The Representability Theorem

Professor Cian Dorr

6th December 2022

New York University

Min, the minimal theory of strings

$M1$	$\forall x \forall y(\neg(c \oplus x = \text{""}))$ (for $c \neq \text{""}$)
$M2$	$\forall x \forall y(c \oplus x = c \oplus y) \rightarrow x = y$
$M3$	$\forall x \neg(c_1 \oplus x = c_2 \oplus x)$
$M4$	$\forall x(\text{""} \oplus x = x)$
$M5$	$\forall x \forall y((c \oplus x) \oplus y = c \oplus (x \oplus y))$
$M6$	$c = c \oplus \text{""}$
$M7$	$\forall x(\text{""} \leq x)$
$M8$	$\forall x(x \leq \text{""} \leftrightarrow x = \text{""})$
$M9$	$\forall x \forall y((c_1 \oplus x \leq c_2 \oplus y) \leftrightarrow x \leq y)$
$M10$	$\forall x \forall y(x \leq y \vee y \leq x)$
$M11$	$\forall x(x = \text{""} \vee \exists y(x = c_1 \oplus y \vee \cdots \vee x = c_n \oplus y))$

Representation theorem

(i) If a set is defined in $\mathbb{S}$ by a bounded formula, it is representable in Min.

(ii) If a set is defined in $\mathbb{S}$ by a Σ_1 formula, it is semi-representable in Min.

(iii) If a partial function is defined in $\mathbb{S}$ by a bounded or Σ_1 formula, it is capturable in Min.

Σ_1 Omniscience

We derive this from:

Σ_1 Omniscience

If P is a Σ_1 sentence of the language of strings, P is true in $\mathbb{S}$ iff $\mathsf{Min} \vDash P$.

Capturing functions

We still have to show the third part of the representation theorem: that if a function f is defined in $\mathbb{S}$ by a Σ_1 formula, it is capturable in Min.

Let's just focus on 1-ary functions (the general case is similar). Suppose f is defined in $\mathbb{S}$ by the Σ_1 formula $Q(x,y) := \exists x\, P(x,y,z)$, where $P(x,y,z)$ is a bounded formula.

Then by Σ_1-omniscience, we have $\text{Min} \vDash \exists z P(\langle s\rangle, \langle fs\rangle, z)$ for all s. Unfortunately, Min may not prove the *unique existence claim*, $\forall x \exists ! y \exists x (P(x,y,z))$, and if it doesn't, $Q(x,y)$ doesn't capture f in Min. What we'll show is that there's a *different* formula, $Q'(x,y)$, that does capture f in Min.

- Confession: I am probably being dumb, but right now I don't see how to show that there's a formula $Q'(x,y)$ that captures f in Min. I do however see how to show that there's such a formula that captures f in a slightly stronger Min+ that adds the axioms $\forall x \forall y (x \leq x+y)$ and $\forall x \forall y (x \leq x+y)$ to Min. Let's just do that.

The clever move

Now we invoke the trick. Consider the (non-Σ_1) formula

$$Q'(x, y) := \exists z\, P'(x, y, z) \vee \left(\neg\exists y \exists z\, P'(x, y, z) \wedge y = \text{""}\right)$$

where

$$P'(x, y, z) := P(x, y, z) \wedge (\forall y' \leq y \oplus z)(\forall z' \leq y \oplus z)(P(x, y', z') \to y' = y)$$

- Given the assumption that $\exists x\, P(x, y, z)$ defines f in $\mathbb{S}$, $\exists x\, P'(x, y, z)$ does too. Given Σ_1-omniscience, it follows that for every string s, $\text{Min} \vdash \exists z\, P'(\langle s \rangle, \langle fs \rangle, z)$ and hence $\text{Min} \vdash Q'(\langle s \rangle, \langle fs \rangle)$
- And thanks to the little bit of universal knowledge we have built into Min, Min *does* prove the unique existence claim for $Q'(x, y)$.

Here is a sketch of a proof of that.

First, note that $\forall x \exists y\, Q'(x, y)$ is a logical truth. Now, suppose for contradiction that $Q'(x, y)$, $Q'(x, y')$, and $y \neq y'$. We can't have $\neg\exists y \exists z\, P(x, y, z)$, since then we'd have have $y = ""$ and $y' = ""$ and hence $y = y'$. So, there must exist z and z' such that both

$$P(x, y, z) \wedge (\forall y'' \leq y \oplus z)(\forall z'' \leq y \oplus z)(P(x, y'', z'') \rightarrow y'' = y))$$
$$P(x, y', z') \wedge (\forall y'' \leq y' \oplus z')(\forall z'' \leq y' \oplus z')(P(x, y'', z'') \rightarrow y'' = y'))$$

By axiom M10, we have that $y \oplus z \leq y' \oplus z'$ or $y' \oplus z' \leq y \oplus z$. Suppose it's the former. Then we also have $y \leq y' \oplus z'$ and $z \leq y' \oplus z'$ by axiom ***. So we have $P(x, y, z) \rightarrow y = y'$, which contradicts our assumptions. The other possibility leads to a contradiction in the same way.

Showing that things are Σ_1-definable

Definability and closure

The following will be a key tool in showing that things of interest are definable.

Fact

Suppose that X is a set of newline-free strings defined by a bounded formula $P_0(x)$, and $R_1, \ldots, R_n$ are relations among newline-free strings defined by bounded formulae $P_1(x, y) \ldots P_n(x, y)$. Then, the closure of X under $R_1, \ldots, R_n$ is Σ_1-definable.

Proof: just translate the following into the language of strings:

There exists a string d (for 'derivation') such that x is the last line of d, and for all e, y, f no longer than d, if y is a line of d $d = e \oplus y \oplus f$, either $P_0(y)$, or there exists a line z shorter than and part of of e such that $P_1(z, y)$, or $\ldots$, or there exists a line z shorter than and part of e such that $P_n(z, y)$.

(Of course we could use any character as our 'separator', not just the newline.)

An application

We can use this to, e.g., give a Σ_1 definition of the set of *terms* of a first order language (with finitely many function symbols/constants).

- First, find a bounded formula that defines being a variable: 'the first character of x is "x" or "y" or "z" and every character of x after the first is "0" or "1" or ....
- Disjoin this with $x = \langle c \rangle$ for every constant of the signature: call this AtomicTerm(x).
- For each n-ary function symbol f, we represent the relation $t_{n+1} = f(t_1, \ldots, t_n)$ via the quantifier-free formula $x_{n+1} = \langle f \rangle \oplus \mathrm{lpa} \oplus x_1 \oplus \mathrm{com} \oplus \cdots \oplus \mathrm{com} \oplus x_n \oplus \mathrm{rpa}$.
- So we can apply the theorem from the previous slide and we are done.

We can Σ_1 define the set of *formula* of a given first-order language (with finite signature) in a parallel way.

Extending this to relations

We've been talking here about giving Σ_1 definitions of *sets of strings* defined as the closure of some set under some given list of relations. But we can extend this idea to give definitions of *sets of pairs of strings* defined as the closure of some given set of pairs under some given list of relations among pairs; and similarly for other n-tuples.

We just need to pick some other separator character, e.g. the semicolon, that doesn't occur in the strings of interest. We first give a Σ_1 definition $P(x)$ of the set of all strings that derived from an n-tuple in the set we care about by joining its elements with semicolons. Then the formula that defines our relation can just be $P(x_1 \oplus \mathsf{sco} \oplus \cdots \oplus \mathsf{sco}\, x_n)$ (where sco is the constant that denotes `;` in the standard string structure).

An application

We can use this to define the substitution function—the set of quadruples $\langle P, v, t, P[v/t]\rangle$. It is the closure of the set AtomicSub of all such quadruples where P is atomic (which is easily seen to be bounded definable) under the following relations among quadruples, which correspond to relations among semicolon-separated strings that are obviously bounded definable:

- Being two 4-tuples $\langle s_1, s_2, s_3, s_4\rangle$, $\langle s_1', s_2', s_3', s_4'\rangle$ such that $s_1' = \neg s_1$ and $s_4' = \neg s_4$ and $s_2 = s_2'$ and $s_3' = s_3$.
- Being three 4-tuples $\langle s_1, s_2, s_3, s_4\rangle$, $\langle s_1', s_2', s_3', s_4'\rangle$, $\langle s_1'', s_2'', s_3'', s_4''\rangle$ such that $s_1'' = (s_1 \wedge s_1')$ and $s_4'' = (s_4 \wedge s_4')$ and $s_2 = s_2' = s_2''$ and $s_3 = s_3' = s_3''$. And similary for $\to$ and $\vee$.
- Being two 4-tuples $\langle s_1, s_2, s_3, s_4\rangle$, $\langle s_1', s_2', s_3', s_4'\rangle$ such that either $\langle s_1' = \forall v s_1\rangle$ for some variable $v \neq s_2$ and $\langle s_4' = \forall v s_4\rangle$, or $\langle s_1' = s_4' = \forall s_2 s_1\rangle$. And similarly for $\exists$.

Going bounded

Actually we can define terms, formulae, and substitution using *bounded* formula. That's because there is a particular n such that for any term t, any derivation of t's termhood is no more than n times longer than t itself—so we can replace 'there exists a derivation d such that…' with 'there exists a derivation d no more than n times longer than d such that…'; and similarly for the others.

Defining proof

At this point, writing down a bounded formula that defines the set of all proofs is a straightforward exercise. First we define 18 relations among semicolon-separated formula-lists (our stand-ins for sequents): *s is an instance of Assumption*; s_2 *follows by Weakening from* s_1; s_2 *follows by* $\rightarrow$*Intro from* s_1; s_3 *follows by* $\rightarrow$*Elim from* s_1 *and* s_2; . . . . This is easy once we have a definition of formula and substitution (which is needed for the quantifier and identity rules). Then, a proof is just a string in which each line is either an instance of Assumption, or follows from an earlier line by Weakening, or...

The labelling function

The labelling function raises a new issue: even strings containing newlines and semicolons have labels, so we can't use those as separator characters in the way we've been doing.

But actually it turns out that thanks to our decision to make all the constants of the language of strings other than "" have the same length, we can tackle this by brute force, by filling in the following schematic definition of $y = \text{label}(x)$:

$$\begin{aligned}
&\exists y_1 \exists y_2 (y = y_1 \oplus quo \oplus quo \oplus y_2 \\
&\qquad \wedge \text{EquallyLong}(y_2, x) \wedge \text{AllRightParens}(y_2) \\
&\qquad \wedge \forall x_1 \forall x_2 \forall x_3 (x = x_1 \oplus x_2 \oplus x_3 \wedge \text{LengthOne}(x_2) \rightarrow \\
&\qquad\quad \exists z_1 \exists z_2 \exists z_3 (y_1 = z_1 \oplus z_2 \oplus z_3 \wedge \text{6TimesAsLong}(z_1, x_1) \\
&\qquad\quad\ \wedge \text{6TimesAsLong}(z_3, x_3) \wedge z_2 = " \oplus " \oplus \text{lpa} \oplus \text{constantOf}(x_2) \oplus \text{com})))
\end{aligned}$$

Doing without separator characters

One thing to note about the labelling function is that it maps every string injectively to a newline-and-semicolon free string. Once we have a bounded definition of some function like this, we can use it to extend our previous result about how to define the closure of a set of tuples under some relations among tuples to remove the requirement that the strings not contain newlines or semicolons.

Idea: instead of defining the set we really are interested in, define the set of *labels* of elements of the set we are interested in (which is the closure of the set of all labels of elements of the starting set under appropriate relations among labels). Where $P(x)$ is the formula that defines that, $P(\text{label}(x))$ defines the set we want.

The Second Incompleteness Theorem

Review

If T is any theory extending Min, axiomatized by a decidable set Ax, there is a formula $\text{Proof}_{Ax}(x, y)$ which represents the relation *D is a proof of P from Ax* in T. In terms of this, we define two other formulas:

Review

If T is any theory extending Min, axiomatized by a decidable set Ax, there is a formula $\text{Proof}_{Ax}(x, y)$ which represents the relation *D is a proof of P from Ax* in T. In terms of this, we define two other formulas:

(i) A formula $\text{Prov}_T(x)$ of one free variable, defined as $\exists y(\text{Proof}_{Ax}(y, x))$. Given our assumption about Proof, we have for every sentence A:

(NEC) If $T \vDash A$ then $T \vDash \text{Prov}_T\langle A\rangle$

Review

If T is any theory extending Min, axiomatized by a decidable set Ax, there is a formula $\text{Proof}_{Ax}(x, y)$ which represents the relation *D is a proof of P from Ax* in T. In terms of this, we define two other formulas:

(i) A formula $\text{Prov}_T(x)$ of one free variable, defined as $\exists y(\text{Proof}_{Ax}(y, x))$. Given our assumption about Proof, we have for every sentence A:

(NEC) If $T \vDash A$ then $T \vDash \text{Prov}_T\langle A\rangle$

(ii) A sentence G_T—the Gödel sentence of T—for which we have

(D→) $T \vDash G_T \to \neg\text{Prov}_T\langle G_T\rangle$

(D←) $T \vDash \neg\text{Prov}_T\langle G_T\rangle \to G_T$

Review

Review

(NEC) and (D$\rightarrow$) already imply

(C) $$\text{If } T \vDash G_T \text{ then } T \vDash \bot$$

The key observation is just that if T is a theory it is closed under modus ponens ($\rightarrow$Elim). So by (D$\rightarrow$), if $T \vDash G_T$, then $T \vDash \neg \mathrm{Prov}\langle G_T \rangle$, while by NEC, if $T \vDash G_T$, then $T \vDash \mathrm{Prov}\langle G_T \rangle$.

Any proof can be turned into a proof that it's a proof

From NEC, we know that when there's a proof K of P from Ax, there's also a proof K^+ of $\text{Prov}_T\langle P\rangle$ from Ax.

In fact it's easy to see intuitively that there's a *computable* function f such that whenever K is a proof of P from Ax, fK is a proof of $\text{Prov}_T\langle P\rangle$ from Ax.

- For any given proof K with conclusion $\Gamma \triangleright P$, fK is a proof that just applies ∃Intro to a proof gK of $\text{Proof}_{Ax}(\langle K\rangle, \langle P\rangle)$.
- The main part of gK is a proof of $\text{Proof}(\langle K\rangle)$. This is built up one sequent at a time, proving at each step that the newly added sequent follows by a given rule from 0-3 earlier sequents. Finally we add a proof that P occurs as the conclusion of the final sequent, and for each member Q of Ax that's a premise of the final sequent, we add some canonically chosen proof of $\text{Axiom}\langle Q\rangle$ (where $\text{Axiom}(x)$ is the formula that represents Ax in Q).

So by the fact that every computable function is capturable in Min, we can

Peano string theory

We can also definitionally extend T with a 1-ary function label representing labelling and a 3-ary function subst representing substitution.

We're interested in theories that can prove that "reflection" works as we said it works:

$$(*) \qquad \forall x \forall y\, \mathrm{Proof}_{Ax}(x, y) \to \mathrm{Proof}_{Ax}(\mathrm{reflect}(x), \mathrm{subst}(\langle \mathrm{Prov}_T(z) \rangle, \langle z \rangle, \mathrm{label}(y)))$$

Min isn't strong enough: it's totally terrible at proving universal generalisations. But if we go up a step of strength (while still staying effectively axiomatisable), we'll be able to do it. It's enough (actually much more than enough) to add every instance of this **induction schema** as an axiom:

$$\text{(Induction)} \qquad (P(\text{""}) \wedge \forall x \forall y((A(x)x \wedge y \approx \text{"}a\text{"}) \to P(y \oplus x))) \to \forall x P(x)$$

Let's call theories that both extend Min and include all instances of Induction *moderately strong*. Fact: a moderately strong theory can prove (*).

Proving that whatever is provable is provable provable

Suppose T is moderately strong. Then for every formula P, we have

(Internal NEC) $$T \vDash \mathrm{Prov}_T\langle A\rangle \to \mathrm{Prov}_T\langle \mathrm{Prov}_T\langle A\rangle\rangle.$$

Compare this to:

(NEC) $$\text{If } T \vDash A \text{ then } T \vDash \mathrm{Prov}_T\langle A\rangle$$

Internal NEC says that T "knows" NEC.

(NEC-Int) follows from (*), given that by the assumptions about subst and label, we have $T \vDash \langle \mathrm{Prov}_T\langle P\rangle\rangle = \mathrm{subst}(\langle \mathrm{Prov}_T(z)\rangle, \langle z\rangle, \mathrm{label}\langle \mathrm{P}\rangle)$ for every P.

One more fact

We'll also need one other, easier assumption about our T's ability to prove things about provability:

(Internal MP) $$T \vDash \mathrm{Prov}_T\langle P \to Q\rangle \to (\mathrm{Prov}_T\langle P\rangle \to \mathrm{Prov}_T\langle Q\rangle)$$

To show this, we just need to show that we can capture in Min the function that concatenates a proof of $P \to Q$ from Ax with a proof of P from Ax, adds two sequents derived from the final sequents of those two proof by Weakening which combine the two sets of premises, and finally adds one more sequent derived from the previous to by $\to$Elim. This can actually be done in Min.

Collecting our facts

If T is moderately strong and axiomatizable, then:

(NEC)	If $T \vDash P$ then $T \vDash \mathrm{Prov}_T\langle P\rangle$
(Internal NEC)	$T \vDash \mathrm{Prov}_T\langle P\rangle \to \mathrm{Prov}_T\langle \mathrm{Prov}_T\langle P\rangle\rangle$.
(Internal MP)	$T \vDash \mathrm{Prov}_T\langle P \to Q\rangle \to (\mathrm{Prov}_T\langle P\rangle \to \mathrm{Prov}_T\langle Q\rangle)$
(D$\to$)	$T \vDash G_T \to \neg\mathrm{Prov}_T\langle G_T\rangle$
(D$\leftarrow$)	$T \vDash \neg\mathrm{Prov}_T\langle G_T\rangle \to G_T$

and as consequence of NEC and D$\to$,

(C)	If $T \vDash G_T$ then $T \vDash \bot$

The second incompleteness theorem

As we will see, these facts suffice to show:

Gödel's Second Incompleteness Theorem

If T is moderately strong and axiomatizable, then if $T \vDash \neg \mathrm{Prov}_T(\bot)$, then $T \vDash \bot$.

The second incompleteness theorem

As we will see, these facts suffice to show:

Gödel's Second Incompleteness Theorem

If T is moderately strong and axiomatizable, then if $T \vDash \neg \mathrm{Prov}_T(\bot)$, then $T \vDash \bot$.

We'll derive this by "internalizing" the proof of C from Nec and D$\rightarrow$ to get:

(Internal C) $$T \vDash \mathrm{Prov}_T\langle G_T \rangle \rightarrow \mathrm{Prov}_T\langle \bot \rangle$$

The second incompleteness theorem

As we will see, these facts suffice to show:

Gödel's Second Incompleteness Theorem

If T is moderately strong and axiomatizable, then if $T \vDash \neg \mathrm{Prov}_T(\bot)$, then $T \vDash \bot$.

We'll derive this by "internalizing" the proof of C from Nec and D$\rightarrow$ to get:

(Internal C) $$T \vDash \mathrm{Prov}_T\langle G_T \rangle \rightarrow \mathrm{Prov}_T\langle \bot \rangle$$

Once we have Internal C, we have all we need for the theorem:

- Suppose $T \vDash \neg \mathrm{Prov}_T\langle \bot \rangle$.

The second incompleteness theorem

As we will see, these facts suffice to show:

Gödel's Second Incompleteness Theorem

If T is moderately strong and axiomatizable, then if $T \vDash \neg \text{Prov}_T(\bot)$, then $T \vDash \bot$.

We'll derive this by "internalizing" the proof of C from Nec and D$\rightarrow$ to get:

(Internal C) $$T \vDash \text{Prov}_T\langle G_T\rangle \rightarrow \text{Prov}_T\langle \bot \rangle$$

Once we have Internal C, we have all we need for the theorem:

- Suppose $T \vDash \neg \text{Prov}_T\langle \bot \rangle$.
- Then $T \vDash \neg \text{Prov}_T\langle G_T\rangle$ by Internal C (since T is closed under modus tollens)

The second incompleteness theorem

As we will see, these facts suffice to show:

Gödel's Second Incompleteness Theorem

If T is moderately strong and axiomatizable, then if $T \vDash \neg \mathrm{Prov}_T(\bot)$, then $T \vDash \bot$.

We'll derive this by "internalizing" the proof of C from Nec and D$\to$ to get:

(Internal C) $$T \vDash \mathrm{Prov}_T\langle G_T\rangle \to \mathrm{Prov}_T\langle\bot\rangle$$

Once we have Internal C, we have all we need for the theorem:

- Suppose $T \vDash \neg \mathrm{Prov}_T\langle\bot\rangle$.
- Then $T \vDash \neg \mathrm{Prov}_T\langle G_T\rangle$ by Internal C (since T is closed under modus tollens)
- So $T \vDash G_T$ by D$\leftarrow$.

The second incompleteness theorem

As we will see, these facts suffice to show:

Gödel's Second Incompleteness Theorem

If T is moderately strong and axiomatizable, then if $T \vDash \neg\mathrm{Prov}_T(\bot)$, then $T \vDash \bot$.

We'll derive this by "internalizing" the proof of C from Nec and D$\to$ to get:

(Internal C) $$T \vDash \mathrm{Prov}_T\langle G_T\rangle \to \mathrm{Prov}_T\langle\bot\rangle$$

Once we have Internal C, we have all we need for the theorem:

- Suppose $T \vDash \neg\mathrm{Prov}_T\langle\bot\rangle$.
- Then $T \vDash \neg\mathrm{Prov}_T\langle G_T\rangle$ by Internal C (since T is closed under modus tollens)
- So $T \vDash G_T$ by D$\leftarrow$.
- So $T \vDash \bot$ by C.

Proving Internal C: $T \vDash \mathrm{Prov}_T\langle G_T\rangle \to \mathrm{Prov}_T\langle\bot\rangle$

Step One: applying NEC to D→, we have

$$T \vDash \mathrm{Prov}_T\langle G_T \to \neg\,\mathrm{Prov}_T\langle G_T\rangle\rangle$$

so by Internal MP,

$$T \vDash \mathrm{Prov}_T\langle G_T\rangle \to \mathrm{Prov}_T\langle\neg\,\mathrm{Prov}_T\langle G_T\rangle\rangle$$

Step Two: by Internal NEC, we have

$$T \vDash \mathrm{Prov}_T\langle G_T\rangle \to \mathrm{Prov}_T\langle\mathrm{Prov}_T\langle G_T\rangle\rangle$$

Proving Internal C: $T \vDash \mathrm{Prov}_T\langle G_T\rangle \to \mathrm{Prov}_T\langle\bot\rangle$

Step Three: Since T is a theory, for every P, we have

$$T \vDash A \to (\neg A \to \bot)$$

hence by NEC,

$$T \vDash \mathrm{Prov}_T\langle P \to (\neg P \to \bot)\rangle$$

so by two applications of Internal MP,

$$T \vDash \mathrm{Prov}_T\langle A\rangle \to (\mathrm{Prov}_T\langle\neg A\rangle \to \mathrm{Prov}_T \bot)$$

In particular,

$$T \vDash \mathrm{Prov}_T\langle G_T\rangle \to (\mathrm{Prov}_T\langle\neg G_T\rangle \to \mathrm{Prov}_T \bot)$$