

Volby a volební systémy (NI-VOL), Přednáška č. 7

Složitost problému Swap Bribery

Dušan Knop

Fakulta informačních technologií
České vysoké učení technické v Praze
<https://courses.fit.cvut.cz/NI-VOL/>



LS 2024/2025,

(Verze dokumentu: 8. 4. 2025 10:49)

Volby

Definice

Volby jsou uspořádaná dvojice $\mathcal{E} = (C, V)$, kde

- C je konečná množina **kandidátů (alternativ)**, $|C| = m \geq 2$ a
- V je konečná a neprázdná množina **voličů (agentů)**, $|V| = n$.

Množinu všech lineárních uspořádání na množině C značíme $\mathcal{L}(C)$.

Profil preferencí $\mathcal{P} = (p_{v_1}, \dots, p_{v_n})$ plně charakterizuje volby $\mathcal{E}$.

Volby vítěze(ů) Cílem je naléznout neprázdnou množinu vítězných alternativ $W \subseteq C$. Ideálně samo sebou $|W| = 1$.

Definice

Nechť jsou (C, V) dány. **Volební pravidlo** je funkce

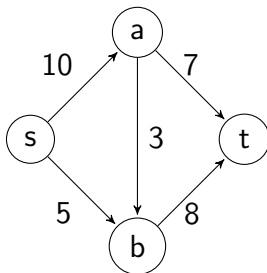
$$f: \mathcal{L}(C)^V \rightarrow 2^C \setminus \emptyset.$$

Co je tok v síti?

Definice

Tok v síti je orientovaný graf $G = (V, E)$ kde:

- Každá hrana (u, v) má **kapacitu** $c(u, v) \geq 0$
- Existuje **zdroj** s a **stok** t
- Tok splňuje $0 \leq f(u, v) \leq c(u, v)$ a
- zákon zachování toku $\sum_{(u,v) \in E} f(u, v) = \sum_{(v,w) \in E} f(v, w)$ pro každý $v \in V$.

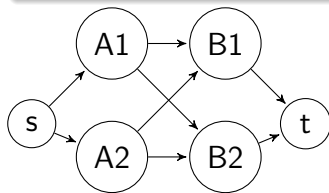


Párování v bipartitním grafu jako tok

Redukce

Problém párování v bipartitním grafu lze řešit pomocí maximálního toku:

- Přidáme zdroj spojený se všemi levými vrcholy
- Přidáme stok spojený se všemi pravými vrcholy
- Všechny hrany mají kapacitu 1



Maximální tok minimální ceny

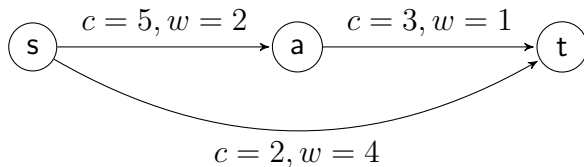
Problém

Najděte maximální tok s nejnižší celkovou cenou, kde každá hrana má:

- Kapacitu $c(u, v)$
- Cenu $w(u, v)$ za jednotku toku

Vlastnosti

- **Složitost:** $O(|V|^2 \cdot |E|)$ pro Successive Shortest Path
- **Aplikace:** Doprava zboží, rozvoz energií, plánování



Toky v sítích – shrnutí

- Toky v sítích modelují mnoho reálných problémů
- Bipartitní párování je speciální případ
- Minimální cena maximálního toku kombinuje dva cíle
- Efektivní algoritmy existují pro různé varianty

Swap Bribery

Swap Bribery

Každý volič má funkci $\pi_v: C \times C \rightarrow \mathbb{R}$ vyjadřující kolik mu musíme zaplatit za prohození kandidátů, liší-li se jejich pozice v halsu právě o 1.

- Mějme hlas $v = a \succ b \succ c \succ d$.
- Můžeme aplikovat swapy ab , bc a cd přímo na hlas v .
- Můžeme ale dostat i hlas $\hat{v} = c \succ b \succ a \succ d$ a po aplikací swapů bc , ac , ab (případně ab , ac , bc).
- Pozorujme, že aplikované swapy jsou vždy pro dvojici hlasů v a $\hat{v}$ určené jednoznačně.

Swap Bribery

Plurality

Hlavní myšlenka je, že každý volič dává právě jeden hlas jednomu kandidátovi.

- Každý kandidát tedy má celkové skóre $1, 2, \dots, n$.
- Úplata hlasu má dvojí efekt:
 - 1 Zvýšíme skóre kandidátovi v jehož prospěch upláčíme.
 - 2 Snížíme skóre kandidátovi, který byl doposud na první pozici v hlasu.
- Obdobně lze rozmyslet i pro pravidlo Veto.
- Toto dvojí chování lze modelovat pomocí toků v sítích s cenami (Faliszewski '07).

3-Approval

Věta (Elkind, Faliszewski, Slinko '09)

Swap Bribery pro 3-Approval je NP-těžké.

Co bychom po dnešku měli znát