Věty o hierarchii

14 December, 2020 16:20

- funkce $s(n): \mathbb{N} \to \mathbb{N}$ je prostorově konstruovatelná, pokud existuje TS taký, že na každém vstupu délky n naváží právě $s(n)$ políček na pracovní pásce a pak se zastaví.

(prostorová hierarchie)

$S(n) \geq \lg n$

Věta: Nechť $s(n) = o(S(n))$ jsou dvě prostorově konstruovatelné funkce. Pak $DSPACE(s(n)) \subsetneq DSPACE(S(n))$.

Dk: • $DSPACE(s(n)) \subseteq DSPACE(S(n))$ ✓

stačí tedy ukázat, že $\exists A \in DSPACE(S(n)) \setminus DSPACE(s(n))$

Popíšeme algoritmus pro jazyk A, který pracuje v prostoru $S(n)$. Zároveň diagonalizací zajistíme, že $A \notin DSPACE(s(n))$.

→ chceme, aby pro každý TS M pracující v prostoru $O(s(n))$ $\exists$ vstup x t.ž.

$x \in A \iff M$ nepřijímá x.

$\Rightarrow$ alg pro A: na vstupu x:

1) pokud $x = \langle M \rangle 1 0^l$ pro nějaký TS M pokračuj dále, jinak odmítni x
2) vyznač na pásce $S(|x|)$ políček.
3) ve vyznačeném prostoru simuluj běh M na vstupu x po $2^{S(|x|)}$ kroků.

4) Pokud M přijme x, zamítni, jinak přijmi. $\square$

Algoritmus pracuje v prostoru $O(S(n))$.
(používáme čítač kroků ($S(n)$ bitů) + pracovní prostor $S(n)$ políček pro simulaci.)

$\Rightarrow$ $A \in DSPACE(S(n))$

Chceme ukázat, že $A \notin DSPACE(s(n))$.
Sporem: Nechť M je TS rozpoznávající A a pracující v prostoru $s'(n) = O(s(n))$.
Simulace jeho výpočtu algoritmus A vyžaduje prostor $ds'(n)$ (pracovní abeceda M se kóduje pomocí $\leq d$ symbolů algoritmu.)

Jelikož $ds'(n) < S(n)$ pro dostatečně velké n, pro dostatečně dlouhý vstup $x = \langle M \rangle 1 0^l$, algoritmus pro A dokáže odsimulovat celý průběh výpočtu M na vstupu $x = \langle M \rangle 1 0^l$. Pak ovšem na vstupu x chování M a našeho algoritmu na x se liší: $x \in A \iff M$ nepřijme x $\square$

$\Rightarrow$ M nepočítá A

• **Př.:** • $DSPACE(n) \subsetneq DSPACE(n^2)$

• $DSPACE(n) \subsetneq DSPACE(n \log \lg n) \subsetneq DSPACE(n \lg n)$

• $NL \subseteq DSPACE(\lg^2 n) \subsetneq DSPACE(n) \subseteq PSPACE$

$\Rightarrow NL \neq PSPACE \Rightarrow QBF \notin NL$.

• $\forall$ racionální čísla $\varepsilon_1 > \varepsilon_2 > 0$

$$DSPACE(n^{\varepsilon_2}) \subsetneq DSPACE(n^{\varepsilon_1}).$$

$$L \subseteq \underbrace{NL \subseteq P \subseteq PSPACE}_{\neq} \subseteq EXP.$$

• funkce $t(n): \mathbb{N} \to \mathbb{N}$ je časově konstruovatelná, pokud $\exists$ TS M, který na každém vstupu délky n spočítá binárně zapsané $t(n)$ v čase $O(t(n))$.

• **Př.:** $n^2, n \lg n, n,$ $2^n, n^{\lg n}$ } jsou časově konstruovatelné

Věta (o časové hierarchii): Nechť $T(n), t(n)$ jsou časově konstruovatelné, kde $(t(n))^2 = o(T(n))$. Pak $DTIME(t(n)) \subsetneq DTIME(T(n))$.

Dk.: Podobný jako pro prostorovou hierarchii.

Navrhneme alg. počítající jazyk $A \in DTIME(T(n))$

Alg pro A: na vstupu x

1) pokud $x = \langle M \rangle 1 0^{\ell}$ pro nějaký TS M, pokračuj dále, jinak zamítni

2) spočítej $T(n)$

3) simuluj M na vstupu x tak, aby simulace zabrala nejvýše $T(n)$ kroků.

4) Pokud M skončí výpočet a přijme x, zamítni, jinak přijmi.

$\Rightarrow A \in DTIME(T(n))$

Pokud by A bylo rozpoznáváno TS M pracujícím v čase $O(t(n))$, pro dostatečně velké ℓ dostaneme spor na vstupu $x = \langle M \rangle 1 0^{\ell}$ protože stihneme odsimulovat celý výpočet M na x. (simulace M na vstupech délky n trvá $\leq d\,(t(n))^2$ kroků, pro nějakou konstantu d závislou pouze na M.) $\boxed{\text{QED}}$

Nejlepší známá věta: $T(n) = \omega(t(n) \lg t(n))$

$t(n) \geq n$, pak $DTIME(t(n)) \subsetneq DTIME(T(n))$

– vyžaduje rychlejší simulaci výpočtu $t(n)$ kroků za čas $O(t(n) \lg t(n))$.

Pro jednopáskové TS platí: $DTIME(t(n)) \subsetneq DTIME(T(n))$ pokud $t(n) = o(T(n))$.

pokud $t(n) = o(T(n))$.

Př: • $DTIME(n) \subsetneq DTIME(n^3)$

• $\forall k: DTIME(n^k) \subsetneq DTIME(n^{\log n}) \subsetneq EXP$

$\Rightarrow P \subsetneq EXP$

$P \neq DTIME(n^{10})$

$\underbrace{P \subseteq NP \subseteq EXP}_{\neq}$