4. CVIČENÍ Z ÚVODU DO APROXIMACÍ

hlad a SAT

PŘÍKLAD PRVNÍ Uvažujme ROZVRHOVÁNÍ SE ZÁVISLOSTMI: rozvrhujeme úlohy různých délek na m počítačů (m součástí vstupu), ale navíc máme závislosti mezi úlohami. Přesněji, na úlohách je definovaný acyklický graf a úlohu je možné zahájit až ve chvíli, kdy jsou všechny úlohy jí předcházející v grafu dokončeny.

1. Dokažte, že následující dolní odhad na optimum je pravdivý:
   „$OPT \geq$ délka libovolného řetězu úloh, kde řetěz chápeme jako orientovanou cestu v grafu závislostí. Jeho délka je pak součet délek úloh na řetězu."
2. Navrhněte hladový algoritmus a dokažte, že je 2-aproximační.

PŘÍKLAD DRUHÝ Mějme klasický NP-těžký PROBLÉM BATOHU, to jest n objektů $a_1, \ldots, a_n$, kde každý objekt a_i má váhu w_i a cenu c_i, a my máme batoh o nosnosti B.

1. Rozmyslete si, proč „naivně hladový algoritmus", to jest „vložím nejdražší objekt, co se vejde, do batohu, a pokračuji dále", je špatným.
2. OK, tak zkusme tento algoritmus: setřiďme si objekty podle „hustoty" (poměr cena/velikost), procházejme je od nejhustšího a berme jen ty, které se vejdou. Prozradím vám, že i tento algoritmus nebude mít dobrý výsledek. Najděte vstup, kde selže.
3. Navrhněte už funkčni 2-aproximační algoritmus pro tento problém. Algoritmus nemusí nutně být hladový.
   Hint: Když procházíte objekty podle hustoty, může se vám stát, že objekt P se vám do batohu nevejde s věcmi, co už tam máte. Co je chytré udělat potom?

PŘÍKLAD TŘETÍ Na minulém cvičení jsme formulovali pravděpodobnostní aproximační algoritmus pro MAX SAT, který byl efektivní pro klauzule délky 2 a delší, ale byl jen 1/2-aproximační v instancích, které obsahovaly mnoho klauzulí (x_i) nebo $(\neg x_j)$.

Pojďme nahlédnout, že můžeme předpokládat trochu hezčí vstup:

1. Dokažte, že máme-li c-aproximační algoritmus pro MAX SAT na vstupech, které neobsahují negativní jednoklauzule $(\neg x_i)$, tak ho umíme přetvořit na c-aproximační algoritmus pro MAX SAT s všemi přípustnými vstupy.
2. Dokažte, že stejný převod platí i pro VÁŽENÝ MAX SAT, kde každá klauzule má váhu w_i a my maximalizujeme vážený součet splněných klauzulí $\max \sum_i w_i C_i$.

PŘÍKLAD ČTVRTÝ Z předchozího příkladu víme, že nyní řešíme MAX SAT na vstupech, které neobsahují negativní jednoklauzule $(\neg x_i)$. To by nám mohlo pomoci v lepším nastavení pravděpodobnosti p tak, aby se zvýšila pravděpodobnost splnění klauzule:

1. Dokažte, že pokud x_i je nastavena nezávisle náhodně na 1 s pravděpodobností $p > \frac{1}{2}$, tak pravděpodobnost splnění libovolné klazule obsahující literál x_i je alespoň $\min(p, 1 - p^2)$.
2. Zvolte vhodné p a dokončete analýzu naznačeného pravděpodobnostního algoritmu pro MAX SAT.