Bonyolultságelmélet

Thursday 1st December, 2016, 23:02

A polinomiális tár

QSat

Adott: $\exists x_1 \forall x_2 \dots Q_m x_m \varphi$ kvantifikált Boole-formula prenex normálalakban úgy, hogy a φ konjunktív normálalakú kvantormentes formula, melyben legfeljebb az $x_1, \dots, x_m$ változók fordulnak elő.

Kérdés: Igaz-e az adott formula?

Megjegyzés

- A kvantorok szigorú alternálása nem lényeges.
- Az sem lényeges, hogy az első kvantor $\exists$.

QSAT példa

Példa

$$\exists x\ \forall y(\ (x \vee y)\ \wedge\ (\neg x \vee \neg y)\ )$$

hamis, hiszen $x = 0$ esetén $y = 0$, $x = 1$ esetén pedig $y = 1$ választása mellett hamis lesz a formula magja.

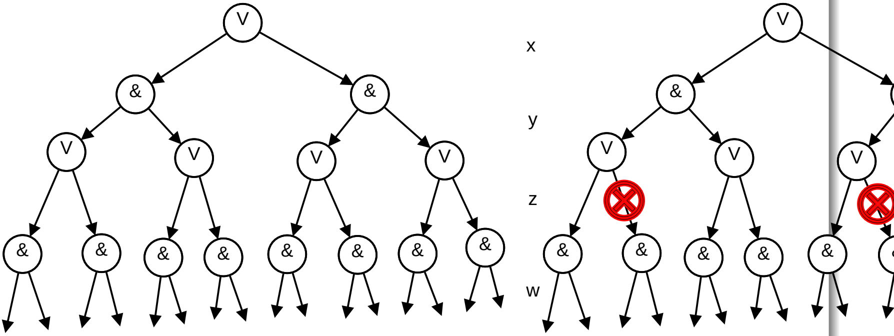
Példa

$$\exists x\ \forall y\ \exists z\ \forall w\big((\neg x \vee z \vee w)\ \wedge\ (y \vee \neg z)\ \wedge\ (x \vee \neg y \vee z)\big)$$

?

A hosszabb példa

$\exists x\ \forall y\ \exists z\ \forall w\big((\neg x \vee z \vee w)\ \wedge\ (y \vee \neg z)\ \wedge\ (x \vee \neg y \vee z)\big)$



A formula **igaz**: $x = 0$ választása esetén tetszőleges y-ra legyen $z = y$, ekkor a formula minden w-re igaz lesz.

Állítás

A $\textsc{QSat}$ probléma $\mathbf{PSPACE}$-ben van.

Algoritmus

Bemenet: $Q_1x_1\dots Q_mx_m\varphi$ alakú zárt formula.
Kimenet: a formula értéke.

function $\textsc{QSat}(F)$
  if $F == \exists x F'$ **then**
    return $\textsc{QSat}(F'[x/0])$ or $\textsc{QSat}(F'[x/1])$
  else if $F == \forall x F'$ **then**
    return $\textsc{QSat}(F'[x/0])$ and $\textsc{QSat}(F'[x/1])$
  else
    return $\textsc{eval}(F)$

Ez az algoritmus $\mathcal{O}(n^2)$ tárban kiszámítja a formula értékét (ahol $\textsc{Eval}$ a változómentes formulát polinom időben és lineáris tárban kiértékelő függvény).

Megjegyzés

A probléma akkor is **PSPACE**-ben van, ha

- φ tetszőleges kvantormentes Boole-formula.
- a kvantorok nem feltétlenül váltakoznak.
- φ-ben az $x_1, \ldots, x_m$ változókon kívül egyéb változók is előfordulnak, és azt kérdezzük, kielégíthető-e az egész formula.
- a formula nem feltétlenül prenex alakú.
- azt kérdezzük, hogy a formula hamis-e.

QSat

Tétel

A QSat probléma **PSPACE**-teljes.

Ötlet

- Egy tetszőleges n^k tárkorlátos M programhoz és x inputjához konstruálunk egy φ formulát, mely pontosan akkor igaz, ha M elfogadja x-et.
- Egy konfigurációt n^k darab bináris változóval kódolunk.
- Tetszőleges t-re a φ_t formulát, aminek szabad változói $x_1, \ldots, x_{n^k}, y_1, \ldots, y_{n^k}$, úgy konstruáljuk meg, hogy $\varphi_t(x_1, \ldots, x_{n^k}, y_1, \ldots, y_{n^k})$ pontosan akkor igaz, ha az $(x_1, \ldots, x_{n^k})$ konfigurációból az M program $\mathbf{2^t}$ **lépésen belül** az $(y_1, \ldots, y_{n^k})$ konfigurációba jut; továbbá φ_t ,,nem túl hosszú".
- A keresett formula: φ_{n^k}, behelyettesítve a kezdő- és az elfogadó konfigurációt.

$\varphi_{t+1}(x_1, \ldots, x_{n^k}, y_1, \ldots, y_{n^k})$

Jelölje X az $x_1, \ldots, x_{n^k}$ változó-vektort, Y az $y_1, \ldots, y_{n^k}$-t stb.
A Savitch-tételben látottal analóg módon:

$$\exists Z\big(\varphi_t(X, Z) \wedge \varphi_t(Z, Y)\big),$$

de ez túl hosszú! (Exponenciális hosszú formula.)
Ehelyett:

$$\exists Z \forall X', Y' \Big(\big((X' = X \wedge Y' = Z) \vee (X' = Z \wedge Y' = Y)\big) \to \varphi_t(X', Y')\Big),$$

ahol $A = B$ az $(a_1 \leftrightarrow b_1) \wedge \ldots \wedge (a_{n^k} \leftrightarrow b_{n^k})$ formula rövidítése.
A formula magja (polinom méretű) CNF-re is könnyen hozható.

További **PSPACE**-teljes problémák

HELYBEN ELFOGADÁS

Adott: M determinisztikus program és x bemenő szó.

Kérdés: Elfogadja-e M az x-et legfeljebb $|x|$ tárban?

REGULÁRIS KIFEJEZÉSEK EKVIVALENCIÁJA

Adott: Két reguláris kifejezés.

Kérdés: Ugyanazt a nyelvet jelölik-e?

VÉGES AUTOMATÁK EKVIVALENCIÁJA

Adott: M_1 és M_2 véges **nemdeterminisztikus** automaták.

Kérdés: M_1 és M_2 ekvivalensek-e?

Tétel

A fenti három probléma mind **PSPACE**-teljes.

Készülődés a **PSPACE** egy meglepő karakterizációjára

Alternálás

Visszatérünk még a **PSPACE**-re, de előbb bevezetünk egy fogalmat, az **alternálást**.

Alternáló program

Definíció

Az **alternáló program** olyan program, melyben (a nemdeterminisztikushoz hasonlóan) egy sorszámot több sornak is kioszthatunk, **továbbá** minden sorszám vagy ÉS, vagy VAGY típusú sorszám.
Tekintsük a program számítási fáját az x bemeneten.
Azt mondjuk, hogy egy C konfiguráció **végül elfogadó**, ha

- C-ben a gép ACCEPT soron áll, vagy
- ÉS-sorszámon van, és **minden** közvetlen leszármazottja végül elfogadó, vagy
- VAGY-sorszámon van, és **valamely** közvetlen leszármazottja végül elfogadó.

A program akkor **fogadja el** az x bemenetet, ha a kezdő konfiguráció végül elfogadó, különben elutasítja azt. Az $f(n)$ idő- vagy tárkorlátos alternáló program fogalmát értelemszerűen definiáljuk.

Alternáló program

Definíció

Legyen $f(n)$ megengedett bonyolultsági függvény. (Idő esetén $f(n) > n$.)
$\mathbf{ATIME(\boldsymbol{f(n)})}$: az összes $f(n)$ időkorlátos alternáló programmal eldönthető nyelvek.
$\mathbf{ASPACE(\boldsymbol{f(n)})}$: az összes $f(n)$ tárkorlátos alternáló programmal eldönthető nyelvek.
$\mathbf{AL}= \mathrm{ASPACE}(\log n)$
$\mathbf{AP}= \mathrm{ATIME}(n^k)$

Világos, hogy $\mathrm{NTIME}(f(n)) \subseteq \mathrm{ATIME}(f(n))$ és $\mathrm{NSPACE}(f(n)) \subseteq \mathrm{ASPACE}(f(n))$.

Hiszen a nemdeterminisztikus program olyan alternáló program, melynek csak VAGY típusú sorai vannak.

Alternálás és determinizmus

Az alternáló bonyolultsági osztályok és a ,,klasszikus'' bonyolultsági osztályok közt érdekes összefüggések állnak fenn:

Tétel

$$\mathbf{AL} = \mathbf{P}.$$

Tétel

$$\mathbf{AP} = \mathbf{PSPACE}.$$

Az utóbbi tételt megfogalmazhatjuk így is:
PSPACE-be pontosan azok a problémák tartoznak, melyek reprezentálhatók zéró összegű, teljes információs, polinom lépésszámú kétszemélyes játékként.

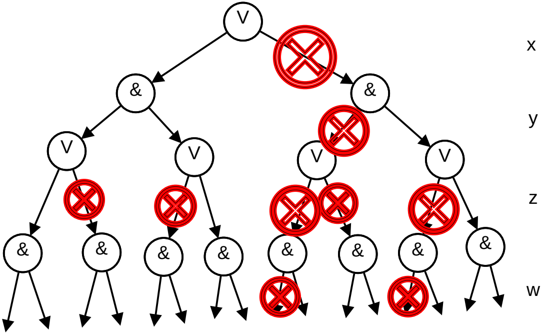
Kétszemélyes játékok

A QSAT felfogható **kétszemélyes játékként**:

- Játékosok: $\exists$, $\forall$
- Lépések felváltva: x_1 értéke, x_2 értéke, ...
- $\exists$ célja: kielégíteni a formula magját
- $\forall$ célja: hamissá tenni

A hosszabb példa

$\exists x\ \forall y\ \exists z\ \forall w\big((\neg x \vee z \vee w)\ \wedge\ (y \vee \neg z)\ \wedge\ (x \vee \neg y \vee z)\big)$



,,A formula **igaz**: $x = 0$ választása esetén tetszőleges y-ra legyen $z = y$ és az minden w-re jó lesz."

Egy nyerő stratégia $\exists$-nek: válassza $x = 0$-t, majd $\forall$ választja y értékét, $\exists$ ugyenezt az értéket adja z-nek, $\forall$ bárhogy is választja meg w értékét, a formula magja igaz lesz.

Egy kétszemélyes játék

FÖLDRAJZI JÁTÉK

Adott: $G = (V, E)$ irányított gráf, $1 \in V$ kezdőcsúcs.

Kérdés: Az I. játékos nyer-e az alábbi játékban?

- Az I. játékos kezd az 1 csúcs megnevezésével.
- Minden lépésben minden játékosnak egy olyan csúcsot kell választani, amely még nem szerepelt, és amelybe vezet él az előzőleg megnevezett csúcsból.
- Egy játékos veszít, ha végül nem tud ilyen csúcsot megnevezni.

Állítás

A Földrajzi Játék probléma **PSPACE**-ben van.

Ötlet

- A lépéssorozatok hossza a bemenet méretében polinomiális.
- Létezik olyan polinom tárigényű algoritmus, mely adott állásra megkonstruálja az állás rákövetkezőit, vagy ha ilyen nincs, eldönti, melyik játékos nyert.

Minden ilyen kétszemélyes játék **PSPACE**-ben van:

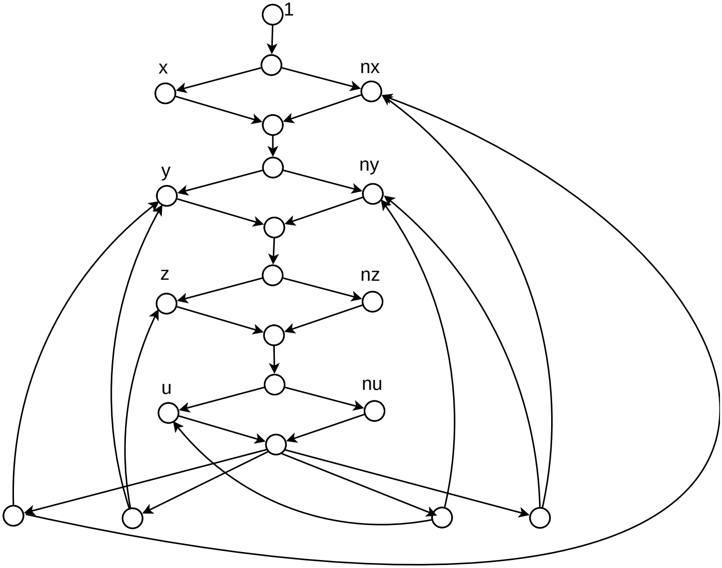
- Polinom tárral elkészítjük az adott játék fáját.
- Mélységgel arányos tárral eldöntjük, kinek van nyerő stratégiája.

A két algoritmus összetehető: polinom tárkorlát.

Állítás

A Földrajzi Játék probléma **PSPACE**-teljes.

$$\exists x \forall y \exists z \forall u \big(\ (y \vee \neg x) \wedge (y \vee z) \wedge (\neg y \vee u) \wedge (\neg x \vee \neg y)\big)$$



- Az első játékos dönti el az egzisztenciálisan, a második az univerzálisan kvantifikált változó értékét, mely a címkével ellentétes.
- A második játékos választ egy tagot.
- Az első játékos akkor és csak akkor nyer, ha a tag valamely literálja igaz (visszavezető él választható).
- Mivel ez minden tagra igaz, az első játékos nyer $\Leftrightarrow$ a formula igaz.

Megjegyzés

Irányítatlan gráfok esetén a probléma $\mathbf{P}$-ben van.

További közismert **PSPACE**-teljes problémák

- GÓ: input egy gó állás, melyik játékosnak van nyerő stratégiája?
- HEX: input egy hex állás, melyik játékosnak van nyerő stratégiája?
- REVERSI: input egy reversi állás...
- DÁMA: input egy dámaállás...
- AMŐBA: egy amőbaállás...
- SOKOBAN: megoldható-e az input Sokoban feladvány? (egyszemélyes!)
- RUSH HOUR: megoldható-e az input Rush Hour feladvány?
- ÉLETJÁTÉK: kipusztul-e minden cella egy adott kezdőállásból elindítva valahány lépésben? („nulla személyes”!)

Túl a $\mathbf{PSPACE}$ osztályon

Definíció

$\mathbf{EXP} = \mathrm{TIME}(2^{n^k})$
$\mathbf{NEXP} = \mathrm{NTIME}(2^{n^k})$

Világos, hogy $\mathbf{EXP} \subseteq \mathbf{NEXP}$ és tudjuk, hogy $\mathbf{PSPACE} \subseteq \mathbf{EXP}$.
Nem ismert, hogy az $\mathbf{EXP}$ és $\mathbf{NEXP}$ osztályok megegyeznek-e.

Tétel

Ha $\mathbf{P} = \mathbf{NP}$, akkor $\mathbf{EXP} = \mathbf{NEXP}$.

Definíció

$\mathbf{EXPSPACE} = \mathrm{SPACE}(2^{n^k})$

Világos, hogy $\mathbf{NEXP} \subseteq \mathbf{EXPSPACE}$.

Tétel

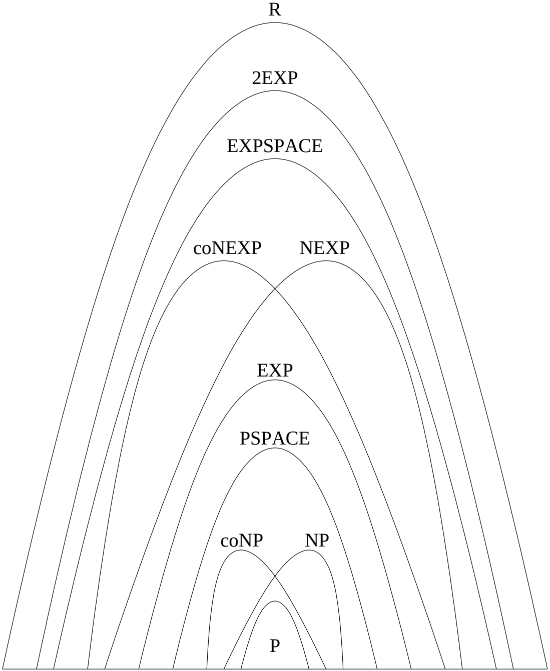
Az alábbi probléma $\mathbf{EXPSPACE}$-teljes: adott két reguláris kifejezés, melyekben négyzetreemelés is szerepelhet, ekvivalensek-e a kifejezések?

Definíció

$\mathbf{ELEMI} = \mathrm{TIME}(2^{n^k}) \cup \mathrm{TIME}(2^{2^{n^k}}) \cup \mathrm{TIME}(2^{2^{2^{n^k}}}) \cup \ldots$

Tétel

Az alábbi eldönthető probléma **nem elemi**: adott két reguláris kifejezés, melyekben komplementerképzés is szerepelhet, ekvivalensek-e a kifejezések?


R
2EXP
EXPSPACE
coNEXP
NEXP
EXP
PSPACE
coNP
NP
P

Összefoglalás

- Megismertük a $\textsc{QSat}$ problémát és beláttuk, hogy **PSPACE**-teljes.
- Kimondtuk, hogy a $\textsc{Helyben Elfogadás}$, $\textsc{Regex Ekvivalencia}$ és $\textsc{NFA Ekvivalencia}$ problémák **PSPACE**-teljesek.
- Megismertük az alternáló programokat és az $\mathrm{ATIME}(f(n))$, $\mathrm{ASPACE}(f(n))$ osztályokat.
- Beláttuk, hogy $\mathrm{NTIME}(f(n)) \subseteq \mathrm{ATIME}(f(n))$ és $\mathrm{NSPACE}(f(n)) \subseteq \mathrm{ASPACE}(f(n))$.
- Kimondtuk, hogy $\mathbf{AL} = \mathbf{P}$ és $\mathbf{AP} = \mathbf{PSPACE}$.
- Megismertük a $\textsc{Földrajzi Játék}$ problémát és beláttuk, hogy **PSPACE**-teljes.
- Felsoroltunk néhány még nehezebb problémát.