

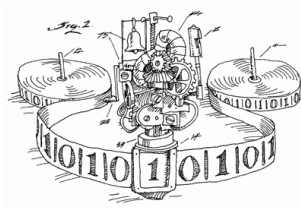
Języki, automaty i obliczenia

Wykład 10: Maszyny Turinga

Sławomir Lasota

Uniwersytet Warszawski

4 maja 2016



- 1 Maszyny Turinga
- 2 Deterministyczne maszyny Turinga
- 3 Warianty maszyn Turinga
- 4 Teza Churcha-Turinga: maszyny Turinga a obliczalność

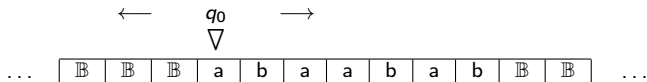
(Niedeterministyczna) *maszyna Turinga* $\mathcal{M} = (A, Q, q_0, F, T, \mathbb{B}, \delta)$

- A – alfabet wejściowy
- Q – zbiór stanów
- $q_0 \in Q$ – stan początkowy
- $F \subseteq Q$ – stany akceptujące (bez u.o. jeden stan akceptujący)
- T – alfabet taśmowy, $A \subseteq T$
- $\mathbb{B} \in T - A$ – symbol pusty (ang. blank)
- $\delta \subseteq Q \times T \times Q \times T \times \{\leftarrow, \circlearrowleft, \rightarrow\}$ – relacja przejścia

Co oznacza przejście?

$(q, a, q', a', k) \in \delta$: zmień stan z q na q' , czytaj a , **zapisz a'** , zmień pozycję wg. k

Konfiguracja początkowa maszyny $\mathcal{M}$ wygląda tak:



- $A = \{a, b\}$
- $Q = \{\text{start}, \text{start}_a, \text{start}_b, \text{ret}, \text{go}, \text{go}_a, \text{go}_b, \text{go}'_a, \text{go}'_b, \text{ret}', \text{check}, \text{ok}\}$
- $q_0 = \text{start}$
- $F = \{\text{ok}\}$
- $T = A \cup \{\mathbb{B}, \#\}$
- relacja przejścia (na następnym slajdzie)

$$\delta : Q \times T \rightarrow \mathcal{P}(Q \times T \times \{\leftarrow, \circlearrowleft, \rightarrow\})$$

Pytanie

Jaki język rozpoznaje ta maszyna?

Przykład (relacja przejścia)

	a	b	$\mathbb{B}$	$\#$
start	$(\text{start}_a, \#, \rightarrow)$	$(\text{start}_b, \#, \rightarrow)$	$(\text{ok}, \mathbb{B}, \circ)$	
start _a	$(\text{start}_a, a, \rightarrow)$ $(\text{ret}, \#, \leftarrow)$	$(\text{start}_a, b, \rightarrow)$	?	
start _b	$(\text{start}_b, a, \rightarrow)$	$(\text{start}_b, b, \rightarrow)$ $(\text{ret}, \#, \leftarrow)$	?	
ret	$(\text{ret}, a, \leftarrow)$	$(\text{ret}, b, \leftarrow)$		$(\text{go}, \#, \rightarrow)$
go	$(\text{go}_a, \#, \rightarrow)$	$(\text{go}_b, \#, \rightarrow)$		$(\text{check}, \#, \circ)$
go _a	$(\text{go}_a, a, \rightarrow)$	$(\text{go}_a, b, \rightarrow)$		$(\text{go}'_a, \#, \rightarrow)$
go _b	$(\text{go}_b, a, \rightarrow)$	$(\text{go}_b, b, \rightarrow)$		$(\text{go}'_b, \#, \rightarrow)$
go' _a	$(\text{ret}', \#, \leftarrow)$	?	?	$(\text{go}'_a, \#, \rightarrow)$
go' _b	?	$(\text{ret}', \#, \leftarrow)$	?	$(\text{go}'_b, \#, \rightarrow)$
ret'	$(\text{ret}, a, \leftarrow)$	$(\text{ret}, b, \leftarrow)$	$(\text{check}, \mathbb{B}, \rightarrow)$	$(\text{ret}', \#, \leftarrow)$
check	?	?	$(\text{ok}, \mathbb{B}, \circ)$	$(\text{check}, \#, \rightarrow)$

$$L(\mathcal{M}) = \{ ww : w \in A^* \}$$

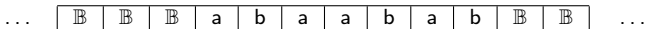
Pytanie

Jak zmodyfikować maszynę $\mathcal{M}$, aby obliczała funkcję:

$$w \mapsto ww?$$

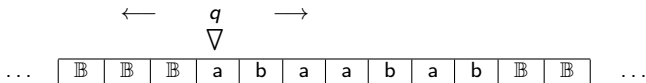
Relacja przejścia

$$\delta \subseteq (Q - \{\text{ok}\}) \times T \times Q \times T \times \{\leftarrow, \circ, \rightarrow\}$$



- Taśma jest nieskończona w obydwie strony.
- Zawartość taśmy reprezentuje zawartość pamięci maszyny: nie nieskończoną, ale dowolnie dużą skończoną.
- Bez u.o. możemy założyć, że maszyna nigdy nie zapisuje symbolu B; wtedy B oznacza „nieużywane” pozycje taśmy.
- Poza obszarem używanym przez maszynę, na wszystkich pozycjach taśmy jest symbol B.
- Nieistotne, na której pozycji taśmy zaczyna się obszar używany przez maszynę (przesunięcie tego obszaru w prawo lub w lewo nic istotnego nie zmienia).

Zapisując konfiguracje, pomijamy nieskończenie wiele symboli $\mathbb{B}$ poza obszarem odwiedzionym przez maszynę, i poza słowem wejściowym:



$(q, a) b a a b a b$ albo $q a b a a b a b$

Formalnie, *konfiguracja* maszyny $\mathcal{M}$ to

$$c = w q w' \in T^* Q T^*$$

Umowa

$$q a b a a b a b = \mathbb{B} q a b a a b a b = q a b a a b a b \mathbb{B} = \mathbb{B} \mathbb{B} q a b a a b a b \mathbb{B} = \dots$$

Konfiguracje początkowa: $c_0 = q_0 w$ ($w \in A^*$ to słowo wejściowe)

Konfiguracje akceptujące: $T^* F T^* = \{ w q w' : q \in F; w, w' \in T^* \}$

Przejścia pomiędzy konfiguracjami $c \rightarrow_{\mathcal{M}} c'$:

$(w, v \in T^*)$

- jeśli $(q, a, q', a', \circ) \in \delta$ to

$$w q a v \rightarrow_{\mathcal{M}} w q' a' v$$

- jeśli $(q, a, q', a', \rightarrow) \in \delta$ to

$$w q a v \rightarrow_{\mathcal{M}} w a' q' v$$

- jeśli $(q, a, q', a', \leftarrow) \in \delta$ to

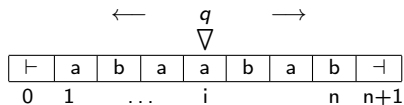
$$w b q a v \rightarrow_{\mathcal{M}} w q' b a' v$$

Umowa

$$q a b a a b a b = \mathbb{B} q a b a a b a b = q a b a a b a b \mathbb{B} = \mathbb{B} \mathbb{B} q a b a a b a b \mathbb{B} = \dots$$

Język rozpoznawany przez maszynę $\mathcal{M}$:

$$L(\mathcal{M}) = \{ w \in A^* : q_0 w \rightarrow_{\mathcal{M}}^* c \text{ dla jakiejś konfiguracji akceptującej } c \}$$



Automaty dwukierunkowe to... maszyny Turinga, które nie mogą pisać na taśmie.

Klasa języków rozpoznawanych przez maszyny Turinga:

- języki **częściowo** *rozstrzygalne*
- języki *rekurencyjnie przeliczalne*

Hierarchia Chomsky'ego:

- (typ 0) języki rekurencyjnie przeliczalne
- (typ 1) języki kontekstowe
- (typ 2) języki bezkontekstowe
- (typ 3) języki regularne

- 1 Maszyny Turinga
- 2 **Deterministyczne maszyny Turinga**
- 3 Warianty maszyn Turinga
- 4 Teza Churcha-Turinga: maszyny Turinga a obliczalność

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

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

$$\delta : (Q - \{\text{ok}, \text{nok}\}) \times T \rightarrow Q \times T \times \{\leftarrow, \circ, \rightarrow\}$$

Czy tę maszynę można zdeterminizować?

	a	b	$\mathbb{B}$	$\#$
start	$(\text{start}_a, \#, \rightarrow)$	$(\text{start}_b, \#, \rightarrow)$	$(\text{ok}, \mathbb{B}, \circ)$	
start _a	$(\text{start}_a, a, \rightarrow)$ $(\text{ret}, \#, \leftarrow)$	$(\text{start}_a, b, \rightarrow)$	?	
start _b	$(\text{start}_b, a, \rightarrow)$	$(\text{start}_b, b, \rightarrow)$ $(\text{ret}, \#, \leftarrow)$	?	
ret	$(\text{ret}, a, \leftarrow)$	$(\text{ret}, b, \leftarrow)$		$(\text{go}, \#, \rightarrow)$
go	$(\text{go}_a, \#, \rightarrow)$	$(\text{go}_b, \#, \rightarrow)$		$(\text{check}, \#, \circ)$
go _a	$(\text{go}_a, a, \rightarrow)$	$(\text{go}_a, b, \rightarrow)$		$(\text{go}'_a, \#, \rightarrow)$
go _b	$(\text{go}_b, a, \rightarrow)$	$(\text{go}_b, b, \rightarrow)$		$(\text{go}'_b, \#, \rightarrow)$
go' _a	$(\text{ret}', \#, \leftarrow)$	?	?	$(\text{go}'_a, \#, \rightarrow)$
go' _b	?	$(\text{ret}', \#, \leftarrow)$	?	$(\text{go}'_b, \#, \rightarrow)$
ret'	$(\text{ret}, a, \leftarrow)$	$(\text{ret}, b, \leftarrow)$	$(\text{check}, \mathbb{B}, \rightarrow)$	$(\text{ret}', \#, \leftarrow)$
check	?	?	$(\text{ok}, \mathbb{B}, \circ)$	$(\text{check}, \#, \rightarrow)$

Pytanie

Czy dla każdej maszyny Turinga istnieje równoważna maszyna deterministyczna?

- 1 Maszyny Turinga
- 2 Deterministyczne maszyny Turinga
- 3 Warianty maszyn Turinga
- 4 Teza Churcha-Turinga: maszyny Turinga a obliczalność

Maszyny z:

- taśmą jednostronnie nieskończoną
- wieloma taśmami
- „taśmą” wielowymiarową
- ...

Maszyny z taśmą jednostronnie nieskończoną

⊢	a	b	a	a	b	a	b	B	B
---	---	---	---	---	---	---	---	---	---

 ...

Pytanie

Czy maszyna z taśmą jednostronnie nieskończoną potrafi symulować maszynę z taśmą dwustronnie nieskończoną?

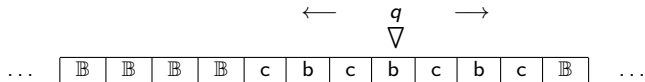
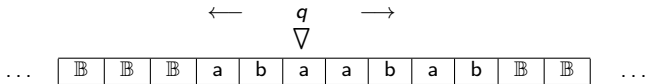
...

B	B	B	a	b	a	a	b	a	b	B	B
---	---	---	---	---	---	---	---	---	---	---	---

 ...

⊢	B	B	B	B	B	B	B		
	a	b	a	a	b	a	b	B	B

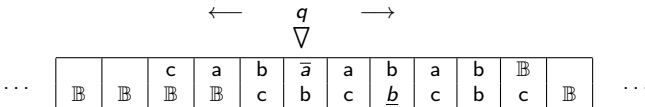
 ...



$$\delta \subseteq Q \times T^2 \times Q \times T^2 \times \{\leftarrow, \circ, \rightarrow\}^2$$

Pytanie

Czy maszyna jednotaśmowa potrafi symulować maszynę wielotaśmową?



Maszyzny z taśmą dwuwymiarową

...

␣	␣	␣	␣	␣	␣	␣	␣	␣	␣	␣	␣	
␣	␣	a	b	b	c	q	a	a	a	␣	␣	
␣	a	b	a	a	b	▽	a	a	b	␣	␣	
␣	␣	␣	a	b	a		b	a	b	␣	␣	...
␣	␣	␣	␣	c	b	c	b	c	b	c	␣	
␣	␣	␣	␣	␣	␣	␣	␣	␣	␣	␣	␣	

...

$$\delta \subseteq Q \times T \times Q \times T \times \{\leftarrow, \circ, \rightarrow, \uparrow, \downarrow\}$$

Pytanie

Czy maszyna z taśmą jednowymiarową potrafi symulować maszynę z taśmą dwuwymiarową?

⊢	a	#	a	b	b	c	b	a	b	b	#	a	a	a	c	b	...
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	-----

- gramatyki typu 0
- automaty wielostosowe
- maszyny (automaty) licznikowe
- automaty z kolejką
- ...
- maszyny RAM (ang. *random access machines*) albo maszyny rejestrowe
- rachunek λ
- ...

- 1 Maszyny Turinga
- 2 Deterministyczne maszyny Turinga
- 3 Warianty maszyn Turinga
- 4 Teza Churcha-Turinga: maszyny Turinga a obliczalność

Języki będziemy utożsamiać z *problemami decyzyjnymi* (*zadaniami obliczeniowymi*).

Przykład

Dane wejściowe: graf skierowany G

Wynik: rozstrzygnąć, czy G ma cykl Hamiltona?

Graf można opisać jako słowo nad $A = \{0, 1, \#\}$, np. 110#101#001

Ogólnie, język $L \subseteq A^*$ można utożsamiać z następującym zadaniem obliczeniowym:

Dane wejściowe: $w \in A^*$

Wynik: rozstrzygnąć, czy $w \in L$?

Teza Churcha-Turinga:

języki rozpoznawane
(problemy obliczane)
przez maszyny Turinga

=

problemy, dla których istnieje
efektywny algorytm, przy założeniu
nieograniczonych zasobów

albo:

maszyny Turinga

=

komputery

Wątpliwości:

- nieskończoność taśmy?
- alfabet taśmowy większy niż rozmiar dysku?
- liczba stanów większa niż rozmiar dysku?
- niedeterminizm?

Problem stopu

Dane: maszyna Turinga $\mathcal{M}$ nad alfabetem A i słowo $w \in A^*$

Wynik: czy $w \in L(\mathcal{M})$?

Maszynę $\mathcal{M}$ można opisać jako słowo

$$\text{kod}_{\mathcal{M}} \in \{0, 1, \#\}^*,$$

na przykład

00#000000#001000#000011#000##000100#001#010000#100#001##...

Nie ma istotnej różnicy między programem a daną...

Pytanie

Czy alfabet $\{0, 1, \#\}$ można zmniejszyć?

Bez u.o. możemy ograniczyć się do maszyn, których alfabet taśmowy $T = \{0, 1, \mathbb{B}\}$. Faktycznie, słowo nad alfabetem $T - \{\mathbb{B}\} = \{a, b, c, d\}$ można zapisać jako słowo nad $\{0, 1\}$, na przykład:

$$a b c a d = 1000 \quad 0100 \quad 0010 \quad 1000 \quad 0001$$

Język uniwersalny (problem stopu):

$$\begin{aligned} & \{ (\mathcal{M}, w) : w \in L(\mathcal{M}) \} \\ & \{ \text{kod}_{\mathcal{M}} \$ w : w \in L(\mathcal{M}) \} \subseteq \{0, 1, \#, \$\}^* \end{aligned}$$

Twierdzenie

Język uniwersalny jest rekurencyjnie przeliczalny.

Dowód:

Uniwersalna maszyna Turinga:

```
000100 $
00#000000#001000#000011#000##000100#001#010000#100#001## ...
... $ 001010101010101111101011
```

W następnym odcinku:

Obliczalność i nieobliczalność