30. dubna 2010

4. úkol z předmětu Složitost

Petr Zemek
xzemek02@stud.fit.vutbr.cz
Fakulta Informačních Technologií, Brno

Příklad 1

Bez újmy na obecnosti lze předpokládat, že velikost formule je dána počtem proměnných (pro n proměnných nedává od určité hranice, závislé pouze na n, smysl formuli dále zvětšovat).

(a) Nechť n označuje počet proměnných v předané formuli. Vygenerování náhodného čísla lze udělat v $O(1)$ (složitost nezávisí na délce formule), takže cyklus *for* má časovou složitost $\Theta(n)$. Otestování pravdivosti zabere $O(n)$. Celková časová složitost funkce *SAT* je tedy $O(n)$.

(b) Ano, lze. Libovolný NTS lze převést na NTS, který má v každém okamžiku přesně dvě nedeterministické volby. To, aby každý výpočet skončil po přesně daném počtu kroků, který je závislý jen od velikosti vstupu, lze u implementace funkce *SAT* na NTS zřejmě také docílit.

(c) Ne, nemůžeme. NTS z bodu (b) není Monte Carlo TS, protože počet přijímajících výpočtů je závislý na velikosti a tvaru formule (přesněji: neexistuje konstanta $0 < k \leq 1$ taková, že pokud $w \in SAT$, tak poměr mezi počtem zamítajících a akceptujících výpočtů NTS z bodu (b) na w je alespoň k, a to nezávisle na velikosti a tvaru formule). Nicméně, pouze na základě této konkrétní funkce ještě nelze říci, že problém SAT nepatří do **RP** – čistě teoreticky totiž může existovat Monte Carlo TS, který rozhoduje SAT.

Příklad 2

Věta 1. *Algoritmus popsaný funkcí* klika *ze zadání není ε-aproximační pro žádné* $0 \leq \varepsilon < 1$.

Důkaz. Sporem. Předpokládejme, že algoritmus je ε-aproximační pro nějaké $0 \leq \varepsilon < 1$.

Nechť G_k, kde $k \geq 4$, označuje neorientovaný graf daný následovně. G_k je sám o sobě nesouvislý a obsahuje dva souvislé grafy, G_k^1 a G_k^2. G_k^1 je graf ve tvaru hvězdy o $k+1$ vrcholech (obsahuje tedy jeden vrchol stupně k a k vrcholů stupně 1). G_k^2 je úplný graf o $k-1$ vrcholech, tedy tvoří kliku o velikosti $k-1$. Všimněte si, že maximální stupeň vrcholu v G_k^2 je $k-1$, tedy méně, než v G_k^1.

Pokud dáme G_k na vstup funkci *klika*, dostaneme vždy jako výsledek kliku o velikosti 2, nezávisle na hodnotě k (v inicializaci se do *max* se přiřadí uzel s největším stupněm, tj. k, který je z G_k^1, a po jednom kroku cyklu se funkce ukončí). Optimální velikost kliky v G_k je ale $k-1$, kterou dává G_k^2. Jelikož předpokládáme $k \geq 4$, tak optimální řešení se od nalezeného liší vždy o $k-3$.

Pokud tedy budeme zvyšovat k, tak nám bude růst relativní chyba, což je spor s předpokladem, že algoritmus je ε-aproximační pro nějaké $0 \leq \varepsilon < 1$ (pokud je algoritmus ε-aproximační, tak nesmí růst relativní chyba při zvyšování velikosti vstupu). Tudíž, algoritmus není ε-aproximační pro žádné $0 \leq \varepsilon < 1$. □