.(Recall)
Let $F(\vec{x})$ represents R in the theory.
Consider $\vec{a} \in \mathbb{N}^n$.
Therefore, either $F(s^{\vec{a}}(0))$ or $\neg F(s^{\vec{a}}(0))$ in the theory.
Since the theory is consistent, only one of the two can be in the theory.
Therefore, either of the two will eventually occur in the enumeration.
Hence, R is decidable. □

Recursive relatations are representable in $\mathcal{T}_D$

Theorem 22.23

A relation R is recursive iff R is representable in $\mathcal{T}_D$.

The cumbersome construction culminates here.

Proof.

forward direction:

Let R is represented by $F(\vec{x})$ in consistent finitely axiomatizable theory A.
Let
$f(\vec{a}) = min\{d | d \in proofs(A) \text{ and } de(d, 0) = \#F(s^{\vec{a}}(0)) \text{ or } \#\neg F(s^{\vec{a}}(0))\}$.

$$\vec{a} \in R \qquad \text{iff} \qquad de(f(\vec{a}), 0) = \#\neg F(s^{\vec{a}}(0))$$

Since R is decidable, RHS is representable in $\mathcal{T}_{D(\text{why?})}$.

backward direction: claim is immediate. □

Now we can use representable and recursive synonymously.

Exercise 22.7

Any recursive relation R is definable in $m_{\mathbb{N}}$.

Definable

#Cn(A) may not be recursive

Theorem 22.24

Let A be a set of sentences s.t. $\#A$ is recursive. $\#Cn(A)$ is definable.

Proof.

$a \in \#Cn(A)$ iff there is d s.t. $d \in proofs(A)$, $en(h(\neg)) * a = de(d, 0)$, and $a \in frms$. □

Since there is no upper bound on d, $\#Cn(A)$ is definable but not recursive.

Topic 22.5

Incompleteness theorem

Fixed point lemma

Theorem 22.25

For a formula $F(x)$ (single free variable), there is a sentence G s.t.

$$A_D \vdash (G \Leftrightarrow F(s^{\#G}(0)))$$

Proof.

Consider a function $f : \mathbb{N}^2 \to \mathbb{N}$ that satisfies $f(\#H(x), n) = \#H(s^n(0))$.

f is functionally representable in $A_{D\text{(why?)}}$.
Let $F'(x_1, x_2, x_3)$ functionally represents f.

Now consider

$$F''(x_1) \triangleq \forall x_3. (F'(x_1, x_1, x_3) \Rightarrow F(x_3))$$

Let $q = \#F''(x_1)$. We define

$$G \triangleq F''(q) = \forall x_3. (F'(q, q, x_3) \Rightarrow F(x_3)).$$

Fixed point lemma (contd.)

Proof(contd.)

We know

$$A_D \vdash \forall y.\ (F'(q, q, y) \Leftrightarrow y \approx s^{\#G}(0)) \qquad (*)$$

claim: $A_D \vdash G \Rightarrow F(s^{\#G}(0))$

- Using backward implication in (*), $A_D \vdash F'(q, q, s^{\#G}(0))$.
- Therefore, $A_D \cup \{G\} \vdash F(s^{\#G}(0))$.
- Therefore, $A_D \vdash G \Rightarrow F(s^{\#G}(0))$.

claim: $A_D \vdash F(s^{\#G}(0)) \Rightarrow G$

- Due to the fwd implication in (*), $A_D \cup \{F'(q, q, y)\} \vdash y \approx s^{\#G}(0)$
- Therefore, $A_D \cup \{F'(q, q, y), F(s^{\#G}(0))\} \vdash F(y)$
- Therefore, $A_D \cup \{F(s^{\#G}(0))\} \vdash \underbrace{\forall y.\ (F'(q, q, y) \Rightarrow F(y))}_{G}$ $\square$

Gödel's Incompleteness theorem

Theorem 22.26

For each recursive $A \subseteq \mathcal{T}_{\mathbb{N}}$, there is a sentence G s.t. $m_{\mathbb{N}} \models G$ and $A \not\vdash G$

Proof.

Since A is recursive, there is a formula $F(x)$ that defines $\#Cn(A)$ in $m_{\mathbb{N}}$.

Defines not represents

Due to the fixed point lemma, there is G s.t.

$$A_D \vdash (G \Leftrightarrow \neg F(s^{\#G}(0))).$$

Therefore, $m_{\mathbb{N}} \models (G \Leftrightarrow \neg F(s^{\#G}(0)))$.

two cases

$m_{\mathbb{N}} \not\models G$ and $m_{\mathbb{N}} \models F(s^{\#G}(0)))$
Therefore, $G \in Cn(A)$ (why?)
$m_{\mathbb{N}} \models G$. Contradiction.

$m_{\mathbb{N}} \models G$ and $m_{\mathbb{N}} \models \neg F(s^{\#G}(0)))$
Therefore, $G \notin Cn(A)$
$A \not\vdash G$. □

End of Lecture 22