MI-SPOL-6

Význam tříd NP a NPH pro praktické výpočty.

Kombinatorická matematika: konečné a diskrétní problémy, konečný počet proměnných a jejich hodnot

Problém (abstraktní) X instance (konkrétní)

Problém:

- vstupní proměnné
- výstupní proměnné (občas shoda s konfiguračními)
- konfigurační proměnné (to, co nastavuje hrubá sílá -- to, co vyjádří všechny kombinace)
- omezení (věci se musí vejít do baohu)
- optimalizační kritérium (hledá se nejvyší cena)

Typy problémů:
I instance, Y konfigurace, $R(I,Y)$ -- Y je řešením

- **rozhodovací**: Existuje Y takové, že $R(I,Y)$? Platí pro všechna Y, že $R(I,Y)$?
- **konstruktivní:** Sestrojit nějaké Y takové, že $R(I,Y)$
- **enumerační:** Sestrojit všechna Y taková, že $R(I,Y)$

Optimalizační problémy:
I instance, Y konfigurace, $R(I,Y)$ -- Y je řešením, $C(Y)$ optimalizační kritérium

- **optimalizační rozhodovací**: Existuje Y takové, že $R(I,Y)$ a $C(Y)$ je aspoň tak dobré jako daná konstanta Q?
- **optimalizační konstruktivní:** Sestrojit nějaké Y takové, že $R(I,Y)$ a $C(Y)$ je nejlepší možné
- **optimalizační enumerační:** Sestrojit všechna Y taková, že $R(I,Y)$ a $C(Y)$ je nejlepší možná
- **optimalizační evaluační:** Zjistit nejlepší možné $C(Y)$ takové, že $R(I,Y)$

Čas výpočtu: počet kroků jednotného výpočetního modelu

Turingův stroj: neomezená páska s políčky (každé jeden symbol abecedy), čtecí a zapisovací hlava, končné stavové zařízení

- program:
  - množina Γ symbolů pásky
  - $\Sigma \subset \Gamma$ množina vstupních symbolů
  - množina stavů Q, počáteční stav $q_0 \in Q$, koncové stavy $q_{ano}, q_{ne} \in Q$
  - přechodová funkce $\delta : (Q - \{q_{ano}, q_{ne}\}) \times \Gamma \rightarrow Q \times \Gamma \times \{-1, +1\}$
- inicializace: stav q_0, políčko 1
- konec: q_{ano}, q_{ne}
- výpočet: $(q \in Q, s \in \Gamma) \rightarrow (q', s', \Delta)$
  (stav, čtený symbol) --> (nový stav, zapsaný symbol, pohyb pásky)

Řešení problému deterministickým TS:

- program M pro DTS řeší **rozhodovací problém** Π, jestliže se výpočet zastaví po konečném počtu kroků pro každou instanci problému Π
- program M pro DTS řeší **rozhodovací problém** Π **v čase** t, jestliže se výpočet zastaví po t krocích pro každou instanci problému Π
- program M pro DTS řeší **rozhodovací problém** Π **s pamětí** m, jestliže počet použitých políček je nejvýše m pro každou instanci problému Π

Třída P

Rozhodovací **problém patří do třídy P**, jestliže pro něj existuje program pro deterministrický TS, který jej řeší v čase $O(n^k)$, kde n je velikost instance a k konečné číslo.

Třída **PSPACE:** DTS, který problém řeší v paměti $O(n^k)$
Třída **EXPTIME:** DTS, který problém řeší v čase $O(2^{P(n)})$, kde $P(n)$ je polynom ve velikosti instance n

Nedeterministický Turingův stroj:
Místo přechodové funkce **přechodová relace**: $\delta \subset (Q - \{q_{ano}, q_{ne}\}) \times \Gamma \times Q \times \Gamma \times \{-1, +1\}$
Výpočet: $(q \in Q, s \in \Gamma) \rightarrow \{(q', s', \Delta)\}$ (přechází se do množiny stavů)

Řešení problému nedeterministickým TS:

- Π_{ano} množina instancí problému Π, které mají výstup ano
- program M pro NTS řeší **rozhodovací problém Π v čase** t, jestliže se výpočet zastaví po t krocích pro každou instanci $I \in \Pi_{ano}$ problému Π

Pokud NTS řeší problém v čase $T(n)$, DTS ho řeší v čase $2^{O(T(n))}$ -- **exponential gap**

Třída NP

Rozhodovací **problém Π patří do třídy NP**, jestliže pro něj existuje program pro nedeterministický TS, který každou instanci $I \in \Pi_{ano}$ řeší v čase $O(n^k)$, kde n je velikost instance a k konečné číslo.

Rozhodovací **problém Π patří do třídy NP**, jestliže pro každou instanci $I \in \Pi_{ano}$ existuje konfigurace Y taková, že kontrola, zda Y je řešením, patří do P. Y se potom nazývá **certifikátem.**

$$P \subseteq NP$$

Třída co-NP:
Instance je charakterizována řetězcem q^* symbolů $q \in \Gamma$.
Každý problém je charakterizován množinou řetězců kódujících instance s výstupem ANO, tedy podmnožinou množiny $\{q^*\}$
Problém **komplementární** je charakterizován doplňkkem této podmnožiny do $\{q^*\}$

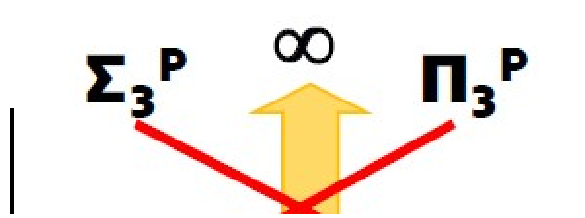
NP problém:

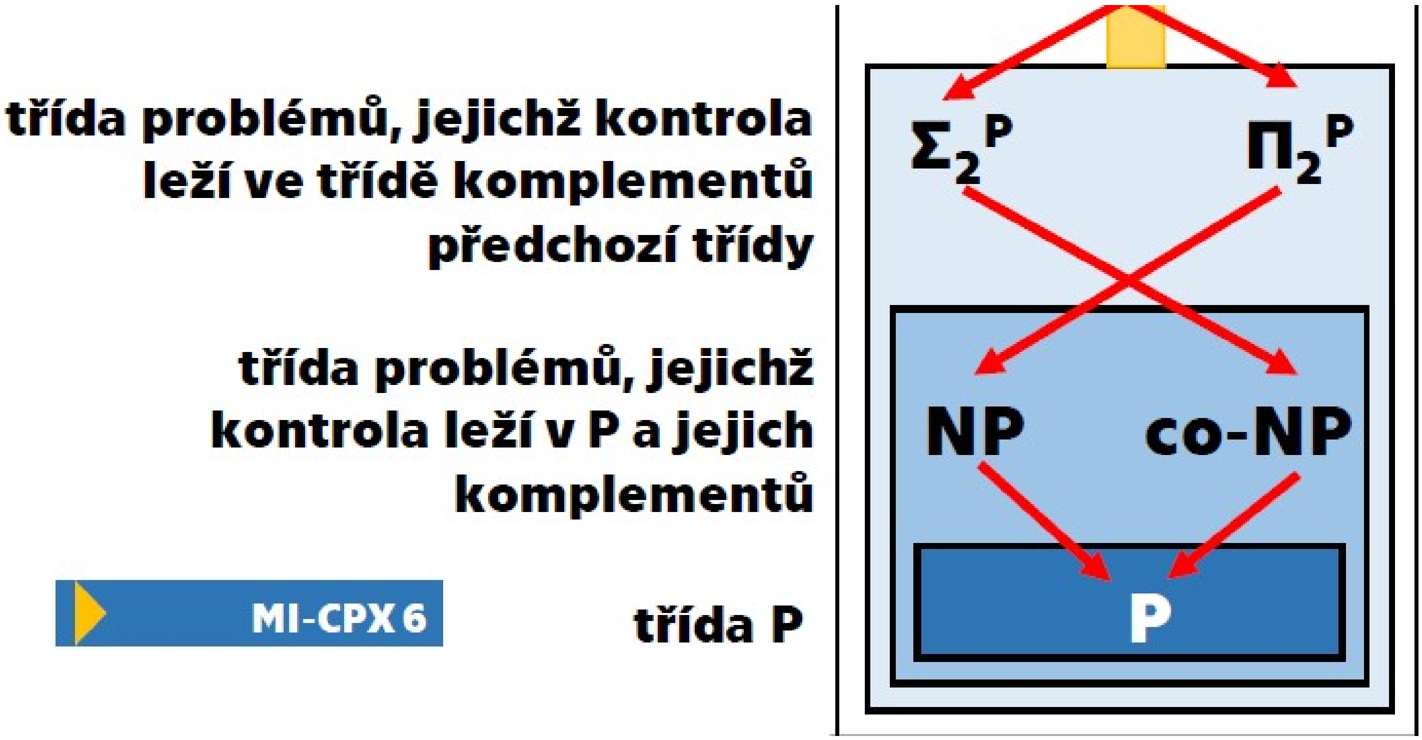
- $\exists Y, R(I, Y)?$
- *Existuje v grafu Hamiltonova kružnice?*
- Y je krátký svědek odpovědi ano ($\exists$-certifikát)
- Odpověď ne nemá krátkého svědka

co-NP problém:

- $\forall Y, R'(I, Y)?$, kde R' je komplement k omezujícím podmínkám
- *Je tento graf prost Hamiltonových kružnic?*
- Y, pro které neplatí R, je krátký svědek odpovědi ne
  *ano krátkého svědka nemá (dlouhý $\forall$-certifikát)

obvykle se polynomiální hierarchie zavádí pomocí Turingových strojů s orákulem





Problém Π je **X-těžký**, jestliže se efektivní řešení všech problému třídy X dá zredukovat na efektivní řešení problému Π.

Problém Π je **X-úplný**, jestliže je X-těžký a sám patří do třídy X.

Karpova redukce: Rozhodovací problém Π_1 je Karp-redukovatelný na Π_2, jestliže existuje polynomiální program pro DTS, kteerý převede každou instanci I_1 problému Π_1 na instanci I_2 problému Π_2 tak, že výstup obou instancí je shodný.

Tranzitivní

Třída NP-úplný

Problém Π je **NP-těžký**, jestliže pro všechny problémy $\Pi' \in$ NP platí, že jsou Karp-redukovatelné na Π

Problém Π je **NP-úplný**, jestliže $\Pi \in$ NP a pro všechny problémy $\Pi' \in$ NP platí, že jsou Karp-redukovatelné na Π

Cookova věta: SAT je NP-úplný
NPC tedy není prázdná množina.
Důkaz, že $\Pi \in$ NP lze provést převodem Π na SAT.

Třída NPO

Optimalizační problém patří do NPO, pokud splňuje:

- velikost výstupu instance je omezena polynomem ve velikosti instance (výstup lze zapsat v poly čase)
- problém, zda je daná konfigurace řešením, patří do P
- existuje program pro TS, který vypočítá hodnotu optimalizačního kritéria pro každé řešení každé instance v polynomiálním čase

Řešení optimalizačního problému TS:

- program M pro DTS řeší **optimalizační problém Π v čase t**, jestliže se výpočet zastaví po t krocích pro každou instanci problému Π, která má řešení, a na pásce je zapsán výstup instance
- program M pro DTS počítá **optimalizační kritérium problému Π v čase t**, pro každé řešení každé instance problému Π, zapsané na pásce jako vstupní data, se výpočet zastaví po t krocích a na pásce je zapsána hodnota optimalizačního kritéria.

Turingova redukce:
Rozhodovací problém Π_1 je Turing-redukovatelný na Π_2, jestliže existuje program pro DTS, který řeší každou instanci I_1 problému Π_1 tak, že používá program M_2 pro problém Π_2 jako podprogram *(nevyžaduje polynomiální čas)*

Karpova redukce je speciální případ Turingovy redukce.

Třída NP-těžký

Problém Π je NP-těžký, pokud pro všechny problémy $\Pi' \in$ NP platí, že jsou na Π Turing-redukovatelné v polynomiálním čase.

NPC $\subset$ NPH

Třída NP-intermediate (NPI)

Problémy, které nemohou mít polynomiální algoritmus, ale ani na ně nikdy nemůže bý převeden SAT
Např. izomorfismus grafů