

# A megállási probléma

$$L_h \in RE$$

## Bizonyítás

Korábbi tétel alapján elég megmutatni, hogy  $L_h \leq L_u$

- Tetszőleges  $M$  Turing-gépre, legyen  $M'$  az alábbi Turing-gép:
  - $M'$  tetszőleges  $u$  bemeneten a következőket teszi:
    1. Futtatja  $M$ -et  $u$ -n
    2. Ha  $M$   $q_i$ -be vagy  $q_n$ -be lép, akkor  $M'$   $q_i$ -be lép
- Belátható, hogy
  1. Az  $f: \langle M \rangle \rightarrow \langle M' \rangle$  leképezés egy kiszámítható függvény
  2. Választartó is: tetszőleges  $M, w$  Turing-gép — bemenet párosra,  
 $\langle M, w \rangle \in L_h \Leftrightarrow \text{Az } M \text{ megáll } w\text{-n} \Leftrightarrow \text{A } M' \text{ elfogadja } w\text{-t} \Leftrightarrow \langle M', w \rangle \in L_u$
- Tehát  $M'$  konstrukciója az  $L_h$  visszavezetése  $L_u$ -ra
- Következik, hogy  $L_h \in RE$

# Rice tétele

Eldönthető egyáltalán **bármilyen kérdés** a Turing-gépek által felismert nyelvekkel kapcsolatban?

Egy ilyen kérdés **triviális** ha minden Turing-gép által felismerhető nyelvre *igen* a válasz vagy mindre *nem* a válasz

Valójában mindössze **két triviális kérdést** tudunk megfogalmazni egy tetszőleges  $M$  Turing-gép által felismert nyelvvel kapcsolatban:

- Vajon  $L(M)$   $RE$ -beli-e el? (triviális, hogy a válasz erre *igen*) és az, hogy
- Vajon  $L(M)$   $RE$ -n kívüli-e? (triviális, hogy erre a válasz *nem*)

Legyen  $M$  egy Turing-gép

Nem triviális kérdések például: vajon az  $M$  által felismert nyelv

- üres-e
- véges-e
- reguláris-e

**Rice tétele:** A Turing-gépek által felismert nyelvekkel kapcsolatos összes nem triviális kérdés eldönthetetlen

# Post Megfelelkezési Probléma

Legyen  $u_1, \dots, u_n, v_1, \dots, v_n \in \Sigma^+ (n \geq 1)$

A  $D = \left\{ \frac{u_1}{v_1}, \dots, \frac{u_n}{v_n} \right\}$  halmazt **dominókészletnek** nevezzük

Az  $\frac{u_{i_1}}{v_{i_1}} \dots \frac{u_{i_m}}{v_{i_m}}$  ( $m \geq 1, 1 \leq i_1, \dots, i_m \leq n$ ) dominósorozat a  $D$  **egy megoldása**, ha

$$u_{i_1} \dots u_{i_m} = v_{i_1} \dots v_{i_m}$$

A **PMP probléma**

- Adott egy  $D$  dominókészlet
- Van-e  $D$ -nek megoldása?

A  $\left\{ \frac{b}{ca}, \frac{a}{ab}, \frac{ca}{a}, \frac{abc}{c} \right\}$  egy megoldása:

$$\frac{a}{ab} \frac{b}{ca} \frac{ca}{a} \frac{a}{ab} \frac{abc}{c}$$

**PMP Turing-felismerhető, de nem eldönthető**

- Félig eldönthető, mert a potenciális megoldások felsorolhatók és tesztelhetők
- Nem eldönthető, mert vissza lehet rá vezetni az  $L_u$  problémát

# CF nyelvekkel kapcsolatos eldönthetetlen problémák

*ECF* probléma: Adott egy  $G$  CF nyelvtan, döntsük el, hogy  $G$  egyértelmű-e

Az *ECF* probléma eldönthetetlen

Bizonyítás

Visszavezetjük a *PMP* problémát (ami eldönthetetlen) az *ECF* komplementerére

Ebből már következik, hogy *ECF* is eldönthetetlen

- Legyen  $D = \left\{ \frac{u_1}{v_1}, \dots, \frac{u_n}{v_n} \right\}$  egy dominókészlet
- Konstruáljuk meg a következő CF nyelvtant  $G_D := (\{S, A, B\}, \Sigma \cup \Delta, P \cup \{S \rightarrow A, S \rightarrow B\}, S)$ , ahol
  - $\Delta = \{a_1, \dots, a_n\}, \Sigma \cap \Delta = \emptyset$
  - $P = \{A \rightarrow u_1 A a_1, \dots, A \rightarrow u_n A a_n, A \rightarrow \varepsilon\} \cup \{B \rightarrow v_1 B a_1, \dots, B \rightarrow v_n B a_n, B \rightarrow \varepsilon\}$

# CF nyelvekkel kapcsolatos eldönthetetlen problémák

## Bizonyítás (folytatás)

Megmutatjuk, hogy az  $f: \langle D \rangle \rightarrow \langle G_D \rangle$  konstrukció tényleg visszavezetés

- Ha  $\frac{u_{i_1}}{v_{i_1}} \dots \frac{u_{i_m}}{v_{i_m}}$  a  $D$  egy megoldása
  - Akkor  $u_{i_1} \dots u_{i_m} = v_{i_1} \dots v_{i_m}$
  - A  $G_D$  konstrukciója miatt érvényesek az alábbi levezetések:  $S \Rightarrow A \Rightarrow^* u_{i_1} \dots u_{i_m} a_{i_m} \dots a_{i_1}$  és  $S \Rightarrow B \Rightarrow^* v_{i_1} \dots v_{i_m} a_{i_m} \dots a_{i_1}$
  - Azaz ugyanannak a szónak van két különböző baloldali levezetése, azaz  $G_D$  nem egyértelmű
- Most tegyük fel azt, hogy  $G_D$  nem egyértelmű
  - Akkor van olyan  $w \in L(G_D)$  szó, aminek van két különböző baloldali levezetése
  - Az egyik levezetés  $S \rightarrow A$ -val a másik pedig  $S \rightarrow B$ -vel kell kezdődjön
  - Viszont  $w$  alakja csak a következő lehet:  $w = xy$ , ahol  $x \in \Sigma^*$  és  $y \in \{a_1, \dots, a_n\}^*$
  - Ezért a fenti két levezetésben a szabályok ugyanolyan sorrendben kell alkalmazásra kerüljenek
  - Következik, hogy  $D$ -nek van megoldása
- Tehát a megadott konstrukció tényleg a  $PMP$  visszavezetése az  $ECF$  komplementerére

# Bonyolultságelmélet – P és NP

**Cél:** A megoldható (eldönthető) problémák **osztályozása** a megoldáshoz szükséges erőforrások (idő, tár) **menyisége** szerint

**P**-vel jelöljük a

- **polinom időben**
- **determinisztikus** Turing-géppel eldönthető problémák osztályát

**NP**-vel jelöljük a

- **polinom időben**
- **nemdeterminisztikus** Turing-géppel eldönthető problémák osztályát

Világos, hogy  $P \subseteq NP$

Az a sejtés, hogy  $P \subsetneq NP$

- Azaz van olyan probléma ami nemdeterminisztikusan megoldható polinom időben, de determinisztikusan nem

# Bonyolultságelmélet – P és NP

P tartalmazza a gyakorlatban is **hatékonyan megoldható** problémákat, de milyen problémák vannak NP-ben?

Minden NP-beli  $L$  problémára a következő jellemző: létezik egy olyan **polinom idejű  $T$  nemdeterminisztikus** Turing-gép ami következőt tudja

- $T$  az  $L$  minden  $I$  bemenetére **nemdeterminisztikusan generálja**  $I$  egy lehetséges  $M$  megoldását
  - Fontos, hogy  $I$  minden lehetséges megoldása generálva legyen
  - **Determinisztikus számítással leellenőrzi**, hogy  $M$  valóban megoldása-e  $I$ -nek

**Formálisan:** Azt mondjuk, hogy egy  $L$  nyelv **polinom időben verifikálható**, ha van olyan  $K \in P$  nyelv és  $k$  szám, hogy

$$L = \{x \mid \exists y (x, y) \in K \text{ és } |y| = O(|x|^k)\}$$

Bemenet

Megoldás

Polinom időben  
verifikálható

$y$   $x$ -hez képest  
"nem túl nagy"

$K$  polinom időben eldönthető  
determinisztikusan

Egy nyelv akkor és csak akkor NP-beli, ha polinom időben verifikálható

# Polinom időben verifikálhatóság - példák

## SAT polinom időben verifikálható (tehát NP-beli)

- mert egy  $T$  Turing-gép egy tetszőleges  $\varphi$  konjunktív normálformára
  - **Nemdeterminisztikusan** generál minden egyes potenciális  $I$  tanút ( $I$  nem más, mint  $\varphi$  változóinak egy értékadása)
  - **Polinom időben ellenőrzi**, hogy  $I$  kielégíti-e  $\varphi$ -t
  - Ha igen, akkor elfogadja a bemenetet, ha nem, akkor elutasítja
- Ekkor  $T$  számítási fáájában pontosan akkor lesz az egyik levél egy elfogadó konfiguráció, ha  $\varphi$ -nek van egy kielégítő értékadása, azaz pontosan akkor ha  $\varphi$  kielégíthető
- Tehát  $T$  pontosan akkor fogadja el  $\varphi$ -t ha az kielégíthető

## A HAMILTON-ÚT is **polinom időben verifikálható**

- mert egy  $T$  Turing-gép egy tetszőleges  $G$   $n$  csúcsú irányított gráfra
  - **Nemdeterminisztikusan** generál minden lehetséges  $n$  csúcsból álló csúcssorozatot
  - **Polinom időben ellenőrzi**, hogy ezek Hamilton utat alkotnak-e  $s$ -ből  $t$ -be

**HAMILTON ÚT (HÚ):** Adott egy  $G = (V, E)$  irányított gráf és  $s, t \in V$

**Kérdés:** Van-e  $G$ -ben  $s$ -ből  $t$ -be Hamilton út (azaz minden csúcst pontosan egyszer érintő út)?



# Polinomidejű visszavezetések

Milyen visszavezetésre van szükségünk, ha **NP-beli problémák bonyolultságát** szeretnénk összehasonlítani?

- **Az eddig használt nem jó**, mert attól, hogy  $L_1 \leq L_2$  és  $L_2$  könnyen megoldható még nem következik, hogy  $L_1$  is könnyen megoldható (hiszen a visszavezetés lehet nehezen, mondjuk exponenciális időben kiszámítható)
- Olyan visszavezetés kell, ami feltehetőleg **nem elég erős**, hogy minden NP-beli problémát megoldjon
- Mivel valószínűleg  $P \neq NP$ , a **polinom időben kiszámítható** visszavezetések megfelelnek a céljainknak

Egy  $L_1$  nyelv **polinom időben visszavezethető** az  $L_2$  nyelvre (jele:  $L_1 \leq_p L_2$ ), ha

- $L_1 \leq L_2$  és
- a visszavezetéshez használt függvény kiszámítható egy **polinom időkorlátos determinisztikus Turing-géppel**

# Polinomidejű visszavezetések – Példa

2SZÍN: A bemenet egy  $G = (V, E)$  irányítatlan gráf, és azt kell eldönteni, hogy **színezhetők-e  $G$  csúcsai 2 színnel** (pl. pirossal és kékkel) úgy, hogy a szomszédos csúcsok különböző színűek

2SAT: A bemenet egy olyan ítéletkalkulusbeli  $\varphi$  KNF, ahol minden tag pontosan két literált tartalmaz, és azt kell eldönteni, hogy **kelégíthető-e  $\varphi$**

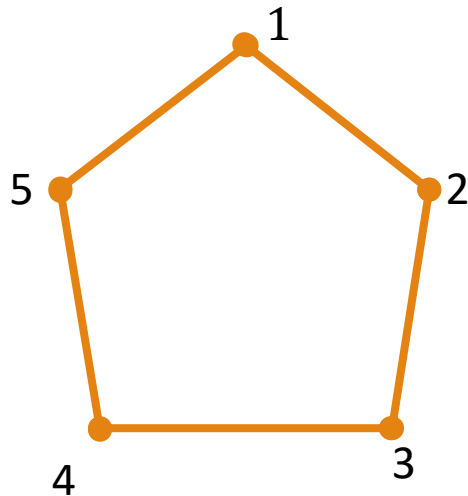
2SZÍN polinom idejű visszavezetése 2SAT-ra:

Tetszőleges  $G = (V, E)$  gráfhoz konstruáljunk meg egy  $\varphi$  formulát a következőképpen

- Minden  $u \in V$  csúcshoz rendeljünk hozzá egy  $x_u$  ítéletváltozót, ami mondjuk annak az állításnak felel meg, hogy „az  $u$  csúcs színe piros”
- Akkor  $\neg x_u$  felel meg annak, hogy „az  $u$  csúcs színe kék”
  - Minden  $(u, v)$  élhez írjuk fel, hogy nem lehetnek egyforma színűek:  $(x_u \leftrightarrow \neg x_v) \equiv (\neg x_u \vee \neg x_v) \wedge (x_v \vee x_u)$
- $\varphi$  **polinomidőben** megkonstruálható
- És  $\varphi$  **pontosan akkor kielégíthető ha  $G$  színezhető két színnel**

# Polinomidejű visszavezetések – Példa

Konstruáljuk meg  $\varphi$ -t a következő gráfhoz:



$$\begin{aligned} &(x_1 \vee x_2) \wedge (\neg x_1 \vee \neg x_2) \wedge \\ &(x_2 \vee x_3) \wedge (\neg x_2 \vee \neg x_3) \wedge \\ &(x_3 \vee x_4) \wedge (\neg x_3 \vee \neg x_4) \wedge \\ &(x_4 \vee x_5) \wedge (\neg x_4 \vee \neg x_5) \wedge \\ &(x_5 \vee x_1) \wedge (\neg x_5 \vee \neg x_1) \end{aligned}$$

Ez **kielégíthetetlen**:

- Ha  $x_1$  igaz,  $x_2$  hamis kell legyen, de akkor  $x_3$  igaz, amiből  $x_4$  hamis, ebből  $x_5$  igaz adódik, és így az utolsó klóz hamis
- Ha  $x_1$  hamis, akkor  $x_2$  igaz, ...,  $x_5$  hamis, és így az utolsó előtti klóz lesz hamis

Tehát nincs a gráfban megfelelő 2-színezés

# Polinomidejű visszavezetések

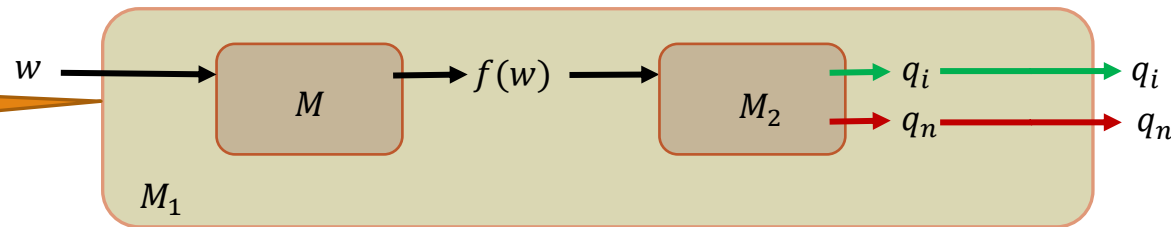
Legyen  $L_1$  és  $L_2$  két nyelv úgy hogy  $L_1 \leq_p L_2$ . Ha  $L_2 \in P$ , akkor  $L_1 \in P$ . Továbbá ha  $L_2 \in NP$ , akkor  $L_1 \in NP$ .

## Bizonyítás

Tegyük fel, hogy  $L_2 \in NP$  és legyen  $M_2$  az  $L_2$ -t eldöntő nemdeterminisztikus,  $M$  pedig az  $f$  visszavezetést kiszámító (determinisztikus) Turing-gép

- Konstruáljuk meg  $M_1$ -et:

Ugyanaz a konstrukció, mint amit a felismerhető nyelveknél láttunk az (általános) visszavezetések kapcsán!



- Világos, hogy  $M_1$  egy  $L_1$ -et eldöntő nemdeterminisztikus Turing-gép
- $M_1$  polinom idejű: legyen  $M$  időigénye  $p_1$ ,  $M_2$ -é pedig  $p_2$  ( $p_1$  és  $p_2$  polinomok)
  - ha  $w$   $n$  hosszú, akkor  $f(w)$  legfeljebb  $p_1(n)$  hosszú lehet
  - ezért az  $M_1$  időigénye  $O(p_1(n) + p_2(p_1(n)))$ , ami szintén polinom
- Tehát  $L_1 \in NP$
- Továbbá, ha  $L_2 \in P$ , azaz  $M_2$  választható determinisztikusnak, akkor  $M_1$  is az
- Kapjuk, hogy  $L_1 \in P$

Láttuk, hogy  $2SZÍN \leq_p 2SAT$   
Logikáról tudjuk, hogy  $2SAT \in P$   
Következik, hogy  $2SZÍN \in P$

# NP-teljesség

Legyen  $\mathbb{C}$  egy problémaosztály

Egy  $L$  probléma  $\mathbb{C}$  -nehéz (a polinom idejű visszavezetésre nézve), ha minden  $L' \in \mathbb{C}$  esetén  $L' \leq_p L$

Egy  $\mathbb{C}$  -nehéz  $L$  probléma  $\mathbb{C}$  -teljes, ha  $L \in \mathbb{C}$

Az NP-teljes problémák a legnehezebben megoldhatók az NP osztályon belül (az előző fólián bizonyított állítás alapján az nem lehet, hogy egy NP-teljest megoldok „könnyen”, de van olyan NP-beli, amit nem tudok megoldani „könnyen”)

Legyen  $L$  egy NP-teljes probléma. Ha  $L \in P$ , akkor  $P = NP$

Bizonyítás

Mivel  $P \subseteq NP$ , elég megmutatni, hogy  $NP \subseteq P$

Legyen  $L' \in NP$  egy tetszőleges probléma; ekkor  $L' \leq_p L$  (mert  $L$  NP-teljes)

De akkor  $L' \in P$  (mert  $P$  zárt a polinomidejű visszavezetésre nézve)

# Az első NP-teljes probléma: a SAT

**Cook tétele:** SAT NP-teljes (NP-beliséget láttuk, a teljességet nem bizonyítjuk)

A polinom idejű visszavezetések tranzitívak:

- Korábban láttuk, hogy ha  $p_1(n)$  és  $p_2(n)$  polinomok, akkor  $p_1(n) + p_2(p_1(n))$  szintén polinom

Következmény: ha  $L$  NP-teljes,  $L' \in \text{NP}$  és  $L \leq_p L'$ , akkor  $L'$  is NP-teljes

Bizonyítás: Mert tetszőleges  $L''$  problémára  $L'' \leq_p L$  (hiszen  $L$  NP-teljes)

Tehát  $L'' \leq_p L \leq_p L'$ , azaz  $L'' \leq_p L'$  (mert a polinom idejű visszavezetések tranzitívak)

Ezért  $L'$  is NP-teljes (mert NP-beli, és minden NP-beli visszavezethető rá polinom időben)

# 3SAT NP-teljes

---

## 3SAT:

- Adott egy  $\varphi$  zérusrendű KNF, melynek minden tagjában pontosan 3 literál szerepel
- **Kérdés:** kielégíthető-e  $\varphi$ ?

## 3SAT NP-teljes

### Bizonyítás

- Világos, hogy  $3SAT \in NP$
- Megmutatjuk, hogy  $SAT \leq_p 3SAT$

# 3SAT NP-teljes

## Bizonyítás

- Tetszőleges  $\varphi$  KNF-ben lévő formulához konstruáljuk meg  $\varphi'$ -t a következő táblázat alapján, ahol  $p, p_1, \dots, p_{n-2}$  mindig új, korábban nem használt ítéletváltozók:

$\varphi$ egy $C$ tagja	$\varphi'$ klózai
$l$	$(l \vee l \vee l)$
$l_1 \vee l_2$	$(l_1 \vee l_2 \vee l_2)$
$l_1 \vee l_2 \vee l_3$	$l_1 \vee l_2 \vee l_3$
$l_1 \vee l_2 \vee l_3 \vee l_4$	$(l_1 \vee l_2 \vee p) \wedge (\neg p \vee l_3 \vee l_4)$
$l_1 \vee \dots \vee l_n \ (n \geq 5)$	$(l_1 \vee l_2 \vee p_1) \wedge (\neg p_1 \vee l_3 \vee p_2) \wedge \dots \wedge (\neg p_{n-2} \vee l_{n-1} \vee l_n)$

- Belátható, hogy  $\varphi$  kielégíthető  $\Leftrightarrow \varphi'$  kielégíthető
- Tegyük fel, hogy egy  $A$  változóértékadásra  $A \models \varphi$  és legyen  $C$  a  $\varphi$  egy klóza



# 3SAT NP-teljes

## Bizonyítás (folyt.)

Ha a  $C$  klóz  $l$ ,  $l_1 \vee l_2$  vagy  $l_1 \vee l_2 \vee l_3$  alakú, akkor  $A$  triviálisan kielégíti  $\varphi'$  megfelelő klózeit

Ha  $C = l_1 \vee l_2 \vee l_3 \vee l_4$ , akkor

- $A \models (l_1 \vee l_2)$  vagy  $A \models (l_3 \vee l_4)$
- Terjesszük ki  $A$ -t a  $p$  változóra a következőképpen
- Legyen  $A(p) = 0$  ha  $A \models (l_1 \vee l_2)$ , és legyen  $A(p) = 1$  ha  $A \models (l_3 \vee l_4)$  (ha mindkét eset fennáll, akkor  $A(p)$  értéke tetszőleges)
- Ekkor  $A$  kielégíti  $(l_1 \vee l_2 \vee p)$ -t és  $(\neg p \vee l_3 \vee l_4)$ -t is, azaz kielégíti a  $C$ -hez megkonstruált  $(l_1 \vee l_2 \vee p) \wedge (\neg p \vee l_3 \vee l_4)$  formulát

Ha  $C = l_1 \vee \dots \vee l_n$  ( $n \geq 5$ ), akkor a fentihez hasonlóan kiegészíthetjük  $A$ -t úgy, hogy kielégítse  $\varphi'$  megfelelő klózát

A fentiek alapján a **másik irány** könnyen látható: Ha egy  $A$  értékadásra  $A \models \varphi'$ , akkor  $A \models \varphi$