

IB107 Vyčísitelnost a složitost

časová složitost algoritmu, časové složitostní třídy

Jan Strejček

Fakulta informatiky
Masarykova univerzita

- 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**

Definice 1.1 (časová složitost TM)

Nechť \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 $T_{\mathcal{M}} : \mathbb{N} \rightarrow \mathbb{N}$ 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	...
------------------	---	---	---	---	---	---	----------	----------	-----

motivace pro asymptotickou analýzu

- jaká složitost je lepší: $6n$ nebo $n^2 + 5$?

- na delších vstupech jsou výpočty obvykle delší
- zajímá nás chování pro $n \rightarrow \infty$
- konstantní faktor neuvažujeme (výpočet lze zrychlit)

Věta (o zrychlení)

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:

Definice 1.2 (asymptotická horní závora)

Nechť $f, g : \mathbb{N} \rightarrow \mathbb{R}_0^+$ 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 pro každé $n \geq n_0$ platí

$$f(n) \leq c \cdot g(n).$$

Příklad: $15n^3 + 3n^2 + 11n + 7$

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

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 vyškrtání všech 0 zbude na pásce nějaká 1, zamítne. Jinak akceptuje.

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

Definice 1.5 (časová složitostní třída problémů)

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

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

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

- ukázali jsme jednopáskový det. TM rozhodující L v čase $\mathcal{O}(n^2)$
- existuje jednopáskový det. TM rozhodující L v čase $\mathcal{O}(n \log n)$
 - 1 Zjistí, zda vstup obsahuje nějakou 0 za 1. Pokud ano, zamítne.
 - 2 Dokud páska obsahuje nějakou 0 i 1, opakuje následující:
Pokud je celkový počet 0 a 1 n pásce lichý, zamítne. Pokud je sudý, škrtně každý lichý výskyt 0 a každý lichý výskyt 1.
 - 3 Jestliže na pásce zůstane nějaká 0 nebo 1, zamítne. Jinak akceptuje.

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

- neexistuje jednopáskový det. TM rozhodující L s menší složitostí než $\mathcal{O}(n \log n)$
(platí, že každý jazyk rozhodnutelný jednopáskovým det. TM v čase $\mathcal{O}(n \log n)$ je regulární)
- existuje dvoupáskový deterministický TM rozhodující L v čase $\mathcal{O}(n)$, tedy L je ve třídě TIME(n)

- 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 1.6

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 pásky odsunutím obsahu dalších pásek, max. k políček) $\rightarrow \mathcal{O}(f(n))$
- 3 simulujeme $f(n)$ kroků \rightarrow celkem $\mathcal{O}(n) + \mathcal{O}(f^2(n))$ ■

Definice 1.7 (časová složitost nedeterministického TM)

Nechť \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 $T_{\mathcal{M}} : \mathbb{N} \rightarrow \mathbb{N}$ definovaná vztahem

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

Definice 1.16 (nedeterministická časová složitostní třída problémů)

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

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

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

Věta 1.8

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

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

nedeterministické

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

$$P \subseteq NP \subseteq \text{EXPTIME} \subseteq \text{NEXPTIME}$$

- 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.

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

- 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