1. n největších měst, přidělení do stanovišť a projeli ostatních lodí, aby v žádném nebyl 2x. Odplutí z posledního města n. Ohodnocení cest na základě bodů. Vyhrál ten, který cestu hodnotila aspoň poloviny všech bodů.

Amundsenův problém A: $G_A = (V_A, E_A)$, každá hrana i G_A má ohodnocení $w_i \in N$, pokud loď vlastní hrany na hraně zajímavosti, jinak $w_i = 0$. Hledáme cestu $path_A = e_1, e_2, \ldots e_{n-1}$ kde mají hrany příslušná ohodnocení $cost_{path_A} = w_1, w_2, \ldots w_{n-1}$. Je dána mezní hodnota M, kde $M = \frac{1}{2} \sum w_i$, kde $\sum w_i$ je součet ohodnocení všech hran v grafu. Hledáme tedy cestu, která všechny tyto podmínky splňuje. Musí platit, že $\sum cost_{path_A} \geq M$. Dokažte, že Amundsenův problém A je NP-úplný.

Důkaz:

I. $A \in NP$ (členství v problému)

Konstruujeme NTS M, co problém rozhoduje v konečném (polynomiálním) čase, pracující následovně:

1. M ověří, zda jeho vstup je platná instance problému A. Pokud ne, odmítne.

Složitost - probíhá kontrola oddělovačů, kódování, neduplikátů - zachování polynomiální složitosti

2. Guess: M vybere náhodně cestu v daném grafu G_A. Hádáme kandidáta na řešení problému A.

Složitost - vybírání $O(n)$ kandidátů - zachována nejvýše polynomiální složitost

3. Check: M ověří, zda daná zvolená cesta má celkové ohodnocení hran cesty aspoň M, což je mezní hodnota. Dále musí ověřit, že cesta vychází z prvního navštíveného města, končí v n-tém městě, žádné není navštíveno více jak 1x.

Pokud jsou všechny tyto podmínky splněny, tak zvolenou cestu v kroku guess přijme, jinak odmítne.

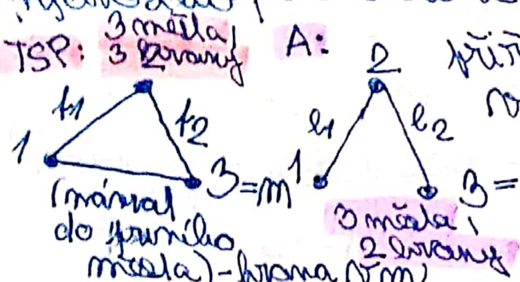
Složitost - probíhá porovnávání - zachována nejvýše polynomiální složitost

Kateřina Fortová (xforto00)

II. A je NP-těžký

- redukce nějakého známého NP-úplného problému
- vybrán problém TSP (Travelling Salesman Problem - Problém obchodního cestujícího)
- důkaz polynomiální redukcí z TSP

Travelling Salesman Problem: graf $G_{TSP} = \langle V_{TSP}, E_{TSP} \rangle$, hledání nejkratší hamiltonovské kružnice. Úkolem je najít nejkratší možnou cestu $path_{TSP} = f_1, f_2, \dots, f_{m-1}, f_m$ pro m měst, která navštíví všechna místa pouze 1x kromě výchozího, do kterého se na konci vracíme. Cesta $path_{TSP}$ má přiřazené příslušné váhy hran $cost_{path_{TSP}} = v_1, v_2, \dots, v_{m-1}, v_m$ kde hledáme cestu s $\min(cost_{path_{TSP}})$.



Problém A redukujeme na problém TSP, kde $G_A = G_{TSP}$, nutnost projít všech vrcholů jen 1x (kromě poč. města). Počet měst $n = m$, pro A přidána hrana z posl. města do prvního (cena nové přidané hrany 0). Pak $path_{A_{corrected}} = e_1, e_2, \dots, e_{n-1}, e_n = path_{TSP}$. Ohodnocení příslušných hran mají cenu $cost_{path_{A_{corrected}}} = w_1, w_2, \dots, w_{n-1}, w_n = cost_{path_{TSP}}$.

Přičemž u A problému hledáme cestu co splňuje podmínku $\sum cost_{path_{A_{corrected}}} \geq M$, kde uvažujeme, že cena poslední hrany je 0. (z posledního do prvního města). Nehledáme tedy cestu s minimální cenou (délkou) jako u TSP, ale splňující tuto podmínku.

Je tedy nutné u redukce uvažovat přidání/odebrání hrany mezi koncovým a poč. městem, který TSP uvažuje, ale A ne. Pak se dále uvažuje jiná podmínka pro vybrání vhodné cesty (nejkratší (minimální váhy) u TSP, splňující nerovnost s mezní podmínkou u A)

I přes tyto malé změny se dá snadno nahlédnout, že TSP je vhodným známým NP-úplným problém pro vytvoření důkazu NP-těžkosti problému A a redukce se tedy dá realizovat v polynomiálním čase. □

2. L nad abecedou $\Sigma = \{0,1\}$

L' je jazyk všech slov co mají 2 různá podslova z L

$L' = \{ w \in \Sigma^* \mid \exists\, u,v : u \neq v \wedge u \sqsubseteq w \wedge v \sqsubseteq w \;\; u,v \in L \}$

$\sqsubseteq$ - relace být podslovem

$L \in DTIME[n^2] \Rightarrow L' \in DTIME[n^k]$

Intuice: $\exists$ - tedy stačí najít 2 různá podslova

$\boxed{w}$ vstupní řetězec w z L'

$\boxed{u}$ páska pro kopírování prvního vybraného podslova z w $O(n)$ - podslovo u

- nalezení druhého podslova - dvě zarážky - obě v libovolném místě ve slově (n pozic zarážky · n pozic zarážky $= n^2$)
- porovnání zda podslova nejsou stejná $O(n^2)$

$\boxed{v}$ páska pro kopírování druhého vybraného podslova z w $O(n)$ - podslovo v

$\boxed{w' \in L}$ páska pro vygenerování slov z L

- ověření že $u,v \in L$ - $O(n^2)$

$O(n^2) \cdot O(n^2) = O(n^4)$

Důkaz: Zřejmě je L přijímáno k-páskovým DTS M_L v čase $O(n^2)$. Nechť $M_{L'}$ je k-páskový DTS přijímající L' v čase $O(n^4)$. Na 1. pásce bude vstupní řetězec $w \in L'$. Na 2 pásku bude vybráno první podslovo z w na 1. pásce (kopírování - $O(n)$) - podslovo u. Dále bude využito na 1. pásce dvou zarážek, tedy se nalezne druhé podslovo v a proběhne porovnání, že $u \neq v$ ($O(n^2)$), na 3. pásku zkopírujeme vybrané podslovo v. 4. páska slouží pro generování slov z L a je provedeno ověření, že obě podslova $u,v \in L$ ($O(n^2)$).

Celkem tedy: $O(n^2) \cdot O(n^2) = O(n^4)$

Hledaná konstanta je tedy $k = 4$.

$\square$