ALGORITMUSOK ÉS BONYOLULTSÁGELMÉLET

Matematika MSc hallgatók számára

10. Előadás

Előadó: Hajnal Péter
Jegyzetelő: Hajnal Péter

2012. Április 18.

Röviden tekintsük át, hogy jelenleg mit tudunk mondani a különböző bonyolultsági osztályok egymás közötti viszonyairól.

$$\mathcal{L} \subseteq \mathcal{NL} = co\,\mathcal{NL} \subseteq \begin{array}{|ccc|} \hline & \mathcal{NP} & \mathcal{NPSPACE} \\ \subsetneq & \subsetneq & \| \\ \mathcal{P} & & \mathcal{PSPACE} \\ \subsetneq & \subsetneq & \| \\ & co\,\mathcal{NP} & co\,\mathcal{NPSPACE} \\ \hline \end{array} \subseteq \mathcal{EXP}$$

Az ábrának nyilván a bekeretezett része a legérdekesebb és ennél többet a megjelölt osztályokról nem tudunk. Még azt sem tekinthetjük kizártnak, hogy $\mathcal{P} = \mathcal{PSPACE}$.

1. $\mathcal{P}$ és $\mathcal{PSPACE}$ közötti problémák

Emlékeztető. Akkor mondjuk, hogy $L \in \mathcal{NP}$, ha létezik olyan tanúszalagos (nemdeterminisztikus) T Turing-gép, ami úgy működik, hogy pontosan az L-beli ω szavakhoz létezik τ tanúszalag-tartalom úgy, hogy $T(\omega, \tau)$ ELFOGAD állapottal és ω hosszában polinomiális időben leáll. Formálisan:

$$\begin{array}{rcl} L \in \mathcal{NP} & \iff & \text{van olyan } T \text{ nem-determinisztikus Turing-gép, hogy} \\ & & T \text{ polinomiális } |\omega|\text{-ban és} \\ & & \omega \in L \quad \Leftrightarrow \quad \exists \tau: \ \ T(\omega,\tau) \text{ elfogadó} \end{array}$$

Definíció. Akkor mondjuk, hogy $L \in co\,\mathcal{NP}$, azaz $\overline{L} \in \mathcal{NP}$, ha létezik olyan tanúszalagos (nemdeterminisztikus) T Turing-gép, ami úgy működik, hogy pontosan az L-beli ω szavakhoz nem létezik τ tanúszalag-tartalom úgy, hogy $T(\omega, \tau)$ ELFOGAD állapottal áll le ω hosszában polinomiális időben. Formálisan:

$$\begin{array}{rcl} L \in co\,\mathcal{NP} & \iff & \text{van olyan } T \text{ nem-determinisztikus Turing-gép, hogy} \\ & & T \text{ polinomiális } |\omega|\text{-ban és} \\ & & \omega \in L \quad \Leftrightarrow \quad \forall \tau: \ \ T(\omega,\tau) \text{ elvető} \\ & \iff & \text{van olyan } \widetilde{T} \text{ nem-determinisztikus Turing-gép, hogy} \\ & & \widetilde{T} \text{ polinomiális } |\omega|\text{-ban és} \\ & & \omega \in L \quad \Leftrightarrow \quad \forall \tau: \ \ T(\omega,\tau) \text{ elfogadó} \end{array}$$

1.1. Példák

A következőkben példákat mutatunk a fenti ábra bekeretezett részéből. Az első példához szükségünk lesz a konjuktív normálformákról tanultakra, ezért elevenítsük fel ezeket.

Definíció. A φ formula konjuktív normálforma (CNF), ha előáll

$$\varphi = \bigwedge_{i=1}^{l} C_i$$

alakban, ahol minden C_i klóz literálok diszjunkciója, azaz

$$C_i = \bigvee_{j=1}^{l_i} l_j^{(i)}.$$

A φ CNF felfogható úgy is, mint egy $\{C_i\}$ klózhalmaz, és minden klóz értelmezhető úgy, mint egy $\{l_i\}$ literálhalmaz. Ez utóbbi meggondolás alapján a φ CNF hosszán a $|\varphi| = \Sigma_i |C_i|$ számot értjük. Például, ha $\varphi = (x \vee y \vee \neg t) \wedge (\neg x \vee z) \wedge (z \vee \neg w \vee \neg u) \wedge u$, akkor $|\varphi| = 3 + 2 + 3 + 1 = 9$.

Példa. Legyen OPT-CNF az a probléma, hogy egy φ CNF-fel kapcsolatban azt akarjuk igazolni, hogy a φ által leírt Boole-függvényt nem tudjuk $|\varphi|$-nál kisebb méretű CNF-fel megfogalmazni, azaz a φ a lehető legrövidebben írja le a Boole-függvényt.

- OPT-CNF $\in \mathcal{PSPACE}$

  Ennek a bizonyítására adható egy olyan naív algoritmus, mellyel megvizsgálunk minden szóba jöhető lehetőséget. Így tulajdonképpen exponenciálisan sok lehetőséget írunk a munkaszalagra, viszont ezek felülírhatók, ezért elég a polinomtár. Csak azokból a változókból építkezünk, amik φ-ben is vannak, kiválasztunk bizonyos kisebb méretű literál-részhalmazokat, és az összes értékadásra teszteljük, és azt kell tapasztalnunk, hogy minden $|\varphi|$-nél kisebb méretű ψ CNF esetén van olyan kiértékelés, amire ψ és φ értéke különbözik.

- A probléma formalizált változata a következő:

$$\ulcorner \varphi \urcorner \in \text{OPT-CNF} \iff \forall \ulcorner \psi \urcorner \text{ CNF-hez } \exists \ulcorner \upsilon \urcorner \text{ kiértékelés úgy, hogy } (|\psi| < |\varphi| \Rightarrow \psi(\upsilon) \neq \varphi(\upsilon)).$$

  Emlékezzünk vissza az fejezet elején felelevenített definíciókra. Ott $\exists\tau$ és $\forall\tau$ szerepel. Ez is hasonló probléma, csak itt két kvantorok szerepel/alternál.

- A probléma „valahol $\mathcal{P}$ és $\mathcal{PSPACE}$ között” található.

Példa. Legyen PONTOS_FGTLEN_CSÚCSOK az a probléma, hogy egy adott G gráfról és egy adott t számról azt akarjuk igazolni, hogy a G gráfban lévő független csúcsok maximális száma pontosan t, azaz

$$\text{PFC} := \text{PONTOS\_FGTLEN\_CSÚCSOK} = \{\ulcorner G, t \urcorner : \ \alpha(G) = t\}.$$

- A probléma formalizált változata a következő:

$$\ulcorner G, t \urcorner \in \text{PFC} \iff (\exists I \subseteq V(G) : \ I \text{ független és } t = |I|) \bigwedge (\forall I \subseteq V(G) : \ I \text{ független} \rightarrow |I| \leq t).$$

Azt vehetjük észre, hogy a konjukció bal oldalán álló probléma $\mathcal{NP}$-beli, a jobb oldalán álló pedig *co* $\mathcal{NP}$-beli. Mindkét kvantorra szükségünk van, továbbá

$$\text{PONTOS\_FGTLEN\_CSÚCSOK} \in \mathcal{PSPACE},$$

viszont a PFC $\in \mathcal{NP}$ és PFC $\in$ *co* $\mathcal{NP}$ megállapítások közül egyiket sem mondhatjuk biztosnak. (A probléma ugyan $\mathcal{PSPACE}$-en belül van, viszont úgy érezzük, hogy $\mathcal{NP}$-n és *co* $\mathcal{NP}$-n kívül.)

Definíció. Legyen $\mathcal{H}$ egy halmazrendszer V felett, azaz $\mathcal{H} \subseteq \mathcal{P}(V)$. Bevezetjük a $\mathcal{H}$-nak az A-beli nyomát:

$$\text{Trace}_A\mathcal{H} = \{A \cap E :\ E \in \mathcal{H}\},\ \text{ahol } A \subseteq V.$$

Nyilván $\text{Trace}_A\mathcal{H} \subseteq \mathcal{P}(A)$. Az A ponthalmazt telítettnek nevezzük, ha $\text{Trace}_A\mathcal{H} = \mathcal{P}(A)$. Most már definiálhatjuk a Vapnyik—Cservonyenszki-dimenziót:

$$\text{VCs-dim}\mathcal{H} = \max\{|A| :\ A \text{ telített}\}.$$

Észrevétel. A Vapnyik—Cservonyenszki-dimenzió a matematika sok területén előfordul, például a geometriában, kombinatorikában, mesterséges intelligenciában, illetve a statisztikában is. A dimenzió névadói is statisztikusok voltak.

Példa. Legyen VCS-DIM az a nyelv, hogy egy adott $\mathcal{H}$ halmazrendszerről és egy adott k számról azt akarjuk igazolni, hogy $\mathcal{H}$ VCs-dimenziója legalább k, azaz

$$\text{VCS-DIM} = \{\ulcorner V, \mathcal{H}, k\urcorner :\ \text{VCs-dim}\mathcal{H} \geq k\}.$$

- A halmazrendszereket tömör kódolással kódoljuk.
- A $\ulcorner V, \mathcal{H}, k\urcorner$ hármasban a $(V, \mathcal{H})$ pár tulajdonképpen felfogható egy $C_{\mathcal{H}}$ hálózatnak, ahol a $v \in V$ csúcsok $\log|V|$ biten kódolhatók, az $E \in \mathcal{H}$ élek pedig $\log|\mathcal{H}|$ biten, és $C_{\mathcal{H}}$ kiszámolja, hogy v eleme-e E-nek. Ilyen kódolásnál a véletlen séta generálása rendkívül gyorsan megy, még egy milliószor milliós szomszédségi mátrix-szal megadható gráf esetén is.
- A probléma formalizált változata a következő:

$$\ulcorner V, \mathcal{H}, k\urcorner \in \text{VCS-DIM} \Longleftrightarrow \exists A\, \forall R\, \exists E\ \left(|A| = k \bigwedge (R \subseteq A \Rightarrow R = A \cap E)\right).$$

- VCS-DIM $\in \mathcal{PSPACE}$.

2. További osztályok, polinomiális hierarchia

Definíció ($\Sigma_i\mathcal{P}$ nyelvosztály).

$$L \in \Sigma_i\mathcal{P} \iff \exists T \text{ polinomiális Turing-gép, melyre } \exists\tau_1, \forall\mu_2, \exists\tau_3, \ldots, QX_i \text{ úgy,}$$
$$\text{hogy } \omega \in L \Leftrightarrow T(\omega, \tau_1, \mu_2, \tau_3, \ldots, X_i) \text{ ELFOGADÓ,}$$
$$\text{ahol } Q = \begin{cases}\exists, & \text{ha } i \text{ páratlan,}\\ \forall, & \text{ha } i \text{ páros.}\end{cases} \text{ és } X_i = \begin{cases}\mu_i, & \text{ha } i \text{ páros,}\\ \tau_i, & \text{ha } i \text{ páratlan,}\end{cases}$$

Definíció $\Pi_i\mathcal{P}$ nyelvosztály.

$$L \in \Pi_i\mathcal{P} \iff \exists T \text{ polinomiális Turing-gép, melyre } \forall\mu_1, \exists\tau_2, \forall\mu_3, \ldots, QY_i \text{ úgy, hogy } \omega \in L \Leftrightarrow T(\omega, \mu_1, \tau_2, \mu_3, \ldots, Y_i) \text{ ELFOGADÓ,}$$

$$\text{ahol } Q = \begin{cases} \exists, & \text{ha } i \text{ páros,} \\ \forall, & \text{ha } i \text{ páratlan.} \end{cases} \quad \text{és } Y_i = \begin{cases} \mu_i, & \text{ha } i \text{ páratlan,} \\ \tau_i, & \text{ha } i \text{ páros,} \end{cases}$$

Észrevétel.

- $\Pi_0\mathcal{P} = \Sigma_0\mathcal{P} = \mathcal{P}$.
- $\Sigma_1\mathcal{P} = \mathcal{NP}$.
- $\Pi_1\mathcal{P} = co\,\mathcal{NP}$.
- $\mathcal{P} = \Sigma_0\mathcal{P} = \Pi_0\mathcal{P} \subseteq \Sigma_1\mathcal{P},\ \Pi_1\mathcal{P} \subseteq \Sigma_2\mathcal{P} \cap \Pi_2\mathcal{P} \subseteq \Sigma_2\mathcal{P},\ \Pi_2\mathcal{P} \subseteq \Sigma_3\mathcal{P} \cap \Pi_3\mathcal{P} \subseteq \Sigma_3\mathcal{P},\ \Pi_3\mathcal{P} \subseteq \ldots \subseteq \Sigma_n\mathcal{P} \cap \Pi_n\mathcal{P} \subseteq \Sigma_n\mathcal{P},\ \Pi_n\mathcal{P} \subseteq \Sigma_{n+1}\mathcal{P} \cap \Pi_{n+1}\mathcal{P} \subseteq \ldots \subseteq \mathcal{PSPACE}$.

Definíció (Polinomiális hierarchia, $\mathcal{PH}$). $\mathcal{PH} = \bigcup_{i\in\mathbb{N}} \Pi_i\mathcal{P} = \bigcup_{i\in\mathbb{N}} \Sigma_i\mathcal{P}$.

Észrevétel. A definíció második egyenlősége tulajdonképpen egy állítás, aminek a bizonyítása egy egyszerű meggondolás, ahol azt kell belátni, hogy a jobb oldali felülről becsüli a bal oldalit, és fordítva.

Definíció (Alternáló polinomiális idő, $\mathcal{AP}$).

$$L \in \mathcal{AP} \iff \exists T \text{ Turing-gép, melyre } \exists\tau_1, \forall\mu_1, \exists\tau_2, \forall\mu_2, \ldots, \exists\tau_N, \forall\mu_N \text{ úgy, hogy } \omega \in L \Leftrightarrow T(\omega, \tau_1, \mu_1, \ldots, \tau_N, \mu_N) \text{ ELFOGADÓ,}$$

$$T \text{ polinomiális } |\omega| \text{-ban.}$$

Megjegyzés. A gép polinomialitásából következik, hogy N is legfeljebb polinomiális. Lehetséges, hogy N függ $|\omega|$-tól. Ez egy többlet a polinomiális hierarchiához képest.

1. Tétel.

1. $\mathcal{PH} \subseteq \mathcal{AP}$.
2. $\mathcal{AP} = \mathcal{PSPACE}$.

Bizonyítás. (Vázlat)

A tétel első része triviális.

A második rész bizonyításához azt kell kihasználnunk, hogy QBF $\in \mathcal{PSPACE}$ és QBF $\in \mathcal{AP}$. Tudjuk, hogy a QBF az $\mathcal{PSPACE}$-teljes és könnyen belátható, hogy a QBF nyelv $\mathcal{AP}$-teljes is. Korábban tanult dolgokat kell alkalmazni, például egy Turing gépet átírni Boole-formulává polinomidőben, megfelelő hálózatot gyártani, ahogy ezeket már korábban megcsináltuk. Továbbá, visszavezetések alkalmazásával kijön, hogy $\mathcal{AP} \subseteq \mathcal{PSPACE}$ és $\mathcal{AP} \supseteq \mathcal{PSPACE}$, és ebből már következik a köztük lévő egyenlőség. ■

3. SAT és hálózatok

Emlékeztető. $L \in_T \mathcal{P}$ akkor, ha létezik $\{C_n\}$ hálózat-sorozat, amely

(i) n inputbitből számol ki egyet,

(ii) csúcsainak száma/mérete kisebb, mint $p(n)$, valamely p polinomra,

(iii) a hálózat input bitjei Σ^ν elemeit kódolják ($n = (\log_2 |\Sigma|) \cdot \nu$, $\omega \in \Sigma^n$ input esetén ω kódja legyen $\lceil \omega \rceil$), továbbá $\omega \in L$ akkor és csakis akkor, ha $C_n(\lceil \omega \rceil) = 1$, azaz C_n az ω kódján az elfogadást/elvetést kódoló bitet számolja ki,

(iv) C_n $\mathcal{L}$-ben megkonstruálható 1^n-ből.

Megjegyzés. A visszafele irány is igaz. Azaz, ha az L nyelvhez van fenti típusú hálózat, akkor ω-ból felírható az ω-t kódoló bitek, ha ezek száma ν, akkor C_ν a hálózatot is ki tudjuk számolni, végül értékelni. Az eredő eljárás mutatja, hogy $L \in \mathcal{P}$.

Definíció. $L \in \mathcal{P}^{\text{nem-uniform}}$ akkor és csakis akkor, ha (i)-(iii) igaz. Azaz $\mathcal{P}$ fenti leírásából csak a hálózat-sorozat „uniformitását” nem követeljük meg.

Megjegyzés. A $\mathcal{P}^{\text{nem-uniform}}$ nyelvosztály „furcsa” az eddigiekhez képest. Nincs benne az eddig ismert legbővebb kiszámíthatósággal kapcsolatos osztályban, $\mathcal{S}$-ben, azaz a felsorolható nyelvek osztályában.

$$\begin{array}{ll} \mathcal{S} & \supseteq \mathcal{D} \supseteq \mathcal{EXPSPACE} \supseteq \ldots \supseteq \mathcal{NP} \supseteq \mathcal{P} \ldots \\ \not\supseteq & \\ \mathcal{P}^{\text{nem-uniform}} & \end{array}$$

Vegyünk egy $\mathcal{B} \subset \mathbb{N}$ bonyolult (nem felsorolható) halmazt. Legyen $L := \{1^\ell : \ell \in \mathcal{B}\}$. L nyilván nincs benne $\mathcal{S}$-ben, hiszen ha ezt a halmazt fel tudnánk sorolni, akkor $\mathcal{B}$ elemeit is felsorolnánk.

$L \subset \{0,1\}^*$ benne van $\mathcal{P}^{\text{nem-uniform}}$-ban: Kétféle inputméret van, $\mathcal{B}$-beli és nem $\mathcal{B}$-beli. Egyik inputmérethez semmit sem kell elfogadni, mert olyan hosszban nincs semmi a nyelvben, a másik inputméret pedig olyan, hogy csupa egyes inputot kell elfogadni. Az első esetben hálózatunk, ez az első változót és negáltját ÉS-sel köti össze. A második esetben a hálózat az összes változót ÉS-sel köti össze. Észrevehetjük, hogy a kétféle hálózat önmagában nagyon egyszerű (mérete polinomiális, sőt lineáris), de nagyon bonyolultan, rapszódikusan változik. Mivel nem követeljük meg, hogy az n inputból megkonstruálható legyen, így beleillik a modellünkbe.

Emlékeztető. Sejtés: $SAT \notin \mathcal{P}$. A sejtés alapja, hogy $SAT \in \mathcal{P}$-ből következik, hogy $\mathcal{P} = \mathcal{NP}$. Ez pedig „váratlan” lenne.

A sejtés átfogalmazása: SAT nem számolható ki $\mathcal{L}$-ben megkonstruálható polinomiális hálózat-sorozattal.

Kérdés: $SAT \in \mathcal{P}^{\text{nem-uniform}}$? **Sejtés:** $SAT \notin \mathcal{P}^{\text{nem-uniform}}$. Azaz SAT polinomiális méretű hálózatok nem-uniform sorozatával sem számolható ki. A sejtés cáfolatának lenne-e bonyolultságelméleti következménye?

Mielőtt a kérdést körüljárnánk nézzük $\mathcal{P}^{\text{nem-uniform}}$ ekvivalens leírásait.

2. Lemma. *A következők ekvivalensek:*

(i) $L \in \mathcal{P}^{\textit{nem-uniform}}$.

(ii) Létezik az ω input hosszában polinomiális T nemdeterminisztikus (tanúszalagos) Turing-gép, és létezik $\{t_n\}_{n=1}^{\infty}$ tanúsorozat, hogy $\omega \in L$ akkor és csakis akkor, ha $T(\omega, t_{|\omega|}) = 1$ (azaz ω-n a számítás ELFOGAD állapotba vezet).

(iii) $L \preceq_{\mathcal{P}}^{\textit{Turing}} S$, *valamely $S \subset \Sigma^*$ alkalmas ritka nyelvre (definíciók alább).*

Definíció. Egy S nyelv ritka, ha benne az n hosszú szavak száma n-ben polinomiális (azaz nagy n-re jóval kevesebb, mint a teljes lehetőségek $|\Sigma|^n$ exponenciális száma). Formálisan van olyan $q(n)$ polinom, hogy $|S \cap \Sigma^n| \leq q(n)$.

Definíció (Turing-redukció). $L \preceq_{\mathcal{P}}^{\text{Turing}} S$ akkor és csak akkor van olyan polinomiális idejű Turing-gép, amely ω L-hez tartozását dönti el. Számolása alatt „?" állapotba mehet a gép, aminek van egy kérdezőszalagja is. A speciális állapotba kerülés egy kérdés: a kérdezőszalagra az előző kérdés óta felírt karaktersorozatról derül ki (egy lépésben) az S-hez tartozás. Az „egy lépésben kiderül" alatt azt értjük, hogy a rákövetkező konfigurációban BENNE-VAN vagy pedig a NINCS-BENNE állapotkomponense lesz, aszerint, hogy a kérdezőszalag tartalma S-beli vagy sem.

Az S-et szokták orákulumnak is nevezni. S tulajdonképpen egy „meg nem írt szubrutin", ami ha valamilyen módon megírható, mondjuk megírása könnyű, akkor L sem lehet nehéz.

Bizonyítás. (i)⇒(ii): Minden n-hez tartozik egy hossz, amelyben bitekkel kódoljuk az Σ^n elemeit. Legyen ez ν. Ha L nem-uniform polinomiális időben van, akkor tartozik hozzá egy $\{C_n\}$ hálózat-sorozat. Legyen t_n a C_ν hálózat kódja. A T Turing-gép a tanú-szalag tartalmából kiolvassa C_ν-t, az input-szalag tartalmát $0-1$ bitekkel kódolja, majd a C_ν-t ennek tartalmán kiértékeli. Ha 1-et kap ELFOGAD, ha 0-t ELVET állapotba kerül.

(ii)⇒(iii): Tegyük fel, hogy létezik egy nem determinisztikus, polinomiális gép az input hosszától függő tanúval. Ezt vissza tudjuk vezetni egy S ritka nyelvre. Ha ismerjük a tanút, $\mathcal{P}$ időben egyszerű a feladat, de ez ritka. Legyen $S := \{(1^k, t_k^{\leq \ell}) : k \in \mathbb{N},\ \ell \leq t_k \text{ hossza}\}$. S a következő alakú: $\underbrace{1\ 1\ 1\ \ldots\ 1}_{k \text{ db}}$; „$t_k$ első ℓ' karaktere" ($\ell' \leq \ell$).

3. Állítás. *Az előbb definiált S nyelv ritka.*

Valóban, a kezdő 1-es blokk elemszáma meghatározza, hogy melyik tanúnak a része következik. Azaz a kezdő 1-es blokk hossza és a teljes hossz ismeretében azonsítani tudjuk S elemét. Ebből adódik S ritkasága.

Adott egy input, aminel L-hez tartozását kell eldönteni. Ehhez egy polinomiális S-orákuliumos gépet írunk le.

Először $t_{|\omega|}$-t határozzuk meg. Beírjuk az $1^{|\omega|}$ karaktert, majd megkérdezzük, hogy $\Sigma := \{\sigma_1, \ldots, \sigma_k\}$ valamely elemét utánaírva S-beli elemet kapunk-e:

$$\begin{aligned} 1^{|\omega|};&\quad \sigma_1 \overset{?}{\in} S \\ 1^{|\omega|};&\quad \sigma_2 \overset{?}{\in} S \\ &\quad\ \vdots \\ 1^{|\omega|};&\quad \sigma_k \overset{?}{\in} S \end{aligned}$$

Ha igent kapunk, akkor megvan a tanúnak az első bitje. Rekurzíven továbbhaladunk.

Ha végigmenve az ábécén végig azt kapjuk, hogy „nem” akkor az eddigi bitek megadják a tanút.

A redukáló algoritmus a fenti módon polinom időben kiszámolja az inputhoz tartozó $t_{|\omega|}$ szót és a feltételben szereplő T algoritmussal eldönti az input L-hez tartozását.

(iii)⇒(i): Konstruálunk egy hálózatsorozatot. Korábban láttuk, hogy egy Turing-gépből (adott inputhossz esetén) hogyan lehet egy a számolását szimuláló hálózatot konstruálni. A gondolatmenetet megismételjük a redukciót bizonyító orákulumos Turing-gépre.

Természetesen a konfigurációban ott szerepel e kérdezőszalag tartalma is. Egyetlen lényeges különbség van. Kezelnünk kell a ‘?’ állapotot. Az időkorlát miatt tudjuk a kérdés hosszát becsülni. Az S-hez tartozás így egy polinomiális méretű S-beli részhalmazhoz tartozással ekvivalens. Ha egy ℓ hosszú k kérdést teszünk fel az orákulumnak, akkor az

$$(s_1 = k) \vee (s_2 = k) \vee \ldots \vee (s_{q(\ell)} = k),$$

ahol s_i az S ritka halmaz „rövid” elemei. A lista ismeretében egy polinomiális méretű, ezt eldöntő hálózat megtervezhető, bármi is legyen az inputméret.

A bizonyítás befejezése a korábbiak alapján standard módon megy. ■

A fent definiált hálózatról nem állíthatjuk/állítjuk, hogy uniform. Egy konfigurációból a következő kiszámolása esetén számolnunk kell, hogy egy S-hez tartozást döntünk el. Ez S ritkasága miatt megtehető, de uniformitásról szó sincs.

⋆ ⋆ ⋆

Előrevetítjük merre haladunk. Tudjuk, hogy $SAT \in \mathcal{P}$-nek nem várt következményei lennének. Mi a helyzet $SAT \in \mathcal{P}^{\text{nem-uniform}}$ esetén?

4. Tétel (Karp—Lipton, Sipser tétele). *Ha $SAT \in \mathcal{P}^{nem\text{-}uniform}$, akkor*

$$\Pi_2\mathcal{P} \subset \Sigma_2\mathcal{P}.$$

A fenti karakterizáció alapján ez a tétel megfogalmazható a következő alakban.

5. Tétel (Karp—Lipton, Sipser tétele). *Ha $SAT \preceq_{\mathcal{P}}^{Turing} S$, ahol S ritka nyelv, akkor $\Pi_2\mathcal{P} \subseteq \Sigma_2\mathcal{P}$.*

A Karp-redukció felfogható Turing-redukcióként is, orákulum nélkül számolunk, egyetlen kérdést generálunk és csak ezt az egy kérdést tehetjük fel S-nek, amire kapott válasz az outputja. Mahaney vette észre, ha a Karp—Lipton, Sipser tételének feltételben szereplő Turing-redukaciónál erősebb Karp-redukciót tesszük fel, akkor erősebb következményt igazolhatunk.

6. Tétel (Mahaney-tétel). *Ha $SAT \preceq_{\mathcal{P}}^{Karp} S$, ahol S ritka nyelv, akkor $\mathcal{P} = \mathcal{NP}$.*

A jövő héten ezeket a tételeket igazoljuk.