

Vztah teorie vyčísitelnosti a teorie složitosti

Časová složitost algoritmu

- počet “kroků” výpočtu
- závisí na vstupu a výpočetním modelu
- jako základní model použijeme **Turingův stroj**
- zkoumáme **nejhorší případ**, tedy maximální počet kroků v závislosti na délce vstupu
- lze zkoumat i **průměrný případ**

Časová složitost deterministického Turingova stroje

Definice. Necht' \mathcal{M} je úplný deterministický (jednopáskový nebo vícepáskový) Turingův stroj se vstupní abecedou Σ . Pro každé $w \in \Sigma^*$ definujeme $t_{\mathcal{M}}(w)$ jako počet kroků výpočtu stroje \mathcal{M} na vstupu w . **Časová složitost** stroje \mathcal{M} je pak funkce $T_{\mathcal{M}} : \mathbb{N}_0 \rightarrow \mathbb{N}$ definovaná vztahem

$$T_{\mathcal{M}}(n) = \max\{t_{\mathcal{M}}(w) \mid w \in \Sigma^n\}.$$

Říkáme, že \mathcal{M} **pracuje v čase** $T_{\mathcal{M}}(n)$.

Příklad

$$\mathcal{M} = (\{q_0, q_1, q_{acc}, q_{rej}\}, \{0, 1\}, \{0, 1, \triangleright, \sqcup\}, \triangleright, \sqcup, \delta, q_0, q_{acc}, q_{rej})$$

δ	\triangleright	0	1	\sqcup
q_0	(q_0, \triangleright, R)	$(q_0, 0, R)$	$(q_1, 1, R)$	$(q_{acc}, -, -)$
q_1		$(q_{rej}, -, -)$	$(q_1, 1, R)$	$(q_{acc}, -, -)$

Příklad

$$\mathcal{M} = (\{q_0, q_1, r, s_0, s_1, q_{acc}, q_{rej}\}, \{0, 1\}, \{0, 1, X, \triangleright, \sqcup\}, \triangleright, \sqcup, \delta, q_0, q_{acc}, q_{rej})$$

δ	\triangleright	0	1	X	\sqcup
q_0	(q_0, \triangleright, R)	$(q_0, 0, R)$	$(q_1, 1, R)$		(r, \sqcup, L)
q_1		$(q_{rej}, -, -)$	$(q_1, 1, R)$		(r, \sqcup, L)
r	(s_0, \triangleright, R)	$(r, 0, L)$	$(r, 1, L)$	(r, X, L)	
s_0		(s_1, X, R)	$(q_{rej}, -, -)$	(s_0, X, R)	$(q_{acc}, -, -)$
s_1		$(s_1, 0, R)$	(r, X, L)	(s_1, X, R)	$(q_{rej}, -, -)$

\triangleright	0	0	0	1	1	1	\sqcup	\sqcup	...
------------------	---	---	---	---	---	---	----------	----------	-----

Asymptotická analýza

Motivace

Věta. Pro každý deterministický úplný TM \mathcal{M} a pro každé $m > 1$ lze zkonstruovat deterministický úplný TM \mathcal{M}' tak, že $L(\mathcal{M}) = L(\mathcal{M}')$ a

$$T_{\mathcal{M}'}(n) \leq \frac{T_{\mathcal{M}}(n)}{m} + n + 1.$$

Důkaz.



\mathcal{O} -notace

Definice. Necht' $f, g : \mathbb{N}_0 \rightarrow \mathbb{R}^+$ jsou funkce. Řekneme, že g je **asymptotická horní závora** pro f , a píšeme $f \in \mathcal{O}(g)$ nebo $f = \mathcal{O}(g)$, jestliže existují konstanty $c, n_0 \in \mathbb{N}$ takové, že

$$\forall n \geq n_0 : f(n) \leq cg(n).$$

Příklad. $15n^3 + 3n^2 + 11n + 7$

Počítání s \mathcal{O} -notací

- logaritmy: $\mathcal{O}(\log n)$
- sčítání: $\mathcal{O}(n^3) + \mathcal{O}(n)$
- mocniny: $2^{\mathcal{O}(n)}$

Příklad

TM \mathcal{M} rozhodující $\{0^k 1^k \mid k \geq 0\}$ lze popsat i takto:

- 1** Zjistí, zda vstup obsahuje nějakou 0 za 1. Pokud ano, zamítne.
- 2** Dokud je na pásce nějaká 0, projíždí pásku a vždy škrtně jednu 0 a jednu 1. Pokud se nepovede k nějaké 0 najít 1, zamítne.
- 3** Pokud po vyškrtnání všech 0 zbude na pásce nějaká 1, zamítne. Jinak akceptuje.

Deterministické časové složitostní třídy problémů

časová složitost problému = nejmenší časová složitost, s jakou lze daný problém rozhodnout

Definice. Každá funkce $f : \mathbb{N}_0 \rightarrow \mathbb{R}^+$ definuje **(deterministickou) časovou složitostní třídu problémů**

$\text{TIME}(f(n)) = \{L \mid L \text{ je rozhodovaný nějakým deterministickým jedno- nebo vícepáskovým TM } \mathcal{M} \text{ s časovou složitostí } T_{\mathcal{M}}(n) = \mathcal{O}(f(n))\}.$

Příklad

$$L = \{0^k 1^k \mid k \geq 0\}$$

- ukázali jsme jednopáskový det. TM rozhodující L v čase $\mathcal{O}(n^2)$
- existuje jednopáskový det. TM rozhodující L v čase $\mathcal{O}(n \log n)$
- neexistuje jednopáskový det. TM rozhodující L s menší složitostí
- existuje dvoupáskový deterministický TM rozhodující L v čase $\mathcal{O}(n)$, tedy L je ve třídě $\text{TIME}(n)$

Vliv výpočetního modelu

- na rozdíl od vyčíslitelnosti, ve složitosti **na výpočetním modelu záleží**
- rozdíl je i mezi jednopáskovým a dvoupáskovým deterministickým TM
- jaký model zvolit?
- je volba deterministického vícepáskového TM správná?
- rozdíly jsou u běžných sekvenčních deterministických výpočetních modelů poměrně malé
- např. RAM (random-access machine) pracující v čase $f(n)$ lze převést na vícepáskový deterministický TM pracující v čase $\mathcal{O}(f^3(n) \cdot (f(n) + n)^2)$
- nedeterminismus přináší výrazný rozdíl

Převod vícepáskového TM na jednopáskový

Věta. Pro každý vícepáskový deterministický TM pracující v čase $f(n) \geq n$ lze sestavit ekvivalentní jednopáskový deterministický TM pracující v čase $\mathcal{O}(f^2(n))$.

Důkaz.

- 1 neprázdný obsah k pásek a polohy hlav zapíšeme za sebe na 1 pásku $\rightarrow \mathcal{O}(n)$
- 2 simulace jednoho kroku
 - zjistit informace pod hlavami = projít pásku, každá původní páska má max. $f(n)$ neprázdných polí $\rightarrow \mathcal{O}(f(n))$
 - provést krok, zapsat nové symboly a posunout hlavy (případně přidat další políčka na původní pásy odsunutím obsahu dalších pásek, max. k políček) $\rightarrow \mathcal{O}(f(n))$
- 3 simulujeme $f(n)$ kroků \rightarrow celkem $\mathcal{O}(n) + \mathcal{O}(f^2(n))$

Časová složitost nedet. Turingova stroje

Definice. Necht' \mathcal{M} je úplný nedeterministický Turingův stroj se vstupní abecedou Σ . Pro každé $w \in \Sigma^*$ definujeme $t_{\mathcal{M}}(w)$ jako počet kroků nejdelšího výpočtu stroje \mathcal{M} na vstupu w . **Časová složitost** stroje \mathcal{M} je pak funkce $T_{\mathcal{M}} : \mathbb{N}_0 \rightarrow \mathbb{N}$ definovaná vztahem

$$T_{\mathcal{M}}(n) = \max\{t_{\mathcal{M}}(w) \mid w \in \Sigma^n\}.$$

Nedeterministické časové složitostní třídy problémů

Definice. Každá funkce $f : \mathbb{N}_0 \rightarrow \mathbb{R}^+$ definuje (**nedeterministickou**) **časovou složitostní třídu problémů**

$$\text{NTIME}(f(n)) = \{L \mid L \text{ je rozhodovaný nějakým nedeterministickým jedno- nebo vícepáskovým TM } \mathcal{M} \text{ s časovou složitostí } T_{\mathcal{M}}(n) = \mathcal{O}(f(n))\}.$$

Z definic plyne $\text{TIME}(f(n)) \subseteq \text{NTIME}(f(n))$.

Převod nedeterministického TM na deterministický

Věta. Pro každý nedeterministický jednopáskový TM pracující v čase $f(n) \geq n$ lze sestavit ekvivalentní deterministický jednopáskový TM pracující v čase $2^{\mathcal{O}(f(n))}$.

Důkaz. Nedet. TM \mathcal{M} , který má z každé konfigurace max. k přechodů, simulujeme 3-páskovým deterministickým strojem, který prohledává strom výpočtů \mathcal{M} do šířky. Pro každý uzel provedeme znovu výpočet z iniciální konfigurace.

Strom výpočtů má hloubku nejvýše $f(n)$ a tudíž má nejvýše $k^{f(n)}$ listů a méně než $2 \cdot k^{f(n)}$ uzlů. $\rightarrow \mathcal{O}(k^{f(n)})$ uzlů

Simulace výpočtu do jednoho uzlu zabere nejvýše $\mathcal{O}(f(n))$ kroků. 3-páskový stroj tedy pracuje v čase $\mathcal{O}(f(n) \cdot k^{f(n)}) = 2^{\mathcal{O}(f(n))}$.

Převod na jednopáskový det. stroj: $(2^{\mathcal{O}(f(n))})^2 = 2^{\mathcal{O}(2f(n))} = 2^{\mathcal{O}(f(n))}$. □

Nejvýznamnější časové složitostní třídy

deterministické

nedeterministické

$$P = \text{PTIME} = \bigcup_{k \in \mathbb{N}} \text{TIME}(n^k)$$

$$\text{EXPTIME} = \bigcup_{k \in \mathbb{N}} \text{TIME}(2^{n^k})$$

$$NP = \bigcup_{k \in \mathbb{N}} \text{NTIME}(n^k)$$

$$\text{NEXPTIME} = \bigcup_{k \in \mathbb{N}} \text{NTIME}(2^{n^k})$$

- z definic a předchozí věty plyne: $P \subseteq NP \subseteq \text{EXPTIME} \subseteq \text{NEXPTIME}$
- běžné deterministické sekvenční modely výpočtu lze mezi sebou převádět s polynomiálním nárůstem časové složitosti
 \implies definice P a EXPTIME nejsou citlivé na volbu modelu
- EXPTIME je obvykle složitost algoritmů řešících problém hrubou silou
- **Cook-Karpova teze**
 P obsahuje právě všechny prakticky řešitelné problémy.

Vztah P a NP

$$P \stackrel{?}{=} NP$$

- asi nejznámější otevřený problém teoretické informatiky
- věří se, že platí $P \subsetneq NP$
- důsledky do počítačové bezpečnosti
- Clay Mathematics Institute (CMI) vypsal **1.000.000 USD** za řešení

Příslušnost problémů v P

- stačí ukázat, že problém je řešitelný v polynomiálním počtu kroků a že každý krok je implementovatelný v polynomiálním čase
- kódování/dekódování objektů O do slov $\langle O \rangle$ musí být proveditelné v polynomiálním čase
- příklad vhodného kódování: reprezentace grafu maticí sousednosti

- příklad nevhodného kódování: reprezentace sekvence číslic unárním zápisem čísla

Problém existence cesty

Problém existence cesty je problém rozhodnout, zda v daném orientovaném grafu G existuje cesta z s do t .

$$PATH = \{ \langle G, s, t \rangle \mid G \text{ je orientovaný graf obsahující cestu z } s \text{ do } t \}$$

Věta. $PATH \in P$.

Důkaz. Postupně spočítáme uzly dosažitelné z s .

- 1 označ uzel s
- 2 dokud lze označit nový uzel opakuj:
projdi všechny hrany v G a označ každý uzel,
do kterého vede hrana z označeného uzlu
- 3 je-li t označeno, akceptuj; jinak zamítni

Celkem $\mathcal{O}(n)$ kroků (n je počet uzlů v G), každý lze provést v polynomiálním čase.



Problém Hamiltonovské cesty

Hamiltonovská cesta = cesta procházející každým uzlem právě jednou

Problém Hamiltonovské cesty je problém rozhodnout, zda v daném orientovaném grafu G existuje Hamiltonovská cesta z s do t .

$$HAMPATH = \{ \langle G, s, t \rangle \mid G \text{ je orientovaný graf obsahující Hamiltonovskou cestu z } s \text{ do } t \}$$

Problém Hamiltonovské cesty

Věta. $HAMPATH \in NP$.

Důkaz.

- Hamiltonovská cesta v grafu G s n uzly má délku $n - 1$
- Hamiltonovskou cestu budeme nedeterministicky hádat
 - 1 začni budovat cestu z uzlu s
 - 2 $(n - 1)$ -krát opakuj: nedeterministicky vyber hranu vedoucí z posledního uzlu cesty a přidej ji na konec cesty
 - 3 je-li t poslední uzel cesty a žádný uzel se neopakuje, akceptuj; jinak zamítni
- každý výpočet má $\mathcal{O}(n)$ polynomiálních kroků
- Hamiltonovská cesta existuje \iff existuje akceptující výpočet



Polynomiální redukce

Definice. Necht' $A \subseteq \Sigma^*$ a $B \subseteq \Phi^*$ jsou jazyky. Řekneme, že A se **polynomiálně redukuje** na B , píšeme $A \leq_p B$, právě když $A \leq_m B$ a redukční funkce f je vyčíslitelná Turingovým strojem pracujícím v polynomiálním čase. Funkci f nazveme **redukcí A na B v polynomiálním čase**.

Polynomiální redukce a složitostní třídy

Věta. Necht' $A \leq_p B$.

- $B \in P \implies A \in P$
- $B \in NP \implies A \in NP$

Důkaz. Necht' f je redukce A na B v polynomiálním čase a \mathcal{M}_B je TM rozhodující B . Stroj \mathcal{M}_A rozhodující A na vstupu w

- 1 spočítá $f(w)$
- 2 spustí \mathcal{M}_B na vstupu $f(w)$ a vrátí stejný výsledek jako \mathcal{M}_B

Je-li \mathcal{M}_B deterministický, pak je i \mathcal{M}_A deterministický. Krok 1 lze provést v polynomiálním čase vzhledem k $|w|$, krok 2 v polynomiálním čase vzhledem k $|f(w)|$, což je polynomiální i vzhledem k $|w|$. \mathcal{M}_A tedy pracuje v polynomiálním čase. □

Polynomiální redukce a složitostní třídy

Definice. Necht' \mathcal{C} je složitostní třída splňující $P \subseteq \mathcal{C}$. Jazyk L nazveme **těžký** ve třídě \mathcal{C} (**\mathcal{C} -těžký**), právě když pro každý jazyk $L' \in \mathcal{C}$ platí $L' \leq_p L$.

Řekneme, že L je **úplný** ve třídě \mathcal{C} (**\mathcal{C} -úplný**), pokud navíc $L \in \mathcal{C}$.

Problém splnitelnosti (SAT)

Problém splnitelnosti (SAT) je problém rozhodnout, zda je daná Booleovská formule (formule poskládaná z výrokových proměnných s využitím operací \wedge , \vee a \neg) splnitelná.

$$SAT = \{ \langle \varphi \rangle \mid \varphi \text{ je splnitelná Booleovská formule} \}$$

Věta. *SAT* je NP-úplný.

Důkaz. Postupně ukážeme $SAT \in NP$ a *SAT* je NP-těžký. □

$SAT \in NP$

$SAT = \{ \langle \varphi \rangle \mid \varphi \text{ je splnitelná Booleovská formule} \}$

$SAT \in NP$: