TUTORIAT LFA 5: ECHIVALENȚA MODULO UN LIMBAJ. MINIMIZAREA DFA.

RADU COSTACHE, MARIA PREDA

1. Breviar Teoretic

1.1. Echivalenţa modulo un limbaj.

Definiţie 1.1. *Spunem că două cuvinte v, w sunt echivalente modulo un limbaj L (notat $v \equiv_L w$) dacă:*

$$\forall z \in \Sigma^*, vz \in L \Leftrightarrow wz \in L$$

.

Observaţie 1.2. *Pentru a demonstra că două cuvinte w, v nu sunt echivalente modulo un limbaj L este suficient să gă sim un cuvânt x astfel încât $vx \in L$ şi $wx \notin L$ sau $vx \notin L$ şi $wx \in L$. Pentru a demonstra că cele două sunt echivalente, trebuie să arătă m că nu există un astfel de x.*

Exemplu 1.3. *Fie limbajul $L = \{a^n b^m \mid m, n \geq 5\}$. Avem că:*

- $a^7 b^7 \equiv_L a^5 b^6$

  Demonstraţie. Cu orice cuvânt format exclusiv din b-uri, cele două cuvinte rămân parte din L. Dacă x are şi alte simboluri, niciun cuvânt nu face parte din limbaj. □

- $a^5 b^2 \not\equiv_L a^5 b^3$

  Demonstraţie. Se verifică pentru $x = b^2$ □

1.2. Minimizarea DFA.

Teoremă 1.4. *Pentru orice DFA există şi este unic până la izomorfism un DFA cu număr minim de stări.*

Intuitiv: Vom căuta acele stă ri care sunt echivalente şi astfel să le transformă m în câte o singură stare. Ne punem urmă toarele întrabări:

- Când sunt două stă ri sunt echivalente?
  Similar echivalenţei pe stă ri vom verifica faptul că toate sufixele din acea stare au acelaşi comportament în cadrul automatului.
- Ce stă ri nu vor putea fi niciodată echivalente?
  O stare nefinală şi una finală nu vor putea niciodată să coincidă.
- Ce se întâmplă cu starea iniţială dacă e echivalentă cu o altă stare?
  Noua stare formată din combinarea celor două este stare iniţială.

Algoritmul lui Hopcroft. Vom menţine o mulţime de mulţimi de stă ri, reprezentând o partiţionare a mulţimii de stări. Fiecare element reprezintă o mulţime de stări potenţial echivalente (care la finalul algoritmului vor fi chiar stările echivalente). La fiecare pas vom repartitţiona fiecare mulţime, pentru acele stări care nu sunt echivalente (diferă la cel puţin o tranziţie). Vom începe prin a elimina stă rile inaccesibile (nu există niciun drum din starea iniţială că tre acestea). Apoi vom adăuga un *sink state*, pentru a face DFA-ul complet definit. Vom începe de la partiţionara în stări nefinale şi finale, deoarece doar despre acestea ştim iniţial că nu au cum să fie echivalente.

Algorithm 1 Algoritmul lui Hopcroft

```
Input: Q, F, δ, Σ                                      ▷ Componentele automatului
Output: P                                              ▷ Mulţimea de echivalenţe
  P ← {F, Q \ F}
  gasitDiviziune ← ⊤
  while gasitDiviziune is ⊤ do
      gasitDiviziune ← ⊥
      for R ∈ P do
          S ← random state from R
          R′ ← split(R, S)  ▷ Funcţia split returnează stările din R echivalente cu S la momentul curent
          if R ≠ R′ then
              P ← (P \ R) ∪ {R′, R \ R′}
              gasitDiviziune ← ⊤
              break
          end if
      end for
  end while
```

Observaţie 1.5. *Implementarea prezentată poate avea complexitatea de până la* $O(\Sigma \cdot n \log n)$ *dacă sunt utilizate structuri de date eficiente.*

Lema 1.6. *La pasul de split, dacă* R *trebuie partiţionat, nu contează care este starea* S *aleasă.*

Demonstraţie. Avem următoarele cazuri:

- R nu trebuie partiţionat. Înseamnă că $\forall S, S' \in R$ avem că $S \equiv S'$ în configuraţia curentă.
- R trebuie partiţionat. Înseamnă că $\forall S \in R, \exists S'$ astfel încât $S \not\equiv S'$ la configuraţia curentă.

□

2. Exerciţii

(1) Pentru limbajele date, spuneţi dacă perechile de cuvinte sunt sau nu echivalente conform L. În caz că sunt echivalente justificaţi pe scurt afirmaţia în caz contrar, daţi un cuvânt care să facă "diferenţa" intre cele doua cuvinte:

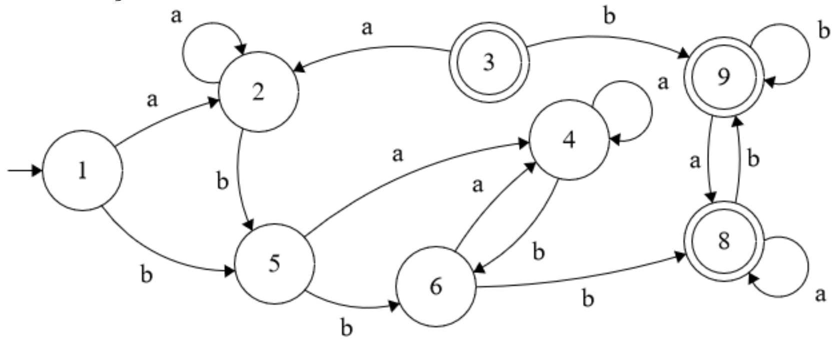
(a) $L = \{a^n b^n \mid n \geq 15\}$

$$\begin{array}{cc} a^4b^4 & \lambda \\ a^7b^5 & a^8b^6 \\ a^8b & a^7 \\ a^{10}b^{10} & a^5b^5 \\ a^{10}b^{10} & a^5b^6 \end{array}$$

(b) $L\{a^n b^{2n} \mid n \geq 3\}$

$$\begin{array}{cc} a^6b^8 & a^3b^2 \\ a^5b^{13} & \lambda \\ a^3b^5 & a^4b^8 \\ a^4 & a^5b^2 \\ a^2b^4 & a^3b^6 \end{array}$$

(2) (a) Minimizaţi automatul:



(b) Minimizaţi automatul: