Kapitola 1

Nerozhodnutelnost a neúplnost

1.1 Rozhodnutelnost

1.1.1 Rekurzivní axiomatizovatelnost

1.1.2 Rozhodnutelné teorie

1.2 Aritmetika

1.2.1 Robinsonova a Peanova aritmetika

1.2.2 Hilbertův desátý problém

1.3 Nerozhodnutelnost predikátové logiky

1.4 Gödelovy věty

1.4.1 První věta o neúplnosti

1.4.2 Důsledky první věty

1.4.3 Druhá věta o neúplnosti

1.4.4 Důsledky druhé věty

1.5 Skolemization

Introduction

Equisatisfiability

We will see that the problem of satisfiability can be reduced to open theories.

- Theories T, T' are *equisatisfiable* if T has a model $\Leftrightarrow$ T' has a model.
- A formula φ is in the *prenex (normal) form (PNF)* if it is written as

$$(Q_1x_1)\dots(Q_nx_n)\varphi',$$

where Q_i denotes $\forall$ or $\exists$, variables $x_1, \ldots, x_n$ are all distinct and φ' is an open formula, called the *matrix.* $(Q_1x_1)\ldots(Q_nx_n)$ is called the *prefix.*

- In particular, if all quantifiers are $\forall$, then φ is a *universal* formula.

To find an open theory equisatisfiable with T we proceed as follows.

(1) We replace axioms of T by equivalent formulas in the prenex form.

(2) We transform them, using new function symbols, to equisatisfiable universal formulas, so called Skolem variants.

(3) We take their matrices as axioms of a new theory.

Prenex normal form

Conversion rules for quantifiers

Let Q denote $\forall$ or $\exists$ and let $\overline{Q}$ denote the complementary quantifier.

For every formulas φ, ψ such that x is not free in the formula ψ,

$$\begin{array}{lrcl}
\models & \neg(Qx)\varphi & \leftrightarrow & (\overline{Q}x)\neg\varphi \\
\models & ((Qx)\varphi \wedge \psi) & \leftrightarrow & (Qx)(\varphi \wedge \psi) \\
\models & ((Qx)\varphi \vee \psi) & \leftrightarrow & (Qx)(\varphi \vee \psi) \\
\models & ((Qx)\varphi \rightarrow \psi) & \leftrightarrow & (\overline{Q}x)(\varphi \rightarrow \psi) \\
\models & (\psi \rightarrow (Qx)\varphi) & \leftrightarrow & (Qx)(\psi \rightarrow \varphi)
\end{array}$$

The above equivalences can be verified semantically or proved by the tableau method (*by taking the universal closure if it is not a sentence*).

Remark The assumption that x is not free in ψ is necessary in each rule above (except the first one) for some quantifier Q. For example,

$$\not\models \ ((\exists x)P(x) \wedge P(x)) \ \leftrightarrow \ (\exists x)(P(x) \wedge P(x))$$

Conversion to the prenex normal form

Proposition *Let φ' be the formula obtained from φ by replacing some occurrences of a subformula ψ with ψ'. If $T \models \psi \leftrightarrow \psi'$, then $T \models \varphi \leftrightarrow \varphi'$.*

Proof Easily by induction on the structure of the formula φ. □

Proposition *For every formula φ there is an equivalent formula φ' in the prenex normal form, i.e. $\models \varphi \leftrightarrow \varphi'$.*

Proof By induction on the structure of φ applying the conversion rules for quantifiers, replacing subformulas with their variants if needed, and applying the above proposition on equivalent transformations. □

For example,

$$\begin{aligned}
((\forall z)P(x,z) \wedge P(y,z)) &\rightarrow \neg(\exists x)P(x,y)\\
((\forall u)P(x,u) \wedge P(y,z)) &\rightarrow (\forall x)\neg P(x,y)\\
(\forall u)(P(x,u) \wedge P(y,z)) &\rightarrow (\forall v)\neg P(v,y)\\
(\exists u)((P(x,u) \wedge P(y,z)) &\rightarrow (\forall v)\neg P(v,y))\\
(\exists u)(\forall v)((P(x,u) \wedge P(y,z)) &\rightarrow \neg P(v,y))
\end{aligned}$$

Skolem variants

Skolem variants

Let φ be a sentence of a language L in the prenex normal form, let $y_1, \ldots, y_n$ be the existentially quantified variables in φ (in this order), and for every $i \leq n$ let $x_1, \ldots, x_{n_i}$ be the variables that are universally quantified in φ before y_i. Let L' be an extension of L with new n_i-ary function symbols f_i for all $i \leq n$.

Let φ_S denote the formula of L' obtained from φ by removing all $(\exists y_i)$'s from the prefix and by replacing each occurrence of y_i with the term $f_i(x_1, \ldots, x_{n_i})$. Then φ_S is called a *Skolem variant* of φ.

For example, for the sentence φ

$$(\exists y_1)(\forall x_1)(\forall x_2)(\exists y_2)(\forall x_3)R(y_1, x_1, x_2, y_2, x_3)$$

the following formula φ_S *is a Skolem variant of* φ

$$(\forall x_1)(\forall x_2)(\forall x_3)R(f_1, x_1, x_2, f_2(x_1, x_2), x_3),$$

where f_1 *is a new constant symbol and* f_2 *is a new binary function symbol.*

Properties of Skolem variants

Lemma *Let* φ *be a sentence* $(\forall x_1) \ldots (\forall x_n)(\exists y)\psi$ *of* L *and* φ' *be a sentence* $(\forall x_1) \ldots (\forall x_n)\psi(y/f(x_1, \ldots, x_n))$ *where* f *is a new function symbol. Then*

(1) *the reduct* $\mathcal{A}$ *of every model* $\mathcal{A}'$ *of* φ' *to* L *is a model of* φ,

(2) *every model* $\mathcal{A}$ *of* φ *can be expanded into a model* $\mathcal{A}'$ *of* φ'.

Remark Compared to extensions by definition of a function symbol, the expansion in (2) *does not need to be unique now.*

Proof (1) Let $\mathcal{A}' \models \varphi'$ and $\mathcal{A}$ be the reduct of $\mathcal{A}'$ to L. Since $\mathcal{A} \models \psi[e(y/a)]$ for every assignment e where $a = (f(x_1, \ldots, x_n))^{A'}[e]$, we have also $\mathcal{A} \models \varphi$.

(2) Let $\mathcal{A} \models \varphi$. There exists a function $f^A \colon A^n \rightarrow A$ such that for every

assignment e it holds $\mathcal{A} \models \psi[e(y/a)]$ where $a = f^A(e(x_1), \ldots, e(x_n))$, and thus the expansion $\mathcal{A}'$ of $\mathcal{A}$ by the function f^A is a model of φ'. □

Corollary *If* φ' *is a Skolem variant of* φ, *then both statements* (1) *and* (2) *hold for* φ, φ' *as well. Hence* φ, φ' *are equisatisfiable.*

Skolem's theorem

Skolem's theorem

Theorem *Every theory T has an open conservative extension T^*.*

Proof We may assume that T is in a closed form. Let L be its language.

- By replacing each axiom of T with an equivalent formula in the prenex normal form we obtain an equivalent theory T°.
- By replacing each axiom of T° with its Skolem variant we obtain a theory T' in an extended language $L' \supseteq L$.
- Since the reduct of every model of T' to the language L is a model of T, the theory T' is an extension of T.
- Furthermore, since every model of T can be expanded to a model of T', it is a conservative extension.
- Since every axiom of T' is a universal sentence, by replacing them with their matrices we obtain an open theory T^* equivalent to T'. □

Corollary *For every theory there is an equisatisfiable open theory.*

1.6 Herbrand's theorem

Introduction

Reduction of unsatisfiability to propositional logic

If an open theory is unsatisfiable, we can demonstrate it "via ground terms".

For example, in the language $L = \langle P, R, f, c\rangle$ the theory

$$T = \{P(x,y) \vee R(x,y),\ \neg P(c,y),\ \neg R(x, f(x))\}$$

is unsatisfiable, and this can be demonstrated by an unsatisfiable conjunction of finitely many instances of (some) axioms of T in ground terms

$$(P(c, f(c)) \vee R(c, f(c)))\ \wedge\ \neg P(c, f(c))\ \wedge\ \neg R(c, f(c)),$$

which may be seen as an unsatisfiable propositional formula

$$(p \vee r)\ \wedge\ \neg p\ \wedge\ \neg r.$$

An instance $\varphi(x_1/t_1, \ldots, x_n/t_n)$ of an open formula φ in free variables

$x_1, \ldots, x_n$ is a *ground instance* if all terms $t_1, \ldots, t_n$ are ground terms (i.e. terms without variables).

Herbrand model

Herbrand model

Let $L = \langle \mathcal{R}, \mathcal{F} \rangle$ be a language with at least one constant symbol. *(If needed, we add a new constant symbol to L.)*

- The *Herbrand universe* for L is the set of all ground terms of L.

  For example, for $L = \langle P, f, c \rangle$ with f a binary function symbol, c a constant symbol:
  $$A = \{c, f(c,c), f(f(c,c),c), f(c,f(c,c)), f(f(c,c),f(c,c)), \dots\}$$

- An L-structure $\mathcal{A}$ is a *Herbrand structure* if its domain A is the Herbrand universe for L and for each n-ary function symbol $f \in \mathcal{F}$, $t_1, \dots, t_n \in A$,
  $$f^A(t_1, \dots, t_n) = f(t_1, \dots, t_n)$$

  (including $n = 0$, i.e. $c^A = c$ for every constant symbol c).

  Remark Compared to a canonical model, the relations are not specified.

  E.g. $\mathcal{A} = \langle A, P^A, f^A, c^A \rangle$ with $P^A = \emptyset$, $c^A = c$, $f^A(c,c) = f(c,c)$, …

- A *Herbrand model* of a theory T is a Herbrand structure that models T.

Theorem and corollaries

Herbrand's theorem

Theorem *Let T be an open theory of a language L without equality and with at least one constant symbol. Then*

(*a*) *either T has a Herbrand model, or*

(*b*) *there are finitely many ground instances of axioms of T whose*

conjunction is unsatisfiable, and thus T has no model.

Proof Let T' be the set of all ground instances of axioms of T. Consider a finished (e.g. systematic) tableau τ from T' in the language L (without adding new constant symbols) with the root entry $F\bot$.

- If the tableau τ contains a noncontradictory branch B, the canonical

  model from B is a Herbrand model of T.

- Else, τ is contradictory, i.e. $T' \vdash \bot$. Moreover, τ is finite, so $\bot$ is provable from finitely many formulas of T', i.e. their conjunction is unsatisfiable. □

Remark If the language L is with equality, we extend T to T^* by axioms of equality for L and if T^* has a Herbrand model $\mathcal{A}$, we take its quotient by $=^A$.

Corollaries of Herbrand's theorem

Let L be a language containing at least one constant symbol.

Corollary *For every open $\varphi(x_1, \dots, x_n)$ of L, the formula $(\exists x_1) \dots (\exists x_n)\varphi$ is valid if and only if there exist mn ground terms t_{ij} of L for some m such that*

$$\varphi(x_1/t_{11}, \dots, x_n/t_{1n}) \vee \dots \vee \varphi(x_1/t_{m1}, \dots, x_n/t_{mn})$$

is a (propositional) tautology.

Proof $(\exists x_1) \dots (\exists x_n)\varphi$ is valid $\Leftrightarrow$ $(\forall x_1) \dots (\forall x_n)\neg\varphi$ is unsatisfiable $\Leftrightarrow$ $\neg\varphi$ is unsatisfiable. The rest follows from Herbrand's theorem for $\{\neg\varphi\}$. □

Corollary *An open theory T of L is satisfiable if and only if the theory T' of all ground instances of axioms of T is satisfiable.*

Proof If T has a model $\mathcal{A}$, every instance of each axiom of T is valid in $\mathcal{A}$, thus $\mathcal{A}$ is a model of T'. If T is unsatisfiable, by H. theorem there are (finitely many) formulas of T' whose conjunction is unsatisfiable, thus T' is unsatisfiable. □

1.7 Resolution in predicate logic

Introduction

Resolution method in predicate logic - introduction

- A refutation procedure - its aim is to show that a given formula (or theory) is unsatisfiable.

- It assumes open formulas in CNF (and in clausal form).

  A *literal* is *(now)* an atomic formula or its negation.

  A *clause* is a finite set of literals, □ denotes the empty clause.

  A *formula (in clausal form)* is a (possibly infinite) set of clauses.

  Remark Every formula (theory) can be converted to an equisatisfiable open formula (theory) in CNF, and then to a formula in clausal form.

- The resolution rule is more general - it allows to resolve through literals that are unifiable.

- Resolution in predicate logic is based on resolution in propositional logic and unification.

Local scope of variables

Variables can be renamed locally within clauses.

Let φ be an *(input)* open formula in CNF.

- φ is satisfiable if and only if its universal closure φ' is satisfiable.

- For every two formulas ψ, χ and a variable x
$$\models \quad (\forall x)(\psi \wedge \chi) \;\leftrightarrow\; (\forall x)\psi \wedge (\forall x)\chi$$
(also in the case that x is free both in ψ and χ).
- Every clause in φ can thus be replaced by its universal closure.
- We can then take any variants of clauses (to rename variables apart).

For example, by renaming variables in the second clause of (1) *we obtain an equisatisfiable formula* (2)*.*

1 $\{\{P(x), Q(x,y)\}, \{\neg P(x), \neg Q(y,x)\}\}$

2 $\{\{P(x), Q(x,y)\}, \{\neg P(v), \neg Q(u,v)\}\}$

Reduction to propositional level (grounding)

Herbrand's theorem gives us the following (inefficient) method.

- Let S be the *(input)* formula in clausal form.
- We can assume that the language contains at least one constant symbol.
- Let S' be the set of all ground instances of all clauses from S.
- By introducing propositional letters representing atomic sentences we may view S' as a (possibly infinite) propositional formula in clausal form.
- We may verify that it is unsatisfiable by resolution on propositional level.

E.g. for $S = \{\{P(x,y), R(x,y)\}, \{\neg P(c,y)\}, \{\neg R(x, f(x))\}\}$ *the set*
$$\begin{aligned} S' = \{&\{P(c,c), R(c,c)\}, \{P(c,f(c)), R(c,f(c))\}, \{P(f(c),f(c)), R(f(c),f(c))\}\ldots, \\ &\{\neg P(c,c)\}, \{\neg P(c,f(c))\}, \ldots, \{\neg R(c,f(c))\}, \{\neg R(f(c),f(f(c)))\}, \ldots\}\end{aligned}$$
is unsatisfiable since on propositional level
$$S' \supseteq \{\{P(c,f(c)), R(c,f(c))\}, \{\neg P(c,f(c))\}, \{\neg R(c,f(c))\}\} \vdash_R \Box.$$

Substitutions

Substitutions - examples

It is more efficient to use suitable substitutions. For example, in

a) $\{P(x), Q(x,a)\}, \{\neg P(y), \neg Q(b,y)\}$ substituting $x/b, y/a$ gives $\{P(b), Q(b,a)\}, \{\neg P(a), \neg Q(b,a)\}$, which resolves to $\{P(b), \neg P(a)\}$.

Or, substituting x/y and resolving through $P(y)$ gives $\{Q(y,a), \neg Q(b,y)\}$.

b) $\{P(x), Q(x,a), Q(b,y)\}, \{\neg P(v), \neg Q(u,v)\}$ substituting $x/b, y/a, u/b, v/a$ gives $\{P(b), Q(b,a)\}$, $\{\neg P(a), \neg Q(b,a)\}$, resolving to $\{P(b), \neg P(a)\}$.

c) $\{P(x), Q(x,z)\}, \{\neg P(y), \neg Q(f(y),y)\}$ substituting $x/f(z), y/z$ gives $\{P(f(z)), Q(f(z),z)\}$, $\{\neg P(z), \neg Q(f(z),z)\}$, resolving to $\{P(f(z)), \neg P(z)\}$.

Alternatively, substituting $x/f(a), y/a, z/a$ gives $\{P(f(a)), Q(f(a),a)\}, \{\neg P(a), \neg Q(f(a),a)\}$, which resolves to $\{P(f(a)), \neg P(a)\}$. But the previous substitution is more general.

Substitutions

- A *substitution* is a (finite) set $\sigma = \{x_1/t_1, \ldots, x_n/t_n\}$, where x_i's are distinct variables, t_i's are terms, and the term t_i is not x_i.
- If all t_i's are ground terms, then σ is a *ground substitution.*
- If all t_i's are distinct variables, then σ is a *renaming of variables.*
- An *expression* is a literal or a term.
- An *instance* of an expression E *by substitution* $\sigma = \{x_1/t_1, \ldots, x_n/t_n\}$ is the expression $E\sigma$ obtained from E by simultaneous replacing all occurrences of all x_i's for t_i's, respectively.
- For a set S of expressions, let $S\sigma = \{E\sigma \mid E \in S\}$.

Remark Since we substitute for all variables simultaneously, a possible occurrence of x_i *in* t_j *does not lead to a chain of substitutions.*

For example, for $S = \{P(x), R(y,z)\}$ *and* $\sigma = \{x/f(y,z), y/x, z/c\}$ *we have*

$$S\sigma = \{P(f(y,z)), R(x,c)\}.$$

Composing substitutions

For substitutions $\sigma = \{x_1/t_1, \ldots, x_n/t_n\}$ and $\tau = \{y_1/s_1, \ldots, y_n/s_n\}$ we define the *composition* of σ and τ to be

$$\sigma\tau = \{x_i/t_i\tau \mid x_i \in X,\ t_i\tau \text{ is not } x_i\} \cup \{y_j/s_j \mid y_j \in Y \setminus X\}$$

where $X = \{x_1, \ldots, x_n\}$, $Y = \{y_1, \ldots, y_m\}$.

For example, for $\sigma = \{x/f(y), w/v\}$, $\tau = \{x/a, y/g(x), v/w, u/c\}$ *we have* $\sigma\tau = \{x/f(g(x)), y/g(x), v/w, u$

Proposition (without proof) *For every expression E and subst. σ, τ, ϱ,*

1 $(E\sigma)\tau = E(\sigma\tau)$,

2 $(\sigma\tau)\varrho = \sigma(\tau\varrho)$.

Remark Composition of substitutions is not commutative, for the above:

$$\tau\sigma = \{x/a, y/g(f(y)), u/c, w/v\} \neq \sigma\tau.$$

Unification

Unification

Let $S = \{E_1, \ldots, E_n\}$ be a (finite) set of expressions.

- A *unification* of S is a substitution σ such that $E_1\sigma = E_2\sigma = \cdots = E_n\sigma$, i.e. $S\sigma$ is a singleton.
- S is *unifiable* if it has a unification.
- A unification σ of S is a *most general unification (mgu)* if for every

  unification τ of S there is a substitution λ such that $\tau = \sigma\lambda$.

For example, $S = \{P(f(x), y), P(f(a), w)\}$ *is unifiable by a most general unification* $\sigma = \{x/a, y/w\}$. *A unification* $\tau = \{x/a, y/b, w/b\}$ *is obtained as* $\sigma\lambda$ *for* $\lambda = \{w/b\}$. τ *is not mgu, it cannot give us* $\varrho = \{x/a, y/c, w/c\}$.

Observation If σ, τ are two most general unifications of S, they differ only in renaming of variables.

Unification algorithm

Let S be a (finite) nonempty set of expressions and p be the leftmost position in which some expressions of S differ. Then *the difference* in S is the set $D(S)$ of subexpressions of all expressions from S starting at the position p.

For example, $S = \{P(x, y), P(f(x), z), P(z, f(x))\}$ *has* $D(S) = \{x, f(x), z\}$.

Input Nonempty (finite) set of expressions S.

Output A most general unification σ of S or *"S is not unifiable"*.

(0) Let $S_0 := S$, $\sigma_0 := \emptyset$, $k := 0$. *(initialization)*

1 If S_k is a singleton, output $\sigma = \sigma_0\sigma_1 \cdots \sigma_k$. *(mgu of S)*

2 Check if $D(S_k)$ contains a variable x and a term t with no occurrence of x.

3 If not, output *"S is not unifiable"*.

4 Otherwise, $\sigma_{k+1} := \{x/t\}$, $S_{k+1} := S_k\sigma_{k+1}$, $k := k+1$, GOTO(1).

Remark The occurrence check of x in t in step (2) *can be* "expensive"*.*

Unification algorithm - an example

$$S = \{P(f(y,g(z)),h(b)),\ P(f(h(w),g(a)),t),\ P(f(h(b),g(z)),y)\}$$

1 $S_0 = S$ is not singleton, $D(S_0) = \{y, h(w), h(b)\}$ has a term $h(w)$ and a var. y not occurring in $h(w)$. Then $\sigma_1 = \{y/h(w)\}$, $S_1 = S_0\sigma_1$:

$S_1 = \{P(f(h(w),g(z)),h(b)),\ P(f(h(w),g(a)),t),\ P(f(h(b),g(z)),h(w))\}$

2 $D(S_1) = \{w, b\}$, $\sigma_2 = \{w/b\}$, $S_2 = S_1\sigma_2$, i.e.

$S_2 = \{P(f(h(b),g(z)),h(b)),\ P(f(h(b),g(a)),t)\}$

3 $D(S_2) = \{z, a\}$, $\sigma_3 = \{z/a\}$, $S_3 = S_2\sigma_3$, i.e.

$S_3 = \{P(f(h(b),g(a)),h(b)),\ P(f(h(b),g(a)),t)\}$

4 $D(S_3) = \{h(b), t\}$, $\sigma_4 = \{t/h(b)\}$, $S_4 = S_3\sigma_4$, i.e.

$S_4 = \{P(f(h(b),g(a)),h(b))\}$

5 S_4 is a singleton and a most general unification of S is

$\sigma = \{y/h(w)\}\{w/b\}\{z/a\}\{t/h(b)\} = \{y/h(b), w/b, z/a, t/h(b)\}$

Unification algorithm - correctness

Proposition *The unification algorithm outputs a correct answer in finite time for any input S, i.e. a most general unification σ of S or it detects that S is not unifiable.* $(*)$ *Moreover, for every unification τ of S it holds that $\tau = \sigma\tau$.*

Proof It eliminates one variable in each round, so it ends in finite time.

- If it ends negatively, $D(S_k)$ is not unifiable, and neither is S.
- If it outputs $\sigma = \sigma_0\sigma_1 \cdots \sigma_k$, clearly σ is a unification of S.
- If we show the property $(*)$ for σ, then σ is a most general unification of S.

(1) Let τ be a unification of S. We show $\tau = \sigma_0\sigma_1 \cdots \sigma_i\tau$ for all $i \leq k$.

(2) For $i = 0$ it holds. Let $\sigma_{i+1} = \{x/t\}$ and assume $\tau = \sigma_0\sigma_1 \cdots \sigma_i\tau$.

(3) It suffices to show that $v\sigma_{i+1}\tau = v\tau$ for every variable v.

(4) If $v \neq x$, $v\sigma_{i+1} = v$, so (3) holds. Otherwise $v = x$ and $v\sigma_{i+1} = x\sigma_{i+1} = t$.

(5) Since τ unifies $S_i = S\sigma_0\sigma_1 \cdots \sigma_i$ and both the variable x and the term t are in $D(S_i)$, τ has to unify x and t, i.e. $t\tau = x\tau$, as required for (3). □

Resolution proof

The general resolution rule

Let C_1, C_2 be clauses with distinct variables such that

$$C_1 = C_1' \sqcup \{A_1, \ldots, A_n\}, \quad C_2 = C_2' \sqcup \{\neg B_1, \ldots, \neg B_m\},$$

where $S = \{A_1, \ldots, A_n, B_1, \ldots, B_m\}$ is unifiable and $n, m \geq 1$. Then the clause

$$C = C_1'\sigma \cup C_2'\sigma,$$

where σ is a most general unification of S, is the *resolvent* of C_1 and C_2.

For example, in clauses $\{P(x), Q(x, z)\}$ and $\{\neg P(y), \neg Q(f(y), y)\}$ we can unify $S = \{Q(x, z), Q(f(y), y)\}$ applying a most general unification $\sigma = \{x/f(y), z/y\}$, and then resolve to a clause $\{P(f(y)), \neg P(y)\}$.

Remark The condition on distinct variables can be satisfied by renaming variables apart. This is sometimes necessary, e.g. from $\{\{P(x)\}, \{\neg P(f(x))\}\}$ after renaming we can get □, but $\{P(x), P(f(x))\}$ is not unifiable.

Resolution proof

We have the same notions as in propositional logic, up to renaming variables.

- *Resolution proof (deduction)* of a clause C from a formula S is a finite sequence $C_0, \ldots, C_n = C$ such that for every $i \leq n$, we have $C_i = C_i'\sigma$ for some $C_i' \in S$ and a renaming of variables σ, or C_i is a resolvent of some previous clauses.

- A clause C is (resolution) *provable* from S, denoted by $S \vdash_R C$, if it has a resolution proof from S.

- A (resolution) *refutation* of a formula S is a resolution proof of □ from S.

- S is (resolution) *refutable* if $S \vdash_R \square$.

Remark Elimination of several literals at once is sometimes necessary, e.g. $S = \{\{P(x), P(y)\}, \{\neg P(x), \neg P(y)\}\}$ is resolution refutable, but it has no refutation that eliminates only a single literal in each resolution step.

Resolution in predicate logic - an example

Consider $T = \{\neg P(x,x),\ P(x,y) \rightarrow P(y,x),\ P(x,y) \wedge P(y,z) \rightarrow P(x,z)\}$.

Is $T \models (\exists x)\neg P(x, f(x))$? Equivalently, is the following T' unsatisfiable?

$$T' = \{\{\neg P(x,x)\}, \{\neg P(x,y), P(y,x)\}, \{\neg P(x,y), \neg P(y,z), P(x,z)\}, \{P(x, f(x))\}\}$$

chapters/files-sub/rezolucePLpriklad.pdf

Soundness and completeness

Soundness of resolution

First we show soundness of the general resolution rule.

Proposition Let C be a resolvent of clauses C_1, C_2. Then for every L-structure $\mathcal{A}$:

$$\mathcal{A} \models C_1 \text{ and } \mathcal{A} \models C_2 \quad \Rightarrow \quad \mathcal{A} \models C.$$

Proof Let $C_1 = C_1' \sqcup \{A_1, \ldots, A_n\}$, $C_2 = C_2' \sqcup \{\neg B_1, \ldots, \neg B_m\}$, σ be a most general unification for $S = \{A_1, \ldots, A_n, B_1, \ldots, B_m\}$, and $C = C_1'\sigma \cup C_2'\sigma$.

- Since C_1, C_2 are open, it holds also $\mathcal{A} \models C_1\sigma$ and $\mathcal{A} \models C_2\sigma$.
- We have $C_1\sigma = C_1'\sigma \cup \{S\sigma\}$ and $C_2\sigma = C_2'\sigma \cup \{\neg(S\sigma)\}$.
- We show $\mathcal{A} \models C[e]$ for every e. If $\mathcal{A} \models S\sigma[e]$, then $\mathcal{A} \models C_2'\sigma[e]$, and thus $\mathcal{A} \models C[e]$. Otherwise $\mathcal{A} \not\models S\sigma[e]$, so $\mathcal{A} \models C_1'\sigma[e]$, and thus $\mathcal{A} \models C[e]$. □

Theorem (soundness) *If S is resolution refutable, then S is unsatisfiable.*

Proof Let $S \vdash_R \square$. Suppose $\mathcal{A} \models S$ for some structure $\mathcal{A}$. By soundness of the general resolution rule we have $\mathcal{A} \models \square$, which is impossible. □

Lifting lemma

A resolution proof on propositional level can be "lifted" to predicate level.

Lemma *Let $C_1^* = C_1\tau_1$, $C_2^* = C_2\tau_2$ be ground instances of clauses C_1, C_2 with distinct variables and C^* be a resolvent of C_1^* a C_2^*. Then there exists a resolvent C of C_1 and C_2 such that $C^* = C\tau_1\tau_2$ is a ground instance of C.*

Proof Let C^* be a resolvent of C_1^*, C_2^* through a literal $P(t_1, \ldots, t_k)$.

- We have $C_1 = C_1' \sqcup \{A_1, \ldots, A_n\}$ and $C_2 = C_2' \sqcup \{\neg B_1, \ldots, \neg B_m\}$, where $\{A_1, \ldots, A_n\}\tau_1 = \{P(t_1, \ldots, t_k)\}$ & $\{\neg B_1, \ldots, \neg B_m\}\tau_2 = \{\neg P(t_1, \ldots, t_k)\}$

- Thus $(\tau_1\tau_2)$ unifies $S = \{A_1, \ldots, A_n, B_1, \ldots, B_m\}$ and if σ is mgu of S from the unif. algorithm, then $C = C_1'\sigma \cup C_2'\sigma$ is a resolvent of C_1, C_2.

- Moreover, $(\tau_1\tau_2) = \sigma(\tau_1\tau_2)$ by the property $(*)$ for σ, and hence

$$\begin{aligned} C\tau_1\tau_2 &= (C_1'\sigma \cup C_2'\sigma)\tau_1\tau_2 = C_1'\sigma\tau_1\tau_2 \cup C_2'\sigma\tau_1\tau_2 = C_1'\tau_1 \cup C_2'\tau_2 \\ &= (C_1 \setminus \{A_1, \ldots, A_n\})\tau_1 \cup (C_2 \setminus \{\neg B_1, \ldots, \neg B_m\})\tau_2 \\ &= (C_1^* \setminus \{P(t_1, \ldots, t_k)\}) \cup (C_2^* \setminus \{\neg P(t_1, \ldots, t_k)\}) = C^*. \quad \square \end{aligned}$$

Completeness

Corollary *Let S' be the set of all ground instances of clauses of a formula S. If $S' \vdash_R C'$ (on propositional level) where C' is a ground clause, then $C' = C\sigma$ for some clause C and a ground substitution σ such that $S \vdash_R C$ (on pred. level).*

Proof By induction on the length of resolution proof using lifting lemma. □

Theorem (completeness) *If S is unsatisfiable, then $S \vdash_R \square$.*

Proof If S is unsatisfiable, then by the (corollary of) Herbrand's theorem, also the set S' of all ground instances of clauses of S is unsatisfiable.

- By completeness of resolution in prop. logic, $S' \vdash_R \square$ (on prop. level).

- By the above corollary, there is a clause C and a ground substitution σ such that $\square = C\sigma$ and $S \vdash_R C$ (on pred. level).

- The only clause that has $\square$ as a ground instance is the clause $C = \square$. □

Linear resolution

As in propositional logic, the resolution method can be significantly refined (without using completeness).

- A *linear proof* of a clause C from a formula S is a finite sequence

  of pairs $(C_0, B_0), \ldots, (C_n, B_n)$ such that $C_0 \in S$ and for every $i \leq n$

  i) $B_i \in S$ or $B_i = C_j$ for some $j < i$, and

  ii) C_{i+1} is a resolvent of C_i and B_i where $C_{n+1} = C$.

- C_0 is called a *starting* clause, C_i a *central* clause, B_i a *side* clause.

- C is *linearly provable* from S, $S \vdash_L C$, if it has a linear proof from S.

- A *linear refutation* of S is a linear proof of $\square$ from S.

- S is *linearly refutable* if $S \vdash_L \Box$.

Theorem *S is linearly refutable, if and only if it is unsatisfiable.*

Proof ($\Rightarrow$) Every linear proof can be transformed to a (general) resolution proof. ($\Leftarrow$) Follows from completeness of propositional resolution, the lifting lemma preserves linearity of proofs. $\Box$

LI-resolution

As in prop. logic, for Horn formulas we can further refine linear resolution.

- *LI-resolution* ("linear input") from S is a linear resolution from S in which every side clause B_i is a variant of a clause from S. We write $S \vdash_{LI} C$ to denote that C is provable by LI-resolution from S.
- a *Horn clause* is a clause containing at most one positive literal,
- a *Horn formula* is a (possibly infinite) set of Horn clauses,
- a *fact* is a (Horn) clause $\{p\}$ where p is a positive literal,
- a *rule* is a (Horn) clause with exactly one positive literal and at least one negative literal. Rules and facts are *program clauses*,
- a *goal* is a nonempty (Horn) clause with only negative literals.

Theorem *If T is a satisfiable Horn formula but $T \cup \{G\}$ is unsat. for some goal G, then $\Box$ has a LI-resolution from $T \cup \{G\}$ with starting clause G.*

Proof Follows from Herbrand's Theorem, the same theorem in propositional logic, and the lifting lemma.

A program in Prolog

A (Prolog) *program* is a Horn formula containing only program clauses, i.e. facts and rules.

$son(X,Y) : -father(Y,X), man(X)$	$\{son(X,Y), \neg father(Y,X), \neg man(X)\}$
$son(X,Y) : -mother(Y,X), man(X)$	$\{son(X,Y), \neg mother(Y,X), \neg man(X)\}$
$man(john).$	$\{man(john)\}$
$father(george, john).$	$\{father(george, john)\}$
$mother(julie, john).$	$\{mother(julie, john)\}$

$? - son(john, X)$	$P \models (\exists X) son(john, X)$	$\{\neg son(john, X)\}$

We want to know if the given query follows from the program.

Theorem *Let P be a program and $G = \{\neg A_1, \ldots, \neg A_n\}$ a goal in variables $X_1, \ldots, X_m$. TFAE:*

(1) $P \models (\exists X_1) \ldots (\exists X_m)(A_1 \wedge \cdots \wedge A_n)$*, if and only if*

(2) $\Box$ *has a LI-resolution from* $P \cup \{G\}$ *staring with (a variant of) the goal* G.

LI-resolution over the program

If the answer to the query is positive, we also want to know the output substitution. The *output substitution* σ for LI-resolution of $\square$ from $P \cup \{G\}$ starting from $G = \{\neg A_1, \dots, \neg A_n\}$ is the composition of mgu from individual steps (only for variables of G. Note that:

$$P \models (A_1 \wedge \dots A_n)\sigma$$

lecture/figures/resolution-prolog-picture.png

1.8 Hilbert's calculus

Introduction

Hilbert's calculus in predicate logic

- basic connectives and quantifier: $\neg$, $\rightarrow$, $(\forall x)$ (others are derived)

- allows to prove any formula (not just sentences)
- *logical axioms* (schemes of axioms):

$$\begin{array}{rcl}
(i) & \varphi \to (\psi \to \varphi) & \\
(ii) & (\varphi \to (\psi \to \chi)) \to ((\varphi \to \psi) \to (\varphi \to \chi)) & \\
(iii) & (\neg\varphi \to \neg\psi) \to (\psi \to \varphi) & \\
(iv) & (\forall x)\varphi \to \varphi(x/t) & \text{if } t \text{ is substitutable for } x \text{ to } \varphi \\
(v) & (\forall x)(\varphi \to \psi) \to (\varphi \to (\forall x)\psi) & \text{if } x \text{ is not free in } \varphi
\end{array}$$

  where φ, ψ, χ are any formulas (of a given language), t is any term,

  and x is any variable

- in a language with equality we include also the axioms of equality
- *rules of inference*

$$\frac{\varphi,\ \varphi \to \psi}{\psi} \text{ (modus ponens)}, \qquad \frac{\varphi}{(\forall x)\varphi} \text{ (generalization)}$$

Hilbert-style proofs

A *proof* (in *Hilbert-style*) of a formula φ from a theory T is a finite sequence $\varphi_0, \dots, \varphi_n = \varphi$ of formulas such that for every $i \le n$

- φ_i is a logical axiom or $\varphi_i \in T$ (an axiom of the theory), or
- φ_i can be inferred from the previous formulas applying a rule of inference.

A formula φ is *provable* from T if it has a proof from T, denoted by $T \vdash_H \varphi$.

Theorem (soundness) *For every T and φ, $T \vdash_H \varphi \;\Rightarrow\; T \models \varphi$.*

Proof

- If φ is an axiom (logical or from T), then $T \models \varphi$ (l. axioms are tautologies),
- if $T \models \varphi$ and $T \models \varphi \to \psi$, then $T \models \psi$, i.e. modus ponens is sound,
- if $T \models \varphi$, then $T \models (\forall x)\varphi$, i.e. generalization is sound,
- thus every formula in a proof from T is valid in T. □

Remark The completeness holds as well, i.e. $T \models \varphi \Rightarrow T \vdash_H \varphi$.

1.9 Model theory

Algebraic theories

Basic algebraic theories

- theory of *groups* in the language $L = \langle +, -, 0 \rangle$ with equality:

$$x + (y + z) = (x + y) + z \qquad \text{(associativity of } +\text{)}$$

$$0 + x = x = x + 0 \qquad \text{(0 is neutral to } +\text{)}$$

$$x + (-x) = 0 = (-x) + x \qquad (-x \text{ is inverse of } x)$$

- theory of *Abelian groups* has moreover ax. $x + y = y + x$ (commutativity)
- theory of *rings* in $L = \langle +, -, \cdot, 0, 1 \rangle$ with equality has additionally

$$1 \cdot x = x = x \cdot 1 \qquad \text{(1 is neutral to } \cdot\text{)}$$

$$x \cdot (y \cdot z) = (x \cdot y) \cdot z \qquad \text{(associativity of } \cdot\text{)}$$

$$x \cdot (y + z) = x \cdot y + x \cdot z,\ (x + y) \cdot z = x \cdot z + y \cdot z \qquad \text{(distributivity)}$$

- theory of *commutative rings* has moreover the axiom $x \cdot y = y \cdot x$ (commutativity)
- theory of *fields* in the same language has additionally the axioms

$$x \neq 0 \rightarrow (\exists y)(x \cdot y = 1) \qquad \text{(existence of inverses to } \cdot\text{)}$$

$$0 \neq 1 \qquad \text{(nontriviality)}$$

Elementary equivalence

Theories of structures

What properties hold in particular structures?

The *theory of a structure* $\mathcal{A}$ is the set $\mathrm{Th}(\mathcal{A})$ of all sentences (of the same language) that are valid in $\mathcal{A}$.

Observation For every structure $\mathcal{A}$ and a theory T of a language L,

(i) $\mathrm{Th}(\mathcal{A})$ *is a complete theory,*

(ii) if $\mathcal{A} \models T$, then $\mathrm{Th}(\mathcal{A})$ *is a simple (complete) extension of T,*

(iii) if $\mathcal{A} \models T$ and T is complete, then $\mathrm{Th}(\mathcal{A})$ *is equivalent with T, i.e. $\theta^L(T) = \mathrm{Th}(\mathcal{A})$.*

E.g. $\mathrm{Th}(\underline{\mathbb{N}})$ *where $\underline{\mathbb{N}} = \langle \mathbb{N}, S, +, \cdot, 0, \leq \rangle$ is the arithmetics of natural numbers.*

Remark Later, we will see that $\mathrm{Th}(\underline{\mathbb{N}})$ *is (algorithmically) undecidable although it is complete.*

Elementary equivalence

- Structures $\mathcal{A}$ and $\mathcal{B}$ of a language L are *elementarily equivalent*, denoted $\mathcal{A} \equiv \mathcal{B}$, if they satisfy the same sentences (of L), i.e. $\mathrm{Th}(\mathcal{A}) = \mathrm{Th}(\mathcal{B})$.

  For example, $\langle \mathbb{R}, \leq \rangle \equiv \langle \mathbb{Q}, \leq \rangle$ *and* $\langle \mathbb{Q}, \leq \rangle \not\equiv \langle \mathbb{Z}, \leq \rangle$ *since every element has an immediate successor in* $\langle \mathbb{Z}, \leq \rangle$ *but not in* $\langle \mathbb{Q}, \leq \rangle$*.*

- T is complete iff it has a single model, up to elementary equivalence.

  For example, the theory of dense linear orders without ends (DeLO).

How to describe models of a given theory (up to elementary equivalence)?

Observation For every models $\mathcal{A}$*,* $\mathcal{B}$ *of a theory* T*,* $\mathcal{A} \equiv \mathcal{B}$ *if and only if* $\mathrm{Th}(\mathcal{A})$*,* $\mathrm{Th}(\mathcal{B})$ *are equivalent (simple complete extensions of* T*).*

Remark If we can describe effectively (recursively) for a given theory T *all simple complete extensions of* T*, then* T *is (algorithmically) decidable.*

Simple complete extensions - an example

The theory $DeLO^*$ of dense linear orders of $L = \langle \leq \rangle$ with equality:

$$
\begin{array}{lr}
x \leq x & \text{(reflexivity)} \\
x \leq y \;\wedge\; y \leq x \;\rightarrow\; x = y & \text{(antisymmetry)} \\
x \leq y \;\wedge\; y \leq z \;\rightarrow\; x \leq z & \text{(transitivity)} \\
x \leq y \;\vee\; y \leq x & \text{(dichotomy)} \\
x < y \;\rightarrow\; (\exists z)\,(x < z \;\wedge\; z < y) & \text{(density)} \\
(\exists x)(\exists y)(x \neq y) & \text{(nontriviality)}
\end{array}
$$

where '$x < y$' is a shortcut for '$x \leq y \;\wedge\; x \neq y$'.

Let φ, ψ be the sentences $(\exists x)(\forall y)(x \leq y)$, resp. $(\exists x)(\forall y)(y \leq x)$. We will show that the following are all (inequivalent) simple complete extensions of the theory $DeLO^*$:

$$
\begin{array}{ll}
DeLO \;\;= DeLO^* \cup \{\neg\varphi, \neg\psi\}, & DeLO^{\pm} = DeLO^* \cup \{\varphi, \psi\}, \\
DeLO^{+} = DeLO^* \cup \{\neg\varphi, \psi\}, & DeLO^{-} = DeLO^* \cup \{\varphi, \neg\psi\}
\end{array}
$$

Corollary of the Löwenheim-Skolem theorem

We already know the following theorem, by a canonical model (with =).

Theorem *Let* T *be a consistent theory of a countable language* L*. If* L *is without equality, then* T *has a countably infinite model. If* L *is with equality, then* T *has a model that is countable (finite or countably infinite).*

Corollary *For every structure $\mathcal{A}$ of a countable language without equality there exists a countably infinite structure $\mathcal{B}$ with $\mathcal{A} \equiv \mathcal{B}$.*

Proof Th($\mathcal{A}$) is consistent since it has a model $\mathcal{A}$. By the previous theorem, it has a countably inf. model $\mathcal{B}$. Since Th($\mathcal{A}$) is complete, we have $\mathcal{A} \equiv \mathcal{B}$. □

Corollary *For every infinite structure $\mathcal{A}$ of a countable language with equality there exists a countably infinite structure $\mathcal{B}$ with $\mathcal{A} \equiv \mathcal{B}$.*

Proof Similarly as above. Since the sentence *"there is exactly n elements"* is false in $\mathcal{A}$ for all n and $\mathcal{A} \equiv \mathcal{B}$, it follows that B is infinite. □

A countable algebraically closed field

We say that a field $\mathcal{A}$ is *algebraically closed* if every polynomial (of nonzero degree) has a root in $\mathcal{A}$; that is, for every $n \geq 1$ we have

$$\mathcal{A} \models (\forall x_{n-1}) \ldots (\forall x_0)(\exists y)(y^n + x_{n-1} \cdot y^{n-1} + \cdots + x_1 \cdot y + x_0 = 0)$$

where y^k is a shortcut for the term $y \cdot y \cdot \ \cdots \ \cdot y$ ($\cdot$ applied $(k-1)$-times).

For example, the field $\underline{\mathbb{C}} = \langle \mathbb{C}, +, -, \cdot, 0, 1 \rangle$ is algebraically closed, whereas the fields $\underline{\mathbb{R}}$ and $\underline{\mathbb{Q}}$ are not (since the polynomial $x^2 + 1$ has no root in them).

Corollary *There exists a countable algebraically closed field.*

Proof By the previous corollary, there is a countable structure elementarily equivalent with the field $\underline{\mathbb{C}}$. Hence it is algebraically closed as well. □

Isomorphism

Isomorphisms of structures

Let $\mathcal{A}$ and $\mathcal{B}$ be structures of a language $L = \langle \mathcal{F}, \mathcal{R} \rangle$.

- A bijection $h \colon A \to B$ is an *isomorphism* of structures $\mathcal{A}$ and $\mathcal{B}$ if

  (i) $h(f^A(a_1, \ldots, a_n)) = f^B(h(a_1), \ldots, h(a_n))$
  for every n-ary function symbol $f \in \mathcal{F}$ and every $a_1, \ldots, a_n \in A$,

  (ii) $R^A(a_1, \ldots, a_n) \quad \Leftrightarrow \quad R^B(h(a_1), \ldots, h(a_n))$
  for every n-ary relation symbol $R \in \mathcal{R}$ and every $a_1, \ldots, a_n \in A$.

- $\mathcal{A}$ and $\mathcal{B}$ are *isomorphic* (via h), denoted $\mathcal{A} \simeq \mathcal{B}$ ($\mathcal{A} \simeq_h \mathcal{B}$), if there is

  an isomorphism h of $\mathcal{A}$ and $\mathcal{B}$. We also say $\mathcal{A}$ is *isomorphic with* $\mathcal{B}$.

- An *automorphism* of a structure $\mathcal{A}$ is an isomorphism of $\mathcal{A}$ with $\mathcal{A}$.

For example, the power set algebra $\underline{\mathcal{P}(X)} = \langle \mathcal{P}(X), -, \cap, \cup, \emptyset, X \rangle$ with $X = n$ is isomorphic to the Boolean algebra $\underline{{}^n2} = \langle {}^n2, -_n, \wedge_n, \vee_n, 0_n, 1_n \rangle$ via $h : A \mapsto \chi_A$ where χ_A is the characteristic function of the set $A \subseteq X$.

Isomorphisms and semantics

We will see that isomorphism preserves semantics.

Proposition *Let $\mathcal{A}$ and $\mathcal{B}$ be structures of a language $L = \langle \mathcal{F}, \mathcal{R} \rangle$. A bijection $h\colon A \to B$ is an isomorphism of $\mathcal{A}$ and $\mathcal{B}$ if and only if both*

$$\begin{array}{lll} (i) & h(t^A[e]) = t^B[he] & \textit{for every term } t \textit{ and } e\colon \mathrm{Var} \to A, \\ (ii) & \mathcal{A} \models \varphi[e] \;\Leftrightarrow\; \mathcal{B} \models \varphi[he] & \textit{for every formula } \varphi \textit{ and } e\colon \mathrm{Var} \to A. \end{array}$$

Proof ($\Rightarrow$) By induction on the structure of t, resp. φ.

($\Leftarrow$) By applying (i) for each term $f(x_1, \ldots, x_n)$ or (ii) for each atomic formula $R(x_1, \ldots, x_n)$ and assigning $e(x_i) = a_i$ we verify that h is an isomorphism. □

Corollary *For every structures $\mathcal{A}$ and $\mathcal{B}$ of the same language,*

$$\mathcal{A} \simeq \mathcal{B} \quad \Rightarrow \quad \mathcal{A} \equiv \mathcal{B}.$$

Remark The other implication ($\Leftarrow$) does not hold in general. For example, $\langle \mathbb{Q}, \leq \rangle \equiv \langle \mathbb{R}, \leq \rangle$ *but* $\langle \mathbb{Q}, \leq \rangle \not\simeq \langle \mathbb{R}, \leq \rangle$ *since* $|\mathbb{Q}| = \omega$ *and* $|\mathbb{R}| = 2^\omega$.

Finite models in language with equality

Proposition *For every finite structures $\mathcal{A}$, $\mathcal{B}$ of a language with equality,*

$$\mathcal{A} \equiv \mathcal{B} \quad \Rightarrow \quad \mathcal{A} \simeq \mathcal{B}.$$

Proof $|A| = |B|$ since we can express *"there are exactly n elements"*.

- Let $\mathcal{A}'$ be expansion of $\mathcal{A}$ to $L' = L \cup \{c_a\}_{a \in A}$ by names of elements.
- We show that $\mathcal{B}$ has an expansion $\mathcal{B}'$ to L' such that $\mathcal{A}' \equiv \mathcal{B}'$. Then clearly $h\colon a \mapsto c_a^{B'}$ is an isomorfism of $\mathcal{A}'$ to $\mathcal{B}'$, and thus also $\mathcal{A}$ to $\mathcal{B}$.
- If suffices to find $b \in B$ for every $c_a^{A'} = a \in A$ s.t. $\langle \mathcal{A}, a \rangle \equiv \langle \mathcal{B}, b \rangle$.
- Let Ω be set of all formulas $\varphi(x)$ s.t. $\langle \mathcal{A}, a \rangle \models \varphi(x/c_a)$, i.e. $\mathcal{A} \models \varphi[e(x/a)]$
- Since A is finite, there are finitely many formulas $\varphi_0(x), \ldots, \varphi_m(x)$ such that for every $\varphi \in \Omega$ it holds $\mathcal{A} \models \varphi \leftrightarrow \varphi_i$ for some i.
- Since $\mathcal{B} \equiv \mathcal{A} \models (\exists x) \bigwedge_{i \leq m} \varphi_i$, there exists $b \in B$ s.t. $\mathcal{B} \models \bigwedge_{i \leq m} \varphi_i[e(x/b)]$.
- Hence for every $\varphi \in \Omega$ it holds $\mathcal{B} \models \varphi[e(x/b)]$, i.e. $\langle \mathcal{B}, b \rangle \models \varphi(x/c_a)$. □

Corollary *If a complete theory T in a language with equality has a finite model, then all models of T are isomorphic.*

Definable sets and automorphisms

The set *defined by* $\varphi(\bar{x}, \bar{y})$ *with parameters* $\bar{b} \in A^{|\bar{y}|}$ *in* $\mathcal{A}$ is

$$\varphi^{\mathcal{A},\bar{b}}(\bar{x}, \bar{y}) = \{\bar{a} \in A^{|\bar{x}|} \mid \mathcal{A} \models \varphi[e(\bar{x}/\bar{a}, \bar{y}/\bar{b})]\}$$

Proposition Let $D \subseteq A^n$ be a set definable in a structure $\mathcal{A}$ with parameters $\bar{b}$ and let h be an automorphism of $\mathcal{A}$ which is identical on $\bar{b}$. Then $h[D] = D$.

Proof Let $D = \varphi^{\mathcal{A},\bar{b}}(\bar{x}, \bar{y})$. Then for any $\bar{a} \in A^{|\bar{x}|}$:

$$\begin{aligned} \bar{a} \in D &\Leftrightarrow \mathcal{A} \models \varphi[e(\bar{x}/\bar{a}, \bar{y}/\bar{b})] \\ &\Leftrightarrow \mathcal{A} \models \varphi[(e \circ h)(\bar{x}/\bar{a}, \bar{y}/\bar{b})] \\ &\Leftrightarrow \mathcal{A} \models \varphi[e(\bar{x}/h(\bar{a}), \bar{y}/h(\bar{b}))] \\ &\Leftrightarrow \mathcal{A} \models \varphi[e(\bar{x}/h(\bar{a}), \bar{y}/\bar{b})] \\ &\Leftrightarrow h(\bar{a}) \in D \end{aligned}$$

Example: find automorphisms of a given graph.

Categoricity

Categoricity

- The (isomorphism) *spectrum* of a theory T is given by the number $I(\kappa, T)$ of mutually nonisomorphic models of T for every cardinality κ.
- A theory T is κ*-categorical* if it has exactly one (up to isomorphism) model of cardinality κ, i.e. $I(\kappa, T) = 1$.

Proposition *The theory DeLO* (i.e. "without ends") *is* ω*-categorical.*

Proof Let $\mathcal{A}$, $\mathcal{B} \models DeLO$ with $A = \{a_i\}_{i\in\mathbb{N}}$, $B = \{b_i\}_{i\in\mathbb{N}}$. By induction on n we can find injective partial functions $h_n \subseteq h_{n+1} \subset A \times B$ preserving the ordering s.t. $\{a_i\}_{i<n} \subseteq \mathrm{dom}(h_n)$ and $\{b_i\}_{i<n} \subseteq \mathrm{rng}(h_n)$. Then $\mathcal{A} \simeq \mathcal{B}$ via $h = \bigcup_n h_n$. □

Similarly we obtain that (e.g.) $\mathcal{A} = \langle \mathbb{Q}, \leq \rangle$, $\mathcal{A} \restriction (0, 1]$, $\mathcal{A} \restriction [0, 1)$, $\mathcal{A} \restriction [0, 1]$ *are (up to isomorphism) all countable models of* $DeLO^*$. *Then*

$$I(\kappa, DeLO^*) = \begin{cases} 0 & \text{for } \kappa \in \mathbb{N}, \\ 4 & \text{for } \kappa = \omega. \end{cases}$$

ω-categorical criterium of completeness

Theorem *Let L be at most countable language.*

(i) If a theory T in L without equality is ω-categorical, then it is complete.

(ii) If a theory T in L with equality is ω-categorical and without finite models, then it is complete.

Proof Every model of T is elementarily equivalent with some countably infinite model of T, but such model is unique up to isomorphism. Thus all models of T are elementarily equivalent, i.e. T is complete. □

For example, DeLO, DeLO$^+$, DeLO$^-$, DeLO$^\pm$ are complete and they are the all (mutually nonequivalent) simple complete extensions of DeLO*.

Remark A similar criterium holds also for cardinalities bigger than ω.

Axiomatizability

Can the given part of the world be "nicely" described?

Let $K \subseteq M(L)$ be a class of L-structures. We say that K is

- *axiomatizable* if there exists a theory T such that $M(T) = K$,
- *finitely axiomatizable* if is it axiomatizable by a finite theory, and
- *openly axiomatizable* if is it axiomatizable by an open theory.
- a theory T is finitely [openly] axiomatizable if $M(T)$ is.

Observation If K is axiomatizable, then it is closed under elementary equivalence. For example:

- linear orders are finitely and openly axiomatizable,
- fields are finitely but not openly axiomatizable, and
- infinite groups are axiomatiable, but not finitely axiomatizable.

A consequence of compactness

Theorem If a theory T has for every $n > 0$ an at least n-element model, then T has an infinite model.

Proof Obvious for languages without equality, conside L with $=$.

- Consider the extension $T' = T \cup \{c_i \neq c_j \mid i \neq j\}$ of T in the language extended by countably infinitely many new constant symbols c_i.
- By assumption, every finite part of T' has a model.
- By the Compactness theorem, T' has a model $\mathcal{A}'$ but that model is necessarily infinite.
- The redukt of $\mathcal{A}'$ to the original language is an infinite modle of T.

Corollary If a theory T has for every $n > 0$ an at least n-element model, then the class of all finite models of T is not axiomatizable.

For example, finite groups, finite fields etc. are not axiomatizable. But the class of all infinite models of a theory T in a language with equality is axiomatizable.

Finite axiomatizability

Theorem Let $K \subseteq M(L)$ and $\bar{K} = M(L) \setminus K$, where L is a langauge. Then K is finitely axiomatizable, if and only if both K and $\bar{K}$ are axiomatizable.

Proof ($\Rightarrow$) If T is a finite axiomatization of K in closed form, then the theory with a single axiom $\bigvee_{\varphi \in T} \neg\varphi$ axiomatizes $\bar{K}$.

($\Leftarrow$) To prove this implication:

- Let T, S be theories of a language L such that $M(T) = K$ and $M(S) = \bar{K}$.
- Then $M(T \cup S) = M(T) \cap M(S) = \emptyset$ and by compactness, there exist finite $T' \subseteq T$ and finite $S' \subseteq S$ such that $\emptyset = M(T' \cup S') = M(T') \cap M(S')$.
- The finite theory T' axiomatizes K, since

$$M(T) \subseteq M(T') \subseteq \overline{M(S')} \subseteq \overline{M(S)} = M(T).$$

Finite axiomatizability – an example

Let T be the theory of fields. We say that a field $\mathcal{A} = \langle A, +, -, \cdot, 0, 1\rangle$ is

- *of characteristic 0* if there is no $p \in \mathbb{N}^+$ such that $\mathcal{A} \models p1 = 0$ where $p1$ denotes the term $1 + 1 + \cdots + 1$ (where $+$ is applied $(p-1)$-times).
- *of characteristic* p, where p is a prime number, if p is smallest such that $\mathcal{A} \models p1 = 0$
- The class of fields of characteristic p, for a fixed prime p, is finitely axiomatizable by the theory $T \cup \{p1 = 0\}$.
- The class of fields of characteristic 0 is axiomatized by an (infinite) theory $T' = T \cup \{p1 \neq 0 \mid p \in \mathbb{N}^+\}$.

Proposition The class K of field of characteristic 0 is not finitely axiomatizable.
Proof It suffices to show that $\bar{K}$ is not axiomatizable. If $M(S) = \bar{K}$, then $S' = S \cup T'$ has a model $\mathcal{B}$, because every finite $S^* \subseteq S'$ has a model (a field of characteristic p' where p' is a prime greater than any prime p appearing in the axioms of S^*), But then $\mathcal{B} \in M(S) = \bar{K}$ and at the same time $B \in M(T') = K$ which is not possible.

Open axiomatizability

Theorem If a theory T is openly axiomatizable, then every substructure of a model of T is also a model of T.

Proof Let T' be an open axiomatization of $M(T)$, $\mathcal{A} \models T'$ and $\mathcal{B} \subseteq \mathcal{A}$. We know that for every $\varphi \in T'$, $\mathcal{B} \models \varphi$ because φ is open. Therefore $\mathcal{B}$ is a model of T'.

Note The converse is also true: if every substructure of a model of a theory T is a model of T as well, then T is openly axiomatizable.

For example, the theory DeLO is not openly axiomatizable, because for example a finite substructure of a model of DeLO is not a model of DeLO.

As another example, at most n-element groups, for a fixed $n > 1$, are openly axiomatizable:

$$T \cup \{ \bigvee_{i,j \leq n, i \neq j} x_i = x_j \}$$

where T is the (open) theory of groups

1.10 Undecidability

Introduction

Recursive and recursively enumerable sets

Which problems are algorithmically solvable?

- The notion of *"algorithm"* can be rigorously formalized (e.g. by TM).
- We may encode decision problems into sets of natural numbers corresponding to the positive instances (with answer `yes`). For example,
  $$SAT = \{\ulcorner\varphi\urcorner \mid \varphi \text{ is a satisfiable proposition in CNF}\}.$$
- A set $A \subseteq \mathbb{N}$ is *recursive* if there is an algorithm that for every input $x \in \mathbb{N}$ halts and correctly tells whether or not $x \in A$. We say that such algorithm decides $x \in A$.
- A set $A \subseteq \mathbb{N}$ is *recursively enumerable* (*r. e.*) if there is an algorithm that for every input $x \in \mathbb{N}$ halts if and only if $x \in A$. We say that such algorithm recognizes $x \in A$. Equivalently, A is recursively enumerable if there is an algorithm that generates (i.e. *enumerates*) all elements of A.

Observation *For every $A \subseteq \mathbb{N}$ it holds that A is recursive $\Leftrightarrow$ A, $\overline{A}$ are r. e.*

Decidable theories

Decidable theories

Is the truth in a given theory algorithmically decidable?

We (always) assume that the language L is recursive. A theory T of L is *decidable* if $Thm(T)$ is recursive; otherwise, T is *undecidable*.

Proposition *For every theory T of L with recursively enumerable axioms,*

i $Thm(T)$ is recursively enumerable,

ii if T is complete, then $Thm(T)$ is recursive, i.e. T is decidable.

Proof The construction of systematic tableau from T with a root $F\varphi$ assumes a given enumeration of axioms of T. Since T has recursively enumerable axioms, the construction provides an algorithm that recognizes $T \vdash \varphi$.

If T is complete, then $T \nvdash \varphi$ if and only if $T \vdash \neg\varphi$ for every sentence φ. Hence, the parallel construction of systematic tableaux from T with roots $F\varphi$ resp. $T\varphi$ provides an algorithm that decides $T \vdash \varphi$. $\square$

Recursively enumerable complete extensions

What happens if we are able to describe all simple complete extensions?

We say that the set of all (up to equivalence) simple complete extensions of a theory T is *recursively enumerable* if there exists an algorithm $\alpha(i, j)$ that generates i-th axiom of j-th extension (in some enumeration) or announces that it (such an axiom or an extension) does not exist.

Proposition *If a theory T has recursively enumerable axioms and the set of all (up to equivalence) simple complete extensions of T is recursively enumerable, then T is decidable.*

Proof By the previous proposition there is an algorithm to recognize $T \vdash \varphi$. On the other hand, if $T \not\vdash \varphi$ then $T' \vdash \neg\varphi$ is some simple complete extension T' of T. This can be recognized by parallel construction of systematic tableaux with root $T\varphi$ from all extensions. In the i-th step we construct tableaux up to i levels for the first i extensions. □

Examples of decidable theories

The following theories are decidable although not complete.

- the theory of pure equality; with no axioms, in $L = \langle\rangle$ with equality,
- the theory of unary predicate; with no axioms, in $L = \langle U \rangle$ with equality, where U is a unary relation symbol,
- the theory of dense linear orders $DeLO^*$,
- the theory of algebraically closed fields in $L = \langle +, -, \cdot, 0, 1 \rangle$ with equality, with the axioms of fields, and the axioms (for all $n \geq 1$)
$$(\forall x_{n-1}) \dots (\forall x_0)(\exists y)(y^n + x_{n-1} \cdot y^{n-1} + \dots + x_1 \cdot y + x_0 = 0),$$
where y^k is a shortcut for the term $y \cdot y \cdot \;\cdots\; \cdot y$ ($\cdot$ applied $(k-1)$-times).
- the theory of Abelian groups,
- the theory of Boolean algebras.

Recursive axiomatizability

Recursive axiomatizability

Can we "effectively" describe common mathematical structures?

- A class $K \subseteq M(L)$ is *recursively axiomatizable* if there exists a recursive theory T of language L with $M(T) = K$.
- A theory T is recursively axiomatizable if $M(T)$ is recursively axiomatizable, i.e. there is an equivalent recursive theory.

Proposition *For every finite structure $\mathcal{A}$ of a finite language with equality the theory* $\mathrm{Th}(\mathcal{A})$ *is recursively axiomatizable. Thus,* $\mathrm{Th}(\mathcal{A})$ *is decidable.*

Proof Let $A = \{a_1, \ldots, a_n\}$. $\mathrm{Th}(\mathcal{A})$ can be axiomatized by a single sentence (thus recursively) that describes $\mathcal{A}$. It is of the form *"there are exactly n elements $a_1, \ldots, a_n$ satisfying exactly those atomic formulas on function values and relations that are valid in the structure $\mathcal{A}$."* □

Recursive axiomatizations

Examples of recursive axiomatizability

The following structures $\mathcal{A}$ have recursively axiomatizable $\mathrm{Th}(\mathcal{A})$.

- $\langle \mathbb{Z}, \leq \rangle$, by the theory of discrete linear orderings,
- $\langle \mathbb{Q}, \leq \rangle$, by the theory of dense linear orderings without ends (*DeLO*),
- $\langle \mathbb{N}, S, 0 \rangle$, by the theory of successor with zero,
- $\langle \mathbb{N}, S, +, 0 \rangle$, by so called Presburger arithmetic,
- $\langle \mathbb{R}, +, -, \cdot, 0, 1 \rangle$, by the theory of real closed fields,
- $\langle \mathbb{C}, +, -, \cdot, 0, 1 \rangle$, by the theory of algebraically closed fields with characteristic 0.

Corollary *For all the above structures $\mathcal{A}$ the theory* $\mathrm{Th}(\mathcal{A})$ *is decidable.*

Remark However, $\underline{\mathbb{N}} = \langle \mathbb{N}, S, +, \cdot, 0, \leq \rangle$ is not recursively axiomatizable.
(This follows from the Gödel's incompleteness theorem).

Theories of arithmetic

Robinson arithmetic

How to effectively and "almost" completely axiomatize $\underline{\mathbb{N}} = \langle \mathbb{N}, S, +, \cdot, 0, \leq \rangle$?

The language of arithmetic is $L = \langle S, +, \cdot, 0, \leq \rangle$ with equality.

Robinson arithmetic Q has axioms (finitely many)

$$\begin{array}{ll} S(x) \neq 0 & x \cdot 0 = 0 \\ S(x) = S(y) \rightarrow x = y & x \cdot S(y) = x \cdot y + x \\ x + 0 = x & x \neq 0 \rightarrow (\exists y)(x = S(y)) \\ x + S(y) = S(x + y) & x \leq y \leftrightarrow (\exists z)(z + x = y) \end{array}$$

Remark Q is quite weak; for example, it does not prove commutativity or associativity of $+$, $\cdot$, or transitivity of $\leq$. However, it suffices to prove, for example, existential sentences on numerals that are true in $\underline{\mathbb{N}}$.

For example, for $\varphi(x,y)$ in the form $(\exists z)(x+z=y)$ it is

$$Q \vdash \varphi(\underline{1},\underline{2}), \quad \textit{where } \underline{1}=S(0) \textit{ and } \underline{2}=S(S(0)).$$

Peano arithmetic

Peano arithmetic PA has axioms of

a Robinson arithmetic Q,

b scheme of induction; that is, for every formula $\varphi(x,\overline{y})$ of L the axiom

$$(\varphi(0,\overline{y}) \wedge (\forall x)(\varphi(x,\overline{y}) \to \varphi(S(x),\overline{y}))) \to (\forall x)\varphi(x,\overline{y}).$$

Remark PA is quite successful approximation of Th($\underline{\mathbb{N}}$)*, it proves all "elementary" properties that are true in $\underline{\mathbb{N}}$ (e.g. commutativity of $+$). But it is still incomplete, there are sentences that are true in $\underline{\mathbb{N}}$ but independent in PA.*

Remark In the second-order language we can completely axiomatize $\underline{\mathbb{N}}$ (up to isomorphism) by taking directly the following (second-order) axiom of induction instead of scheme of induction

$$(\forall X)\ ((X(0) \wedge (\forall x)(X(x) \to X(S(x)))) \to (\forall x)\ X(x)).$$

Undecidability of predicate logic

Hilbert's 10th problem

- Let $p(x_1,\dots,x_n)$ be a polynomial with integer coefficients. Does the

  Diophantine equation $p(x_1,\dots,x_n)=0$ have a solution in integers?

- Hilbert (1900) *"Find an algorithm that determines in finitely many steps whether a given Diophantine equation in an arbitrary number of variables and with integer coefficient has an integer solution."*

Remark Equivalently, one may ask for an algorithm to determine whether there is a solution in natural numbers.

Theorem (DPRM, 1970) *The problem of existence of integer solution to a given Diophantine equation with integer coefficients is alg. undecidable.*

Corollary *There is no algorithm to determine for given polynomials $p(x_1,\dots,x_n)$, $q(x_1,\dots,x_n)$ with natural coefficients whether*

$$\underline{\mathbb{N}} \models (\exists x_1)\dots(\exists x_n)(p(x_1,\dots,x_n)=q(x_1,\dots,x_n)).$$

Undecidability of predicate logic

Is there an algorithm to decide if a given sentence is (logically) true?

- We know that Robinson arithmetic Q has finitely many axioms, model $\underline{\mathbb{N}}$, and proves existential sentences on numerals that are true in $\underline{\mathbb{N}}$.
- Precisely, for every existential formula $\varphi(x_1,\ldots,x_n)$ in arithmetic,
$$Q \vdash \varphi(x_1/\underline{a_1},\ldots,x_n/\underline{a_n}) \quad\Leftrightarrow\quad \underline{\mathbb{N}} \models \varphi[e(x_1/a_1,\ldots,x_n/a_n)]$$
for every $a_1,\ldots,a_n \in \mathbb{N}$ where $\underline{a_i}$ denotes the a_i-th numeral.
- In particular, for φ of the form $(\exists x_1)\ldots(\exists x_n)(p(x_1,\ldots,x_n) = q(x_1,\ldots,x_n))$, where p, q are polynomials with natural coefficients (numerals) we have
$$\underline{\mathbb{N}} \models \varphi \quad\Leftrightarrow\quad Q \vdash \varphi \quad\Leftrightarrow\quad \vdash \psi \rightarrow \varphi \quad\Leftrightarrow\quad \models \psi \rightarrow \varphi,$$
where ψ is the conjunction of (closures) of all axioms of Q.
- Thus, if there were an algorithm deciding logical truth of sentences, there would be also an algorithm deciding $\underline{\mathbb{N}} \models \varphi$, which is impossible.

1.11 Incompleteness

Introduction

Gödel's incompleteness theorems

Theorem (1st) *For every consistent recursively axiomatized extension T of Robinson arithmetic there is a sentence true in $\underline{\mathbb{N}}$ and unprovable in T.*

Remarks

- *"Recursively axiomatized" means that T is* "effectively given"*.*
- *"Extension of R. arithmetic" means that T is* "sufficiently strong"*.*
- *If, moreover, $\underline{\mathbb{N}} \models T$, the theory T is incomplete.*
- *The sentence constructed in the proof says* "I am not provable in T"*.*
- *The proof is based on two principles:*

  (*a*) *arithmetization of syntax,*

  (*b*) *self-reference.*

Arithmetization

Arithmetization - provability predicate

- Finite objects of syntax (symbols of language, terms, formulas, finite tableaux, proofs) can be (effectively) encoded by natural numbers.
- Let $\lceil \varphi \rceil$ denote the code of formula φ and let $\underline{\varphi}$ denote the numeral (a term of arithmetic) representing $\lceil \varphi \rceil$.
- If T has recursive axiomatization, the relation $\mathrm{Prf}_T \subseteq \mathbb{N}^2$ is recursive.
$$\mathrm{Prf}_T(x, y) \quad \Leftrightarrow \quad \textit{a (tableau) } y \textit{ is a proof of (a sentence) } x \textit{ in } T.$$
- If, moreover, T extends Robinson arithmetic Q, the relation Prf_T can be represented by some formula $Prf_T(x, y)$ s.t. for every $x, y \in \mathbb{N}$
$$\begin{aligned} &Q \vdash Prf_T(\underline{x}, \underline{y}), && \textit{if} \quad \mathrm{Prf}_T(x, y), \\ &Q \vdash \neg Prf_T(\underline{x}, \underline{y}), && \textit{otherwise.} \end{aligned}$$
- $Prf_T(x, y)$ expresses that *"y is a proof of x in T"*.
- $(\exists y) Prf_T(x, y)$ expresses that *"x is provable in T"*.
- If $T \vdash \varphi$, then $\underline{\mathbb{N}} \models (\exists y) Prf_T(\underline{\varphi}, y)$ and moreover $T \vdash (\exists y) Prf_T(\underline{\varphi}, y)$.

Self-reference

Self-reference principle

- *This sentence has 24 letters.*

  In formal systems self-reference is not always available straightforwardly.

- *The following sentence has 32 letters "The following sentence has 32 letters".*

  Such direct reference is available, if we can "talk" about sequences of symbols. But the above sentence is not self-referencial.

- *The following sentence written once more and then once again between quotation marks has 116 letters "The following sentence written once more and then once again between quotation marks has 116 letters".*

  With use of direct reference we can have self-reference. Instead of *"it has x letters"* we can have other properties.

Fixed-point theorem

Theorem *Let T be consistent extension of Robinson arithmetic. For every formula $\varphi(x)$ in language of theory T there is a sentence ψ s.t. $T \vdash \psi \leftrightarrow \varphi(\underline{\psi})$.*

Remark *ψ is self-referencial, it says* "This formula satisfies condition φ".

Proof (idea) Consider the *doubling* function d: for every formula $\chi(x)$

$$d(\ulcorner\chi(x)\urcorner) = \ulcorner\chi(\underline{\chi(x)})\urcorner$$

- It can be shown that d is expressible in T. Assume *(for simplicity)* that it is expressible by some term, denoted also by d.
- Then for every formula $\chi(x)$ in language of theory T it holds that
$$T \vdash d(\underline{\chi(x)}) = \underline{\chi(\underline{\chi(x)})} \tag{1.1}$$
- We take $\varphi(d(\underline{\varphi(d(x))}))$ for ψ. If suffices to verify that $T \vdash d(\underline{\varphi(d(x))}) = \underline{\psi}$.
- This follows from (1.1) for $\chi(x)$ being $\varphi(d(x))$, since in this case
$$T \vdash d(\underline{\varphi(d(x))}) = \underline{\varphi(d(\underline{\varphi(d(x))}))} \quad \square$$

Undefinability of truth

Undefinability of truth

We say that a formula $\tau(x)$ *defines truth* in theory T of arithmetical language if for every sentence φ it holds that $T \vdash \varphi \leftrightarrow \tau(\underline{\varphi})$.

Theorem *Let T be consistent extension of Robinson arithmetic. Then T has no definition of truth.*

Proof By the fixed-point theorem for $\neg\tau(x)$ there is a sentence φ such that

$$T \vdash \varphi \leftrightarrow \neg\tau(\underline{\varphi}).$$

Supposing that $\tau(x)$ defines truth in T, we would have

$$T \vdash \varphi \leftrightarrow \neg\varphi,$$

which is impossible in a consistent theory T. $\square$

Remark This is based on the liar paradox, the sentence φ would express
"This sentence is not true in T".

First incompleteness theorem

Proof of the first incompleteness theorem

Theorem (Gödel) *For any consistent recursively axiomatized extension T of Robinson arithmetic there is a sentence true in $\underline{\mathbb{N}}$ and unprovable in T.*

Proof Let $\varphi(x)$ be $\neg(\exists y)Prf_T(x,y)$, it says *"$x$ is not provable in T"*.

- By the fixed-point theorem for $\varphi(x)$ there is a sentence ψ_T such that
$$T \vdash \psi_T \leftrightarrow \neg(\exists y)Prf_T(\underline{\psi_T}, y). \tag{1.2}$$
ψ_T says *"I am not provable in T"*. More precisely, ψ_T is equivalent to a sentence expressing that ψ_T is not provable T (where the equivalence holds both in $\underline{\mathbb{N}}$ and in T).

- First, we show ψ_T *is not provable in T*. If $T \vdash \psi_T$, i.e. ψ_T is contradictory in $\underline{\mathbb{N}}$, then $\underline{\mathbb{N}} \models (\exists y)Prf_T(\underline{\psi_T}, y)$ and moreover $T \vdash (\exists y)Prf_T(\underline{\psi_T}, y)$. Thus from (1.2) it follows $T \vdash \neg\psi_T$, which is impossible since T is consistent.

- It remains to show ψ_T is true in $\underline{\mathbb{N}}$. If not, i.e. $\underline{\mathbb{N}} \models \neg\psi_T$, then $\underline{\mathbb{N}} \models (\exists y)Prf_T(\underline{\psi_T}, y)$. Hence $T \vdash \psi_T$, which we already disproved. □

Corollaries and a strengthened version

Corollary *If, moreover, $\underline{\mathbb{N}} \models T$, then the theory T is incomplete.*

Proof Suppose T is complete. Then $T \vdash \neg\psi_T$ and thus $\underline{\mathbb{N}} \models \neg\psi_T$, which contradicts $\underline{\mathbb{N}} \models \psi_T$. □

Corollary $\mathrm{Th}(\underline{\mathbb{N}})$ *is not recursively axiomatizable.*

Proof $\mathrm{Th}(\underline{\mathbb{N}})$ is consistent extension of Robinson arithmetic and has a model $\underline{\mathbb{N}}$. Suppose $\mathrm{Th}(\underline{\mathbb{N}})$ is recursively axiomatizable. Then by previous corollary, $\mathrm{Th}(\underline{\mathbb{N}})$ is incomplete, but $\mathrm{Th}(\underline{\mathbb{N}})$ is clearly complete. □

Gödel's first incompleteness theorem can be strengthened as follows.

Theorem (Rosser) *Every consistent recursively axiomatized extension T of Robinson arithmetic has an independent sentence. Thus T is incomplete.*

Remark Hence the assumption in the first corollary that $\underline{\mathbb{N}} \models T$ is superfluous.

Second incompleteness theorem

Gödel's second incompleteness theorem

Let Con_T denote the sentence $\neg(\exists y)Prf_T(\underline{0 = \underline{1}}, y)$. We have that $\underline{\mathbb{N}} \models Con_T \Leftrightarrow T \nvdash 0 = \underline{1}$. Thus Con_T expresses that *"T is consistent"*.

Theorem (Gödel) *For every consistent recursively axiomatized extension T of Peano arithmetic it holds that Con_T is unprovable in T.*

Proof (idea) Let ψ_T be the Gödel's sentence *"This is not provable in T"*.

- In the first part of the proof of the 1st theorem we showed that

  $$\text{“\textit{If } } T \text{ \textit{is consistent, then} } \psi_T \text{ \textit{is not provable in} } T.\text{”} \tag{1.3}$$

  In other words, we showed it holds $Con_T \rightarrow \psi_T$.
- If T is an extension of Peano arithmetic, the proof of (1.3) can be formalized within the theory T itself. Hence $T \vdash Con_T \rightarrow \psi_T$.
- Since T is consistent by the assumption, from (1.3) we have $T \nvdash \psi_T$.
- Therefore from the previous two bullets, it follows that $T \nvdash Con_T$. □

Remark Hence such a theory T cannot prove its own consistency.

Corollaries of the second theorem

Corollary *Peano arithmetic has a model $\mathcal{A}$ s.t. $\mathcal{A} \models (\exists y) Prf_{PA}(\underline{0 = 1}, y)$.*

Remark $\mathcal{A}$ has to be nonstandard model of PA, the witness must be some nonstandard element (other than a value of a numeral).

Corollary *There is a consistent recursively axiomatized extension T of Peano arithmetic such that $T \vdash \neg Con_T$.*

Proof Let $T = PA \cup \{\neg Con_{PA}\}$. Then T is consistent since $PA \nvdash Con_{PA}$. Moreover, $T \vdash \neg Con_{PA}$, i.e. T proves inconsistency of $PA \subseteq T$, and thus also $T \vdash \neg Con_T$. □

Remark $\underline{\mathbb{N}}$ cannot be a model of T.

Corollary *If the set theory ZFC is consistent, then Con_{ZFC} is unprovable in ZFC.*