

Bonyolultságelmélet

Monday 3rd October, 2016, 16:58

Memó: első kiszámítható teljes problémáink

Emlékezzünk: egy \mathcal{C} -beli probléma \mathcal{C} -teljes, ha **minden** \mathcal{C} -beli probléma visszavezethető rá.

Memó: első kiszámítható teljes problémáink

Emlékezzünk: egy \mathcal{C} -beli probléma \mathcal{C} -teljes, ha **minden** \mathcal{C} -beli probléma visszavezethető rá.

A HÁLÓZAT-KIÉRTÉKELÉS probléma \mathbf{P} -beli.

Memó: első kiszámítható teljes problémáink

Emlékezzünk: egy \mathcal{C} -beli probléma \mathcal{C} -teljes, ha **minden** \mathcal{C} -beli probléma visszavezethető rá.

A HÁLÓZAT-KIÉRTÉKELÉS probléma \mathbf{P} -beli.

Tétel

A HÁLÓZAT-KIELÉGÍTHETŐSÉG probléma \mathbf{NP} -teljes.

Memó: első kiszámítható teljes problémáink

Emlékezzünk: egy \mathcal{C} -beli probléma \mathcal{C} -teljes, ha **minden** \mathcal{C} -beli probléma visszavezethető rá.

A HÁLÓZAT-KIÉRTÉKELÉS probléma \mathbf{P} -beli.

Tétel

A HÁLÓZAT-KIELÉGÍTHETŐSÉG probléma \mathbf{NP} -teljes.

Következmény (Cook tétele)

A SAT probléma is \mathbf{NP} -teljes.

Mivel a SAT probléma NP-teljes,

Mivel a SAT probléma NP-teljes,

- ha polinomidőben megoldható lenne, akkor $P = NP$.

Mivel a SAT probléma NP-teljes,

- ha polinomidőben megoldható lenne, akkor $P = NP$.
(Ebben persze lehet **hinni**. De eddig még senki nem oldotta meg polinomidőben. . .)

Mivel a SAT probléma NP-teljes,

- ha polinomidőben megoldható lenne, akkor $P = NP$.
(Ebben persze lehet **hinni**. De eddig még senki nem oldotta meg polinomidőben. . .)
- Nem ismert rá **szubexponenciális** algoritmus.

Mivel a SAT probléma NP-teljes,

- ha polinomidőben megoldható lenne, akkor $P = NP$.
(Ebben persze lehet **hinni**. De eddig még senki nem oldotta meg polinomidőben. . .)
- Nem ismert rá **szubexponenciális** algoritmus.
- Ilyenkor bevethetők (a teljesség igénye nélkül):

Mivel a SAT probléma NP-teljes,

- ha polinomidőben megoldható lenne, akkor $P = NP$.
(Ebben persze lehet **hinni**. De eddig még senki nem oldotta meg polinomidőben. . .)
- Nem ismert rá **szubexponenciális** algoritmus.
- Ilyenkor bevethetők (a teljesség igénye nélkül):
 - **heurisztikák**

Mivel a SAT probléma NP-teljes,

- ha polinomidőben megoldható lenne, akkor $P = NP$.
(Ebben persze lehet **hinni**. De eddig még senki nem oldotta meg polinomidőben. . .)
- Nem ismert rá **szubexponenciális** algoritmus.
- Ilyenkor bevethetők (a teljesség igénye nélkül):
 - **heurisztikák**
 - **randomizált algoritmusok** alkalmazása

Mivel a SAT probléma NP-teljes,

- ha polinomidőben megoldható lenne, akkor $P = NP$.
(Ebben persze lehet **hinni**. De eddig még senki nem oldotta meg polinomidőben. . .)
- Nem ismert rá **szubexponenciális** algoritmus.
- Ilyenkor bevethetők (a teljesség igénye nélkül):
 - **heurisztikák**
 - **randomizált algoritmusok** alkalmazása
 - a követelmények **relaxálása**. . .

Mivel a SAT probléma NP-teljes,

- ha polinomidőben megoldható lenne, akkor $P = NP$.
(Ebben persze lehet **hinni**. De eddig még senki nem oldotta meg polinomidőben. . .)
- Nem ismert rá **szubexponenciális** algoritmus.
- Ilyenkor bevethetők (a teljesség igénye nélkül):
 - **heurisztikák**
 - **randomizált algoritmusok** alkalmazása
 - a követelmények **relaxálása. . .** (pl. nem az összes, de minél több klóz egyszerre történő kielégítése)

Mivel a SAT probléma NP-teljes,

- ha polinomidőben megoldható lenne, akkor $P = NP$.
(Ebben persze lehet **hinni**. De eddig még senki nem oldotta meg polinomidőben. . .)
- Nem ismert rá **szubexponenciális** algoritmus.
- Ilyenkor bevethetők (a teljesség igénye nélkül):
 - **heurisztikák**
 - **randomizált algoritmusok** alkalmazása
 - a követelmények **relaxálása**. . . (pl. nem az összes, de minél több klóz egyszerre történő kielégítése)
 - . . . majd a relaxált követelmények **approximálása**

Mivel a SAT probléma NP-teljes,

- ha polinomidőben megoldható lenne, akkor $P = NP$.
(Ebben persze lehet **hinni**. De eddig még senki nem oldotta meg polinomidőben. . .)
- Nem ismert rá **szubexponenciális** algoritmus.
- Ilyenkor bevethetők (a teljesség igénye nélkül):
 - **heurisztikák**
 - **randomizált algoritmusok** alkalmazása
 - a követelmények **relaxálása**. . . (pl. nem az összes, de minél több klóz egyszerre történő kielégítése)
 - . . . majd a relaxált követelmények **approximálása**
 - vagy olyan **speciális esetek** keresése, melyre van hatékony algoritmus.

A kielégíthetőség változatai

A SAT problémának vizsgálhatjuk **speciális eseteit**, pl:

A kielégíthetőség változatai

A SAT problémának vizsgálhatjuk **speciális eseteit**, pl:

- klónként csak **konstans sok** literált engedünk meg

A kielégíthetőség változatai

A SAT problémának vizsgálhatjuk **speciális eseteit**, pl:

- klózonként csak **konstans sok** literált engedünk meg
- csak speciális (pl. **Horn**) alakú klózokat engedünk meg

A kielégíthetőség változatai

A SAT problémának vizsgálhatjuk **speciális eseteit**, pl:

- klózonként csak **konstans sok** literált engedünk meg
- csak speciális (pl. **Horn**) alakú klózokat engedünk meg

Definíció

Legyen $k \geq 1$. A $k\text{SAT}$ a SAT azon speciális esete, amelyben minden klóz pontosan k (nem feltétlenül különböző) literálból áll.

Állítás

3SAT **NP**-teljes, és így $k \geq 3$ esetén k SAT **NP**-teljes.

Állítás

3SAT **NP**-teljes, és így $k \geq 3$ esetén k SAT **NP**-teljes.

Bizonyítás

Állítás

3SAT **NP**-teljes, és így $k \geq 3$ esetén k SAT **NP**-teljes.

Bizonyítás

$$l \mapsto l \vee l \vee l$$

Állítás

3SAT **NP**-teljes, és így $k \geq 3$ esetén k SAT **NP**-teljes.

Bizonyítás

$$l \mapsto l \vee l \vee l$$
$$l_1 \vee l_2 \mapsto l_1 \vee l_2 \vee l_2$$

Állítás

3SAT **NP**-teljes, és így $k \geq 3$ esetén k SAT **NP**-teljes.

Bizonyítás

$$l \mapsto l \vee l \vee l$$

$$l_1 \vee l_2 \mapsto l_1 \vee l_2 \vee l_2$$

$$l_1 \vee l_2 \vee l_3 \mapsto l_1 \vee l_2 \vee l_3$$

Állítás

3SAT NP-teljes, és így $k \geq 3$ esetén k SAT NP-teljes.

Bizonyítás

$$l \mapsto l \vee l \vee l$$

$$l_1 \vee l_2 \mapsto l_1 \vee l_2 \vee l_2$$

$$l_1 \vee l_2 \vee l_3 \mapsto l_1 \vee l_2 \vee l_3$$

$$l_1 \vee l_2 \vee l_3 \vee l_4 \mapsto (l_1 \vee l_2 \vee x) \wedge (\neg x \vee l_3 \vee l_4)$$

Állítás

3SAT NP-teljes, és így $k \geq 3$ esetén k SAT NP-teljes.

Bizonyítás

$$l \mapsto l \vee l \vee l$$

$$l_1 \vee l_2 \mapsto l_1 \vee l_2 \vee l_2$$

$$l_1 \vee l_2 \vee l_3 \mapsto l_1 \vee l_2 \vee l_3$$

$$l_1 \vee l_2 \vee l_3 \vee l_4 \mapsto (l_1 \vee l_2 \vee x) \wedge (\neg x \vee l_3 \vee l_4)$$

$$\vdots \quad \quad \quad \vdots$$

$$l_1 \vee l_2 \vee \dots \vee l_n \mapsto (l_1 \vee l_2 \vee x) \wedge (\neg x \vee l_3 \vee y) \wedge \\ (\neg y \vee l_4 \vee z) \wedge \dots \wedge (\neg u \vee l_{n-1} \vee l_n)$$

Állítás

Legyen $\varphi = c_1 \wedge c_2 \wedge \dots \wedge c_k$ konjunktív normálformájú formula.

Állítás

Legyen $\varphi = c_1 \wedge c_2 \wedge \dots \wedge c_k$ konjunktív normálformájú formula. Minden c_i klózhoz legyen c'_i az előzőekben adott kifejezés, ahol minden kifejezésben új változókat használunk.

Állítás

Legyen $\varphi = c_1 \wedge c_2 \wedge \dots \wedge c_k$ konjunktív normálformájú formula. Minden c_i klózhoz legyen c'_i az előzőekben adott kifejezés, ahol minden kifejezésben új változókat használunk.

Ekkor φ akkor és csak akkor kielégíthető, ha $\varphi' = c'_1 \wedge c'_2 \wedge \dots \wedge c'_k$ az.

Állítás

Legyen $\varphi = c_1 \wedge c_2 \wedge \dots \wedge c_k$ konjunktív normálformájú formula. Minden c_i klózhoz legyen c'_i az előzőekben adott kifejezés, ahol minden kifejezésben új változókat használunk.

Ekkor φ akkor és csak akkor kielégíthető, ha $\varphi' = c'_1 \wedge c'_2 \wedge \dots \wedge c'_k$ az.

Mivel $\text{SAT} \leq_{\mathcal{P}} 3\text{SAT}$ és $3\text{SAT} \in \mathbf{NP}$, ezért SAT \mathbf{NP} -teljességéből következik, hogy 3SAT is \mathbf{NP} -teljes.

A 2SAT problémára viszont adható hatékony algoritmus.

A 2SAT problémára viszont adható hatékony algoritmus.
Legyen φ egy 2 – CNF (minden klóz két literálból áll).

A 2SAT problémára viszont adható hatékony algoritmus.
Legyen φ egy 2 – CNF (minden klóz két literálból áll).

A $G(\varphi)$ gráf

A 2SAT problémára viszont adható hatékony algoritmus.
Legyen φ egy 2 – CNF (minden klóz két literálból áll).

A $G(\varphi)$ gráf

- csúcsok: a φ -ben előforduló változók és negáltjaik.

A 2SAT problémára viszont adható hatékony algoritmus.
Legyen φ egy 2 – CNF (minden klóz két literálból áll).

A $G(\varphi)$ gráf

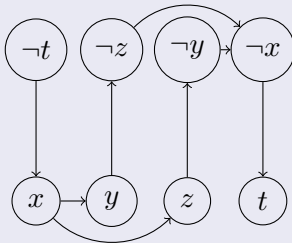
- csúcsok: a φ -ben előforduló változók és negáltjaik.
- élek: (α, β) él $\Leftrightarrow \bar{\alpha} \vee \beta$ vagy $\beta \vee \bar{\alpha}$ a φ tagja (azaz ha $\alpha \rightarrow \beta$ vagy $\bar{\beta} \rightarrow \bar{\alpha}$ a φ tagja.)

Példa

Legyen $\varphi = (\neg x \vee z) \wedge (\neg x \vee y) \wedge (\neg y \vee \neg z) \wedge (x \vee t)$.

Példa

Legyen $\varphi = (\neg x \vee z) \wedge (\neg x \vee y) \wedge (\neg y \vee \neg z) \wedge (x \vee t)$.
 Ekkor $G(\varphi)$:



Tétel

A φ formula akkor és csak akkor kielégíthető, ha nincs olyan x változó, amelyre létezik $G(\varphi)$ -ben irányított út x -ből $\neg x$ -be és vissza.

Tétel

A φ formula akkor és csak akkor kielégíthető, ha nincs olyan x változó, amelyre létezik $G(\varphi)$ -ben irányított út x -ből $\neg x$ -be és vissza.

Ötlet

Tétel

A φ formula akkor és csak akkor kielégíthető, ha nincs olyan x változó, amelyre létezik $G(\varphi)$ -ben irányított út x -ből $\neg x$ -be és vissza.

Ötlet

Minden él egy implikációnak felel meg.

Tétel

A φ formula akkor és csak akkor kielégíthető, ha nincs olyan x változó, amelyre létezik $G(\varphi)$ -ben irányított út x -ből $\neg x$ -be és vissza.

Ötlet

Minden él egy implikációnak felel meg.

Így ha l_1 -ből l_2 elérhető, akkor $\varphi \models (l_1 \rightarrow l_2)$.

Tétel

A φ formula akkor és csak akkor kielégíthető, ha nincs olyan x változó, amelyre létezik $G(\varphi)$ -ben irányított út x -ből $\neg x$ -be és vissza.

Ötlet

Minden él egy implikációnak felel meg.

Így ha l_1 -ből l_2 elérhető, akkor $\varphi \models (l_1 \rightarrow l_2)$.

Ha x -ből is elérhető $\neg x$ és fordítva, akkor $\varphi \models (x \leftrightarrow \neg x)$, így φ kielégíthetetlen.

Tétel

A φ formula akkor és csak akkor kielégíthető, ha nincs olyan x változó, amelyre létezik $G(\varphi)$ -ben irányított út x -ből $\neg x$ -be és vissza.

Ötlet

Minden él egy implikációnak felel meg.

Így ha l_1 -ből l_2 elérhető, akkor $\varphi \models (l_1 \rightarrow l_2)$.

Ha x -ből is elérhető $\neg x$ és fordítva, akkor $\varphi \models (x \leftrightarrow \neg x)$, így φ kielégíthetetlen.

Ha nincs ilyen változó, akkor ciklusban:

Tétel

A φ formula akkor és csak akkor kielégíthető, ha nincs olyan x változó, amelyre létezik $G(\varphi)$ -ben irányított út x -ből $\neg x$ -be és vissza.

Ötlet

Minden él egy implikációnak felel meg.

Így ha l_1 -ből l_2 elérhető, akkor $\varphi \models (l_1 \rightarrow l_2)$.

Ha x -ből is elérhető $\neg x$ és fordítva, akkor $\varphi \models (x \leftrightarrow \neg x)$, így φ kielégíthetetlen.

Ha nincs ilyen változó, akkor ciklusban:

- válasszunk egy olyan l literált, amiből nem érhető el \bar{l} és melynek még nem adtunk értéket;

Tétel

A φ formula akkor és csak akkor kielégíthető, ha nincs olyan x változó, amelyre létezik $G(\varphi)$ -ben irányított út x -ből $\neg x$ -be és vissza.

Ötlet

Minden él egy implikációnak felel meg.

Így ha l_1 -ből l_2 elérhető, akkor $\varphi \models (l_1 \rightarrow l_2)$.

Ha x -ből is elérhető $\neg x$ és fordítva, akkor $\varphi \models (x \leftrightarrow \neg x)$, így φ kielégíthetetlen.

Ha nincs ilyen változó, akkor ciklusban:

- válasszunk egy olyan l literált, amiből nem érhető el \bar{l} és melynek még nem adtunk értéket;
- állítsuk 1-re l -t és minden belőle elérhető literált, komplementereiket pedig nullára;

Tétel

A φ formula akkor és csak akkor kielégíthető, ha nincs olyan x változó, amelyre létezik $G(\varphi)$ -ben irányított út x -ből $\neg x$ -be és vissza.

Ötlet

Minden él egy implikációnak felel meg.

Így ha l_1 -ből l_2 elérhető, akkor $\varphi \models (l_1 \rightarrow l_2)$.

Ha x -ből is elérhető $\neg x$ és fordítva, akkor $\varphi \models (x \leftrightarrow \neg x)$, így φ kielégíthetetlen.

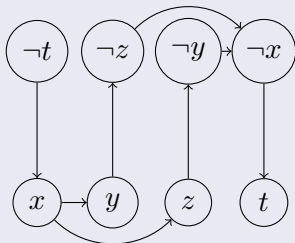
Ha nincs ilyen változó, akkor ciklusban:

- válasszunk egy olyan l literált, amiből nem érhető el \bar{l} és melynek még nem adtunk értéket;
- állítsuk 1-re l -t és minden belőle elérhető literált, komplementereiket pedig nullára;

amíg minden változónak értéket nem adtunk (nem lesz ütközés, ha nincs olyan x , akiből $\neg x$ elérhető és viszont). Az így kapott értékelés kielégíti φ -t.

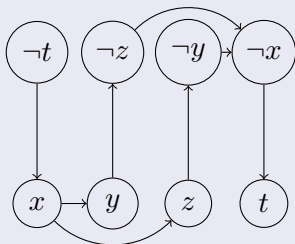
Példa

$$\varphi = (\neg x \vee z) \wedge (\neg x \vee y) \wedge (\neg y \vee \neg z) \wedge (x \vee t).$$



Példa

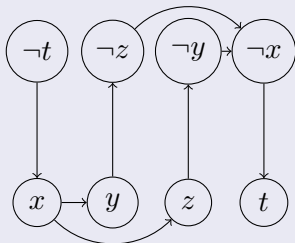
$$\varphi = (\neg x \vee z) \wedge (\neg x \vee y) \wedge (\neg y \vee \neg z) \wedge (x \vee t).$$



Válasszuk először x -et.

Példa

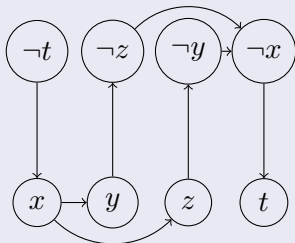
$$\varphi = (\neg x \vee z) \wedge (\neg x \vee y) \wedge (\neg y \vee \neg z) \wedge (x \vee t).$$



Válasszuk először x -et. $G(\varphi)$ -ben x -ből elérhető $\neg x$.

Példa

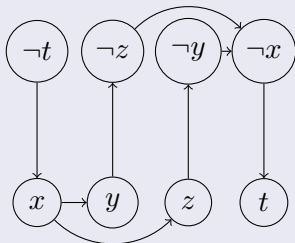
$$\varphi = (\neg x \vee z) \wedge (\neg x \vee y) \wedge (\neg y \vee \neg z) \wedge (x \vee t).$$



Válasszuk először x -et. $G(\varphi)$ -ben x -ből elérhető $\neg x$. igaz-ra állítjuk $\neg x$ -et és t -t, hamis-ra x -et és $\neg t$ -t.

Példa

$$\varphi = (\neg x \vee z) \wedge (\neg x \vee y) \wedge (\neg y \vee \neg z) \wedge (x \vee t).$$



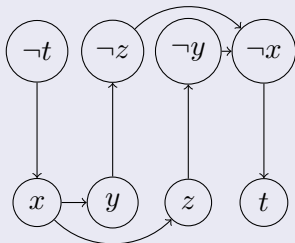
Válasszuk először x -et. $G(\varphi)$ -ben x -ből elérhető $\neg x$.

igaz-ra állítjuk $\neg x$ -et és t -t, hamis-ra x -et és $\neg t$ -t.

Ezek után válasszuk mondjuk y -t. y -ból nem érhető el $\neg y$, tehát $\alpha := y$.

Példa

$$\varphi = (\neg x \vee z) \wedge (\neg x \vee y) \wedge (\neg y \vee \neg z) \wedge (x \vee t).$$



Válasszuk először x -et. $G(\varphi)$ -ben x -ből elérhető $\neg x$.

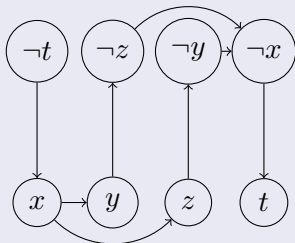
igaz-ra állítjuk $\neg x$ -et és t -t, hamis-ra x -et és $\neg t$ -t.

Ezek után válasszuk mondjuk y -t. y -ból nem érhető el $\neg y$, tehát $\alpha := y$.

igaz-ra állítjuk y -t és $\neg z$ -t, hamis-ra pedig $\neg y$ -t és z -t.

Példa

$$\varphi = (\neg x \vee z) \wedge (\neg x \vee y) \wedge (\neg y \vee \neg z) \wedge (x \vee t).$$



Válasszuk először x -et. $G(\varphi)$ -ben x -ből elérhető $\neg x$.

igaz-ra állítjuk $\neg x$ -et és t -t, hamis-ra x -et és $\neg t$ -t.

Ezek után válasszuk mondjuk y -t. y -ból nem érhető el $\neg y$, tehát $\alpha := y$.

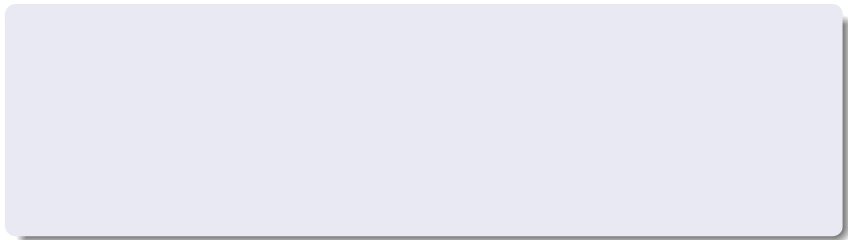
igaz-ra állítjuk y -t és $\neg z$ -t, hamis-ra pedig $\neg y$ -t és z -t.

A kapott értékadás kielégíti φ -t.

Következmény

$2\text{SAT} \in \mathbf{P}$.

Horn-formulák és Horn-átnevezhető formulák



Egy klóz **Horn-klóz**, ha benne maximum egy pozitív literál szerepel.

Horn-formulák és Horn-átnevezhető formulák

Egy klóz **Horn-klóz**, ha benne maximum egy pozitív literál szerepel.
Egy CNF **Horn-formula**, ha benne minden klóz Horn-klóz.

Horn-formulák és Horn-átnevezhető formulák

Egy klóz **Horn-klóz**, ha benne maximum egy pozitív literál szerepel.
Egy CNF **Horn-formula**, ha benne minden klóz Horn-klóz.
Egy CNF **Horn-átnevezhető**, ha bizonyos változói
komplementálásával Horn-formulává tehető

Horn-formulák és Horn-átnevezhető formulák

Egy klóz **Horn-klóz**, ha benne maximum egy pozitív literál szerepel.
Egy CNF **Horn-formula**, ha benne minden klóz Horn-klóz.
Egy CNF **Horn-átnevezhető**, ha bizonyos változói komplementálásával Horn-formulává tehető (vagyis ha van változók egy olyan X halmaza, hogy minden $x \in X$ -re x helyébe $\neg x$ -et, $\neg x$ helyébe x -et írva a formulába, az eredmény Horn-formula lesz).

Horn-formulák és Horn-átnevezhető formulák

Egy klóz **Horn-klóz**, ha benne maximum egy pozitív literál szerepel.
Egy CNF **Horn-formula**, ha benne minden klóz Horn-klóz.
Egy CNF **Horn-átnevezhető**, ha bizonyos változói komplementálásával Horn-formulává tehető (vagyis ha van változók egy olyan X halmaza, hogy minden $x \in X$ -re x helyébe $\neg x$ -et, $\neg x$ helyébe x -et írva a formulába, az eredmény Horn-formula lesz).

Példa

Horn-formulák és Horn-átnevezhető formulák

Egy klóz **Horn-klóz**, ha benne maximum egy pozitív literál szerepel.
Egy CNF **Horn-formula**, ha benne minden klóz Horn-klóz.
Egy CNF **Horn-átnevezhető**, ha bizonyos változói komplementálásával Horn-formulává tehető (vagyis ha van változók egy olyan X halmaza, hogy minden $x \in X$ -re x helyébe $\neg x$ -et, $\neg x$ helyébe x -et írva a formulába, az eredmény Horn-formula lesz).

Példa

- $x \wedge (\neg x \vee y) \wedge (\neg y \vee z) \wedge (\neg y \vee \neg z)$ Horn-formula.

Horn-formulák és Horn-átnevezhető formulák

Egy klóz **Horn-klóz**, ha benne maximum egy pozitív literál szerepel.
Egy CNF **Horn-formula**, ha benne minden klóz Horn-klóz.
Egy CNF **Horn-átnevezhető**, ha bizonyos változói komplementálásával Horn-formulává tehető (vagyis ha van változók egy olyan X halmaza, hogy minden $x \in X$ -re x helyébe $\neg x$ -et, $\neg x$ helyébe x -et írva a formulába, az eredmény Horn-formula lesz).

Példa

- $x \wedge (\neg x \vee y) \wedge (\neg y \vee z) \wedge (\neg y \vee \neg z)$ Horn-formula.
- $(x \vee y) \wedge (\neg x \vee \neg y)$ Horn-átnevezhető.

Horn-formulák és Horn-átnevezhető formulák

Egy klóz **Horn-klóz**, ha benne maximum egy pozitív literál szerepel.
Egy CNF **Horn-formula**, ha benne minden klóz Horn-klóz.
Egy CNF **Horn-átnevezhető**, ha bizonyos változói komplementálásával Horn-formulává tehető (vagyis ha van változók egy olyan X halmaza, hogy minden $x \in X$ -re x helyébe $\neg x$ -et, $\neg x$ helyébe x -et írva a formulába, az eredmény Horn-formula lesz).

Példa

- $x \wedge (\neg x \vee y) \wedge (\neg y \vee z) \wedge (\neg y \vee \neg z)$ Horn-formula.
- $(x \vee y) \wedge (\neg x \vee \neg y)$ Horn-átnevezhető.
- $(\neg x \vee \neg y) \wedge (\neg x \vee y) \wedge (x \vee \neg y) \wedge (x \vee y)$ **nem** Horn-átnevezhető.

Tétel

Horn-átnevezhető formulák kielégíthetősége polinom időben eldönthető.

Horn-formulák és Horn-átnevezhető formulák

Tétel

Horn-átnevezhető formulák kielégíthetősége polinom időben eldönthető.

Ötlet

Az **egységrezolúció** alkalmazása ezekre a formulákra teljes:

Tétel

Horn-átnevezhető formulák kielégíthetősége polinom időben eldönthető.

Ötlet

Az **egységrezolúció** alkalmazása ezekre a formulákra teljes:

- amíg találunk egy egyetlen literálból álló $\{l\}$ klózt:

Tétel

Horn-átnevezhető formulák kielégíthetősége polinom időben eldönthető.

Ötlet

Az **egységrezolúció** alkalmazása ezekre a formulákra teljes:

- amíg találunk egy egyetlen literálból álló $\{l\}$ klózt;
- l értékét 1-re állítjuk;

Tétel

Horn-átnevezhető formulák kielégíthetősége polinom időben eldönthető.

Ötlet

Az **egységrezolúció** alkalmazása ezekre a formulákra teljes:

- amíg találunk egy egyetlen literálból álló $\{l\}$ klózt;
- l értékét 1-re állítjuk;
- töröljük az összes, l -t tartalmazó klózt;

Tétel

Horn-átnevezhető formulák kielégíthetősége polinom időben eldönthető.

Ötlet

Az **egységrezolúció** alkalmazása ezekre a formulákra teljes:

- amíg találunk egy egyetlen literálból álló $\{l\}$ klózt;
- l értékét 1-re állítjuk;
- töröljük az összes, l -t tartalmazó klózt;
- a maradék klózokból töröljük az \bar{l} literált.

Tétel

Horn-átnevezhető formulák kielégíthetősége polinom időben eldönthető.

Ötlet

Az **egységrezolúció** alkalmazása ezekre a formulákra teljes:

- amíg találunk egy egyetlen literálból álló $\{l\}$ klózt;
- l értékét 1-re állítjuk;
- töröljük az összes, l -t tartalmazó klózt;
- a maradék klózokból töröljük az \bar{l} literált.

Ha közben megkapjuk az üres klózt, a formula kielégíthetetlen; ellenkező esetben kielégíthető.

Tétel

Horn-átnevezhető formulák kielégíthetősége polinom időben eldönthető.

Ötlet

Az **egységrezolúció** alkalmazása ezekre a formulákra teljes:

- amíg találunk egy egyetlen literálból álló $\{l\}$ klózt;
- l értékét 1-re állítjuk;
- töröljük az összes, l -t tartalmazó klózt;
- a maradék klózokból töröljük az \bar{l} literált.

Ha közben megkapjuk az üres klózt, a formula kielégíthetetlen; ellenkező esetben kielégíthető.

Az algoritmus (ésszerű reprezentációval) lineáris időben végrehajtható és Horn-átnevezhető formulákra helyes.

Megjegyzés

Az is polinom időben eldönthető egy CNF-ről, hogy Horn-átnevezhető-e.

Horn-formulák és Horn-átnevezhető formulák

Megjegyzés

Az is polinom időben eldönthető egy CNF-ről, hogy Horn-átnevezhető-e.

Megjegyzés

„Keverni” nem tudjuk a két hatékonyan eldönthető speciális esetet:

Megjegyzés

Az is polinom időben eldönthető egy CNF-ről, hogy Horn-átnevezhető-e.

Megjegyzés

„Keverni” nem tudjuk a két hatékonyan eldönthető speciális esetet: NP-teljes a következő probléma: adott egy olyan CNF, melyben minden klóz

Megjegyzés

Az is polinom időben eldönthető egy CNF-ről, hogy Horn-átnevezhető-e.

Megjegyzés

„Keverni” nem tudjuk a két hatékonyan eldönthető speciális esetet: NP-teljes a következő probléma: adott egy olyan CNF, melyben minden klóz

- vagy max. két literálból áll,

Megjegyzés

Az is polinom időben eldönthető egy CNF-ről, hogy Horn-átnevezhető-e.

Megjegyzés

„Keverni” nem tudjuk a két hatékonyan eldönthető speciális esetet: NP-teljes a következő probléma: adott egy olyan CNF, melyben minden klóz

- vagy max. két literálból áll,
- vagy háromelemű negatív klóz,

Megjegyzés

Az is polinom időben eldönthető egy CNF-ről, hogy Horn-átnevezhető-e.

Megjegyzés

„Keverni” nem tudjuk a két hatékonyan eldönthető speciális esetet: NP-teljes a következő probléma: adott egy olyan CNF, melyben minden klóz

- vagy max. két literálból áll,
- vagy háromelemű negatív klóz,

kielégíthető-e?

Megjegyzés

Az is polinom időben eldönthető egy CNF-ről, hogy Horn-átnevezhető-e.

Megjegyzés

„Keverní” nem tudjuk a két hatékonyan eldönthető speciális esetet: NP-teljes a következő probléma: adott egy olyan CNF, melyben minden klóz

- vagy max. két literálból áll,
- vagy háromelemű negatív klóz,

kielégíthető-e?

Ötlet: $3SAT \leq ez$

Változónként $(x \vee \hat{x}) \wedge (\neg x \vee \neg \hat{x})$ és ehhez hozzá minden eredeti klózban a pozitív x literálokat $\neg \hat{x}$ -re cseréljük

Az előfordulásszám csökkentése

Azt is próbálhatjuk korlátozni, hogy egy változó (vagy egy literál) hányszor fordulhat elő legfeljebb a formulában.

Azt is próbálhatjuk korlátozni, hogy egy változó (vagy egy literál) hányszor fordulhat elő legfeljebb a formulában.

Tétel

NP-teljes a következő probléma: adott egy 3CNF, melyben minden változó legfeljebb háromszor szerepel, kielégíthető-e?

Azt is próbálhatjuk korlátozni, hogy egy változó (vagy egy literál) hányszor fordulhat elő legfeljebb a formulában.

Tétel

NP-teljes a következő probléma: adott egy 3CNF, melyben minden változó legfeljebb háromszor szerepel, kielégíthető-e?

P-ben van a következő probléma: adott egy CNF, melyben minden változó legfeljebb kétszer szerepel, kielégíthető-e?

NP-teljes gráfelméleti problémák

FÜGGETLEN CSÚCSHALMAZ

FÜGGETLEN CSÚCSHALMAZ

Adott: $G = (V, E)$ irányítatlan gráf, K szám.

FÜGGETLEN CSÚCSHALMAZ

Adott: $G = (V, E)$ irányítatlan gráf, K szám.

Kérdés: Létezik-e K elemű független csúcshalmaz?

NP-teljes gráfelméleti problémák

FÜGGETLEN CSÚCSHALMAZ

Adott: $G = (V, E)$ irányítatlan gráf, K szám.

Kérdés: Létezik-e K elemű független csúcshalmaz?

Tétel

A FÜGGETLEN CSÚCSHALMAZ probléma **NP**-teljes.

NP-teljes gráfelméleti problémák

FÜGGETLEN CSÚCSHALMAZ

Adott: $G = (V, E)$ irányítatlan gráf, K szám.

Kérdés: Létezik-e K elemű független csúcshalmaz?

Tétel

A FÜGGETLEN CSÚCSHALMAZ probléma **NP**-teljes.

Bizonyítás

Világos, hogy a probléma **NP**-ben van.

NP-teljes gráfelméleti problémák

FÜGGETLEN CSÚCSHALMAZ

Adott: $G = (V, E)$ irányítatlan gráf, K szám.

Kérdés: Létezik-e K elemű független csúcshalmaz?

Tétel

A FÜGGETLEN CSÚCSHALMAZ probléma **NP**-teljes.

Bizonyítás

Világos, hogy a probléma **NP**-ben van. Megmutatjuk, hogy 3SAT visszavezethető a FÜGGETLEN CSÚCSHALMAZ-ra.

NP-teljes gráfelméleti problémák

FÜGGETLEN CSÚCSHALMAZ

Adott: $G = (V, E)$ irányítatlan gráf, K szám.

Kérdés: Létezik-e K elemű független csúcshalmaz?

Tétel

A FÜGGETLEN CSÚCSHALMAZ probléma **NP**-teljes.

Bizonyítás

Világos, hogy a probléma **NP**-ben van. Megmutatjuk, hogy 3SAT visszavezethető a FÜGGETLEN CSÚCSHALMAZ-ra.

Legyen φ a 3SAT egy példánya, $\varphi = c_1 \wedge \dots \wedge c_m$.

NP-teljes gráfelméleti problémák

FÜGGETLEN CSÚCSHALMAZ

Adott: $G = (V, E)$ irányítatlan gráf, K szám.

Kérdés: Létezik-e K elemű független csúcshalmaz?

Tétel

A FÜGGETLEN CSÚCSHALMAZ probléma NP-teljes.

Bizonyítás

Világos, hogy a probléma NP-ben van. Megmutatjuk, hogy 3SAT visszavezethető a FÜGGETLEN CSÚCSHALMAZ-ra.

Legyen φ a 3SAT egy példánya, $\varphi = c_1 \wedge \dots \wedge c_m$. A G gráf álljon m darab **háromszögből**, melyek a c_i tagoknak felelnek meg, s melyek csúcsai a c_i -kben lévő literáloknak felelnek meg. Ezen kívül még kössünk össze éllel **minden ellentétes literált**. Végül legyen $K = m$.

NP-teljes gráfelméleti problémák

FÜGGETLEN CSÚCSHALMAZ

Adott: $G = (V, E)$ irányítatlan gráf, K szám.

Kérdés: Létezik-e K elemű független csúcshalmaz?

Tétel

A FÜGGETLEN CSÚCSHALMAZ probléma NP-teljes.

Bizonyítás

Világos, hogy a probléma NP-ben van. Megmutatjuk, hogy 3SAT visszavezethető a FÜGGETLEN CSÚCSHALMAZ-ra.

Legyen φ a 3SAT egy példánya, $\varphi = c_1 \wedge \dots \wedge c_m$. A G gráf álljon m darab **háromszögből**, melyek a c_i tagoknak felelnek meg, s melyek csúcsai a c_i -kben lévő literáloknak felelnek meg. Ezen kívül még kössünk össze éllel **minden ellentétes literált**. Végül legyen $K = m$.

φ kielégíthető $\Leftrightarrow G$ -ben létezik K elemű független csúcshalmaz.

Példa

Legyen

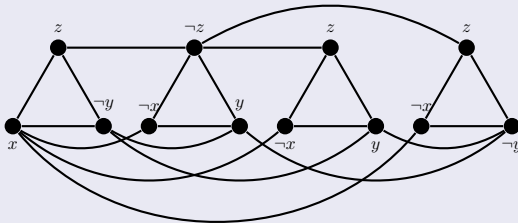
$$\varphi = (x \vee \neg y \vee z) \wedge (\neg x \vee y \vee \neg z) \wedge (\neg x \vee y \vee z) \wedge (\neg x \vee \neg y \vee z).$$

Példa

Legyen

$$\varphi = (x \vee \neg y \vee z) \wedge (\neg x \vee y \vee \neg z) \wedge (\neg x \vee y \vee z) \wedge (\neg x \vee \neg y \vee z).$$

Ekkor a G gráf:



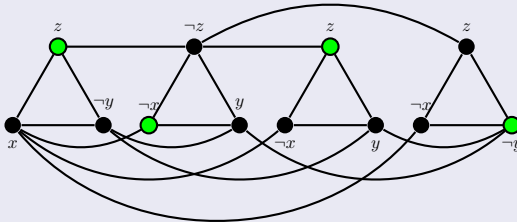
és $K = 4$.

Példa

Legyen

$$\varphi = (x \vee \neg y \vee z) \wedge (\neg x \vee y \vee \neg z) \wedge (\neg x \vee y \vee z) \wedge (\neg x \vee \neg y \vee z).$$

Ekkor a G gráf:



és $K = 4$.

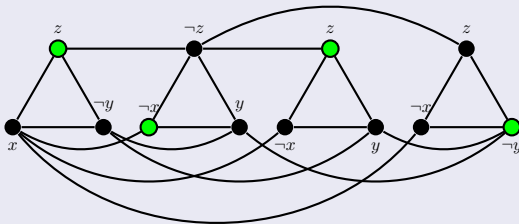
A zölddel jelölt csúcsok egy négyelemű független csúcshalmazt alkotnak;

Példa

Legyen

$$\varphi = (x \vee \neg y \vee z) \wedge (\neg x \vee y \vee \neg z) \wedge (\neg x \vee y \vee z) \wedge (\neg x \vee \neg y \vee z).$$

Ekkor a G gráf:



és $K = 4$.

A zölddel jelölt csúcsok egy négyelemű független csúcshalmazt alkotnak; egy kielégítő kiértékelés pedig: $x = y = 0, z = 1$.

KLIKK

Adott: $G = (V, E)$ irányítatlan gráf, K szám.

Kérdés: Létezik-e K elemű klikk (teljes részgráf)?

KLIKK

Adott: $G = (V, E)$ irányítatlan gráf, K szám.

Kérdés: Létezik-e K elemű klikk (teljes részgráf)?

Tétel

KLIKK **NP**-teljes.

KLIKK

Adott: $G = (V, E)$ irányítatlan gráf, K szám.

Kérdés: Létezik-e K elemű klikk (teljes részgráf)?

Tétel

KLIKK **NP**-teljes.

Bizonyítás (visszavezetéssel): gyakorlaton.

CSÚCSLEFEDÉS

Adott: $G = (V, E)$ irányítatlan gráf, K szám.

Kérdés: Létezik-e olyan K elemű csúcshalmaz, hogy minden él illeszkedik a halmaz egy csúcsához?

CSÚCSLEFEDÉS

Adott: $G = (V, E)$ irányítatlan gráf, K szám.

Kérdés: Létezik-e olyan K elemű csúcshalmaz, hogy minden él illeszkedik a halmaz egy csúcsához?

Tétel

A CSÚCSLEFEDÉS probléma **NP**-teljes.

CSÚCSLEFEDÉS

Adott: $G = (V, E)$ irányítatlan gráf, K szám.

Kérdés: Létezik-e olyan K elemű csúcshalmaz, hogy minden él illeszkedik a halmaz egy csúcsához?

Tétel

A CSÚCSLEFEDÉS probléma **NP**-teljes.

Bizonyítás (visszavezetéssel): gyakorlaton.

Tétel

A HAMILTON-ÚT probléma NP-teljes.

Tétel

A HAMILTON-ÚT probléma NP-teljes.

Ötlet

„Értékválasztó” gadget:

Tétel

A HAMILTON-ÚT probléma NP-teljes.

Ötlet

„Értékválasztó” gadget:



Tétel

A HAMILTON-ÚT probléma NP-teljes.

Ötlet

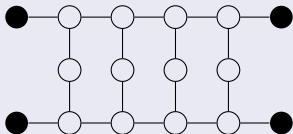
„Értékválasztó” gadget:



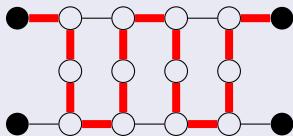
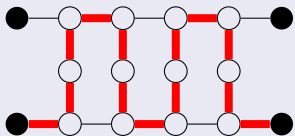
(Balról jobbra ha megy egy Hamilton-út, akkor **választanunk** kell a „fenti” és a „lenti” útvonal közül egyet.)

⇒ mint egy értékadás – ilyen gadgetből változónként lesz egy

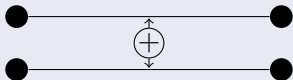
„XOR” gadget:



Minden Hamilton-út egy olyan gráfban, ami ezt a gadgetot **fesztett részgráfként** tartalmazza, a következők egyike ezen a gráfon belül:



Hogy átláthatóbb legyen a rajzunk, ezt a gadgetet így rajzoljuk:



A XOR gadgettel egy gráfban olyan Hamilton-kört keresünk, melyben **ki tudjuk kötni bizonyos élpárokra**, hogy a két él közül a keresett Hamilton-kör **pontosan egyen** haladjon át.

A teljes konstrukció a $3SAT \leq HAMILTON-ÚT$ visszavezetéséhez:

- minden változóhoz létrehozunk egy **értékválasztó gadgetet**, az x_i -hez tartozó két párhuzamos élt x_i -vel ill. $\neg x_i$ -vel címkézzük;
- minden klózhoz létrehozunk egy **háromszöget**, az $\ell_1 \vee \ell_2 \vee \ell_3$ klózhoz létrehozott háromszög éleit rendre ℓ_1 -gyel, ℓ_2 -vel és ℓ_3 -mal címkézzük;
- az ellentétes literálokkal címkézett élek között XOR gadgetet hozunk létre;
- az értékválasztó gadgeteket láncba kötjük;
- a háromszögek csúcsaiból, az utolsó értékválasztó csúcsból és még egy plusz csúcsból pedig egyetlen nagy klikket készítünk;
- végül az előző pontbeli plusz csúcsból még egy új csúcsba

Összefoglalás

- Definiáltuk a k SAT problémákat, ahol $k \geq 1$ egész.
- Láttuk, hogy $SAT \leq 3SAT$, így $3SAT$ is **NP**-teljes.
- Megadtunk egy polinomidejű algoritmust $2SAT$ eldöntésére.
- Definiáltuk a Horn-átnevezhető formulákat és megadtunk egy polinomidejű algoritmust, ami eldönti, hogy egy input Horn-átnevezhető formula kielégíthető-e.
- A $3SAT$ -ot visszavezettük a következő problémára: input egy CNF, melynek minden klóza vagy három negatív literált tartalmaz, vagy kételemű, kielégíthető-e? Így ez (a két hatékonyan eldönthető speciális eset „keveréke”) is **NP**-teljes.
- Definiáltuk a FÜGGETLEN CSÚCSHALMAZ, KLIKK és CSÚCSLEFEDÉS problémákat.
- Láttuk, hogy $3SAT \leq$ FÜGGETLEN CSÚCSHALMAZ, így ez utóbbi is **NP**-teljes.
- Gyakorlaton láttuk, hogy FÜGGETLEN CSÚCSHALMAZ \leq KLIKK és hogy FÜGGETLEN CSÚCSHALMAZ \leq CSÚCSLEFEDÉS, így ezek is **NP**-teljesek.
- Vázlatosan megnéztük, hogy $3SAT \leq$ HAMILTON-ÚT, így ez utóbbi is **NP**-teljes.