6. CVIČENÍ Z ADS2

Toky v sítích – Metoda tří Indů, aplikace toků

V prvních dvou příkladech se budeme zabývat metodou, kterou vymysleli Malhotra, Kumar a Maheswari, a proto se jí říká „metoda tří Indů".

D(Rezerva vrcholu**):** Definujme *vstupní rezervu vrcholu* $r^+(v)$ jako součet jeho vstupních rezerv, a *výstupní rezervu vrcholu* $r^-(v)$ jako součet výstupních rezerv. Nakonec definujeme *celkovou rezervu* jako $r(v) = \min(r^+(v), r^-(v))$.

Příklad první Mějme rozvrstvenou a začištěnou síť se zdrojem a stokem. Nechť se náhle stane, že ve vrcholu v síti s nejmenší hodnotou $r(v)$ se objeví $r(v)$ jednotek toku a musíme je přesměrovat do stoku.

1. Dokažte, že je určitě možné přesměrovat všech $r(v)$ jednotek do stoku bez porušení kapacit.
2. Navrhuji tento algoritmus pro přeposlání $r(v)$ jednotek: vezměme vrchol v, vezměme jeho výstupní hrany $e_1, e_2, \ldots$ jednu po druhé a každou pošleme tolik, co to jen jde, dokud v obsahuje nějaký přebytek. Jakmile vyřešíme vrchol v, přecházíme na další vrchol w s kladným přebytkem a opakujeme.
   Dokažte, že i tato metoda bude fungovat.
3. Odhadněte složitost přeposlání metodou v bodu 2. Jak spočítat složitost co nejlépe? Třeba takto: dokažte, že v každém kroku se buď nasytí celá hrana, nebo už je přebytek roven 0. Jak využít toto tvrzení v odhadu složitosti?

Příklad druhý Využijte proceduru z minulého cvičení a doplňte ji tak, aby spočítala maximální tok ve vrstevnaté síti v dobrém čase. (V jakém?)

D(Vrcholové pokrytí**):** Množina vrcholů $X \subseteq V(G)$ je vrcholové pokrytí, pokud pro každou hranou $uv \in E(G)$ platí, že buď $u \in X$, nebo $v \in X$, nebo oboje.

Obvykle se hledá ta nejmenší množina vrcholového pokrytí.

Příklad třetí Použijte klasickou „jednotkovou bipartitní síť" k tomu, abyste našli v bipartitním grafu nejmenší vrcholové pokrytí.

Příklad čtvrtý Občas se v matematice stane, že informatický přístup k věcem vytvoří hezké důkazy známých faktů. Zkuste použít maximální tok k tomu, abyste dokázali následující větu:

Mějme bipartitní graf G, který má všechny stupně rovny k. Pak G má perfektní párování.

Hint: Už víme, že má-li graf celočíselné kapacity, tak má i celočíselný maximální tok. Zkuste to použít – najděte nejdřív jakýkoli zlomkový tok a pak použijte předchozí větu.

Příklad pátý **Z minula:** Existuje mnoho problémů, které jdou řešit hledáním maximálního toku; asi jste už třeba slyšeli o maximálním párování v bipartitním grafu.

Jeden z nejtěžších problémů, který jde převést na toky, je tenhle, jehož řešení si nikdy nepamatuji:

„*Máme doly $\mathcal{D} = \{D_1, D_2, \ldots, D_n\}$ a továrny $\mathcal{T} = \{T_1, \ldots, T_m\}$. Otevření jednoho dolu nás stojí d_i korun, a každá továrna nám vydělá t_j korun. Nicméně každá továrna j vyžaduje podmnožinu dolů $P_j \subseteq \mathcal{D}$, a vydělá oněch t_j korun jen tehdy, pokud jsme všechny její doly otevřeli. (Jeden důl stačí otevřít jen jednou, pak je použitelný pro všechny továrny, které ho potřebují.)*

Našim úkolem je spočítat, které doly otevřít, abychom vydělali co nejvíce peněz. "

Pomozte mi připomenout si, jak vyřešit tento problém pomocí hledání toků v sítích!

Hint: Jediné, co si pamatuji, je že je chytřejší dívat se na minimální řez a rozmyslet si, proč mi spočítá, co chci.

Příklad šestý

- Co kdyby síť měla více zdrojů a více stoků? Ukažte, jak to simulovat pomocí toků v síti.
- No a co kdyby neměla síť žádné zdroje a stoky, a místo toho by každý vrchol měl *požadovaný přebytek* nebo *požadovaný záporný přebytek*, a nás zajímalo jen to, jestli existuje Kirchhofova cirkulace, která respektuje všechny přebytky? Ukažte, jak to simulovat pomocí toků v síti.
- Nakonec: co kdybychom kromě cirkulací a požadovaných přebytků měli ještě dolní odhady na hranách $d(e)$ – požadovaný dolní odhad na tok, který jimi musí jít? Ukažte, jak to simulovat pomocí toků v síti.

Příklad sedmý

Ještě jedna hezká matematická věta, co jde dokázat informaticky:

Nechť máme matici s reálnými čísly A. Napočítejme si součty řadků r_i a součty sloupců v_j. Potom existuje zaokrouhlení všech čísel v A, čísel r_i a čísel v_j buď nahoru nebo dolů (každé číslo může být jinak zakrouhleno) takové, že součet zaoukrouhleného řádku bude přesně zaokrouhlené číslo r_i.

Dokažte.

Info na konec: Za 7 dní bude písemka.