Bonyolultságelmélet gyakorlat – 09
Formulás visszavezetések II.

Nem Mind Egyenlő SAT

- **Input:** Egy φ CNF.
- **Output:** Van-e olyan értékadás, mely φ minden klózában állít legalább egy literált igazra és legalább egy literált hamisra is?

Pl. az $(x \vee y \vee \neg z) \ \wedge \ (\neg x \vee y \vee z)$ formula a problémának egy „igen” példánya, hiszen pl. az $x = 1$, $y = 0$, $z = 1$ értékadás az első klózban az x-et 1-re, y-t pedig 0-ra, a második klózban $\neg x$-et 0-ra, z-t pedig 1-re állítja.

Az $(x \vee y) \ \wedge \ (\neg x \vee y)$ formula viszont egy „nem” példány, ezt a lehetséges értékadások végignézésével (is) megállapíthatjuk.

1. feladat.

Mutassuk meg, hogy a Nem Mind Egyenlő SAT probléma **NP**-teljes!

Másik SAT

- **Input:** egy φ CNF és egy $\mathcal{A}$ kielégítő értékadása.
- **Output:** van-e φ-nek egy **$\mathcal{A}$-tól különböző** kielégítő értékadása is?

Itt természetesen a „különböző értékadás” azt jelenti, hogy olyan értékadás, mely legalább egy φ-ben ténylegesen szereplő változón eltér $\mathcal{A}$-tól.

2. feladat.

Mutassuk meg, hogy Másik SAT is **NP**-nezéz!

FormSAT

- **Input:** egy φ ítéletkalkulus-beli formula.
- **Output:** kielégíthető-e φ?

3. feladat.

Adjunk meg egy FormSAT $\leq_P$ SAT visszavezetést!

1. feladat megoldása.

Kéznfekvőnek tűnik, hogy a SATot vezessük vissza erre a problémára. A gond az, hogy vannak olyan formulák, amik ugyan kielégíthetőek, de nincs ilyen „vegyes” kielégítő értékadásuk (pl. ha a formulában van egységklóz, akkor már eleve nem lehet „igen” példánya a Nem Mind Egyenlő SATnak).

Valamilyen módon „bele kéne csempészni” egy olyan literált minden egyes klózba, melyet akár hamisra is állíthatunk, első ötletnek adhatja magát a következő konstrukció: vegyünk fel egy új x változót, és azt rakjuk bele az összes klózba pluszban! Azaz,

$$C_1 \wedge C_2 \wedge \ldots \wedge C_n \;\mapsto\; (C_1 \vee x) \wedge (C_2 \vee x) \wedge \ldots \wedge (C_n \vee x)$$

Az világos, hogy ha $\mathcal{A}$ egy kielégítő értékadása az eredeti φ formulának, akkor az az $\mathcal{A}'$, melyet úgy kapunk, hogy $\mathcal{A}'(x) = 0$ és minden eredeti y változóra $\mathcal{A}'(y) = \mathcal{A}(y)$, ezt a generált φ' formulát „vegyesen” elégíti ki: az eredeti klózokban $\mathcal{A}$ mellett mindegyikben lesz 1 értékű literál, és az $\mathcal{A}'(x) = 0$ miatt így minden φ'-beli klózban lesz 1 értékű és 0 értékű literál is.

Vajon kielégíthető-e az eredeti φ formulánk, ha tudjuk, hogy φ' a Nem Mind Egyenlő SAT problémának egy „igen” példánya? Ekkor van olyan $\mathcal{A}'$ értékadás, mely mellett minden $(C_i \vee x)$ klózban szerepel 1 értékű literál is és 0 értékű literál is. Ha $\mathcal{A}'(x) = 0$, akkor ez azt jelenti, hogy $\mathcal{A}'$ kielégíti az eredeti φ formulát is (mert ekkor minden C_i klózban szerepel emellett az értékadás mellett 1 értékű literál), ez az ág OK. Ha viszont $\mathcal{A}'(x) = 1$, akkor $\mathcal{A}'$-ról csak annyit tudunk, hogy φ minden klózában legalább egy literált 0-ra állít (hiszen x-et állítja 1-re és minden klózban lesz 0 és lesz 1 értékű literál is).

Ekkor viszont az a $\mathcal{B}$ értékadás, melyre minden y változó esetén $\mathcal{B}(y) = \neg\mathcal{A}'(y)$, biztos, hogy φ minden klózában legalább egy literált 1-re állít, így φ ekkor is kielégíthető, és ez a konverzió valóban egy (polinomidejű) választartó visszavezetés SATról Nem Mind Egyenlő SATra.

2. feladat megoldása.

Erre a problémára is a SAT visszavezetése látszik egy kézenfekvő opciónak.

Kellene tehát egy olyan $\varphi \mapsto (\varphi', \mathcal{A})$ konstrukciót adnunk, mely egy φ CNF-ből egy olyan φ' CNF-et készít, mely **mindenképp** kielégíthető, mégpedig például a szintén a konstrukció által készített $\mathcal{A}$ értékadással, de

- ha φ kielégíthetetlen, akkor csak $\mathcal{A}$ legyen az egyetlen, φ'-t kielégítő értékadás,
- ha pedig φ kielégíthető, akkor φ'-nek legyen másik kielégítő értékadása is.

Az előző feladat visszavezetése:

$$C_1 \wedge C_2 \wedge \ldots \wedge C_n \mapsto (C_1 \vee x) \wedge (C_2 \vee x) \wedge \ldots \wedge (C_n \vee x)$$

abból a szempontból hasonló volt ehhez a követelményhez, hogy egy olyan formulát konstruál, mely mindenképp kielégíthető pl. úgy, hogy minden változót 1-re állítunk (hiszen ekkor x is igaz lesz), azonban ez még nem teljesíti azt a kritériumot, hogy ez legyen az egyetlen kielégítő értékadás, ha φ kielégíthetetlen (hiszen pl. az $x = 1$ és minden más 0 értékadás is biztosan kielégíti ezt a formulát).

Azonban, ha ezt a konstrukciót kiegészítünk egy „ha $x = 1$, akkor minden eredeti változó értéke is 1 kell legyen" feltétellel, akkor már olyan konverziót kapunk, mely teljesíti a feltételeket, hiszen:

- ha φ kielégíthető mondjuk egy $\mathcal{A}$ értékadással, akkor az az értékadás, mely ugyanaz, mint $\mathcal{A}$, és melyben az új x értéke 0, egy, a konstans 1-től eltérő, tehát „másik" értékadás lesz,
- ha pedig φ kielégíthetetlen, akkor a generált φ'-t csak olyan értékadás tudja kielégíteni, melyben x értéke 1, ami miatt pedig minden eredeti változó értéke is 1 kell legyen, azaz ekkor csak egy kielégítő értékadása lesz a generált φ' formulának, mégpedig a konstans 1.

Ezt elérhetjük azzal, hogy ha φ eredeti változói $x_1, \ldots, x_n$, akkor felveszünk $x \to x_i$, azaz $(\neg x \vee x_i)$ alakú klózokat. Tehát a visszavezetésünk, ha $\varphi = C_1 \wedge \ldots \wedge C_k$, és φ-ben az $x_1, \ldots, x_n$ változók szerepelnek:

$$\begin{aligned}\varphi' = (C_1 \vee x) \ \wedge \ (C_2 \vee x) \ \wedge \ \ldots \ \wedge \ (C_k \vee x) \\ \wedge \ (\neg x \vee x_1) \ \wedge \ (\neg x \vee x_2) \ \wedge \ \ldots \ \wedge \ (\neg x \vee x_n),\end{aligned}$$

és $\mathcal{A}$ legyen az az értékadás, mely minden változóhoz 1-et rendel.

3. feladat megoldása.

A két probléma közt annyi a különbség, hogy a SATban CNF az input, a FORMSATban meg nem feltétlenül. Tanultuk több tárgyból is korábban, hogy minden formulát lehet CNF-re hozni (pl. $\rightarrow$ és $\leftrightarrow$ eliminálásával, majd deMorgan-azonosságokkal, végül disztributivitás alkalmazásával), és ez valóban egy választartó inputkonverzió lenne, **de nem polinomidejű**, hiszen pl. ha a formulánk a

$$(x_1 \wedge y_1) \vee (x_2 \wedge y_2) \vee \ldots \vee (x_n \wedge y_n)$$

(ún. diszjunktív normálalakú formula), akkor ezen ha a disztributivitást alkalmazzuk, úgy kapunk 2^n darab, egyenként n literálból álló klózt, tehát ez a konverzió, mely potenciálian egy exponenciális méretű formulát állít elő, semmiképp nem lehet polinomidejű.

Az előadáson a hálózat-kiértékelésre látott módszer viszont működhet: ott minden logikai kaput egy-egy **új változóval** kódoltunk el, és „kényszerítettük", hogy ezek a változók a kapuk tényleges kimeneti értékét vegyék fel értékül minden kielégítő értékadás mellett.

Ez most is működik, hiszen egy formula valójában egy nagyon speciális (fa) alakú hálózat: φ minden egyes ψ részformulájára bevezetünk egy új, x_ψ változót és:

- ha $\psi = \psi_1 \vee \psi_2$ alakú, akkor az outputba generáljuk az $x_\psi \leftrightarrow (x_{\psi_1} \vee x_{\psi_2})$ formulát, azaz a $(\neg x_\psi \vee x_{\psi_1} \vee x_{\psi_2}) \wedge (\neg x_{\psi_1} \vee x_\psi) \wedge (\neg x_{\psi_2} \vee x_\psi)$ klózokat;
- ha $\psi = \psi_1 \wedge \psi_2$ alakú, akkor az outputba generáljuk az $x_\psi \leftrightarrow (x_{\psi_1} \wedge_{\psi_2})$ formulát, azaz a $(\neg x_\psi \vee x_{\psi_1}) \wedge (\neg x_\psi \vee x_{\psi_2}) \wedge (\neg x_{\psi_1} \vee \neg x_{\psi_2} \vee x_\psi)$ klózokat;
- ha $\psi = \neg\psi_1$ alakú, akkor az outputba generáljuk az $x_\psi \leftrightarrow \neg x_{\psi_1}$, azaz a $(\neg x_\psi \vee \neg x_{\psi_1}) \wedge (x_\psi \vee x_{\psi_2})$ klózokat;
- ha $\psi = \psi_1 \rightarrow \psi_2$, akkor az outputba generáljuk az $x_\psi \leftrightarrow (x_{\psi_1} \rightarrow x_{\psi_2})$, azaz a $(\neg x_\psi \vee \neg x_{\psi_1} \vee x_{\psi_2}) \wedge (x_{\psi_1} \vee x_\psi) \wedge (\neg x_{\psi_2} \vee x_\psi)$ klózokat;
- ha $\psi = \psi_1 \leftrightarrow \psi_2$, akkor az outputba generáljuk az $x_\psi \leftrightarrow (x_{\psi_1} \leftrightarrow x_{\psi_2})$, azaz a $(\neg x_\psi \vee \neg x_{\psi_1} \vee \neg x_{\psi_2}) \wedge (\neg x_\psi \vee \neg x_{\psi_2} \vee x_{\psi_1}) \wedge (\neg x_{\psi_1} \vee \neg x_{\psi_2} \vee x_\psi) \wedge (x_{\psi_1} \vee x_{\psi_2} \vee x_\psi)$ klózokat;
- ha pedig $\psi = x_i$, akkor eljárhatunk úgy is, hogy az x_ψ változót x_i-nek deklaráljuk, vagy úgy is, hogy felvesszük az $x_\psi \leftrightarrow x_i$, azaz a $(\neg x_\psi \vee x_i) \wedge (\neg x_i \vee x_\psi)$ klózokat,
- továbbá, felvesszük az x_φ egységklózt.

Példa: ha $\varphi = x \vee (y \wedge (\neg z \vee x))$, akkor a generált formulánk (azzal, hogy $\neg z$ helyére az x_1, $\neg z \vee x$ helyére az x_2, $(y \wedge (\neg z \vee x))$ helyére az x_3, az egész formula helyére pedig az x_4 változót vezetjük be, úgy a (most már polinomidejű) visszavezetés kimenete:

$$\begin{aligned}
&(x_1 \vee z) \wedge (\neg x_1 \vee \neg z) \wedge \\
&(\neg x_2 \vee x_1 \vee x) \wedge (\neg x_1 \vee x_2) \wedge (\neg x \vee x_2) \wedge \\
&(\neg x_3 \vee y) \wedge (\neg x_3 \vee x_2) \wedge (\neg y \vee \neg x_2 \vee x_3) \wedge \\
&(\neg x_4 \vee x \vee x_3) \wedge (\neg x \vee x_4) \wedge (\neg x_3 \vee x_4) \wedge \\
&x_4.
\end{aligned}$$