

1. Egy $PSPACE$ -teljes probléma

Definíció (kvantifikált Boole-formula probléma, QBF). Legyen a φ Boole-formula a következő alakú: $\varphi(x_1, y_1, x_2, y_2, \dots, x_n, y_n)$. Ekkor a

$$\forall x_1 \exists y_1 \forall x_2 \exists y_2 \dots \forall x_n \exists y_n \varphi(x_1, y_1, x_2, y_2, \dots, x_n, y_n)$$

vagy igaz, vagy hamis. A probléma az, hogy döntsük el, hogy melyik.

1. Tétel. $QBF \in PSPACE$ -teljes, vagyis

- (i) $QBF \in PSPACE$
- (ii) $\forall L \in PSPACE : L \leq_P QBF$

Bizonyítás. Azt látjuk be, hogy $QBF \in NPSPACE$ -teljes. Ez a tétel állítása, hiszen tudjuk, hogy $NPSPACE = PSPACE$.

(i) Könnyen belátható.

(ii) Ahogy általában a tár korlátozott döntéseknél tettük, egy tetszőleges $L \in NPSPACE$ nyelvre az ω L -hez tartozásának döntési feladatát visszavezethetjük az ELÉRHETŐSÉG eldöntésére a $(G_{T,\omega}, START, ELFOGAD)$ input esetén, ahol $G_{T,\omega}$ a redukált konfigurációk gráfja:

$$\omega \in L \Leftrightarrow \text{létezik START-ELFOGAD (irányított) út } G_{T,\omega}\text{-ban.}$$

$G_{T,\omega}$ egy 2^{n^α} pontszámú gráf. Egy csúcspont kódja logaritmikus a csúcsok számában. Vagyis mindegyik csúcspont n^α hosszban kódolható.

Persze nem az ELÉRHETŐSÉG inputját gyártjuk le. Ennek „költsége” $PSPACE$, ami megegyezik L bonyolultságával, így nincs értelme. (A gondolatmenet NL és L viszonyában működött.)

Az „igazi” visszavezetés egy formulát gyárt le. Ehhez szükségünk lesz néhány jelölésre.

Jelölés. Legyen $N := n^\alpha$.

$u \rightsquigarrow v$ jelölje azt, hogy alapgráfunkban u -ből elérhető (irányított értelemben) a v csúcspont.

$u \rightsquigarrow_{\leq i} v$ jelölje azt, hogy alapgráfunkban u -ből legfeljebb i lépéssel elérhetünk v -be.

Esetünkben az alapgráf $G_{T,\omega}$, pontszáma 2^N . Így $u \rightsquigarrow v$ akkor és csak akkor, ha $u \rightsquigarrow_{\leq 2^N} v$.

Az alapötletünk az, hogy felosztjuk az utunkat két részre egy középső k csúccsal, előbb csak egy k csúcsig megyünk el START-ból maximum 2^{N-1} lépésben, majd onnan szintén maximum 2^{N-1} lépésben ELFOGAD-ig.

Eddigi ötleteink formalizálva:

$$\begin{aligned} \omega \in L & \Leftrightarrow \text{START} \overset{\sim}{\leq}_{2^N} \text{ELFOGAD}, \\ \text{START} \overset{\sim}{\leq}_{2^N} \text{ELFOGAD} & \Leftrightarrow \exists_V k (\text{START} \overset{\sim}{\leq}_{2^{N-1}} k) \wedge (k \overset{\sim}{\leq}_{2^{N-1}} \text{ELFOGAD}). \end{aligned}$$

Az összes csúcs (START, k ,ELFOGAD) N (n -ben polinomiális) hosszú bitsorozattal kódolt. A $\exists_V k$ azt jelenti, hogy N hosszú kódszavak (csúcsokat kódoló bitsorozatok) között létezik. Azaz N darab, bitekre vonatkozó egzisztenciális kvantor sorozata.

Gondolhatunk arra, hogy nincs mit tenni csak iterálni kell ezeket az ötleteket. Formulánkban két helyen is szerepel az $\overset{\sim}{\leq}_{2^{N-1}}$ reláció. Iterálásnál (N mélységre van szükségünk) a formula nagysága exponenciálisan nagy lenne. Még egy ötletre van szükségünk. A fenti formulánk ekvivalens azzal, hogy

$$\exists_V k \forall_V c \forall_V c' : (c = \text{START} \wedge c' = k) \vee (c = k \wedge c' = \text{ELFOGAD}) \rightarrow c \overset{\sim}{\leq}_{2^{N-1}} c'.$$

Ezt a gondolatmenetet iterálva már a kapott formula mérete polinomiális lesz. $N = n^\alpha$ iterálás kell és mindegyik csak hozzáad a formulához egy polinomiális hosszú új részt. A bizonyítás befejezéseként csak ellenőrizzük, hogy az iteráció elindul: Az iteráció legmélyén lévő formulák: $u \overset{\sim}{\leq}_1 v$ azt jelentik, hogy ' $u = v$ vagy u -ből vezet el v -hez'. Ez egy egyszerű (kvantor nélküli) Boole-formulával megfogalmazható reláció.

Az $\exists_V v$ jelölés csalóka. v N darab bittel kódolt, ezek mindegyike létezik kvantորral kötött. Úgy tűnik, hogy az alternáló kvantորok feltétele nem teljesül. Nincs probléma. Új változók bevezetésével a kvantորok alternálása megoldható. Ha az új változókat csak kvantifikált formában használjuk, akkor nem befolyásolják a formula logikai értékét. ■

Megjegyzés. A QBF feladat inputjában nem az volt a fontos, hogy a létezik kvantոր után egy minden kvantոր következik és fordítva, hanem az, hogy kvantորok váltakozásának száma tetszőlegesen nagy lehet.

Az

$$\mathcal{NL} \subset \mathcal{P} \subset \mathcal{NP} \subset \mathcal{PSPACE}$$

osztályok mindegyikére találtunk teljes problémát. Azaz olyan problémákat, amelyek tükrözik a nyelvosztály teljes nehézségét. Ha ezekről a teljes problémákról tudunk valami „okosat” mondani, akkor az egész osztályról kapunk fontos információt.

A következőkben a \mathcal{P} és \mathcal{PSPACE} osztályok közötti részt nézzük meg „finomabban”.

2. További osztályok, polinomiális hierarchia

Emlékeztető. Akkor mondjuk, hogy $L \in \mathcal{NP}$, ha létezik olyan tanúszalagos (nem-determinisztikus) 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:

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

Emlékeztető. Akkor mondjuk, hogy $L \in co \mathcal{NP}$, azaz $\bar{L} \in \mathcal{NP}$, 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 állapotban áll le ω hosszában polinomiális időben. Formálisan:

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

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

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

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

$$L \in \Pi_i \mathcal{P} \iff \exists T \text{ polinomiális Turing-gép, melyre } \forall \mu_1, \exists \tau_2, \forall \mu_3, \dots, Q Y_i \text{ úgy,} \\ \text{hogy } \omega \in L \iff T(\omega, \mu_1, \tau_2, \mu_3, \dots, Y_i) \text{ ELFOGADÓ,} \\ \text{ahol } Q = \begin{cases} \exists, & \text{ha } i \text{ páros,} \\ \forall, & \text{ha } i \text{ páratlan.} \end{cases} \text{ és } Y_i = \begin{cases} \mu_i, & \text{ha } i \text{ páratlan,} \\ \tau_i, & \text{ha } i \text{ páros,} \end{cases}$$

Észrevétel.

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

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 megfontolás, ahol azt kell belátni, hogy a jobb oldali felülről becsüli a bal oldalit, és fordítva.

2.1. Példák

A következőkben példákat mutatunk olyan problémákra, amelyek a polinomiális hierarchiából jönnek.

Az első példához szükségünk lesz a konjunktív normálformákról tanultakra, ezért elevenítsük fel ezeket.

Definíció. A φ formula konjunktív 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 $|\varphi| = \sum_i |C_i|$ 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:

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

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:

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

Azt vehetjük észre, hogy a konjunkció 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 $\text{PFC} \in \mathcal{NP}$ és $\text{PFC} \in co\ \mathcal{NP}$ 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} halmazrendszerrel 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:

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

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

3. Alternáló polinomiális idő

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

$$\begin{aligned} L \in \mathcal{AP} \iff & \exists T \text{ Turing-gép, melyre } \exists \tau_1, \forall \mu_1, \exists \tau_2, \forall \mu_2, \dots, \exists \tau_N, \forall \mu_N \text{ úgy,} \\ & \text{hogy } \omega \in L \iff T(\omega, \tau_1, \mu_1, \dots, \tau_N, \mu_N) \text{ ELFOGADÓ,} \\ & T \text{ polinomiális } |\omega| \text{-ban.} \end{aligned}$$

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

2. 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 $\text{QBF} \in \mathcal{PSPACE}$ és $\text{QBF} \in \mathcal{AP}$. 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. ■