

Bonyelm invitational

Bonyolultság elmélet gyakorlat

Toffalini Leonardo

Feladat 1

Mutast meg, hogy ha a \mathbf{P} és az \mathbf{NP} osztályok nincsenek összetapadva (azaz nem egyenlőek), akkor létezik olyan $L \in \mathbf{NP} \setminus \mathbf{P}$ nyelv, ami nem \mathbf{NP} -teljes!

Megoldás: Legyen A tetszőleges \mathbf{NP} -teljes nyelv, például $A = \text{SAT}$. Ekkor legyen $L = \{x0^{f(|x|)} : x \in A\}$, ahol $f(n) = n^{\log n}$.

Azt állítom, hogy $L \in \mathbf{NP}$ és $L \notin \mathbf{P}$ és $L \notin \mathbf{NPC}$.

Érthetően $L \in \mathbf{NP}$, mivel egy NDTG könnyen felismeri egy inputról, hogy $x^{f(|x|)}$ alakú és kinyeri x -et az inputból. Mivel $x \in A$ és $A \in \mathbf{NP}$, ezért van olyan NDTG ami felismeri A -t, hát akkor a mi NDTG-énk is tudja szimulálni ezt polinomiális időben mind.

$L \notin \mathbf{P}$, mivel ha \mathbf{P} -ben lenne, akkor az eredeti A nyelvet is fel tudnánk ismerni \mathbf{P} -ben, viszont feltettük, hogy $\mathbf{P} \neq \mathbf{NP}$ és $A \in \mathbf{NPC}$, ezért ez ellentmondás.

$L \notin \mathbf{NPC}$, mivel ha \mathbf{NP} -teljes volna, akkor létezne polinomiális visszavezetés A -ról, mivel $A \in \mathbf{NPC}$. Az $R : A \rightarrow L$ visszavezetés egy $w \in A$ szót átalakít egy $y \in L$ szóra polinomiális időben. Viszont $y = x0^{f(|x|)}$, ahol $x \in A$, így $|y| = |x| + f(|x|)$, ami nem polinomiális $|x|$ -ben. Így nem létezik polinomiális visszavezetés A -ról L -re, és így $L \notin \mathbf{NPC}$. \square

Feladat 2

Mutasd meg, hogy $\mathbf{P}^{\Sigma_k} = \Sigma_{k+1} \cap \Pi_{k+1}$

Lemma 0.1

$$\Sigma_{k+1} = \mathbf{NP}^{\Sigma_k}$$

$$\Pi_{k+1} = \mathbf{coNP}^{\Sigma_k}$$

Biz:

$$\Sigma_{k+1} \subseteq \mathbf{NP}^{\Sigma_k}$$

Definíció szerint: $L \in \Sigma_{k+1} \iff \exists P$ polinomiális reláció, hogy $\exists y_1 \forall y_2 \dots Q y_{k+1} P(x, y_1, \dots, y_{k+1})$.

Egy NDTG nem determinisztikusan meg tudja tippelni y_1 -et, így annyi marad, hogy $\forall y_2 \dots Q y_{k+1} P(x, y_1, \dots, y_{k+1})$, ami pont Π_k alakú, tehát $\Sigma_{k+1} \subseteq \mathbf{NP}^{\Pi_k}$.

Viszont tudjuk, hogy $\Pi_k = \mathbf{co}(\Sigma_k)$ a De Morgan azonosság miatt, viszont egy NDTG-nek nem változtat, hogy egy $\mathbf{co}(A)$ vagy A orákuluma van. Így $\Sigma_{k+1} \subseteq \mathbf{NP}^{\Sigma_k}$, ahogyan kívántuk.

$$\mathbf{NP}^{\Sigma_k} \subseteq \Sigma_{k+1}$$

Egy NDTG egy Σ_k orákuluma nem determinisztikusan meg tud tippelni egy polinomiális tanut z , hogy $\exists y_1 \forall y_2 \dots Q y_k P(x, z, y_1, \dots, y_k)$. Ez a nem determinisztikus tippelés ekvivalens a következő formulával $\exists z \exists y_1 \forall y_2 \exists y_3 \dots Q y_k P(x, z, y_1, y_2, \dots, y_k)$. Itt az első kettő egzisztenciális kvantort össze

tudjuk olvasztani, és így az kapjuk, hogy $\exists(z, y_1) \forall y_2 \exists y_3 Q y_k P(x, z, y_1, y_2, \dots, y_k)$, ami pont egy Σ_k formula, és mivel $\Sigma_k \subseteq \Sigma_{k+1}$ így készen vagyunk.

A második állítást a lemmában hasonló módon lehet bizonyítani. □

Megoldás: Az előző két lemma segítségével már csak azt kell bizonyítani, hogy

$$\mathbf{P}^{\Sigma_k} = \mathbf{NP}^{\Sigma_k} \cap \mathbf{coNP}^{\Sigma_k}.$$

Fenomenális lenne ezt bizonyítani, mert az azt jelentené, hogy $k = 0$ -ra megoldottuk a hosszú idők óta nyílt kérdést $\mathbf{P} \stackrel{?}{=} \mathbf{NP} \cap \mathbf{coNP}$.

Mindenesetre, annyit tudunk bizonyítani, hogy

$$\mathbf{P}^{\Sigma_k} \subseteq \mathbf{NP}^{\Sigma_k} \cap \mathbf{coNP}^{\Sigma_k}.$$

Ha $L \in \mathbf{P}^{\Sigma_k}$, akkor $L \in \mathbf{NP}^{\Sigma_k}$, mivel egy nemdegeterminisztikus TG tud működni mint egy determinisztikus TG.

Továbbá, ha $L \in \mathbf{P}^{\Sigma_k}$, akkor $\bar{L} \in \mathbf{P}^{\Sigma_k} \implies L \in \mathbf{coNP}^{\Sigma_k}$. □

Feladat 3

Mutassuk meg, hogy ha $\mathbf{BPP} = \mathbf{RP} \cup \mathbf{coRP}$, akkor $\mathbf{BPP} = \mathbf{ZPP}$.

Megoldás: Ha $\mathbf{BPP} = \mathbf{RP} \cup \mathbf{coRP}$, akkor

$$\mathbf{BPP} = \mathbf{co}(\mathbf{BPP}) = \mathbf{co}(\mathbf{RP} \cup \mathbf{coRP}) = \mathbf{coRP} \cap \mathbf{RP} = \mathbf{ZPP}.$$

□

Feladat 4

Mutassuk meg, hogy a módosított GG játékban eldönteni, hogy ki nyer PSPACE teljes.

Megoldás: A visszavezetés nagyon hasonló lesz az előadáson látott $\text{TQBF} \propto \text{GG}$ visszavezetésre. Annyi kell csak változtatnunk az ötleten, hogy amikor Bob lép, akkor ne tudja megváltoztatni a kvantált formula jelentését, amit kódolunk a GG gráf struktúrájával.

Ezt úgy fogjuk csinálni, hogy a gráf első fázisában, ami kódolja a kvantorokat és a változók értékadásait, ott Bob mindig csak olyan élen léphet, aminek a végén egy 1 befokú csúcs van.

Láthatjuk, hogy csak annyit kell csinálni, hogy a standard rombusz Figure 1, helyett egy széthúzott rombuszt kell adni Bob-nak. És Bob a pirosra színezett éleken léphet.

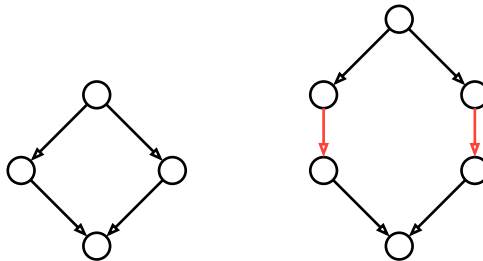


Figure 1: Rombusz és széthúzott rombusz

□

Feladat 5

1. Mennyi $D^{(3)}(EQ)$?
2. Mutass egy függvényt, amire $\Omega(n)$ bit kommunikációja szükséges.

Megoldás:

1.

Protokoll:

1. Ha Alíz azt látja, hogy $y = z$, akkor küld egy 1 bitet, különben 0-át küld.
2. Innentől már Bob és Charlie is tudják $f(x, y, z)$ értékét.

Mivel csak akkor lehet a három szám egyenlő, ha Alíz azt mondta, hogy $y = z$ és Bob látja, hogy $x = z$. Alíz üzenete alapján Bob már tudja, hogy az ő száma megegyezik Charlie-ével, továbbá látja, hogy Charlie száma megegyezik Alízéval.

Így a protokoll költsége 1. Tehát $D^{(3)}(EQ) = 1$.

2.

Hasonlóképpen definiáljuk a kommunikációs mátrixot mint a standard determinisztikus két játékos esetben. Azaz legyen A-nak, B-nek, és C-nek egy-egy dimenziója. Tehát legyen M a kommunikációs mátrix, ahol $M[x, y, z] = 1$, pontosan akkor, ha $f(x, y, z) = 1$.

Figyeljük meg, hogy ezen játékszabályok szerint egy játékos tudja a másik két játékos értékét, tehát olyan mintha a kommunikációs mátrixnak egy oldalára vett vetületét látja csak.

Ötlet: Találjunk ki egy olyan függvényt aminek a vetülete mindegyik oldalra egy csupa 1-es mátrix. Továbbá, legyen ez a függvény a lehető *legritkább*. Azaz, ha lehet legyen olyan, hogy bármelyik oldal felől nézzük, mindegyik $1 \times 1 \times 2^n$ -es vektorban csak egy darab 1-es.

$$\forall i, j \in [2^n] \quad \sum_{k=1}^{2^k} M[i, j, k] = 1$$

Képzeljünk el egy $2^n \times 2^n$ -es táblázatot, ami olyan mint egy sudoku, csak elfelejtkezünk a négyzetekről és csak a sorokat és oszlopokat figyeljük. Azaz, mindegyik sorban és mindegyik oszlopban mindegyik szám 1-től 2^n pontosan egyszer szerepel.

Fixáljunk le egy S helyesen kitöltött ilyen sudokut táblát és legyen az az $f(i, j, k)$ függvény, hogy $f(i, j, k) = 1$ pontosan akkor, ha $S[i, j] = k$.

A sudoku szabályaink szerint így tényleg igaz, hogy bármelyik oldalról nézve csupa 1-es mátrixot látunk, és a lehető legritkább.

Azt állítom, hogy ehhez a függvényhez legalább n bit kommunikációja szükséges.

Mivel mindegyik oldalról csupa 1-et lát mindegyik játékos, ez azt jelenti, hogy nem kapnak semmi új információt az alapján, hogy látják a másik két játékos számait. Továbbá, egy játékos ha ránéz a saját vetületére nem tudja a mélységét egyik 1-esnek sem. Tehát ahhoz, hogy bármelyik játékos tudja a saját vetületének a mélységét, a saját számát ismernie kell. Más szóval, legalább n bit kommunikációja szükséges.

□