3 triedy NP-úplných problémov

- odpoveď ÁNO/NIE – SAT
- nájdeme konkrétneho riešenie – { NIE ; $x_1=\square, x_2=\square$
- optimalizačné úlohy – najlepšie riešenie

ak ex. pol. alg pre jednoduchšiu triedu, tak ex. pol. alg aj pre zložitejšiu triedu

SAT: ÁNO/NIE → konkrét. riešenie

- zistí, či $f(x_1,..,x_n)$ je splniteľná
- ak nie: NIE
- ak áno: zistí či $f'(x_1,..,x_{n-1}) = f(x_1,..,x_{n-1},0)$ je splnit.
  - ak nie: $x_n=1$
  - ak áno: $x_n=0$

  zistí, či $f''(x_1,..,x_{n-2},0,c_n)$ je splniteľná

$$čas \le n\cdot(p'(n)+p(n)) \le p''(n)$$

riešenie splnit. / výpočet fcie, dosadzovanie

farbenie ÁNO/NIE → optimálne riešenie

- n-vrchol. graf sa nedá nafarb. 0 farbami
- n-vrchol. graf sa dá nafarbiť n-farbami

binárne vyhľadávanie $L:=0, R:=n$

$i = \lfloor (L+R)/2 \rfloor$

dá sa graf nafarbiť i farbami? { ÁNO: $R:=i$; NIE: $L:=i$

opakuj, kým $R=L$, posled. krok dá graf s min farbením

čas: $p(n)\ \log_2(n)$

Deterministická simulácia nedeterministic. problémov

V: ak vieme rozhodovací problém vypočítať nedeterm. v čase $p(n)$, vieme ho riešiť determ. v čase $\le c^{p(n)}$

D: majme k-pásk TS, nedeterm.

$\vdash a_1 \dots a_n \dashv$ n teraz pásky čisté

[q] → $\vdash a'_2 \dots$ R/W ; → $\vdash a'_3 \dots$ R/W

inštrukcie ($q, a_1, a'_2, a'_3 \dots$; $a_1, a''_2, a''_3 \dots$, ←, →, 0, ...)

stupeň nedeterminizmu s: max. počet inštrukcií s rovnakými začiatkami. stupeň vetvenia výpočtu

ak je dĺžka výpočtu (čas) $p(n)$, počet rôznych výpočtov je $\le s^{p(n)}$

[1] simulujúci TS bude mať (k+1) pások (I)

1) zistí dĺžku výpočtu: $p(n)$ (+ zistí dĺžku vstupu bin. kód)
vyznač na (k+1)-páske úsek dĺžky $p(n)$

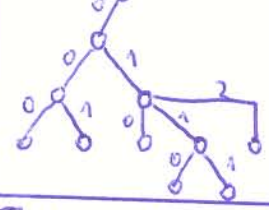
2) prechádzaj v cykle $s^{p(n)}$ možností

$\vdash$ | | | ... 0..s-1 0..s-1 0..s-1 — počítadlo v s-árnej sústave

pre každú kombináciu na počítadle simuluj výpočet pôvod. stroja. ak sa pôvod. stroj vetví, postupuj podľa i-tej vetvy (ak vetva nie je, preruš simuláciu i-tej vetvy)

- ak sa narazí na akcept. vetvu – AKCEPTUJ

$s=3\quad p(n)=4$



0000. nesedí
0001.. ×
0110. ✓

čas: $s^{p(n)}$ počet kombinácií (cyklov)

$s^{p(n)}\cdot(p(n)+p(n)) \le$ (nastavenie kombinácie na páske; simulácia vetvy)

$s^{p(n)}\cdot 2p(n) \le s^{p(n)}\cdot 2^{p(n)} \le (2s)^{p(n)} \le c^{p(n)}$

V: ak sa dá rozhodovací problém vypočítať v pamäti $p(n)$ nedeterm., tak sa dá vypočítať determinis. v pamäti $c\,p^2(n)$

- ak pôvod. alg. používa pamäť $p(n)$, nepoužije viac času ako $d^{p(n)}$

2-pásk TS: počet rôznych konfigurácií:

$$\|\Sigma\|^{p(n)}\cdot\|\Sigma\|^{p(n)}\cdot p(n)\cdot p(n)\cdot\|Q\|$$

možné symboly na 1. a 2. páske; pozície hláv; stavy

$$\le 2\|\Sigma\|^{p(n)}\cdot 2\cdot 2^{p(n)}\cdot\|Q\|^{p(n)} \le d^{p(n)}$$

riešenie algoritmu existuje ⇔ existuje riešenie, kt. sa nájde na $< d^{p(n)}$ krokov

⇐ triv.

⇒ ak alg. nájdel riešenie na $> d^{p(n)}$ krokov, potom sa stroj musel nachádzať 2-krát v rovnakej konfigurácii (det. stroj sa zacyklí, nedet. sa môže stále niekde vetviť)

Potom ex. akcept. konf. a kratší výpočet – prezrieme strom všetkých výpočtov do vzdialenosti $d^{p(n)}$ krokov

procedure dosiahnuteľnosť (C_1, C_2, t): boolean;
// vráti TRUE, ak sa z C_1 do C_2 dá dostať na $\leq t$ krokov

dosiahnuteľnosť (C_1, C_2, 0): ak $C_1 = C_2$
dosiahnuteľnosť (C_1, C_2, 1): podľa programu – nájdením príslušnej inštrukcie

možné, či ex. C: že platí: dosiah (C_1, C, $\lfloor t/2 \rfloor$) & dosiah (C, C_2, $\lceil t/2 \rceil$)

```
begin
  if t=0 then {
    if C1=C2 then
      return TRUE;
    else
      return FALSE;
  } if t=1 then {
    if z C1 do C2 na 1 krok then
      return TRUE
    else
      return FALSE;
  } else for ∀C  C nepoužije pamäť > p(n) do {
      if dosiah (C1, C, ⌊t/2⌋) &
         dosiah (C, C2, ⌈t/2⌉) then
        return TRUE;
      else
        return FALSE;
    }
end;

main () {      // C0 .. počiatočná konfigurácia
  for ∀C  C nepoužije > p(n) pamäte a TS akceptuje
          v konfigurácii C do begin
    if dosiah (C0, C, d^p(n)) then
      AKCEPTUJ; KONIEC
  end;
  ODMIETNI;
}
```

pamäť:
cena vyvolania dosiah (C_1, C_2, t)
C_1, C_2: zaberajú p(n) pamäte
t: na bin. zápis stačí $\log_2 t$, $t \leq d^{p(n)}$

$$\log_2 d^{p(n)} = p(n) \log_2 d = h\, p(n)$$

cena vyvolania dosiah() $\in h'' \cdot p(n)$

strom: dosiah (?, ?, $d^{p(n)}$) — počet vnorení $\log_2 d^{p(n)}$

dosiah (?, ?, $\frac{d^{p(n)}}{2}$) dosiah (?, ?, $\frac{d}{2}$)

dosiah (?, ?, $\frac{d^{p(n)}}{2}$)

pamäť sa recykluje: ak je prvé volanie dosiah() FALSE, tak nevoláme druhý | 2-hý dosiah() nevyužíva pamäť 1-vého dosiah()

$$h' \cdot p(n) \cdot h''(p(n)) + h''' p(n) \leq (h' \cdot h'' + h''') \cdot p^2(n)$$

($h' \cdot p(n)$: počet vnorení; $h''(p(n))$: pamäť pre 1 vnorenie; $h''' p(n)$: hlav. program) $= c \cdot p^2(n)$

DEF Problém (jazyk) L je NP-úplný, ak
- sa dá riešiť nedet. pol. algoritmom, či pre daný vstup má riešenie kladný
- pre iný jazyk L', kt. má nedet. pol. alg. ex. funkcia f vypočítateľná det. v pol. čase taká, že

$$\forall y \quad y \in L' \iff f(y) \in L$$

nech L' je problém s nedet. polyn. algoritmom
L je NP-úplný ⟹ ex. f vyp. v det. pol. čase
$y \in L' \iff f(y) \in L$
$dl(y) = n$ ... $dl(f(y)) = p(n)$

ak $dl(x) = m$, ex. nedet. pol. alg. rozhodujúci ?$x \in L$? v čase $p'(m)$

alg pre L':
- vypočítaj pre dané y hodnotu f(y) — $p(n)$
- rieš, či $f(y) \in L$ — $p'(p(n))$ (m)

celk. čas: $p(n) + p'(p(n))$ = polynóm

SAT je NP úplný problém
- riešiteľnosť nedet. polyn. algoritmom

binár. kód fle
1) skontroluj syntax vstupu (zátvorky ...) $O(n^3)$
2) vytvor tabuľku premenných formule $O(n^3)$
3) nedet. priraď premennej 0/1 $O(n)$
4) dosaď hodnoty z tabuľky do premenných
5) výsledok 0 .. ODMIETNI
   1 AKCEPTUJ

2^m rôznych vetiev, ak je fla splniteľná, aspoň 1 je úspešná
inak všetky zlyhajú

prevoditeľnosť:

1 pásk. nedeterministic TS, pracuje v čase $p(n)$

- ak $x_1, \dots, x_n \in L$ .. TS nastaví po $p(n)$ krokoch v konc. stave (aspoň 1 vetva)
- ak $x_1, \dots, x_n \notin L$ .. TS nastaví v nekonc. stave

$\Sigma = \{s_1, \dots s_{\|\Sigma\|}\}$ $s_1 = \#$

vstup $x_i = s_{k_i}$ $k_i \in \{1, \dots, \|\Sigma\|\}$

$Q = \{q_0, q_1, \dots q_{\|Q\|}\}$

q_1 .. počiat. q_2 .. konc. stav

inštrukcia $(q_k, s_j, s'_j, q'_k, d)$ nedeterminizmus

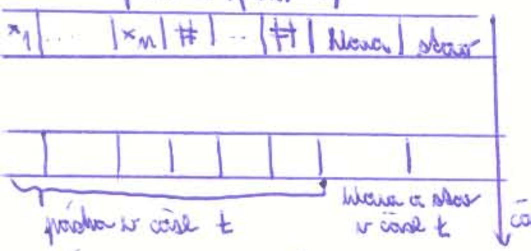
$d \in \{-1, 0, 1\}$

aby skončil výpočet na práve $p(n)$ krokov, pridáme inštrukcie $(q_2, s_j, s_j, q_2, 0)$ $\forall s_j \in \Sigma$

(akceptácia konc. stavom, nie zastavením!)

TS akceptuje $\equiv$ po práve $p(n)$ krokoch skončí niekt. vetva v q_2

tabuľka $p(n) \times (p(n)+2)$



premenné: (bool. premenné)

- $C_{i,j,t}$ .. v čase t na pozícii i je symbol s_j
  $i = 1, \dots, p(n)$ $j = 1 \dots \|\Sigma\|$ $\sim p^2(n)$ bool. premenných
  $t = 1, \dots, p(n)$
- $H_{i,t}$ .. v čase t hlava na pozícii i $\sim p^2(n)$ premenn.
- $Q_{k,t}$ .. v čase t stroj v stave q_k
  $k = 1, \dots, \|Q\|$

pre daný vstup $x_1, \dots, x_n$, polynóm $p(n)$, program TS zostavíme bool. fciu splniteľnú akk TS akceptuje v čase $p(n)$

- $U(v_1, \dots, v_p)$ .. práve jedna z hodnôt $v_1, \dots, v_p$ = TRUE ostatné false

cez $\vee, \&, \neg$, dĺžka zápisu $\sim p^2(n)$

1) v každom čase je na políčku jediný symbol

$$\bigwedge_{t=1}^{p(n)} \bigwedge_{i=1}^{p(n)} U(C_{i,1,t}, C_{i,2,t}, \dots, C_{i,\|\Sigma\|,t})$$

dĺžka zápisu $\sim p^3(n)$.

2) v každom čase je hlava na jedinom mieste

$$\bigwedge_{t=1}^{p(n)} U(H_{1,t}, H_{2,t}, \dots, H_{p(n),t}) \quad \sim p(n) \cdot p^2(n) \sim p^3(n)$$

3) v každom čase je TS v jedinom stave

$$\bigwedge_{t=1}^{p(n)} U(Q_{1,t}, Q_{2,t}, \dots, Q_{\|Q\|,t}) \quad \sim p(n) \cdot \|Q\|^2$$

4) prechodom medzi kr. a nasled. riadkom sa mení políčko len jedno

$$\bigwedge_{t=1}^{p(n)-1} \bigwedge_{i=1}^{p(n)} \bigvee_{j=1}^{\|\Sigma\|} (C_{i,j,t} \& (C_{i,j,t+1} \vee H_{i,t}))$$

čas políčka symboly

5) prechod od 1 riadku ku ďalšiemu ide v súlade s inštrukciami TS $\sim p^2(n) \cdot \|Q\| \cdot \|\Sigma\| \cdot$ konšt. $\sim p^2(n)$

$$\bigwedge_{t=1}^{p(n)} \bigwedge_{i=1}^{p(n)} \bigwedge_{j=1}^{\|\Sigma\|} \bigwedge_{k=1}^{\|Q\|} \left(\neg Q_{k,t} \vee \neg C_{i,j,t} \vee \neg H_{i,t} \vee \left(\bigvee_{l=1}^{N} Q_{v(j,k,l),t+1} \& C_{i,w(j,k,l),t+1} \& H_{i+d(j,k,l)}\right)\right)$$

6) v prvom riadku je vstup, za ním #-symboly, hlava je na ľ. okraji, stroj je v poč. stave

$$\left(\bigwedge_{j=1}^{n} C_{j,k_j,1}\right) \& C_{j,1,1} \& H_{1,1} \& Q_{1,1}$$

pozícia / symboly s_j ; $s_1 = \#$ $\sim p(n)$

7) v čase $p(n)$ je stroj v akcept. stave

$Q_{2,p(n)}$ ~ 1

máme bool. fciu (1) & (2) & ... & (7),

$f(y_1, \dots, y_m)$ rádovo $p^2(n)$ premenných

dĺžka zápisu $\sim p^3(n)$

ak $H_{i,t}$, $Q_{m,t}$, $C_{i,j,t}$ ako 1 symbol, zápis sa dá generovať v čase $p^3(n)$

kódi splniteľnosti: binár. kód. pre 1 premennú $\log_2 n$ bitov

pre $p^2(n)$ premenných: $\log_2 p^2(n)$ bitov

dĺžka bin. zápisu f: $p^3(n) \cdot \log_2 p^2(n) \leq c\, p^4(n)$

f je splniteľ. $\Leftrightarrow$ TS má aspoň 1 akcept. výpočet končiaci na $\leq p(n)$ krokov

[V] CNF-SAT je NP úplný

4

každá bool fcia sa dá prepísať na CNF - zápis môže narásť exponenciálne

vezmeme 0-vé riadky, 0-vé premenné dáme normálne
1-vé ako $\neg x_i$, medzi ne $\vee$

môžeme ale dostať až 2^n klauzúl

alebo upravujeme lokálne

fcia v tvare nieCNF z predch. vety mala dĺžku $\leq c\, n^4$ vzhľadom na dĺžku vstupu. Takmer celá v CNF, okrem 4)

$$\bigwedge_{t=1}^{p(n)-1} \bigwedge_{i=1}^{p(n)} \bigvee_{j=1}^{||\Sigma||} \left(C_{ijt} \,\&\, (C_{ijt+1} \vee H_{i,t})\right)$$

$(p(n)-1)(p(n))$ konjunkcií, $||\Sigma||\cdot 3$ premenných

prevod na CNF: veľkosť bude $c \cdot 2^{3||\Sigma||} \cdot 3||\Sigma||$

5) $(p(n)-1)\, p(n)\, ||Q||\, ||\Sigma||$ konjunkcií, v každej najviac $3+3N$ premenných. po CNF: $\leq c\, 2^{3+3N}(3+3N)$

[V] 3CNF-SAT je NP úplný

prevod CNF $\xrightarrow[\text{det. pol.}]{}$ 3CNF

klauzulu $(l_1 \vee l_2 \vee .. \vee l_s)$ $s>3$ prevedieme

$(l_1 \vee l_2 \vee y_1) \,\&\, (\neg y_1 \vee l_3 \vee y_2) \,\&\, (\neg y_2 \vee l_4 \vee y_3) \ldots$

$(\neg y_{s-4} \vee l_{s-2} \vee y_{s-3}) \,\&\, (\neg y_{s-3} \vee l_{s-1} \vee l_s)$

[D] f je pre $x_1, .., x_n$ splnená $\Leftrightarrow$ je splnená každá klauzula f (kvôli &) $\Rightarrow$ je splnená $(l_1 \vee l_2 \vee .. \vee l_s)$

$\Rightarrow$ nech $\bar{v}(l_i)=1$, $\bar{v}: EV^* \to \{0,1\}$

klauzula pre l_i: $(\neg y_{i-2} \vee l_i \vee y_{i-1})$

ak položíme $y_1=1, y_2=1, .. y_{i-2}=1$ .. klauzuly naľavo budú splnené

$y_{i-1}=0, y_i=0, .. y_{s-3}=0$ klauzuly napravo budú splnené ✓

$\Leftarrow$ nech je splnená každá klauzula: potom

• $(l_1 \vee l_2 \vee y_1)=1$
- $l_1=1$ potom OK
- $l_2=1$ .. OK
- $l_2=0 \,\&\, l_1=0$ potom ale $y_1=1$

ak $y_1=1$ potom $l_3 \vee y_2=1$ { $l_3=1$ OK; $y_2=1$ }

• na konci $l_{s-1} \vee l_s=1$ (lebo $y_{s-3}=1$ alebo je niektoré $l_{s-1}, l_s=1$

ak dĺžka f v CNF je $c\, p^4(n)$, dĺžka v 3-CNF je $c' \cdot p^4(n)$
pribudnú len rádovo s-2 a s-3 premenných

[V] 2CNF splniteľnosť je riešiteľná det. pol. algoritmom

• zisti, či neexistuje bezkonfliktná premenná
$\ldots \,\&\, (x \vee y) \,\&\, \ldots \,\&\, (\neg x \vee y)$

- ak ex. bezkonfliktná premenná, dosaď: $x_i := 1$
$x_i := 0$ ak je neg.
- vyhoď tým pádom splnené klauzuly
- rieš splniteľnosť $f(x_1, .. x_{i-1}, x_{i+1}, ..)$

• máme už len samé konfliktné premenné

3-ZAFARBITEĽNOSŤ

SAT → 3ZAF

graf ako matica incidencií $(m \times m)$

• ofarbenie v det. pol. čase:
zisti, či dĺžka vstupu $= m^2$, symetria, nájdi farby, zisti, či je nakreslenie 3 farieb správne (2 cykly) $3^{\sqrt{m}}$ výpočtov

• úplnosť SAT → zafarbiteľnosť

vstup dĺžky n, f binárne kódovaná, |zápis| $\leq n$

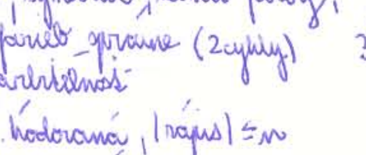
v .. počet premenných f $\quad v \leq n$

o .. počet bool operátorov $\quad o \leq n$

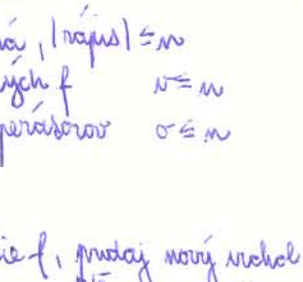
1) vytvor  BUNV

vezmi premenné $x_1, .., x_v$ funkcie f, pridaj nový vrchol pre každú premennú

čierna = TRUE
biela = FALSE

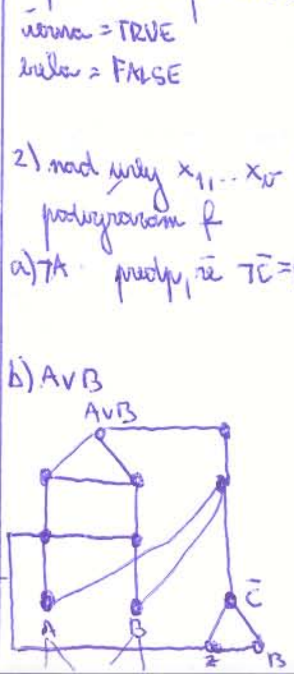


2) nad uzly $x_1, .. x_v$ budujeme podgrafy prislúchajúce podvzorcom f



a) $\neg A$ predp., že $\neg \bar{C} = B$, $\neg B = \bar{C}$

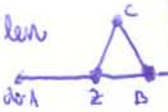
b) $A \vee B$



do A, B idú hrany z $v_1, .. v_s$

teda
- $A = B \quad B = Z$
- $A = \bar{C} \quad B = \bar{C}$ (4 možnosti)
- $A = b \quad B = \bar{C}$
- $A = \bar{C} \quad B = b$ (symetrická k 3)

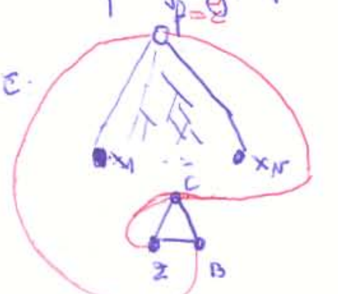
c) A&B - graf taký istý, len



namiesto do cievky [5] sa pripojí do tretej farby zhora

budujeme graf pre stále väčšie podstromy, kým nemáme celú funkciu f

záverečný stav: hore je E (TRUE)



- graf konštruujeme deterministicky.
- s každým operátorom pridáme najviac 5 vrcholov
- 3 vrcholy v Δ
- n vrcholov pre premenné

$3+n+5\sigma \le 3+n+5n \le 7n$ vrchol ($\sigma \le n$)

pre funkciu dĺžky n vznikne graf dĺžky $\le (7n)^2 = 49n^2$.

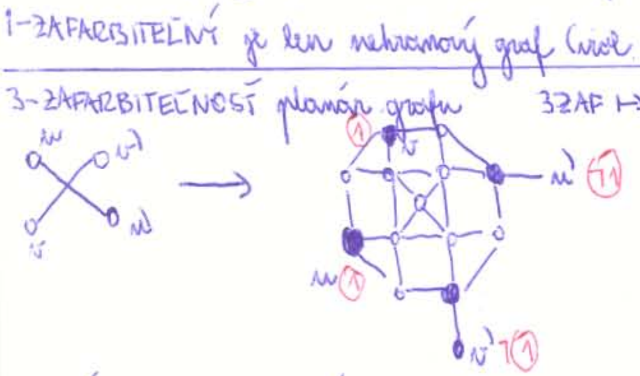
k-ZAFARBITEĽNOSŤ $k\ge 3$ 3ZAF → K-ZAF.

◦ vytvoríme k-3 vrcholový kompletný graf G'
◦ z G vedieme k-3 hrán do G', na časť G sa použijú zvyšné 3 farby (tieto už vieme zafarbiť)

2-ZAFARBITEĽNOSŤ je deterministická!
prechádzame farby, vynútené vrcholy farbíme opačne

1-ZAFARBITEĽNÝ je len nehranový graf (iba vrcholy)

3-ZAFARBITEĽNOSŤ planár. grafu   3ZAF ↦ planár 3ZAF



ofarbíme, rozriešime prípady, nový graf sa rozšíri o d vrcholov

POKRYTIE MNOŽINY (PRESNÉ)   (3ZAF ↦ POKRYTIE

máme $\{1,\dots,m\}$, množiny $A_1,\dots,A_l \subseteq \{1,\dots,m\}$

existuje $B \subseteq \{1,\dots,l\}$, že ◦ $\bigcup_{i\in B} A_i = \{1,\dots,m\}$
◦ $A_i \cap A_j = \emptyset$ ak $i\ne j$, $i,j\in B$

výber podmnožiny $P(\{1,\dots,m\})$, aby tvorili rozklad

◦ nedet. pol.: bruteforce 2^l možností: generuj & testuj
◦ NP-úplnosť: $G=(V,E)$ je k-zaf ⟺ ex. presné pokrytie A

klika → VP
$V'=V$ $k'=\|V\|-k$
$E'=V\times V - E$

ex. klika v G ≡ ex. $v_1,\dots,v_k$: $(v_i,v_j)\in E$ $\forall i\in[1,k]$ ≡ ex. $v_1,\dots,v_k$: $(v_i,v_j)\notin E'$ ≡ každá hrana G' má aspoň 1 vrchol z $V-\{v_1,\dots,v_k\}$ ≡ ∃ $u_1,\dots,u_z \in G$: pokrývajú ∀ hrany G', $z = k' = \|V\|-k$

◦ nahraď množinu: $(E\cup V)\times\{1,\dots,k\}$ (111)
◦ $\forall i\in[1,k]$ $\forall (u,v)\in E$ pridaj do množín množinu $((u,v),i)$
◦ ~~nahraď množinu~~
$\forall v\in V$ $\forall i\in[1,k]$ pridaj $\{v\}\cup\{((v,w),i) : (v,w)\in E\}$

KLIKA (CLIQUE)   CNF-SAT ↦ CLIQUE

daný je $G=(V,G)$, $k\in\mathbb{N}$. dá sa nájsť v G k-vrcholová kružnica?

◦ nedet. polyn.: bruteforce všetky k-tice, determ. testuj úplnosť
◦ NP-úplnosť
nech f má k klauzúl
m premenných
◦ uprav klauzuly typu $(x_i \vee \dots \vee x_i) \mapsto (x_i \vee \dots)$
$(\neg x_i \vee \dots \vee \neg x_i \dots) \mapsto (\neg x_i \dots)$
$(\dots x_i \vee \dots \vee \neg x_i) \mapsto \emptyset$ (vždy splnená)
◦ máme f v CNF, v každej klauzule je premenná iba raz
◦ zostroj graf G: $V=\{l_{ij} : i\in[1,k], j\in[1,k_i]\}$
všetky literály vo všetkých klauzulách
$E=\{(l_{ij}, l_{i'j'}) : i\ne i' \wedge l_{ij}$ nie je v konflikte s $l_{i'j'}\}$

$i\ne i'$: vedieme len hranu cez klauzuly
konfliktné: $l_{ij}=x_n$ & $l_{i'j'}=\neg x_n$
$l_{ij}=\neg x_n$ & $l_{i'j'}=x_n$

[L] f je splniteľná ⟺ v G existuje k-klika

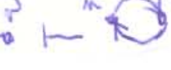
⇒ musia byť splnené ∀ klauzuly
v klauzule musí byť splnený aspoň 1 literál $\exists j\ v(l_{ij})=1$
majme množinu ~~…~~ literálov (splnených) zo všetkých klauzúl: $\{l_{1j_1}, l_{2j_2}, \dots, l_{kj_k}\}$, pričom z každej klauzuly berieme len 1 splnený literál
literály v množine sú spojené hranou, lebo:
◦ patria do rozličných klauzúl
◦ nemôžu byť v konflikte (všetky sú true) } ⇒ máme kliku

⇐ majme k-kliku
- každý z vrcholov kliky musí patriť do inej klauzuly (inak by neboli spojené hranou) ⇓ k klauzúl
- premenné nahodíme: ak $l_{ij}=x_n \Rightarrow x_n:=1$
$l_{ij}=\neg x_n \Rightarrow x_n:=0$

zvyšné premenné ľubovoľne, už je f splniteľná

VRCHOL. POKRYTIE   klika → VP
majme $G'=(V,E)$, k
? ex. k vrcholov, aby každá hrana G' susedila s jedným z týchto vrcholov?

KONTROLA CYKLOV   VP → KC
daný je $G=(V,E)$ orientovaný, k
? ex. k vrcholov, aby každý cyklus išiel cez jeden z k vrch.?
◦ $G'=(V',E')$ neorientovaný
◦ $V=V'$
◦ v G':



6

VYVAŽOVANIE

- ex. DET. NEPOLYN. NEEXP. ALGORITMUS

dané sú $X, a_1, \dots, a_m \in \mathbb{N}$ $\exists$ ex $A \subseteq \{1, \dots, m\}: X = \sum_{i \in A} a_i 2$

veľké rozšírenie (X) malé rozšírenie ($a_1, \dots, a_m$)

- det. pol. alg:
  - 2^m kombinácií, sčítať a overiť
  - alg. $p(n) \cdot 2^{\sqrt{n}}$

1) rozdeľ $a_1, \dots, a_m$ do 2 skupín: čísla s dĺžkou bin zápisu $\le \sqrt{n}$: $b_1 \dots b_k$; $> \sqrt{n}$: $c_1 \dots c_l$

2) vyrob všetky možné súčty malých čísel: $B_1 \dots B_{k'}$ a možné súčty veľkých čísel $C_1 \dots C_{l'}$

ohraničíme pre k':

- možno najviac m rozšírení, malé rozšírenia majú dĺžku bin zápisu $\le \sqrt{n} \Rightarrow$ hodnota rozšírenia nie je $> 2^{\sqrt{n}}$

$0 \le B_i \le m \cdot 2^{\sqrt{n}} \le n \cdot 2^{\sqrt{n}}$ ($m \le n$) $\forall i \in [1, \dots, k']$

tj. $B_i \in [0, n 2^{\sqrt{n}}]$, k' je rozšírení $\Rightarrow k' \le 1 + n 2^{\sqrt{n}}$ (Dirichlet) ← nula

ohraničíme pre l': $l' \le 2^l \le 2^{\sqrt{n}}$

- všetkých možných súčtov veľkých rozšírení $\le 2^{\sqrt{n}}$; l čísel dĺžky $\ge n$, ale $l\sqrt{n}$ musí $\le n$, $l\sqrt{n} \le n \Rightarrow l \le \sqrt{n}$ ✓

hľadajme kombinácie $B_i + C_j$, ktoré dávajú X (je ich $\le k' \cdot l'$)

utriedime postupnosti $B_1, \dots B_{k'}$ i $C_1, \dots C_{l'}$. Ideme vyhľadávať $c = k' + l'$ krokov.

$B_1, B_2, \dots B_{k'} \mid C_1, \dots C_{l'}$ ← ↑ ⇔ ↑ (súčty, nájdená pozícia)

ideme v B sprava doľava, sčítavame tým nadoplnku k rovnosti

zložitosť $((1+n2^{\sqrt{n}}) \cdot 2^{\sqrt{n}} + (1+n2^{\sqrt{n}}))\, p(n) \le p'(n) 2^{\sqrt{n}}$

- NP-úplnosť 3ZAF → balance

nech je $G = (V, E)$, kódujeme binárne $X, a_1, \dots, a_m$

máme H-cifer bin čísla $H = 2\|V\| + 6\|E\|$

$|\overset{1}{22}|\overset{2}{22}| \dots |\overset{\|V\|}{22}|\,22\,22\,22\,|\,222222\,| \dots |\overset{\|E\|}{222222}|$ (1 farba, 2 f, 3 farba)

- rozšírenia $a_1 \dots a_m$

1) $\forall v \in V, \forall f \in \{1,2,3\}$ pridaj rozšírenie

00|00| . |00|01|00| ... |00||xxxxxx| ... |xxxxxx|

$3\|V\|$ rozšírení; vrchol v; pre hranu $(v, ?)$ resp. $(?, v)$ nastav na: 010000, f=1; 000100, f=2; 000001, f=3

2) pomoc. rozšírenia

$\forall (u_1, u_2) \in E\ \forall f \in \{1,2,3\}$ pridaj

00| .. |00|000000| |xxxxxx| |000000| inak 000000

$3\|E\|$ rozšírení

- rozšírime X: 01|01| ... |010101| .. |010101| } $2\|V\| + 6\|E\|$

L) nenastáva prenos pri sčítaní dvojíc do vyšších rádov

XX	YY
+ XX	XX
22	22

(1) prvých $2\|V\|$ rádov:

- rozšírenia pre v, f: pre 1 vrchol sú len 3 rozšírenia, tj. 01 sa vyplytne na i-tom mieste pre $v_i \le 3\times$ → preplňovaný prípad $01+01+01=11$
  - hrana incidujúca s 2 vrcholmi, preto v šestici nastane preplňovaný prípad $01+01=10$
- pomoc. rozšírenia $3\|E\|$: pre jednu hranu sa v trojici šestíc vyplytne na p-tej pozícii jednotka najviac raz

jedno pomoc. rozšírenie: 001 .. 100| .. 010000, —||— 001000, —||— 000001

L) G je 3zafarbiteľný $\equiv X = \sum a_i$

⇒ nech f_{ij} označuje farbu v_j, $j \in [1, \|V\|]$, $i \in [1,2,3]$ platí pre (u, v) $f_{iu} \ne f_{iv}$

1) pouvažujeme rozšírenia

(v_1, f_{i1})	01 00 .. 00	...
(v_2, f_{i2})	00 01 .. 00	
⋮		
$(v_{\|V\|}, f_{i\|V\|})$	00 00 .. 01	
	01 01 .. 01	....

2) zoraď hranové rozšírenia

napr. nech (u, v), $f(u) = 1$, $f(v) = 3$

01 00 00	... — $f(u)=1$
00 00 01	... $f(v)=3$
00 00 00	... } ostatné rozšírenia
00 00 00	...
01 00 01	} plus uvažujeme permutácie šíc

prípad, že by oba vrcholy 1 hrany boli rovnakej farby nenastane

súčet

01|01| .. |01||010000| ... |010001|

dopočítame ešte pomocné rozšírenia a máme 01|01| 01 ✓

⇐ nech $X = \sum a_i$ má riešenie ? vieme urobiť 01|011..101 súčet na ráváni? [7]

- vrcholové disjunkce $XX = 01$
  - pomocné rávária nemajú vplyv
  - vrcholové rávária iné od vrcholu v tiei
  - $\Sigma = 01$ musel vzniknúť len v $[v,1], [v,2], [v,3]$ ale mohlo sa použiť len jedno (inak $\Sigma \neq 01$) vrchol ofarbíme podľa použitého rávária
- ofarbenie hrán: vrcholy spojené hranou majú rovnú farbu

spor: majme $[v_1, i], [v_2, i]$ potom ale súčet na XXXXX by bol

$$\begin{array}{r} \dots 01 \dots \\ + \dots 01 \dots \\ \hline \dots 10 \dots \end{array} \quad / \dots 11 \dots \text{ ak pomoc. ráváriа}$$

spor

---

graf veľkosti n → $1 + 3\|V\| + 3\|E\|$ rávárií

krát $2\|V\| + 6\|E\|$ číslic na ráváriа

$(1 + 3\|V\| + 3\|E\|) \cdot (2\|V\| + 6\|E\|) \sim 18n^2$ veľkosť

($\|V\| \sim \sqrt{n}$, $\|E\| \sim n$)

VYVAŽOVANIE S PODMIENKOU

$X; a_1, \dots, a_m$: $\sum_{i=1}^{m} a_i < 2X$? $\exists A \subseteq \{1,\dots,m\}: \sum_{i \in I} a_i = X$

vyvažovanie $X, a_1, \dots, a_m \longrightarrow X', a_1', \dots, a_m'$

$X' = X + \sum_{i=1}^{m} a_i$, pridáme rávárie $a_0 = \sum_{i=1}^{m} a_i$

[L] platí vyvaž. podmienka

(1) ? $\sum_{i=0}^{m} a_i < 2X'$? $\sum_{i=0}^{m} a_i = a_0 + \sum_{i=1}^{m} a_i = \underbrace{\sum_{i=1}^{m} a_i}_{a_0} + \sum_{i=1}^{m} a_i \leq$

$\leq X + \sum_{i=1}^{m} a_i = X' < 2X'$ ✓

[L] vyvažovanie má riešenie ⟺ vyvaž. s podm. má riešenie

⇒ nech $X, a_1, \dots, a_m$ má rieš. ⇒

$\exists A \subseteq \{1,\dots,m\}$ $\sum_{i \in A} a_i = X$ ⟺ $a_0 + \sum_{i \in A} a_i = a_0 + X = X'$

⇐ nech $X', a_1', \dots, a_m'$ má rieš. ⇒

$\exists A' \subseteq \{0, 1, \dots, m\}$ $\sum_{i \in A'} a_i = X'$. Muselo sa použiť a_0, ? $0 \in A'$?

nech $0 \notin A'$: $\sum_{i \in A'} a_i \leq \sum_{i=1}^{m} a_i < X + \sum_{i=1}^{m} a_i = X'$

$\sum_{i \in A'} a_i \leq X$ .. ale spor lebo v predp. platí =.

ak sa použilo a_0, $\sum_{i \in A'} a_i = a_0 + \sum_{i \in A' - \{0\}} a_i = X' = X + a_0$

$\sum_{i \in A' - \{0\}} a_i = X$

$\sum_{i \in A} a_i = X$ ✓

PRESNÉ VYVÁŽENIE

(IV)

$a_1, \dots, a_m$. ? $\exists A \subseteq \{1,\dots,m\}: \sum_{i \in A} a_i = \sum_{i \notin A} a_i$

vyváženie s podmienkou.

$X; a_1, \dots, a_m$, $\sum_{i=1}^{m} a_i \leq 2X$? $\sum_{i \in A} a_i = X$, ak $A \subseteq \{1,\dots,m\}$?

presné vyvaž.

$a_1', a_2', \dots, a_m'$. Položme $a_0 = 2X - \sum_{i=1}^{m} a_i$

a riešme $a_0, a_1, \dots, a_m$ presné vyv.

overenie ⟺ problémov

⇒ nech $\exists A \subseteq \{1,\dots,m\}$: $\sum_{i \in A} a_i = X$

riešením nového problému bude táto istá A

$\sum_{i \in A} a_i = X$ (1) $\sum_{i \notin A} a_i$ ⇒ $a_0 + \sum_{i \notin A} a_i = \left(2X - \sum_{i=1}^{m} a_i\right) + \sum_{i \notin A} a_i$

$0 \notin A$, lebo v pôvod. probléme a_0 nebolo

$= \left(2X - \sum_{i=1}^{m} a_i + \left(\sum_{i=1}^{m} a_i - \sum_{i \in A} a_i\right)\right) = X$ ✓

⇐ nech $\exists A \subseteq \{1,\dots,m\} \cup \{0\}$ $\sum_{i \in A} a_i = \sum_{i \notin A} a_i$

BUNV $a_0 \in A$. $\exists A' \subseteq \{1,\dots,m\}$ $\sum_{i \in A'} a_i = a_0 + \sum_{i \notin A} a_i$

$\sum_{i \in A'} a_i =$

MINIMALIZÁCIA ROZDIELU NA VÁHACH

dané: $a_1, \dots, a_m$? $\exists A \subseteq \{1,\dots,m\}$: $\max\{\sum_{i \in A} a_i, \sum_{i \notin A} a_i\} \to \min$

(1) redukujeme naň presné vyváženie

nech $Y = \sum_{i=1}^{m} a_i$. zrejme: $\max\{\sum_{i \in A} a_i, \sum_{i \notin A} a_i\} \geq \frac{Y}{2}$

V] NLOGSPACE ⊆ PSPACE

① TS: $\vdash x_1 \dots x_n \dashv$

↑ ⟷ R len čítacia

[q]

↓

$\vdash y_1 \dots y_m \# \dots$ R/W .. prac. páska

inštrukcie $(q, x, y, y', q', \{R,L,N\}, \{R,L,N\})$

páska | prac. páska — najviac model.

pre vstup dĺžky n možno použiť $\leq k \cdot \log(n)$ bitov

začiatok $\vdash x_1 \dots x_n$, ↑ q_I, ↓ $\vdash \# \dots \#$

koniec
- akceptuje, ak stroj skončí v q_F
- inak zamietne

počet konfigurácií stroja

$\|Q\| (n+2) (k \log(n) + 1) \cdot 2^{k \log(n)} \leq \|Q\| \cdot 3n (2kn) \cdot n^k$

n bitov, ⊢, ⊣ — pozícia hlavy; ⊢ — obsahy pamäte pásky; $\leq n^{k'}$ | je ohraničený polynómom

chceme simulovať NLOG-TS polynomiálne deterministicky!

1.) vyrobíme zoznam $C_0, C_1 \dots C_m$ všetkých možných konfig. $m \leq n^{k'}$, C_0 počiatoč. konf.

2.) ideme v cykle: repeat until false
- prehľadáme ∀ možné dvojice C_i, C_j ($\sim m^2$)
  ak je C_j dosiahnuteľná na 1 krok z C_i NLOG strojom, označíme C_j ako dosiahnuteľnú (└ musí byť dosiahnuteľná)
- opakujeme cyklus kým: o označili sme ako dosiahnuteľnú konf. so q_F (ACCEPT)
  o už sa nedá nájsť $C_i \vdash C_j$ (REJECT)

cyklus sa opakuje najviac m×, lebo v každej iterácii sa označí aspoň 1 konf.

celk. čas: polynóm $\sim m \leq p'(m) \leq p'(p(n)) = p''(n)$ ✓

VI] Existuje P-úplný problém — dokazovanie z predpokladov

máme: bool. premenné $p_1, \dots, p_{m_1}$ (axiómy)

implikácie: $I_1, \dots, I_{m_2}$ (tvaru $v_1 \wedge v_2 \wedge \dots \to v_0$)

tvrdenie: Z .. treba zistiť, či sa dá dokázať alebo nie (či platí)

I) dá sa riešiť v det. pol. čase

1) vytvoríme zoznam ∀ bool. premenných na vstupe $v_1, v_2, v_3, \dots v_m$

2) označíme ako dosiahnuteľné tie, čo sú medzi predpokladmi $p_1 \dots p_{m_1}$

3) v cykle prejdi všetky implikácie $I_1, \dots I_{m_2}$

- ak I_j: $v_{i_1} \& \dots \& v_{i_k} \to v_{i_0}$ a všetky v_{ij} sú dosiahnuteľné, označ v_{i_0} ako dosiahnuteľnú
- opakuj kým a) Z je označ. ako dosiah. alebo
  b) nič sa v iterácii neoznačilo

cyklus sa opakuje $\leq m$ krát, $m \leq n$, m .. počet premenných
- vnútor cyklus cez implikácie: max m_2× na vstupe, $m_2 \leq n$
- pre 1 implikáciu cez všetky premenné: $\leq n$

$\sim n^3 \Rightarrow$ P problém

II. P-ÚPLNOSŤ

DEF deterministic. transducer pracujúci v pamäti $\leq k \log(n)$

$\vdash x_1 \dots x_n \dashv$ ↑ ⟵ R ⟶

[q] ⟶ $z_1 z_2 \dots z_4 \# \#$ (výstup)

↓ ⟵ R/W ⟶

$\vdash y_1 \dots y_n$

pre ľub. polyn. alg. ex. 1-pásk. TS pracujúci v pol. čase (Church. téza)

upravíme TS, aby počítal (akceptoval) presne v p(n) čase:

pridaj $(q_F, x, x, q_F, 0)$ $\forall x \in \Sigma$

pre tento TS M' a vstup $x_{i_1}, \dots, x_{i_n}$ $(\Sigma = \{x_1, \dots x_{\|\Sigma\|}\})$, ⊢ = x_1, $x_2 = \#$

vieme rozbiť problém dokazovania, riešiteľný akk TS M' akceptuje tento vstup

1) bool. premenné:

$C_{i,j,t}$ .. v i-tom políčku je x_j (v čase t)

$H_{i,t}$ .. hlava je na pozícii i } čas .. t

$Q_{i,t}$ .. TS je v stave q_i

2) predpoklady

$Q_{1,0}$ .. $t = 0$, $q_1 = q_F$

$H_{0,0}$ .. $x_0 = \vdash$

$C_{0,1,0}$, $\forall j \in [1, n]$: $C_{i,j,0}$ .. vstup

koniec pamäte $\forall j \in [n+1, p(n)]$: $C_{j,2,0}$; $x_2 = \#$

$C_{j,i_j,0}$ .. na výstup zadať $j \in [1, n]$, potrebných je $\leq \log n$ bitov

v cykle (vnútri) načítaj symbol zo vstupu, posuň sa →, napíš $C_{j,i_j,0}$

$C_{j,2,0}$ pre $j = n+1, \dots, p(n)$ nezávisle od vstupu

V

3) implikácia

TS M' má len konečný počet inštrukcií typu $(q_{k_1}, x_{k_2}, x_{k_3}, q_{k_4}, \leftarrow)$. Všetky inštrukcie sa vygenerujú takto:

$(q_{k_1}, x_{k_2}, x_{k_3}, q_{k_4}, \rightarrow) \vdash Q_{k_1,t} \& H_{i,t} \& C_{i,k_2,t}$
$\rightarrow Q_{k_4,t+1}$ | $i \in [0, p(n)-1]$
$\rightarrow C_{i,k_3,t+1}$ | $t \in$ –"–
$\rightarrow H_{i+1,t+1}$

symbol sa nemení, ak tam nie je hlava:

$C_{i',j,t} \& H_{i'',t} \rightarrow C_{i',j,t+1}$ $\forall t \in [0,..,p(n)-1]$ $\forall j \in \{1,..\|\Sigma\|\}$ $\forall i', i'' \in [0, p(n)], i' \neq i''$

generovanie implikácií sa dá v pamäti $k \cdot \log_2 n$

Pr: $C_{i',j,t} \& H_{i'',t} \rightarrow C_{i',j,t+1}$

```
for t:=0 to p(n)-1 do
  for j=0 to ||Σ|| do
    for i':=0 to p(n) do
      for i'':=0 to p(n) do
        if i'≠i'' then
          generuj implikáciu
```

} 4 premenné, nadobudne žiadna nebude väčšia ako $p(n) \le n^k$ .. rozsah bude $\le k \cdot \log_2 n$ bitov

4) záver

bola odvodená $Q_{2,p(n)} \rightarrow$ M' sa nachádza v $t = p(n)$ v q_F

---

odhad dĺžky vygener. problému:

$l \le (3 + p(n)) + konšt \cdot p^2(n) + konšt' \cdot p(n)^2 + 1)$ konšt'' $\cdot \log(p(n)^3)$

predp. — implik. pre inštrukcie — kopírovanie nahradí — dĺžka implik. — bin. kód premenných Q, H, C

$l \le p'(p(n)) = p''(n) = n^{k''}$

Predp. že ex. alg., kt. vie rozlíšiť, či zhornový dôkaz má riešenie a pre vstup dĺžky l nepotrebuje použiť $> b \log n$ bitov, tak pre problém vygenerovaný prevodníkom by použil $\le k$ konšt $\log(l)$ pamäte