Vyčíslitelnost

Doc. RNDr. Antonín Kučera, CSc.

15. července 2021

Zapsal Kyrylo Karlov na základě přednášek v LS 2020/21

Obsah

1 Zkratky

1. ČRF - částečně rekurzivní funkce.
2. ORF - obecně rekurzivní funkce.
3. PRF - primitivně rekurzivní funkce.
4. r.s. - rekurzivně spočetná.
5. ZAS - zakladní aritmetická síla.
6. PA - Peanova Aritmetika.
7. ORP - obecně rekurzivní predikát.
8. PNF - prenexní normální tvar.

2 Úvod

2.1 Historická vsuvka

1900: 10. Hilbertův problém, úplnost aritmetiky. Gödel dokázal že nejde. V prvním větě použil *Primitivně rekurzivní funkce.*

Definice 2.1 (Primitivně rekurzivní funkce). Primitivně rekurzivní funkce - podmnožina efektivně vyčíslitelných funkcí, jsou všude definované.

Tyto ale nestačí pro hlavní problém dokazatelnosti.
Při dalším vývoji se vyvinul kalkulus tzv. obecně rekurzivních funkcí ORF a částečně rekurzivních funkcí ČRF.

Značení 2.2 (Definiční obor). $dom(\varphi)$ - definiční obor.

Značení 2.3 (Obor hodnot). $range(\varphi)$ - obor hodnot.

Definice 2.4 (Rekurzivní funkce). Funkce $f : \Sigma^* \to \Sigma^*$ je *částečně rekurzivní*, pokud je turingovsky vyčíslitelná (tedy existuje Turingův stroj M takový, že $\forall x \in dom(f) \Leftrightarrow M(x) \downarrow \wedge f(x)$ odpovídá obsahu pásky $M(x)$ na konci výpočtu).
Funkce $f : \Sigma^* \to \Sigma^*$ je *obecně rekurzivní*, je-li částečně rekurzivní a $dom(f) = \Sigma^*$ (všude definovaná).

Poznámka 2.5 (Church-Turing teze). Historický vývoj:

- A. Church vyvinul λ-konverze (λ-calculus) a dokázal, že neexistuje algoritmus tzv "rozhodovací". Lambda konverze jsou poměrně obtížné.
- Turing, nezávislé na Churchovi v roce 1936 vyvinul Turing Machines a dokázal nevyčíslitelnost Halting problému.

Pak ostatní prohlásili Church-Turing tezi, že všechno, co je efektivně vyčíslitelné je Turingovsky nebo λ-konverzi vyčíslitelné.

Definice 2.6 (λ-calculus). Nechť C je množina konstant, nechť V je (spočetná) množina proměnných. Množina tzv. λ terms Λ je nejmenší množina tž:

- $C \subseteq \Lambda$.
- $V \subseteq \Lambda$
- nechť $t_1, t_2 \in \Lambda$ termy, pak aplikace $t_1\ t_2$ jako v Haskellu je taky term
- $t \in \Lambda, x \in V \Rightarrow \lambda x.t \in \Lambda$. V Haskellu:
  $(\backslash x \to t)$
  Což je funkce s parametrem x a vrací t.

Jako závěr, formální, efektivně dokazovací systém nemůže uplně popsat pravdu. Je mnohem složitější.

Poznámka 2.7 (System PRF(odbočka)). Funkcionální systém, postavený na axiomech:

- Základní funkce: 0, +1, id (resp. vydělení i-té složky)
- 2 Odvozovací pravidla:
  1) substituce
  2) operátor primitivní rekurze. Jednoduše řečeno, výpočet v bode $(y+1)$ uděláme rekurzivně z bodu "y".

Pak se vezme *tranzitivní uzávěr* - všechno co jde odvodit ze základních funkci pomocí odvozovacích pravidel. Na rozdíl od ČRF nemáme **while**, čímž dostaneme jenom podmnožinu ČRF.
Substituce:

$$S_n^m(f, g_1, \ldots, g_n) = f(g_1(y_1, \ldots, y_n), \ldots, g_m(y_1, \ldots, y_n))$$

Primitivní rekurze:

Poznámka 2.8 (Kleeneho system ČRF). Pak přidáním operátoru μ tzn. **while** k předchozímu systému dostaneme ČRF (znovu *tranzitivní uzávěr*). Je komplikovaný, je lepší používat nějaké dokazování.

2.2 Terminologie

Přibližně do roku 1990 převládala terminologie ORF, ČRF zavedená Kleene. Pak byla snaha změnit na **computable functions** - efektivně vyčíslitelné.

Definice 2.9 (Rekurzivní množina). Množina je rekurzivní, neboli rozhodnutelná (decidable, computable) - efektivně rozhodnutelná. Jednoduše řečeno, máme program, který na každém vstupu se vždy zastaví a rozhodne ANO nebo NE (jestli slovo patří do ni).

Definice 2.10 (Rekurzivně spočetná množina). Množina je rekurzivně spočetná (částečně rozhodnutelná), nebo computably enumerable. Formálně je definičním oborem nějakého programu (tzn. částečně rekurzivní funkce, TS etc.).

Poznámka 2.11. Na rozdíl od kurzu ZSV, kde jsme definovali funkce $f : \{0,1\}^n \to \{0,1\}^n$, budeme zkoumat funkce aritmetické.

$$f : \mathbb{N} \to \mathbb{N}, f : \mathbb{N}^k \to \mathbb{N}$$

Nemusí být všude definované.
Přístupy jsou ekvivalentní, protože můžeme očíslovat slova.

Poznámka 2.12. Formálně nemáme klasické k-tice jako vektory, ale kodujeme všechno do přirozených čísel. Je známá jako Cantorova metoda párování.

Značení 2.13 (Konvergence). Program **konverguje** - znamená že se zastaví za konečný počet kroků.

Připomenutí 2.14. Rekurzivita, rekurzivní spočetnost se zachovává na $\cup, \cap$.
Rekurzivita se taky zachovaná při $\neg$ (doplněk). Rekurzivní spočetnost nikoliv.

Věta 2.15 (Postova). *L je rozhodnutelný $\iff$ $L \wedge \bar{L}$ jsou r.s. (c.e.).*

Důkaz. $\Rightarrow$. Z TS pro L sestavíme pro doplněk znegovaním všech odpovědí.
$\Leftarrow$. Nechť $L(M) = L \wedge L(B) = \bar{L}$, pak sestavíme TS pro rozhodnutí L.

```
1. Pust B, M paralelně
2. if(Acc(M, x))
3.      accept
4. if(Acc(B, x))
5. reject
```

Paralelní spuštění lze implementovat pomocí 2 pásek, případně je slepit do 1. □

Definice 2.16 (Gödelové číslo). Index programu. Nechť φ je ČRF, P_e je program který ji vyčísluje. Pak index funkce φ je e.

Poznámka 2.17. Každá ČRF má nekonečně mnoho programů, takže i nekonečně mnoho indexů. Očíslování programů generuje očíslování funkcí.
V jistém smyslu, nezáleží na konkretním očíslování pokud je efektivní (nemáme čas toto dokazovat). 2 různé efektivní numerace jsou *efektivně ekvivalentní*.

Připomenutí 2.18 (Univerzální TS). Dostane program M a data x, simuluje výpočet $M(x)$.
Pro nás to bude *univerzální ČRF*.

Definice 2.19 (Univerzální ČRF).

$$\Psi_n(e, x_1, x_2, \ldots, x_n)$$

kde e je index programu, x_i jsou data.
Občas se značí

$$\varphi_e^n(x_1, \ldots, x_n) \simeq \{e\}(x_1, \ldots, x_n)$$

Značení 2.20 (Univerzální ČRF).

$$\varphi_e^n(x_1, \ldots, x_n) \simeq U(\mu_y(T_n(x_1, \ldots, x_n, y)))$$

- Kde T_n je primitivně rekurzivní predikát, který říká "za n kroků".

- U je primitivně rekurzivní funkce 1 proměnné, která "vydělí"výsledek z mezivýsledků (jelikož máme všechno zakodované jako přirozená čísla).
- μ_y říká "nejmenší y".

Věta 2.21 (s-m-n (BD)).

$$\varphi_e^{m+n}(x_1,\dots,x_n,y_1,\dots,y_m) \simeq \varphi^n_{s_n^m(e,x_1,\dots,x_n)}(y_1,\dots,y_m)$$

V Ψ notace

$$\Psi_{n+m}(e,\bar{x},\bar{y}) \simeq \Psi_m(s_n^m(e,\bar{x},\bar{y}))$$

kde $\bar{x},\bar{y}$ jsou vektory pro kratší zápis.
Funkce $s_n^m : e, x_1,\dots,x_n$ vyrobý nový program. Ten čeká na vstup $y_1,\dots,y_m$, k tomu přidá zahardkodované data $x_1,\dots,x_n$ a spustí na to e. Je to pouze syntaktická manipulace dat.

Definice 2.22 (e-tá rekurzivně spočetná množina).

$$W_e = dom(\varphi_e) = \{x : \varphi_e(x)\downarrow\} = \{\Psi_1(e,x)\downarrow\}$$

Poznámka 2.23. Rekurzivně spočetné množiny se někdy definují jako obor hodnot ČRF.
M je rekurzivní množina $\iff$ je oborem hodnot **rostoucí** ČRF.
M je rekurzivně spočetná množina $\iff$ je oborem hodnot **prosté** ČRF.
Rozdíl v obou ekvivalencích souvisí s Halting problémem.

Definice 2.24 (1,m převoditelnost). $A \leq_1 B \iff \exists$ ORF f prostá:

$$x \in A \iff f(x) \in B$$

$A \leq_m B \iff \exists$ ORF f (ne nutně prostá):

$$x \in A \iff f(x) \in B$$

Definice 2.25 (1-úplnost). Množina M je 1-úplná jestliže je rekurzivně spočetná a každá rekurzivně spočetná je 1-převoditelná na M.

Definice 2.26 (K, DIAG).

$$K = \{x : x \in W_x\} = \{x : \varphi_x(x)\downarrow\} = \{x : \Psi_1(x,x)\downarrow\}$$

Taky

$$K_0 = \{\langle x,y\rangle : \varphi_x(y)\downarrow\} = \{\langle x,y\rangle : y \in W_x\}$$

Značení z ZSV

$$DIAG = \{\langle M\rangle : M \in L(M)\} = \{\langle M\rangle : M(\langle M\rangle)\}$$

Lemma 2.27 (DIAG, K). *DIAG je rekurzivně spočetný (částečně rozhodnutelný) ale není rekurzivní (rozhodnutelný).*
Důkaz pomoci Cantorovy diagonální metody.

Důkaz.

$$\overline{K} = \{x : x \notin W_x\}$$

W_x ale jsou všechny rekurzivně spočetné. Z toho

$$\forall x : \overline{K} \neq W_x$$

Takže $\overline{K}$ není částečně rekurzivní. Dle Postovy věty 2.15 K není rozhodnutelná. $\square$

Věta 2.28 (K 1-úplná). *K (taky K_0) je 1-úplná.*

Důkaz. Zavedeme ČRF

$$\alpha(x, y, w)\downarrow \iff \varphi_x(y)\downarrow$$

kde w je fiktivní proměnná. α je ekvivalentní

$$\Psi_1(e, x, y, w)$$

použijeme 2.21

$$\Psi_1(e, x, y, w) \simeq \Psi_1(s_2^1(e, x, y), w) \simeq \varphi_{s_2^1(e,x,y)}(w)$$

dosadíme za $w = s_2^1(e, x, y)$.
Pomoci w se dostáváme na diagonálu. Pak

$$x \in W_y \iff s_2^1(e, x, y) \in K$$

□

Věta 2.29 (Rozšíření Ψ_n). *Ψ_n nemá obecně rekurzivní rozšíření. Jinými slovy neexistuje ORF h rozšíření Ψ takové že*

$$\forall \overline{z} : \Psi_n(\overline{z}) = h(\overline{z})$$

a h je definovaná pro vstupy mimo dom(Ψ_n).
Dokonce, pokud α částečně rekurzivně rozšiřuje Ψ_1, tak najdeme vstup na kterém diverguje

$$\exists x_1 : \alpha(x_1, x_1)\uparrow$$

Důkaz. Použijeme Cantorovu diagonální metodu. Definujme pomocnou ČRF:

$$\beta(x) \simeq 1 \dot{-} \alpha(x, x)$$

Kde $\dot{-}$ je dodefinovaná operace odečítaní pro přirozená čísla. Např $1 \dot{-} 100 = 0$. Jelikož je ČRF $\Rightarrow$ má index e_β, neboli

$$\beta(e_\beta) \simeq \Psi_1(e_\beta, e_\beta) \simeq 1 \dot{-} \alpha(e_\beta, e_\beta)$$

Nechť sporem $\alpha(e_\beta, e_\beta)\downarrow$, pak $\beta(e_\beta)\downarrow$ a tedy

$$\Psi_n(e_\beta, e_\beta)\downarrow$$

Protože α je rozšíření

$$\Psi_n(e_\beta, e_\beta) = \alpha(e_\beta, e_\beta)$$

což je spor protože

$$1 \dot{-} \alpha(e_\beta, e_\beta) = \Psi_n(e_\beta, e_\beta) = \alpha(e_\beta, e_\beta)$$

□

3 Rekurzivně spočetné množiny a predikáty

Poznámka 3.1. R.s. množiny a predikáty je jedno a totéž protože obor pravdivostí predikátu je množina a naležení do množiny je predikát.

Lemma 3.2 ($\exists Q$ r.s). *Pokud Q je rekurzivní $\Rightarrow \exists y Q(...)$ je rekurzivně spočetný.*

Důkaz. Uvažme charakteristickou funkci C_Q predikátu Q. Je všude definovaná, čili je ORF.
Pak následující je ČRF:

$$\mu_y Q(...) \simeq \mu_y (C_Q(...) = 1)$$

□

Věta 3.3 (Univerzální Kleeneho r.s. predikát). *Každý rekurzivně spočetný predikát je tvaru:*

$$\exists y Q(...)$$

Pak r.s. množiny jsou definiční obory ČRF.
Dokonce máme univerzální rekurzivně spočetný predikát

$$\exists y T_n(e, x_1, \ldots, x_n, y)$$

Důsledek 3.4 (Index r.s. predikátů). *Lze definovat index rekurzivně spočetných predikátů.*

Poznámka 3.5. s-m-n věta 2.21 platí i pro predikáty T_n.

Věta 3.6 (Uzavřenost RS). *Rekurzivní spočetnost je uzavřená na $\cup, \cap$. Dokonce efektivně z indexu.*
Máme ORF(dokonce PRF)

$$W_{\alpha(a,b)} = W_a \cap W_b$$

Důkaz. Formálně pro $\cap$:

$$\exists s_1 T_1(a, x, s_1) \wedge \exists s_2 T_1(b, x, s_2) \iff \exists w (T_1(a, x, (w)_{2,1}) \wedge T_1(b, x, (w)_{2,2}))$$

kde w koduje dvojici s_1, s_2.

$$(w)_{2,1}$$

říká, že w je n-tice velikosti 2, vezmi 1. složku.

$$\exists s_1 T_1(a, z, s_1)$$

je reprezentace množiny W_a pomoci univerzálního predikátu.
Dohromady máme rekurzivně spočetný predikát. Takže

$$\exists z T_3(e, a, b, x, z)$$

Kde e je program, který pouští oba dva programy a čeká až se jeden z nich zastaví. Použijeme s-m-n 2.21 pro predikáty

$$\exists z T_3(e, a, b, x, z) \iff \exists z T_1(s_2^1(e, a, b), x, z)$$

Pak definujme

$$\alpha(a, b) := s_2^1(e, a, b)$$

Analogicky pro $\cup$. □

Definice 3.7 (Omezený existenční kvantifikátor).

$$\exists_{y<z} Q$$

Jmenuje se konečná dizjunkce.

Definice 3.8 (Omezený všeobecný kvantifikátor).

$$\forall_{y<z} Q$$

Jmenuje se konečná konjunkce.

Věta 3.9 (Omezená kvantifikace (BD)). *Rekurzivní spočetnost je uzavřená na omezené kvantifikace. Dokonce efektivně na indexech.*

Důkaz. Pro existenční spustíme z programů paralelně a čekáme až jeden přijme.
Pro všeobecný spustíme paralelně a čekáme jestli všechny přijmou. □

Věta 3.10 (Neomezená kvantifikace). *Rekurzivní spočetnost je uzavřená na existenční kvantifikace.*

Důkaz. Analogicky jako důkaz pro $\cap$, nahradíme dva existenční kvantifikátory jediným s dvojicí.

$$\exists y \exists s : T_n(...) \simeq \exists k = \langle y, s \rangle ...$$

□

Poznámka 3.11. Pro všeobecnou kvantifikaci již neplatí (ani pro částečně rozhodnutelné). Protipříkladem je $\overline{K}$ která není č.r. (tzn. r.s.), kterou lze zapsat pomocí všeobecného kvantifikátoru

$$x \in \overline{K} \iff \forall s \neg T_1(x, x, s)$$

Kde T_1 je částečně rozhodnutelný predikát (dokonce PRP), negace taky.

3.1 10. Hilbertův problém

V moderní terminologii 10. Hilbertův problém zní: "zda existuje algoritmus, který by pro libovolný celočíselný polynom rozhodnul jestli existuje řešení v celých číslech.
Libovolný celočíselný polynom je ekvivalentní 2 polynomům v $\mathbb{N}$, řešení pak taky hledáme v $\mathbb{N}$. Nejprve dáme záporné koeficienty na pravou stranu, pak aplikujeme Lagrangeovou větu o 4 čtvercích.

Věta 3.12 (RDPM (BD)). *Predikát Q je rekurzivně spočetný $\iff$ je tzv diofantický:*

$$\exists x_1, \ldots, x_k \in \mathbb{N} : (p_1(x_1, \ldots, x_k, y_1, \ldots, y_n) = p_2(x_1, \ldots, x_k, y_1, \ldots, y_n))$$

Důsledek 3.13 (10. Hilbertův problém). *10. Hilbertův problém má negativní odpověď. Protože máme množiny které nejsou rozhodnutelné, třeba DIAG.*

Dodatek 3.14. *Jako byprodukt dostáváme ekvivalenci:*

$$\exists (PRP) \iff \exists (p_1(\ldots) = p_2(\ldots))$$

Bez existenčního kvantifikátoru vůbec není pravda. V aritmetice lze vytvořit i superexponencielu e^{n^n} atd, polynomy jsou ale omezené. Taky polynomy lze elementárně vyjádřit pomocí Robinsonovy aritmetiky.

3.2 Selektory

Obecné, Selektor je definovaný pro "hezké relace", např. r.s. $Q(x,y)$. Pak selektor vybírá y pro každé x pokud existuje takové.

Věta 3.15 (O selektoru). *Nechť $Q(x,y)$ je rekurzivně spočetný (resp $Q(x_1,\dots,x_n,y)$), pak $\exists\varphi \in$ ČRF:*

$$\varphi(x)\downarrow \iff \exists y : Q(x,y)$$

$$\varphi(x)\downarrow \Rightarrow Q(x,\varphi(x))$$

Jinými slovy vybere y pokud existuje.

Důkaz. Pozor, nemůžeme vzít nejmenší, musíme vzít první které najdeme, protože lepší už třeba nebude.
Q je r.s. $\Rightarrow$ má index e, napíšeme pomocí univerzálního predikátu

$$\exists s : T_2(e,x,y,s)$$

Predikát zapíšeme jako množinu

$$dom(\varphi_e(x,y))$$

Pak pro dáne x probereme všechny $\langle y,s\rangle$ a hledáme nejmenší dvojici tž platí

$$T_2(e,x,y,s)$$

Neboli hledáme první $\langle y,s\rangle$ tž za s kroků $\varphi_e(x,y)\downarrow$.
Formálně:

$$\varphi(x) \simeq (\mu_{\langle y,s\rangle} T_2(e,x,y,s))_{2,1}$$

□

Definice 3.16 (Graf ČRF).

$$graph(\varphi) = \{\langle x,y\rangle |\ \varphi(x) = y\}$$

Důsledek 3.17 (Graf ČRF). *φ je ČRF $\iff$ $graph(\varphi)$ je r.s.*

Důkaz. "$\Rightarrow$"$\langle x,y\rangle \in graph(\varphi) \Rightarrow \exists s$ (za s kroků $\varphi(x) = y$).
Což je r.s. predikát.
"$\Leftarrow$"Aplikuj selektor. Volba v totalitním režimu, buď jeden kandidát nebo nic. □

Poznámka 3.18. Při zobecněních vyčíslitelnosti do vyšších hierarchií, definujme vyčíslitelnost tak, že graf je rozumný.

Věta 3.19 (Postova věta podruhe).

$$Q(x,y) = (x \in M \wedge y = 1) \vee (x \in \overline{M} \wedge y = 0)$$

Což je rekurzivně spočetný predikát. Selektor φ pro Q je ORF. φ je charakteristická funkce množiny M.

3.3 Imunní množiny

Definice 3.20 (Imunní množina). A je *imunní* pokud je nekonečná a neobsahuje nekonečnou rekurzivní spočetnou podmnožinu.

$$\forall W_x \subseteq A : |W_x| < \infty$$

Je nekonečná, ale nemůžeme to efektivně zkontrolovat. Protože veškeré algoritmicky zkontrolovatelné podmnožiny jsou konečné.

Definice 3.21 (Simple). A je *simple* pokud je rekurzivní spočetná a $\overline{A}$ je imunní.

Poznámka 3.22. Postův problém: co je mezí rekurzivními množinami a nerekurzivními r.s. množinami ve smyslu T-převoditelnosti (viz později)?
Definoval Simple, hypersimple, hyper-hyper ... atd až do Maximalní. Ale tato klasifikace neuspěla.

Věta 3.23 (Existence Simple). *Existuje Simple množina.*

Důkaz. Uděláme predikát

$$Q(x,y) \iff y \in W_x \wedge y > 2x$$

je rekurzivně spočetný protože nalezení je r.s. a druhá podmínka taky. Nechť $\varphi \in$ ČRF je selektor pro Q. Pak

$$A = range(\varphi)$$

Podrobněji:

$$W_x \subseteq \overline{A} \Rightarrow W_x \subseteq \{0,\dots,2x\}$$

Neboli $\overline{A}$ neobsahuje nekonečnou r.s. množinu.
$\overline{A}$ nekonečná?
Do $\{0,\dots,2x\}$ mohou přispět nejvýše $W_0,\dots,W_{x-1}$ množiny. Neboli nejvýše x čísel. Pak ale v $\overline{A}$ zůstane nejméně $(x+1)$ čísel, neboli $\overline{A}$ je nekonečná.
Dohromady $\overline{A}$ je imunní.

□

4 Věty o rekurzi

Taky se jmenují věty o pevném bodě. Používá se self-refrenční trik

Věta 4.1 (O rekurzi 1). *Pokud f je ČRF (pro jednoduchost 1 proměnné) $\Rightarrow$ (efektivně z indexů)*

$$\exists a\ \forall x : \varphi_a(x) \simeq \varphi_{f(a)}(x)$$

Jinými slovy: pokud $f(a)\downarrow \Rightarrow \varphi_a$ a $\varphi_{f(a)}$ jsou stejné funkce. Programy nejsou stejné, ale vyčíslují stejnou funkci.
Pokud ale $f(a)\uparrow \Rightarrow \forall x : \varphi_a(x)\uparrow$.

Důkaz.

$$\varphi_{f(s_1(z,z))}(x) \simeq \Psi_2(e,z,x)$$

Protože levá strana je efektivně vyčíslitelná. e je program který počítá levou stranu.

Pak dle s-m-n věty 2.21

$$\varphi_{f(s_1(z,z))}(x) \simeq \Psi_2(e,z,x) \simeq \varphi_{s_1(e,z)}(x)$$

polož $z = e$, dostaneme

$$\varphi_{f(s_1(e,e))}(x) \simeq \varphi_{s_1(e,e)}(x)$$

Neboli

$$a = s_1(e,e)$$

Který program počítá déle?
Program $e = f(s_1(z,z))$, vstup $\langle z, x\rangle$:

1. spočítej $s_1(z,z)$.
2. spočítej $f(s_1(z,z))$.
   který ale nemusí konvergovat
3. $if(f(s_1(z,z))\downarrow)$
   spusť e na vstup x.

Program $a = s_1(e,e)$:

1. dostane x na vstupu, kvůli s-m-n přidá e ke vstupu
2. spustí program e na vstup $\langle e, x\rangle$.
3. spočítej $a = s_1(e,e)$.
   Tady spočítal svůj vlastní index.
4. spočítej $f(s_1(e,e))$ tzn $f(a)$.
   který ale nemusí konvergovat
5. $if(f(s_1(e,e))\downarrow)$ then
   spust $f(a)$ na vstup x.

Takže a počítá déle. □

Věta 4.2 (O rekurzi 2). *Pokud f je ČRF $(n+1)$ proměnných $\Rightarrow$ ORF (dokonce PRF)*

$$\varphi_{h(y_1,\ldots,y_n)}(x) \simeq \varphi_{f(h(y_1,\ldots,y_n,y_1,\ldots,y_n))}(x)$$

Pokud smažeme $y_1,\ldots,y_n$, tak dostaneme právě Větu o rekurzi 1 4.1.
Pevné body efektivně na parametrech.

Důkaz. Analogicky jako důkaz Věty o rekurzi 1 4.1. Jenom aplikujeme s-m-n 2.21 na větší počet parametrů.

$$\varphi_{f(s_{n+1}(z,z,y_1,\ldots,y_n),y_1,\ldots,y_n)}(x) \simeq \Psi_{n+2}(e,z,y_1,\ldots,y_n,x) \simeq \varphi_{s_{n+1}(e,z,y_1,\ldots,y_n)}(x)$$

Pak

$$h(y_1,\ldots,y_n) = s_{n+1}(e,e,y_1,\ldots,y_n)$$

□

Věta 4.3 (O rekurzi ∞). *Pokud f je ČRF $\Rightarrow \exists$ prostá ORF g (dokonce PRF)*

$$\varphi_{g(j)}(x) \simeq \varphi_{f(g(j))}(x)$$

Pak pevných bodů je nekonečno

$$g(0), g(1), \ldots$$

Důkaz.

$$\varphi_{f(s_2(z,z,j))}(x) \simeq \Psi_2(e,z,j,x) \simeq \varphi_{s_2(e,z,j)}(x)$$

Zvolme

$$g(j) = s_2(e,e,j)$$

□

Věta 4.4 (O rekurzi 3). *Pokud $h(x, z_1, \ldots, z_n)$ je ČRF, pak existuje $a \in \mathbb{N}$ t.ž. a je indexem funkce*

$$h(a, z_1, \ldots, z_n)$$

Důkaz.

$$h(x, z_1, \ldots, z_n) \simeq \Psi_{n+1}(e, x, z_1, \ldots, z_n) \simeq \varphi_{s_1(e,x)}(z_1, \ldots, z_n)$$

Pak aplikujeme Větu o rekurzi 4.1 na funkci $s_1(e,x)$. □

Poznámka 4.5. Věty o rekurzi platí nejen pro ČRF ale taky pro jejich definiční obory. Takže

$$f \in \check{C}RF \ \Rightarrow \exists a : W_a = W_{f(a)}$$

Věta 4.6 (Program vlastní kod). *Existuje n_0:*

$$\varphi_{n_0}(w) = n_0 \text{ pro libovolné } w$$

Program který vypíše svůj vlastní kód.

Důkaz. Pořídíme si pomocnou ORF

$$\alpha(n,w) = n$$

použijeme s-m-n 2.21

$$\alpha(n,w) \simeq \varphi_{f(n)}(w)$$

Kde

$$\alpha(n,w) \simeq \Psi(e,n,w) \simeq \varphi_{s_1(e,n)}(w)$$

Tzn. $f(n) = s_1(e,n)$. Stačí použít Větu o rekurzi 4.1 k f

$$\varphi_{f(n_0)}(w) \simeq \varphi_{n_0}(w) = n_0$$

□

Věta 4.7 (Rekurze v indexech (BD)). *Existuje ORF f:*

$$W_{f(y)} = \{f(0), \ldots, f(y-1)\}$$

Důkaz. Hint: hledáme index f kterých je spočetně mnoho. Musíme použít Větu o rekurzi na úrovně indexů. □

Věta 4.8 (Rice podruhe). *Pokud F je netriviální třída ČRF (nebo r.s množin). Není prázdná, nebo nemá všechny. Pak indexová množina*

$$A_F = \{x | \ \varphi_x \in F\}$$

Je nerekurzivní.

Důkaz. Z netriviálity F

$$\exists a \in A \wedge b \in \overline{A}$$

Nechť sporem A je rekurzivní.
Uděláme funkci h t.ž

$$\forall x \in A : h(x) = b$$
$$\forall y \in \overline{A} : h(y) = a$$

Protože A je rekurzivní, tak h je ORF $\Rightarrow$ existuje pevný bod třeba n_0

$$n_0 \in A \Rightarrow h(n_0) \in \overline{A}$$

Z Věty o rekurzi ale víme

$$\varphi_{n_0} = \varphi_{h(n_0)}$$

takže $n_0, h(n_0)$ musí být ve stejné množině.
Z toho A není rekurzivní. □

Věta 4.9 (O rekurzi (BD)). *Nechť $f \in$ ČRF pak:*

$$\exists a \forall x : \varphi_a(x) = \varphi_{f(a)}(x)$$

Důkaz.

$$\varphi_{\varphi_u(u)}(z) \simeq \Psi_2(a, u, z) \overset{\text{s-m-n}}{\simeq} \varphi_{s_1(a,u)}(z) \simeq \varphi_{d(u)}(z) \simeq \varphi_{\varphi_e(u)}(z)$$

Všimneme si, že

$$\varphi_u(x)$$

je *matice* funkcí. Implikace nahoře ukazuje, že její diagonála $\varphi_u(u)$ se rovná řádku φ_e.
Ukážeme, že f permutuje řádky.

$$\varphi_{f \circ \varphi_u(x)}(z) \simeq \varphi_{\beta(u,x)} \overset{\text{s-m-n}}{\simeq} \varphi_{\varphi_{H(u)}(x)}(z)$$

Kde $H(u) = s_1(b, u)$, kde b je index fince β. Z toho u-tý řádek se zobrazí na $H(u)$-tý. Speciálně

$$e \to H(e)$$

Z toho $\varphi_e(H(e))$ je pevný bod. Protože:

$$\varphi_{f \circ \varphi_e(H(e))}(z) \simeq \varphi_{\varphi_{H(e)}(H(e))}(z) \overset{\text{diagonála}}{\simeq} \varphi_{\varphi_e(H(e))}(z)$$

□

5 Produktivní množiny

Definice 5.1 (Produktivní množina). B je *produktivní* pokud

$$\exists \varphi \in ČRF : W_x \subseteq B \Rightarrow (\varphi(x)\downarrow) \wedge \varphi(x) \in B \setminus W_x$$

Jinými slovy: efektivní non-rekurzivní spočetnost. Pokud máme uvnitř množinu W_x tak se nemůže rovnat B. Taky máme stroječek který najde $\varphi(x)$ který leží mimo danou W_x.

Definice 5.2 (Kreativní množina). Množina A je *kreativní* pokud A je rekurzivně spočetná a $\overline{A}$ je produktivní.

Příklad 5.3 (K). $\overline{K}$ je produktivní funkce je id, K je kreativní.
Protože

$$W_x \subseteq \overline{K} \Rightarrow x \in (\overline{K} - W_x)$$

Věta 5.4 (Modifikace K). *Modifikace předchozího příkladu:*
Nechť máme $f \in ORF$ prostá, pak uvažme množinu:

$$A = \{f(x) |\ f(x) \in W_x\}$$

A je kreativní, $\overline{A}$ produktivní s f.

Důkaz. Nechť $W_x \subseteq \overline{A}$, kdyby $f(x) \in W_x$ tak

$$f(x) \in \overline{A}$$

ale dle definice A

$$f(x) \in A$$

Tedy

$$f(x) \notin W_x$$

a jelikož je **prostá** tak

$$f(x) \in \overline{A} - W_x$$

□

Věta 5.5 (Produktivní funkce ORF). *Každá produktivní množina má ORF produktivní funkce.*

Důkaz. Jednoduše dodefinovat ČRF na ORF nejde.
Chceme najít ORF h:

$$W_{h(y)} = \begin{cases} W_y & \text{pro } \varphi(h(y))\downarrow \\ \emptyset & \text{pro } \varphi(h(y))\uparrow \end{cases}, kde\ \varphi \in ČRF\ prod.$$

Formálně:

$$\varphi \circ h \in ORF$$

Kdyby $\varphi(h(y))\uparrow$ tak

$$\Rightarrow W_{h(y)} = \emptyset \subseteq B \Rightarrow \varphi(h(y))\downarrow spor$$

Dal

$$\forall y : W_{h(y)} = W_y$$

Taky
$$W_y \subseteq B \Rightarrow W_{h(y)} \subseteq B$$
Z toho
$$\varphi(h(y)) \in B - W_{h(y)} = B - W_y$$
Hledaná funkce je $\varphi \circ h$.
Získáme funkci h pomoci věty o rekurzi 4.1. Vezmeme pomocnou $f \in ORF$:

$$W_{f(x,y)} = \begin{cases} W_y & \text{pro } \varphi(x)\downarrow \\ \emptyset & \text{pro } \varphi(x)\uparrow \end{cases}$$

f pomoci s-m-n 2.21

$$f(x,y,w) \simeq \alpha(x,y,w)\downarrow \iff w \in W_y \wedge \varphi(x)\downarrow$$

Taky
$$\alpha(x,y,w) \simeq \varphi_{f(x,y)}(w)$$
kde
$$f(x,y) = s_2(a,x,y)$$
kde a je index α. Pak dle věty of rekurze:

$$W_{h(y)} = W_{f(h(y),y)} = \begin{cases} W_y & \text{pro } \varphi_{h(y)}\downarrow \\ \emptyset & \text{pro } \varphi_{h(y)}\uparrow \end{cases}$$

□

Věta 5.6 (Produktivní funkce ORF prostá(BD)). *Každá produktivní množina má dokonce prostou ORF produktivní funkci.*
Dokonce rekurzivní permutace.

Věta 5.7 (Nekonečná množina). *Každá produktivní množina obsahuje nekonečnou r.s. podmnožinu.*

Důkaz. Máme B a $f \in ORF$ produktivní.
Vezmeme takové z_0:
$$W_{z_0} = \emptyset$$
Množinu vytváříme iterativně, vždy na jeden z bodů co máme aplikujeme f a vezmeme sjednocení.
Formálně:
$$W_{g(x)} = W_x \cup \{f(x)\}$$
rekurze

$$h(0) = z_0 \tag{1}$$
$$h(y+1) = f(h(y)) \tag{2}$$

Pak
$$W_{h(y)} = \{f(z_0), \ldots, f(h(y)-1)\}$$
což je y bodů z B. □

Poznámka 5.8. Imunní a produktivní množiny jsou disjunktní pojmy.

Dodatek 5.9. *Jak dlouho lze pokračovat v konstrukci množiny popsané ve Větě o nekonečné množině 5.7?*
Odpověď: pokud to bude efektivní proces neboli aby množiny byly r.s.
Můžeme iterovat $\omega, 2\omega\ldots$ podél tzv rekurzivních ordinálů (viz ordinální číslo v teorii množin).

Lemma 5.10 (Produktivita a $\leq_m$). *A produktivní a $A \leq_m B \Rightarrow B$ je produktivní. Neboli produktivita se zachovává směrem vzhůru při $\leq_m$.*

Důkaz. Máme ORF funkce g z převoditelnosti. Pak nechť $W_x \subseteq B$, najdeme její preimage v A

$$P = g^{-1}(W_x) \subseteq A$$

Z toho že A je kreativní, pomocí kreativní funkce f najdeme bod $f(y) \notin P$, kde y je index P. Zobrazíme pomoci $g(f(y))$, tím dostaneme bod $\in B - W_x$.
Formálně:

$$W_{h(x)} = g^{-1}(W_x) = \{y |\ g(y) \in W_x\}$$

Pak

$$W_x \subseteq B \Rightarrow W_{h(x)} \subseteq A$$

Poslední krok

$$g \circ f \circ g^{-1}(x) \in B - W_x$$

□

Věta 5.11 (Ekvivalence Kreativní). *Nechť M je r.s. množina. Následující tvrzení jsou ekvivalentní:*

(a) M je kreativní $\iff \overline{M}$ produktivní.

(b) M je 1-úplná $\iff \overline{K} \leq_1 \overline{M}$

(c) M je m-úplná $\iff \overline{K} \leq_m \overline{M}$

Každý z pojmů zahrnuje rekurzivní spočetnost.
Ekvivalence mezi totálně různými pojmy. 1-úplnost jako u NP znamená, že je to nejtěžší ze všech r.s. množin při 1-převoditelnosti.

Důkaz. $(b) \Rightarrow (c)$ z vlastnosti 1 a m převoditelnosti.
$(c) \Rightarrow (a)$
Z vlastnosti převoditelnosti

$$K \leq_m M \iff \overline{K} \leq_m \overline{M}$$

pak použijeme lemma lemma 5.10. Víme že $\overline{K}$ je produktivní, takže i $\overline{M}$. Pak dle definice, M je kreativní.
$(a) \Rightarrow (b)$ ($\overline{M}$ produktivní $\Rightarrow \overline{K} \leq_1 \overline{M}$)
Cil

$$W_{h(y)} = \begin{cases} \{f \circ h(y)\} & \text{pro } y \in K \\ \emptyset & \text{pro } y \notin K \end{cases}$$

kde f je ORF prostá, produktivní pro $\overline{M}$.
Konstrukce funkce h

$$W_{g(x,y)} = \begin{cases} \{f(x)\} & \text{pro } y \in K \\ \emptyset & \text{pro } y \notin K \end{cases}$$

g dostaneme pomoci s-m-n věty 2.21:

$$\alpha(x,y,w) \simeq \varphi_{g(x,y)}(w)\downarrow \iff y \in K \wedge w = f(x)$$

Pak použijeme větu o rekurzi

$$W_{h(y)} = W_{g(h(y),y)}$$

Z toho platí

$$y \notin K \Rightarrow W_{h(y)} = \emptyset \subseteq \overline{M} \Rightarrow f \circ h(y) \in \overline{M}$$
$$y \in K \Rightarrow W_{h(y)} = \{f \circ h(y)\}$$

kdyby $f \circ h(y) \in \overline{M}$ tak

$$W_{h(y)} \subseteq \overline{M} \Rightarrow f \circ h(y) \in \overline{M} - W_{h(y)}$$

což je spor.
Neboli

$$f \circ h(y) \in M \Rightarrow \overline{K} \leq_1 \overline{M}$$

□

Důsledek 5.12 ($\overline{K}$ produktivní). *$\overline{K}$ je nejjednodušší produktivní množinou při $\leq_1$ nebo $\leq_m$. Protože všechny produktivní množiny jsou*

$$\{B|\ \overline{K} \leq_m B\}$$

Definice 5.13 (Úplně produktivní). B je úplně produktivní když existuje ORF f tž:

$$f(x) \in B - W_x \vee f(x) \in W_x - B$$

Příklad 5.14. $\overline{K}$ je úplně produktivní dle definice K.

$$x \in \overline{K} - W_x \vee x \in W_x - \overline{K}$$

neboli funkce je *id.*

Věta 5.15 (Úplná produktivita ekvivalence). *B je úplně produktivní $\iff$ B je produktivní.*

Důkaz. $\Rightarrow$ triviálně z definice.
$\Leftarrow$ lze dokázat dvěma způsoby. První je inspekce minulého důkazu. Uděláme

1. $g^{-1}(W_x)$
2. $f(\ldots)$
3. $g \circ f \circ g^{-1}$.

Jen se musí ověřit o 1 disjunkci víc.
Druhý pomoci věty o rekurzi:

$$W_{h(y)} = \begin{cases} \{f \circ h(x)\} & \text{pro } f \circ h(x) \in W_y \\ \emptyset & \text{pro } f \circ h(x) \notin W_y \end{cases}$$

f je ORF produktivní funkce, h dostaneme pomoci věty o rekurzi a s-m-n věty. Pak

$$f \circ h(x) \notin W_y \Rightarrow W_{h(y)} = \emptyset \Rightarrow f \circ h(x) \in B \Rightarrow f \circ h(x) \in B - W_y$$

$$f \circ h(x) \in W_y \Rightarrow W_{h(y)} = \{f \circ h(x)\}$$

Kdyby $f \circ h(x) \in B$ tak

$$\Rightarrow W_{h(y)} \subseteq B \Rightarrow f \circ h(x) \in B - W_{h(y)}$$

to je spor. Z toho

$$f \circ h(x) \in W_y - B$$

□

Definice 5.16 (Totální množina).

$$Tot = \{x| \ \varphi_x \text{ totální}\} = \{x| \ \exists y \varphi_x(y) \downarrow\}$$

Lemma 5.17 (Totalní množina je produktivní). *Totální množina je produktivní.*

Důkaz. Pomoci m-převodu z $\overline{K}$.

$$\varphi_{h(x)}(y) \downarrow \iff x \notin K_j$$

Kde $x \notin K_j$ znamená, že $x \notin K$ za j kroků.

$$x \notin K \iff h(x) \in Tot$$

Pokud x není v K, tak tam nebude za žádný počet kroků. Pak i $h(x)$ je všude definovaná. Jinak

$$x \in K \Rightarrow |dom(\varphi_{h(x)})| < \infty$$

Definiční obor je konečný a rovna se nějakému $\{0, ..., j_0\}$, což je počet kroků za který x vstoupí do K.
Problém ale je, že dostáváme nový program, ale ne zaručeně novou funkci.
Dokážeme silnější tvrzení a konkretně vytvoříme novou funkci. Máme

$$W_y \subseteq Tot$$

uděláme novou $F \in ORF$ která roste rychleji než $\varphi_a : \forall a \in W_y$. Jinými slovy

$$\forall a \in W_y \exists z_0 \forall z \geq z_0 : F(z) \geq \varphi_a(z)$$

F majorizuje $\varphi_a : \forall a \in W_y$.
BUNO: W_y je nekonečná, jinak přidáme nekonečně indexů prázdného programu. Kvůli enumeratoru, můžeme W_y efektivně generovat, neboli vypisovat

$$a_0, a_1, \ldots$$

Pak

$$F(x) = \max_{j \in \{1, \ldots, x\}} (\varphi_{a_j}(x) + 1)$$

□

Důsledek 5.18 (Omezení logiky). *Z věty plyne omezení logiky.*
Vezmeme třeba Peanovu aritmetiku (PA). Můžeme efektivně generovat sentence které PA dokazuje, tudíž lze efektivně generovat ty $a : \varphi_a$ *totální.*
Pak můžeme dle předchozí věty můžeme najít F *která roste rychleji, než cokoliv co PA dokazuje.*
Libovolná efektivně zadaná teorie má jen r.s. množinu dokazatelných sentenci. Pokud z nich vybereme ty, co dokazuji o nějakém programu že je všude definovaný, tak vyrobíme sentenci na kterou daná teorie nestačí.

6 Dvojice množin

Poznámka 6.1. Motivace z logiky: pokud máme rozumnou teorii, tak určuje sentence které dokazuje a sentence které vyvrací. Když teorie je bezesporná, tak množiny jsou disjunktní.

Definice 6.2 (Rekurzivní neoddělitelnost). Disjunktní dvojice množin A, B jsou *rekurzivně neoddělitelné* když neexistuje M t.ž.

$$A \subseteq M \wedge M \cap B = \emptyset (B \subseteq \overline{M})$$

Definice 6.3 (Efektivní neoddělitelnost). Disjunktní dvojice množin A, B jsou *efektivně neoddělitelné* když existuje $f \in \check{C}RF$ t.ž.

$$\forall x, y : A \subseteq W_x \wedge B \subseteq W_y \wedge W_x \cap W_y = \emptyset \Rightarrow f(x,y) \downarrow \notin W_x \cup W_y$$

Efektivně můžeme nejít bod který leží mimo obaly A, B.

Poznámka 6.4. Efektivní neoddělitelnost $\Rightarrow$ rekurzivní neoddělitelnost.

Věta 6.5 (Efektivní neoddělitelné (BD)). *Existují rekurzivně neoddělitelné které nejsou efektivně neoddělitelné.*

Důkaz. Podobná konstrukce jako u Simple. □

Poznámka 6.6. Efektivní neoddělitelnost vždy lze definovat tak, aby $f \in ORF$ (neboli všude definovaná).

Věta 6.7 (Existence efektivní neoddělitelné). *Existuji disjunktní r.s* E, F *které jsou efektivně neoddělitelné.*

Důkaz. Znovu diagonální metoda.
Vezmeme

$$E = \{x | \ \varphi_x(x) \simeq 0\}$$
$$F = \{x | \ \varphi_x(x) \simeq 1\}$$

Na konkretních hodnotách nezáleží, šlo by vzít $i, j \in \mathbb{N} : i \neq j$.
Hned je vidět, že E, F jsou disjunktní a r.s.
Podle s-m-n 2.21 věty existuje PRF taková, že:

$$\varphi_{\alpha(x,y)}(w) = \begin{cases} 1 & w \text{ padne dříve do } W_x \text{ než do } W_y \\ 0 & w \text{ padne dříve do } W_y \text{ než do } W_x \\ \uparrow & w \notin W_x \cup W_x \end{cases}$$

Formálně (dříve než ...)

$$\exists j (T_1(x,w,j) \wedge \forall i \leq j : \neg T_1(y,w,i))$$

Pokud oba dva programy skončí za stejný počet kroků, tak vezmeme libovolný.
Nechť $W_x \supseteq E$ je r.s. obal E, nápodobně $W_y \supseteq F$, které jsou disjunktní.
Uvažme

$$\varphi_{\alpha(x,y)}(\alpha(x,y))$$

Kdyby $\alpha(x,y)$ padlo do W_x, potom zřejmě padne dříve do W_x než do W_y. Pak

$$\varphi_{\alpha(x,y)}(\alpha(x,y)) = 1$$

a tedy by muselo $\alpha(x,y)$ padnout do F. To nelze.
Symetricky $\alpha(x,y)$ nemůže padnout do W_y neboli

$$\alpha(x,y) \notin W_x \cup W_y$$

□

Poznámka 6.8. Paradox lháře: tento výrok je lživý. Používá self-referenci.
Ten ale operuje s pojmem pravdy, který není matematický. Můžeme ale podobný trik provést s např pojmem dokazatelnosti (viz Godelová 1. věta o neúplnosti 7.9).

Definice 6.9 (1-převoditelnost dvojic). Disjunktní dvojice

$$(C,D) \leq_1 (A,B)$$

právě když existuje prostá $f \in ČRF$:

$$\begin{aligned} x \in C &\iff f(x) \in A \\ x \in D &\iff f(x) \in B \\ x \notin C \cup D &\iff f(x) \notin A \cup B \end{aligned}$$

Definice 6.10 (1-úplnost dvojic). Disjunktní dvojice r.s. množin (A,B) je 1-úplná pravě když libovolná disjunktní dvojice r.s. množin (C,D) platí:

$$(C,D) \leq_1 (A,B)$$

Věta 6.11 (Dvojná věta o rekurzi). *Pro libovolné $f,g \in ORF$:*

$$\exists m,n : \varphi_m = \varphi_{f(m,n)},\ \varphi_n = \varphi_{g(m,n)}$$

Obecněji: pro libovolné $f,g \in ORF$, obě $(k+2)$ proměnných, existují $w_1, w_2 \in PRF$:

$$\varphi_{w_1(y_1,\ldots,y_k)} = \varphi_{f(w_1(y_1,\ldots,y_k),w_2(y_1,\ldots,y_k),y_1,\ldots,y_k)}$$

$$\varphi_{w_2(y_1,\ldots,y_k)} = \varphi_{g(w_1(y_1,\ldots,y_k),w_2(y_1,\ldots,y_k),y_1,\ldots,y_k)}$$

Důkaz. Z Věty o rekurzi 4.1 existuje $h \in ORF$:

$$\varphi_{h(y)} = \varphi_{f(h(y),y)}$$

Vezmeme

$$\varphi_{g(h(y),y)}, \exists n : \varphi_n = \varphi_{g(h(n),n)}$$

Položme $m = h(n)$. □

Věta 6.12 (Efektivní neoddělitelné, 1-úplnost). *Disjunktní r.s dvojice množin jsou efektivně neoddělitelné $\iff$ jsou 1-úplné.*

Důkaz. "1-úplnost $\Rightarrow$ efektivní neoddělitelnost".
Nechť (C,D) efektivně neoddělitelné s funkcí f a $(C,D) \leq_1 (A,B)$ s funkcí g, neboli (A,B) je 1-úplná.
Vezmeme vzory disjunktních r.s. obalů $A \subseteq W_x, B \subseteq W_y$:

$$W_{\alpha(x)} = g^{-1}(W_x), W_{\alpha(y)} = g^{-1}(W_y)$$

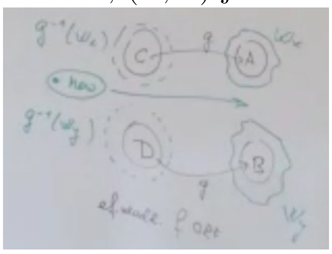
Jelikož, (C,D) efektivně neoddělitelné a platí $W_{\alpha(x)} \cap W_{\alpha(y)} = \emptyset$. Taky funkce f dá bod

$$new = f(\alpha(x), \alpha(y)) : new \notin W_{\alpha(x)} \cup W_{\alpha(y)}$$

Zobrazíme new pomoci g zpátky. Pak

$$g(new) = f \circ g(\alpha(x), \alpha(y)) \notin W_x \cup W_y$$

Z čehož, (A,B) je efektivně neoddělitelná.



"1-úplnost $\Leftarrow$ efektivní neoddělitelnost".
Nechť (A,B) efektivně neoddělitelné s funkcí $f \in \check{C}RF$. Nechť (C,D) libovolné disjunktní r.s. množiny.
Sestavíme množiny

$$W_{w_1(x)} = \begin{cases} A \cup \{f(w_1(x), w_2(x))\} & \text{pro } x \in D \\ A & \text{pro } x \notin D \end{cases}$$

$$W_{w_2(x)} = \begin{cases} B \cup \{f(w_1(x), w_2(x))\} & \text{pro } x \in C \\ B & \text{pro } x \notin C \end{cases}$$

Které dostaneme pomocí dvojné věty o rekurzi 6.11

$$W_{\alpha(y_1,y_2,x)} = \begin{cases} A \cup \{f(y_1(x), y_2(x))\} & \text{pro } x \in D \\ A & \text{pro } x \notin D \end{cases}$$

$$W_{\beta(y_1,y_2,x)} = \begin{cases} B \cup \{f(y_1(x), y_2(x))\} & \text{pro } x \in C \\ B & \text{pro } x \notin C \end{cases}$$

Zkontrolujeme 2 případy:

$$x \notin C \cup D \Rightarrow W_{w_1(x)} = A, W_{w_2(x)} = B \Rightarrow f(w_1(x), w_2(x)) \notin A \cup B$$

jinak např $x \in C \Rightarrow x \notin D$ protože jsou disjunktní dle předpokladu. Pak

$$W_{w_1(x)} = A, W_{w_2(x)} = B \cup \{f(w_1(x), w_2(x))\}$$

Pokud $f(w_1(x), w_2(x)) \notin A$ tak množiny $W_{w_1(x)}, W_{w_2(x)}$ jsou obaly A, B. Pak z efektivní neoddělitelnosti f by měla vracet nový bod, ležící mimo:

$$f(w_1(x), w_2(x)) \notin W_{w_1(x)} \cup W_{w_2(x)}$$

což je spor s konstrukci $W_{w_2(x)}$, protože $f(w_1(x), w_2(x)) \in W_{w_2(x)}$.
Symetricky pro D.
Dohromady $f(w_1(x), w_2(x))$ 1-převádí (C,D) k (A,B). $\square$

7 Gödelovy věty

Definice 7.1 (Rozumná teorie). Rozumná teorie musí být:

- bezesporná
- axiomatizovatelná
- Základní aritmetické síly (adekvátní)

Definice 7.2 (Axiomatizovatelná teorie). Axiomatizovatelná teorie je právě když množina dokazatelných formulí je r.s.

Způsoby řešení paradoxu v teorii množin:

- intuicionisty/kontruktivisty finitisty - odmítají nekonečno. Jde vybudovat spoustu věcí, ale vznikající teorie je kostrbatá a nepříjemná.

  Již Bolzano upozorňoval, že nekonečno je nevlastní pojem, překročující lidskou existenci. Viz [1]

- Hilbertův formalismus - vybudovat teorie z logiky 1. řádu a aby každé tvrzení šlo jednoznačně dokázat nebo vyvrátit + aby teorie byla bezesporná.

Poznámka 7.3. Tzv. Presburgerova Aritmetika bez násobení je rekurzivní a konzistentní.

Poznámka 7.4. Teorie vyčíslitelnosti dává ekvivalentní pohled na Gödelovy věty. Které původně byly vyjádřené přes jazyk aritmetiky.

Definice 7.5 (Reprezentovatelnost). $f \in ČRF$ je reprezentovatelná v teorii T pokud existuje formule F:

$$f(x) = y \Rightarrow \vdash_T F(\overline{x}, \overline{y})$$

$$\vdash_T F(x,y) \wedge F(x,z) \Rightarrow y = x \text{ (funkční vlastnost, aby nebyla jen relace)}$$

Pokud platí obě podminky a T je bezesporná, tak

$$\{(x,y), \vdash_T F(\overline{x}, \overline{y})\} \text{ graf nějaké funkce} \supseteq f$$

Pozorování 7.6. ČRF jsou reprezentovatelné v libovolné teorii ZAS.
Víme, že pro $\varphi \in ČRF$ graf

$$\{(x,y) : \varphi(x) \simeq y\}$$

je r.s.
Z důsledku RDPM věty 3.14 máme

$$\text{r.s. } \approx \exists (p_1(\dots) = p_2(\dots))$$

Rovnost dvou polynomů je jednodušé formalizovatelná v teorii ZAS.

Věta 7.7 (Vztah $\mathbb{N}$ a T). *Pak pokud máme Σ_1 formule v jazyce ZAS, které jsou pravdivé v $\mathbb{N}$ jsou v T (ZAS) dokazatelné.*

$$\mathbb{N} \models \exists (\dots) \Rightarrow \vdash_T \exists (\dots)$$

Důkaz. Vezmeme např následující formule:

$$\exists x(x+\overline{7}=\overline{17})$$

Od teorie T chceme, aby formuli ověřila.
Pokud v $\mathbb{N}$ je pravdivá nějaká Σ_1 formule, tak existuje tzv. Σ_1 svědek. Což je jedno nebo několik přirozených čísel které splňují formuli po dosazení. Teorie zkontroluje rovnost termu. $\square$

Lemma 7.8 (Disjunktní množiny formule(BD)). *Nechť T je bezesporná, ZAS. Pokud A, B jsou disjunktní r.s. tak existuje Σ_1-formule G:*

$$x \in A \Rightarrow \vdash_T G(\overline{x})$$
$$x \in B \Rightarrow \vdash_T \neg G(\overline{x})$$

Důkaz. Sestavíme formuli:

$$\begin{cases} \varphi(x)=0 & \text{pro } x \in A \\ \varphi(x)=1 & \text{pro } x \in B \end{cases}$$

Pak

$$x \in A \Rightarrow \vdash_T F(\overline{x},\overline{0})$$
$$x \in B \Rightarrow \vdash_T F(\overline{x},\overline{1})$$

Z bezespornosti

$$x \in B \Rightarrow \vdash_T F(\overline{x},\overline{1}) \Rightarrow \vdash_T \neg F(\overline{x},\overline{0})$$

Pak hledaná formule je

$$G(x)=F(x,\overline{0})$$

$\square$

Věta 7.9 (Gödelovy věty). *Jestliže teorie T 1. řádu má základní aritmetickou sílu a je bezesporná, pak:*

1. *množina dokazatelných v T formulí není rekurzivní*
2. *pokud je T navíc axiomatizovatelná, tak existuje uzavřená formule (sentence) F taková, že:*

$$T \nvdash F \wedge T \nvdash \neg F$$

3. *(2. Věta) axiomatizovatelnost + Indukce Σ_1 (stačí i trochu méně) tak v T nelze dokázat její bezespornost (consistency). Formálně:*

$$T \nvdash Con_T$$

kde Con_T je formule vyjádřující konsistence, např

$$\neg\exists proof\,(\overline{0}=\overline{1})$$

Důkaz.

1. Nechť T ZAS, bezesporná. Nechť (A,B) r.s. efektivně neoddělitelné.

$$A_1 = \{x : \vdash_T G(\overline{x})\}$$
$$B_1 = \{x : \vdash_T \neg G(\overline{x})\}$$

   $A \subseteq A_1$ je r.s. obal, podobně $B \subseteq B_1$.

   Jelikož A,B jsou efektivně neoddělitelné $\Rightarrow$ jsou rekurzivně neoddělitelné. Z toho A_1, B_1 nejsou rekurzivní. Kdyby byly rekurzivní, tak by každá z nich rekurzivně oddělovala (A,B).

2. Když přidáme axiomatizovatelnost, tak A_1, B_1 jsou r.s.

   Pak z efektivní neoddělitelnosti efektivně najdeme takové $k \notin A_1 \cup B_1$ že:

$$\nvdash_T G(\overline{k}) \wedge \vdash_T \neg G(\overline{k})$$

3. BD, formalizace, hodně logiky.

Jinými slovy, máme následující:

$$\exists \text{můj důkaz IF existuje kratší důkaz mé negace}$$

A symetricky pro gace. Nedochází k žádnému paradoxu, oproti paradoxu lháře. $\square$

7.1 Kalibrace síly teorie

Poznámka 7.10. Konečná verze Ramsey věty je v PA nedokazatelná.

Poznámka 7.11. PA má sílu

$$\varepsilon_0 = \omega^{\omega^{\cdots}}$$

kde exponent je dlouhý ω.
Zkoumá tuto oblast *proof theory.*

8 Relativní vyčíslitelnost

Zobecnění 1-převoditelnosti na relativní výpočet (s Orákulem).

Definice 8.1 (tt-Převoditelnost). Tzv tt (truth table) převoditelnost znamená, že existuje $f \in ORF$ která vrátí:

$$x \to \begin{cases} n_x & \\ \alpha_x & \text{n-arní booleovskou funkci} \\ y_1, \ldots, y_{x_n} & \text{body} \end{cases}$$

Pro niž platí:

$$C_A(x) = \alpha_x(C_B(y_1), \ldots, C_B(y_{n_x}))$$

Kde C_i je charakteristická funkce.
Neboli

$$x \in A \iff \alpha_x(\ldots) = 1$$

Značení:

$$A \leq_{tt} B$$

Poznámka 8.2. TT převoditelnost musí napřed říct, na které body se bude ptát. Což je omezení.

8.1 Formalizace relativního vypočtu

Existuje několik možností formalizace:

Definice 8.3 (Formalizace relativního vypočtu).

1. TS s orákulem. Přidáme další pásku, kde TS bude umisťovat slova pro dotazy k orákulu. Pak množina B (asi jazyk orákula) je dalším vstupem programu.
2. ČRF. Přidáme charakteristické funkce C_B, kde B je proměnná.
3. Programovací jazyk. Přidáme funkci B, v console se objeví dotaz, jestli slovo patří nebo nepatří do jazyka.

Poznámka 8.4. Relativní výpočet je jedním z druhu paralelizace. Pro konkretní vstup x vzniká tzv výpočtový strom.
Důležité je, že množina konečných větví je r.s. Každou z větví lze charakterizovat pomocí

$$\langle x, v, y, n \rangle$$

Kde x je vstup, v je výstup, y je index konečné množiny kladně zodpovězených orákulem dotazů, n je index množiny negativních dotazů. Přitom

$$D_y \subseteq B, D_n \subseteq \overline{B}$$

Předpokládáme, že jazyk orákula je korektní, neboli

$$D_y \cap D_n = \emptyset$$

Tento přístup formalizace není nejvýhodnější, protože body na které se ptáme netvoří souvislý počátek přirozených čísel.

Úmluva 8.5 (Strings). Nadále pracujeme s konečnými binárními řetízky (string), které značíme buď $\{0,1\}^*$, nebo $2^{<w}$.
Operace:

- konkatenace: $\sigma * \tau$.
- délka: $|\sigma|$
- indexovaní: $\sigma(0) * \sigma(1) * \ldots * \sigma(|\sigma| - 1)$.
- počátek(ostrý): $\alpha \preccurlyeq \beta (\alpha \prec \beta)$.
- počátek množiny: $\alpha \prec B$ což znamená $\alpha \preccurlyeq C_B$.

Definice 8.6 (Částečně rekurzivní funkcionál). Částečně rekurzivní funkcionál je r.s. množina Φ trojic taková, že pokud platí:

$$\langle \sigma, x, y \rangle \in \Phi$$
$$\langle \sigma^*, x, y^* \rangle \in \Phi$$
$$\sigma \preccurlyeq \sigma^*$$

Tak $y = y^*$.
Funkcionál je funkce vyššího řádu, vrací funkce.
Význam: k danému x a s kompatibilní orákulovou informací je výstup jednoznačně určen.

Poznámka 8.7. Přístupy formalizace relativizovaného výpočtu pomoci čtveřic $\langle x, v, y, n\rangle$ a částečně rekurzivního funkcionálu jsou ekvivalentní, což není těžké dokázat.

Příklad 8.8. Program má na vstupu x, na výstup vypíše y s použitím $\alpha \preccurlyeq B$.

Definice 8.9 (ČRF-nál zobrazení). Částečně rekurzívní funkcionál určuje částečné zobrazení:

$$\Phi(\sigma)(x) \simeq y \iff \langle \sigma, x, y\rangle \in \Phi$$
$$\Phi(\tau)(x) \simeq y \iff \text{pro nějaké } \sigma \preccurlyeq \tau : \Phi(\sigma)(x) \simeq y$$
$$\Phi(B)(x) \simeq y \iff \text{pro nějaké } \sigma \preccurlyeq B : \Phi(\sigma)(x) \simeq y$$

Poznámka 8.10. Máme funkcionální term, který aplikujeme na $0,1$ (charakteristickou) funkci. Tím dostaneme funkční term. Aplikace funkčního termu na číselný term může ale nemusí dávat číselnou hodnotu.

Vlastnosti 8.11 (Částečně rekurzívní funkcionál)**.**

1. $\Phi(B)$ je korektně definováno.
2. $\Phi(B)$ je intuitivně efektivně vyčíslitelné pomoci B. Postup: efektivně generuj trojice $\langle \sigma, x, y\rangle$. Pak $\sigma \prec B$? Pokud ano, stop. Jinak pokračuj dal.
3. Výpočetní strom $\to \Phi$ vystihuje pojem efektivní vyčíslitelnosti vzhledem k B.

Definice 8.12 (T-Převoditelnost).

$$A \leq_T B$$

Pokud existuje nějaký ČRFunkcionál Φ:

$$\Phi(B) = A, \forall x (A(x) = \Phi(B)(x))$$

Taky se říká: A je B-rekurzívní, A je rekurzivní vzhledem k B.

Definice 8.13 (T-Převoditelnost pro funkce). φ je B-ČRF pokud

$$\varphi(x) \simeq \Phi(B)(x)$$

Lemma 8.14 (Regularizační funkce). *Existuje ORF (dokonce PRF) ρ regularizační funkce. Splňující:*

1. $W_{\rho(x)} \subseteq W_x$.
2. $W_{\rho(x)}$ *je ČRFunkcionál.*
3. W_x *je ČRFunkcionál* $\Rightarrow W_{\rho(x)} = W_x$.

Důkaz. Není formální důkaz.
Budeme efektivně generovat W_x.

1. foreach($tmp = \langle \sigma, x, y\rangle \in W_x$)
2.   if($W_{\rho(x),s} \cup \{tmp\}$ je regulární)

3. $W_{\rho(x)} = W_{\rho(x)} \cup \{tmp\}$ //add

$\square$

Definice 8.15 (Numerace funkcionálů). $W_{\rho(x)}$ z lemmatu je e-tý ČRF-nál, značíme Φ_e.

$$\Phi_e(B)(x) \simeq y \iff \langle \sigma, x, y \rangle \in W_{\rho(x)} : \sigma \prec B$$

Taky za s kroků: $\Phi_{e,s}(B)(x)$.

Pozorování 8.16 (ČRF-nál r.s.). $\Phi_e(\sigma)(x)\downarrow$ je r.s.
$\Phi_{e,s}(\sigma)(x)\downarrow$ je rekurzívní.
$\Phi_e(B)(X)\downarrow$ je r.s. v B.
$\Phi_{e,s}(B)(X)\downarrow$ je rekurzívní v B.

Věta 8.17 (s-m-n pro Relativní). *Existují ORF (dokonce PRF)* $\overline{s}_m$ *takové, že*

$$\forall B, \forall x_1, \ldots, x_m, y_1, \ldots, y_n : \Phi_e(B)(x_1, \ldots, x_m, y_1, \ldots, y_n) \simeq \Phi_{\overline{s_m}(e, x_1, \ldots, x_m)}(B)(y_1, \ldots, y_n)$$

Důkaz. Nemůžeme rovnou použít standardní s-m-n větu, protože ne každá W_x splňuje funkční vlastnost. Proto potřebujeme Regularizační funkce lemma 8.14.
Pak $\Phi_e(B)(x)$ je univerzální B-ČRF. Pak platí

$$\forall B, \forall x_1, \ldots, x_m, y_1, \ldots, y_n : \Phi_e(B)(x_1, \ldots, x_m, y_1, \ldots, y_n) \simeq \Phi_{\overline{s_m}(e, x_1, \ldots, x_m)}(B)(y_1, \ldots, y_n)$$

Kde $\overline{s_m}$ jsou ORF (dokonce PRF).
Formálně, uděláme ČRF takovou, že

$$\alpha(e, x, w)\downarrow \iff w = \langle \sigma, y, t \rangle : \langle \sigma, \langle x, y \rangle, t \rangle \in W_{\rho(e)}$$

S tím, že

$$\alpha(e, x, w) \simeq \varphi_{s_2(a, e, x)}(w)$$

kde a je index α. Položme

$$\overline{s_2}(e, x) = s_2(a, e, x)$$

Rozbor: s pomocí σ orákulu a vstupu y vypočti t jestliže

$$\Phi_e(\sigma)(\langle x, y \rangle) = t$$

Což znamená (pro jednoduchost pro 2 proměnné):

$$\Phi_e(B)(\langle x, y \rangle) \simeq \Phi_{\overline{s_2}(e, x)}(B)(y)$$

$\square$

Vlastnosti 8.18 (Turingovská převoditelnost)**.**

1. $\leq_T$ je reflexivní, tranzitivní.
2. A rekurzívní $\Rightarrow \forall B : A \leq_T B$. Pokud umíme spočítat A, tak to můžeme udělat s libovolným orákulem bez dotazů.
3. B rekurzívní $\wedge A \leq_T B \Rightarrow A$ je rekurzívní. Pro dotazy k orákulu B použijeme TS který rozhoduje B. Jako vnoření vypočtu v složitosti.

Definice 8.19 (Turingovská ekvivalence).

$$A =_T B \iff A \leq_T \land B \leq_T A$$

Definice 8.20 (T Stupně převoditelnosti).

$$deg_T(A) = \{B : B \leq_T A\}$$

Poznámka 8.21. $\{\varphi_e\}_x$ a $\{\Phi_e(\emptyset)(x)\}$ jsou různá vyjádření právě všech ČRF. Jsou rekurzivně izomorfní: máme efektivní překladač mezi těmito systémy.

Definice 8.22 (B-rekurzívní spočetnost). A je B-r.s. právě když

$$A = dom(\Phi_e(B))$$

Značení 8.23 (e-tá B-r.s. množina).

$$W_e^B = dom(\Phi_e(B))$$

Podobně za s kroků:

$$W_{e,s}^B = dom(\Phi_e(B))$$

Definice 8.24 (T-úplnost). A je T-úplná pravě když je r.s. a platí

$$\forall B \in r.s. : B \leq_T A$$

Taky

$$A <_T B \iff A \leq_T B \land B \nleq_T A$$

8.2 Struktura T-stupňů

Definice 8.25 (T stupně struktura). T-stupně tvoří horní polosvaz (upper semilattice), označuji se $\mathcal{D}(\leq)$.
Nechť a, b třídy ekvivalence v $\mathcal{D}(\leq)$. Pak

$$a \leq b \iff \exists A \in a, \exists B \in b : A \leq_T B$$

Definice 8.26 (Join).

$$A \text{ join } B = A \oplus B = \{2x : x \in A, 2x+1 : x \in B\}$$

Vlastnosti 8.27 (Join).

- $A \leq_T A \oplus B$.
- $B \leq_T A \oplus B$.
- $B \leq_T C \land A \leq_T C \Rightarrow A \oplus B \leq_T C$.

8.3 Relativizace dřívějších výsledků

Pozorování 8.28.

- Postova věta: A je B-rekurzívní $\iff A, \overline{A}$ jsou B-r.s.
- r.s. které mají enumerator jsou efektivně generovatelné. Podobně, B-r.s která má enumerator relativní v B je efektivně generovatelná relativně k B.
- r.s. množiny jsou právě ty, které lze vyjádřit pomoci $\exists$(rekurzívní podmínka). Podobně: B-r.s. množiny jsou právě ty, které lze vyjádřit pomoci $\exists(B-$rekurzívní podmínka).

8.4 Operace skoku

Definice 8.29 (Jump). Skok neboli relativizovaný Halting problém.

$$A' = \{x : \Phi_x(A)(x)\downarrow\} = \{x : x \in W_x^A\}$$

Věta 8.30 (Vlastnosti skoku). *1. A' je r.s.*

2. A' není A-rekurzívní & $\overline{A'}$ není A-r.s.

3. B je A-r.s $\iff B \leq_1 A'$.

4. B je A-r.s & $A \leq_T C \Rightarrow B$ je C-r.s.

5. $A \leq_T B \iff A' \leq_1 B'$

6. $A \equiv_T B \iff A' \equiv_1 B'$
Kde $\equiv$ je znak rekurzívní izomorfie.

Důkaz. 1. z definice, A' je definičním oborem programu $\Phi_x(A)(x)$.

2. Cantorova diagonální metoda. Formálně:

$$\overline{A'} = \{x : x \notin W_x^A\} \Rightarrow \forall x : \overline{A'} \neq W_x^A$$

Pak z relativní Postovy věty 8.28: A' není A-rekurzívní.

3. "$\Leftarrow$". Nechť $B \leq_1 A'$.Pak

$$\exists f \in ORF : x \in B \iff f(x) \in A'$$

Z toho můžeme spočítat $f(x)$ a pokud $f(x) \in A'$ tak $x \in B$. Ale A' je A-r.s. a tedy i B je A-r.s. Což je program pro rozhodnutí B.

"$\Rightarrow$"Pomocí fiktivní proměnné. Sestavíme

$$\alpha(x, y, w)\downarrow \iff y \in W_x$$

Dle s-m-n věty 8.17

$$\alpha(x, y, w) \simeq \varphi_{h(x,y)}(w)$$

Dosadíme za fiktivní proměnnou $w = h(x, y)$. Pak

$$h(x, y) \in K \iff y \in W_x$$

Zvolme $x = x_0$, pak $h(x_0, y)$ 1-převádí W_x na K.
V relativním případě máme W_x^A a A' místo K.

4. Když B je A-r.s. a $A \leq_T C$. Víme:

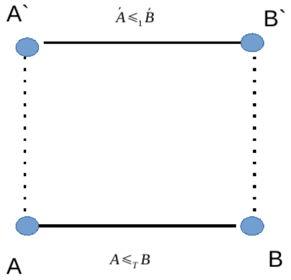
$$B = dom(\Phi_e(A))$$

v průběhu vypočtu se objeví dotazy $z \in A$? Vnoříme pro každý dotaz proceduru, která rozhoduje A pomoci C. Pak

$$B = dom(\Phi_i(C))$$

Pozor, důkaz není formální.

5. Máme následující diagram:



"$\Rightarrow$". Nechť $A \leq_T B$. Víme z 1), že A' je A-r.s. Podle 4) $A \leq_T B \Rightarrow A'$ je B-r.s. Tedy dle 3) protože B' je "nejtěžší"mezí B-r.s (je 1-úplná pro B-r.s):

$$A' \leq_1 B'$$

"$\Leftarrow$". Nechť $A' \leq_1 B'$. Triviálně: $A, \overline{A}$ jsou A-rekurzívní proto dle relativní Postovy věty 8.28: $A, \overline{A}$ jsou A-r.s.

Pak dle 3) A' je 1-úplná pro všechny A-r.s.

$$A, \overline{A} \leq_1 A'$$

Z předpokladu, relace je tranzitivní

$$A, \overline{A} \leq_1 B'$$

Proto $A, \overline{A}$ jsou B-r.s. Konečně dle relativní Postovy věty 8.28 A je rekurzívní v B:

$$A \leq_T B$$

6. Přímý důsledek 5), aplikujeme relaci na obou stranách.

□

Poznámka 8.31.

$$deg_T(A) = \{B : A \equiv_T B\}$$

Se po skoku zobrazí na třídu 1-ekvivalence A'.

Definice 8.32 (Jump na T-stupních). $\underline{a}'$ skok T-stupně $\underline{a}$ je třída

$$\{B : B \equiv_T A', A \in a\}$$

Na volbě A nezáleží protože relace $\equiv_T$ je ekvivalence.

Poznámka 8.33. Skok lze iterovat:

$$(A)^0 = A, (A)^{(n+1)} = (A^{(n)})'$$

Taky všechny konečné:

$$A^{(\omega)} = \{\langle x, y \rangle : x \in A^{(y)}\}$$

Analogicky na třídách

$$\underline{a}^0 = \underline{a}, (\underline{a})^{(n+1)} = (\underline{a}^{(n)})'$$

Pozorování 8.34 ($deg_T(\emptyset)$)**.**
$$\underline{0} = deg_T(\emptyset)$$

Což jsou právě všechny rekurzívní množiny.

Lemma 8.35 (K **a** $\emptyset'$)**.** *K a $\emptyset'$ jsou různá vyjádření Halting problému. Jsou rekurzivně izomorfní: máme efektivní překladač mezi těmito systémy.*

Důkaz. Protože $\emptyset'$ je $\emptyset$-r.s. $\Rightarrow \emptyset'$-r.s. $\Rightarrow \emptyset' \leq_1 K$.
Opačně K je r.s. (absolutně) $\Rightarrow \emptyset$-r.s. $\Rightarrow K \leq_1 \emptyset'$. □

8.5 Stejnoměrnost

Tvrzení o skoku platí stejnoměrně (jako v analýze).

Věta 8.36 (Stejnoměrnost skoku).

$$\exists z_0 \forall A (W_{z_0}^A = A')$$

Existuje pevný program, který funguje pro všechny množiny.

Důkaz.
$$W_{z_0} = \{\langle \sigma, x, y \rangle : \langle \sigma, x, y \rangle \in W_{\rho(x)}\}$$
Protože chceme $\Phi_x(A)(x)$. Pak W_{z_0} je regulární, protože prvky bereme z regulární množiny $W_{\rho(x)}$. Dal

$$x \in A' \iff \Phi_x(A)(x)\downarrow \iff \exists \sigma, \exists y(\langle \sigma, x, y \rangle \in W_{\rho(x)} \wedge \sigma \prec A) \iff x \in W_{z_0}^A$$

Podobně:
$$\exists f \in ORF \forall A, B, \forall z : A = \Phi_z(B) \Rightarrow A' \leq_1 B' \text{ pomoci } \varphi_{f(z)}$$
Kde $\varphi_{f(z)}$ je ORF, prostá. □

9 Limitní vyčíslitelnost

9.1 Limitní vyčíslitelnost pro ORF

Definice 9.1 (Limitní vyčíslitelnost). A je *limitně vyčíslitelná* právě když

$$\exists f \in ORF, \forall x : A(x) = \lim_s f(x, s)$$

Taky je známá jako *efektivní aproximace.*

Poznámka 9.2. V diskretním prostoru, limita znamená že hodnota se stabilizuje (od určitého okamžiku je vždy 0 nebo 1).

Věta 9.3 (Limitní vyčíslitelnost).

$$A \leq_T \emptyset' \iff A \text{ je limitně vyčíslitelná}$$

Věta je jednodušší verzí limitní vyčíslitelnosti, protože pro A máme všude definovanou charakteristickou funkce C_A.

Důkaz. "⇐"Nechť

$$\exists f \in ORF, \forall x : A(x) = \lim_s f(x,s)$$

Hledáme místo, od kterého se hodnota funkce nemění:

$$\mu_s(\forall j \geq s : (f(x,j) = f(x,y)))$$

$\forall j \geq s : f(x,j) = f(x,s)$ je $\emptyset'$-rekurzívní, protože negace je

$$\exists j \geq s : (f(x,j) \neq f(x,s))$$

Vnitřek je rekurzívní + existenční kvantifikátor dává r.s. (while cyklus). R.S. jsou $\leq_1 \emptyset'$, což implikuje taky $\leq_T \emptyset'$. Pokud vrátíme negaci, tak je formule je $\emptyset'$-r.s. (rekurzivita je uzavřená na negaci).
Pak máme

$$\mu_s(\emptyset' - \text{rekurzívní podmínka})$$

Neboli je to $\emptyset'$-ČRF. Je to procedura, na kterou Halting problém umí odpovědět. Nejmenší s hledáme ve while cyklu.
Jelikož $A(x) = C_A(x)$ je všude definovaná, tak existuje příslušná limita:

$$\exists s \forall j \geq s : (f(x,j) = f(x,s))$$

Neboli je to dokonce $\emptyset'$-ORF, z čehož

$$A \leq_T \emptyset'$$

"⇒"Předpokládáme $A \leq_T \emptyset'$, ekvivalentně:

$$\Phi_z(\emptyset')(x) = A(x)$$

Potřebujeme konvergentní posloupnost, která aproximuje A.

$$f(x,s) = \begin{cases} \Phi_{z,s}(\emptyset'_s)(x) & \text{pro } \Phi_{z,s}(\emptyset'_s)(x)\downarrow \\ s+1 & \text{pro jinak} \end{cases}$$

Jelikož $\emptyset'$ je r.s., lze jí generovat po krocích. Takže $\emptyset'_s$ je $\emptyset'$ za s kroků. Zřejmé:

$$\emptyset' = \bigcup \emptyset'_s$$

Pak

$$A'_s = W^A_{z_0,s}$$

Z definice, $f \in ORF$.
Ověříme existence limity. Pro dané x:

$$\Phi_z(\emptyset')(x) = A(x)\downarrow$$

Neboli musí existovat konečný začátek, pomoci kterého počítáme:

$$\exists \sigma \prec \emptyset' \, \Phi_z(\sigma)(x) = A(x)$$

Taky počet kroků je konečný:

$$\exists s_1 : \Phi_{z,s_1}(\sigma)(x) = A(x)$$

σ je konečný začátek $\emptyset'$

$$\sigma \prec \emptyset' \Rightarrow \forall j < |\sigma| : \sigma(j) = 1 \iff j \in \emptyset'$$

Najdeme maximum s_2 takový, že

$$\sigma \prec \emptyset'_{s_2} \wedge \forall j \geq s_2 (\sigma \prec \emptyset'_j)$$

Tedy $\sigma \prec \emptyset'$.
Vezmeme $s \geq \max(s_1, s_2)$, pak platí:

$$f(x,s) = A(x)$$

s_1 nám stačí na generovaní $A(x)$, s_2 zaručí, že σ je korektní aproximace $\emptyset'$. □

Definice 9.4 (Modulus limity (P)).

$$m(x) = \mu_s(\forall j \leq x, \forall t \geq s : f(j,t) = f(j,s))$$

Definice 9.5 (Weak Modulus(P)). Nebo taky *first true moment*. Okamžik kdy aproximativní posloupnost se rovná $A(x)$. Ještě nezaručuje, že nadále bude stabilní.
Až v moment úplného modulu.

Věta 9.6 (Limitní vyčíslitelnost (P)). *Nechť g je všude definovaná funkce, A je r.s.*

$$g \leq_T A \iff g = \lim_s (x,s)$$

Kde modulus limity $\leq_T A$.
Podmínka se jmenuje definite stable.

Důkaz. "$\Leftarrow$"Pokud $m(x) \leq_T A \Rightarrow g \leq_T A$. Plyne z vlastnosti modulu.
"$\Rightarrow$"Inspekce důkazu předchozí věty 9.3.
V průběhu vypočtu se ptáme, jestli σ je již korektní začátek A. Pokud ne, pokračujeme dál. □

9.2 Obecná limitní vyčíslitelnost

Věta 9.7 (Limitní vyčíslitelnost (P)). *1. Pokud $f \in ORF \Rightarrow \lim_s f(x,s)$ je $\emptyset'$-ČRF.*

2. Každá $\emptyset'$-ČRF je limitně vyčíslitelná:

$$\exists h \in ORF : \Phi_z(\emptyset')(x) \simeq \lim_s h(z,x,s)$$

*Aproximace je **uniformní**.*

Důkaz. 1) pokud $g(x) \simeq \lim_s f(x,s)$, tak jako v důkazu 9.3:

$$\mu_s(\forall j \geq s : f(x,j) = f(x,s))$$

Hledáme s od kterého funkce se stabilizuje, a existuje limita. Pak

$$g(x) \simeq f(x, \mu_s(\ldots))$$

Jako i minule je to $\emptyset'$-ČRF. Problém ale je, že s nemusí existovat, pak ale $g(x)\uparrow$.

2) Je potřeba dokazovat opatrněji.

$$h_0(z,x,s) = \begin{cases} \langle \sigma, x, y\rangle & \text{pro } \sigma \prec \emptyset'_s \wedge \Phi_{z,s}(\sigma)(x) = y \\ s+1 & \text{pro jinak} \end{cases}$$

V předchozím důkazu bylo možné dávat přímo hodnotu výstupu a čekat pokud se stabilizuje. Nyní ale čekáme na stabilizaci celého aproximačního výpočtu.
V průběhu výpočtu si ukládáme nejen výstup y ale i jak jsme se k tomu dostali. Pak vydělíme y pokud bude vyhovující:

$$h(z,x,s) = \begin{cases} (h_0(z,x,s))_{3,3} & \text{pro } h_0(z,x,s) = h_0(z,x,s-1) \\ s+1 & \text{pro jinak} \end{cases}$$

Podobně jako v předchozím důkazu najdeme pokud $\Phi_z(\emptyset')(x)\downarrow$, najdeme σ, $\Phi(\sigma)(x)\downarrow$ a $\sigma \prec \emptyset'$ a s_1, s_2 atd.

$$\Phi_z(\emptyset')(x)\downarrow \Rightarrow \exists s_2 \forall j \geq s_2 : \sigma \prec \emptyset'_j$$

Pak pro $s = \max(s_1, s_2)$ se stabilizuje celý výpočet.

$$h_0(z,x,s) = \langle \sigma, x, y\rangle \Rightarrow \lim_s h(z,x,s) = y$$

2) Pokud existuje limita, tak z definice h musí existovat i $\lim_s h_0(z,x,s)$. Pak

$$\lim_s h_0(z,x,s_0) = \langle \sigma, x, y\rangle$$

Pak pro s_0 platí: $\forall j \geq s_0 : \sigma \prec \emptyset'_j \Rightarrow \sigma \prec \emptyset'$. Dohromady:

$$\Phi_z(\emptyset')(x)\downarrow \wedge \Phi_z(\emptyset')(x) = y \wedge \sigma \prec \emptyset'$$

□

Věta 9.8 (Limitní vyčíslitelnost relativní (P)). *1. Pokud $f \in A-ORF \Rightarrow \lim_s f(x,s)$ je A'-ČRF.*

2. Každá A'-ČRF je limitně vyčíslitelná

$$\exists l, \forall s \forall z \forall x : \Phi_z(A')(x) \simeq \lim_s \Phi_l(A)(z,x,s)$$

*Na levé stráně je A-ČRF, na pravé je limita A-ORF. Aproximace je **uniformní**.*

Důkaz. Relativizace předchozího důkazu. □

Důsledek 9.9 (Jump and lim). *Jeden skok definition 8.29 odpovídá 1 limitnímu přechodu.*

Věta 9.10 (Limitní vyčíslitelnost relativní nejobecnější (P)). *Pokud $\exists h \in ORF$:*

$$\Phi_z(\emptyset^{(n+1)})(x) \simeq \lim_{s_0} \dots \lim_{s_n} h(z,x,s_0,\dots,s_n)$$

Poznámka 9.11 (R.S. limitní hierarchie).

- Pokud A je rekurzívní, tak $f(x,0) = f(x,s) = A(x)$. Není potřeba aproximovat, f je ORF.

- A je r.s. tak $f(x,s) = 1 \iff x \in A_s$. BUNO $f(x,0) = 0$. Pak A je tzv 1-r.s., protože v každém sloupci dochází k nejvýš 1 změně. f je ORF.
- 2-r.s. (2 změny) bude rozdílem dvou r.s. množin.
- analogicky n-r.s. je booleovským rozdílem n množin.
- ω-r.s. existuje efektivní odhad počtů změn v každém sloupci. Ekvivalentně: $\exists h \in ORF : \#$ změn $\leq h(x)$. Obecně limita existuje, ale nejde udělat žadný odhad počtu změn ve sloupci.

10 Aritmetická hierarchie

Poznámka 10.1. Aritmetická hierarchie je v jistém smyslu efektivní verzí Borelovské hierarchie. Vezmeme konečně mnoho intervalu, budeme střídat $\cup, \cap$.
$\cup$ odpovídá $\exists$ a $\cap - \forall$.

Poznámka 10.2. Podobná konstrukce jako polynomiální hierarchie v teorii složitosti.

Definice 10.3 (Σ_n, Π_n). Σ_n resp Π_n prefix je skupina n aritmetických kvantifikátorů. Σ_n začíná $\exists$, Π_n naopak $\forall$.
Každá ze skupin je homogenní - několik kvantifikátoru stejného typu, např $\exists\exists\exists$ nebo $\forall\forall$.

Definice 10.4 (Redukovaný prefix). Každá ze skupin obsahuje pouze jeden kvantifikátor.

Příklad 10.5. $\exists\exists\forall\exists\exists$ je Σ_3.
$\exists\forall\exists$ je redukovaný Σ_3.

Poznámka 10.6. Aritmetická hierarchie se označuje Σ^0_n, Π^0_n protože kvantifikace je přes $\mathbb{N}$ (aritmetická). Dolní index označuje počet střídavých kvantifikátoru.
Σ^1_n, Π^1_n by byla kvantifikace navíc přes funkce $f : \mathbb{N} \to \mathbb{N}$.

Definice 10.7 (Predikát $\in \Sigma_n$). Predikát (resp množina) je ve třídě $\Sigma_n(\Pi_n)$ jestliže je vyjadřítelný ve tvaru $\Sigma_n(\Pi_n)$ prefix na rekurzivní základ (ORP).
Podobně pro relativní:
$\Sigma^{0,A}_n$, predikáty jsou A-ORP.

Pozorování 10.8. $\Sigma^0_0 = \Pi^0_0$ jsou právě rekurzivní predikáty.
Rovnost protože nultá úroveň nemá vůbec kvantifikátory, má tedy pouze rekurzivní relace.

Definice 10.9 (Aritmetický predikát). Predikát je *aritmetický* právě když ho lze vyjádřít pomoci logiky 1. řádu, kde atomické části jsou rekurzivní.

Pozorování 10.10. Predikát je aritmetický právě když patří do Σ_n nebo Π_n.

Důkaz. "$\Leftarrow$"zřejmé.
"$\Rightarrow$"Úpravou výrazu do prenexního normálního tvaru. Z logiky, libovolnou formuli lze do tohoto tvaru převést. $\square$

Poznámka 10.11. Pokračováním do rekurzívních ordinálů lze studovat hyperaritmetickou hierarchii. Dál analytická hierarchie.
V rámci kurzu končíme u konečných, neboli ω.

Příklad 10.12. Množina Tot definition 5.16 je v Π^0_2.

Důkaz.

$$x \in Tot \iff \forall y \exists s : \varphi_{x,s}(y)\downarrow$$

Máme 2 střídavé kvantifikátory, $\varphi_{x,s}(y)\downarrow$ je rekurzivní. □

Věta 10.13 (Omezené kvantifikátory). *Omezené kvantifikátory nezvyšují složitost (lze prohodit doprava).*

Důkaz. Rozebereme 2 případy dle typu omezeného kvantifikátoru na začátku formule $(\forall_{x\leq t}, \exists_{x\leq t})$.
BUNO máme formuli v PNF:

$$\forall_{x\leq t}\exists y(\dots)$$

vezmeme w jako kódování $(t+1)$-tice, pak formuli lze ekvivalentně upravit na následující tvar

$$\exists w \forall_{x\leq t}(\dots(w)_{t+1,x}\dots)$$

Protože existence svědka pro všechny $x \leq t$ je stejný jako říct, že existuje skupina $(t+1)$ svědků.
Pak existenční kvantifikátor, BUNO máme formuli:

$$\exists_{x\leq t}\forall y(\dots)$$

Použijeme negaci a předchozí případ:

$$\forall_{x\leq t}\exists y \neg(\dots)$$

$$\exists w \forall_{x\leq t}\neg(\dots(w)_{t+1,x}\dots)$$

Teď odstraníme negaci:

$$\forall w \exists_{x\leq t}(\dots)$$

V libovolné formuli můžeme postupem popsaném nahoře posunout omezené kvantifikátory doprava. Pak dle věty o omezené kvantifikace 3.9 ORP a omezený kvantifikátor jsou dohromady ORP. Omezený kvantifikátor lze nahradit konečnou disjunkce/konjunkce. □

Věta 10.14 (Redukovaný prefix). *Libovolnou formuli lze převést do redukovaného prefixu.*

Důkaz. Znovu 2 případy dle typu kvantifikátoru.
Podobně jako ve větě o neomezené kvantifikace 3.10 nahradíme n kvantifikátoru jediným kvantifikátorem n-tice.

$$\exists x \exists y \to \exists w((w)_{2,1}\dots(w)_{2,2})$$

Analogicky pro $\forall$. □

Příklad 10.15. Množina $\text{Rec} = \{x : W_x \text{ je rekurzivní }\} \in \Sigma^0_3$.

Důkaz. Dle Postovy věty 2.15:

$$x \in Rek \iff \exists y(W_x \cup W_y = W \wedge W_x \cap W_y = \emptyset)$$

Neboli $W_y = \overline{W_x}$, dohromady vyplní celý prostor, ale průnik je prázdný.

Přepíšeme formuli:

$$\exists y(\forall z(z \in W_x \cup W_y) \wedge \forall z(z \notin W_x \cap W_y))$$

Po krocích:

$$\exists y(\forall z \exists s(z \in W_{x,s} \cup W_{y,s}) \wedge \forall z \forall s(z \notin W_{x,s} \cap W_{y,s}))$$

Pak šikovně vytáhneme jeden všeobecný kvantifikátor z levé části a 2 všeobecné z pravé části. V posledním kroku vytáhneme existenční kvantifikátor z levé části. Čímž dostaneme

$$\exists \forall \exists (\dots) \in \Sigma_3^0$$

□

Věta 10.16 (Základní vlastnosti hierarchie). *1.* $A \in \Sigma_n \iff \overline{A} \in \Pi_n$

2. $B \in \Sigma_n(\Pi_n) \Rightarrow \forall m > n : B \in \Sigma_m \cup \Pi_m$.

3. $A \leq_m B \wedge B \in \Sigma_n(\Pi_n) \Rightarrow A \in \Sigma_n(\Pi_n)$

Důkaz. 1. Plyne z De Morgan pravidla. Negace mění kvantifikátor na opačný.

2. Přidáme redundantní kvantifikátory přes fiktivní proměnné.

   Pokud jdeme směrem $\Sigma_n \to \Sigma_{n+1}$, tak přidáme kvantifikátor na konec prefixe. Opačně $\Sigma_n \to \Pi_{n+1}$, přidáme kvantifikátor na začátek prefixe.

3. dle definice $\leq_m \exists f \in ORF$:

$$x \in A \iff f(x) \in B$$

   $f(x)$ můžeme jednoduše kvantifikovat.

□

10.1 Numerace

Věta 10.17 (Neexistence Σ_0^0 univerzálního ORP). *Třída $\Sigma_0^0 = \Pi_0^0$ nemá univerzální ORP (rekurzivní numerace).*

Důkaz. Pomoci Cantorovy diagonální metody. Nechť $R(e,x)$ je ORP.
Pak musí platit:

$$\neg R(e,e) = R(a_0,e)$$

položme $a_0 = e$ a dostáváme spor. □

Poznámka 10.18. Univerzální ČRF, neboli univerzální r.s. predikát 3.3 je univerzální Σ_1^0 2 proměnných pro třídu Σ_1^0 1 proměnné.

Věta 10.19 (O numeraci, univerzálním predikátu). *Pro $(n \geq 1)$ třída $\Sigma_n(\Pi_n)$ má univerzální $\Sigma_n(\Pi_n)$ predikát.*
Tedy máme pojem $\Sigma_n(\Pi_n)$-indexu.

Důkaz. Pro Σ_n, Π_n analogicky.
Nechť máme Σ_n predikát. Nechť n liché. Pak je tvaru

$$\exists \forall \dots \exists Q(\dots)$$

Ořízneme poslední existenční kvantifikátor a predikát, dle věty o univerzálním r.s. predikátu 3.3:

$$\exists y_n Q(\ldots, y_n) = \exists y_n T_n(e, \ldots, y_n)$$

Tím dostaneme vyjádření přes univerzální predikát:

$$\exists y_1, \forall y_2, \ldots \exists y_n T_n(e, y_1, y_2, \ldots, y_n)$$

Pokud n je sudé, tak máme predikát:

$$\exists \forall \ldots \forall Q(\ldots)$$

Znovu použijeme negaci na

$$\neg \forall Q(\ldots) = \exists \neg Q(\ldots)$$

Což je r.s., proto se rovná univerzálnímu predikátu:

$$\exists \neg Q(\ldots) = \exists T_n(e, \ldots)$$

zpět negace:

$$\exists T_n(e, \ldots) = \forall y_n \neg T_n(e, \ldots)$$

Dohromady:

$$\exists y_1, \ldots, \forall y_n \neg T_n(e, y_1, \ldots, y_n)$$

□

Poznámka 10.20. Ve třídě složitosti nemáme univerzální polynom, proto $P \neq NP$ problém.

Důsledek 10.21 (Predikat mimo Σ_n^0, Π_n^0). *Pro* $(n \geq 1) \Sigma_n^0 - \Pi_n^0 \neq \emptyset$.

Důkaz. Pro $n = 1$ máme $K \in \Sigma_1^0 - \Pi_1^0$.
Pro ostatní n stejný důkaz. Nechť $U(e, x)$ je univerzální predikát pro Σ_n^0. Kdyby $U(e, e) \in \Pi_n^0 \Rightarrow \neg U(e, e) \in \Sigma_n^0$. Z existenci univerzálního $\neg U(e, e)$ má index i. Dosadíme index, dostaneme spor

$$U(i, i) = \neg U(i, i)$$

Tedy $U(i, i) \notin \Pi_n^0$. □

Definice 10.22 (Δ_n).

$$\Delta_n = \Sigma_n \cap \Pi_n$$

Definice 10.23 (Σ_n^0-úplnost). Pro $n \geq 1, B$ je Σ_n^0-úplná právě když $B \in \Sigma_n^0$ a

$$\forall A \in \Sigma_n^0 : A \leq_1 B$$

Věta 10.24 (O aritmetické hierarchii).
1. *$\emptyset^{(n)}$ je Σ_n^0-úplná pro $(n \geq 1)$.*
2. *A je r.s. v $\emptyset^{(n)} \iff A \in \Sigma_{n+1}^0$.*
3. *$A \leq_T \emptyset^{(n)} \iff A \in \Delta_{n+1}$.*

Tato věta propojuje skok definition 8.29 s aritmetickou hierarchií a vyjádřitelnosti v PA.

Důkaz. Indukcí, pro $n = 0$ platí, protože

$$A \text{ je r.s.} \iff A \in \Sigma_1^0$$

1. z vlastnosti operace skoku definition 8.29 je $\emptyset^{(n+1)}$ r.s. v $\emptyset^{(n)}$. Podle 2) $\emptyset^{(n+1)} \in \Sigma^0_{n+1}$. Pak $\emptyset^{(n+1)}$ je Σ^0_{n+1}-úplná.

2. "$\Leftarrow$"Jelikož
$$A \in \Sigma^0_{n+1}$$
A Lze vyjádřit jako
$$\exists\forall \dots Q(\dots)$$
Ořízneme od prvního kvantifikátoru
$$\forall \dots Q(\dots) \in \Pi^0_n$$
Použijeme trik s negací jako ve větě o numeraci 10.19. Čímž dostaneme predikát $P \in \Sigma^0_n$ který je dle i.p. $P \leq_1 \emptyset^{(n)}$. Tedy i $P \leq_T \emptyset^{(n)}$. Dáme zpět negace a dostaneme predikát tvaru
$$\exists(\emptyset^{(n)} \text{ rekurzivní relace})$$
Z toho A je r.s. v $\emptyset^{(n)}$.

"$\Rightarrow$". Lze dokázat dvěma způsoby:

a) Jelikož A je r.s. v $\emptyset^{(n)}$. Tak
$$A = dom(\varphi)$$
Kde φ je $\emptyset^{(n)}$-ČRF. Dále
$$x \in A \iff \varphi(x) \downarrow \Rightarrow \Phi(\emptyset^{(n)})(x) \downarrow$$
Kde $f = \Phi(\emptyset^{(n)})$ je $\emptyset^{(n)}$-ČRF. Jelikož, $\exists y : \Phi(\sigma)(x) \simeq y$ je taky existenční, je to ekvivalentní
$$\exists\sigma\exists y(\Phi(\sigma)(x) \simeq y \wedge \sigma \prec \emptyset^{(n)})$$
Máme následující kvantifikátory:
$$\exists(\exists \wedge \sigma \prec \emptyset^{(n)})$$
Tvrdíme, že $(\sigma \prec \emptyset^{(n)})$ je $\Sigma^0_n \wedge \Pi^0_n$. Protože pro $j \leq |\sigma|$:
$$\sigma(j) = 1 \Rightarrow j \in \emptyset^{(n)}$$
což dle i.p. je Σ^0_n. Opačně:
$$\sigma(j) = 0 \Rightarrow j \notin \emptyset^{(n)}$$
což dle i.p. je Π^0_n.

Dohromady:
$$\exists(\Sigma^0_n \wedge \Pi^0_n)$$
Vytáhneme existenční kvantifikátor
$$\exists(\Pi^0_{n-1} \wedge \Pi^0_n) = \Sigma^0_{n+1}$$

b) Jelikož A je r.s. v $\emptyset^{(n)}$. Tak
$$A = dom(\varphi)$$
Kde φ je $\emptyset^{(n)}$-ČRF.

Dle věty o limitní vyčíslitelnosti 9.3

$$\varphi(x) \simeq \lim_s F(x, s)$$

kde F je $\emptyset^{(n-1)}$-ORF. Pak

$$x \in A \iff \varphi(x) \downarrow = \exists \lim_s F(x, s)$$

Existenci limity lze zapsat:

$$\exists s_0 \forall t \geq s_0 : F(x, t) = F(x, s_0)$$

Protože jsme v diskretním prostoru, limita existuje když hodnota funkce se stabilizuje. Navíc $F(x,t) = F(x, s_0)$ je $\emptyset^{(n-1)}$-ORF. Dle i.p. je Π^0_n a Σ^0_n. Vezmeme jen Π^0_n a dostaneme predikát:

$$\exists\forall(\Pi^0_n) \in \Sigma^0_{n+1}$$

3. "$\Leftarrow$". Podle 2) A i $\overline{A}$ jsou r.s. v $\emptyset^{(n)}$. Tedy dle Postovy Věty 8.28 $A \leq_T \emptyset^{(n)}$.

   "$\Rightarrow$". Pokud $A \leq_T \emptyset^{(n)}$ tak podle indukčního předpokladu $A, \overline{A}$ je r.s. v $\emptyset^{(n)}$. Tedy $A \in \Delta_{n+1}$.

$\square$

Poznámka 10.25. Předchozí věta souvisí s elementární aritmetikou, protože dle RDPM věty 3.12:

$$\Sigma^0_1 \models_{\mathbb{N}} \Sigma_1 - \text{formule}$$

Což je taky rovnost dvou polynomů.

Z 1) bodů předchozí věty:

Důsledek 10.26 **($K, \overline{K}$).** *1. $K, \emptyset'$ jsou Σ^0_1-úplné.*

2. $\overline{K}, \overline{\emptyset'}$ jsou 1-úplné neboli Π^0_1-úplné.

Věta 10.27 (Tot úplnost). *Množina Tot definition 5.16 je Π^0_2-úplná.*

Důkaz. $Tot \in \Pi^0_2$, viz example 10.12.
Nechť $B \in \Pi^0_2$ libovolná. Pak

$$x \in B \iff \forall y \exists s : Q(x, y, s)$$

Kde Q je rekurzivní predikát. Uděláme program:

$$\varphi_{\alpha(x)}(y) \simeq \mu_s Q(x, y, s)$$

Pak

$$x \in B \Rightarrow \alpha(x) \in Tot$$

Protože program najde s pro všechna y. Opačně:

$$x \notin B \Rightarrow \alpha(x) \notin Tot$$

Protože $\exists y \forall s : \neg Q(x, y, s)$. Dokonce α lze udělat prostou.
Z toho

$$B \leq_1 Tot$$

$\square$

Definice 10.28 (Fin, Inf).

$$Fin = \{x : |W_x| < \infty\}$$

$$Inf = \{x : |W_x| = \infty\}$$

Věta 10.29 (Fin úplnost). *Množina Fin je Σ^0_2-úplná.*

Důkaz.

$$x \in Fin \iff \exists y \forall z, s(z > y \Rightarrow z \notin W_{z,s})$$

Pokud je konečná, tak od určitého místa do ní nepadne žádný prvek.
Formule je Σ^0_2.
Dokážeme přes komplement

$$\varphi_{\beta(x)}(y) \downarrow \iff \forall j \leq y(\varphi_x(j) \downarrow)$$

Pak

$$x \in Tot \iff \beta(x) \in Inf$$

Protože pokud není nekonečná, tak na nějakém vstupu nekonverguje a totiž není všude definovaná. Taky opačně:

$$x \notin Tot \iff \beta(x) \notin Inf$$

Alternativně:

$$\varphi_{\gamma(x)}(j) \downarrow \iff \exists j - \text{prvků } \in W_x$$

Pak platí

$$x \in Tot \iff \gamma(x) \in Inf$$

a i opačně. Dohromady:

$$Inf \equiv_1 Tot$$

□

Definice 10.30 (Rek).

$$Rek = \{x : W_x \text{ je rekurzivní }\}$$

Věta 10.31 (Rek úplnost (BD)). *Množina Rek je Σ^0_3-úplná.*

11 Pokročilejší vyčíslitelnost

11.1 R.S. množiny

Definice 11.1 (Postův problém podruhe).

$$\exists A : \emptyset <_T A <_T \emptyset'$$

a A je r.s.
Byl vyřešen nezávislé Friedberg-Mučník pomoci tzv. prioritních metod.

1. finite injury $\emptyset'$-priority
2. infinite injury $\emptyset''$
3. $\emptyset''$-priority

11.2 Forcing

Definice 11.2 (Cantor space). Hlavní myšlenka je použít tzv Cantorův prostor 2^ω. Což je prostor všech zobrazení

$$f : \mathbb{N} \to \{0,1\}$$

Je úplný metrický prostor.

Poznámka 11.3. Okolí v Cantorovém prostoru je

$$o_\sigma = \{B : \sigma \prec B\}$$

Pak vzdálenost

$$\rho(\dots) \leq 2^{-|\sigma|}$$

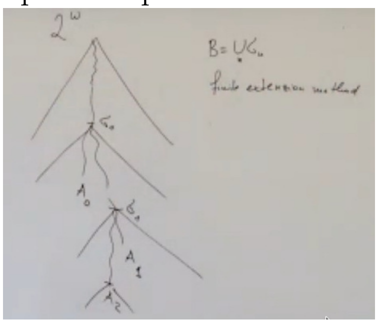
Definice 11.4 (Finite extension method (Cohen)). Souvisí s Cohenovým forcingem v teorii množin kterým vyřešil hypotézu continua.

Věta 11.5 (Bairová věta o kategoriích). *Máme úplný metrický prostor. Pak*

$$\bigcap_{\text{spočetná}} (\textit{otevřené, husté}) \neq \emptyset$$

Jde o ekvivalentní formulace. Viz Baire category theorem.

Důkaz. Hint:
Nechť první bod je A_0 leží v otevřené husté množině. Jelikož je otevřená, existuje okolí A_0 které je uvnitř množiny. Z hustoty v tomto okolí je další bod z další otevřené husté množiny, je taky s okolím atd.
Takovým postupem vytvoříme Cauchy posloupnost, která kvůli úplnosti má limitu. Tato limita leží v průniku.
Speciálně platí v Cantorovém prostoru.



□

Definice 11.6 (1-generická). A je 1-generická když

$$\forall e \exists \sigma (\sigma \prec A \wedge (\Phi_e(\sigma)(e) \downarrow \vee (\forall \tau \geq \sigma : \Phi_e(\tau)(e) \uparrow))$$

Buď s nějakým začátkem konverguje, nebo tzv. silně diverguje (i v okolí).
První podmínka je ekvivalentní

$$\forall \sigma \prec B : \Phi_e(B)(e) \downarrow$$

Druhá je ekvivalentní

$$\forall \sigma \prec B : \Phi_e(B)(e) \uparrow$$

Věta 11.7 (Existence 1-generické). *Existuje 1-generická:*

$$A \leq_T \emptyset'$$

Důkaz. Používá se finite extension method.
Pomoci $\emptyset'$ vytvoříme $\emptyset'$-posloupnost $\{\sigma_n\}_n$:

$$\sigma_n \preccurlyeq \sigma_{n+1}$$

Pak

$$A = \bigcup \sigma_n$$

BUNO: $\sigma_0 = \emptyset$. Indukční krok: máme σ_n chceme další. Zkusíme:

$$\exists \sigma_e \preccurlyeq \tau : (\Phi_e(\tau)(e) \downarrow)$$

Pokud ano, vezmeme první takové a $\sigma_{e+1} = \tau$. Jinak $\sigma_{e+1} = \sigma_e$.
Otázka je 1-kvantifikatorová neboli $\leq_T \emptyset'$. Neboli $\emptyset'$ umí rozhodnout. □

Věta 11.8 (Kleene-Post). *Existuji nerekurzívní $A, B \leq_T \emptyset'$ takové, že A, B jsou T-neporovnatelné.*

Důkaz. Pomoci $\emptyset'$ vytvoříme monotonní $\emptyset'$-posloupnosti:

$$A = \bigcup \{\alpha_n\}_n, B = \bigcup \{\beta\}_n$$

BUNO: $\alpha_0 = \beta_0 = \emptyset$.
Indukční krok: 2 podkroky, protože potřebujeme zajistit

$$\forall e : (A \nleq_T B \simeq \Phi_e(B) \neq A) \wedge (B \nleq_T A \simeq \Phi_e(A) \neq B)$$

Stačí jedná z podmínek, druhá symetricky.
Otázka

$$\exists \beta_e \preccurlyeq \tau : (\Phi_e(\tau)(x_0) \downarrow = 0)$$

kde $x_0 = |\alpha_e|$, $\alpha_e(x_0)$ není definované.
Pokud ANO, tak

$$\alpha_{e+1} = \alpha_e * 1, \beta_{e+1} = \tau$$

Kde $\alpha_e * 1$ je konkatenace. S tím, že τ první které padlo. Pak pro libovolnou $\beta_{e+1} \preccurlyeq B$ platí

$$\Phi_e(B)(x_0) = 0 \wedge A(x_0) = 1 (\alpha_{e+1} \preccurlyeq A)$$

Pokud NE (Jinak):

$$\alpha_{e+1} = \alpha_e * 0, \beta_{e+1} = \beta_e$$

Pak

$$A(x_0) = 0 \wedge (\Phi_e(B)(x_0) \downarrow = 1 \vee \Phi_e(B)(x_0) \uparrow)$$

□

Věta 11.9 (R.S. a 1-generické). *Žádná rekurzívní není 1-generická.*

Důkaz. A je rekurzívní, najdeme e_0 takové, že

$$\Phi_{e_0}(A)(e_0)\uparrow \wedge \forall B \neq A : \Phi_{e_0}(B)(e_0)\downarrow$$

$$\mu_y(B(y) \neq A(y))$$

Pak máme divergenci v množině A ale nikoliv silnou (v okolí konverguje). □

Poznámka 11.10. Těch A, které nejsou 1-generické je málo. Přesněji v topologickém smyslu (2 kategorie) 1-generická jsou "všude".
Odbočka, pro funkcionál $dom(\Phi_e)$ je otevřená. Doplněk je dle definice uzavřená, vnitřek je největší otevřená, takže je ok. Zbývá hranice, a takových je málo.

Definice 11.11 (n-generická). Podobně jako 1-generická, ale místo Σ^0_1 vezmeme Σ^0_n.

Poznámka 11.12. Problém prioritních metod.

Věta 11.13 (Low). *A je 1-generická a $A \leq_T \emptyset' \Rightarrow A$ je tzv. low.*

$$A' \equiv_T \emptyset'$$

Skok libovolné množiny B je $\geq_T \emptyset'$.

Důkaz. □

11.3 Algoritmická náhodnost

Historicky existovaly následující přístupy ve zkoumaní algoritmické náhodnosti:

- frekvenční stabilita, tzv. v limitě v posloupnosti $\{a_n\}$ musí být stejně 0 a 1. Pak ještě musíme modifikovat: pro každou r.s. vybranou část.
- Kolmogorovská složitost (vice variant)
- Teorie míry (Martingale nebo Martin-Löf testy, jsou ekvivalentní).

11.4 Kolmogorovská složitost

Složitost napřed pro konečné řetízky $2^{<\omega}$ pak rozšíření na 2^ω neboli množiny.

Poznámka 11.14 (Kolmogorovské náhodné řetízky). Náhodné jsou nestlačitelné: nejde popsat kratším způsobem než délka.
"Popis"ale není přesně definovaný: polynomiální čas, ČRF, s orákulem atd. Čim máme silnější rozpoznávací mechanizmus, tím více nenáhodných řetízku rozpoznáme.

Zvolíme přístup s ČRF (v literatuře spíš TM).

Definice 11.15 (Kolmogorov complexity).

$$C_f(x) = \min\{|y|, y \in \{0,1\}^* : f(y)\downarrow \wedge f(y) = x\}$$

Délka nejkratšího popisu x pomocí funkce f.

11.5 Martingale

List of Theorems

List of Theorems

Reference

[1] Bernard Bolzano. *Paradoxes of the Infinite (Routledge Revivals)*. Routledge, 2014.