

Języki, automaty i obliczenia

Wykład 12: Nieobliczalność c.d.

Sławomir Lasota

Uniwersytet Warszawski

20 maja 2026

- 1 Gramatyki równoważne maszynom Turinga
- 2 Wszystkie własności semantyczne programów są nierozstrzygalne.
- 3 Funkcje nieobliczalne: pracowite bobry

- gramatyki typu 0
- automaty z wieloma stosami
- automaty z licznikami
- automaty z kolejką
- maszyny RAM (ang. *random access machines*) albo maszyny rejestrowe
- while-programy
- rachunek λ
- ...

typ	języki	automaty	gramatyki
typ 0	częściowo rozstrzygalne	maszyny Turinga	?
typ 1	kontekstowe	?	?
typ 2	bezkontekstowe	automaty ze stosem	bezkontekstowe
typ 3	regularne	automaty skończone	regularne

$$\mathcal{G} = (A, N, S, \alpha)$$

- A to skończony zbiór symboli końcowych (*terminalnych*)
- N to skończony zbiór symboli niekończących (*nieterminalnych*), $A \cap N = \emptyset$
- $S \in N$ to symbol początkowy
- $\alpha \subseteq (A \cup N)^+ \times (A \cup N)^*$ to skończony zbiór reguł przepisywania (*produkcji*)

Notacja

Reguły $(v, v') \in \alpha$ będziemy zapisywać $v \rightarrow_{\mathcal{G}} v'$ albo $v \rightarrow v'$.

Reguły rozszerzamy do relacji $\rightarrow_{\mathcal{G}}^* \subseteq (A \cup N)^* \times (A \cup N)^*$:

$$w \rightarrow_{\mathcal{G}}^* w' \iff \exists (v \rightarrow_{\mathcal{G}} v') \in \alpha, \exists u, t \in (A \cup N)^*, w = uv t, w' = uv' t$$

i domykalimy zwrotno-tranzytywnie: $w \rightarrow_{\mathcal{G}}^* w'$ wtw. gdy istnieje ciąg

$$w = v_0 \rightarrow_{\mathcal{G}} v_1 \rightarrow_{\mathcal{G}} \dots \rightarrow_{\mathcal{G}} v_n = w', \quad n \geq 0$$

zwany *wyprowadzeniem* (wywodem) w' z w (w gramatyce $\mathcal{G}$).

Język *generowany* przez gramatykę:

$$L(\mathcal{G}) = \{ w \in A^* : S \xrightarrow{*}_{\mathcal{G}} w \}$$

Przykład

Gramatyka $\mathcal{G}$:

$$A = \{a, b, c\}$$

$$S \longrightarrow \varepsilon$$

$$X \longrightarrow aBXc$$

$$Ba \longrightarrow aB$$

$$N = \{S, X, B\}$$

$$S \longrightarrow X$$

$$X \longrightarrow abc$$

$$Bb \longrightarrow bb$$

$$L(\mathcal{G}) = ?$$

$$B \longrightarrow b$$

Twierdzenie

Gramatyki $\equiv$ maszyny Turinga.

Dowód:

- gramatyka $\mathcal{G} \rightsquigarrow$ maszyna $\mathcal{M}$: maszyna symuluje gramatykę „wstecz”

$$v \twoheadrightarrow_{\mathcal{G}}^* w \iff qw \rightarrow_{\mathcal{M}}^* qv,$$

zaczyna w konfiguracji qw , akceptuje „w konfiguracji qS ”.

- maszyna $\mathcal{M}$ nad $\{0, 1\}$ z taśmą jednostronnie nieograniczoną $\rightsquigarrow$ gramatyka $\mathcal{G}$:

$$A = \{0, 1\} \quad N = Q \times \{0, 1, \mathbb{B}\} \cup \{\$, \mathbb{B}, S\}$$

Faza 1: $S \twoheadrightarrow_{\mathcal{G}}^* \$[q_{\text{tak}}, \mathbb{B}] \mathbb{B}^+$

Faza 2: $b[q', c] \rightarrow [q, a] c$ jeśli $(q, a, q', b, \rightarrow) \in \delta$

$[q', c] b \rightarrow c [q, a]$ jeśli $(q, a, q', b, \leftarrow) \in \delta$

$\$[q_0, a] \rightarrow a$

$a \mathbb{B} \rightarrow a$

poprawność: $w \in L(\mathcal{M}) \iff S \twoheadrightarrow_{\mathcal{G}}^* w$

Problem słów

Dane: gramatyka $\mathcal{G}$ i dwa słowa v, w
Wynik: czy $v \rightarrow_{\mathcal{G}}^* w$?

Wniosek

Problem słów jest nierozstrzygalny.

Dowód:

Redukujemy problem stopu do problemu słów:

funkcja obliczalna:	maszyna $\mathcal{M}$, słowo w	$\mapsto$	$\mathcal{G}$ i słowa S, w
poprawność:	$w \in L(\mathcal{M})$	$\iff$	$S \rightarrow_{\mathcal{G}}^* w$

Pytanie

Czy problem słów jest częściowo rozstrzygalny?

- 1 Gramatyki równoważne maszynom Turinga
- 2 Wszystkie własności semantyczne programów są nierozstrzygalne.
- 3 Funkcje nieobliczalne: pracowite bobry

Ustalmy A , np. $A = \{0, 1\}$. *Własność* języków to dowolny podzbiór $\mathcal{L} \subseteq \mathcal{P}(A^*)$.

Własność $\mathcal{L}$ jest *trywialna* jeśli

- albo $\mathcal{L}$ nie zawiera żadnego języka częściowo rozstrzygalnego,
- albo $\mathcal{L}$ zawiera je wszystkie.

Przykładowe własności nietrywialne:

- skończoność,
- ko-skończoność,
- rozstrzygalność,
- język zawiera wszystkie słowa długości parzystej.

Twierdzenie (Rice 1951)

*Dla każdej **nietrywialnej** własności $\mathcal{L}$, następujący problem jest nierozstrzygalny:*

<i>Dane:</i>	<i>maszyna Turinga $\mathcal{M}$</i>
<i>Wynik:</i>	<i>czy $L(\mathcal{M}) \in \mathcal{L}$?</i>

Każda nietrywialna własność semantyczna programów jest nierozstrzygalna.

Dowód:

B.u.o. założmy, że $\emptyset \notin \mathcal{L}$. Wybierzmy dowolną maszynę $\mathcal{M}_0$ t.ż. $L(\mathcal{M}_0) \in \mathcal{L}$.

Problem stopu na ε

Dane: maszyna Turinga $\mathcal{M}$
Wynik: czy $\varepsilon \in L(\mathcal{M})$?

$\leq$

Własność $\mathcal{L}$

Dane: maszyna Turinga $\mathcal{M}$
Wynik: czy $L(\mathcal{M}) \in \mathcal{L}$?

Funkcja obliczalna:

$\mathcal{M} \mapsto \mathcal{M}'$: najpierw sprawdź, czy $\varepsilon \in L(\mathcal{M})$, ignorując słowo wejściowe;
następnie uruchom $\mathcal{M}_0$ na słowie wejściowym.

Poprawność:

$$\varepsilon \in L(\mathcal{M}) \iff L(\mathcal{M}') \in \mathcal{L}$$

- Wszystkie semantyczne własności maszyn Turinga (np. problem pustości)
- Wiele problemów dla języków bezkontekstowych (np. uniwersalność)
- PCP: Problem odpowiedniości Posta
- Problem stopu dla modeli obliczeń równoważnych maszynom Turinga
- Problem słów dla gramatyk typu 0
- Problem słów dla grup skończenie prezentowanych
- Kafelkowanie nieskończonej płaszczyzny
- Prawdziwość zdań arytmetyki ($\mathbb{N}, +, \times$)
- 10-ty problem Hilberta: rozwiązywalność diofantycznych równań wielomianowych
- ...

- 1 Gramatyki równoważne maszynom Turinga
- 2 Wszystkie własności semantyczne programów są nierozstrzygalne.
- 3 Funkcje nieobliczalne: pracowite bobry

Ustalmy alfabet taśmowy $T = \{a, \mathbb{B}\}$.

Bóbr $\stackrel{\text{def}}{=}$ deterministyczna maszyna Turinga $\mathcal{M}$,

$$\delta : (Q - \{q_{\text{stop}}\}) \times T \rightarrow Q \times T \times \{\leftarrow, \circlearrowleft, \rightarrow\},$$

która **zatrzymuje się** na słowie pustym:

$$q_0 \varepsilon \xrightarrow{*}_{\mathcal{M}} u q_{\text{stop}} v \quad \text{dla pewnych } u, v \in T^*.$$

Bóbr zaczyna obliczenie z pustą taśmą, a jego **wynik** to **liczba kroków, po której się zatrzymuje.**

Bobra o najlepszym wyniku spośród bobrów o n stanach (nie licząc q_{stop}) nazwijmy **pracowitym**.

$BB(n)$ $\stackrel{\text{def}}{=}$ wynik pracowitego bobra o n stanach.

$BB(n) \stackrel{\text{def}}{=} \text{wynik pracowitego bobra o } n \text{ stanach.}$

Wynik bobra = liczba kroków, po której się zatrzymuje.

$$BB(1) = 1$$

$$BB(2) = 6$$

$$BB(3) = 21$$

$$BB(4) = 107$$

$$BB(5) = 47\,176\,870$$

$$(BB(5) \geq 47\,176\,870 \text{ rok temu})$$

$$BB(6) \geq t(t(t(t(2))))$$

$$t(n) = 2^{2^{2^{\dots^{2^2}}}} \Bigg\}_n$$

Twierdzenie (Radó 1962)

Funkcja $BB : \mathbb{N} \rightarrow \mathbb{N}$ jest nieobliczalna.

Dowód:

Gdyby była, to rozstrzygalny byłby problem stopu na pustym słowie wejściowym: dla danej maszyny $\mathcal{M}$ o n stanach oblicz $BB(n)$ a następnie symuluj $\leq BB(n)$ jej kroków.

Fakt

BB dominuje,^a dla wystarczająco dużych argumentów, każdą funkcję obliczalną.

^aDla każdej funkcji obliczalnej $f : \mathbb{N} \rightarrow \mathbb{N}$ istnieje $n_0 \in \mathbb{N}$ takie, że $f(n) < BB(n)$ dla $n \geq n_0$.

Dowód:

Niech $F : \mathbb{N} \rightarrow \mathbb{N}$ obliczalna. Wtedy obliczalna jest również funkcja $n \mapsto F(2n) + 1$, przez pewną całkowitą maszynę $\mathcal{M}$.

Niech N będzie dowolną liczbą $\geq$ liczby stanów $\mathcal{M}$.

Niech $\mathcal{M}_N$ będzie maszyną, która rozpoczynając z pustą taśmą wypisuje a^N i zatrzymuje się. Jej liczba stanów wynosi N . Złożenie maszyn

$$\mathcal{M}_N; \mathcal{M}$$

to maszyna o $\leq 2N$ stanach, która startując z pustej taśmy oblicza $F(2N) + 1$.

Zatem $F(2N) + 1 \leq BB(2N)$.

Analogicznie dla nieparzystych argumentów.

Fakt

Istnieje deterministyczna maszyna $\mathcal{M}$ o 748 stanach taka, że $\varepsilon \notin L(\mathcal{M})$, ale nie da się tego dowieść na gruncie teorii zbiorów.^a

^aZakładając niesprzeczność aksjomatów teorii zbiorów.

Wniosek

Wartość $BB(748)$ nie może być wyznaczona na gruncie teorii zbiorów.

Fakt

Wyznaczenie wartości $BB(27)$ pozwoliłoby rozwiązać hipotezę Goldbacha.

W następnym odcinku:

Złożoność czasowa i pamięciowa, klasa NP