Formális nyelvek és fordítóprogramok

3. Véges automaták és reguláris nyelvek I.

Bodó Zalán

Babeș–Bolyai Tudományegyetem
Matematika és Informatika Kar
Magyar Matematika és Informatika Intézet

Véges automaták
Elérhetetlen és nemproduktív állapotok
Elérhetetlen állapotok
Nemproduktív állapotok
NDVA szimulációja
NDVA $\rightarrow$ DVA
Ekvivalencia vizsgálata

Véges automaták

Def. NDVA

$A = (Q, \Sigma, E, I, F)$ rendezett ötös, ahol

- Q az automata **állapot**ainak halmaza (véges, nem üres)
- Σ a **bemeneti ábécé**
- E az **átmenetek** vagy **élek** halmaza, $E \subseteq Q \times \Sigma \times Q$
- $I \subseteq Q$ a **kezőállapotok** halmaza
- $F \subseteq Q$ a **végállapotok** halmaza

Def. DVA

Ha igazak a fentiek, és

- $|I| = 1$
- $|\delta(q, a)| \leq 1, \forall q \in Q, a \in \Sigma$, ahol $\delta : Q \times \Sigma \to \mathcal{P}(Q)$, $\delta(q, a) = \{p \in Q | (q, a, p) \in E\}$

Véges automaták

Elérhetetlen és nemproduktív állapotok

Elérhetetlen állapotok

Nemproduktív állapotok

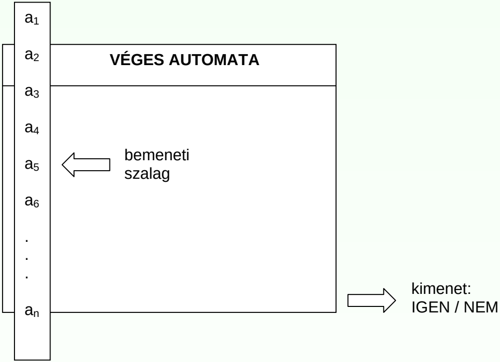
NDVA szimulációja

NDVA $\to$ DVA

Ekvivalencia vizsgálata

Def. Teljes DVA

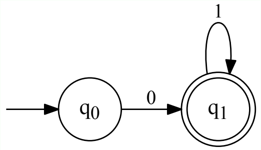
Ha igazak a *NDVA* definiciójában tett kijelentések és $|\delta(q, a)| = 1$, $\forall q \in Q, a \in \Sigma$.



ábra: A véges automata sematikus ábrázolása

Példák

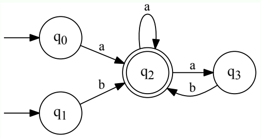
1. Példa – DVA



$Q = \{q_0, q_1\}$, $\Sigma = \{0, 1\}$,
$I = \{q_0\}$, $F = \{q_1\}$

δ	0	1
q_0	$\{q_1\}$	$\emptyset$
q_1	$\emptyset$	$\{q_1\}$

2. Példa – NDVA



$Q = \{q_0, q_1, q_2, q_3\}$, $\Sigma = \{a, b\}$,
$I = \{q_0, q_1\}$, $F = \{q_2\}$

δ	a	b
q_0	$\{q_2\}$	$\emptyset$
q_1	$\emptyset$	$\{q_2\}$
q_2	$\{q_2, q_3\}$	$\emptyset$
q_3	$\emptyset$	$\{q_2\}$

Def. Séta

$$(q_0, a_1, q_1), (q_1, a_2, q_2), \ldots, (q_{n-1}, a_n, q_n)$$

vagy

$$q_0 \xrightarrow{a_1} q_1 \xrightarrow{a_2} q_2 \xrightarrow{a_3} \ldots \xrightarrow{a_n} q_n$$

- ha $w = a_1 a_2 \ldots a_n$, akkor röviden: $q_0 \xrightarrow{w} q_n$
- **üres séta**: ha $n = 0$
- **produktív séta**: ha q_0 kezdőállapot és q_n végállapot
- azt mondjuk, hogy **egy véges automata felismer egy szót**, ha a szó egy **produktív séta címkéje**

Def. Véges automata által felismert nyelv

$$L(A) = \{w \in \Sigma^* \mid \exists p \xrightarrow{w} q, \text{ ú,h. } p \in I, q \in F\}$$

Elérhetetlen és nemproduktív állapotok

Elérhetetlen állapotok

Def.

Elérhetetlen állapot (p): ha a kezőállapotok egyikéből sem lehet eljutni ebbe az állapotba, semmilyen bemeneti szó hatására, vagyis $\forall q \in I, \forall w \in \Sigma^*, \nexists q \xrightarrow{w} p$.

ALG 1 Elérhetetlen állapotok kiküszöbölése

```
U_0 = I
i = 0
repeat
    i++
    U_i = U_{i-1}
    for minden q ∈ U_{i-1} do
        for minden a ∈ Σ do U_i = U_i ∪ δ(q, a)
        end for
    end for
until U_i == U_{i-1}
return U_i (elérhető állapotok)
```


Véges automaták
Elérhetetlen és nemproduktív állapotok
Elérhetetlen állapotok
Nemproduktív állapotok
NDVA szimulációja
NDVA → DVA
Ekvivalencia vizsgálata

Nemproduktív állapotok

Def.

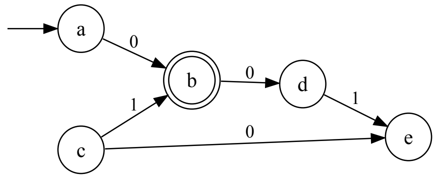
Nemproduktív állapot (p): ha az állapotból semmilyen szóra nem lehet végállapotba jutni, vagyis $\forall q \in F, \forall w \in \Sigma^*$, $\nexists p \xrightarrow{w} q$.

ALG 2 Nemproduktív állapotok kiküszöbölése

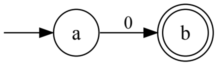
```
U_0 = F
i = 0
repeat
    i++
    U_i = U_{i-1}
    for minden q ∈ U_{i-1} do
        for minden a ∈ Σ do U_i = U_i ∪ δ^{-1}(q, a)
        end for
    end for
until U_i == U_{i-1}
return U_i (produktív állapotok)
```


Véges automaták
Elérhetetlen és nemproduktív állapotok
Elérhetetlen állapotok
Nemproduktív állapotok
NDVA szimulációja
NDVA → DVA
Ekvivalencia vizsgálata

3. Példa – Elérhetetlen és nemproduktív állapotok kizárása



Elérhetetlen állapotok: c
Nemproduktív állapotok: d, e
Az állapotok kizárása után kapott automata:



NDVA szimulációja

Rekurzív módszer

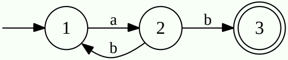
Feladat: Adott egy NDVA, $(Q, \Sigma, \delta, I, F)$ és egy $w = w_1 w_2 \ldots w_n$ bemeneti szó, $w_i \in \Sigma, i = 1, \ldots, n$. **Kérdés**: Felismeri-e az automata a bemeneti karaktersorozatot?

ALG 3 Rekurzív módszer

procedure $\mathrm{NDVA}(q,\ w,\ i)$
    if $i > |w|$ **then**
        if $q \in F$ **then**
            return IGEN
        else
            return
        end if
    else
        for $\forall p \in \Delta(q, w_i)$ **do**
            NDVA$(p,\ w,\ i + 1)$
        end for
    end if
end procedure

Egyelőre elegendő úgy vennünk, hogy $\Delta(q, a) = \delta(q, a)$, mivel még nem volt szó ε-lépéses automatákról.
(Helyesen $\Delta(q, a) =$ az összes olyan állapot, amely elérhető q-ból ε-lépesekkel, egy a szimbólummal, majd megint ε-lépésekkel.)

4. Példa



$w = ab$
NVA(1, ab, 1)
  NVA(2, ab, 2)
    NVA(1, ab, 3) $\rightarrow$ return
    NVA(3, ab, 3) $\rightarrow$ IGEN

Véges automaták
Elérhetetlen és nemproduktív állapotok
Elérhetetlen állapotok
Nemproduktív állapotok
NDVA szimulációja
NDVA $\rightarrow$ DVA
Ekvivalencia vizsgálata

Véges automaták
Elérhetetlen és nemproduktív állapotok
Elérhetetlen állapotok
Nemproduktív állapotok
NDVA szimulációja
NDVA $\to$ DVA
Ekvivalencia vizsgálata

Listás módszer

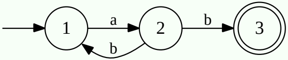
Lista: $S = (i_1, i_2, \ldots, i_{|Q|})$
$S^{(0)} =$ a kezdőállapotoknak megfelelő helyeken 1, a többi index alatt 0.

ALG 4 Listás módszer

for $i = 1, 2, \ldots, n$ **do**
  $S^{(i)} = (0, 0, \ldots, 0)$
  for $j = 1, 2, \ldots, |Q|$ **do**
    if $S_j^{(i-1)} = 1$ **then**
      $A = \Delta(q_j, w_i)$
      for $p = q_k \in A$ **do**
        $S_k^{(i)} = 1$
      end for
    end if
  end for
end for

Ha valamely végső állapotnak megfelelő elem értéke 1, akkor az automata felismerte a bemeneti szót.

5. Példa



$w = abab$
$S^{(0)} = (1,0,0) \to S^{(1)} = (0,1,0) \to S^{(2)} = (1,0,1) \to$
$S^{(3)} = (0,1,0) \to S^{(4)} = (1,0,1) \Rightarrow$ felismerte!

Véges automaták
Elérhetetlen és nemproduktív állapotok
Elérhetetlen állapotok
Nemproduktív állapotok
NDVA szimulációja
NDVA $\to$ DVA
Ekvivalencia vizsgálata

Mátrixos módszer

A bemeneti ábécé minden egyes $c \in \Sigma$ szimbólumához felépítünk egy M_c mátrixot a következőképpen:

$$M_c[i,j] = \begin{cases} 1, & \text{ha } q_j \in \Delta(q_i, c) \\ 0, & \text{különben} \end{cases}$$

Egy $w = w_1 w_2 \dots w_n$ bemeneti szóra kiszámoljuk az

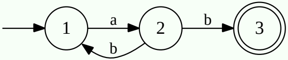
$$M_{w_1} \otimes M_{w_2} \otimes \dots \otimes M_{w_n}$$

„szorzatot". A $\otimes$ jellel jelölt művelet hasonló a mátrixok szorzásához, azzal a különbséggel, hogy a szorzás és összeadás helyett a következő műveleteket végezzük:

- szorzás $\rightarrow$ AND (logikai „és")
- összeadás $\rightarrow$ OR (logikai „vagy")

A végső mátrixban a kezdőállapot(ok)nak megfelelő sort kell megvizsgálnunk. Ha ezen sorok valamelyikének egy végállapotnak megfelelő oszlopában 1 található, akkor az automata felismerte a szót.

6. Példa



$w = ab$

$$M_a = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} ; M_b = \begin{pmatrix} 0 & 0 & 0 \\ 1 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix}$$

$$M_a \otimes M_b = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \otimes \begin{pmatrix} 0 & 0 & 0 \\ 1 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix} = \begin{pmatrix} \mathbf{1} & \mathbf{0} & \mathbf{1} \\ 0 & 0 & \mathbf{0} \\ 0 & 0 & \mathbf{0} \end{pmatrix}$$

$\Rightarrow$ felismerte a szót, mert $(M_a \otimes M_b)_{1,3} = 1$

NDVA $\rightarrow$ DVA

- a NDVA szimulációja/kiértékelése sokkal nagyobb bonyolultságú mint a DVA szimulációja
- átalakítás: NDVA $\rightarrow$ DVA:
  - megnöveljük az állapotok számát: képezzük az összes nemüres állapothalmazt ($2^{|Q|} - 1$)
  - az átmeneteket átírjuk az új állapotok függvényében – így biztosítjuk a két automata *ekvivalenciáját*: $\delta(q', a) = \bigcup_{p \in q'} \delta(p, a)$, ahol $q' \in \mathcal{P}(Q) \setminus \emptyset$
  - az új végállapotok azok lesznek, amelyekben legalább **egy** *régi* végállapot szerepel
  - kezőállapot: marad
  - több kezdőállapot esetén: $q_1, q_2, \ldots, q_k \in I \Rightarrow$ az új kezdőállapot: $\{q_1, \ldots, q_k\}$
- **Jelölések:**
  - $A = (Q, \Sigma, E, I, F) \rightarrow \overline{A} = (\overline{Q}, \Sigma, \overline{E}, \overline{I}, \overline{F})$
  - M átmenettáblázat
  - $BENNEVAN(\overline{q}, \overline{Q})$ = IGAZ, ha $\overline{q}$ benne van a $\overline{Q}$ halmazban
  - $\Sigma = \{a_1, a_2, \ldots, a_m\}$

Véges automaták
Elérhetetlen és nemproduktív állapotok
Elérhetetlen állapotok
Nemproduktív állapotok
NDVA szimulációja
NDVA $\rightarrow$ DVA
Ekvivalencia vizsgálata

ALG 5 NDVA $\to$ DVA

$\overline{q}_0 = I$
$\overline{Q} = \{\overline{q}_0\}$
$i = 0$
$k = 0$
repeat
    for $j = 1, 2, \ldots, m$ **do**
        $\overline{q} = \bigcup_{p \in \overline{q}_i} \delta(p, a_j)$
        if $\overline{q} \neq \emptyset$ **then**
            if $BENNEVAN(\overline{q}, \overline{Q})$ **then**
                $M[i, j] = \{\overline{q}\}$
            else
                $k = k + 1$
                $\overline{q}_k = \overline{q}$
                $M[i, j] = \{\overline{q}_k\}$
                $\overline{Q} = \overline{Q} \cup \{\overline{q}_k\}$
            end if
        else
            $M[i, j] = \emptyset$
        end if
        $i = i + 1$
    end for
until i=k+1
return $\overline{A}$ automata M átmenettáblázata

Véges automaták

Elérhetetlen és nemproduktív állapotok

Elérhetetlen állapotok

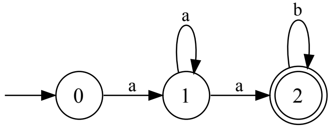
Nemproduktív állapotok

NDVA szimulációja

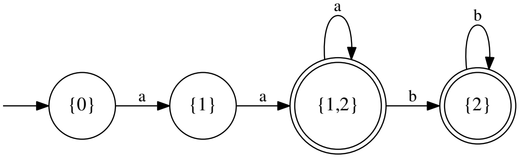
NDVA $\rightarrow$ DVA

Ekvivalencia vizsgálata

7. Példa



M	a	b
$\rightarrow \{0\}$	$\{1\}$	$\emptyset$
$\{1\}$	$\{1,2\}$	$\emptyset$
$\underline{\{1,2\}}$	$\{1,2\}$	$\{2\}$
$\underline{\{2\}}$	$\emptyset$	$\{2\}$



Megjegyzés: az átmenettáblázatban $\rightarrow$ jelöli a kezdőállapotot, a végállapotokat pedig aláhúztuk.

Ekvivalencia vizsgálata

Def. Automaták ekvivalenciája

Az A és A' automata egymással ekvivalens, ha az automaták által felismert nyelvek megegyeznek, $L(A) = L(A')$.

- csak teljes, determinisztikus automatákkal foglalkozunk (az algoritmus kevés módosítással működik nem teljes determinisztikus automatákra is)
- minden NDVA általakítható DVA-vá a korábban látott módszerrel
- minden DVA átalakítható teljessé a köv. egyszerű művelettel: behozunk egy új állapotot (nem kezdő-, nem vég-) és minden csúcsból a „hiányzó” szimbólummal egy átmenetet húzunk ebbe az állapotba
- **az alapötlet**: a sétákat vizsgáljuk az automatákban párhuzamosan; ha mindig egyszerre érünk végállapotba, akkor az automaták ekvivalensek, ellenkező esetben nem

Véges automaták

Elérhetetlen és nemproduktív állapotok

Elérhetetlen állapotok

Nemproduktív állapotok

NDVA szimulációja

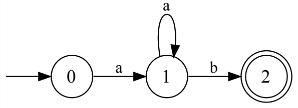
NDVA $\rightarrow$ DVA

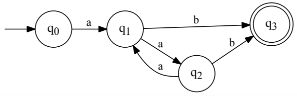
Ekvivalencia vizsgálata

ALG 6 Automaták ekvivalenciája

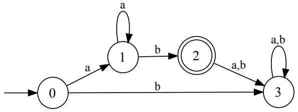
kiindulási állapotpár: (q_0, q_0')
if q_0 és q_0' valamelyike végállapot, míg a másik nem **then**
  return NEM
end if
$i = 0$
repeat
  $i = i + 1$
  Legyen (q, q') a táblázat i-edik sorának első oszlopában szereplő állapotpár
  for minden $a \in \Sigma$ betűre **do**
    A táblázat i-edik sorának a oszlopába a $(\delta(q, a), \delta(q', a))$ állapotpár kerül
    if $\delta(q, a)$ és $\delta(q', a)$ valamelyike végállapot, míg a másik nem **then**
      return NEM
    else
      Írjuk be a $(\delta(q, a), \delta(q', a))$ állapotpárt a táblázat következő üres sorának első oszlopába, feltéve, hogy még nem szerepel ott
    end if
  end for
until nincs új elempár
return IGEN

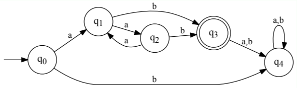
8. Példa





Átalakítva:





	a	b
$(0, q_0)$	$(1, q_1)$	$(3, q_4)$
$(1, q_1)$	$(1, q_2)$	$(2, q_3)$
$(3, q_4)$	$(3, q_4)$	$(3, q_4)$
$(1, q_2)$	$(1, q_1)$	$(2, q_3)$
$(2, q_3)$	$(3, q_4)$	$(3, q_4)$

$\Rightarrow$ ekvivalensek!