Algoritmuselmélet és bonyolultságelmélet MSc hallgatók számára

Hálózatok, $\mathcal{P}$- és $\mathcal{NP}$-teljes problémák

2017. Előadó: Hajnal Péter

1. Hálózatok és egy $\mathcal{P}$-teljes probléma

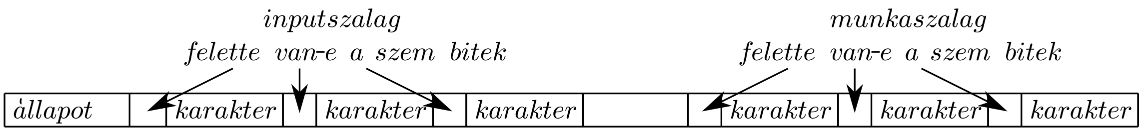
Emlékeztető. $L \in_T \mathcal{TIME}(t(n))/\mathcal{NTIME}(t(n))$ esetén mindig feltesszük, hogy $t(n)$ szép idő függvény, azaz Turing-géppel megvalósítható egy óra, ami n hosszú inputra „$t(n)$ idő után üt”.

$\omega \in \Sigma^n$ inputon T futása a

$$\kappa_0(\omega) \to \kappa_1 \to \kappa_2 \to \ldots \to \kappa_\ell,$$

konfigurációsorozat, ahol $\kappa_0(\omega)$ az ω-hoz tartozó kiinduló konfiguráció, κ_{i+1} a κ_i konfiguráció rákövetkezője, és κ_ℓ az első olyan konfiguráció, ahol az állapot ELFOGAD vagy ELVET. Feltehető, hogy $\ell = t(n)$.

Észrevétel 1. κ konfigurációk kódolhatók bitsorozatokkal.



A fenti ábra (megjegyzéseivel) magáért beszél. A fej pozícióját is kell kódolnunk. Ezt „szétszórtuk”, minden mező raktunk egy bitet. Így nem miden adott hosszúságú sorozat kódol konfigurációt. Az állapotot és karaktereket kódoló blokkkok hossza függ $|S|$, $|\Sigma|$ és $|\Gamma|$ értékétől. Minden esetre konstans sok bit szükséges (ami a Turing-géptől függ). S elemeinek kódolására $\lceil \log_2 |S| \rceil$ bit legendő. Ha $|S|$ nem kettőhatvány, akkor lesznek olyan 0-1 bitsorozatok, amik a megfelelő pozíciókban állnak, de nem kódolnak állapotot. Ennek ellenére könnyű tervezni egy olyan tesztelő hálózatot, ami egy adott (megfelelő hosszú) kódról eldönti, hogy konfigurációt kódol-e.

Megjegyzés (FONTOS). A megállapodát úgy választhatjuk adott n hosszú ω input esetén $\lceil \omega \rceil$ hossza $\alpha_T \cdot n$ legyen. Ha T időigénye legfeljebb $t(n)$, akkor konfigurációk kódjának hossza $\beta_T \cdot t(n)$ legyen. (A konfigurációban a munkaszalag elvágható az első $t(n)$ mező után. Az elhagyott/levágott rész csak érintetlen mezőket tartalmaz.)

A továbbiakban n és a kódolási megállapodás mindig rögzített (ennek megfelelően a megfelelő kódok hossza mindig ismert).

Észrevétel 2. $\lceil \kappa_i \rceil \to \lceil \kappa_i^+ \rceil = \lceil \kappa_{i+1} \rceil$ egyszerű hozzárendelés/függvény.

Az észrevétel állítása matematikailag nem pontos, az „egyszerű” jelző értelmezése nem jól definiált.

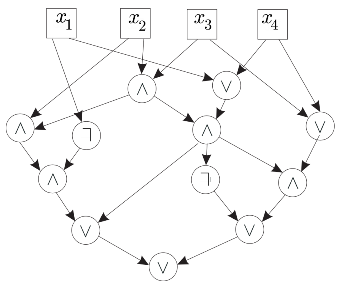
Azt mutatjuk, hogy egyszerűen meghatározható/kiszámolható egy kicsi (polinomiális méretű) hálózat, amely egy konfiguráció kódjából kiszámolja a rákövetkező konfiguráció kódját. Először azonban lássuk a szükséges fogalmakat.

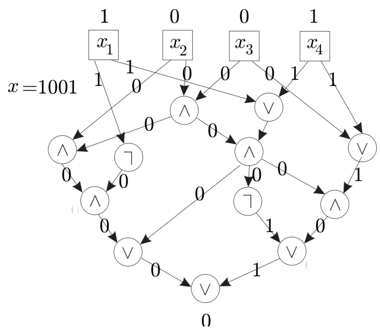
Definíció. Egy hálózat egy $\overrightarrow{G}$ irányított gráf, amely nem tartalmaz irányított kört (azaz lerajzolható úgy, hogy minden él „lefelé haladjon”). Minden csúcs befoka $0, 1$ vagy 2. A 0 befokú csúcsok neve inputcsúcs. Legye I az input csúcsok halmaza. A nem inputcsúcsokra mint kapuk hívatkozunk. A kapuk halmaza $K = V(\overrightarrow{G}) - I$. Speciális csúcsot, esetleg csúcsokat nevezünk ki, amelykre mint output csúcsok hívatkozunk.

Legyen $\ell_I : I \to \{x_1, x_2, \ldots, x_n\} \cup \{0, 1\}$ egy címkézéese az output csúcsoknak. Legyen $\ell_K : K \to \{\neg, \vee, \wedge\}$ egy címkézése a kapuknak, amelyre teljesül, hogy egy kapunknak akkor és csak akkor $\neg$ a címkéje, ha befoka 1. Legyen $\ell = (\ell_I, \ell_K)$ az összes csúcsot címkéző függvény.

$(\overrightarrow{G}, \ell)$ címkézett irányított gráfot hálózatnak nevezzük.

A fenti egy statikus számítási modell. Lássuk a fenti modell dinamikus változatát. Egy hálózat kiszámol egy Boole-függvényt a következő módon. Az $\{x_1, x_2, \ldots, x_n\}$ változók egy kiértékelését kiterjesztjük a kapuk egy kiértékeléséve: Feltesszük, hogy hálózatunkat úgy rajzoljuk le, hogy minden él lefelé haladjon. A kapuk kiértékelése fentről lefelé halad.

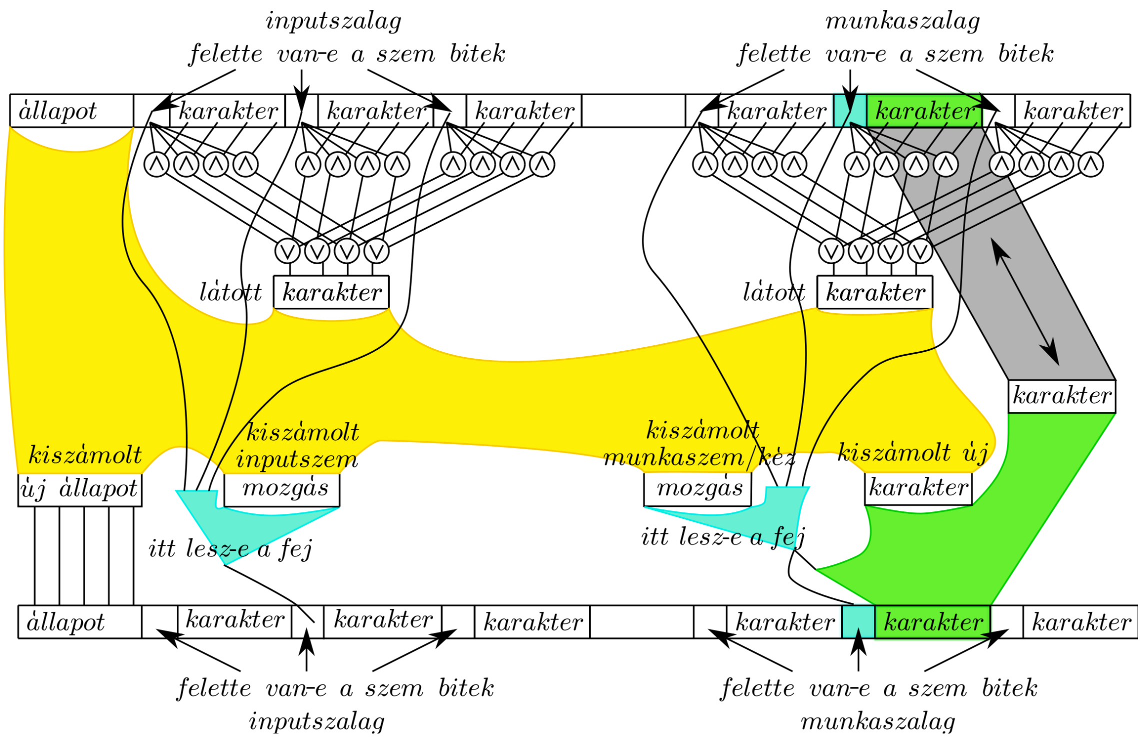




Ha egy kapuhoz jutunk, akkor azok a kapuk, ahonnan él vezet hozzá már kiértékeltek, azaz egy kiszámolt bitet rendeltünk hozzá. Az aktuális kapu által kiszámolt bit az hozzá vezető éleken áramló bit és a kapu címkéje alapján természetesen értelmezhető. A hálózat által kiszámolt bitsorozat az otuput csúcs(ok) által kiszámolt bit(ek).

Azaz a $\mathcal{C}$ hálózat egy $f_{\mathcal{C}}$ Boole-függvényt számol ki/valósít meg.

A fentiek „matematizálása”: Egy konfigurációt kódoló bitsorozatból egy egyszerűen leírható, kicsi hálózat kiszámolja a rákövetkező konfiguráció kódját. A konstrukciónk egyszerű, de sok esetlegességet, megállapodást követel. Egy formális leírás helyett egy példán szemléltetjük milyen ötletek vezethetnek el egy megoldáshoz.



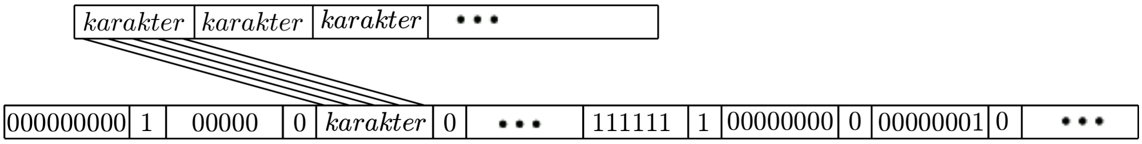
Egy mezőnél a szem/kéz-ott-van-e bitet és a mező tartalmát kódoló bitek mindegyikét össze-és-eljük. A kapott bitsorozat vagy a csupa 0 (ha nincs ott a szem/kéz, mindent 0-val és-eltünk) vagy a látott karakter kódja (ha ott a szem/kéz). Az inputszalag mezőinél így kapott bitsorozatok mindegyikének első, majd második és így tovább karakterét össze-vagy-olva kapjuk az inputszalagon látott karakter kódját. Hasonlóan számolhatunk a munkaszalagnál.

A sárga területben egy összetettebb számolás történik: konstans sok bitbből számolunk ki konstans sok bitet (a konstanok függnek a Turing-géptől). Konkrét kidolgozása az átmenetifüggvénytől függ. Nem okozhat problémát akkor sem, ha az egyes bitek függését a nyilvánvaló DNF formula alapján írjuk fel. Ekkor is konstans sok kapuval dolgozva megoldjuk a feladatunkat.

A világos kék területen számolódik kis a fej pozícióját leíró egyik bit. Ez attól függ, hogy a fej ott volt-e vagy valamelyik szomszéd felett állt, illetve merre írja elő mozgását az átmeneti függvény. A hálózatunk ezen részét konkrétan felírhatnánk, de ez a munka felesleges az előző megjegyzésünk alapján. Ez a kék rész minden fej-pozíció-bithez ott van. Az átláthatóság kedvéért rajzoltuk fel csak egyszer az input- és egyszer A munkaszalaghoz. A zöld területen számoljuk ki a munkaszalag új tartalmát. Minden munkaszalag mezőhöz tartozik egy ilyen zöld blokk (egyszerűség kedvéért csak egyet tüntettünk fel). Az új karaktertől, a régitől és attól az információtól függ, hogy a fej ott áll-e. Ez a rész is könnyen megvalósítható lenne, ha tudnánk Γ elemeinek kódolásához használt bitek ℓ számát ismernénk. A zöld területen az $f(\epsilon, k_0, k_1) = k_\epsilon$ függvényt számoljuk ki, ami $1 + 2\ell$ bitből számol ki ℓ-et.

Észrevétel 3. $\lceil \omega \rceil \to \lceil \kappa_0(\omega) \rceil$ egy egyszerű hozzárendelés/függvény.

Ez a korábbiaknál egyszerűbb észrevétel. Ismét egy ábrára utalunk.



Feltettük, hogy a START állapot kódja $00\dots0$ (hossza $\lceil\log_2|S|\rceil$, páldánkban 9). Az inputszalagon $\triangleright$ kódja $00\dots0$, a $\triangleleft$ kódja $11\dots1$ (hosszaik $\lceil\log_2|\Sigma|\rceil$, példánkban 5). Az munkaszalagon $\triangleright$ kódja $0\dots00$, a szűzkarakter kódja $0\dots01$, (hossza $\lceil\log_2|\Gamma|\rceil$, példánkban 8).

Először definiálunk egy problémát/nyelvet, majd összefoglaljuk a fenti megállapításainkat egy tételben.

Definíció. Legyen

$$\text{HÁLÓZAT-KIÉRTÉLÉS} = \{\lceil\mathcal{C},\omega\rceil : \mathcal{C}(\omega) = 1\},$$

azaz az a döntési probléma, amely egy $\mathcal{C}$ hálózat és ω bitsorozat esetén eldönti, hogy az ω bitsorzatot adva az $x_1, x_2, \dots$ inputkapuk értékeinek a hálózat az 1 bitet számolja-e ki (azaz kiértékeli $\mathcal{C}_n(\omega)$-t).

1. Tétel. *HÁLÓZAT-KIÉRTÉLÉS $\mathcal{P}$-teljes (az $\mathcal{L}$ redukcióra nézve).*

Bizonyítás. Legyen $L \in_T \mathcal{P}$. Legyen $t(n)$ egy polinom, T időkorlátja. Feltehető, hogy T olyan, hogy ELFOGAD/ELVET állapot elérése után „tartja” állapotát. Így $\omega \in L$? kérdés megválaszolása ($\omega \in \Sigma^n$) ekvivalens annak megállapításával, hogy az ω- történő futás során a $\kappa_{t(n)}$ konfigurációban az állapot ELFOGAD-e. Megállapodhatunk abban, hogy a konfigurációkat 0-1 sorozatokkal kódoljuk, hogy az állapotot kódoló blokkban az ELFOGAD állapot kódja a csupa 1 sorozat legyen.

A fentiek alapján bármi volt is $L \in_T \mathcal{P}$ $\omega \in \Sigma^n$ tetszőleges eleméhez felépíthető egy $\mathcal{T}_{T,\omega}$ hálózat, amely inputkapui ω-t kódolják (3. észrevétel) és bizonyos szintje a Turing-számítás konfiguráció-sorozatának elemeit kódolja (2. észrevétel). $t(n)$ ilyen szint felépítése után az érdekel minket, hogy egy bizonyos blokkban csupa 1-es bit szerepel-e. Ezt ÉS kapuk segítségével könnyen kifejezhetjük.

Ezzel leírtuk a redukáló algoritmust. A redukció elméleti része a korábbiakból adódik. A konstrukció/redukció logtár-beli. ■

2. Egy $\mathcal{NP}$-teljes nyelv

Definíció. Legyen

$$\text{HÁLÓZAT-SAT} = \{\lceil\mathcal{C}\rceil : \text{ van olyan } \omega \text{ bitsorozat, amelyre } \mathcal{C}(\omega) = 1\},$$

azaz az a döntési probléma, amely egy $\mathcal{C}$ hálózat esetén el kell döntenünk, hogy kielégíthető-e.

2. Következmény.

- *(i) Tetszőleges $L \in \mathcal{NP}$ nyelvre $L \prec_{\mathcal{L}}$ HÁLÓZAT-SAT*
- *(ii) HÁLÓZAT-SAT $\mathcal{NP}$-teljes*
- *(iii) $\mathcal{P} = \mathcal{NP} \Leftrightarrow$ HÁLÓZAT-SAT $\in \mathcal{P}$*

Bizonyítás. (i) $L \in_T \mathcal{NP}$, így egy tetszőleges $\omega \in L$ esetén létezik hozzá egy tanú: $\tau = (t_1, t_2, \ldots, t_{p(n)})$, amelyre $T(\omega, \tau)$ ELFOGAD állapotba jut. Azaz az előbb megkonstruált C hálózatra, $C(\lceil \omega \rceil, y_1, y_2, \ldots, y_{q(n)})$ az 1 értéket számolja ki, ha az y változók helyére a τ kódjának bitjeit írjuk. Megfordítva is igaz. Ha $C(\lceil \omega \rceil, y_1, y_2, \ldots, y_{q(n)})$-nek találunk egy kielégítését, akkor egy tanú kódját találjuk. Azaz $C(\lceil \omega \rceil, y_1, y_2, \ldots, y_{q(n)})$ kódjának legyártása (ami az emlélekeztető alapján $\mathcal{L}$-ben megoldható) egy jó redukció.

(ii) Az (i) részből és abból, hogy HÁLÓZAT-SAT$\in \mathcal{NP}$ (tanú egy kielégítő bemenet), következik, hogy HÁLÓZAT-SAT $\mathcal{NP}$-teljes.

(iii) Mivel HÁLÓZAT-SAT $\in \mathcal{NP}$, így ha $\mathcal{P} = \mathcal{NP}$, akkor $\mathcal{P}$-beli is. Viszafelé, ha HÁLÓZAT-SAT$\in \mathcal{P}$, akkor tetszőleges $L \in \mathcal{NP}$ nyelvet redukáljunk HÁLÓZAT-SAT-ra, a redukált problémát $\mathcal{P}$-ben el tudjuk dönteni. A két lépés együtt is polinomiális és az L nyelv eldöntési problémáját oldja meg. Ebből $L \in \mathcal{P}$ következik, így $\mathcal{NP} \subseteq \mathcal{P}$, tehát $\mathcal{P} = \mathcal{NP}$ adódik. ■

A továbbiakban a legklasszikusabb $\mathcal{NP}$-teljes nyelvet imsertetjük.

3. Cook—Levin-tétel

Definíció. Legyen $V = \{x_1, x_2, \ldots, x_n\}$ egy változó halmaz. Legyen $L = V \dot{\cup} \overline{V}$ a literálok halmaza ($\overline{V}$ a negált változók halmaza, azaz $\{\overline{x}_1, \overline{x}_2, \ldots, \overline{x}_n\}$). L egy részhalmazát klóznak nevezzük. Esetünkben a klózra úgy gondolunk, hogy a hozzá tartozó literálokat $\vee$ logikai művelettel, azaz diszjunkcióval kapcsoljuk össze. Egy φ konjunktív normálformában lévő formula (CNF formula) klózok egy halmaza. Erre a klózhalmazra úgy gondolunk, hogy a klózok $\wedge$ logikai művelettel, azaz konjukcióval kapcsoljuk össze. Egy φ CNF formula kielégíthető, ha adható V egy kiértékelése (ami természetes módon kiterjeszthető L egy kiértékelésévé), amelyre minden klózban lesz igazzá kiértékelt literál.

$$\mathrm{SAT} = \{\lceil \varphi \rceil : \varphi \text{ kielégíthető CNF}\}$$

Azaz SAT az a probléma, ahol adott egy CNF formula és el kel döntenünk, hogy kielégíthető-e.

3. Tétel (Cook—Levin-tétel). *SAT (CNF formula kielégíthetősége) $\mathcal{NP}$-teljes.*

Bizonyítás. Korábban már szerepelt, hogy SAT $\in \mathcal{NP}$, így elég adnunk egy visszavezetést HÁLÓZAT-SAT-ról SAT-ra, hiszen ekkor az előző következmény (i) pontja és a polinomiálsi redukcio tranzitívitása alapján tetszőleges $L \in \mathcal{NP}$ nyelvet vissza tudunk vezetni SAT-ra. Ezt is két lépésben tesszük, a HÁLÓZAT-SAT-ot visszavezetjük BOOLE-EGYENLETRENDSZER-SAT-ra, majd azt SAT-ra.

Definíció. $\varphi_i(x_1, x_2, \ldots, x_n) = \psi_i(x_1, x_2, \ldots, x_n)$, $i = 1, 2, \ldots, \ell$ egyenletrendszert, Boole-egyenletrendszernek nevezzük, ha φ_i és ψ_i Boole-formulák. Az $\{x_i\}_{i=1}^n$ változók egy 0-1/igaz-hamis értékadás az egyeneletrendszer megoldása, ha minden $i = 1, 2, \ldots, \ell$ esetén φ_i és ψ_i értéke ugyanaz.

BOOLE-EGYENLETRENDSZER-SAT az a nyelv, ami a megoldható/kielégíthető Boole-egyenletrendszerek kódját tartalmazza.

Legyen H egy hálózat. Minden csúcsával azonosítunk egy változót. Ez input-csúcsok esetén a csúcs címkéje. A többi csúcsra (kapukra) mind különváltozókat feleltetünk meg. Minden kapuhoz tartozik egy egyenlet

- $x_g = \neg x_h$, ha a g kapu negáció kapu és a h kapuból kapja inputját ($\overrightarrow{hg}$ él a hálózatban).
- $x_g = x_h \wedge x_{h'}$, ha a g kapu hálózatbeli címkéje konjunkció és inputjait h és h' kapukból kapja.
- $x_g = x_h \vee x_{h'}$, ha a g kapu hálózatbeli címkéje diszjunkció és inputjait h és h' kapukból kapja.
- $x_g = 1$, ha a g kapu a hálózat output kapuja.

Ezzel megkaptuk egy Boole-egyenletrendszert a hálózatból. Ha a hálózat 1-et számol ki egy értékadáson (az értékadás kielégíthetőséget bizonyít), akkor a kapuk által kiszámolt bitekkel együtt egy megoldását kapjuk az egyeneletrendszernek. Fordítva is igaz, ha az egyenletrendszer megoldásából kiemeljük az eredeti input változók értékadásait, akkor ezen a hálózat 1-et számol ki (sőt minden kapu a hozzárendelt változó megoldásbeli értékét számolja ki). Azaz a hálózatból legyártott egyenletrendszer megoldhatósága ekvivalens azzal, hogy a hálózat kielégíthető.

Az '=' jeleket '$\leftrightarrow$' logikai jelekre cserélve az egyes egyenleteknek megfelelő formulákat kapunk. Egy értékadás akkor és csak akkor teljesíti az egyenletet, ha igazzá teszi a hozzárendelt formulát. A kapott logikai kifejezések mindegyike legfeljebb három változót tartalmaznak. Könnyű őket CNF formára hozni. Ha az összes egyenletnek megfeleltetett CNF formulát 'és' logikai jellel összekapcsoljuk, szintén CNF formát kapunk. Így az egyenletrendszerhez hozzárendeltünk egy φ CNF formulát. A hozzárendelés $\mathcal{P}$-ben kiszámolható. Az egyenletrendszer megoldhatósága ekvivalens a formula kielégíthetőségével.

Vagyis „programunk" második redukcióját is megadtuk, a tételt bebizonyítottuk. ∎

Eddig több bonyolultsági osztályra láttunk teljes problémákat ($\mathcal{NL}, \mathcal{P}, \mathcal{NP}$). Az $\mathcal{NP}$ osztály esete azonban különbözik. Igen sok, első látásra teljesen különböző, fontosnak tartott problémákról derült ki, hogy $\mathcal{NP}$-teljesek. Az osztály azért különösen fontos, mert a sejtett $\mathcal{P} \neq \mathcal{NP}$ összefüggés esetén ezek a problémákra nem létezik polinomiális idejű algoritmus. Azaz azt mondhatjuk rájuk, hogy elméletileg nem kezelhetőek hatékonyan.

4. További $\mathcal{NP}$-teljes problémák

Legyen $(= 3)$-SAT azon kielégíthető CNF formulák kódja, melyekben minden klóz pontosan 3 literált tartalmaz. Legyen (≤ 3)-SAT azon kielégíthető CNF formulák kódja, melyekben minden klóz legfeljebb 3 literált tartalmaz.

4. Lemma. *$(= 3)$-SAT $\preceq_{\mathcal{P}}$ (≤ 3)-SAT, továbbá (≤ 3)-SAT$\preceq_{\mathcal{P}}(= 3)$-SAT.*

Bizonyítás. Az első redukció nyilvánvaló, hisz a $(=3)$-SAT probléma egy speciális esete (≤ 3)-SAT-nek.

Fordítva legyen φ a (≤ 3)-SAT egy inputja. Egy a kielégíthetőség szempontjából ekvivalens formulává alakítjuk, úgy, hogy a klózok mérete három legyen. Egy φ input 3 méretű klózait tartsuk meg, míg a kicsi klózaikkal az alábbi operációt végezzük el (párhuzamosan): Egy $C : \langle \ell_1, \ell_2 \rangle$ „kicsi" klózt helyettesítsünk a $\langle \ell_1, \ell_2, u \rangle$, $\langle \ell_1, \ell_2, \overline{u} \rangle$ klózpárral. Természetesen formulánk kielégítése esetén C is igazzá válik, ami a C-t helyettesítő klózokat is igazzá teszi. Fordítva: A C-t helyettesítő klózpárt csak úgy tudjuk kielégíteni, ha az eredeti C-t is kielégítjük.

A fenti példában a kicsi klóz két literált tartalmazott. Ennél is kisebb klózokra is alkalmazható ötletünk. Eredménye: ekvivalens formula eggyel nagyobb klózokkal. Ötletünket iteráltan kell alkalmazni, amíg minden klóz három méretűvé válik. ■

Ha két nyelv „oda-vissza" redukálható, akkor ekvivalensnek (polinomiális redukcióra) mondjuk. Ekkor a két probléma egyenértékű. A fenti tétel egy ekvivalenciát állít. Ez alapján a két probléma egyenértékű. Ha a 3-SAT nyelvet említjük, akkor alatta a fenti két nyelv bármelyikét érthetjük. Természetesen, ha 3-SAT-ot redukáljuk, akor az input CNF-ről feltesszük, hogy minden klóza három literált tarrtalmaz. Ha 3-SAT-ra redukálunk, akkor nem zavar, ha a redukciós algoritmus háromnál kisebb klózokat is „gyárt".

5. Tétel. *3-SAT $\mathcal{NP}$-teljes.*

Bizonyítás. 3-SAT $\in \mathcal{NP}$ triviális (a SAT speciális esete).

3-SAT $\mathcal{NP}$-nehéz: SAT visszavezetése 3-SAT-ra (emlékeztetőül: egy SAT $\to$ 3-SAT: $\mathcal{C} \mapsto \mathcal{C}'$ polinomidőben kiszámítható függvény kell, amelyre teljesül, hogy $\mathcal{C} \in$ SAT $\Leftrightarrow \mathcal{C}' \in$ 3-SAT). A hozzárendelés a következő lesz: a $C = \langle \ell_1, \ldots, \ell_k \rangle$ klózra vezessünk be $u_1, \ldots, u_{k-1}$ új változókat és a következő klózokat vegyük fel $\mathcal{C}'$-be:

$$\langle \ell_1, \overline{u}_1 \rangle, \langle u_1, \ell_2, \overline{u}_2 \rangle, \ldots, \langle u_{i-1}, \ell_i, \overline{u}_i \rangle, \ldots, \langle u_{k-2}, \ell_{k-1}, \overline{u}_{k-1} \rangle, \langle u_{k-1}, \ell_k \rangle$$

Ezt minden $\mathcal{C}$-beli klózra végezzük el. Amit kapunk, az egy 3-CNF. A következőket kell belátni:

(i) $\mathcal{C}'$ meghatározható ($\mathcal{C}$ kódjának hosszában) polinomiális időben. Ez nyilvánvaló.

(ii) $\mathcal{C}$ pontosan akkor kielégíthető, ha $\mathcal{C}'$ az:

Ha $\mathcal{C}$ kielégíthető, akkor vegyük egy kielégíthető kiértékelését. $C = \langle \ell_1, \ldots, \ell_k \rangle$ klóz esetén legyen ℓ_i a klózban az első igaz literál. Ekkor ℓ_i értékeit megtartva, $u_1 = \ldots = u_{i-1} = h$ és $u_i = \ldots = u_{k-1} = i$ kiértékelés jó lesz.

Másik irány: $\mathcal{C}'$-nek nincs olyan kielégítő kiértékelése, amelyben valamely $\mathcal{C}$-beli $C = \langle \ell_1, \ldots, \ell_k \rangle$ klóz esetén $\ell_1 = \ldots = \ell_k = h$ lenne, mivel az

$$\langle \overline{u}_1 \rangle, \langle u_1, \overline{u}_2 \rangle, \ldots, \langle u_{i-1}, \overline{u}_i \rangle, \ldots, \langle u_{k-2}, \overline{u}_{k-1} \rangle \langle u_{k-1} \rangle$$

klózokat tartalmazó formula kielégíthetetlen.

■

Legyen k-SAT a következő probléma: adott egy konjunktív normálforma, amelyben minden klóz legfeljebb k literált tartalmaz (k-CNF). Kielégíthető-e?

Megjegyzés. Egyszerűen látható, hogy 2-SAT $\in \mathcal{P}$. Sőt 2-SAT $\in co\mathcal{NL}$.

Megjegyzés. Nyilván igaz a következő redukciólánc:

$$2\text{-}SAT \preceq_{\mathcal{P}} 3\text{-}SAT \preceq_{\mathcal{P}} 4\text{-}SAT \preceq_{\mathcal{P}} \ldots \preceq_{\mathcal{P}} k\text{-}SAT \preceq_{\mathcal{P}} \ldots \preceq_{\mathcal{P}} SAT.$$

Definíció. *V egy kiértékelése egy klózt homogénné tesz, ha a klóz minden literálja ugyanazt a logikai értéket kapja. Azaz a klóz akkor nem lesz honogén, ha igazzá értékelődik ki (lesz benne igaz értékű literál), de nem minden literál igaz értékű (lesz benne hamis literál is)*

Legyen

$$\text{NEM-MIND-IGAZ-SAT} = \{\lceil\varphi\rceil : \varphi \text{ CNF, amely kielégíthető úgy, hogy ne legyen olyan klóz, amelyben minden literál igaz}\}$$

6. Tétel. *NEM-MIND-IGAZ-SAT $\mathcal{NP}$-teljes.*

NEM-MIND-IGAZ-SAT $\in \mathcal{NP}$ triviális. Egy másik $\mathcal{NP}$-teljes problemát és a szükséges redukciót később ismertetjük.

Legyen NEM-MIND-IGAZ-3-SAT azon CNF-ek halmaza, amelyben minden klóz legfeljebb három literált tartalmaz és van olyan kiértékelése a változóknak, hogy φ összes klóza nem homogén.

7. Következmény. *NEM-MIND-IGAZ-3-SAT $\mathcal{NP}$-teljes.*

Bizonyítás. A SAT$\preceq$3-SAT reduckiót ismételjük meg: Ha egy NEM-MIND-IGAZ-SAT φ inputra alkalmazzuk az előző visszavezetést, akkor egy olyan $R(\varphi)$ CNF formulát kapunk, amelyben minden klóz legfeljebb három literált tartalmaz.

Tegyük fel, hogy $\varphi \in$NEM-MIND-IGAZ-SAT, azaz a változók alkalmas kiértékelésénél minden klózban lesz igaz és hamis literál is. Tegyük fel, hogy C egy klóz és egy benne lévő ℓ_i literál igaz, míg ℓ_j literál hamis. Feltehetjük, hogy $i < j$.

A régi változók értékeit tartsuk meg, az alábbiakban leírjuk, hogy az új változók hogyan kapják értékeiket. Nézzük meg C redukcióját:

$$\langle \ell_1, \overline{u}_1\rangle, \langle u_1, \ell_2, \overline{u}_2\rangle, \ldots, \langle u_{i-1}, \ell_i, \overline{u}_i\rangle, \ldots, \langle u_{k-2}, \ell_{k-1}, \overline{u}_{k-1}\rangle, \langle u_{k-1}, \ell_k\rangle.$$

ℓ_i után jön egy új változó negáltja. Az új változónak igaz értéket adva az ℓ_i literál „kis” klózában lesz hamis és igaz érték is. Jobbra haladva a további új változóknak igaz értéket adva eljutunk ℓ_j „kis” klózához. Közben minden klózba kerül igaz és hamis érték is. A szélső klózokban ugyanez elérhető, ha szélről középfelé haladunk és így osztjuk ki az új változók értékeit a hiányzó logikai értéket pótolva. Összegezve kaptuk, hogy $R(\varphi) \in$NEM-MIND-IGAZ-3-SAT.

Fordítva tegyük fel, hogy $R(\varphi) \in$NEM-MIND-IGAZ-3-SAT. Láttuk, hogy egy kiértékelés, amely minden $R(\varphi)$ klózt kielégít (ilyen egy nem-mind-igaz kiértékelés) akkor elképzelhetetlen, hogy az eredeti klózok ne tartalmazzanak igaz értéket. Azt kell csak kizárnunk, hogy $R(\varphi)$ egy nem-mind-igaz kiértékelése az eredeti változókra megszorítva egy eredeti klóz mindegyik literálját igazzá tegye.

Ez azt jelentené, hogy a

$$\langle \overline{u}_1\rangle, \langle u_1, \overline{u}_2\rangle, \ldots, \langle u_{i-1}, \overline{u}_i\rangle, \ldots, \langle u_{k-2}, \overline{u}_{k-1}\rangle\langle u_{k-1}\rangle$$

klózok mindegyikébe lenniek kell hamis értéknek. Ez (mint korábban) lehetetlen. ∎

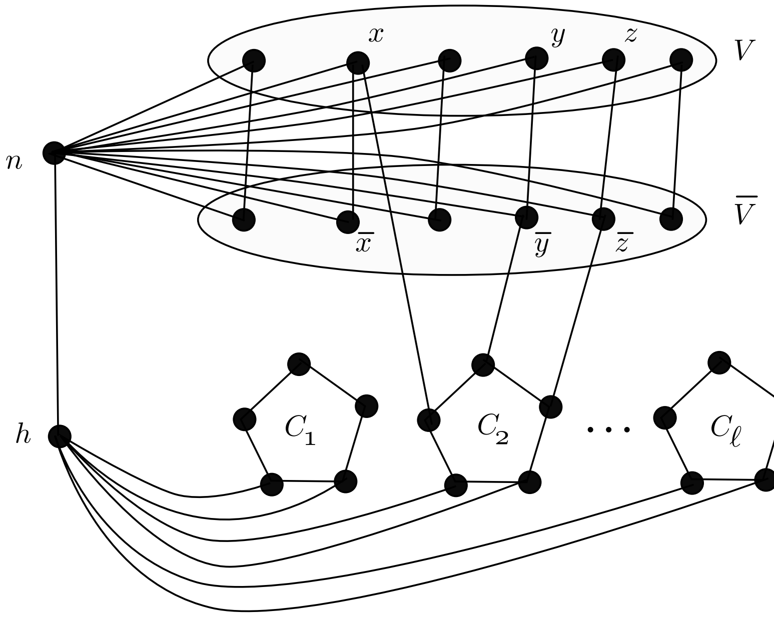
4.1. Gráfelméleti problémák

Definíció. k-SZÍNEZHETŐSÉG a következő probléma: adott egy gráf, kiszínezhető-e k színnel?

8. Tétel. *3-SZÍNEZHETŐSÉG $\mathcal{NP}$-teljes.*

Bizonyítás. 3-SZÍNEZHETŐSÉG $\in \mathcal{NP}$: tanú egy színezés, polinom időben ellenőrizhető, jó színezés-e.

3-SZÍNEZHETŐSÉG $\mathcal{NP}$-nehéz: 3-SAT-ot vezetjük rá vissza. $\mathcal{C}$ 3-CNF-hez hozzárendeljük $G_\mathcal{C}$ gráfot, amely csúcsai n, h, $\mathcal{C}$ változói és azok negáltjai, és minden $C \in \mathcal{C}$ klózra C_1, C_2, C_3, C_4, C_5. $G_\mathcal{C}$ élei a következők lesznek: uv, minden x_i változóra $x_i\overline{x}_i$, ux_i és $u\overline{x}_i$, illetve minden $C = \langle z_1, z_2, z_3\rangle$ klózra C_1C_2, C_2C_3, C_3C_4, C_4C_5, C_5C_1, C_1z_1, C_2z_2, C_3z_3, C_4v, C_5v.



1. ábra.

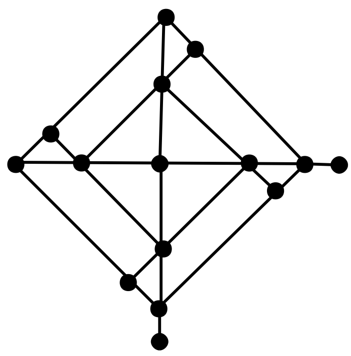
Könnyű ellenőrizni, hogy $G_\mathcal{C}$ polinom időben meghatározható, és pontosan akkor 3-színezhető, ha $\mathcal{C}$ kielégíthető (felhasználva azt az észrevételt, hogy z_1, z_2, z_3, v 3-színezése pontosan akkor terjeszthető ki $C = \langle z_1, z_2, z_3\rangle$ klózhoz tartozó ötszögre jó színezésként, ha a 4 csúcs színe nem azonos). ■

Megjegyzés. Könnyen ellenőrizhető, hogy 2-SZÍNEZHETŐSÉG $\in co\mathcal{NL}$.

9. Tétel. *SÍKGRÁF 3-SZÍNEZHETŐSÉGE $\mathcal{NP}$-teljes.*

Vázlat. $\mathcal{NP}$-beliség triviális.

SÍKGRÁF 3-SZÍNEZHETŐSÉGE $\mathcal{NP}$-nehéz: 3-SZÍNEZHETŐSÉG-et vezetjük rá vissza. Lerajzolás után a kereszteződő éleket kell helyettesíteni az alábbi kis gráffal:



Azt kell meggondolni, hogy megtehetők a helyettesítések úgy, hogy nem hoznak be új metszéseket, és a keletkezett gráf pontosan akkor 3-színezhető, ha az eredeti is. ∎

Megjegyzés. A 2-színezhetőség $co\mathcal{NL}$-beli, síkgráfokra a 4-színezhetőség triviális a négyszíntétel (Appel, Haken 1977) alapján.

Definíció. SZÍNEZÉSI PROBLÉMA: adott egy G gráf és egy k természetes szám. Van-e G-nek jó k-színezése?

10. Tétel. *SZÍNEZÉSI PROBLÉMA $\mathcal{NP}$-teljes.*

Bizonyítás. SZÍNEZÉSI PROBLÉMA $\in \mathcal{NP}$: tanú egy színezés, polinom időben ellenőrizhető, jó színezés-e.

A SZÍNEZÉSI PROBLÉMA $\mathcal{NP}$-nehéz, mivel a 3-SZÍNEZHETŐSÉG általánosítása. ∎

⋆ ⋆ ⋆

Definíció. FÜGGETLEN CSÚCSHALMAZ: adott egy G gráf és egy k természetes szám. Van-e G-ben k-elemű független csúcshalmaz?

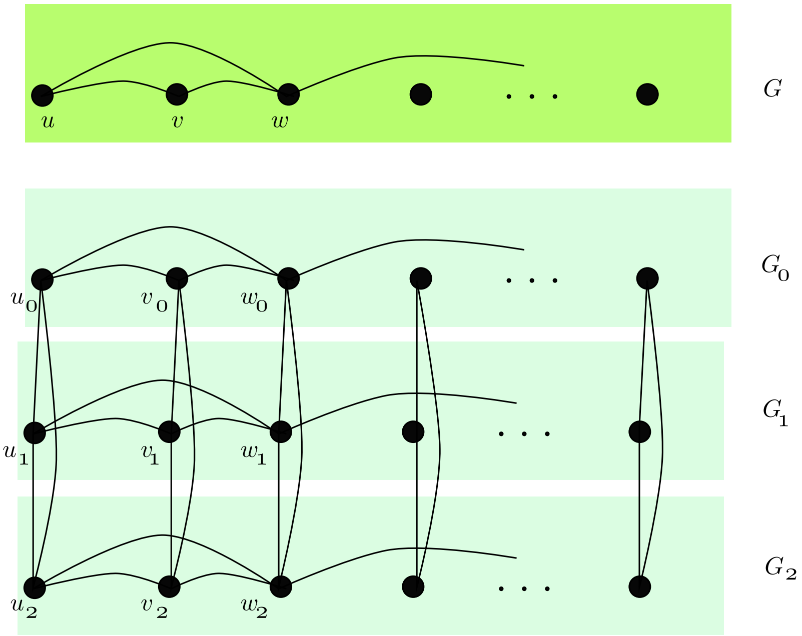
11. Tétel. *FÜGGETLEN CSÚCSHALMAZ $\mathcal{NP}$-teljes.*

Bizonyítás. FÜGGETLEN CSÚCSHALMAZ $\in \mathcal{NP}$: tanú egy független csúcshalmaz.

FÜGGETLEN CSÚCSHALMAZ $\mathcal{NP}$-nehézségére két bizonyítást adunk.

I. SAT visszavezetése: $\mathcal{C} = (C_1 = \langle z_{1,1}, \ldots, z_{1,r_1}\rangle, \ldots, C_k = \langle z_{k,1}, \ldots, z_{k,r_k}\rangle) \mapsto (G_\mathcal{C}, k)$ ((i,j) jelentése: i-edik klóz j-edik literálja), $V(G_\mathcal{C}) = \{(i,j) : i \le k, j \le r_i\}$, $E(G_\mathcal{C}) = \{(i,j),(i',j') : i = i'$ vagy $z_{i,j} = \overline{z}_{i',j'}\}$. Könnyű meggondolni, hogy $G_\mathcal{C}$ polinom időben meghatározható és pontosan akkor van benne k elemű független halmaz, ha $\mathcal{C}$ kielégíthető, mivel kiértékelés akkor kielégítő, ha minden klózból ki tudunk választani egy igaz literált (az élek garantálják, hogy változó és negáltja egyszerre ne szerepeljenek, illetve minden klózból legfeljebb egy literált válasszunk).

II. SZÍNEZÉSI PROBLÉMA visszavezetése: $G \mapsto (G', |V(G)|)$, ahol $V(G') = \{(v,i) : v \in V(G), i \in [3]\}$ (itt (v,i) jelentése az, hogy v az i színt kapja), $E(G') = \{(v,i)(v',i') : v = v', i \ne i'$ vagy $vv' \in E(G), i = i'\}$ (vagyis: élek a tiltások, tiltott, hogy egy csúcs több színt kap vagy összekötött csúcsok azonos színt kapnak).



Könnyű meggondolni, hogy G' és $|V(G)|$ is polinom időben meghatározható, és pontosan akkor van G'-ben $|V(G)|$ elemű független halmaz, ha G gráf 3-színezhető. ■

Megjegyzés. Szemben a SZÍNEZÉSI PROBLÉMÁval, ha k nem az input része, hanem konstans, akkor az így kapott k-FÜGGETLEN HALMAZ probléma már polinom időben megoldható (egy n csúcsú gráfnak n-ben polinomiális sok k-elemű részhalmaza van, ha k fix).

⋆

Definíció. KLIKK probléma: adott egy G gráf és egy k természetes szám. Van-e G-ben k méretű klikk?

Definíció. LEFOGÓ-PONTHALMAZ: adott egy G gráf és egy k természetes szám. Van-e G-ben k-elemű lefogó ponthalmaz?

12. Következmény. *KLIKK és LEFOGÓ-PONTHALMAZ $\mathcal{NP}$-teljesek.*

Bizonyítás. Ekvivalensek a FÜGGETLEN-CSÚCSHALMAZ problémával. ■

⋆

Definíció. HAMILTON, azon gráfok kódjait tartalmazó nyelv, amelyek Hamilton-körrel rendelkeznek.

A HAMILTON nyelv döntési feladat azt jelenti, hogy adott gráfról el kell döntenünk, hogy rendelkezik-e Hamilton körrel.

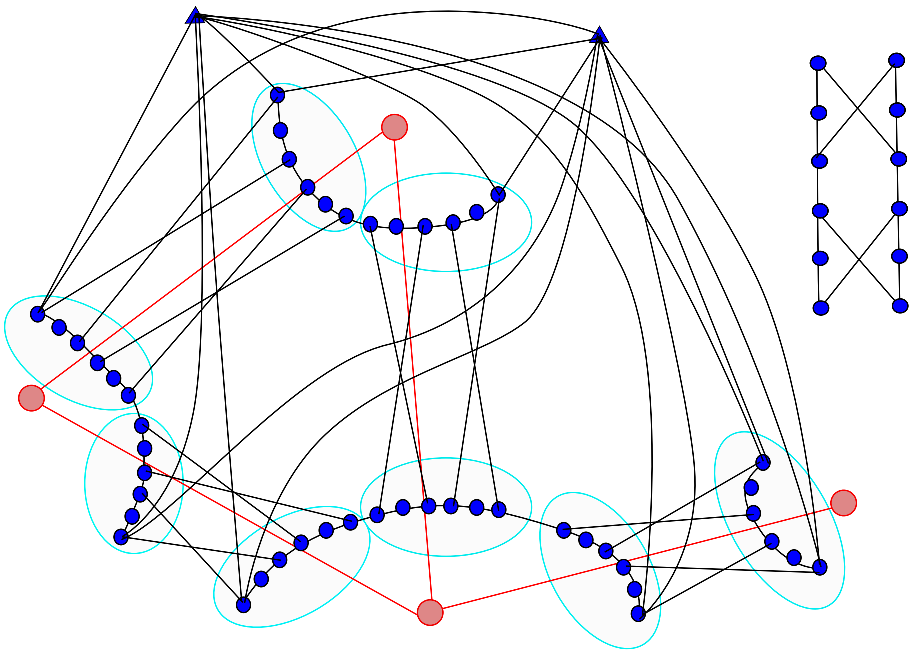
13. Tétel. *HAMILTON $\mathcal{NP}$-teljes.*

Bizonyítás. HAMILTON$\in \mathcal{NP}$ nyilvánvaló.

Belátjuk, hogy LEFOGÓ-PONTHALMAZ visszavezethető HAMILTON-ra. Azaz, ha adott egy G gráf és $k \in \mathbb{N}$, akkor hatékonyan definiálhatunk egy R gráfot, amely pontosan akkor rendelkezik Hamilton-körrel, ha G lefogható k csúccsal.

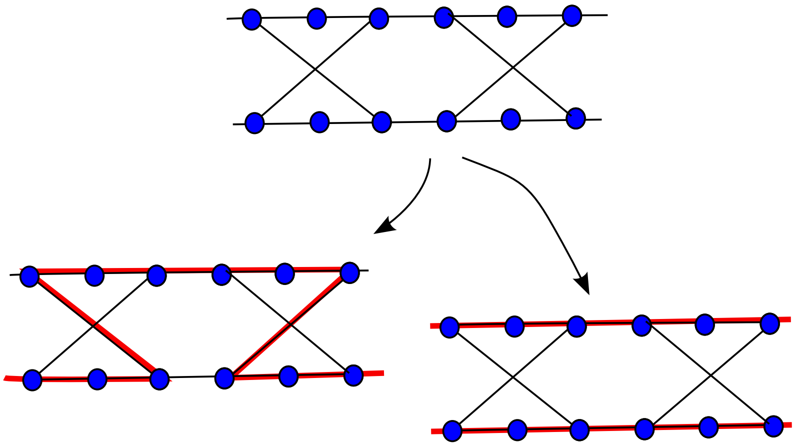
A redukciót egy példával/ábrával demonstráljuk. Az általános, formális leírás könnyen kiolvasható az ábrából, ezt az érdeklődő hallgatóra bízzuk.

Az ábrán ott van az eredeti G gráf pirossal, $k = 2$ a példánkban. Az R gráf kékkel van rajzolva.



2. ábra. Minden G-beli csúcsnak megfelel egy-egy út, amelyekben hatos blokkokban (világos kék ellipszisekkel bekerítettek) szerepelnek csúcsok (kék, kerek csúcsok). Egy csúcsnak megfelelő úton annyi blokk van, ahány él illeszkedik a csúcsra. A blokkok és a csúcs-él illeszkedések azonosítottak. Így minden élnek megfelel két hatos csúcs-blokk, amely a jobb oldalon szemléltetett módon összekötöttünk. $k = 2$ érték alapján $k = 2$ új csúcsot is felvettünk (kék, háromszög alakú csúcsok). Ezeket pontosan az összes csúcshoz rendelt utak két-két végpontjával kötjük össze.

Tegyük fel, hogy ebben a gráfban van Hamilton-kör. Ekkor ezt a k háromszög alakú csúcs k darab ívre szakítja szét. Az ívek egy-egy út végpontjában kezdődnek, végződnek. Könnyű ellenőrizni, hogy egy élnek megfelelő két hatos blokkot ezek az ívek csak kétféleképpen szelhetik át:



3. ábra. (Bal) Az átszelés egy íven történik. Egy csúcs útján haladunk, de egy kitérővel az él másik blokkját is bejárjuk. (Jobb) Az átszelés két külön részben történik és mindegyik egy-egy csúcs útján történik.

Mindkét esetben a Hamilton kör egy-egy íve egy-egy csúcs útját járja be (közben lehetséges kitérőkkel). Így a k háromszög alakú csúcs kiválaszt k csúcsot. Az erre a csúcsra illeszkedő élek mindegyik egy-egy lehetséges kitérőt engedélyez. Ezeket a kitérőket úgy kell megszerveznünk, hogy az összes többi csúcs útja, illetve ezek blokkjai is be legyenek kjárva. Könnyen látható, hogy ez csak akkor szervezhető meg, ha a kijelölt k csúcs lefogó csúcshalmazt alkot. Tehát Hamilton-kör létezése bizontyítja a k csúccsal lefoghatóságot.

Gondolatmenetünk könnyen megfordítható. Ez bizonyítja a redukció elméleti részét. A technikai apróságok (implementáció polinomialis időben) részletezését elhagyjuk. ∎

⋆

Egy gráf vágása csúcsainak két diszjunkt részre bontása. A vágás élhalmaza azon éleket tartalmazza, amelyek egyik végpontja az egyik, másik végpontja a másik részbe esik.

Definíció. MAX-CUT probléma: adott egy G gráf és egy k természetes szám. Van-e G-ben olyan vágás, amely legalább k élű?

14. Tétel. *MAX-CUT $\mathcal{NP}$-teljes.*

Bizonyítás. MAX-CUT $\in \mathcal{NP}$: tanú egy színezés, polinom időben kiszámolható az élszáma, amit k-val összehasonlíthatunk.

MAX-CUT $\mathcal{NP}$-nehéz: NEM-MIND-IGAZ-3-SAT-ot vezetjük rá vissza. $\mathcal{C}$ 3-CNF-hez hozzárendeljük $G_{\mathcal{C}}$ gráfot, amely csúcsai $\mathcal{C}$ változói és azok negáltjai (a literálok). Minden $C \in \mathcal{C}$ klózra a benne szereplő három literált páronként összekötjük. (Ezekre az élekre mint klóz-élekre hivatkozunk.) Minden klózchoz három klóz-él tartozik. Ha két literál több klózban is együtt szerepel, akkor többszörös élek lesznek a redukció által megkonstruált gráfban. Minden x változóra x és $\overline{x}$ közé is behúzunk egy élt. (Ezekre az élekre mint változó-élekre hivatkozunk.) Ezzel leírtuk a $G_{\mathcal{C}}$ gráf összes élét.

Könnyű ellenőrizni, hogy $G_\mathcal{C}$ polinom időben meghatározható, és pontosan akkor van benne $|V| + 2|\mathcal{C}|$ élű vágás, ha van olyan kiértékelése a változóinknak, hogy $\mathcal{C}$ minden klóza nem-homogén. Valóban, $G_\mathcal{C}$ minden vágása lefeljebb (az összes) $|V|$ literál-élet és minden klóz három klózéléből legfeljebb 2-t tartalmaz. Azaz $|V|+2|\mathcal{C}|$ egy felső becslés $G_\mathcal{C}$ tetszőleges vágásának élszámára. Ha egy (I, H) vágás eléri ezt a felső becslést, akkor minden literálélet tartalmaz, azaz minden x változóra x és $\overline{x}$ közül egyik I-be, másik H-ba esik. Azaz a vágás definiál egy kiértékelés'et a változóinknak. Továbbá minden klóz három klózéléből kettőt tartalmaz a vágás, azaz a leírt kiértékelés nem-mind-igaz módon ielégíti (a tetszőlegesen választott) klózt $\mathcal{C}$-ből. ■

Megjegyzés. A MIN-CUT probléma annak tesztelését kéri, hogy van-e olyan vágás, amely élszáma k-nál nem nagyobb. Ez a probléma polinom időben megoldható. A folyamok elmélete alapján a legkisebb vágás élhalmaza meghatározható.

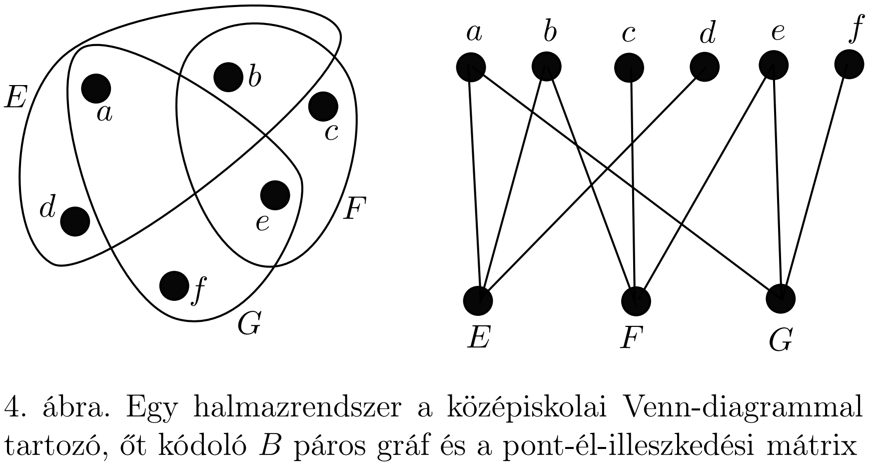
Szintén megjegyezzük, hogy a redukciónk olyan gráfot készített, amely optimális vágása két egyenlő osztályra történt/felezés volt. Így a MAX-FELEZÉS problémáról is adódott, hogy $\mathcal{NP}$-teljes. Az egyszerű komplementálás mutatja, hogy a MAX-FELEZÉS és MIN-FELEZÉS problémák (polinomiális időre nézve) ekvivalensek. Speciálisan a MIN-FELEZÉS problémáról is kapjuk, hogy $\mathcal{NP}$-teljes.

5. Halmazrendszerekre vonatkozó problémák

Definíció. $\mathcal{H}$ halmazrendszer a V halmaz felett, ha $\mathcal{H} \subseteq \mathcal{P}(V)$. $\mathcal{H}$ elemei a halmazrendszer élei. k-uniform halmazrendszer olyan halmazrendszer, amely összes éle k elemű. Tehát az egyszerű gráfok pontosan a 2-uninform halmazrendszerek.

Észrevétel. $V, \mathcal{H}$ halmazrendszer könnyen leírható B páros gráffal. A két színosztály V (felső pontok) és $\mathcal{H}$ (alsó pontok). Az alaphalmaz egy v eleme/csúcs akkor és csak akkor van összekötve $\mathcal{H}$ egy E elemével/éllel, ha $v \in E$.

A halmazrendszert lehet kódolni pont-él illeszkedési mátrixszal. Ez egy $n \times m$ méretű 0-1 mátrix, ahol $n = |V|$ és $m = |\mathcal{H}|$



$$\begin{array}{c} \\ a \\ b \\ c \\ d \\ e \\ f \end{array} \begin{array}{c} \begin{array}{ccc} E & F & G \end{array} \\ \begin{pmatrix} 1 & 0 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{pmatrix} \end{array}$$

4. ábra. Egy halmazrendszer a középiskolai Venn-diagrammal lerajzolva, a hozzá tartozó, őt kódoló B páros gráf és a pont-él-illeszkedési mátrix

⋆

Definíció. A gráfelmélet független csúcshalmazának fogalmát kétféleképpen terjeszthetjük ki halmazrendszerekre:

- I független, ha minden $E \in \mathcal{H}$ élre $E \not\subseteq I$.
- I független*: ha minden $E \in \mathcal{H}$ élre $|E \cap I| \leq 1$.

Definíció. FGTLEN-CSÚCSOK-HALMAZRENDSZERBEN=

$$\{\lceil V, \mathcal{H}, k\rceil : \text{van olyan } I \text{ független csúcshalmaz, hogy } |I| = k\}.$$

FGTLEN*-CSÚCSOK-HALMAZRENDSZERBEN=

$$\{\lceil V, \mathcal{H}, k\rceil : \text{van olyan } I \text{ független}^* \text{ csúcshalmaz, hogy } |I| = k\}.$$

A páros gráf kódoláson alapulva könnyű leírni a független* halmazokat. Ezeknek B-ben felső pontok egy olyan I halmaza tartozik, amelyekre nem illeszkedik V-alak, azaz olyan $a \in A, f, f' \in I \subset F$ ponthármas, ahol a összekötött f és f'-vel.

Definíció. Legyen B egy halmazrendszert leíró páros gráf. Az alsó/felső szerepek felcserélésével a B^* páros gráfot kapunk. Ez a B^* „elolvasható halmazrendszerként” visszaalakítva halmazrendszerré a $V^* = \mathcal{H}$, $\mathcal{H}^* = V$ duális halmazrendszert kapunk.

A redukciók sorozatát egy trivialitással kezdjük.

15. Tétel. *(i) FGTLEN-CSÚCSOK $\preceq$ FGTLEN-CSÚCSOK-HRSZBEN*
(ii) FGTLEN-CSÚCSOK $\preceq$ FGTLEN-CSÚCSOK-HRSZBEN*

Valóban, a gráfelméleti problémá gráfja a halmazrendszerek egy speciális esete. A gráfelméleti függetlenség mindkét halmazrendszeres függetlenség fogalom speciális esete.

Definíció. FGTLEN-ÉLEK-HRSZBEN =

$$\{\lceil V, \mathcal{H}, k\rceil : \mathcal{H}\text{-ban } J\ k \text{ db olyan él, amelyek páronként diszjuntak}\}.$$

A következő tétel már kevésbé nyilvánvaló.

16. Tétel. *FGTLEN*-CSÚCSOK-HRSZBEN $\preceq$ FGTLEN-ÉLEK-HRSZBEN*

Bizonyítás. $V, \mathcal{H}, k$-ból képezzük a duális halmazrendszert, a k értékét pedig tartsuk meg: $V^*, \mathcal{H}^*, k$.

Azt kell eldöntenünk, hogy az erdeti halmazrendszert leíró B páros gráfban van-e k felső pont úgy, hogy ne támaszkodjon rá $\vee$ alak. $V^*, \mathcal{H}^*$ páros gráfja éppen a fejetetejére állított B. Azaz az eredeti döntési feladat ekvivalens azzal, hogy fejrefordított B gráfban van-e k alsó csúcs (k darab él), hogy ne támaszkodjon rá $\wedge$, azaz páronként diszjunktak legyenek. Azaz FGTLEN-ÉLEK-HRSZBEN problémát kell megoldani $V^*, \mathcal{H}^*, k$ esetén.

Azaz a kiinduló transzformáció a tételt igazoló redukció. ■

Megjegyzés. FGTLEN-ÉLEK-GRÁFOKBAN probléma, másképpen PÁROSÍTÁS $= \{\lceil G, k\rceil : \nu(G) \geq k\}$. Az Edmonds-algoritmus alapján ez egy $\mathcal{P}$-beli probléma. Azaz a gráfokra vonatkozó eset könnyen kezelhető.

Definíció. PARKETTÁZÁS $=$

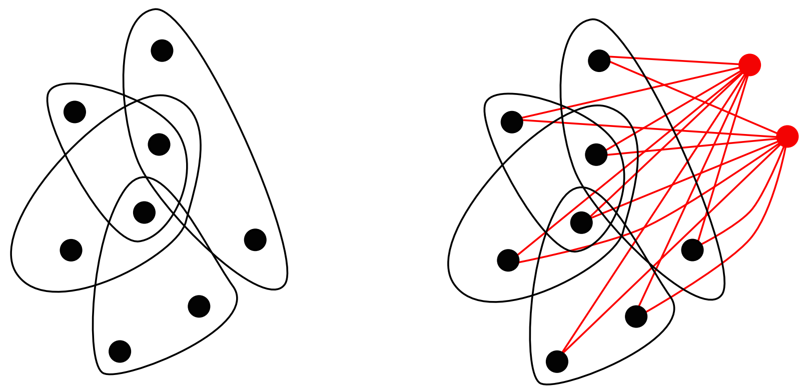
$$\{\lceil V,\mathcal{H}\rceil : \text{ léteznek } E_1,\ldots,E_k \text{ páronként diszjunkt élek, hogy } \dot\cup E_i = V\}$$

17. Tétel. *FGTLEN-ÉLEK-HRSZBEN$\preceq$PARKETTÁZÁS.*

Bizonyítás. Legyen $V,\mathcal{H},k$ az input. Legyen S a maximális élméret paraméter. El kell dönteni, hogy van-e k db diszjunkt él.

A konstrukciót több lépésben végezzük el. Először $\mathcal{H}$-t úgy transzformáljuk, hogy uniform legyen: Minden E élhez $S-|E|$ sok új pontot veszünk fel (különböző élekhez különböző új pontokat használunk). A módosított halmazrendszerre vonatkozó probléma nyilván ekvivalens a kiinduló problémával.

A konstrukció második lépésében már feltesszük, hogy $\mathcal{H}$ egy S-uniform halmazrendszer. Ebben a lépésben $V(H)$-hoz hozzáveszünk $|V(H)|-k\cdot S$ darab új csúcsot (legyen $\widetilde{V}$ a kapott ponthalmaz), $\widetilde{\mathcal{H}}$ elemei pedig $\mathcal{H}$ elemei és minden régi-új csúcspárra egy-egy őket tartalmazó kételemű halmaz.



5. ábra. A 14. Tétel második lépésének redukciója. $|V|-kS=8-2\cdot 3=2$. A két új pont és a hozzájuk tartozó gráfélek a jobb oldalon szerepelnek pirosban.

Észrevétel. $(\widetilde{V},\widetilde{\mathcal{H}})$ parkettázásához le kell fedni a $|V|-kS$ darab új csúcsot, ami csak a $|V|-kS$ darab új csúcskettőssel lehet. A maradék parkettázó élek csak régi élek lehetnek, amelyek kS csúcsot fednek le. Azaz a parkettázás ad k független elt $\mathcal{H}$-ban.

Az észrevétel gondolatmenetének megfordításá teszi teljessé a bizonyítás elméleti részét. ■

⋆ ⋆ ⋆

Definíció. HALMAZRENDSZEREK SZÍNEZÉSI PROBLÉMÁJA: adott egy H halmazrendszer és egy k természetes szám. Kiszínezhetők-e $V(H)$ elemei k színnel, hogy semelyik H-beli halmaz ne legyen monokromatikus?

18. Tétel. *HALMAZRENDSZEREK SZÍNEZÉSI PROBLÉMÁJA $\mathcal{NP}$-teljes.*

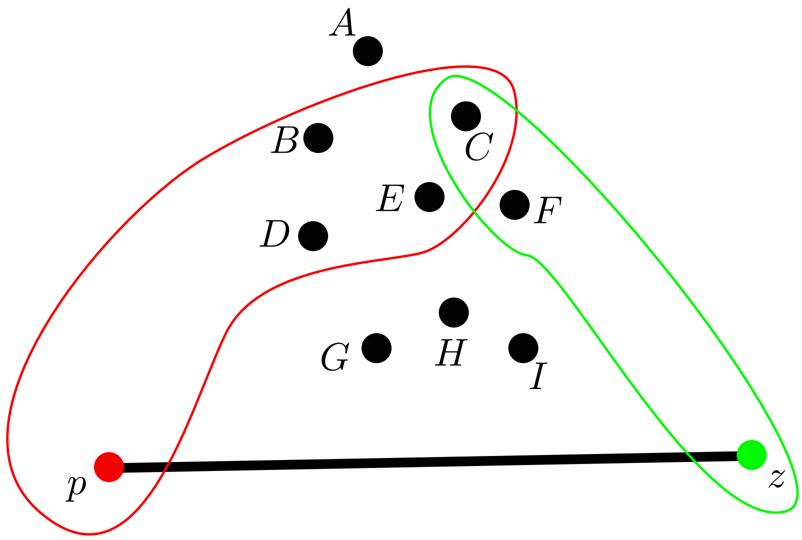
Bizonyítás. A gráfelméleti színezési probléma általánosítása. ■

Gráfok esetén a 2-színezhetőség esete könnyen kezelhető volt.

Definíció. HALMAZRENDSZEREK-2-SZÍNEZHETŐSÉGE: Adott egy $\mathcal{H}$ halmazrendszer. Döntsük el: kiszínezhetők-e $V(H)$ elemei 2 színnel úgy, hogy semelyik H-beli halmaz ne legyen monokromatikus.

19. Tétel. *PARKETTÁZÁS $\preceq$ HALMAZRENDSZEREK-2-SZÍNEZHETŐSÉGE.*

Bizonyítás. Adott egy $V, \mathcal{H}$ input a parkettázás problémához.
Konstrukció: $\widetilde{V} = \mathcal{H} \cup \{p, z\}$. $\widetilde{H}$-hoz minden E, F metsző $\mathcal{H}$-beli élpárra legyen $Z_{E,F} = \{E, F, z\}$ egy él. Minden $v \in V$ esetén legyen $R_v = \{E : v \in E \in \mathcal{H}\} \cup \{p\}$ egy $\widetilde{\mathcal{H}}$-beli él. Továbbá legyen $\{p, z\}$ is egy $\widetilde{\mathcal{H}}$-beli él.



6. ábra. $A, B, C, \ldots, H, I$ pontosan a halmazrendszerünk élei. B, C, D, E pontosan az a elemet tartalamzó élek. C és F élek metszőek. A fenti információkból kiolvasható éleket rajzoltuk be az ábrába, amely a redukció megfelelő töredékét tartalmazza.

Észrevétel. $\widetilde{V}, \widetilde{\mathcal{H}}$ egy 2-színezése esetén legyen p színe piros, z színe zöld (a $\{p, z\}$ él kényszerít a teljes paletta használatára). Az eredeti éleknek megfelelő csúcsok közül a zöld szín kijelöl egy élhalmazt. Ezek parkettázzák az eredti halmazrendszert.

Valóban, köztük két metsző él egy $Z_{E,F}$ típusú zöld-homogén élhez vezetne a redukció eredményében. Míg egy lefedetlen v csúcs (az eredeti halmazrendszerben) adna egy R_v piros-homogén élt.

A gondolatmenet megfordítható, a bizonyítás teljes. ■

Most jegyezzük meg, hogy a HALMAZRENDSZEREK-2-SZÍNEZHETŐSÉGE$\preceq_{\mathcal{P}}$ NEM-MIND-IGAZ-SAT, így kapjuk a hiányzó $\mathcal{NP}$-teljességet: NEM-MIND-IGAZ-SAT is $\mathcal{NP}$-teljes.

A redukció nyilvánvaló: A halmazrendszer alaphalmazának minden elemét azonosítunk egy változóval. Minden élben lévő csúcsok változói vagyolva egy klózt alkotnak. Az ezekből alkotott CNF formula változóinak értékadása azonosítható az alaphalmaz két színnel történő színezésével. Az értékadás akkor lesz nem-mind-igaz kielégítő értékadás, ha nem lesz egyszínű él. A részletek kidolgozását az érdeklődő hallgatóra bízzuk.

⋆ ⋆ ⋆

Az alábbi problémák mindegyike $\mathcal{NP}$-teljes. Ezek nem szerepeltek előadáson. Csak megemlítjük az érdeklődő olvasóknak és ötletet adunk az igazoláshoz.

Definíció. HÁRMASÍTÁS: adott három azonos méretű halmaz és ennek transzverzálisaiból álló 3-uniform halmazrendszer. Van-e a halmazrendszernek olyan részhalmaza, amely a három halmaz uniójának partícióját adja?

Ötlet. Szokásos visszavezetés: 3-SAT.

Definíció. 3-UNIFORM HALMAZRENDSZER PARTÍCIÓ: adott egy 3-uniform halmazrendszer. Van-e olyan részhalmaza, ami az alaphalmaz partíciója?

Ötlet. HÁRMASÍTÁS általánosítása.

5.1. Aritmetizált problémák

Definíció. DIOPHANTOSZI-EGYENLŐTLENSÉGRENDSZER: adott egy $Ax \leq b$ egész együtthatós lineáris egyenlőtlenségrenszer. Van-e megoldása egész számokban?

20. Tétel. *DIOPHANTOSZI-EGYENLŐTLENSÉGRENDSZER $\mathcal{NP}$-teljes.*

Bizonyítás. $\mathcal{NP}$-beliségre tanú egy megoldás.

Az $\mathcal{NP}$-nehézség bizonyításához a SAT-ot vezetjük vissza a problémára: adott egy konjunktív normálforma. Minden x_i változóra bevezetjük a $0 \leq x_i \leq 1$, és minden $\langle z_1, \ldots, z_k \rangle$ klózra a $t_1 + \ldots + t_k$ egyenlőtlenséget, ahol $t_i = x_j$, ha $t_i = x_j$ és $t_i = 1 - x_j$, ha $t_i = \overline{x}_j$. Könnyű látni, hogy az egyenlőtlenségrendszer polinom időben megkonstruálható, és pontosan akkor megoldható, ha a konjunktív normálforma kielégíthető. ■

Definíció. RÉSZLETÖSSZEG=

$$\{\lceil A, b \rceil : A \subset \mathbb{N}, b \in \mathbb{N}, \text{van olyan } R \subset A, \text{ hogy az } R\text{-beli számok összege } b\}.$$

A feladat egy egyszerű értelmezése: A a pénztárcánkban lévő érmék értékeit összegyűjtő halmaz . A, b kódja akkor tartozik az elfogadandó nyelvhez, ha b összeget pontosan ki tudunk fizetni a pénztárcánkból.

21. Tétel. *RÉSZLETÖSSZEG $\mathcal{NP}$-teljes.*

Bizonyítás. RÉSZLETÖSSZEG nyilván $\mathcal{NP}$-beli. Belátjuk, hogy PARKETTÁZÁS $\preceq$ RÉSZLETÖSSZEG.

Legyen $V, \mathcal{H}$ a PARKETTÁZÁS egy inputja. Ki lehet-e választani olyan parkettahalmazt/élhalmazt, amivel ki lehet parkettázni a V-t/padlót?
Konstrukció: Legyen $w : V \to \{1, a, a^2, \ldots, a^{|V|-1}\}$ tetszőleges bijekció. Az értékkészletre gondoljunk mint az a alapú számrendszer helyiértékei.

Legyen $E \in \mathcal{H}$ esetén $a_E = \sum_{v:v\in E} w(v)$. Legyen $A = \{a_E : E \in \mathcal{H}\}$ és $b = 11\ldots1_a = \sum_{v:v\in V} w(v)$. Ezzel leírtuk a részletösszeg probléma egy inputját.

Észrevétel. Ha a értékét $|\mathcal{H}| + 1$-nek választjuk, akkor $a_i \in A$ számok olyanok, hogy minden részletösszeg az a alapú számrendszerben maradék továbbvitele nékül kiszámolható.

Az észrevétel egyből adja, hogy a csupa 1-es számjegyből álló szám előállítása mint részletösszeg ekvivalens az eredeti halmazrendszerre vonatkozó PARKETTÁZÁS feladattal (alkalmasan nagy a esetén).

A redukció során előforduló legnagyobb szám $S = \sum_{i=0}^{|V|-1} a^i = \frac{a^{|V|}-1}{a-1} < a^{|V|}$. Kódjának hossza $|V| \log a = |V| \log(|\mathcal{H}| + 1)$. Redukciónk polinomiális. ■

⋆ ⋆ ⋆

Az alábbi problémák mindegyike $\mathcal{NP}$-teljes. Ezek nem szerepeltek előadáson. Csak megemlítjük az érdeklődő olvasóknak és ötletet adunk az igazoláshoz.

Definíció. HÁTIZSÁK: Adott tárgyak T halmaza. Minden $t \in T$ tárgyhoz tartozik egy V_t térfogat és egy v_t érték ($v_t, V_t \in \mathbb{N}$). Adott egy hátizsák, amelybe legfeljebb H össztérfogatú tárgyakat pakolhatunk. Továbbá adott egy L értékhatár. ($H, L \in \mathbb{N}$.) Kiválasztható-e T egy részhalmaza úgy, hogy elférjen a hátizsákban és összértéke legalább L legyen?

Ötlet. Szokásos visszavezetés: 3-UNIFORM HALMAZRENDSZER PARTÍCIÓ.

Definíció. LÁDAPAKOLÁS: adott egész számok A halmaza, egy b és egy c egész szám. Meg lehet-e adni A egy legfeljebb b osztályú partícióját, amelyben minden osztály összege legfeljebb c?

Ötlet. Szokásos visszavezetés: HÁRMASÍTÁS.