Ako riešiť ťažké problémy?

Chceme:

- Rýchle algoritmy
- Správne algoritmy
- Algoritmy riešiace ťažké problémy

Exaktné metódy

- Inteligentné prehľadávanie (napr. A^* algoritmus)
- Pseudopolynomiálne algoritmy
- Celočíselné lineárne programovanie
- Parametrická zložitosť

Aproximačné algoritmy

Algoritmus A je $\boxed{k\text{-aproximačný}}$, ak pre každý vstup x platí $A(x) \leq kOPT(x)$ (maximalizačné problémy), resp. $A(x) \geq kOPT(x)$ (minimalizačné problémy).

Algoritmus $A(x, \varepsilon)$ je $\boxed{\text{polynomiálny aproximačná schéma (PTAS)}}$, ak jeho časová zložitosť je **polynomiálna vzhľadom k** $|x|$ a pre každé $\varepsilon > 0$
je $(1+\varepsilon)$-aproximačný (maximalizačné problémy),
resp. $(1-\varepsilon)$-aproximačný (minimalizačné problémy).

Algoritmus $A(x, \varepsilon)$ je
$\boxed{\text{plne polynomiálna aproximačná schéma (FPTAS)}}$, ak jeho časová zložitosť je navyše **polynomiálna vzhľadom na** $1/\varepsilon$.

Kladivá: Dôsledná analýza greedy algoritmov

(pre minimalizačné problémy)

- Problém: chceme porovnať výsledok greedy algoritmu s optimálnym riešením, ktorého hodnotu ale nevieme.
- **Dolný odhad D(x)** na hodnotu optimálneho riešenia na základe známych parametrov.
- **Horný odhad H(x)** na hodnotu greedy riešenia.
- Odhad aproximačného faktoru k:

$$k = \frac{\mathsf{GREEDY}(x)}{\mathsf{OPT}(x)} \leq \frac{H(x)}{D(x)}$$

(pre maximalizačné problémy obdobne)

Kladivá: Rozdeľ problém na podstatné a nepodstatné časti

- **Podstatné časti:**
  - Prispievajú **veľkou časťou** do hodnoty riešenia.
  - Majú špeciálne vlastnosti (napr. majú relatívne veľkú hodnotu, je ich pomerne málo, a pod.)
  - Na základe špeciálnych vlastností ich možno **efektívne riešiť optimálnym spôsobom** $\Rightarrow$ riešenie (*)
- **Nepodstatné časti:**
  - Priespievajú len **malou časťou** do hodnoty riešenia.
  - Vďaka tomu **nezáleží na tom**, ako presne ich doplníme ku riešeniu (*), pretože hodnotu príliš nezmenia.

Kladivá: Rozdeľ problém na ľahšie riešiteľné podproblémy

- **Ľahko riešiteľné podproblémy:**
  - Majú špeciálne vlastnosti (napr. malý rozsah, špeciálne usporiadanie prvkov a pod.)
  - Vďaka tomu ich možno **efektívne riešiť optimálnym spôsobom**
  - Každý podproblém $\Rightarrow$ samostatné čiastkové riešenie lokálne optimálne
- **Kombinácia čiastkových riešení:**
  - napr. priamočiaro spojíme čiastkové riešenia dohromady
  - Spojené riešenie nemusí byť optimálne (prekrývajúce sa prvky a pod.)
  - Ukážeme, že sme tým neurobili **príliš veľkú chybu.**

Kladivá: Zaokrúhli veľké čísla

- Predpoklad: pseudopolynomiálny algoritmus
- “Zmenšíme” čísla, ktoré vystupujú ako parametre v pseudopolynomiálnom algoritme, zaokrúhlime kde treba.
- Riešenie nie je exaktné (kôli zaokrúhľovaniu).
- Ukážeme, že zaokrúhľovaním sme neurobili veľkú chybu.

Kladivá: Relaxácia ILP

- Zapíšeme problém ako inštanciu celočíselného lineárneho programovania (ILP).
- **Relaxácia:** Nebudeme rešpektovať požiadavku na celočíselnosť riešenie, t.j. podmienky typu $x \in \{0, 1\}$ nahradíme podmienkou $0 \leq x \leq 1$.
- Riešením výsledného lineárneho programu dostaneme pseudoriešenie, ktorého hodnota **nie je horšia** ako optimálne riešenie.
- **Zaokrúhlenie:** Premenné, ktoré nie sú celé čísla, zaokrúhlime.
- Ukážeme, že sme tým **príliš nezmenili hodnotu riešenia.**

Pravdepodobnostné algoritmy

Algoritmy, ktoré využívajú **náhodné čísla.**

Las Vegas algoritmy.

- Vždy dajú správnu odpoveď.
- Náhodné čísla ovplyvňujú čas $\Rightarrow$ **očakávaná časová zložitosť**

Monte Carlo algoritmy.

- Bežia vždy rýchlo.
- Občas dajú nesprávnu odpoveď $\Rightarrow$ **pravdepodobnosť chyby** p
  - Jednostranné chyby
    (napr. "áno" je vždy dobre, "nie" môže byť chybné)
  - Obojstranné chyby

Dôležité: Rýchlosť/chybovosť algoritmu **nezávisí od vstupu**, ale len od výberu náhodných čísel! (t.j. neexistuje "zlý" vstup)

Kladivá (LV): Znáhodnenie deterministického algoritmu

- Namiesto deterministického kroku, ktorý vyžaduje aby sme urobili vyvážený výber, použijeme náhodný výber.
- Trik pri analýze očakávaného času:
  - Všetky prípady rozdelíme na **dobré** a **zlé**.
  - **Dobré prípady** nastávajú s vysokou pravdepodobnosťou a vieme urobiť dobrý horný odhad $h(x)$ na ich čas.
  - **Zlé prípady** nenastávajú často a aj keby sme použili najhorší možný horný odhad $H(x)$ času, tak to nevadí, lebo pravdepodobnosť je malá.

$$E[T(x,r)] = \sum_r \Pr(r).T(x,r) = \sum_{r \text{ je dobrý}} \Pr(r).T(x,r) + \sum_{r \text{ je zlý}} \Pr(r).T(x,r)$$

$$\leq \Pr(r \text{ je dobrý}).h(x) + \Pr(r \text{ je zlý}).H(x)$$

Kladivá (LV): Kernelizácia problému

- Predpoklad: Niektoré inštancie problému (napr. malé inštancie, riedke grafy, malé váhy a pod.) vieme riešiť efektívnejšie.

- Za pomoci **náhodného výberu** stransformujeme našu inštanciu na **menšiu/riedšiu/...** inštanciu, pričom nová inštancia **s vysokou pravdepodobnosťou** spĺňa podmienky, keď vieme inštancie riešiť efektívne.

Kladivá (LV+MC): Náhodné pochôdzky

- Problém zredukujeme na náhodnú pochôdzku.
- Na odhady potrebných pravdepodobností využijeme matematickú teóriu náhodných pochôdzok:
  - Ako dlho v priemere bude náhodná pochôdzka trvať?
  - Aká je distribúcia dĺžok náhodnej pochôdky?
  - Ako ďaleko sa v priemere počas náhodnej pochôdky dostaneme?
  - ...

Kladivá (LV+MC): Markovova nerovnosť (a ďalšie nerovnosti)

Nech X je náhodná premenná,
pričom $X \geq 0$ a $E[X] = \mu$.
Potom $\Pr(X \geq c\mu) \leq 1/c$

Príklad: Ak máme náhodnú pochôdzku, ktorá trvá v priemere k krokov, tak ak urobíme $2k$ krokov, tak s pravdepodobnosťou nanajvýš $1/2$ pochôdzku ukončíme!

Kladivá (analýza): Pravdepodobnostná metóda

Ak máme niekoľko prvkov, ktorých (váhovaný) priemer je K, potom

- musí existovať prvok s hodnotou $\leq K$
- musí existovať prvok s hodnotou $/geK$

Pozn. Stredná hodnota náhodnej premennej je tiež váhovaný priemer!

Kladivá (LV+MC): Metóda svedkov

- Ak by sme dostali ku problému nejakú ďalšiu informáciu (napr. čiastočnú informáciu o poradí prvkov, Fermatov svedok pre zložené číslo), vedeli by sme problém riešiť veľmi efektívne.
- Takúto informáciu nazývame **svedok**.
- Použijeme **náhodne vygenerovaného na svedka**!
- Ukážeme, že **zlého svedka**, t.j. takého, ktorý vedie k dlhému času (LV) alebo k chybnému riešeniu (MC) vygenerujeme **s malou pravdepodobnosťou**.

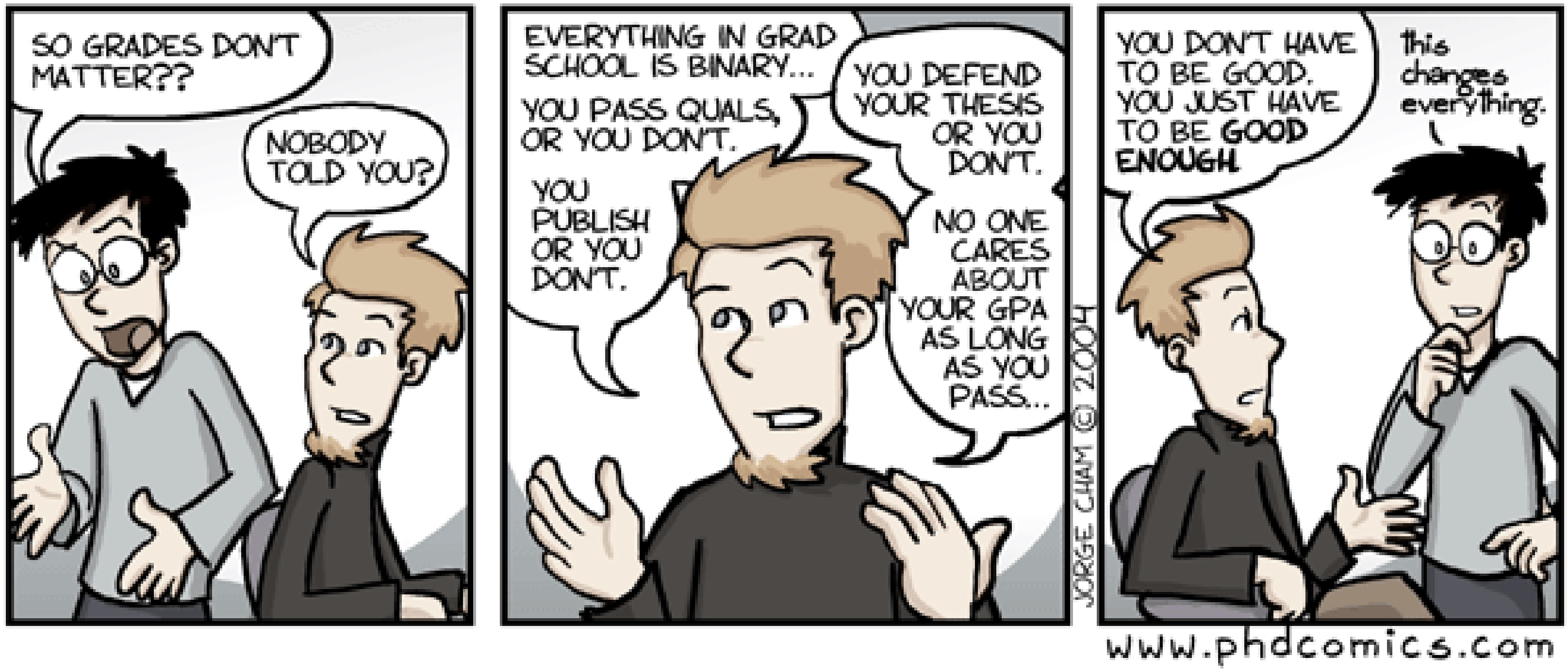
Kladivá (MC): Zlepšovanie úspešnosti opakovaním

- Predpokladajme, že máme MC algoritmus, ktorý urobí **chybu s pravdepodobnosťou** p.
- Ak takýto algoritmus spustíme k, krát, pravdepodobnosť že urobí chybu vo všetkých behoch je p^k
- Jednostranný MC algoritmus s pravdepodobnosťou chyby $1/2$, pravdepodobnosť správnej odpovede pri 4 opakovaniach je $\approx 94\%$.
- Jednostranný MC algoritmus s pravdepodobnosťou chyby 90% 20 opakovaní: pravdepodobnosť **správnej odpovede** $\approx 88\%$

Kladivá (MC): Metóda odtlačkov (fingerprinting)

- Namiesto porovnávania (veľkých) objektov bit po bite porovnávame len ich **odtlačky.**
- Odtlačky sú obvykle krátke a jednoducho porovnateľné.
- Výpočet odtlačku závisí od vygenerovaných náhodných čísel.
- Analýza: aká časť odtlačkov môže viesť ku chybnej zhode?

Doktorandské štúdium na FMFI UK



Náplň doktorandského štúdia

- 5% všeobecné štúdium
- 20% učenie (cvičenia a pod.)
- 75% samostatná vedecká práca

Absolvent magisterského štúdia:

- Dokáže sa učiť veci, ktoré nepozná, ale ktoré ľudia už poznajú a obvykle sú prehľadne spracované v knihách.
- Dokázal dokončiť jeden projekt (diplomovku), ktorého ciele mu určil vedúci diplomovej práce.

... doktorandské štúdium ...

Úspešný profesor / výskumný pracovník:

- Vie o najnovších poznatkoch vo svojom odbore / neustále študuje články opisujúce najnovšie vedecké výsledky.
- Vymýšľa nové veci, ktoré ešte ľudia nepoznajú.
- Sám si stanovuje ciele výskumných projektov, ktoré sú zaujímavé pre užšie alebo širšie skupiny odborníkov.

Čo ak nechcem skončiť v akademickej sfére?

- Získate schopnosť samostatnej práce na nových problémoch.
- Veľa našich absolventov: popredné slovenské startupy, nadnárodné firmy ako Google, Facebook a pod.
- Možnosť a čas pár rokov spojiť dohromady prácu a koníček
  . . . a rozmyslieť si, čo vlastne chcem v budúcnosti robiť.

Kde začať? Budúci školiteľ.

Počítačová grafika a modelovanie materiálov: prof. Ďurikovič, prof. Šrámek (ext.), doc. Chalmovianský, doc. Ferko, doc. Bajla (SAV)

Umelá inteligencia, robotika, kognitívna veda: prof. Farkaš, doc. Markošová, doc. Šefránek

Teoretická informatika: prof. Královič, prof. Rovan, prof. Ďuriš, doc. Pardubská, prof. Škoviera (grafy), doc. Toman (diskrétna matematika)

Distribuované výpočty a algoritmy: doc. Gruska, Dr. Dobrev (SAV), Dr. Vrťo (SAV)

Kryptológia, informačná bezpečnosť: doc. Stanek, doc. Olejár

Bioinformatika: Dr. Vinař, doc. Brejová

Vyučovanie informatiky: prof. Kalaš, doc. Kubincová, doc. Tomcsányiová