Katedra Teoretické informatiky

MFF UK

Meze formální metody [1]

učební text

Petr Štěpánek

7. července 2000

[1] meze.dvi meze.ps http://kocour.ms.mff.cuni.cz/people/stepanek.cz.html

Úvod

Zatím jsme používali jako zřejmý fakt to, že odvozovací pravidla jsou mechanické manipulace, které transformují formule, tedy řetězce symbolů. V dalším výkladu budeme syntaktickému aspektu odvozování věnovat větší pozornost. Začneme jednoduchým příkladem jednoduché teorie.

Příklad Teorie grup je teorie s jazykem $\{e, \cdot\}$ a třemi axiomy

$$x \cdot (y \cdot z) = (x \cdot y) \cdot z \qquad e \cdot x = e = x \cdot e \qquad \forall x \exists y \, (y \cdot x = e)$$

Kdybychom postupně sestrojovali všechny důkazy z této množiny axiomů, a k nim posloupnost dokázaných formulí, taková posloupnost by obsahovala všechny věty teorie grup. Mohli bychom postupovat například tak, že bychom nejprve sestrojili všechny důkazy délky jedna v nějakém pořadí, potom všechny důkazy, které jsou posloupností dvou formulí atd. Poslední formuli každého důkazu bychom zařadili na konec dosud sestrojené posloupnosti vět. Takové metodě sestrojování se říká vyčerpávající (exhaustivní). Jejím výsledkem by patrně byla redundantní posloupnost formulí, která by obsahovala řadu triviálních vět a ve které by se dokázané formule mohly opakovat. Při důsledném dodržení exhaustivního postupu by však bylo zaručeno, že každá věta teorie grup by byla v některém kroku sestrojena.

Takový postup by mohl jen těžko zaujmout algebraika, který pracuje v teorii grup. Jeho zajímá zda nějaká zcela určitá formule je či není větou teorie grup. Jako odpověď na svou otázku čeká důkaz dané formule nebo protipříklad. Protože exhaustivní metoda dává v každém kroku jen konečně mnoho vět teorie grup, pomůže jen v tom případě, že daná formule již byla dokázána. V případě, že daná formule není větou nemůžeme to exhaustivním postupem v konečném čase zjistit.

1 Rozhodnutelnost a enumerovatelnost

Náš příklad ukázal dva důležité pojmy, které jsou motivovány dvěma odlišnými úkoly. Je-li T nějaká teorie, jde o to

- efektivním postupem generovat všechny věty teorie T
- pro libovolnou formuli efektivně rozhodnout zda je či není větou T

kde efektivnost znamená, že existuje algoritmická procedura, která úkol řeší.

1.1 Definice Nechť $\mathcal{F}$ je množina formulí nějakého jazyka L, nechť T je teorie s jazykem L.

(i) Říkáme, že $\mathcal{F}$ *je enumerovatelná*, existuje-li algoritmická procedura, která generuje všechny prvky množiny $\mathcal{F}$.

(ii) Říkáme, že teorie T *je rozhodnutelná*, jestliže existuje algoritmus, který pro libovolnou formuli A jazyka L dovoluje rozhodnout zda A je či není větou teorie T. V opačném případě říkáme, že T *je nerozhodnutelná.* □

1.2 Ukázali jsme, že enumerovatelnost množiny všech vět nemusí zaručovat rozhodnutelnost teorie. Naopak není těžké ukázat, že rozhodnutelnost teorie zaručuje enumerovatelnost množiny všech jejích vět. Je-li T teorie s jazykem L a je-li $\mathcal{P}$ algoritmická procedura, která rozhoduje of větách teorie T, stačí generovat všechny formule jazyka L podle nějakého uspořádání a do enumerující posloupnosti zařadit jen ty formule, které procedura $\mathcal{P}$ označí jako věty.

Tak jako v předchozích odstavcích bude pojem algoritmické procedury hrát svou roli i v dalším výkladu spolu s dalšími pojmy *teorie vyčíslitelnosti*, které se také říká *teorie rekursivních funkcí* nebo *teorie rekurse*.

2 Rekursivní funkce

Rekursivní funkce jsou specifickým vyjádřením pojmu algoritmické procedury. V dalším výkladu použijeme třídy částečných rekursivních funkcí na množině přirozených čísel. Máme pro to dva důvody: je to nejčastěji používaná formalizace vyčíslitelnosti a třída částečných rekursivních funkcí má bezprostřední vztah k jazyku aritmetiky.

Je však známo, že třídu částečných rekursivních funkcí lze definovat na množině slov každé konečné abecedy, popřípadě i na dalších oborech.

2.1 Totální a částečné funkce v oboru přirozených čísel V dalším budeme množinu přirozených čísel označovat písmenem N. Připomeňme, že pro každé přirozené n

$$f : N^n \to N$$

označuje funkci definovanou na množině N^n všech uspořádaných n-tic přirozených čísel, s hodnotami v množině přirozených čísel N. Takovým zobrazením říkáme *totální funkce*.

Teorie vyčíslitenosti pracuje také s částečnými funkcemi n-proměnných na množině N. Funkce f je částečná, je-li definována na nějaké podmnožině $dom(f) \subseteq N^n$ a jejím oborem hodnot je nějaká podmnožina $rng(f) \subseteq N$.

2.2 Částečné rekursivní funkce Pro třídu $\mathcal{R}$ všech částečných rekursivních funkcí je známa řada ekvivalentních definic.

$\mathcal{R}$ je třída

- funkcí vyčíslitelných Turingovým strojem
- funkcí vyčíslitelných URM-strojem
- λ-definovatelných funkcí v lambda kalkulu
- nejmenší třída, která obsahuje jisté základní funkce a je uzavřena na jisté operace

Pro účely tohoto výkladu není třeba rozvádět žádnou z těchto definic. Postačí, když zavedeme obvyklou terminologii a na vhodných místech přijmeme jako fakt některá tvrzení rekursivních funkcích.

2.3 Definice (i) Je-li funkce $f \in \mathcal{R}$ totální, říkáme, že f *je rekursivní funkce.*

(ii) Říkáme, že *množina* $A \subseteq N^n$ *je rekursivní*, je-li její charakteristická funkce rekursivní.

(iii) Říkáme, že *množina* $A \subseteq N^n$ *je rekursivně spočetná*, je-li definičním oborem nějaké částečné rekursivní funkce. □

Termín rekursivně spočetná množina má svůj původ v následujícím tvrzení:

Neprázdná množina $A \subseteq N^n$ je rekursivně spočetná právě když je oborem hodnot rekursivní funkce.

2.4 Vlastnosti třídy $\mathcal{R}$ Pro další výklad mají zásadní význam především následující tři vlastnosti třídy $\mathcal{R}$.

- **enumerovatelnost** zejména množiny všech částečných rekursivních funkcí jedné proměnné.
- **aritmetizovatelnost** syntaxe formálních systémů rekursivními funkcemi.
- **reprezentovatelnost** částečných rekursivních funkcí a rekursivních relací v aritmetice

Zatímco enumerovatelnost množiny částečných rekursivních funkcí jedné proměnné je dána Kleeneho větou o normální formě, kterou v zápětí uvedeme, aritmetizovatelností syntaxe formálního systému rozumíme fakt, že termy, formule, posloupnosti formulí a důkazy tohoto formálního systému lze kódovat přirozenými čísly pomocí rekursivních funkcí.

Representovatelnost rekursivních funkcí a rekursivních predikátů v aritmetice znamená možnost převést vhodným způsobem rovnosti tvaru

$$f(n_1, n_2, \ldots, n_k) = m \quad \text{pro} \quad n_1, n_2, \ldots, n_k, m \in N$$

na dokazatelnost určitých formulí v aritmetice. Tím bude zaručeno, že aritmetizaci, to znamená kódování formálního systému aritmetiky, lze provést v ní samé. Z toho pak plynou důležité výsledky o formálním systému aritmetiky, neúplnost, nerozhodnutelnost, nedefinovatelnost pravdy, nemožnost dokázat bezespornost aritmetiky v ní samé a další výsledky.

2.5 Enumerovatelnost $\mathcal{R}$ a problém zastavení Připomeňme konvenci, která se používá pro rovnost mezi částečnými rekursivními funkcemi.

Rovnost

$$f(n) = g(n)$$

znamená, že funkce f je definována pro číslo n právě když pro n je definována i funkce g a obě hodnoty se sobě rovnají. Podobná úmluva platí i pro funkce více proměnných a pro případ, kdy některé proměnné mohou vystupovat jako parametry, například pro rovnost $h(p,q) = g(p)$, kde číslo q hraje roli parametru.

Je-li funkce f definována pro přirozené číslo n, říkáme také, že $f(n)$ konverguje a píšeme $f(n) < \infty$. Jinak říkáme, že $f(n)$ diverguje.

Enumerovatelnost třídy $\mathcal{R}$ je důsledkem následujícího tvrzení

2.6 Věta o normální formě (Kleene, Turing) Pro každé přirozené číslo $k \geq 1$ existuje částečná rekursivní funkce Φ^{k+1} (k+1) proměnných taková, že

(i)definujeme-li pro každé přirozené n a přirozená čísla $n_1, \ldots, n_k$

$$\phi_n(n_1, \ldots, n_k) = \Phi^{k+1}(n, n_1, \ldots, n_k)$$

pak ϕ_n je částečná rekursivní funkce.

(ii) každá částečná rekursivní funkce k proměnných je rovna funkci ϕ_n pro některé přirozené číslo n. □

2.7 Říkáme, že funkce Φ^{k+1} je universální pro třídu všech částečných rekursivních funkcí k proměnných a že přirozené číslo n je indexem funkce ϕ_n.

Z věty o normální formě pro každé $k \geq 1$ bezprostředně vyplývá enumerovatelnost množiny všech částečných rekursivních funkcí k proměnných, speciálně množiny všech částečných rekursivních funkcí jedné proměnné.

2.8 Problém zastavení Enumerace množiny všech částečných rekursivních funkcí jedné proměnné dovoluje definovat jistou množinu přirozených čísel, která není rekursivní a na kterou lze redukovat řadu problémů rozhodnutelnosti.

2.9 Věta Množina

$$K \;=\; \{n|\; \phi_n(n) < \infty\}$$

není rekursivní. □

Důkaz věty se provádí diagonalizací. Kdyby K byla rekursivní množina, potom funkce g definovaná předpisem

$$g(n) \;=\; \begin{cases} \phi_n(n)+1 & \text{je-li } n \in K \\ 0 & \text{jinak} \end{cases}$$

by byla (totální) rekursivní funkce, která by byla různá ode všech funkcí ϕ_n, $n \in N$. To by bylo ve sporu s větou o normální formě, množina K tedy není rekursivní. □

2.10 Zbývá ještě vysvětlit jaký má množina K vztah k zastavení. Turing sestrojil rekursivní predikát $T(n,m,p)$, který charakterisuje výpočty universálního Turingova stroje. Pro libovolná přirozená čísla n,m je hodnota $\phi_n(m)$ je definována právě když existuje přirozené číslo p takové že platí $T(n,m,p)$. Číslo p koduje protokol výpočtu a hodnotu $\phi_n(m)$ z něj lze vypočítat. Pokud takové p neexistuje ("Turingův stroj se nezastaví"), hodnota $\phi_n(m)$ není definována.

Proto je možno množinu K ekvivalentně vyjádřit následujícím způsobem

$$K \;=\; \{n \,|\, \exists\; p\; T(n,n,p)\}$$

a K sestává ze všech čísel n, pro která se výpočet hodnoty $\phi_n(n)$ universálním Turingovým strojem zastaví.

3 Nerozhodnutelnost predikátové logiky

Nyní ukážeme, že problém rozhodnutelnosti predikátové logiky I. řádu lze redukovat na nerozhodnutelnost problému zastavení.

3.1 Věta (Church) Nechť L je spočetný jazyk prvního řádu, který obsahuje dostatečně mnoho speciálních symbolů. Jinými slovy, předpokládáme, že

- L obsahuje alespoň jednu konstantu a alespoň jeden funkční symbol četnosti $n \geq 1$.
- pro každé přirozené číslo n jazyk L obsahuje spočetně mnoho predikátových symbolů četnosti n a ne více.

Potom množina

$$\{A \,|\, A \text{ je uzavřená formule jazyka } L \text{ a } \models A\} \qquad (1)$$

všech logicky pravdivých sentencí jazyka L není rozhodnutelná. V takovém případě říkáme, že predikátová logika s jazykem L je nerozhodnutelná. □

Náznak důkazu. Ke každému přirozenému číslu n přiřadíme uzavřenou formuli A_n takovou, že

$$\models A_n \text{ právě když } n \in K \qquad (2)$$

Potom každá algoritmická procedura, která by rozhodovala o prvcích množiny (1) by rozhodovala i o prvcích množiny K, která podle Věty 2.9 není rekusivní. Množina (1) tedy není rozhodnutelná.

Zbývá sestrojit uzavřené formule A_n, pro které by platilo (1). Množina K byla sestrojena diagonalizací enumerace všech částečných rekursivních funkcí jedné proměnné. Z věty o normální formě víme, že enumerace je dána jedinou částečnou rekursivní funkcí Φ^2 dvou proměnných.

K sestrojení formulí A_n využijeme logických programů pro rekursivní funkce (viz [7]) v daném jazyce L. Numerály

$$\overline{0}, \overline{1}, \overline{2}, \ldots, \overline{n}, \overline{n+1}, \ldots$$

kódujeme termy

$$c, f(c), f(f(c)), \ldots, f^n(c), f^{n+1}(c), \ldots$$

pokud c je nějaká konstanta a f je nějaký unární funkční symbol jazyka L. Pokud L neobsahuje unární funkční symbol, kodujeme numerály například pomocí termů

$$c, f(c, \ldots, c), f(c, \ldots, c, f(c, \ldots, c,)), \ldots$$

V jazyce L je možné sestrojit logický program P, takový, že pro jistý predikát p_Φ a přirozená čísla n, m, r platí

$$\Phi^2(n,m) = r \text{ právě když existuje SLD-zamítnutí pro } P \cup \{\leftarrow p_\Phi(\overline{n}, \overline{m}, \overline{r})\} \quad (3)$$

Platí-li (3), říkáme, že program P počítá funkci Φ^2. Nyní pro libovolné n dostáváme

$$\begin{array}{ll} n \epsilon K & \text{právě když } \exists r\ (\Phi^2(n,n) = r) \\ & \text{právě když } existuje\ SLD\text{-}zamítnutí\ pro\ P \cup \{\leftarrow p_\Phi(\overline{n}, \overline{n}, r)\} \\ & \text{právě když } P \models \exists r\ p_\Phi(\overline{n}, \overline{n}, r) \\ & \text{právě když } \models P \rightarrow \exists r\ p_\Phi(\overline{n}, \overline{n}, r) \end{array}$$

kde v poslední implikaci chápeme P jako konjunkci všech klauzulí programu P. Označíme-li nyní tuto implikaci A_n, ukázali jsme, že platí (2). Věta 3.1 je dokázána. □

3.2 Podrobnějším rozborem konstrukce programu P z důkazu předchozí věty bychom zjistili, jaké požadavky na jazyk L jsou pro tento důkaz nerozhodnutelnosti postačující. Při konstrukci logického programu pro výpočet Turingova predikátu jsme použili jen konečně mnoho predikátových symbolů. Konstrukce programu P nepoužívá predikát rovnosti.

Další podstatnou redukcí je možné získat například tento výsledek. □

3.3 Věta Je-li L jazyk prvního řádu bez rovnosti, který obsahuje alespoň dva binární predikáty, potom predikátová logika s tímto jazykem je nerozhodnutelná. □

4 Aritmetizovatelnost formálních systémů

Rekursivní funkce dovolují efektivně kódovat syntaktické výrazy, termy, formule a posloupnosti formulí libovolného jazyka prvního řádu L pomocí přirozených čísel. Pokud je T teorie s jazykem L a množina (kódů) axiomů teorie je rekursivní, je možné efektivně kódovat také důkazy prováděné v teorii T.

Kódování řetězců znaků má dvě stránky. Je třeba dát předpis, kterým se kódy generují a potom je třeba ověřit, že takový předpis popisuje vyčíslitelnou, tedy rekursivní funkci. Omezíme se zde jen na první stránku problému. Ověření, že kódujeme pomocí rekursivních funkcí ponecháme na čtenáři, který je seznámen se základy teorie rekurse. Ostatní čtenáři mohou tento fakt přijmout bez důkazu.

4.1 Dělení se zbytkem definuje dvě binární (primitivně) rekursivní funkce q a r definované následovně

$$q(x,y) = minz \leq x[(y(z+1) > x)]$$
$$r(x,y) = x \sim (y \cdot q(x,y))$$

kde $u \sim v = u - v$ pokud $u \geq v$ a jinak $u \sim v = 0$. Je-li $b \neq 0$ potom $q(a,b)$ je celočíselný podíl a $r(a,b)$ je zbytek při dělelní čísla a číslem b. □

4.2 Množina prvočísel se dá podle velikosti uspořádat do posloupnosti

$$p_0, p_1, p_2, \ldots$$

kde $p_0 = 2, p_1 = 3, p_2 = 5, \ldots$ a p_n je n-té prvočíslo. Podle Eukleidova důkazu existence nekonečně mnoha prvočísel je $p_{i+1} \leq p_i! + 1$ proto můžeme definovat posloupnost (funkci) p_i takto

$$p_0 = 2$$
$$p_{i+1} = min\ y \leq (p_i! + 1)\{(y > p_i)\ \&\ \forall z < y[(z \leq 1) \vee (r(y,z) > 0)]\}$$

Dá se ukázat, že p_i je (primitivně) rekursivní funkce proměnné i. Položíme-li

$$(z)_x = min\ y < z(r(z,(p_x)^{y+1}) > 0)$$

pak pro $z > 0$ je $(z)_x$ největší y takové, že $(p_x)^y$ je dělitelem čísla z. Platí také $(0)_x = 0$, ale nebudeme to potřebovat. Říkáme, že $(z)_x$ *je x-tý exponent v prvočíselném rozvoji čísla z.* □

4.3 Kódování konečných posloupností čísel Chceme vzájemně jednoznačně kódovat konečné posloupnosti

$$(n_0, n_1 \ldots, n_k)$$

pomocí přirozených čísel.

K tomu můžeme použít posloupnosti prvočísel

$$2, 3, 5, 7, 11, 13, \ldots$$

kterou enumeruje jistá rekursivní funkce p. Pro libovolné i budeme i-té prvočíslo označovat p_i místo $p(i)$. Nyní můžeme definovat

(i) $\langle n_0, n_1 \ldots, n_k\rangle = p_0^{n_0+1} \cdot p_1^{n_1+1} \cdot \ldots \cdot p_k^{n_k+1}$ pro $k \geq 0$

$\langle\rangle = 1$

(ii) $\pi_i(n) = (n)_i - 1$

kde $(n)_i$ je i-tý exponent čísla n

(iii) $Seq(a) \leftrightarrow a = \langle b_0, \ldots, b_n\rangle$ pro nějaké $\vec{b}$ □

4.4 Lemma

(i) $\langle n_0, n_1, \ldots, n_k\rangle = \langle m_0, m_1, \ldots, m_l\rangle$ právě když

$k = l$ a $n_i = m_i$ pro každé $i \leq k$

(ii) $lh(\langle n_0, \ldots, n_k\rangle) = k + 1$

$lh(\langle\rangle) = 0$

kde $lh(n) = k+1$, jestliže k je index u největšího prvočísla s nenulovým exponentem

čísla n.

$lh(n) = 0$, jestliže $n = 0$ nebo $n = 1$

(iii) $\pi_i(\langle n_0, \ldots, n_k\rangle) = n_i$ pro $i \leq k$

(iv) Pro každé k je $\langle n_0, \ldots, n_k\rangle$ je rekursivní funkce

(v) $\pi_i(n)$ rekursivní v proměnných i, n

(vi) $Seq(n)$ je rekursivní predikát □

Důkaz (vi) $Seq(n) \leftrightarrow \forall x < lh(n)((n)_x \neq 0 \vee n = 1)$ □

4.5 Věta Existuje rekursivní funkce $\star$ taková, že

$$\langle n_0, \ldots, n_k \rangle \star \langle m_0, \ldots, m_l \rangle = \langle n_0, \ldots, n_k, m_0, \ldots, m_l \rangle$$

□

Důkaz Stačí položit

$$a \star b = a \cdot \prod_{i=0}^{i=lh(b)-1} p_{lh(a)+i}^{(b)_i}$$

Potom dokazovaná rovnost platí a také $a \star \langle\rangle = a = \langle\rangle \star a$ □

4.6 Jazyk aritmetiky (i) Jazyk aritmetiky I. řádu má tyto symboly

$x_0, x_1, x_2, \ldots$	proměnné
$\neg, \rightarrow, \forall, =$	logické symboly
$(,)$	závorky
$0, S, +, \cdot$	funkční symboly

(ii) Třída termů $\mathcal{T}$ je definována takto

$$0 \in \mathcal{T} \qquad x_i \in \mathcal{T} \tag{1}$$

Je-li $t_1, t_2 \in \mathcal{T}$ potom také

$$\begin{aligned} &St_1 \in \mathcal{T} \\ &(t_1 + t_2) \in \mathcal{T} \\ &(t_1 \cdot t_2) \in \mathcal{T} \end{aligned} \tag{2}$$

(iii) Třída formulí $\mathcal{F}$ je definována takto

$$\begin{aligned} &\text{je-li } t_1, t_2 \in \mathcal{T}, \text{ potom} \quad t_1 = t_2 \in \mathcal{F} \\ &\text{je-li } t_1, t_2 \in \mathcal{T}, \text{ potom} \quad t_1 \leq t_2 \in \mathcal{F} \end{aligned} \tag{3}$$

$$\begin{aligned} &\text{je-li } A, B \in \mathcal{F}, \text{ potom} \quad \neg A \in \mathcal{F} \\ &\qquad\qquad\qquad\qquad\quad (A \rightarrow B) \in \mathcal{F} \\ &\qquad\qquad\qquad\qquad\quad \forall x_i \, A \in \mathcal{F} \end{aligned} \tag{4}$$

□

Připomeňme, že výrazy

$$A \vee B, \quad A \,\&\, B, \quad A \leftrightarrow B \quad \text{a} \quad \exists x\, A$$

jsou zkratky za formule

$$\neg A \to B, \quad \neg(A \to \neg B), \quad ((A \to B)\,\&\,(B \to A)) \quad \text{a} \quad \neg\forall x\, \neg A$$

Nyní se budeme věnovat teoriím, jejichž jazyk je totožný s jazykem aritmetiky

$$L \;=\; \{0, S, +, \cdot, \leq\}$$

nebo je jeho rozšířením.

Pro kódování syntaktických objektů jazyků prvního řádu se používá různých metod. Většina z nich používá relace dělitelnosti na množině přirozených čísel a jednoznačnost kódů je zaručena jednoznačností prvočíselných rozvojů přirozených čísel. Uvedeme zde jednu z nich.

4.7 Kódování termů a formulí Nejprve přiřadíme každému symbolu s jazyka aritmetiky určité číslo $\sigma(s)$ a potom budeme přiřazovat kódy termům a formulím.

Jazyk aritmetiky obsahuje nekonečně mnoho proměnných, těm přiřadíme po řadě sudá čísla, položíme

$$\sigma(x_i) = 2i \text{ pro každé přirozené } i.$$

Speciálním symbolům přiřadíme lichá čísla například

$$\sigma(0) = 1 \;\; \sigma(S) = 3 \;\; \sigma(+) = 5 \;\; \sigma(\cdot) = 7 \;\; \sigma(=) = 9 \;\; \sigma(\leq) = 11$$

a logickým symbolům další dosud nepoužitá lichá čísla

$$\sigma(\neg) = 13 \;\; \sigma(\to) = 15 \;\; \sigma(\forall) = 17$$

Jako kódy dalších symbolů případného rozšíření jazyka aritmetiky mohou sloužit všechna zbývající lichá čísla.

(i) Nyní každému termu t přiřadíme přirozené číslo $\sharp t$ následujícím předpisem.

Je-li t proměnná x_i pro nějaké přirozené i, potom

$$\sharp t = \langle \sigma(x_i) \rangle = \langle 2i \rangle \qquad (1)$$

Jsou-li r a s termy, přiřadíme

$$\sharp\, 0 = \langle 1 \rangle \qquad \sharp\, Sr = \langle \sigma(S), \sharp\, r \rangle \qquad (2)$$

$$\sharp\, (r + s) = \langle \sigma(+), \sharp r, \sharp s \rangle \qquad \sharp(r \cdot s) = \langle \sigma(\cdot), \sharp r, \sharp s \rangle \qquad (3)$$

Libovolné formuli A přiřadíme přirozené číslo $\sharp A$ následujícím způsobem

Jsou-li r a s termy, B a C formule, potom

$$\sharp\,(r = s) = \langle\sigma(=), \sharp r, \sharp s\rangle \qquad \sharp(r \leq s) = \langle\sigma(\leq), \sharp r, \sharp s\rangle \qquad (4)$$

$$\sharp\neg B = \langle\sigma(\neg), \sharp B, \rangle \qquad \sharp(B \to C) = \langle\sigma(\to), \sharp B, \sharp C\rangle \qquad (5)$$

$$\sharp(\forall x_i B) = \langle\sigma(\forall), \langle 2i\rangle, \sharp B\rangle \qquad \text{pro } i = 0, 1, 2, \ldots \qquad (6)$$

Označíme-li $T(x)$ charakteristickou funkci množiny kódů všech termů, potom platí

$$\begin{aligned} T(x) \leftrightarrow\ & (x = \langle 1\rangle) \vee \exists u < x(x = \langle 2u\rangle) \vee \\ & \vee \exists u < x \exists v < x[T(u)\ \&\ T(v)\ \& \\ & \qquad \&\ \{(x = \langle\sigma(S), u\rangle) \vee (x = \langle\sigma(+), u, v\rangle) \vee (x = \langle\sigma(\cdot), u, v\rangle\}] \end{aligned}$$

Dá se ukázat, že $T(x)$ je (primitivně) rekursivní predikát. Podobně se dá ukázat, že existuje rekursivní funkce $F(x)$, která je charakteristickou funkcí množiny

$\{x | x = \sharp(A)$ a A je formule jazyka aritmetiky$\}$ □

V dalším budeme používat kódy instancí formulí, potřebujeme tedy rozpoznávat z kódu formule její volné proměnné a také sestrojit kód její instance, která vznikne dosazením termu za její volné proměnné.

4.8 Proměnné, termy a formule (i) Položíme-li

$$Var(x) \leftrightarrow x = \langle(x)_0\rangle\ \&\ \exists y \leq x((x)_0 = 2.y)$$

potom Var je rekursivní predikát a $Var(x)$ platí, právě když x se rovná kódu nějaké proměnné x_y.

(ii) Definujeme funkci $sub(x, y, z)$ takovou, že pro libovolný term t, formuli A a term s platí

$$sub(\sharp t, \sharp y, \sharp s) = \sharp(t_y[s]) \qquad sub(\sharp A, \sharp y, \sharp s) = \sharp A_y[s]$$

Položíme

$$sub(x, y, z) = \begin{cases} z & \text{je-li } Var(y)\ \&\ x = y \\ \langle(x)_0, sub((x)_1, y, z)\rangle & \text{je-li } lh(x) = 2 \\ \langle(x)_0, sub((x)_1, y, z), sub((x)_2, y, z)\rangle & \text{je-li } lh(x) = 3 \\ & \qquad \&\ (x)_0 \neq \sigma(\forall) \\ \langle(x)_0, (x)_1, sub((x)_2, y, z)\rangle & \text{je-li } x = \langle\sigma(\forall), (x)_1, (x)_2\rangle \\ & \qquad \&\ (x)_1 \neq y \\ x & \text{jinak} \end{cases}$$

Z předchozí definice plyne, že $sub(x, y, z)$ je rekursivní funkce. Podobným postupem bychom mohli definovat rekursivní funkci $Sub(x, \vec{y}, \vec{z})$, která generuje kód termu nebo formule, do které byly substituovány termy s kódy $z_1, z_2, \ldots, z_k$ za proměnné s kódy $y_1, y_2, \ldots, y_k$. □

Predikát Fvar(x, y) takový, že pro libovolnou formuli A a proměnnou x platí $Fvar(\sharp A, \sharp x)$ právě když proměnná x má volný výskyt ve formuli A lze definovat takto

$$\begin{aligned} Fvar(x,y) \leftrightarrow \{&[x = y \ \& \ Var(x)] \vee [Fvar((x)_1, y) \ \& \ lh(x) = 2] \vee \\ &\vee [(lh(x) = 3 \ \& \ (x)_0 \neq \sigma(\forall)) \ \& \ (Fvar((x)_1, y) \vee Fvar((x)_2, y))] \\ &\vee [Fvar((x)_2, y) \ \& \ (x)_1 \neq y]\} \end{aligned}$$

Nyní by bylo možné definovat predikát $Sbtl(x, y, z)$ takový, že pro libovolnou formuli A, term t a proměnnou x platí $Sbtl(\sharp A, \sharp x, \sharp t)$, právě když t je term substituovatelný do A za proměnnou x. □

4.9 Kódy numerálů a instancí (i) Definujeme-li pro libovolné přirozené číslo n funkci num předpisem

$$num(n) = \sharp \bar{n} \tag{7}$$

potom

$$num(0) = \langle 1 \rangle$$
$$num(n+1) = \langle \sigma(S), num(n) \rangle$$

a z definice (7) plyne, že num je rekursivní funkce.

Podle 4.8 pro libovolné přirozené číslo n, proměnnou y, term t, a formuli A platí

$sub(\sharp t, \sharp y, num(n)) = \sharp t_y[\overline{n}] \quad sub(\sharp A, \sharp y, num(n)) = \sharp A_y[\overline{n}]$ □

5 Representovatelnost

Uvažujeme relace a funkce na množině přirozených čísel N standardního modelu aritmetiky. Je-li R k-ární relace a $n_1, n_2, \ldots, n_k$ jsou přirozená čísla, píšeme $R(n_1, n_2, \ldots, n_k)$ místo $(n_1, n_2, \ldots, n_k) \in R$. Píšeme také $A(n_1, n_2, \ldots, n_k)$ místo instance $A_{x_1, x_2, \ldots, x_k}[n_1, n_2, \ldots, n_k]$. Chceme-li zdůraznit všechny nebo jen některé volné proměnné formule A, píšeme $A(x_1, x_2, \ldots, x_k)$ místo A.

5.1 Definice Nechť T je teorie prvního řádu s jazykem aritmetiky L.

(i) Říkáme, že k-ární *relace* $R \subseteq N^k$ *je representovatelná v teorii* T, jestliže existuje formule A v jazyce L s volnými proměnnými $x_1, x_2, \ldots, x_k$ taková, že pro libovolná $n_1, n_2, \ldots, n_k \in N$ platí

$$\text{je-li } R(n_1, n_2, \ldots, n_k) \text{ potom } T \vdash A(\overline{n_1}, \overline{n_2}, \ldots, \overline{n_k})$$
$$\text{je-li } \neg R(n_1, n_2, \ldots, n_k) \text{ potom } T \vdash \neg A(\overline{n_1}, \overline{n_2}, \ldots, \overline{n_k})$$

Říkáme, že *formule A representuje relaci R v teorii T.*

(ii) Říkáme, že (částečná) k-ární *funkce* f, $y = f(x_1, x_2, \ldots, x_k)$ *je representovatelná v teorii* T, jestliže existuje formule A v jazyce L s volnými proměnnými $x_1, x_2, \ldots, x_k, y$ taková, že pro libovolná $n_1, n_2, \ldots, n_k \in N$ platí

$$T \vdash A(\overline{n}_1, \overline{n}_2, \ldots, \overline{n}_k, y) \leftrightarrow y = \overline{n}_{k+1}$$

kde $n_{k+1} = f(n_1, n_2, \ldots, n_k)$.

Říkáme, že formule A representuje funkci f v teorii T. □

Následující lemma uvádíme bez důkazu.

5.2 Lemma (i) Jsou-li T a S teorie takové, že $T \subseteq S \subseteq Th(\mathfrak{N})$ [2], potom všechny relace a funkce, které jsou representovatelné v T, jsou také representovatelné v S.

(ii) Ve sporné teorii jsou representovatelné všechny relace a funkce.

(iii) Je-li T bezesporná a rekursivně axiomatizovatelná [3], potom každá relace representovatelná v T je rekursivní a každá funkce representovatelná v T je (částečná) rekursivní. □

5.3 Definice Je-li $T \subseteq Th(\mathfrak{N})$ teorie s jazykem aritmetiky a jsou-li všechny rekursivní relace a všechny částečné rekursivní funkce representovatelné v T, píšeme *Repr* T. □

5.4 Robinsonova aritmetika je teorie prvního řádu s jazykem aritmetiky a následujícími osmi axiomy

Q1	$S(x) \neq 0$	Q6	$x \cdot 0 = 0$
Q2	$S(x) = S(y) \rightarrow x = y$	Q7	$x \cdot S(y) = (x \cdot y) + x$
Q3	$x \neq 0 \rightarrow \exists y(x = S(y))$	Q8	$x \leq y \leftrightarrow \exists z(z + x = y)$
Q4	$x + 0 = x$		
Q5	$x + S(y) = S(x + y)$		

[2] viz 5.6
[3] viz 7.1

Tuto teorii nazýváme *Robinsonova aritmetika* a budeme ji označovat Q. Teorie Q má konečně mnoho axiomů, zavedeme ještě dvě aritmetiky s nekonečným počtem axiomů. □

5.5 Peanova aritmetika je teorie prvního řádu s jazykem aritmetiky. Má axiomy Q1, Q2, Q4 - Q8 a nekonečně mnoho axiomů, které specifikuje následující

Schema indukce: Pro každou formuli A a proměnou x je následující formule axiomem indukce

$$A_x[0] \to \{\forall x(A \to A_x[S(x)]) \to \forall x A\}$$

Tuto teorii nazýváme *Peanova aritmetika* a budeme ji označovat P. Dá se ukázat, že axiom Q3 je v ní dokazatelný. Peanova aritmetika je tedy rozšířením Robinsonovy aritmetiky Q. □

5.6 Úplná aritmetika $Th(\mathfrak{N})$ je teorie v jazyku aritmetiky, jejíž axiomy jsou všechny sentence pravdivé ve standardním modelu aritmetiky $\mathfrak{N}$, tedy

$$Th(\mathfrak{N}) = \{A | A \text{ je uzavřená formule a } \mathfrak{N} \models A\}$$

Protože všechy axiomy Peanovy aritmetiky jsou splněny ve standardním modelu $\mathfrak{N}$, jejich uzávěry jsou pravdivé v $\mathfrak{N}$ a jsou prvky množiny $Th(\mathfrak{N})$. Úplná aritmetika je tedy rozšířením P. □

Následující věty uvádíme bez důkazu.

5.7 Věta (i) Repr Q

(ii) Repr P

(iii) Repr $Th(\mathfrak{N})$ □

5.8 Nejtěžší je dokázat representovatelnost rekursivních relací a funkcí v nejslabší z těchto tří aritmetik v Robinsonově aritmetice Q. Zbývající tvrzení pak plynou z Lemmatu 5.2.

5.9 Lemma o diagonalisaci Nechť T je teorie taková, že platí Repr T. Pro každou formuli A s jednou volnou proměnnou x existuje sentence D_A taková, že platí

$$T \vdash D_A \leftrightarrow A_x[\overline{\sharp D_A}]$$

Sentence D_A říká "mám vlastnost A". □

Důkaz. Budeme diagonalizovat množinu všech formulí s jednou volnou proměnnou v_1. Nechť v dalším označuje $B(v_1)$ popřípadě jen B formuli s jednou volnou proměnnou. Definujme funkci $F : N^2 \to N$ předpisem

$$F(n,m) = \begin{cases} \sharp B(\overline{m}) & \text{je-li } n = \sharp B \\ 0 & \text{jinak} \end{cases}$$

F je rekursivní funkce a pro libovolnou formuli B s jednou volnou proměnnou platí

$$F(\sharp B, m) = \sharp B(\overline{m}) = sub(\sharp B, \sharp v_1, num(m))$$

Z representovatelnosti F v T plyne existence formule $C(v_1, v_2, v_3)$, která representuje funkci F v T. Pro libovolnou formuli A s jednou volnou proměnnou položíme

$$D(x) \equiv \forall z(C(x,x,z) \to A(z))$$
$$D_A \equiv \forall z(C(\overline{\sharp D}, \overline{\sharp D}, z) \to A(z))$$

Protože D má jednu proměnnou volnou a D_A je tvaru $D(\overline{\sharp D})$, dostáváme

$$F(\sharp D, \sharp D) = \sharp D_A \tag{1}$$

Nyní zbývá dokázat

$$T \vdash D_A \leftrightarrow A(\overline{\sharp D_A})$$

K důkazu ekvivalence použijeme toho, že formule C representuje funkci F. Proto podle (1)

$$T \vdash C(\overline{\sharp D}, \overline{\sharp D}, z) \leftrightarrow z = \overline{\sharp D_A} \tag{2}$$

Přitom podle vět o rovnosti platí

$$\vdash A(\overline{\sharp D_A}) \leftrightarrow \forall z(z = \overline{\sharp D_A} \to A(z))$$

odtud s použitím (2) a věty o ekvivalenci dostáváme

$$T \vdash A(\overline{\sharp D_A}) \leftrightarrow \forall z(C(\overline{\sharp D}, \overline{\sharp D}, z) \to A(z))$$

Na pravé straně ekvivalence je formule D_A. Tím je věta dokázána. □

6 Nedefinovatelnost pravdy v aritmetice

Budeme nyní zkoumat sémantiku formálních systémů, které se zdají dosti silné, aby mohly svými prostředky vyjadřovat tvrzení o pravdivosti či nepravdivosti svých tvrzení.

Je-li T teorie s jazykem aritmetiky, uvažujeme množinu všech sentencí pravdivých ve všech modelech teorie T, tedy množinu

$$Th(T) = \{A | A \text{ je sentence a } T \models A\} \qquad (1)$$

Abychom mohli říci, že pojem pravdy v teorii T je definovatelný v teorii T samé, je třeba ukázat, že množina (1) je representovatelná v teorii T.

6.1 Věta (A. Tarski) (i) Nechť T je bezesporné rozšíření Robinsonovy aritmetiky Q, pro které platí $Repr(T)$. Je-li množina $Th(T)$ všech pravdivých sentencí teorie T representovatelná v T, pak existuje sentence D jazyka teorie T taková, že ani D ani $\neg D$ není prvkem množiny $Th(T)$.

(ii) $Th(\mathfrak{N})$ není representovatelná v $Th(\mathfrak{N})$. □

K důkazu věty použijeme následující tvrzení.

6.2 Lemma Nechť T je bezesporné rozšíření aritmetiky Q, Je-li množina všech pravdivých sentencí teorie T representovatelná v T, potom existuje sentence D taková, že ani D ani $\neg D$ není dokazatelná v T. □

Důkaz. Předpokládejme, že množina (1) je representovatelná v T a že formule $True(x_0)$ tuto množinu representuje. Potom pro každou sentenci A platí

$$T \vdash True(\overline{\sharp A}) \text{ právě když } T \models A$$

Podle věty o úplnosti můžeme na pravé straně nahradit sémantický symbol pro splňování $\models$ symbolem $\vdash$ pro dokazatelnost. Položíme-li $False(x_0) \equiv \neg True(x_0)$, pak podle lemmatu o diagonalizaci 5.9 existuje sentence D taková, že

$$T \vdash D \leftrightarrow False(\overline{\sharp D}) \qquad (2)$$

Sentence D říká "já nejsem pravdivá". Budeme-li nyní postupovat podobně jako při rozboru paradoxu lháře dokážeme, že ani D ani $\neg D$ nemůže být větou teorie T.

Je-li $T \vdash D$ potom $T \vdash False(\overline{\sharp D})$. Z definice predikátu False pak $T \nvdash D$ ve sporu s předpokladem o dokazatelnosti D. Sentence D tedy není větou.

Je-li naopak $T \vdash \neg D$ potom podle (2) je $T \vdash True(\overline{\sharp D})$ a tedy $T \vdash D$, odkud by plynula spornost teorie T. Proto ani $\neg D$ není větou teorie T. Tím je lemma dokázána. □

Důkaz Věty 6.1 Tvrzení (i) vyplývá předchozího lemmatu a faktu, že podle Věty o úplnosti predikátové logiky platí $T \vdash D$ právě když $T \models D$. Teorie T tedy není úplná.

Tvrzení (ii) plyne z (i) a faktu, že $Th(\mathfrak{N})$ je úplná teorie. □

6.3 Tarského Věta o nedefinovatelnosti pravdy má velký význam pro studium sémantiky. Tvrzení (ii) ukazuje, že dostatečně silný formální systém nemůže současně splňovat následující dva požadavky.

(i) T je úplný, to znamená, že každá sentence je buď pravdivá nebo nepravdivá v T.

(ii) Pravdu, tedy predikát $True(x)$ lze vyjádřit v T.

Známý paradox lháře je založen na předpokladu, že oba požadavky (i) a (ii) jsou splněny v běžném životě.

Tarského větu lze stručně vyjádřit tvrzením "není možné definovat pravdu pro aritmetiku uvnitř aritmetiky ".

7 Gödelovy věty

Je-li T teorie s jazykem aritmetiky, ukázali jsme jak se kódují termy a formule jazyka aritmetiky a můžeme definovat množinu Thm_T kódů vět teorie T následovně

$$Thm_T = \{\sharp A \mid A \text{ je formule a } T \vdash A\} \qquad (1)$$

Nyní můžeme precisovat pojem rozhodnutelnosti, který jsme v intuitivním způsobem zavedli v prvním paragrafu této kapitoly. Teorie T je rozhodnutelná, právě když (1) je rekursivní množina. Abychom pro konkrétní teorii T dokázali, že je rozhodnutelná, potřebujeme algoritmy, které dovolují

- rozpoznávat kódy axiomů teorie T
- rozpoznávat kódy axiomů predikátové logiky
- rozpoznávat kódy důkazů teorie T

Naznačíme jak toho lze dosáhnout. Budeme pracovat s jazykem aritmetiky, ale metoda kódování je obecná. Nejprve budeme charakterizovat množinu axiomů a odvozovací pravidla predikátové logiky.

7.1 Kódy axiomů a odvozovacích pravidel predikátové logiky

Definujeme-li predikát $Avl(x)$ předpisem

$$\begin{aligned} Avl(x) \leftrightarrow \exists u < x \exists v < x \exists w < x \{ & F(u) \,\&\, F(v) \,\&\, F(w) \,\& \\ & \& \{[x = \langle \sigma(\rightarrow), u, \langle \sigma(\rightarrow), v, u \rangle\rangle] \vee \\ & \vee [\quad \cdots \quad] \vee [\quad \cdots \quad]\}\} \end{aligned}$$

První hranatá závorka popisuje výrokové axiomy tvaru

$$(A \rightarrow (B \rightarrow A))$$

kde $u = \sharp A$ a $v = \sharp B$. Charakterizaci zbývajících dvou typů axiomů výrokové logiky není těžké doplnit. Podobně lze pomocí predikátů $Sub(x, y, z), Sbtl(x, y, z)$ a $Fvar(x, y)$ z odstavce 4.8 charakterizovat oba typy axiomů pro universální kvantifikátor a axiomy rovnosti. Tímto postupem bychom definovali predikát $Apl(x)$, charakterizující množinu všech kódů axiomů predikátové logiky. Pokud bychom postupovali rozumně, dalo by se ukázat, že je to rekursivní predikát.

V obecném případě může být množina axiomů teorie T dána jako zcela libovolná podmnožina množiny všech formulí. Proto

$$Ax_T = \{\sharp A \mid A \in T\}$$

nemusí být rekursivní množina. V případě, že je rekursivní říkáme, že *teorie* T *je rekursivně axiomatizovaná.*

Má-li teorie jen konečně mnoho axiomů, pak je rekursivně axiomatizovaná. To je případ Robinsonovy aritmetiky Q. Dá se ukázat, že schema indukce Peanovy aritmetiky má rekursivní množinu kódů, takže Peanova aritmetika je také rekursivně axiomatizovaná.

Na druhé straně ukážeme, že množina kódů axiomů $Th(\mathfrak{N})$ není rekursivní. To znamená, že $Th(\mathfrak{N})$ není rekursivně axiomatizovaná.

Při charakterizací kódů důkazů potřebujeme definovat predikáty, které charakterizují odvozování podle pravidel *Modus ponens* a *Pravidla generalizace.*

Položíme-li

$$Mp(x, y, z) \leftrightarrow (F(x) \,\&\, F(z) \,\&\, y = \langle \sigma(\rightarrow), x, z \rangle)$$

pak $Mp(x, y, z)$ platí právě když formule s kódem z je odvozena z formulí s kódy x a y pravidlem Modus ponens. Podobně definujeme-li predikát $Gen(x, y)$ definovaný předpisem

$$Gen(x,y) \leftrightarrow (F(x) \;\&\; \exists i < y(y = \langle \sigma(\forall), \langle 2i\rangle, x\rangle))$$

potom $Gen(x,y)$ platí právě když y je kód formule, která byla odvozena pravidlem generalizace z fomule s kódem x. Dá se ukázat, že oba predikáty jsou rekursivní.

7.2 Kódování důkazů Následující úvahu lze provést pro rekursivně axiomatizované teorie s obecným jazykem, ne nutně s jazykem aritmetiky. Formule a důkazy takové teorie lze kódovat metodou popsanou v odstavcích 4.6 a 7.1 pro speciální případ jazyka aritmetiky.

Nechť T je rekursivně axiomatizovaná teorie. V důkazech vět teorie T vystupují jednak axiomy predikátové logiky, jednak speciální axiomy teorie T.

Položíme-li

$$Ax(x) \leftrightarrow (Apl(x) \vee Ax_T(x))$$

pak Ax je rekursivní predikát, který charakterizuje množinu všech formulí, které mohou v důkazech vět teorie T vystupovat jako axiomy. Predikát $Proof_T$, takový, že $Proof_T(x,y)$ platí právě když číslo x je kódem důkazu formule s kódem y lze definovat takto

$$\begin{aligned} Proof_T(x,y) \leftrightarrow \{&Seq(x) \;\&\; lh(x) > 0 \;\&\; y = (x)_{lh(x)-1} \;\&\\ &\forall i < lh(x)[Ax((x)_i) \vee \exists j < i \; \exists k < i \; (Mp((x)_j, (x)_k, (x)_i)) \vee \\ &\exists j < i \; (Gen((x)_j, (x)_i))]\} \end{aligned}$$

Potom $Proof_T$ je rekursivní predikát a $Proof_T(d, \sharp A)$ platí právě když d je kódem důkazu formule A.

Můžeme nyní definovat

$$Thm_T(x) \leftrightarrow \exists d Proof_T(d,x)$$

protože na pravé straně je neomezený kvantifikátor, nemůžeme říci, že predikát Thm_T je rekursivní. Z výsledků teorie rekurse však plyne, že neomezenou existenční kvantifikací dostáváme z rekursivního predikátu rekursivně spočetný predikát. Množina (1) je tedy rekursivně spočetná.

Odtud plyne následující tvrzení.

7.3 Věta Je-li T rekursivně axiomatisovaná teorie, potom množina Thm_T kódů vět teorie T je rekursivně spočetná. □

7.4 Věta o nerozhodnutelnosti aritmetiky (Church 1936) Je-li T bezesporné rozšíření Robinsonovy aritmetiky, Q, potom T je nerozhodnutelná teorie. □

Důkaz. Kdyby T byla rozhodnutelná teorie, pak existuje formule $A(x)$ s jednou volnou proměnnou x, representující v Q množinu všech vět teorie T. To znamená, že pro libovolnou sentenci B platí

$$T \vdash B \text{ implikuje } Q \vdash A(\overline{\sharp B})$$
$$T \nvdash B \text{ implikuje } Q \vdash \neg A(\overline{\sharp B})$$

Nechť D je diagonální sentence pro formuli $\neg A(x)$, tedy nechť

$$Q \vdash D \leftrightarrow \neg A(\overline{\sharp D})$$

Můžeme říci, že formule D o sobě tvrdí "já nejsem věta teorie T".

Uvažujme následující dva případy.

(i) Je-li

$$T \vdash D,$$

potom

$$Q \vdash A(\overline{\sharp D}),$$

odkud dostáváme

$$Q \vdash \neg D$$

a také

$$T \vdash \neg D.$$

protože T je rozšířením Q. To protiřečí bezespornosti T.

(ii) Podobně, je-li

$$T \nvdash D$$

potom

$$Q \vdash \neg A(\overline{\sharp D}).$$

Odtud

$$Q \vdash D$$

a také

$$T \vdash D,$$

a to je ve sporu s předpokladem o nedokazatelnosti formule D v T. Teorie T tedy není rozhodnutelná. □

7.5 Lemma Je-li T úplná a rekursivně axiomatizovaná teorie, potom T je rozhodnutelná teorie. □

Důkaz. Úplnost teorie se vztahuje k uzavřeným formulím (sentencím). Bude proto užitečné k danému kódu formule sestrojit kód formule, která je ekvivalentní s jejím uzávěrem. Definujme funkci F následujícím způsobem

$$F(0,a) = a$$
$$F(n+1,a) = \langle \sigma(\forall), \sharp x_n, F(n,a) \rangle$$

a funkci U předpisem

$$U(a) = F(a+1,a)$$

Je-li a kód formule A, potom

$$U(a) = \sharp(\forall x_a \forall x_{a-1} \ldots \forall x_0\ A)$$

Přitom, je-li x_i proměnná, která má volný výskyt v A, platí

$$i \leq \sharp x_i < \sharp A = a,$$

tedy $\forall x_a \forall x_{a-1} \ldots \forall x_0\ A$ je uzavřená formule. Z úplnosti teorie T plyne pro každou uzavřenou formuli A

$$\neg Thm_T(\sharp A) \leftrightarrow Thm_T(\langle \sigma(\neg), \sharp A \rangle)$$

Pro libovolné přirozené číslo a dostáváme

$$\begin{aligned} \neg Thm_T(a) &\leftrightarrow \neg For(a) \vee Thm_T(\langle \sigma(\neg), U(a) \rangle) \\ &\leftrightarrow \exists d(\neg For(a) \vee Proof_T(d, \langle \sigma(\neg), U(a) \rangle)) \end{aligned}$$

kde $For(a)$ znamená, že číslo a je kodem formule. Pravá strana ekvivalence ukazuje, že $\neg Thm_T$ je rekursivně spočetný predikát. Protože Thm_T je také rekursivně spočetný, podle Postovy věty je Thm_T rekursivní predikát a T je rozhodnutelná teorie. □

7.6 Věta o neúplnosti aritmetiky (Gödel, Rosser) Je-li T rekursivně axiomatizované rozšíření Robinsonovy aritmetiky, pak T není úplná teorie. □

Důkaz. Kdyby T byla úplná, pak je podle lemmatu 7.5 rozhodnutelná, a to odporuje tvrzení Věty 7.4. Teorie T není úplná. □

7.7 Označení Je-li T bezesporné, rekursivně axiomatizované rozšíření Peanovy aritmetiky P a A je formule, píšeme

$\Box\ A$ jako zkratku za formuli $Thm_T(\overline{\sharp A})$
Con_T jako zkratku za formuli $\neg\Box\ 0 = 1$ □

7.8 Druhá věta o neúplnosti (Gödel 1931) Nechť T je bezesporné, rekursivně axiomatizovatelné rozšíření Peanový aritmetiky.

Potom

$$T \nvdash Con_T$$

□

Důkaz používá vět 7.10 a 7.12. Zatím uvedeme důsledek, který se týká teorie množin. ZFC označuje teorii množin s axiomem výběru v axiomatice podle Zermela a Fraenkela. Podrobný výklad této teorie množin a její axiomatiky je možné najít v monografii [1]. □

7.9 Důsledek Je-li ZFC bezesporná teorie, potom

$$ZFC \nvdash Con_{ZFC}.$$

□

Následující tři tvrzení se nazývají Löbovy podmínky. Jsou elegantnější versí Hilbertových a Bernaysových podmínek na formalisaci důkazů v aritmetice.

7.10 Věta Je-li T bezesporné, rekursivně axiomatizované rozšíření Peanovy aritmetiky, potom platí

(L1) $T \vdash \Box(A \to B) \to (\Box A \to \Box B)$

(L2) $T \vdash \Box A \to \Box\Box A$

(L3) $T \vdash A$ implikuje $T \vdash \Box A$

□

Důkaz je proveden v 7.15 a 7.19. □

7.11 Princip reflexe Pro dokazatelné formule Peanovy aritmetiky z věty o korektnosti plyne

$$P \vdash A \text{ implikuje } \mathfrak{N} \models A$$

Formalizujeme-li toto tvrzení v Peanově aritmetice dostáváme tak zvaný *princip reflexe* pro formuli A.

$$P \vdash \Box\, A \to A$$

□

Následující věta ukazuje, že princip reflexe v P lze dokázat jen pro věty Peanovy aritmetiky a že obdobné tvrzení platí pro rekursivně axiomatizovaná rozšíření Peanovy aritmetiky.

7.12 Věta (Löb 1955) Nechť T je rekursivně axiomatizované rozšíření Peanovy aritmetiky, nechť A je sentence, potom platí

$$T \vdash \Box\, A \to A \text{ právě když } T \vdash A$$

□

Důkaz a) Je-li $T \vdash A$, pak odtud lze odvodit $T \vdash \Box A \to A$ podle pravidla Modus ponens za pomoci prvního schematu axiomů výrokové logiky.

b) Je-li

$$T \vdash \Box A \to A \tag{2}$$

použijeme diagonalizační lemma 5.9 na formuli $Thm_T(x) \to A$. Nechť B je sentence taková, že

$$T \vdash B \leftrightarrow (\Box B \to A) \tag{3}$$

Důkaz $T \vdash A$ rozdělíme na dvě části. Nejprve dokážeme $T, \Box B \vdash A$ a potom vyloučíme předpoklad $\Box B$.

$T, \Box B \vdash \Box(\Box B \to A)$	(3), (L1), (L3)	(4)
$T, \Box B \vdash \Box\Box B \to \Box A$	(L1)	(5)

Nyní z přepokladu $\Box B$ dostáváme

$T, \Box B \vdash \Box\Box B$	(L2)	(6)
$T, \Box B \vdash \Box A$	(5), (6)	(7)
$T, \Box B \vdash A$	(2), (7)	(8)
$T \vdash \Box B \to A$	věta o dedukci, (8)	(9)
$T \vdash B$	(3), (9)	(10)
$T \vdash \Box B$	(L3), (10)	(11)
$T \vdash A$	(9), (11)	

Tím je věta dokázána. □

7.13 Důkaz druhé Gödelovy věty o neúplnosti

Kdyby

$$T \vdash Con_T$$

potom také

$T \vdash \neg\Box(0 = 1)$

$T \vdash \Box(0 = 1) \to (0 = 1)$ tautologický důsledek

$T \vdash (0 = 1)$ Löbova věta

To je ve sporu s předpokladem bezespornosti T. Tím je věta dokázána. □

7.14 Důsledek Existuje model $\mathfrak{M}$ Peanovy aritmetiky takový, že platí

$$\mathfrak{M} \models \exists d Proof(d, \sharp(0 = 1)) \tag{12}$$

□

Důkaz. Podle druhé Gödelovy věty o neúplnosti

$$P \nvdash Con_P$$

tedy sentence

$$\Box(0 = 1)$$

je bezesporná. Existuje tedy model Peanovy aritmetiky, ve kterém je tato sentence pravdivá. Tento model je nestandardní, protože ve standardním modelu $\mathfrak{N}$ není pravdivá sentence (12). Přirozené číslo $m \in \mathfrak{M}$, takové, že

$$\mathfrak{M} \models Proof_P(m, \sharp(0 = 1))$$

musí být proto nestandardní. Tím je Důsledek 7.14 dokázán. □

Příklad (Shepherdson) Předvedeme důkaz nestandardní délky dokazující libovolně zvolenou sentenci A. Tato posloupnost však není kódovatelná přirozeným číslem v žádném nestandardním modelu Peanovy aritmetiky, a proto není v žádném takovém modelu důkazem.

Je-li A libovolná sentence například $0 = 1$, pak posloupnost

$$A \to (A \to A), A \to (A \to A), \ \ldots \ ; \ \ldots \ A, A \to A, A, A \to A, A \ldots \tag{13}$$

je příkladem důkazu nestandardní délky sentence A.

První úsek důkazu, který odpovídá standardním přirozeným číslům (numerálům) sestává z nekonečně mnoha kopií axiomu výrokové logiky a za ním (po středníku) následuje úsek důkazu sestávající z kopií dokazované sentence A, která se střídá s implikací $A \to A$. Připomeňme, že v nestandardním modelu nemůže existovat nejmenší nestandardní číslo. Jeho předchůdcem by pak bylo největší standardní číslo, a takové neexistuje. Každá formule v nestandardní části důkazu (za středníkem) má nekonečně mnoho nestandardních předchůdců, a je proto odvozena pravidlem modus ponens z předchozích formulí. Posloupnost (13) je tedy

důkazem sentence A, ale dá se ukázat, že v žádném modelu Peanovy aritmetiky není kódována přirozeným číslem, protože koncový úsek důkazu, který tvoří posloupnost formulí za středníkem, je definovatelný, ale nemá první prvek. □

Uvedený příklad naznačuje úskalí, která mohou představovat důkazy nestandardní délky.

7.15 Důkaz Löbových podmínek Podmínku (L1) lze dokázat jako jednoduchou syntaktickou větu. Můžeme sestrojit term $t(d,d')$ takový, že

$$T \vdash Proof_T(d, \overline{\sharp(A \to B)}) \& Proof_T(d', \overline{\sharp A}) \to Proof_T(t(d,d'), \overline{\sharp B}) \qquad (14)$$

kde $t(d,d')$ kóduje odvození podle pravidla modus ponens a je sestrojen takto

$$t(d,d') = \begin{cases} d * d' * \langle b \rangle & \text{jestliže } \exists u < d \; \exists \, v < d' (Proof_T(d,u) \; \& \\ & \quad Proof_T(d',v) \; \& Mp(u,v,b)) \\ 0 & \text{jinak} \end{cases}$$

Potom (L1) je bezprostředním důsledkem (14). □

7.16 Definice. Omezené kvantifikátory a omezené formule Nechť L je jazyk aritmetiky nebo nějaké jeho rozšíření, nechť x, y jsou dvě různé proměnné a A je formule.

(i) výraz $(\forall x \leq y)A$ chápeme jako zkratku za formuli

$$\forall x(x \leq y \to A)$$

a výraz $(\exists x \leq y)A$ chápeme jako zkratku za formuli

$$\exists x(x \leq y \; \& \; A)$$

Říkáme, že výraz $(Qx \leq y)$, kde symbol Q zastupuje universální nebo existenční kvantifikátor je omezený universální nebo existenční kvantifikátor. Při omezené kvantifikaci se vedle neostré nerovnosti používá i nerovnost ostrá.

(ii) Říkáme, že A je omezená formule, jestliže

- A je atomická formule
- A je tvaru $\neg B$ nebo $B \to C$ a B, C jsou omezené formule
- A je tvaru $(\forall x \leq y)B$ nebo $(\forall x < y)B$ a B je omezená formule

Jsou-li B, C omezené formule, potom

$$B \vee C \quad B \,\&\, C \quad B \leftrightarrow C$$

jsou také omezené formule protože jsou to zkratky za formule, které se dají z B a C vyjádřit pomocí negace a implikace. Podobně

$$(\exists x \leq y)B$$

je omezená formule, protože je to zkratka za omezenou formuli $\neg(\forall x \leq y)\neg B$. Totéž platí pro $(\exists x < y)B$.

Množinu všech omezených formulí označujeme Δ_0, ale také Σ_0 nebo Π_0, protože to jsou formule, které nemají žádný neomezený existenční ani universální kvantifikátor. □

7.17 Definice. Σ_1 formule Říkáme, že A je $\Sigma_1 formule$, jestliže A je formule tvaru $\exists x\ B$, kde B je omezená formule. Σ_1 formule mají jeden neomezený existenční kvantifikátor. □

Následující větu uvádíme bez důkazu.

7.18 Věta. Σ_1 úplnost Robinsonovy aritmetiky
(i) Je-li A Σ_1 sentence, potom

$$\mathfrak{N} \models A \text{ implikuje } Q \vdash A$$

(ii) Je-li T rekursivně axiomatizované rozšíření Peanovy aritmetiky, pak pro libovolnou Σ_1 sentenci D platí

$$T \vdash D \rightarrow \Box D$$

Připomeňme, že v tomto případě je $\Box D$ je zkratka za formuli $Thm_T(\overline{\sharp D})$. □

7.19 Důkaz podmínek (L2) a (L3) Nechť T je rekursivně axiomatizované rozšíření Peanovy aritmetiky. Pro libovolnou formuli A je $\Box A$ Σ_1 sentence. Podle tvrzení (ii) věty 7.18 potom

$$T \vdash \Box A \rightarrow \Box\Box A$$

a to je podmínka (L2). K důkazu (L3) si stačí uvědomit, že

$$T \vdash A \text{ implikuje } \mathfrak{N} \models \Box A$$

podmínka (L3) potom plyne z tvrzení (i) předchozí věty. □

Literatura

[1] B. Balcar, P. Štěpánek, *Teorie množin*, ACADEMIA Praha 1986

[2] H. Barendregt, *The Incompletness Theorems,* Communications of Mathematical Institute, Rijksuniversiteit Utrecht, 4 - 1976

[3] J. Bell, M. Machover, *A Course in Mathematical Logic,* North Holland, Amsterdam, New York 1977

[4] H.-D.Ebbinghaus, J. Flum, W. Thomas, *Mathematical Logic,* 2. vydání, Springer-Verlag, Heidelberg, Berlin, 1994

[5] P. Hájek, V. Švejdar, *Matematická logika,* studijní materiál, Praha, listopad 1994

[6] J. R. Shoenfield, *Mathematical Logic,* Addison-Wesley Publishing Co., Reading, MA 1967

[7] J. Šebelík, P. Štěpánek, *Horn Clause Programs for Recursive Functions,* in: Logic Programming, K.L. Clark, S. Å. Tärnlund (editor), ACADEMIC PRESS, London 1982, str 325–340.