Riceova věta:

jazyk $L_C = \{\langle M\rangle \mid L(M) \in C\}$ je rozhodnutelný, pokud C je třída část. rozh. jazyků triviální prázdná nebo úplná.

$\left[\frac{a}{a}\right] \ \forall a \in \Sigma$

$\left[\frac{qa}{rb}\right] \ \forall \delta(q,a) = (r,b,N)$

$\left[\frac{qa}{br}\right] \ \forall \delta(q,a) = (r,b,R)$ → zapíšu na pásku b, posunu se doprava

$\left[\frac{cqa}{rcb}\right] \ \forall \delta(q,a) = (r,b,L)$ → posunu se doleva

$\left[\frac{\#}{\#}\right], \left[\frac{\#}{\#\lambda}\right], \left[\frac{\#}{\lambda\#}\right],$

$\left[\frac{aq_1}{q_1}\right], \left[\frac{q_1a}{q_1}\right] \ \forall a$ → mažu konfiguraci

$$P' = \left\{ \left[\frac{*\,t_i}{b_i\,*}\right] \middle| \left[\frac{t_i}{b_i}\right] \in P \right\} \cup \left\{ \left[\frac{*\,t_1}{*\,b_1\,*}\right], \left[\frac{*\,\diamond}{\diamond}\right] \right\}$$

Jediná první kostka, co může začít, je $\left[\frac{*\,t_1}{*\,b_1\,*}\right]$, protože ostatní nemají dvě hvězdičky.

Problém je algoritmicky nerozhodnutelný. $L_u \le_m$ MPCP $\le_m$ PCP

$$t = \left\lfloor \log_2 \frac{\varepsilon \cdot V[m]}{n} \right\rfloor - 1 \qquad \frac{V[i]}{2^t} \sim \frac{V[i]}{\frac{\varepsilon \cdot V[m]}{n} \cdot \frac{1}{2}} = \frac{2 \cdot n \cdot V[i]}{\varepsilon \cdot V[m]}$$

$$C[i] = \left\lfloor \frac{V[i]}{2^t} \right\rfloor$$

$x-1 \le \lfloor x \rfloor \le x$

$$t = \left\lfloor \log_2 \frac{\varepsilon \cdot v[m]}{n} \right\rfloor - 1$$

$$c[i] = \left\lfloor \frac{v[i]}{2^t} \right\rfloor$$

$\mathrm{DIAG} = \{ \langle M \rangle \mid \langle M \rangle \notin L(M) \}$ → DIAG není rozhodnutelný

$L_u = \{ \langle M, x \rangle \mid x \in L(M) \}$

$\langle M \rangle \in \mathrm{DIAG} \iff M \notin L(M) \iff \langle M, \langle M \rangle \rangle \notin L_u$

Mohl bych pomocí L_u rozhodovat DIAG, proto není rozhodnutelný

$\mathrm{HALT} = \{ \langle M, x \rangle \mid M(x)\downarrow \}$

↳ není rozhodnutelný

$T = b_1 \# b_2 \# \dots$

$R = R_1 | R_2 | \dots$

$R_i = C(a[0])C(a[1])\dots$

$$C(v) = \begin{cases} 0 & \text{pokud } v=1 \\ (0|1) & \text{pokud } v=0 \end{cases}$$

pokud $v=1$ → b na stejné pozici musí mít 0

pokud $v=0$ → je mi jedno, co bude mít b na pozici

REG nemůže běžet pod $O((|T|+|R|)^{2-\varepsilon})$ pro žádné $\varepsilon > 0$

Ukažme, že to jde: $|T| = O(nd) = |R|$ → $O((nd)^{2-\varepsilon})$ lze řešit.

Jenže hypotéza na OV říká: $O(n^{2-\delta} d^c)$ nelze pro $\delta, c > 0$, $c=1, \delta=\varepsilon$

nechť $d = \Theta(n)$

hypotéza říká: $O(n^{2-\delta} n^c) = O(n^{2-\delta+c})$ → tohle podle hypotézy nelze pro žádné δ, c

redukce dává: $O((n \cdot n)^{2-\varepsilon}) = O(n^{4-2\varepsilon})$

tzn. pokud

jenže pokud zvolím

$c = 2 - \delta$

mám $O(n^{4-2\delta})$

$\delta = \varepsilon$

mám $O(n^{4-2\varepsilon})$

Tohle by to bylo v rozporu s hypotézou → a tudíž pokud má platit hypotéza, nemůžeme provést REG v čase $O((n+m)^{2-\varepsilon})$ pro lib. $\varepsilon > 0$

Riceova věta:

jazyk $L_C = \{\langle M \rangle \mid L(M) \in C\}$ pro část. rozh. třídu jazyků C,

je rozhodnutelný pouze pokud třída C má triviální podmínku

– je buď prázdná / obsahuje všechny částečně rozhodnutelné jazyky

Nechť je netriviální:

$L_u \le L_C$ nebo $L_u \le \overline{L_C}$

$\emptyset \notin C$: $L_u \le_m L_C$

Mějme nějaký PD jazyk $L \neq \emptyset$. TS N: $L = L(N)$

$L_C = \{\langle M \rangle \mid L(M) \in C\}$

$f(\langle M, x \rangle) = \langle M' \rangle$

pak totiž M' simuluje M_1

$x \in L(M) \Leftrightarrow L(M') \in C \Leftrightarrow f(\langle M, x \rangle) = \langle M' \rangle \in L_C$

nerozhodnutelné: $NONEMPTY = \{\langle M \rangle \mid L(M) \neq \emptyset\}$

$ALL = \{\langle M \rangle \mid L(M) = \Sigma^*$

$TOT = \{\langle M \rangle \mid (\forall y \in \Sigma^*)[f_M(y)\downarrow]\}$

$FIN = \{\langle M \rangle \mid L(M)$ je konečný$\}$

$INF = \{\langle M \rangle \mid L(M)$ je nekonečný$\}$

Savič

$\forall f(n) \geq \log_2 n$: $NSPACE(f(n)) \subseteq SPACE(f^2(n))$

- použiju graf konfigurací K_0^x, hledám cestu $K_0^x \to K_F^x$
- nemůžu použít BFS/DFS, jelikož t je lin. ve $|V|$, což může být $\leq 2^{c_M f(n)}$
  - nechť známe $t = c_M f(n)$
    - délka cesty $\leq 2^t$
  - aby existovala cesta délky t z $K_0^x \to K_F^x$, musí existovat $K_0^x \to K_{t/2}^x \to K_F$ dvě cesty délky $t/2$ → můžu rekurzivně pokračovat, vždy využiju pouze stejnou paměť. pro $t=0$ ověřím pomocí δ fce.
  - v každém kroku musím zkusit všechny možné konfigurace K.
  Hloubka rekurze musí být $O(f(n))$, vždy trvá $O(f(n))$ času.
  Jelikož ale neznám t dopředu, budu to postupně zkoušet inkrementovat t
- pak ještě zkusím ověřit dosažitelnost pro $t+1$?

  mám konfiguraci co mi zabírá $t+1$ políček a dostanu se k ní z K_0

Riceova věta:

$L_C = \{\langle M\rangle \mid L(M) \in C\}$ je nerozhodnutelný, pokud třída PD jazyků je netriviální.

$L_u \leq_m L_C$, $L_u \leq \overline{L_C}$

Nechť není triviální:

Mějme M_1: $L(M_1) \in C$, uvažme totální vyčíslitelnou fci $f(\langle M,x\rangle) = \langle M'\rangle$

$$L(M') = \begin{cases} L(M_1) & x \in L(M) \\ \emptyset & x \notin L(M) \end{cases}$$

Simuluj $M(x)$
if přijal:
  Simuluj $M_1(y)$
    if přijal:
      přijmi
odmítni

$x \in L(M) \Rightarrow L(M') \in C \Rightarrow f(\langle M, x\rangle) = \langle M' \rangle \in L_C$

$x \notin L(M) \Rightarrow L(M') = \emptyset \notin C \Rightarrow f(\langle M, x\rangle) = \langle M' \rangle \notin L_C$

dostali jsme $L_U \leq_m L_C$

NONEMPTY – neprázdný jazyk

ALL – všechna slova

TOT – vždy zastaví

FIN – konečný

INF – nekonečný jazyk

$f: N \to N$ je prostorově konstruovatelná, $f(n) \geq \log_2 n$, pokud $f(1^n)$ zobrazí na bin. repr. $f(n)$ v prostoru $O(f(n))$

Deterministická prostorová hierarchie:

$\forall f$ prost. konst. $\exists A$ jazyk rozhodnutelný v prostoru $O(f(n))$, nikoliv ale v prostoru $o(f(n))$.

$A \in SPACE(O(f))$, $M \in SPACE(o(f))$

A se od M musí lišit v nekonečně slovech, jinak bychom pokryli výjimkami:

Uvažujme řetězce tvaru $\langle M \rangle 10^*$ → popíšeme stroj N pracující v $O(f(n))$, že žádný stroj $A = L(N)$

N simuluje M a invertuje výstup

$A = \{\langle M \rangle \mid M(\langle M \rangle 10^*)$ nepřijme v čase $o(f(n))\}$

$x \in A \Leftrightarrow M(x)$ odmítne

$x \in L(N) \Leftrightarrow M(x)$ odmítne. $\Rightarrow x \in A \Leftrightarrow x \in L(N)$ → $L(N) = A$

Jenže N invertuje M: $x \in L(N) \Leftrightarrow x \notin M$.

Máme spor. M má rozhodovat $A = L(N)$, nicméně $x \in M(x)$ – tedy $x \in A$ ale $x \notin L(N)$.

Savičova věta:

$$NSPACE(f(n)) \subseteq SPACE(f^2(n)) \quad \forall f \geq \log_2 n$$

$|V| \leq 2^{c_M f(n)}$, tak nemůžu použít DFS ani BFS.

$\mathcal{U}_0^x \longrightarrow \mathcal{U}_F^x$ $\quad t = c_M f(n)$

$\longrightarrow$ cesta má nejvýš délku t

$O(f(n))$

Mám celkem $O(f(n))$ vrstev, v každé strávím $O(f(n))$ času $\longrightarrow O(f^2(n))$

$t = 0$ použiju δ toho NTS.. Používám furt jednu paměť.

$f: \mathbb{N} \longrightarrow \mathbb{N}$ je prost. konst. pokud $f(1^n)$ je bin. zápis. čísla n a
$f \geq \log_2 n$ běží v prostoru $O(f(n))$

$f: \mathbb{N} \longrightarrow \mathbb{N}$ je časově konst. pokud $f(1^n)$ je bin. repr. čísla n
$f(n) = \Omega(n \log_2 n)$ a běží v čase $O(f(n))$

$\forall f$ časově konst. existuje A jazyk rozhodnutelný v $O(f(n))$, nikoliv v čase $O(f(n) / \log(f(n)))$

- pokud výpočet doběhne do $f(n) / \log_2 f(n)$, zacyklí.
- vyžadujeme logaritmický odstup od jednotlivých fcí

- 3 stopy (budou vždy zarovnané)
  - pracovní páska
  - aktuální stav + přechodová fce
  - čítač

Deterministické prostorové hierarchie

$\forall f: \mathbb{N} \to \mathbb{N}$ prostorově konstruovatelnou $\exists A$ jazyk rozhodnutelný v $O(f(n))$ prostoru,

ale nikoliv v $o(f(n))$.

Sporem, pomocí diagonalizace.

Uvážme N, který simuluje $\langle M, x \rangle$ v prostoru $O(f(n))$ a invertuje výsledek $M(x)$.

Vstup N bude $\langle M \rangle 10^*$

Alokuje si $O(f(n))$ paměti, bude si pamatovat, kam $M(x)$ sáhl.

Nechť sporem existuje M rozhodující A: $L(M) = A$ v prostoru $o(f(n))$.

Pak máme nějaké n_0: $\underline{c_M \cdot g(n_0)} \leq f(n_0)$, M poběží v prostoru

Takže speciálně

$x \in A \iff x \notin L(M)$

$x \notin L(N) \iff x \in L(M)$ spor s definicí toho stroje N.
takže neexistuje žádný takový malý stroj

$$SPACE(f_1) \subsetneq SPACE(f_2) \quad \text{pro } \forall f_1, f_2 \text{ prost.-konstr.}$$
$$f_1 = o(f_2)$$

$$\forall \varepsilon_1, \varepsilon_2 \quad \varepsilon_1 < \varepsilon_2$$
$$SPACE(n^{\varepsilon_1}) \subsetneq SPACE(n^{\varepsilon_2})$$

$NL \subset$

$\forall f: \mathbb{N} \to \mathbb{N}$ t.ž. $f(n) = \Omega(n \log_2 n)$ je časově konst., pokud $f(1^n)$ je bin. repr. hodnoty $f(n)$

konstruovatelná v čase $O(f(n))$

Deterministická časová hierarchie

$\forall f: \mathbb{N} \to \mathbb{N}$ časově konstr. existuje A jazyk rozhodnutelný v $O(f(n))$, nikoliv ale v $o(f(n)/\log f(n))$

Uvážím N simulující $\langle M, x \rangle$.
Pokud simulace M doběhne do $f(n)/\log f(n)$, znegují odpověď.
Oproti prostoru mám větší problém určit, zdali program opravdu běžel pomalu, nebo jsi ho jen špatně simuloval. Budeme proto vyžadovat logaritmický odstup daných fcí.

Simulovat budu na 3-stopém stroji, který si udržuje zarovnané stopy.

- pracovní páska
- aktuální stav + δ
- čítač

Mějme $f_1, f_2 : \mathbb{N} \to \mathbb{N}$, t. ž. $f_1(n) \in o(f_2(n)/\log_2 f_2(n))$ a f_2 časově konstr., pak platí:

$$TIME(f_1(n)) \subsetneq TIME(f_2(n))$$

$$0 \leq \alpha < \beta \in \mathbb{R}$$

$$TIME(n^\alpha) \subsetneq TIME(n^\beta)$$

$G(\langle M \rangle)$ je přiřazení čísla TS. Každé číslo odpovídá jednomu stroji, jeden stroj může mít nekonečně mnoho čísel.
TS je nicméně spočetná množina.

$$index(w) = |\{u \mid u < w\}| \text{ short-lex}$$

$\delta(q_i, X_j) = (q_k, X_l, 2)$ lze kódovat jako $(i)_B | (j)_B | (k)_B | (l)_B | 2_B$

$C_1 \# C_2 \# \ldots$ pak je kód všech instrukcí TS.
Gödelovo číslo je tedy $index(\langle M \rangle)$

Univerzální Turingův stroj

vstup: $\langle M, x \rangle$

simuluje $M(x)$

1. páska: obsahuje vstup $\langle M, x \rangle$
2. páska: obsah pracovní pásky M
3. páska: obsahuje aktuální stav stroje M

Simulace:

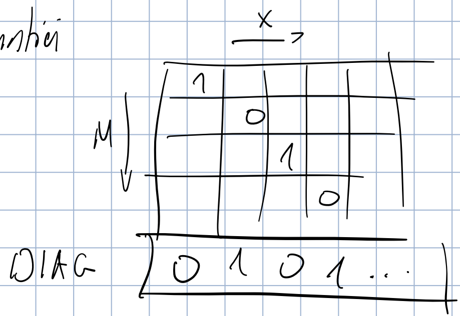
- najdi si instrukci
- zapiš stav
- změň pracovní pásku (X, L)
- přesuň hlavu

Nerozhodnutelnost jazyka L_u

$$L_u = \{\langle M, x \rangle \mid x \in L(M)\}$$

L_u je částečně rozhodnutelný, není ale rozhodnutelný.

Mějme diagonální matici



$$DIAG = \{\langle M \rangle \mid \langle M \rangle \notin L(M)\}$$

Nechť existuje M přijímající $DIAG$: $L(M) = DIAG$

$\langle M \rangle \in DIAG \Rightarrow \langle M \rangle \notin L(M)$

$\Downarrow$ spor

$L(M) = DIAG \Rightarrow \langle M \rangle \in L(M)$

$$\langle M \rangle \in DIAG \Leftrightarrow \langle M \rangle \notin L(M) \Leftrightarrow \langle M, \langle M \rangle \rangle \notin L_u$$

Mohl bych DIAG rozhodnout pomocí L_u. Tudíž L_u není rozhodnutelný.

Godelovo číslo je index(<M>)

$\text{index}(w) = |\{u \mid u < w \text{ v shortlex}\}|$

každé číslo odpovídá jednomu TS

TS má nekonečně mnoho godelových čísel

Máme tedy spočetně mnoho algoritmů

Univerzální TS

na vstupu dostane $\langle M, x \rangle$

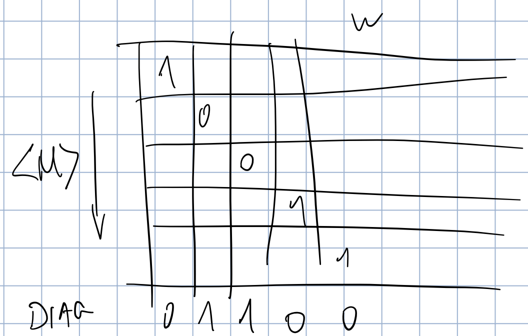
simuluje běh $M(x)$

mám ho na třech páskách

- vstup
- pracovní páska
- páska s aktuálním vstupem

$$L_u = \{\langle M, x\rangle \mid x \in L(M)\}$$

L_u je nerozhodnutelný



$$\text{DIAG} = \{\langle M\rangle \mid \langle M\rangle \notin L(M)\}$$

od každého řádku se liší prvkem na diagonále

DIAG je nerozhodnutelný

Mějme M co přijímá DIAG:

$\langle M\rangle \in \text{DIAG}$ $\xrightarrow{\text{def.}}$ $\langle M\rangle \notin L(M)$

spor

$\langle M\rangle \in \text{DIAG}$ $\xrightarrow{\text{přijímání}}$ $\langle M\rangle \in L(M)$

Nechť M_u rozhoduje L_u. Pomocí M_u bych rozhodnul DIAG.

$$\langle M\rangle \in \text{DIAG} \iff \langle M\rangle \notin L(M) \iff \langle M, \langle M\rangle\rangle \notin L_u$$

HALT

Sestroj stroj M':

Simuluj $M(x)$
pokud odmítl
spusť nekonečnou smyčku

zeptej se $\langle M', x \rangle \in HALT$

takovýto stroj M by pomocí HALT rozhodnul L_u, tedy není rozhodnutelný

$L_u \leq HALT$

RAM <=> TS

Pro každý TS existuje ekvivalentní program napsaný na RAMu simulující jeho výpočet.

Pro každý program napsaný na RAMu existuje ekvivalentní výpočet TS.

TS => RAM

- vstupní pásku napíšu do pole T
  ↳ nebtw mám pásku neomezenou pouze doprava
- přechodovou fci mám schovanou v programu R
  - přechod je posl. podm. příkazů
  - simulace kroku je aktualizace T a stavu

RAM => TS

– musím vymyslet, jak budu reprezentovat paměť pole.

$(i_1)_B | ([r_{i_1}])_B \# (i_2)_B | ([r_{i_2}])_B \dots$

id — hodnota

TS bude mít k pásky:

- vstupní
- výstupní
- pracovní
- paměť RAM

– číslo prováděné instrukce si uložím do stavu

– každá instrukce R je provedení jak. řetězce inst. M

- nalezení registrů
- aritmetické operace
- výpočet δ

pokud $w \in A$, tak existuje posloupnost konf.

$C_0^x \to C_1 \to \dots \to C_F$

kde C_F je konf. v přijímajícím stavu

Cook-Levin:

SAT patří do P, pokud P = NP

SAT je NP-úplný.

$A \leq_m B$ — A je převoditelný na B v polynomiálním čase

B NP-těžký

– $A \leq_m^P B \quad \forall A \in NP$

NP úplný

– pokud je NP-těžký a $\in NP$

SAT: Je formule φ splnitelná?

Převedeme TS na SAT.

Mějme jazyk $A \in \Sigma^*$ v NP.

Mějme Tableau $n^k \times n^k$

– každý řádek popisuje konfiguraci

– konfigurace na dalším řádku je podmíněná δ

L_1, L_2 (část) rozh. jazyky, $L_1 \cdot L_2$, $L_1 \cup L_2$, $L_1 \cap L_2$

Postova věta mi říká:

L rozhodnutelný $\iff$ L i $\bar{L}$ je částečně rozhodnutelný

L je rozhodnutelný $\iff$ $\chi_L(x) = \begin{cases} 1 & x \in L \\ 0 & x \notin L \end{cases}$ je alg. vyčíslitelná

L je (část) rozh. pokud $\exists B$ rozhodnutelný:

$$L = \{x \mid \exists y \in \Sigma^* [\langle x, y \rangle \in B]\}$$

$\forall f: \mathbb{N} \to \mathbb{N}$: $NTIME(f(n)) \subseteq SPACE(g(n))$

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

NTS přijímající jazyk L

má $r = \max\limits_{q \in Q, a \in \Sigma} |\delta(q, a)|$ → max. branching factor

$y = \{1 \ldots r\}^{g|x|}$ → reprezentuje průchod stromem konf.

y_i → větev zvolená v i-tém kroku

M':

for $k = 1$, true, $k{+}{+}$ → dokud všechny větve neodmítly

for $y \in \{1 \ldots r\}^k$

simuluj $M(x)$ s větvením y → ohlaš true, pokud přijal

protože mám $NTIME(f(n))$, tak $k \leq |g(n)|$

$L_1, L_2 \quad L_1 \cap L_2, L_1 \cup L_2, L_1^*, L_1 - L_2$

Postova věta:

L rozh $\iff$ L i $\overline{L}$ jsou část. rozh.

L rozh $\iff$ $\exists\ \chi_L(x) \begin{cases} 1 & x \in L \\ 0 & x \notin L \end{cases}$ alg. vyčíslitelná

L je část. rozh $\iff$ $\exists$ enum. pro L

$L = \{x \mid \exists y \in \Sigma^* [\langle x, y \rangle \in B]\}$ → rozhodnutelný

L je rozh. $\iff$ $\exists$ enum. pro L v shortlex

Základní třídy složitosti: $f: \mathbb{N} \to \mathbb{N}$

$[N]\,TIME(f(n))$

$[N]\,SPACE(f(n))$

$TIME(f(n)) \subseteq SPACE(f(n))$

$TIME(f(n)) \subseteq NTIME(f(n))$

$SPACE(f(n)) \subseteq NSPACE(f(n))$

$P = \bigcup_{k \in \mathbb{N}} TIME(n^k) \qquad NP = \bigcup_{k \in \mathbb{N}} NTIME(n^k)$

$PSPACE = \bigcup_{k \in \mathbb{N}} SPACE(n^k)$

$EXPTIME = \bigcup_{k \in \mathbb{N}} TIME(2^{n^k})$

$L \subseteq NL \subseteq P \subseteq NP \subseteq PSPACE \subseteq NPSPACE \subseteq EXPTIME$

$NTIME(f(n)) \subseteq SPACE(f(n))$

NTS M v čase $O(f(n)) = g(n)$

r --> maximální větvení

$y \in \{1 \ldots r\}^{g(x)}$

TS M'

$k = 1$

repeat → $y = \{1 \ldots r\}^k$ pro všechny

simuluj $M(x)$ s větvením y

$k{+}{+}$

until všechny větvení selhaly

$M'(x)$ skončí do $k \leq |g(x)|$

Vztah prostoru a času

$\forall f(n) \geq \log_2 n$ $\forall L$ jazyk

$L \in NSPACE(f(n)) \Rightarrow (\exists c_L \in \mathbb{N})[L \in TIME(2^{c_L f(n)})]$

speciálně pro $f(n) \geq \log_2 n$, $f(n) = O(g(n))$

$NSPACE(f(n)) \subseteq TIME(2^{cn})$

$|V| \leq |Q| \cdot n \cdot f(n) \cdot |\Sigma|^{f(n)}$

$= 2^{\log Q} \cdot 2^{\log n} \cdot 2^{\log f_n} \cdot 2^{f_n \log |\Sigma|}$

$\leq 2^{f_n \cdot (\log Q + 1 + 1 + \log|\Sigma|)}$ $\quad c_M = \log Q + 1 + 1 + \log \Sigma$

$|V| \leq 2^{c_M f(n)}$ $|E| \leq 2^{2 c_M f(n)}$ Můžeme spustit DFS v $O(|V|^k) = O(2^{k c_M f(n)}) = O(2^{c_L f(n)})$

$[N]TIME(f(n))$

$[N]SPACE(f(n))$

$$TIME(f(n)) \subseteq SPACE(f(n))$$

$$TIME(f(n)) \subseteq NTIME(f(n))$$

$$SPACE(f(n)) \subseteq NSPACE(f(n))$$

$$P = \bigcup_{k \in \mathbb{N}} TIME(n^k), \quad NP = \bigcup_{k \in \mathbb{N}} NTIME(n^k)$$

$$PSPACE = \bigcup_{k \in \mathbb{N}} SPACE(n^k), \quad EXPTIME = \bigcup_{k \in \mathbb{N}} TIME(2^{n^k})$$

$$L \subseteq NL \subseteq P \subseteq NP \subseteq PSPACE \subseteq NPSPACE \subset EXPTIME$$

$$NP = \bigcup_{k \in \mathbb{N}} NTIME(n^k)$$ → nedeterministicky polynomiální

NP je třída jazyků t.ž. každý jazyk má polyn. verifikátor

$L \in NP$ → mám nějaký verifikátor

$M(x)$:

Simuluj verifikátor V

Nedeterministicky zvol znaky y, pokud je V potřebuje.

if V přijme, přijmi / else odmítni

NTS M, $L = L(M)$

nejvíc se větví faktorem r

mám $y \in \{1 \dots r\}^{p(|x|)}$ popisující průchod stromu

y je tedy verifikátorem.

$\forall f: N \to N,\ f \geq \log_2 n\ \forall C$

$$L \in NSPACE(f(n)) \Rightarrow (\exists c_L \in N)[L \in TIME(2^{c_L f(x)})]$$

Speciálně pokud $f(n) = o(g(n))$, tak

$$NSPACE(f(n)) \subsetneq TIME(2^{g(n)})$$

$$V \leq |Q| \cdot n \cdot f(n) \cdot |\Sigma|^{f(n)}$$

$$= 2^{\log |Q|} \cdot 2^{\log n} \cdot 2^{\log_2 f(x)} \cdot 2^{f_n \cdot \log \Sigma}$$

$$= 2^{\log Q + n + f(n) + f(n) \cdot \log \Sigma}$$

$$\leq 2^{\underbrace{f_n(\log Q + 1 + 1 + \log \Sigma)}_{c_N}}$$

$$= 2^{c_N f(n)} \qquad \text{DFS běží v } O(|V|^k) = O(2^{2 c_N f(n)}) = O(2^{c_L f(n)})$$

Poly - převod SAT $\to$ 3-SAT

$$C_i = (x_1 \vee x_2 \vee x_3 \vee x_4 \vee x_5) \simeq$$

$$\alpha_i = (x_1 \vee x_2 \vee l_1) \wedge (\neg l_1 \vee x_3 \vee l_2) \wedge (\neg l_2 \vee x_4 \vee l_3) \wedge (\neg l_3 \vee x_5 \vee y_4)$$
$$\wedge (\neg l_3 \vee x_5 \vee \neg y_4)$$

$$l_1 \wedge (\neg l_1 \vee l_2) \wedge (\neg l_2 \vee l_3) \wedge (\neg l_3)$$
$$l_2 \wedge (\neg l_2 \vee l_3) \wedge (\neg l_3)$$
$$l_3 \wedge (\neg l_3)$$
$$\square \to \text{samo o sobě je to nesplnitelné}$$

Jakmile by mi C_i splní j-tý literál, musím nastavit l_{j+1} nastavit na false

už se nebude snažit ohodnotit ostatní proměnné

$$NP = \bigcup_k NTIME(n^k)$$

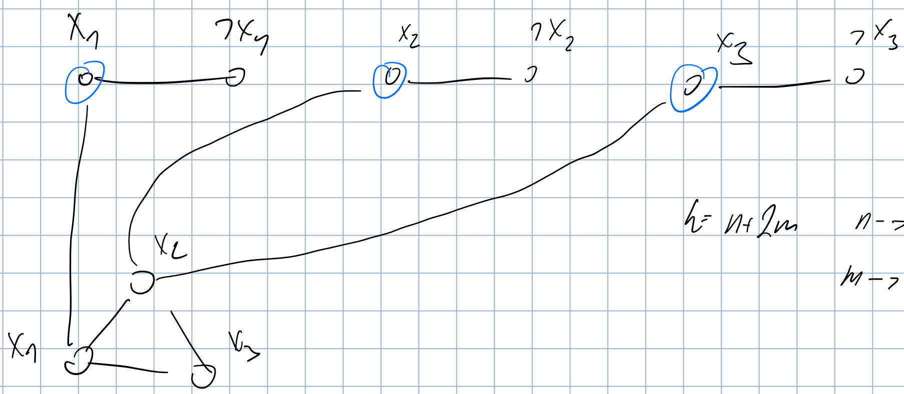
NP – třída jazyků t.ž. ∃ poly. verifikátor V pro daný jazyk

$y \in \{1 - r\}^{p(x)}$ větvení

y je certifikátem

$V(x, y)$

simuluj $M(x)$ s větvením y



$k = n + 2m$ n -> vrcholy
m -> klauzule

$(x_1 \vee x_2 \vee x_3)$

RAM <=> TS

TS => RAM je EZ

RAM => TS je horší

jak reprezentovat paměť?

$(i_1)_B / ([r_{i_1}])_B \# (i_2)_B | ([r_{i_2}])_B \# \dots$ → takto si uložím celou paměť

budu potřebovat násl. pásky
- vstupní
- výstupní
- paměťovou
- mezivýpočtovou

fce programu odkóduju do přechodové fce, aktuální instrukci zakóduju do stavu

Parametrizovaný problém

$L \subseteq \Sigma^* \times \mathbb{N}$ $\quad \langle I, k \rangle \in \Sigma^* \times \mathbb{N}$

k je parametrem instance I

$|\langle I, k \rangle| = |I| + |k|$

L je FTP, pokud lze rozhodnout pomocí A pracující v čase $O(f(k) \cdot |I|^c)$

f.. alg. vyčíslitelnou, c konstanta

kernelizační alg A (kernel) pro problém L:

A je poly time

$\langle I, k \rangle$ a $\langle I', k' \rangle \in A(I, k)$: $\langle I, k \rangle \in L \iff \langle I', k' \rangle \in L$

$\exists g(k)$ vyčíslitelná t.ž. $\forall \langle I, k \rangle$ a $\langle I', k' \rangle \in A$: $|I'| + |k'| \leq g(k)$

$g(k)$ je velikost kernelu, obvykle $k' < k$

Nechť $\langle G, k \rangle$ má VP velikosti k. VP1, VP2 nelze použít.

$|V| \leq k^2 + k \qquad |E| \leq k^2$

VP1 nelze použít -> nemám izolované vrcholy

-> $v \in V \setminus S$ má souseda v S

VP2 nelze použít -> mám max. stupeň $\leq k$

$\rightarrow |V \setminus S| \leq k \cdot |S| \qquad |S| \leq k$

$|V| - k \leq |V \setminus S|$

$|V| \leq |V \setminus S| + k \leq k \cdot |S| + k \leq k^2 + k$

Mám $|E|$ hran, každá má vrchol v S. Celkem musím mít nejvýš $k \cdot |S|$ hran $\leq k^2$

VP3 : $k < 0$ nebo $|V| > k^2 + k$ nebo $|E| > k^2$ -> tak neexistuje VP

$L \subseteq \Sigma^* \times \mathbb{N}$ $\langle I, k \rangle$ k je parametrem instance I

$|\langle I, k \rangle| = |I| + |k|$

FPT – fixed parameter tractable

$L \subseteq \Sigma^* \times \mathbb{N} \in FPT$, pokud existuje A rozhodující L

v čase $\mathcal{O}(f(k) \cdot |I|^c)$ → $f(k)$ je alg. vyčíslitelná, c je konstanta

Alg. A je kernel, pokud pro $\langle I, k \rangle \to A(I', k')$: $\langle I, k \rangle \in L \iff \langle I', k' \rangle \in L$

A běží v polynomiálním čase $|I'| + |k'| \leq g(k)$

$L \in FPT \iff L$ má kernel

Mám kernel $\implies$ FPT

Vstup $\langle I, k \rangle$

převedu na $A(\langle I, k \rangle)$ → polynomiální čas

vyřeším $\langle I', k' \rangle$ hrubou silou → $|I'| + |k'| \leq g(k)$

závisí pouze na k → je to FPT

$\mathcal{O}^*(f(k))$

$L \in FPT \implies L$ má kernel

mám A rozhodující $\langle I, k \rangle$ v čase $f(k) \cdot |I|^c$

Provedu nejvýš $|I|^{c+1}$ kroků A

- přijal dříve: $\langle I', k' \rangle$ triviální instance L

- odmítl dříve: $\langle I'', k'' \rangle$ triviální instance NOT L

return $\langle I, k \rangle$ → jaká je velikost? $|I|^{c+1} \leq f(k) \cdot |I|^c$ → $f(k) \geq |I|$

$|\langle I, k \rangle| = |I| + k \leq f(k) + k$ → .

$|V \setminus S| \geq |V| - k \qquad |V| \leq |V \setminus S| + k$

$\max \deg(G) \leq k$

$|S| \leq k$

min tedy nejvýš $k^2 + k$ vrcholů,

k^2 hran

$|V \setminus S| \leq k|S|$

$|V| \leq |V \setminus S| + k \leq k|S| + k \leq k^2 + k$

$|E| \leq k|S| \leq k^2$

$\tau(k)$:= počet uzlů

ℓ : počet listů

$\tau(k) \leq 2\ell - 1$

vyřešil jsem triviální případ, proto stupeň je alespoň 2.

T má nejvýše $T(k)$ listů

$$T(i) = \begin{cases} T(i-1) + T(i-2) & i \geq 2 \\ 1 & \end{cases}$$

$c \cdot \lambda^{k-1} + c\lambda^{k-2} \leq c \cdot \lambda^k$

$\lambda + 1 \leq \lambda^2 \qquad T(k) \leq 1{,}6181^k$

$O(T(k)(m+n))$

Search VC – $O(1{,}6181^k k^2 + n^c)$

$$T(i) = \begin{cases} T(i-1) + T(i-3) \\ 1 \end{cases}$$

$\lambda^{k-1} + \lambda^{k-3} = \lambda^k$

$\lambda^3 = \lambda^2 + 1$

$\lambda \leq 1{,}4656$

$O(1{,}4656^k k^2 + n^c)$

$\#P$, $\#P$-úplnost

$f: \Sigma^* \to \mathbb{N} \in \#P$ pokud $\exists V$ polyn. a $\exists p(n)$ polynom:

$$f(x) = |\{y \in \{0,1\}^{p(x)},\ V(x,y) \text{ přijme}\}|$$

$$f(x) = \text{počet přijímajících výpočtů } M(x) \text{ pro NTS } M$$

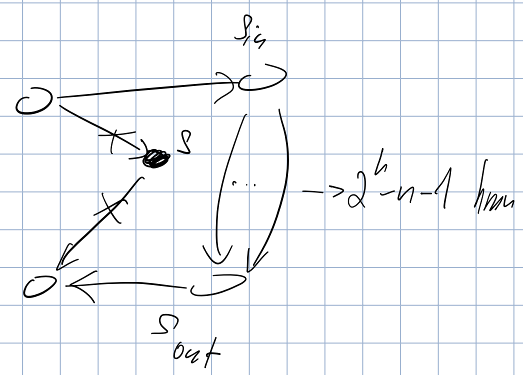
$f(x) > 0$? Pokud $f \in \#P$, pak problém je NP.

$f \leq_p g$ pokud f lze v polyn. čase vyčíslit se znalostí orákula g.

g je $\#P$-úplná, pokud $f \leq_p g$ $\forall f \in \#P$.

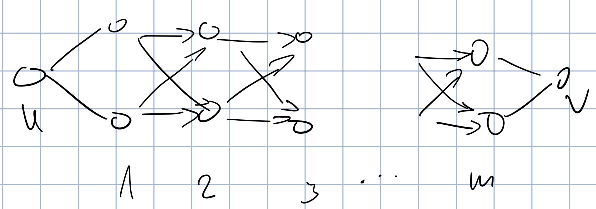
těžkost $\#CYCLE(G)$ HK je NP-úplný

$k = \lceil \log n \rceil$ $2^k - n - 1$



$G \in HK \iff \#CYCLE(G') \geq n^{n^2}$

$m = n \log n$



podgraf je acyklický

cykly G' odpovídají cyklům G

obsahuje 2^m orientovaných cest z u do v.

Pokud mám cyklus v G délky ℓ, má nyní délku $(2^m)^\ell$.

Pokud G má ham. kruž., pak $\#$cyklů v G': $(2^m)^n = (2^{n\log n})^n = (2^{\log_2 n^n})^n = n^{n^2}$

Pokud nemá: $\#$cyklů nejvýše $(2^m)^{n-1} \cdot n^{n-1} = (2^{n\log_2 n} \cdot n)^{n-1} = n^{(n+1)\cdot(n-1)} = n^{n^2-1} < n^{n^2}$

$$T(i) \begin{cases} T(i-1)+T(i-2) & i \geq 2 \\ 1 \end{cases}$$

$\to \lambda + 1 \leq \lambda^2$

$\lambda \leq 1{,}6181$

$O(1{,}6181^k \cdot k^2 + n^c)$

$T(k)$ je počet listí stromu T

$$T(i) \begin{cases} T(i-1)+T(i-3) & i \geq 3 \\ 1 \end{cases}$$

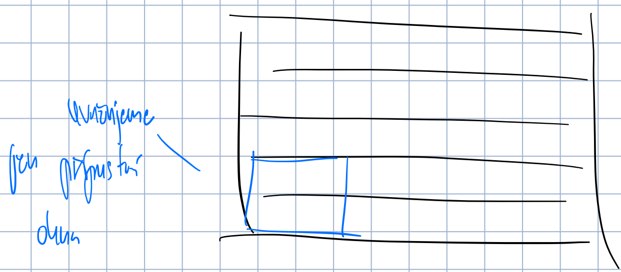
$\lambda^{k-1} + \lambda^{k-3} \leq \lambda^k$

$\lambda^3 = \lambda^2 + 1$

$\lambda \leq 1{,}4656$

$\varphi = \varphi_{start} \wedge \varphi_{cell} \wedge \varphi_{accept} \wedge \varphi_{move}$

tableau



$x_{i,j,s} = T[i,j]=s$

$\# q_0 \, a \, b \, \lambda \lambda \ldots \#$

proměnných mám $(n^k)^2 \cdot |S|$ → lze provést v polyn. čase

$\varphi_{start} = x_{0,0,\#} \wedge x_{0,1,q_0} \wedge x_{0,2,a} \ldots$

$\varphi_{cell} = \bigwedge_{i,j} \left(\bigvee_{a \in \Sigma} x_{ija} \right) \wedge \left(\bigwedge_{a \in \Sigma} \neg x_{ijs} \vee \neg x_{ijt} \right)$

$\varphi_{start} = x_{0,0,\#} \wedge x_{0,1,q_0} \wedge \ldots$

$\varphi_{accept} = \bigvee_{i,j} x_{ijq_1}$ pro q_1 terminální

$\varphi_{accept} = \bigvee_{i,j} x_{i,j,q_1}$ pro q_1 přijímající

$legal_{ij} = \bigvee_{s\text{-př. okno}} \begin{pmatrix} x_{ijs} \wedge x_{ij+1s} \wedge x_{ij+2s} \wedge \\ x_{i+1,j+1,s} \wedge x_{i+1,j,s}, x_{i+1,j+2,s} \end{pmatrix}$

$legal_{ij} = \bigvee_{s\text{-př. okno}} \begin{pmatrix} x_{i,j,s} \wedge x_{i,j+1,s} \wedge x_{i,j+2,s} \wedge \\ x_{i+1,j,s} \wedge x_{i+1,j+1,s} \wedge x_{i+1,j+2,s} \end{pmatrix}$

$\varphi_{move} = \bigwedge_{ij} legal_{ij}$

$\varphi_{move} = \bigwedge_{i,j} legal_{ij}$ → všechny musí být legální

$\varphi_{cell} = \bigwedge_{ij} \left(\bigvee_{s} x_{i,j,s} \right) \wedge \left(\bigwedge_{s \neq t} \neg x_{i,j,s} \vee \neg x_{ijt} \right)$

$\text{co-NP} = \{A \mid \overline{A} \in NP\}$

$A = \{x \mid \forall y \in \{0,1\}^* [V(x,y) \text{ přijme}]\}$

A je co-NP-úplný $\iff$ $\overline{A}$ je NP-úplný

TAUT je co-NP

$\overline{SAT}$ je co-NP

PRG $G: \{0,1\}^n \to \{0,1\}^{\ell(n)}$

$\ell(n) > n$

je vyčíslitelný deterministickým polynomiálním alg.

$\forall$ pravd. alg.

$$\left| \underset{y \in \{0,1\}^{\ell n}}{P} [A(y) = 1] - \underset{y \in \{0,1\}^n}{P} [A(G(y)) = 1] \right| \leq \varepsilon(n)$$

pro zanedbatelnou $\varepsilon(n)$

$\to$ zanedbatelná, pokud $\forall k \in \mathbb{N}\ \exists n_k\ \forall n > n_k\ \ \varepsilon(n) \leq 1/n^k$

Kdyby $P = NP$, tak PRG neexistují

$$I_G = \{y \in \{0,1\}^{\ell(n)} \mid \exists x : G(x) = y\}$$

I_G patří do $NP = P$. Umíme $A(y) = 1 \iff y \in I_G$

$$\left| \underset{y \in \{0,1\}^{\ell(n)}}{P} [A(y) = 1] - \underset{y \in \{0,1\}^n}{P} [A(G(y)) = 1] \right| = \left| 2^{n-\ell(n)} - 1 \right| = 1 - 2^{n-\ell n} \geq 1 - \frac{1}{2} = \frac{1}{2}$$

to není zanedbatelné.

$\mathcal{N}$ po redukci

$$O\left((nd)^{2-c}\right) = O\left(n^{2(2-c)}\right) = O\left(n^{4-2c}\right)$$

$c = 2-\delta$

Jenže OV nemůže být rychlejší jak $O\left(n^{2-\delta} d^{c}\right) \sim O\left(n^{2-\delta} n^{c}\right) = O\left(n^{2-\delta+c}\right) = O\left(n^{4-2\delta}\right)$

takže by to bylo v rozporu s předpokladem o $\mathcal{Q}$

Postova věta:

L je rozhodnutelný $\iff$ $L, \bar{L}$ jsou část. rozh.

$$L \text{ je rozhodnutelný} \iff \exists\, \chi_L(x) \begin{cases} 1 & x \in L \\ \text{algoritmicky vyčíslitelná} & \\ 0 & x \notin L \end{cases}$$

L je část. rozhod. $\iff$ $\exists$ enumerátor jazyka $\to$ rozhodnutelný, pokud enumerátor generuje v shortlex pořadí

$\Rightarrow$

Enumerátor:

generuj $\langle x, y \rangle$ v shortlex pořadí

pokud $\langle x, y \rangle \in B$

zapiš x na výstup

$\Leftarrow$

$M(x)$:

enumeruj x z enumerátoru

jakmile x vypsáno, přijmi

$f: N \to N$ časově konstruovatelná, existuje rozhodnutelný jazyk v $O(f(n))$, ale nerozhodnutelný v $o(f(n)/\log_2 f(n))$

Sestrojím stroj N pracující v $O(f(n))$. Nechť $A = L(N)$

Stroj dostane vstupy tvaru $\langle M \rangle 10^*$. Stroj pak simuluje $M(x)$ a pokud doběhne do času $O(f(n)/\log_2 f(n))$, tak zneguje jeho odpověď.

Vyžadujeme log. odstup od funkcí.

— budu mít třístopou pásku
- pracovní páska
- akt. stav + přechodová fce
- čítač

obsah druhé stopy vždy zachovat pod hlavou při každém kroku

Nechť nyní stroj M rozhoduje jazyk A v čase $o(f(n)/\log_2 f(n)) = g(n)$.
$\exists n_0$ t.ž. $\forall n > n_0$ $n \cdot g(n) \le f(n)$. Ukážeme na vstupu $\langle M \rangle 10^{n_0}$
Výpočet tedy doběhne až do konce, tedy $x \in L(M) \iff x \notin L(N) = A$.
Tedy $A \neq L(N)$. Jenže to byl náš předpoklad, došli jsme ke sporu.

Provede celkem $O(f(n)/\log_2 f(n))$ instrukcí, při každé provede $O(\log_2 f(n))$ na inkrementaci čítače.
Celkem tak provede $O(f(n))$ kroků.

$f: \Sigma^* \to N \in \#P$ pokud $f(x) = \left|\{ y \in \{1,0\}^{p(x)} \mid V(x,y) \text{ přijme} \}\right|$
$\exists V$ verif. $\exists p$ polyn.

$\iff$ počet přijímajících výpočtů stroje M

$f(x) > 0$? $\to$ toto je NP

$f \le_p g$ funkce f je polyn. převoditelná na g pokud lze f vyřešit v polyn. čase se znalostí orákula g.

$g \in \#P$-úplnost pokud $\forall f \in \#P$ $f \le_p g$

#SAT je #P-úplná

$\forall f: \mathbb{N} \to \mathbb{N}$ časově konstruovatelnou existuje jazyk, který je rozhodnutelný v čase $O(f(n))$, ale nikoliv v čase $o(f(n)/\log_2 f(n))$.

Ukážeme stroj N, který bude simulovat stroj $\langle M, x \rangle$. Použijeme diagonalizace, a nakonec dojdeme ke sporu.

Finálně budu mít spor:

Nechť $A = L(N)$ a N běží v čase $O(f(n))$. Nechť sporem $\exists M$, který běží v $o(f(n)/\log_2 f(n))$ a rozhoduje jazyk A.

Simulujme M na stroji N. Vstup bude $\langle M \rangle 1 0^{n_0}$ pro $n_0 : \forall n > n_0 \quad n_0 \cdot g(x) \leq f(x)$

Stroj dosimuluje až dokonce, protože má dost času a tudíž neguje odpověď M.

Mám tedy $x \in L(M) \iff x \notin L(N)$. Tedy $A \neq L(N)$ ↯

to je ale spor s předpokladem.

Detaily:

- musím simulovat na třech stopách: pracovní, stav + δ, čítač
- vyžaduje logaritmický odstup f a g, protože čítač mi zabrá $O(\log_2 f(n))$ bitů a stejně tak dlouho ho trvá posouvat / inkrementovat
- chceme, aby druhá stopa byla vždy zarovnaná s hlavou na začátku každého kroku
- všechny kroky simulace běží v $O(f(n))$.

Důsledek

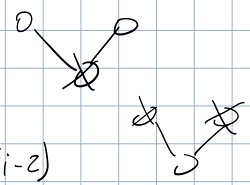
$$f_1(n) \in o(f_2(n)/\log_2 f_2(n)) \Rightarrow TIME(f_1(n)) \subsetneq TIME(f_2(n))$$

$$0 \leq \alpha_1 < \alpha_2 \Rightarrow TIME(n^{\alpha_1}) \subsetneq TIME(n^{\alpha_2})$$

$$P \subsetneq EXPTIME$$

$T(n) \leq 2l-1$ $\quad l$ - počet listů

$T(i-1)$



$T(i-2)$

$$T(i) = \begin{cases} T(i-1) + T(i-2) & i \geq 2 \\ 1 & \text{jinak} \end{cases}$$

$\deg \geq 1$

$$c \cdot \lambda^{n-1} + c\lambda^{n-2} \leq c\lambda^{n} \quad \dots \quad \lambda + 1 = \lambda^2$$

zlatý řez

$$T(n) = \underline{\underline{1{,}6181^{n}}}$$

Mám hranu uv, kde mám $O(n^2)$ vrcholů, hran

Celkem mi trvá najít VP: $O(1{,}6181^{n} n^2 + n^c)$

$$\lambda^{n-1} + \lambda^{n-3} = \lambda^{n} \qquad \lambda^2 + 1 = \lambda^3$$

$$\lambda^{-1} + \lambda^{-3} = 1 \qquad T(n) = 1{,}4656$$

$$\frac{1}{\lambda} = 1 - \frac{1}{\lambda^2}$$

Riceova věta:

Nechť C je třída části. rozhodnutelných jazyků

pak $L_C = \{ \langle M \rangle \mid L(M) \in C \}$ je rozh. pokud C je triviální

Ukážeme, že $L_u \leq_m L_C$, resp. $L_u \leq \overline{L_C}$

Nechť mám $L \in L_C$, $L \neq \emptyset$ $\longrightarrow$ $L(M_1) \in C$, $L(M_1) \neq \emptyset$

Mějme stroj M' se vstupem y

$$L(M') = \begin{cases} L(M_1) & x \in L(M) \\ \emptyset & x \notin L(M) \end{cases}$$

Simuluj $M(x)$

Pokud M přijal

Simuluj $M'(y)$

odmítni

$\longrightarrow$ $x \in L(M) \Rightarrow L(M') = L(M_1) \in C \Rightarrow \langle M' \rangle \in L_C$

$x \notin L(M) \Rightarrow L(M') = \emptyset \notin C \Rightarrow \langle M' \rangle \notin L_C$

HALT problém je část. rozhodnutelný:

$$HALT = \{\langle M, x\rangle \mid M(x)\downarrow\}$$

ukážu $L_u \le HALT$, L_u je část. rozhodnutelný

Vztah prostoru a času

$\forall f(n) \ge \log_2 n, \forall L$ pokud $L \in NSPACE(f(n)) \Rightarrow (\exists c_L \in \mathbb{N})\left[L \in TIME(2^{c_L f(n)})\right]$

$$|V| \le Q \cdot n \cdot f(n) \cdot |\Sigma|^{f(n)}$$

$$= 2^{\log Q} \cdot 2^{\log n} \cdot 2^{\log f(n)} \cdot 2^{f(n) \cdot \log |\Sigma|}$$

$$\le 2^{f(n) \cdot \log(Q+1+1+|\Sigma|)} \qquad c_M = \log Q + 1 + 1 + c$$

$$|V| \le 2^{c_M f(n)}$$

DFS proběhne v čase $O(|V|^k) = O(2^{k c_M f(n)}) = \underline{O(2^{c f(n)})}$

Riceova věta: $L_C = \{\langle M\rangle \mid L(M) \in C\}$

dokážeme $L_u \le L_C$ Mějme $L \in C$, M_1 co rozhoduje L

tedy $L(M_1) = L$

Sestrojíme $L(M') = \begin{cases} L(M_1) & x \in L(M) \\ \emptyset & x \notin L(M) \end{cases}$

Popis M' se vstupem y

Simuluj $M(x)$

pokud M přijal

Simuluj $M_1(y)$

pokud přijal přijmi

odmítni

$x \in L(M) \Leftrightarrow L(M') \in C \Leftrightarrow f(\langle M, x\rangle) = \langle M'\rangle \in L_C$

L_u lze rozhodnout pomocí L_C

$L \in \Sigma^* \times \mathbb{N}$ je param. prob.

$|\langle I, k\rangle| = |I| + |k|$

FPT $\rightarrow$ pokud lze rozhodnout L prob. pomocí alg. A,

který běží v čase $\mathcal{O}(f(k) \cdot |I|^c)$ pro c konst, f alg. vyčíslitelnou.

Kernel je alg. A pro prob. L t.ž. pracuje v polyn. čase,

$\langle I, k\rangle \in L \iff \langle I', k'\rangle \in L$

$|I'| + |k'| \leq g(k) \rightarrow$ je vyčíslitelná

$\hookrightarrow$ takle je velikostí kernelu

$L \in FPT \iff L$ má kernel

mám kernel $\Rightarrow$ chci FPT

$\langle I', k'\rangle = A(\langle I, k\rangle)$

Rozhodni $\langle I', k'\rangle$ hrubou silou $\rightarrow$ složitost závisí pouze na $|I'| + |k'| \leq g(k)$

Tedy slož. alg. je $\mathcal{O}^*(f(k))$ pro vyčísl. fci f

Mám FPT $\Rightarrow$ chci kernel

A řeší prob. L

provedu $|I|^{c+1}$ výpočtů $A(\langle I, k\rangle)$

dostanu-li do $|I|^{c+1}$ přijetí, vrátím jako $\langle I', k'\rangle$ instanci

dostanu-li do $|I|^{c+1}$ odmítnutí, vrátím jako $\langle I'', k''\rangle$ instanci

return $\langle I, k\rangle$

jak je takle velké? $|I|^{c+1} < f(k) \cdot |I|^c$

$f(k) > |I|$

$|\langle I, k\rangle| = |I| + |k| < \underline{\underline{f(k) + k}}$

mám kernel takové velikosti.

$\forall f, \forall L \qquad L \in NSPACE(f(n)) \implies (\exists c_L \in \mathbb{N})[L \in TIME(2^{c_L f(n)})]$

$f \geq \log n$

G neobsahuje izolované vrcholy

$\forall v \in V \setminus S$ má souseda v S

Max deg $\leq k$

$- |V \setminus S| \leq k \cdot |S| \qquad |S| \leq k$

$|V \setminus S| \leq k^2$

$|V| \leq |V \setminus S| + k \leq k^2 + k \to$ každá hrana je pokryta vrcholem z S.

Hran můžu mít $|E| \leq k|S| \leq k^2$

$[N]TIME(f(n)) \qquad TIME(f(n)) \subseteq NTIME(f(n))$

$[N]SPACE(f(n)) \qquad SPACE(f(n)) \subseteq NSPACE(f(n))$

$TIME(f(n)) \subseteq SPACE(f(n))$

$$P = \bigcup_{k \in \mathbb{N}} TIME(n^k)$$

$L \subseteq NL \subseteq P \subseteq NP \subseteq PSPACE \subseteq NPSPACE \subseteq EXPTIME$

$$PSPACE = \bigcup_{k \in \mathbb{N}} SPACE(n^k)$$

$$EXPTIME = \bigcup_{k \in \mathbb{N}} TIME(2^{n^k})$$

OV hypotéza: Nelze řešit OV v $O(n^{2-\delta} \cdot d^c)$ pro všechny $\delta, c > 0$

REG hypotéza Nelze řešit nalezení substringu v $O((n+m)^{2-\varepsilon})$ $\varepsilon \geq 0$

$V(a) = C(a[0])\,C(a[1])\dots$

$C(v) = \begin{cases} 0 & v=1 \\ (0|1) & v=0 \end{cases}$

$R = V(a_1)\,|\,V(a_2)\,|\dots$

$T = b_1 \# b_2 \# \dots$

tímto provedeme převod OV na REG.

Ukážeme, že REG jde řešit rychleji:

$\exists \varepsilon$ $O((n+m)^{2-\varepsilon})$ lze řešit.

nechť $m \in \Theta(n)$

$O(n^{4-2\varepsilon})$, uvažme OV: $d \in O(n)$ $O(n^{2-\delta} n^c)$

mám pak $O(n^{4-2\varepsilon})$ $O(n^{2-\delta+c})$ $c = 2-\delta$

$\delta = \varepsilon$

aby REG běžel rychleji, nesměla by platit OV hypotéza

$\text{co-NP} = \{A \mid \overline{A} \in NP\}$

A je co-NP-úplný pokud $\overline{A}$ NP-úplný

TAUT je co-NP

$\overline{SAT}$ je co-NP

$\overline{SAT} \leq_m TAUT$

$f: \mathbb{N} \to \mathbb{N}$ je čas. konstr. pokud zobrazuje (1^n) na bin. repr. čísla n v čase $O(f(n))$

Savičova věta:

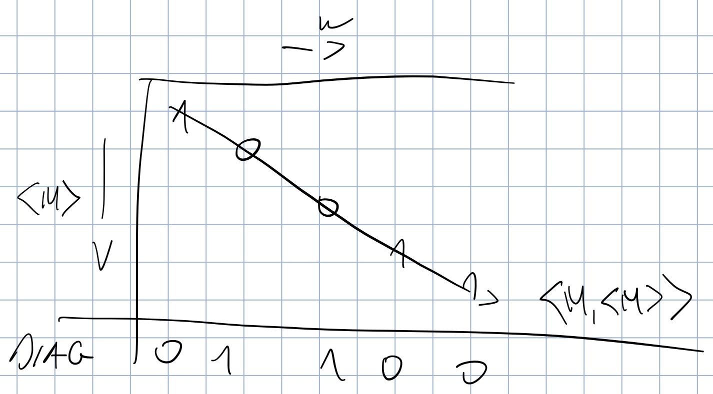
$$\forall f \geq \log n \qquad NSPACE(f(n)) \subseteq SPACE(f^2(n))$$

n^{n-1} cyklů délky $(2^n)^{n-1}$

$$2^{n\log_2 n} \cdot n^{n-1} = n^{(n+1)\cdot(n-1)} = n^{n^2-1} < n^{n^2}$$

$$L_u = \{\langle M, x\rangle \mid x \in L(M)\}$$

L_u je pouze částečně rozhodnutelný.



$$DIAG = \{\langle M\rangle \mid \langle M\rangle \notin L(M)\}$$

Nechť M_D rozhoduje DIAG:

$\langle M_D\rangle \in DIAG$ $\xrightarrow{\text{def.}}$ $\langle M_D\rangle \notin L(M_D)$

$\langle M_D\rangle \in DIAG$ $\xrightarrow{\text{rozhodnost}}$ $DIAG = L(M_D)$ $\quad \langle M_D\rangle \in L(M_D)$

spor

index$(\underset{\frown}{\langle M\rangle})$ $\rightarrow$ index$(w) = |\{u \mid u < w \text{ shortlex}\}|$

$$\delta(q_i, X_j) = (q_k, X_\ell, S) \quad (i)_B \mid (j)_B \mid (k)_B \mid (\ell)_B \mid S$$

$NTIME(f(n)) \subseteq SPACE(f(n))$

$r = \max$ větvení δ $\qquad g(x) = O(f(x))$

$y \in \{1 \dots r\}^{g(x)}$ repr. rozhodnutí v každém větvícím kroku

$k = 1$

for all $y \in \{1 \dots r\}^k$ — každý výpočet skončí do $g(n)$ kroků

Simuluj $M(x)$ s větvením y

pokud přijal, přijmi

opakuj dokud nezamítnuto pro všechny y

Mějme $f: \mathbb{N} \to \mathbb{N}$ prostorově konst. Pro každou takovou f existuje jazyk, který je rozhodnutelný v $O(f(n))$, ale nikoliv v $o(f(n))$.

Použijeme diagonalizace. Nechť mám jazyk A, rozh. v $O(f(n))$, nerozh. v $o(f(n))$.

$A = L(N)$. $\to$ stroj simuluje $M(x)$ následovně:

$$x \in L(N) \iff x \notin L(M)$$ – invertuje výsledek

– ale pouze, pokud simulace nepřekročí limit velikosti.

Nechť mám M' rozhodující A v $o(f(n))$. Rozhodně simulace M v N doběhne.

$x \in L(M) \iff x \notin L(N)$ $\qquad A \neq L(N)$ jenže to je spor.

$L(M) = A$

protože to rozhoduje

$f_1 \in o(f_2)$

$SPACE(f_1) \subsetneq SPACE(f_2)$

$NP = \bigcup_{k \in \mathbb{N}} NTIME(n^k)$ $\qquad NP = \{ L \mid \exists V$ verifikátor polyn. času $\}$

$NP \subseteq NTIME$

$M(x)$

Simuluj verifikátor V

nedeterministicky zvol y, pokud si o něj V žádá

$NTIME \subseteq NP$

$y \in \{1\_r\}^{f(n)}$ --> mi určuje certifikát

Verifikátor $V(x,y)$

Simuluj $M(x)$ s větvením y

přijmi / odmítni podle $M(x)$

totálně vyčíslitelná fce

$$L(M') \begin{cases} L(M_1) & x \in L(M) \\ \emptyset & x \notin L(M) \end{cases}$$

M'

Simuluj $M(x)$

M přijal

Simuluj $M_1(y)$

M_1 přijal

přijmi

odmítni

$\langle M, x\rangle \in L_u \iff x \in L(M) \iff f(\langle M, x\rangle) = \langle M'\rangle \in L_C$

$x \in L(M) \Rightarrow L(M') = L(M_1) \in C \Rightarrow \langle M'\rangle \in L_C$

$x \notin L(M) \Rightarrow L(M') = \emptyset \notin C \Rightarrow \langle M'\rangle \notin L_C$

Umím tedy rozhodnout L_u pomocí L_C.

L_C je tedy alespoň tak těžký jako L_u.

$\forall f \geq \log n \quad NSPACE(f(n)) \subseteq SPACE(f^2(n))$

$SAT \in P$ pokud $NP = P$