

Języki formalne i automaty 30.X.2020

Skończone automaty deterministyczne SAD

$$M = (A, \Sigma, s_0, S_F, \pi),$$

gdzie:

- A jest alfabetem
- Σ jest zbiorem stanów automatu, $\Sigma = \{s_0, s_1, \dots, s_q\}$
- $s_0 \in \Sigma$ jest wyróżnionym stanem startowym
- $S_F \subseteq \Sigma$ jest zbiorem stanów końcowych (akceptujących)
- π jest częściową funkcją przejścia, $\pi : \Sigma \times A \dashrightarrow \Sigma$.

Skończone automaty niedeterministyczne SAN

$$M = (A, \Sigma, s_0, S_F, \pi),$$

inaczej określony jest mechanizm przejścia:

$$\pi \text{ jest relacją przejścia, } \pi(s, a) \subset \Sigma$$

określoną dodatkowo na pustym argumencie literowym, $\pi(s, \lambda) \subset \Sigma$. Można to zapisać jako $\pi : \Sigma \times (A \cup \{\lambda\}) \dashrightarrow \mathcal{P}(\Sigma)$

Przykłady ...(skrypt, str. 47–50)

Operacje na automatach

- uzupełnienie π do funkcji totalnej – dodanie stanu pułapkowego:

$$\Sigma' = \Sigma \cup \{s_p\} \quad \text{oraz rozszerzenie } \pi \quad \pi(s, a) := s_p$$

dla wszystkich par (s, a) , dla których π nie jest jeszcze określona (w szczególności $\pi(s_p, a) = s_p$).

- Zamiana SAN na równoważny SAD
- Optymalizacja SAD pod kątem liczby stanów.

Zamiana SAN na SAD

Przykłady... (p. nagrania MP4 na mojej stronie)

Ogólna konstrukcja $\hat{M} = (A, \hat{\Sigma}, \hat{s}_0, \hat{S}_F, \hat{\pi})$

- Tworzymy nowy zbiór stanów $\hat{\Sigma} = \mathcal{P}(\Sigma)$
- Określamy nową funkcję przejścia

$$\hat{\pi}(\hat{s}, a) = \bigcup_{s' \in \hat{s}} \{s \in \Sigma : s \in \pi(s', a)\}$$

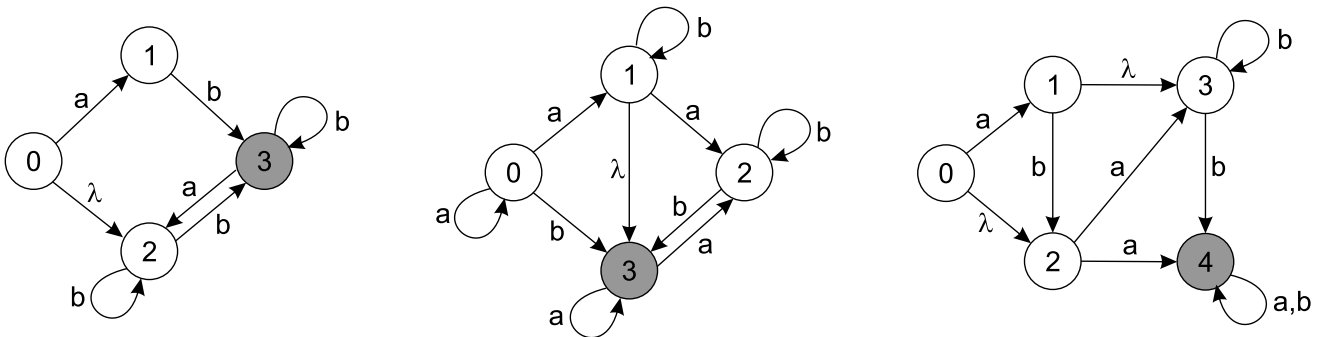
- Nowy stan początkowy

$$\hat{s}_0 = \{s_0\} \cup \{\text{wszystkie stany } s \text{ osiągalne z } s_0 \text{ ciągiem } \lambda\text{-przejęć}\}$$

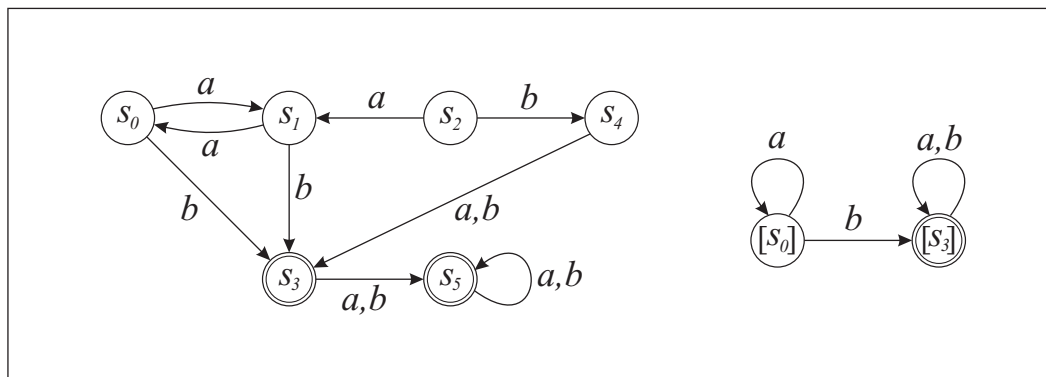
- Nowe stany końcowe: $\hat{s} \in \hat{\Sigma}$ takie że $\hat{s} \cap S_F \neq \emptyset$

Zadanie domowe (na 6.XI)

Przekształcić podane automaty niedeterministyczne na równoważne deterministyczne.



Optimalizacja automatów deterministycznych



(cdn...)