Języki, automaty i obliczenia

Wykład 12: Złożoność czasowa i pamięciowa. Klasa NP

Sławomir Lasota

Uniwersytet Warszawski

29 maja 2024

Plan

Czas i pamięć

$n = |w|$ długość słowa wejściowego w.

Rozważamy funkcje $f : \mathbb{N} \to \mathbb{N}$ t.że $f(n) \geq n$.

Maszyna Turinga $\mathcal{M}$, deterministyczna lub nie,

- *działa w czasie $f(n)$* jeśli dla każdego słowa wejściowego w maszyna wykonuje co najwyżej $f(|w|)$ kroków.
- *działa w pamięci $f(n)$* jeśli dla każdego słowa wejściowego w maszyna odwiedza co najwyżej $f(|w|)$ komórek taśmy.

Fakt

Maszyna deterministyczna działająca w czasie $f(n)$ jest całkowita.

Czas i pamięć (przykład)

Deterministyczna maszyna $\mathcal{M}$:

	a	b	$\mathbb{B}$	$\#$
start$_\rightarrow$	(cont$_\rightarrow$, $\#$, $\rightarrow$)	(check, b, $\rightarrow$)	(nok , $\mathbb{B}$, $\circlearrowleft$)	(nok , $\#$, $\circlearrowleft$)
cont$_\rightarrow$	(cont$_\rightarrow$, a, $\rightarrow$)	(cont$_\rightarrow$, b, $\rightarrow$)	(start$_\leftarrow$, $\mathbb{B}$, $\leftarrow$)	(start$_\leftarrow$, $\#$, $\leftarrow$)
start$_\leftarrow$	(cont$_\leftarrow$, $\#$, $\leftarrow$)	(nok , b, $\circlearrowleft$)	(nok , $\mathbb{B}$, $\circlearrowleft$)	(nok , $\#$, $\circlearrowleft$)
cont$_\leftarrow$	(cont$_\leftarrow$, a, $\leftarrow$)	(cont$_\leftarrow$, b, $\leftarrow$)	(nok , $\mathbb{B}$, $\circlearrowleft$)	(start$_\rightarrow$, $\#$, $\rightarrow$)
check	(nok , a, $\circlearrowleft$)	(nok , b, $\circlearrowleft$)	(ok , $\mathbb{B}$, $\circlearrowleft$)	(ok , $\#$, $\circlearrowleft$)

$L(\mathcal{M}) = \{ a^n b\, a^n : n \in \mathbb{N} \}$

$\mathcal{M}$ działa w czasie n^2 i pamięci n.

Pytanie

A gdyby $\mathcal{M}$ miała 2 taśmy?

klasa języków	języki rozpoznawane przez
DTIME($f(n)$)	deterministyczną maszynę w czasie $f(n)$
NTIME($f(n)$)	niedeterministyczną maszynę w czasie $f(n)$
DSPACE($f(n)$)	deterministyczną maszynę w pamięci $f(n)$
NSPACE($f(n)$)	niedeterministyczną maszynę w pamięci $f(n)$

Pytanie

Czy wystarczy ograniczyć się do obliczalnych funkcji f?

Twierdzenie

Wszystkie klasy są właściwymi podklasami języków rozstrzygalnych.

Dowód:

$$L_p := \{ w_{\mathcal{M}} : w_{\mathcal{M}} \notin L_{\leq f}(\mathcal{M}) \}.$$

Niech $L_p = L(\mathcal{M}_p)$ dla pewnej maszyny $\mathcal{M}_p$. Gdyby $L_p = L_{\leq f}(\mathcal{M})$, to

$$w_{\mathcal{M}} \in L_{\leq f}(\mathcal{M}) \iff w_{\mathcal{M}} \notin L_{\leq f}(\mathcal{M}).$$

Hierarchia

klasa języków	języki rozpoznawane przez
DTIME($f(n)$)	deterministyczną maszynę w czasie $f(n)$
NTIME($f(n)$)	niedeterministyczną maszynę w czasie $f(n)$
DSPACE($f(n)$)	deterministyczną maszynę w pamięci $f(n)$
NSPACE($f(n)$)	niedeterministyczną maszynę w pamięci $f(n)$

Twierdzenie

$$DTIME(f(n)) \subseteq NTIME(f(n)) \subseteq DSPACE(f(n)) \subseteq NSPACE(f(n))$$

$$NSPACE(f(n)) \subseteq \bigcup_c DTIME(2^{c \cdot f(n)})$$ *założenie!*

Klasy złożoności z dokładnością do wielomianu

$$
\begin{array}{rrrrl}
 & \ldots & & & \\
 & \mathrm{EXPSPACE} & = \bigcup_p \mathrm{DSPACE}(2^{p(n)}) & = \bigcup_p \mathrm{NSPACE}(2^{p(n)}) \\
 & \mathrm{NEXPTIME} & = \bigcup_p \mathrm{NTIME}(2^{p(n)}) & & \\
 & \mathrm{EXPTIME} & = \bigcup_p \mathrm{DTIME}(2^{p(n)}) & & \\
 & \mathrm{PSPACE} & = \bigcup_p \mathrm{DSPACE}(p(n)) & = \bigcup_p \mathrm{NSPACE}(p(n)) \\
NP & = \quad \mathrm{NPTIME} & = \bigcup_p \mathrm{NTIME}(p(n)) & & \\
P & = \quad \mathrm{PTIME} & = \bigcup_p \mathrm{DTIME}(p(n)) & &
\end{array}
$$

($p \in$ wielomiany)

Przykładowe problemy w NP

Problem spełnialności formuły zdaniowej (SAT)

Dane: formuła zdaniowa ϕ.

Wynik: czy ϕ jest spełnialna?

np. $\phi \equiv (x \wedge y) \vee \neg(x \wedge (\neg y \vee z))$

3-SAT

Dane: formuła zdaniowa ϕ w postaci 3-CNF.

Wynik: czy ϕ jest spełnialna?

np. $\phi \equiv (x \vee y) \wedge (x \vee \neg y \vee \neg z)$

3-kolorowalność

Dane: graf G.

Wynik: czy da się pokolorować wierzchołki G trzema kolorami tak, żeby kolory sąsiadów były różne?

ogólny schemat

Plan

Redukcje wielomianowe

Problem $K \subseteq A^*$ *redukuje się* *wielomianowo* do problemu $L \subseteq B^*$ (ozn. $K \leq_p L$) jeśli istnieje funkcja obliczalna w czasie wielomianowym

$$f : A^* \to B^*$$

taka, że

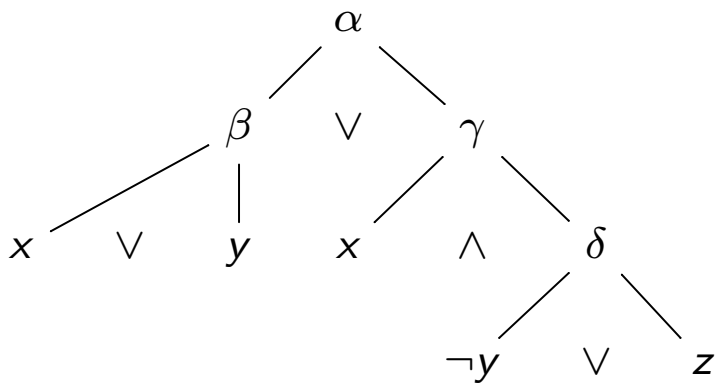
$$w \in K \iff f(w) \in L, \quad \text{dla każdego } w \in A^*.$$

SAT $\leq_p$ 3-SAT

funkcja obliczalna w czasie wiel.:	ϕ	$\longmapsto$	ψ w postaci 3-CNF
poprawność:	ϕ spełnialna	$\iff$	ψ spełnialna

$$\phi \equiv (x \wedge y) \vee (x \wedge (\neg y \vee z)) \quad \mapsto$$

$$\alpha \wedge (\alpha \iff \beta \vee \gamma) \wedge (\beta \iff x \wedge y) \wedge (\gamma \iff x \wedge \delta) \wedge (\delta \iff \neg y \vee z)$$



$$\alpha \iff \beta \vee \gamma$$

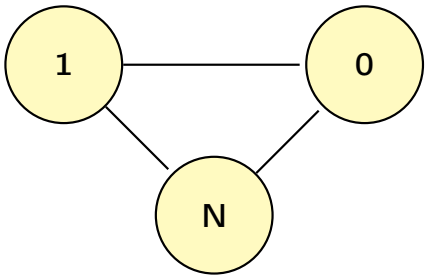
$$(\neg\alpha \vee \beta \vee \gamma) \wedge ((\neg\beta \wedge \neg\gamma) \vee \alpha)$$

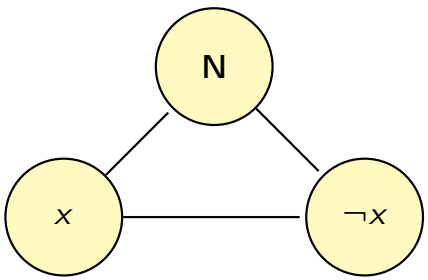
$$(\neg\alpha \vee \beta \vee \gamma) \wedge (\neg\beta \vee \alpha) \wedge (\neg\gamma \vee \alpha)$$

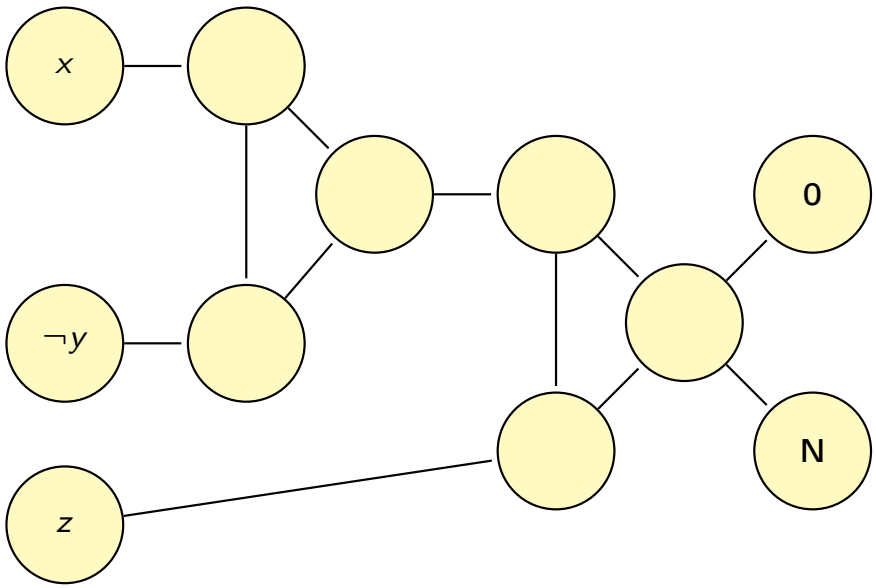
3-SAT $\leq_p$ 3-kolorowalność

funkcja obliczalna w czasie wiel.:	ϕ w postaci 3-CNF	$\longmapsto$	G_ϕ
poprawność:	ϕ spełnialna	$\Longleftrightarrow$	G_ϕ 3-kolorowalny

np. $\phi \equiv (x \vee \neg y \vee z) \wedge (\neg x \vee \neg z \vee u) \wedge \ldots$







Zamkniętość na redukcje wielomianowe

Fakt

Niech $\mathcal{C}$ – klasa złożoności z listy poniżej. Jeśli $K \leq_p L$ i $L \in \mathcal{C}$ to $K \in \mathcal{C}$.

$$
\begin{aligned}
& & & \dots & & \\
& & \mathrm{EXPSPACE} &= \bigcup_p \mathrm{DSPACE}(2^{p(n)}) &&= \bigcup_p \mathrm{NSPACE}(2^{p(n)}) \\
& & \mathrm{NEXPTIME} &= \bigcup_p \mathrm{NTIME}(2^{p(n)}) && \\
& & \mathrm{EXPTIME} &= \bigcup_p \mathrm{DTIME}(2^{p(n)}) && \\
& & \mathrm{PSPACE} &= \bigcup_p \mathrm{DSPACE}(p(n)) &&= \bigcup_p \mathrm{NSPACE}(p(n)) \\
NP &= & \mathrm{NPTIME} &= \bigcup_p \mathrm{NTIME}(p(n)) && \\
P &= & \mathrm{PTIME} &= \bigcup_p \mathrm{DTIME}(p(n)) &&
\end{aligned}
$$

Plan

Trudność i zupełność

Problem L jest *$\mathcal{C}$-trudny* jeśli każdy problem $K \in \mathcal{C}$ redukuje się wielomianowo do L.

Problem L jest *$\mathcal{C}$-zupełny* jeśli jest $\mathcal{C}$-trudny i należy do $\mathcal{C}$.

Fakt

Jeśli $K \leq_p L$ i K jest $\mathcal{C}$-trudny to L jest $\mathcal{C}$-trudny.

$$
\begin{aligned}
& \dots \\
\mathrm{EXPSPACE} &= \bigcup_p \mathrm{DSPACE}(2^{p(n)}) &&= \bigcup_p \mathrm{NSPACE}(2^{p(n)}) \\
\mathrm{NEXPTIME} &= \bigcup_p \mathrm{NTIME}(2^{p(n)}) \\
\mathrm{EXPTIME} &= \bigcup_p \mathrm{DTIME}(2^{p(n)}) \\
\mathrm{PSPACE} &= \bigcup_p \mathrm{DSPACE}(p(n)) &&= \bigcup_p \mathrm{NSPACE}(p(n)) \\
NP = \quad \mathrm{NPTIME} &= \bigcup_p \mathrm{NTIME}(p(n))
\end{aligned}
$$

NP-trudność i zupełność

Problem L jest *NP-trudny* jeśli każdy $K \in$ NP redukuje się wielomianowo do L.

Problem L jest *NP-zupełny* jeśli jest NP-trudny i należy do NP.

Fakt

Jeśli $K \leq_p L$ i K jest NP-trudny to L jest NP-trudny.

Twierdzenie (Cook 1971, Levin 1973)

SAT jest NP-zupełny.

SAT jest NP-zupełny (dowód)

Niech $\mathcal{M}$ – maszyna niedet. działająca w czasie n^c. Pokażemy $L(M) \leq_p$ SAT.

funkcja obliczalna w czasie wiel.:	$w = a_1 \dots a_n$	$\longmapsto$	$\phi_{\mathcal{M},w}$
poprawność:	$w \in L(\mathcal{M})$	$\iff$	$\phi_{\mathcal{M},w}$ spełnialna

Zmienne:

- $t^{i,j,a}$ – po i krokach, na pozycji j taśmy jest symbol a
- $s^{i,q}$ – po i krokach, stan maszyny to q
- $g^{i,j}$ – po i krokach, głowica maszyny jest na pozycji j taśmy

$\bigwedge_{j \leq |w|} t^{0,j,a_j} \wedge \bigwedge_{j > |w|} t^{0,j,\mathbb{B}} \wedge s^{0,q_0} \wedge g^{0,1}$ (konfiguracja początkowa)

$\bigwedge_{i,j} \left(\bigvee_a t^{i,j,a} \wedge \bigwedge_{a \neq b} \neg(t^{i,j,a} \wedge t^{i,j,b}) \right) \wedge \dots$ (niesprzeczność)

$\bigvee_i s^{i,q_{\text{tak}}}$ (akceptacja)

$\bigwedge_{i,j} \bigwedge_{a,q} t^{i,j,a} \wedge s^{i,q} \wedge g^{i,j} \implies$ przejścia z (q,a):

$t^{i+1,j,a'} \wedge s^{i+1,q'} \wedge g^{i+1,j}$ $\vee$ $(q, a, q', a', \circlearrowright)$

$t^{i+1,j,a''} \wedge s^{i+1,q''} \wedge g^{i+1,j-1}$ $(q, a, q'', a'', \leftarrow)$

$\bigwedge_{i,j} \bigwedge_a t^{i,j,a} \wedge \neg g^{i,j} \implies t^{i+1,j,a}$

Problemy NP-zupełne (przykłady)

3-SAT, 3-kolorowalność

Cykl Hamiltona

Dane: graf skierowany G.

Wynik: czy G ma cykl Hamiltona ?

Problem plecakowy (szczególny przypadek)

Dane: zbiór liczb $\{n_1, \ldots, n_k\}$ i liczba m, reprezentowane binarnie.

Wynik: czy istnieje podzbiór $\{n_{i_1}, \ldots, n_{i_l}\}$ taki, że $n_{i_1} + \ldots + n_{i_l} = m$?

Pytanie: A gdyby liczby były reprezentowane unarnie ?

PCP z ograniczeniem

Dane: ciąg par słów $(w_1, v_1), \ldots, (w_n, v_n)$ i liczba k reprezentowana unarnie.

Wynik: czy istnieje niepusty ciąg $(i_1, \ldots, i_m)$, $m \leq k$, t. że

$w_{i_1} \ldots w_{i_m} = v_{i_1} \ldots v_{i_m}$?

Problemy NP-zupełne (przykłady)

Programowanie całkowitoliczbowe

Dane: układ równań liniowych U o całkowitych współczynnikach.

Wynik: czy U ma nieujemne całkowite rozwiązanie ?

Nierówność wyrażeń regularnych bez *

Dane: dwa wyrażenia regularne R, R' bez *.

Wynik: czy $L(R) \neq L(R')$?

np. $(a+b)(a+\varepsilon)b + abb \neq (a+\varepsilon)(a+b)b + bab$?

Problem stopu po n krokach

Dane: Niedeterministyczna maszyna Turinga $\mathcal{M}$ i słowo wejściowe w.

Wynik: czy $\mathcal{M}$ akceptuje w po co najwyżej $|w|$ krokach ?

co-NP-trudność i zupełność

co-NP = problemy, których dopełnienie jest w NP

Problem L jest *co-NP-trudny* jeśli każdy $K \in$ co-NP redukuje się wielomianowo do L.

Problem L jest *co-NP-zupełny* jeśli jest co-NP-trudny i należy do co-NP.

Fakt

Jeśli $K \leq_p L$ i K jest co-NP-trudny to L jest co-NP-trudny.

Problem tautologii zdaniowej

Dane: formuła zdaniowa ϕ.

Wynik: czy ϕ jest tautologią?

Wniosek

Problem tautologii zdaniowej jest co-NP-zupełny.

Problemy w NP, o których nie wiemy ani że są NP-zupełne, ani że są w P:

Izomorfizm grafów

Dane:	Dwa grafy G, H.
Wynik:	Czy G i H są izomorficzne?

Gra parzystości:

- graf skierowany, wierzchołek startowy
- każdy wierzchołek należy do jednego z graczy: Parzystego lub Nieparzystego
- wierzchołki etykietowane liczbami
- gramy do pierwszej powtórki (cyklu)
- zwycięzca określony przez parzystość największej liczby na cyklu

Kto wygrywa grę parzystości?

Dane:	gra parzystości.
Wynik:	czy Parzysty ma strategię wygrywającą?

Problem należy do NP $\cap$ co-NP.

Pierwszość

Dane: liczba naturalna $n \in \mathbb{N}$, reprezentowana binarnie.

Wynik: czy n jest liczbą pierwszą?

Twierdzenie (Agrawal, Kayal, Saxena 2004)

Pierwszość jest w P.

W następnym odcinku:

Pamięć wielomianowa i gramatyki dla maszyn Turinga