A számításelmélet alapjai I.

5. előadás

előadó: Tichler Krisztián
ktichler@inf.elte.hu

Véges automata determinizálása

Tétel

Minden $A = \langle Q, T, \delta, Q_0, F \rangle$ véges nemdeterminisztikus automatához megkonstruálható egy $A' = \langle Q', T, \delta', q_0', F' \rangle$ véges determinisztikus automata úgy, hogy $L(A) = L(A')$ teljesül.

Véges automata determinizálása

Tétel

Minden $A = \langle Q, T, \delta, Q_0, F \rangle$ véges nemdeterminisztikus automatához megkonstruálható egy $A' = \langle Q', T, \delta', q_0', F' \rangle$ véges determinisztikus automata úgy, hogy $L(A) = L(A')$ teljesül.

Bizonyítás:

A konstrukció:

$Q' := \mathcal{P}(Q), \quad q_0' := Q_0, \quad F' := \{q' \in Q' \mid q' \cap F \neq \emptyset\},$

$$\delta'(q', a) := \bigcup_{q \in q'} \delta(q, a).$$

Véges automata determinizálása

Lemma 1

Minden $p, q \in Q$, $q' \in Q'$ és $u, v \in T^*$ esetén, ha

$$qu \Rightarrow_A^* pv \text{ és } q \in q',$$

akkor van olyan $p' \in Q'$, hogy

$$q'u \Rightarrow_{A'}^* p'v \text{ és } p \in p'.$$

Lemma 1 bizonyítása:

Az állítást a $qu \Rightarrow^* pv$ redukcióban szereplő lépések száma szerinti indukcióval bizonyítjuk.

Véges automata determinizálása

Lemma 1

Minden $p, q \in Q$, $q' \in Q'$ és $u, v \in T^*$ esetén, ha

$$qu \Rightarrow_A^* pv \text{ és } q \in q',$$

akkor van olyan $p' \in Q'$, hogy

$$q'u \Rightarrow_{A'}^* p'v \text{ és } p \in p'.$$

Lemma 1 bizonyítása:

Az állítást a $qu \Rightarrow^* pv$ redukcióban szereplő lépések száma szerinti indukcióval bizonyítjuk.

Nulla számú lépés esetében az állítás triviálisan fennáll.

Véges automata determinizálása

Lemma 1

Minden $p, q \in Q$, $q' \in Q'$ és $u, v \in T^*$ esetén, ha

$$qu \Rightarrow_A^* pv \text{ és } q \in q',$$

akkor van olyan $p' \in Q'$, hogy

$$q'u \Rightarrow_{A'}^* p'v \text{ és } p \in p'.$$

Lemma 1 bizonyítása:

Az állítást a $qu \Rightarrow^* pv$ redukcióban szereplő lépések száma szerinti indukcióval bizonyítjuk.

Nulla számú lépés esetében az állítás triviálisan fennáll.

Indukciós feltevés: az állítás teljesül n redukciós lépés esetén ($n \geq 0$).

Véges automata determinizálása

Lemma 1

Minden $p, q \in Q$, $q' \in Q'$ és $u, v \in T^*$ esetén, ha

$$qu \Rightarrow_A^* pv \text{ és } q \in q',$$

akkor van olyan $p' \in Q'$, hogy

$$q'u \Rightarrow_{A'}^* p'v \text{ és } p \in p'.$$

Lemma 1 bizonyítása:

Az állítást a $qu \Rightarrow^* pv$ redukcióban szereplő lépések száma szerinti indukcióval bizonyítjuk.

Nulla számú lépés esetében az állítás triviálisan fennáll.

Indukciós feltevés: az állítás teljesül n redukciós lépés esetén ($n \geq 0$).

Véges automata determinizálása

Álljon a $qu \Rightarrow_A^* pv$ redukció $n+1$ lépésből. Akkor valamely $q_1 \in Q$ és $u_1 \in T^*$ -ra

$$qu \Rightarrow_A q_1 u_1 \Rightarrow_A^* pv$$

teljesül. Így van olyan $a \in T$, amelyre $u = au_1$ és $q_1 \in \delta(q, a)$.

Véges automata determinizálása

Álljon a $qu \Rightarrow_A^* pv$ redukció $n+1$ lépésből. Akkor valamely $q_1 \in Q$ és $u_1 \in T^*$ -ra

$$qu \Rightarrow_A q_1 u_1 \Rightarrow_A^* pv$$

teljesül. Így van olyan $a \in T$, amelyre $u = au_1$ és $q_1 \in \delta(q, a)$. $q \in q'$ esetén δ' definíciójából $\delta(q, a) \subseteq \delta'(q', a)$ adódik.

Véges automata determinizálása

Álljon a $qu \Rightarrow_A^* pv$ redukció $n+1$ lépésből. Akkor valamely $q_1 \in Q$ és $u_1 \in T^*$ -ra

$$qu \Rightarrow_A q_1 u_1 \Rightarrow_A^* pv$$

teljesül. Így van olyan $a \in T$, amelyre $u = au_1$ és $q_1 \in \delta(q,a)$.
$q \in q'$ esetén δ' definíciójából $\delta(q,a) \subseteq \delta'(q',a)$ adódik.
$q_1' := \delta'(q',a)$-nek, ahonnan

$$q'u \Rightarrow_{A'} q_1' u_1 \tag{1}$$

és $q_1 \in \delta(q,a) \subseteq \delta'(q',a) = q_1'$, azaz $q_1 \in q_1'$ adódik.

Véges automata determinizálása

Álljon a $qu \Rightarrow_A^* pv$ redukció $n+1$ lépésből. Akkor valamely $q_1 \in Q$ és $u_1 \in T^*$ -ra

$$qu \Rightarrow_A q_1 u_1 \Rightarrow_A^* pv$$

teljesül. Így van olyan $a \in T$, amelyre $u = au_1$ és $q_1 \in \delta(q,a)$.
$q \in q'$ esetén δ' definíciójából $\delta(q,a) \subseteq \delta'(q',a)$ adódik.
$q_1' := \delta'(q',a)$-nek, ahonnan

$$q'u \Rightarrow_{A'} q_1' u_1 \tag{1}$$

és $q_1 \in \delta(q,a) \subseteq \delta'(q',a) = q_1'$, azaz $q_1 \in q_1'$ adódik.
Alkalmazhatjuk az indukciós feltevést, tehát valamely $p' \in Q'$-re

Véges automata determinizálása

Álljon a $qu \Rightarrow_A^* pv$ redukció $n+1$ lépésből. Akkor valamely $q_1 \in Q$ és $u_1 \in T^*$ -ra

$$qu \Rightarrow_A q_1 u_1 \Rightarrow_A^* pv$$

teljesül. Így van olyan $a \in T$, amelyre $u = au_1$ és $q_1 \in \delta(q,a)$.
$q \in q'$ esetén δ' definíciójából $\delta(q,a) \subseteq \delta'(q',a)$ adódik.
$q_1' := \delta'(q',a)$-nek, ahonnan

$$q'u \Rightarrow_{A'} q_1' u_1 \qquad (1)$$

és $q_1 \in \delta(q,a) \subseteq \delta'(q',a) = q_1'$, azaz $q_1 \in q_1'$ adódik.
Alkalmazhatjuk az indukciós feltevést, tehát valamely $p' \in Q'$-re

$$q_1' u_1 \Rightarrow_{A'}^* p'v \text{ és } p \in p'. \qquad (2)$$

Véges automata determinizálása

Álljon a $qu \Rightarrow_A^* pv$ redukció $n+1$ lépésből. Akkor valamely $q_1 \in Q$ és $u_1 \in T^*$ -ra

$$qu \Rightarrow_A q_1 u_1 \Rightarrow_A^* pv$$

teljesül. Így van olyan $a \in T$, amelyre $u = au_1$ és $q_1 \in \delta(q, a)$.
$q \in q'$ esetén δ' definíciójából $\delta(q, a) \subseteq \delta'(q', a)$ adódik.
$q_1' := \delta'(q', a)$-nek, ahonnan

$$q'u \Rightarrow_{A'} q_1' u_1 \tag{1}$$

és $q_1 \in \delta(q, a) \subseteq \delta'(q', a) = q_1'$, azaz $q_1 \in q_1'$ adódik.
Alkalmazhatjuk az indukciós feltevést, tehát valamely $p' \in Q'$-re

$$q_1' u_1 \Rightarrow_{A'}^* p'v \text{ és } p \in p'. \tag{2}$$

(1) és (2) alapján Lemma 1 teljesül.

Véges automata determinizálása

Álljon a $qu \Rightarrow_A^* pv$ redukció $n+1$ lépésből. Akkor valamely $q_1 \in Q$ és $u_1 \in T^*$ -ra

$$qu \Rightarrow_A q_1 u_1 \Rightarrow_A^* pv$$

teljesül. Így van olyan $a \in T$, amelyre $u = au_1$ és $q_1 \in \delta(q, a)$.
$q \in q'$ esetén δ' definíciójából $\delta(q, a) \subseteq \delta'(q', a)$ adódik.
$q_1' := \delta'(q', a)$-nek, ahonnan

$$q'u \Rightarrow_{A'} q_1' u_1 \tag{1}$$

és $q_1 \in \delta(q, a) \subseteq \delta'(q', a) = q_1'$, azaz $q_1 \in q_1'$ adódik.
Alkalmazhatjuk az indukciós feltevést, tehát valamely $p' \in Q'$-re

$$q_1' u_1 \Rightarrow_{A'}^* p'v \text{ és } p \in p'. \tag{2}$$

(1) és (2) alapján Lemma 1 teljesül.

Véges automata determinizálása

Legyen $u \in L(A)$, azaz teljesüljön $q_0 u \Rightarrow_A^* p$ valamely $q_0 \in Q_0$-ra és legyen $p \in F$.

Véges automata determinizálása

Legyen $u \in L(A)$, azaz teljesüljön $q_0 u \Rightarrow_A^* p$ valamely $q_0 \in Q_0$-ra és legyen $p \in F$.

Ekkor Lemma 1 alapján valamely p'-re fennáll, hogy $q_0' u \Rightarrow_{A'}^* p'$ és $p \in p'$.

Véges automata determinizálása

Legyen $u \in L(A)$, azaz teljesüljön $q_0 u \Rightarrow_A^* p$ valamely $q_0 \in Q_0$-ra és legyen $p \in F$.

Ekkor Lemma 1 alapján valamely p'-re fennáll, hogy $q_0' u \Rightarrow_{A'}^* p'$ és $p \in p'$.

Az F' definíciója alapján $p \in p'$ -ből és $p \in F$-ből az következik, hogy $p' \in F'$, ahonnan $L(A) \subseteq L(A')$.

Véges automata determinizálása

Legyen $u \in L(A)$, azaz teljesüljön $q_0 u \Rightarrow_A^* p$ valamely $q_0 \in Q_0$-ra és legyen $p \in F$.

Ekkor Lemma 1 alapján valamely p'-re fennáll, hogy $q_0' u \Rightarrow_{A'}^* p'$ és $p \in p'$.

Az F' definíciója alapján $p \in p'$ -ből és $p \in F$-ből az következik, hogy $p' \in F'$, ahonnan $L(A) \subseteq L(A')$.

Lemma 2

Minden $p', q' \in Q'$, $p \in Q$ és $u, v \in T^*$ esetén, ha

$$q' u \Rightarrow_{A'}^* p' v \text{ és } p \in p',$$

akkor van olyan $q \in Q$, hogy

$$qu \Rightarrow_A^* pv \text{ és } q \in q'.$$

Véges automata determinizálása

Lemma 2 bizonyítása: A lépések száma szerinti indukcióval történik.

Véges automata determinizálása

Lemma 2 bizonyítása: A lépések száma szerinti indukcióval történik.

Nulla számú lépés esetén az állítás triviális.

Véges automata determinizálása

Lemma 2 bizonyítása: A lépések száma szerinti indukcióval történik.

Nulla számú lépés esetén az állítás triviális.

Tegyük fel, hogy az állítás igaz n lépésre, ahol $n \geq 0$, és álljon a $q'u \Rightarrow^*_{A'} p'v$ redukció $n+1$ lépésből.

Véges automata determinizálása

Lemma 2 bizonyítása: A lépések száma szerinti indukcióval történik.

Nulla számú lépés esetén az állítás triviális.

Tegyük fel, hogy az állítás igaz n lépésre, ahol $n \geq 0$, és álljon a $q'u \Rightarrow^*_{A'} p'v$ redukció $n+1$ lépésből.

Akkor $q'u \Rightarrow^*_{A'} p'_1 v_1 \Rightarrow_{A'} p'v$, ahol $v_1 = av$ valamely $a \in T$-re és $p'_1 \in Q'$.

Véges automata determinizálása

Lemma 2 bizonyítása: A lépések száma szerinti indukcióval történik.

Nulla számú lépés esetén az állítás triviális.

Tegyük fel, hogy az állítás igaz n lépésre, ahol $n \geq 0$, és álljon a $q'u \Rightarrow^*_{A'} p'v$ redukció $n+1$ lépésből.

Akkor $q'u \Rightarrow^*_{A'} p'_1 v_1 \Rightarrow_{A'} p'v$, ahol $v_1 = av$ valamely $a \in T$-re és $p'_1 \in Q'$. Ekkor

$$p \in p' = \delta'(p'_1, a) = \bigcup_{p_1 \in p'_1} \delta(p_1, a),$$

azaz, lennie kell olyan $p_1 \in p'_1$-nek, amelyre $p \in \delta(p_1, a)$.

Véges automata determinizálása

Lemma 2 bizonyítása: A lépések száma szerinti indukcióval történik.

Nulla számú lépés esetén az állítás triviális.

Tegyük fel, hogy az állítás igaz n lépésre, ahol $n \geq 0$, és álljon a $q'u \Rightarrow^*_{A'} p'v$ redukció $n+1$ lépésből.

Akkor $q'u \Rightarrow^*_{A'} p'_1 v_1 \Rightarrow_{A'} p'v$, ahol $v_1 = av$ valamely $a \in T$-re és $p'_1 \in Q'$. Ekkor

$$p \in p' = \delta'(p'_1, a) = \bigcup_{p_1 \in p'_1} \delta(p_1, a),$$

azaz, lennie kell olyan $p_1 \in p'_1$-nek, amelyre $p \in \delta(p_1, a)$.

Ekkor p_1-re teljesül, hogy $p_1 v_1 = p_1 av \Rightarrow_A pv$. (1)

Véges automata determinizálása

Lemma 2 bizonyítása: A lépések száma szerinti indukcióval történik.

Nulla számú lépés esetén az állítás triviális.

Tegyük fel, hogy az állítás igaz n lépésre, ahol $n \geq 0$, és álljon a $q'u \Rightarrow^*_{A'} p'v$ redukció $n+1$ lépésből.

Akkor $q'u \Rightarrow^*_{A'} p'_1 v_1 \Rightarrow_{A'} p'v$, ahol $v_1 = av$ valamely $a \in T$-re és $p'_1 \in Q'$. Ekkor

$$p \in p' = \delta'(p'_1, a) = \bigcup_{p_1 \in p'_1} \delta(p_1, a),$$

azaz, lennie kell olyan $p_1 \in p'_1$-nek, amelyre $p \in \delta(p_1, a)$.

Ekkor p_1-re teljesül, hogy $p_1 v_1 = p_1 av \Rightarrow_A pv$. (1)

Másrészt az indukciós hipotézis alapján van olyan $q \in q'$, hogy

$$qu \Rightarrow^*_A p_1 v_1. \quad (2)$$

Véges automata determinizálása

Lemma 2 bizonyítása: A lépések száma szerinti indukcióval történik.

Nulla számú lépés esetén az állítás triviális.

Tegyük fel, hogy az állítás igaz n lépésre, ahol $n \geq 0$, és álljon a $q'u \Rightarrow^*_{A'} p'v$ redukció $n+1$ lépésből.

Akkor $q'u \Rightarrow^*_{A'} p'_1 v_1 \Rightarrow_{A'} p'v$, ahol $v_1 = av$ valamely $a \in T$-re és $p'_1 \in Q'$. Ekkor

$$p \in p' = \delta'(p'_1, a) = \bigcup_{p_1 \in p'_1} \delta(p_1, a),$$

azaz, lennie kell olyan $p_1 \in p'_1$-nek, amelyre $p \in \delta(p_1, a)$.

Ekkor p_1-re teljesül, hogy $p_1 v_1 = p_1 av \Rightarrow_A pv$. (1)

Másrészt az indukciós hipotézis alapján van olyan $q \in q'$, hogy

$$qu \Rightarrow^*_A p_1 v_1. \tag{2}$$

(1) és (2) alapján Lemma 2 állítása teljesül.

Véges automata determinizálása

Végezetül legyen $q_0'u \Rightarrow_{A'}^* p'$ és $p' \in F'$.

Véges automata determinizálása

Végezetül legyen $q'_0 u \Rightarrow^*_{A'} p'$ és $p' \in F'$.

Az F' definíciója alapján van olyan $p \in p'$, ahol $p \in F$, és így Lemma 2 alapján valamely $q_0 \in q'_0$-re teljesül $q_0 u \Rightarrow^*_A p$.

Véges automata determinizálása

Végezetül legyen $q_0' u \Rightarrow_{A'}^* p'$ és $p' \in F'$.

Az F' definíciója alapján van olyan $p \in p'$, ahol $p \in F$, és így Lemma 2 alapján valamely $q_0 \in q_0'$-re teljesül $q_0 u \Rightarrow_A^* p$.

Így $L(A') \subseteq L(A)$. Ezzel a tétel bizonyítását befejeztük.

Véges automata determinizálása

Példa:

	a	b
$\rightarrow q_0$	$\{\,\}$	$\{q_1, q_2\}$
$\leftrightarrows q_1$	$\{q_0\}$	$\{\,\}$
$\leftarrow q_2$	$\{q_1\}$	$\{q_2\}$

VNDA

Véges automata determinizálása

Példa:

	a	b
$\rightarrow q_0$	$\{\}$	$\{q_1, q_2\}$
$\leftrightarrows q_1$	$\{q_0\}$	$\{\}$
$\leftarrow q_2$	$\{q_1\}$	$\{q_2\}$

VNDA

	a	b
$\{\}$	$\{\}$	$\{\}$
$\{q_0\}$	$\{\}$	$\{q_1, q_2\}$
$\leftarrow$ $\{q_1\}$	$\{q_0\}$	$\{\}$
$\leftarrow$ $\{q_2\}$	$\{q_1\}$	$\{q_2\}$
$\leftrightarrows$ $\{q_0, q_1\}$	$\{q_0\}$	$\{q_1, q_2\}$
$\leftarrow$ $\{q_0, q_2\}$	$\{q_1\}$	$\{q_1, q_2\}$
$\leftarrow$ $\{q_1, q_2\}$	$\{q_0, q_1\}$	$\{q_2\}$
$\leftarrow\{q_0, q_1, q_2\}$	$\{q_0, q_1\}$	$\{q_1, q_2\}$

VDA

$\mathcal{L}_3$ további zártsági tulajdonságai

Következmény

$\mathcal{L}_3$ zárt a komplementerre, a metszetre és a különbségre.

$\mathcal{L}_3$ további zártsági tulajdonságai

Következmény

$\mathcal{L}_3$ zárt a komplementerre, a metszetre és a különbségre.

Bizonyítás:

- Legyen $L \in \mathcal{L}_3$, ekkor az előző tételek alapján L felismerhető egy $A = \langle Q, T, \delta, q_0, F \rangle$ véges determinisztikus automatával.

$\mathcal{L}_3$ további zártsági tulajdonságai

Következmény

$\mathcal{L}_3$ zárt a komplementerre, a metszetre és a különbségre.

Bizonyítás:

- Legyen $L \in \mathcal{L}_3$, ekkor az előző tételek alapján L felismerhető egy $A = \langle Q, T, \delta, q_0, F \rangle$ véges determinisztikus automatával.

  Nyilván az $A' = \langle Q, T, \delta, q_0, Q \setminus F \rangle$ véges determinisztikus automata által felismert nyelv $\overline{L}$.

$\mathcal{L}_3$ további zártsági tulajdonságai

Következmény

$\mathcal{L}_3$ zárt a komplementerre, a metszetre és a különbségre.

Bizonyítás:

- Legyen $L \in \mathcal{L}_3$, ekkor az előző tételek alapján L felismerhető egy $A = \langle Q, T, \delta, q_0, F \rangle$ véges determinisztikus automatával.

  Nyilván az $A' = \langle Q, T, \delta, q_0, Q \setminus F \rangle$ véges determinisztikus automata által felismert nyelv $\overline{L}$.

  Mivel $\overline{L}$ felismerhető véges determinisztikus automatával, ezért korábbi tételünk miatt $\overline{L} \in \mathcal{L}_3$.

$\mathcal{L}_3$ további zártsági tulajdonságai

Következmény

$\mathcal{L}_3$ zárt a komplementerre, a metszetre és a különbségre.

Bizonyítás:

- Legyen $L \in \mathcal{L}_3$, ekkor az előző tételek alapján L felismerhető egy $A = \langle Q, T, \delta, q_0, F \rangle$ véges determinisztikus automatával.

  Nyilván az $A' = \langle Q, T, \delta, q_0, Q \setminus F \rangle$ véges determinisztikus automata által felismert nyelv $\overline{L}$.

  Mivel $\overline{L}$ felismerhető véges determinisztikus automatával, ezért korábbi tételünk miatt $\overline{L} \in \mathcal{L}_3$.

- Mivel $L_1 \cap L_2 = \overline{\overline{L_1} \cup \overline{L_2}}$, ezért a metszetre való zártság következik az unióra és a komplementerre való zártságból.

$\mathcal{L}_3$ további zártsági tulajdonságai

Következmény

$\mathcal{L}_3$ zárt a komplementerre, a metszetre és a különbségre.

Bizonyítás:

- Legyen $L \in \mathcal{L}_3$, ekkor az előző tételek alapján L felismerhető egy $A = \langle Q, T, \delta, q_0, F \rangle$ véges determinisztikus automatával.

  Nyilván az $A' = \langle Q, T, \delta, q_0, Q \setminus F \rangle$ véges determinisztikus automata által felismert nyelv $\overline{L}$.

  Mivel $\overline{L}$ felismerhető véges determinisztikus automatával, ezért korábbi tételünk miatt $\overline{L} \in \mathcal{L}_3$.

- Mivel $L_1 \cap L_2 = \overline{\overline{L_1} \cup \overline{L_2}}$, ezért a metszetre való zártság következik az unióra és a komplementerre való zártságból.

- Mivel $L_1 \setminus L_2 = L_1 \cap \overline{L_2}$, ezért a különbségre való zártság következik a metszetre és a komplementerre való zártságból.

A reguáris nyelvosztályt leíró formális eszközök

Tehát az alábbi formális eszközek mind az $\mathcal{L}_3$ nyelvosztályt írják le egy adott ábécé felle:

- jobblineáris grammatika

A reguáris nyelvosztályt leíró formális eszközök

Tehát az alábbi formális eszközek mind az $\mathcal{L}_3$ nyelvosztályt írják le egy adott ábécé felle:

- jobblineáris grammatika
- ballineáris grammatika

A reguáris nyelvosztályt leíró formális eszközök

Tehát az alábbi formális eszközek mind az $\mathcal{L}_3$ nyelvosztályt írják le egy adott ábécé felle:

- jobblineáris grammatika
- ballineáris grammatika
- 3-as normálformájú grammatika

A reguáris nyelvosztályt leíró formális eszközök

Tehát az alábbi formális eszközek mind az $\mathcal{L}_3$ nyelvosztályt írják le egy adott ábécé felle:

- jobblineáris grammatika
- ballineáris grammatika
- 3-as normálformájú grammatika
- reguláris kifejezések

A reguáris nyelvosztályt leíró formális eszközök

Tehát az alábbi formális eszközek mind az $\mathcal{L}_3$ nyelvosztályt írják le egy adott ábécé felle:

- jobblineáris grammatika
- ballineáris grammatika
- 3-as normálformájú grammatika
- reguláris kifejezések
- a fenti műveletekkel bővített általánosított reguláris kifejezések

A reguáris nyelvosztályt leíró formális eszközök

Tehát az alábbi formális eszközek mind az $\mathcal{L}_3$ nyelvosztályt írják le egy adott ábécé felle:

- jobblineáris grammatika
- ballineáris grammatika
- 3-as normálformájú grammatika
- reguláris kifejezések
- a fenti műveletekkel bővített általánosított reguláris kifejezések
- Véges determinisztikus automata

A reguáris nyelvosztályt leíró formális eszközök

Tehát az alábbi formális eszközek mind az $\mathcal{L}_3$ nyelvosztályt írják le egy adott ábécé felle:

- jobblineáris grammatika
- ballineáris grammatika
- 3-as normálformájú grammatika
- reguláris kifejezések
- a fenti műveletekkel bővített általánosított reguláris kifejezések
- Véges determinisztikus automata
- Véges nemdeterminisztikus automata

Összefüggő automata

Definíció

Legyen $A = \langle Q, T, \delta, q_0, F\rangle$ véges determinisztikus automata. A q állapotot a kezdőállapotból **elérhetőnek** mondjuk, ha létezik $q_0 x \Rightarrow^* q$ redukció, ahol $x \in T^*$.

Összefüggő automata

Definíció

Legyen $A = \langle Q, T, \delta, q_0, F \rangle$ véges determinisztikus automata. A q állapotot a kezdőállapotból **elérhetőnek** mondjuk, ha létezik $q_0 x \Rightarrow^* q$ redukció, ahol $x \in T^*$.

Definíció

Az $A = \langle Q, T, \delta, q_0, F \rangle$ véges determinisztikus automatát **összefüggőnek** mondjuk, ha minden állapota elérhető a kezdőállapotból.

Összefüggő automata

Definíció

Legyen $A = \langle Q, T, \delta, q_0, F \rangle$ véges determinisztikus automata. A q állapotot a kezdőállapotból **elérhetőnek** mondjuk, ha létezik $q_0 x \Rightarrow^* q$ redukció, ahol $x \in T^*$.

Definíció

Az $A = \langle Q, T, \delta, q_0, F \rangle$ véges determinisztikus automatát **összefüggőnek** mondjuk, ha minden állapota elérhető a kezdőállapotból.

Megjegyzés: q elérhető a kezdőállapotból, akkor és csak akkor ha van olyan $x \in T^*$, hogy $\delta(q_0, x) = q$.

Összefüggő automata

Definiáljuk a H halmazt a következőképpen. Legyen

$H_0 = \{q_0\}$,

$H_{i+1} = H_i \cup \{r \,|\, \delta(q,a) = r, q \in H_i, a \in T\}, \quad i = 1, 2, \ldots$

Mivel $H_0 \subseteq H_1 \subseteq \cdots \subseteq H_k \subseteq Q$ ezért létezik olyan $k \geq 0$, amelyre $H_k = H_{k+1}$. Ekkor minden $\ell \geq k$-ra $H_\ell = H_k$. Legyen $H := H_k$. Könnyen látható, hogy H azoknak az állapotoknak a halmaza, amelyek a kezdőállapotból elérhetők.

Összefüggő automata

Definiáljuk a H halmazt a következőképpen. Legyen

$H_0 = \{q_0\}$,

$H_{i+1} = H_i \cup \{r \mid \delta(q, a) = r, q \in H_i, a \in T\}, \quad i = 1, 2, \ldots$

Mivel $H_0 \subseteq H_1 \subseteq \cdots \subseteq H_k \subseteq Q$ ezért létezik olyan $k \geq 0$, amelyre $H_k = H_{k+1}$. Ekkor minden $\ell \geq k$-ra $H_\ell = H_k$. Legyen $H := H_k$. Könnyen látható, hogy H azoknak az állapotoknak a halmaza, amelyek a kezdőállapotból elérhetők.

Ezután definiáljuk az $A' = \langle Q', T, \delta', q_0, F' \rangle$ determinisztikus véges automatát a következőképpen:

Összefüggő automata

Definiáljuk a H halmazt a következőképpen. Legyen

$H_0 = \{q_0\}$,

$H_{i+1} = H_i \cup \{r \mid \delta(q, a) = r, q \in H_i, a \in T\}, \quad i = 1, 2, \ldots$

Mivel $H_0 \subseteq H_1 \subseteq \cdots \subseteq H_k \subseteq Q$ ezért létezik olyan $k \geq 0$, amelyre $H_k = H_{k+1}$. Ekkor minden $\ell \geq k$-ra $H_\ell = H_k$. Legyen $H := H_k$. Könnyen látható, hogy H azoknak az állapotoknak a halmaza, amelyek a kezdőállapotból elérhetők.

Ezután definiáljuk az $A' = \langle Q', T, \delta', q_0, F' \rangle$ determinisztikus véges automatát a következőképpen:

$Q' = H, F' = F \cap H$ és $\delta' : H \times T \to H$ úgy, hogy $\delta'(q, a) = \delta(q, a)$, minden $q \in H$ esetén.

Összefüggő automata

Definiáljuk a H halmazt a következőképpen. Legyen

$H_0 = \{q_0\}$,

$H_{i+1} = H_i \cup \{r \,|\, \delta(q, a) = r, q \in H_i, a \in T\}, \quad i = 1, 2, \ldots$

Mivel $H_0 \subseteq H_1 \subseteq \cdots \subseteq H_k \subseteq Q$ ezért létezik olyan $k \geq 0$, amelyre $H_k = H_{k+1}$. Ekkor minden $\ell \geq k$-ra $H_\ell = H_k$. Legyen $H := H_k$. Könnyen látható, hogy H azoknak az állapotoknak a halmaza, amelyek a kezdőállapotból elérhetők.

Ezután definiáljuk az $A' = \langle Q', T, \delta', q_0, F' \rangle$ determinisztikus véges automatát a következőképpen:

$Q' = H, F' = F \cap H$ és $\delta' : H \times T \to H$ úgy, hogy $\delta'(q, a) = \delta(q, a)$, minden $q \in H$ esetén.

Könnyen megmutatható, hogy A' összefüggő és ugyanazt a nyelvet fogadja el, amelyet A.

Összefüggő automata

Definiáljuk a H halmazt a következőképpen. Legyen

$H_0 = \{q_0\}$,

$H_{i+1} = H_i \cup \{r \mid \delta(q, a) = r, q \in H_i, a \in T\}, \quad i = 1, 2, \ldots$

Mivel $H_0 \subseteq H_1 \subseteq \cdots \subseteq H_k \subseteq Q$ ezért létezik olyan $k \geq 0$, amelyre $H_k = H_{k+1}$. Ekkor minden $\ell \geq k$-ra $H_\ell = H_k$. Legyen $H := H_k$. Könnyen látható, hogy H azoknak az állapotoknak a halmaza, amelyek a kezdőállapotból elérhetők.

Ezután definiáljuk az $A' = \langle Q', T, \delta', q_0, F' \rangle$ determinisztikus véges automatát a következőképpen:

$Q' = H, F' = F \cap H$ és $\delta' : H \times T \to H$ úgy, hogy $\delta'(q, a) = \delta(q, a)$, minden $q \in H$ esetén.

Könnyen megmutatható, hogy A' összefüggő és ugyanazt a nyelvet fogadja el, amelyet A.

Továbbá, A' az A legnagyobb összefüggő részautomatája.

Összefüggő automata

Példa:

	a	b
$\rightarrow q_0$	q_2	q_4
$\leftarrow q_1$	q_3	q_0
$\leftarrow q_2$	q_0	q_2
q_3	q_1	q_2
q_4	q_5	q_2
$\leftarrow q_5$	q_4	q_2

$$H_0 = \{q_0\}$$
$$H_1 = \{q_0, q_2, q_4\}$$
$$H_2 = \{q_0, q_2, q_4, q_5\} =$$
$$H_3 = H$$

	a	b
$\rightarrow q_0$	q_2	q_4
$\leftarrow q_2$	q_0	q_2
q_4	q_5	q_2
$\leftarrow q_5$	q_4	q_2

Összefüggő automata

Példa:

	a	b
$\rightarrow q_0$	q_2	q_4
$\leftarrow q_1$	q_3	q_0
$\leftarrow q_2$	q_0	q_2
q_3	q_1	q_2
q_4	q_5	q_2
$\leftarrow q_5$	q_4	q_2

$H_0 = \{q_0\}$
$H_1 = \{q_0, q_2, q_4\}$
$H_2 = \{q_0, q_2, q_4, q_5\} =$
$H_3 = H$

	a	b
$\rightarrow q_0$	q_2	q_4
$\leftarrow q_2$	q_0	q_2
q_4	q_5	q_2
$\leftarrow q_5$	q_4	q_2

Megjegyzés: H úgyis megkapható, ha az átmenetdiagramon a kezdőállapotból indítunk egy szélességi keresést. Ezen algoritmus egy **sor** adatszerkezetet használ. Az eljárás során a sorba kerülő állapotok éppen az elérhető állapotok.

Összefüggő automata

Példa:

	a	b
$\rightarrow q_0$	q_2	q_4
$\leftarrow q_1$	q_3	q_0
$\leftarrow q_2$	q_0	q_2
q_3	q_1	q_2
q_4	q_5	q_2
$\leftarrow q_5$	q_4	q_2

$$H_0 = \{q_0\}$$
$$H_1 = \{q_0, q_2, q_4\}$$
$$H_2 = \{q_0, q_2, q_4, q_5\} =$$
$$H_3 = H$$

	a	b
$\rightarrow q_0$	q_2	q_4
$\leftarrow q_2$	q_0	q_2
q_4	q_5	q_2
$\leftarrow q_5$	q_4	q_2

Megjegyzés: H úgyis megkapható, ha az átmenetdiagramon a kezdőállapotból indítunk egy szélességi keresést. Ezen algoritmus egy **sor** adatszerkezetet használ. Az eljárás során a sorba kerülő állapotok éppen az elérhető állapotok. Ezzel a megközelítéssel néha csökkenthető a VNDA-k determinizálása révén kapott állapotok száma.

VNDA determinizálása – az összefüggő részautomata

Példa:

	a	b
$\rightarrow q_0$	$\{\,\}$	$\{q_1, q_2\}$
$\leftrightarrows q_1$	$\{q_0\}$	$\{\,\}$
$\leftarrow q_2$	$\{q_1\}$	$\{q_2\}$

VNDA determinizálása – az összefüggő részautomata

Példa:

	a	b
$\rightarrow q_0$	$\{\}$	$\{q_1, q_2\}$
$\leftrightarrows q_1$	$\{q_0\}$	$\{\}$
$\leftarrow q_2$	$\{q_1\}$	$\{q_2\}$

	a	b
$\leftrightarrows \{q_0, q_1\}$	$\{q_0\}$	$\{q_1, q_2\}$
$\{q_0\}$	$\{\}$	$\{q_1, q_2\}$
$\leftarrow \{q_1, q_2\}$	$\{q_0, q_1\}$	$\{q_2\}$
$\{\}$	$\{\}$	$\{\}$
$\leftarrow \{q_2\}$	$\{q_1\}$	$\{q_2\}$
$\leftarrow \{q_1\}$	$\{q_0\}$	$\{\}$

Elért állapotok sora:

$\{q_0, q_1\}$

$\{q_0\}$ $\{q_1, q_2\}$

$\{q_1, q_2\}$ $\{\}$

$\{\}$ $\{q_2\}$

$\{q_2\}$

$\{q_1\}$

ÜRES

Algoritmikus problémák

Akkor beszélünk algoritmikus eldöntési problémáról, ha a bemenet egy olyan objektum, mely egy adott végtelen halmazból kerülhet ki (például egy természetes szám vagy adott ábécé feletti szó), kimenete pedig igen/nem.

Algoritmikus problémák

Akkor beszélünk algoritmikus eldöntési problémáról, ha a bemenet egy olyan objektum, mely egy adott végtelen halmazból kerülhet ki (például egy természetes szám vagy adott ábécé feletti szó), kimenete pedig igen/nem.

Olyan megoldást (algoritmust) keresünk, amely kellően általános ahhoz, hogy a végtelen lehetséges bemenet bármelyike esetén alkalmazható legyen.

Algoritmikus problémák

Akkor beszélünk algoritmikus eldöntési problémáról, ha a bemenet egy olyan objektum, mely egy adott végtelen halmazból kerülhet ki (például egy természetes szám vagy adott ábécé feletti szó), kimenete pedig igen/nem.

Olyan megoldást (algoritmust) keresünk, amely kellően általános ahhoz, hogy a végtelen lehetséges bemenet bármelyike esetén alkalmazható legyen.

Egy probléma **eldönthető**, ha létezik olyan minden lehetséges bemenet esetén termináló algoritmus, amelyik a helyes választ adja.

Algoritmikus problémák

Akkor beszélünk algoritmikus eldöntési problémáról, ha a bemenet egy olyan objektum, mely egy adott végtelen halmazból kerülhet ki (például egy természetes szám vagy adott ábécé feletti szó), kimenete pedig igen/nem.

Olyan megoldást (algoritmust) keresünk, amely kellően általános ahhoz, hogy a végtelen lehetséges bemenet bármelyike esetén alkalmazható legyen.

Egy probléma **eldönthető**, ha létezik olyan minden lehetséges bemenet esetén termináló algoritmus, amelyik a helyes választ adja.

Megjegyzés: Ezt a fogalmat majd a Számításelmélet alapjai II tárgyban pontosabban definiáljuk.

3-as típusú grammatikák algoritmikus problémái

Állítás

Eldönthető, hogy egy reguláris grammatika az üres nyelvet generálja-e.

3-as típusú grammatikák algoritmikus problémái

Állítás

Eldönthető, hogy egy reguláris grammatika az üres nyelvet generálja-e.

Bizonyítás: A tanultak szerint a grammatikához megadható olyan A véges determinisztikus automata, amelyik a grammatika által generált nyelvet ismeri fel. Az automata pontosan akkor nem ismer fel egyetlen szót sem, ha a kezdőállapotából minden végállapota elérhetetlen.

3-as típusú grammatikák algoritmikus problémái

Állítás

Eldönthető, hogy egy reguláris grammatika az üres nyelvet generálja-e.

Bizonyítás: A tanultak szerint a grammatikához megadható olyan A véges determinisztikus automata, amelyik a grammatika által generált nyelvet ismeri fel. Az automata pontosan akkor nem ismer fel egyetlen szót sem, ha a kezdőállapotából minden végállapota elérhetetlen.

Állítás

Eldönthető, hogy két reguláris grammatika által generált nyelv diszjunkt-e.

3-as típusú grammatikák algoritmikus problémái

Állítás

Eldönthető, hogy egy reguláris grammatika az üres nyelvet generálja-e.

Bizonyítás: A tanultak szerint a grammatikához megadható olyan A véges determinisztikus automata, amelyik a grammatika által generált nyelvet ismeri fel. Az automata pontosan akkor nem ismer fel egyetlen szót sem, ha a kezdőállapotából minden végállapota elérhetetlen.

Állítás

Eldönthető, hogy két reguláris grammatika által generált nyelv diszjunkt-e.

Bizonyítás: Generálják a G_1 és G_2 reguláris grammatikák rendre az L_1 és L_2 nyelveket. Korábbi állításunk miatt $L_1 \cap L_2$ is reguláris, így az előző állítás szerint ennek üressége eldönthető.

3-as típusú grammatikák algoritmikus problémái

Állítás

Eldönthető, hogy két reguláris grammatika ugyanazt a nyelvet generálja-e vagy sem.

3-as típusú grammatikák algoritmikus problémái

Állítás

Eldönthető, hogy két reguláris grammatika ugyanazt a nyelvet generálja-e vagy sem.

Bizonyítás:

Generálják a G_1 és G_2 reguláris grammatikák rendre az L_1 és L_2 nyelveket. Az $L_3 = (L_1 \cap \bar{L}_2) \cup (\bar{L}_1 \cap L_2)$ nyelv szintén reguláris, így van olyan G_3 reguláris grammatika, amely L_3-at generálja. Ekkor azonban $L_1 = L_2$ akkor és csak akkor, ha $L_3 = \emptyset$, amely a fentiek szerint eldönthető.

3-as típusú grammatikák algoritmikus problémái

Állítás

Eldönthető, hogy két reguláris grammatika ugyanazt a nyelvet generálja-e vagy sem.

Bizonyítás:

Generálják a G_1 és G_2 reguláris grammatikák rendre az L_1 és L_2 nyelveket. Az $L_3 = (L_1 \cap \bar{L}_2) \cup (\bar{L}_1 \cap L_2)$ nyelv szintén reguláris, így van olyan G_3 reguláris grammatika, amely L_3-at generálja. Ekkor azonban $L_1 = L_2$ akkor és csak akkor, ha $L_3 = \emptyset$, amely a fentiek szerint eldönthető.

Állítás

Eldönthető, hogy egy reguláris grammatika bővebb vagy egyenlő nyelvet generál-e mint egy másik.

3-as típusú grammatikák algoritmikus problémái

Állítás

Eldönthető, hogy két reguláris grammatika ugyanazt a nyelvet generálja-e vagy sem.

Bizonyítás:

Generálják a G_1 és G_2 reguláris grammatikák rendre az L_1 és L_2 nyelveket. Az $L_3 = (L_1 \cap \bar{L}_2) \cup (\bar{L}_1 \cap L_2)$ nyelv szintén reguláris, így van olyan G_3 reguláris grammatika, amely L_3-at generálja. Ekkor azonban $L_1 = L_2$ akkor és csak akkor, ha $L_3 = \emptyset$, amely a fentiek szerint eldönthető.

Állítás

Eldönthető, hogy egy reguláris grammatika bővebb vagy egyenlő nyelvet generál-e mint egy másik.

Generálják a G_1 és G_2 reguláris grammatikák rendre az L_1 és L_2 nyelveket. Az $L_3 = (L_1 \cap \bar{L}_2) = L_1 \setminus L_2$ szintén reguláris. Ennek üressége eldönthető, ami ekvivalens azzal, hogy $L_1 \subseteq L_2$.

Lineáris algoritmus a 3-as típusú szóproblémára

Szóprobléma: Adott a G grammatika és $u \in T^*$. $u \overset{?}{\in} L(G)$.

Tétel

A 3-as típusú grammatikák szóproblémája (hatékonyan) eldönthető.

Lineáris algoritmus a 3-as típusú szóproblémára

Szóprobléma: Adott a G grammatika és $u \in T^*$. $u \overset{?}{\in} L(G)$.

Tétel

A 3-as típusú grammatikák szóproblémája (hatékonyan) eldönthető.

Bizonyítás: Legyen $G = \langle N, T, P, S \rangle$ egy 3-as típusú grammatika normálformában adva és $u = t_1 \cdots t_n$ a levezetendő szó.

Lineáris algoritmus a 3-as típusú szóproblémára

Szóprobléma: Adott a G grammatika és $u \in T^*$. $u \overset{?}{\in} L(G)$.

Tétel

A 3-as típusú grammatikák szóproblémája (hatékonyan) eldönthető.

Bizonyítás: Legyen $G = \langle N, T, P, S \rangle$ egy 3-as típusú grammatika normálformában adva és $u = t_1 \cdots t_n$ a levezetendő szó.

Az algoritmus rekurzívan kiszámol egy N részhalmazaiból álló H_i $(0 \leq i \leq n)$ sorozatot. H_i-ból H_{i+1} (csak G-től függő) konstans időben számolható.

Lineáris algoritmus a 3-as típusú szóproblémára

Szóprobléma: Adott a G grammatika és $u \in T^*$. $u \stackrel{?}{\in} L(G)$.

Tétel

A 3-as típusú grammatikák szóproblémája (hatékonyan) eldönthető.

Bizonyítás: Legyen $G = \langle N, T, P, S \rangle$ egy 3-as típusú grammatika normálformában adva és $u = t_1 \cdots t_n$ a levezetendő szó.

Az algoritmus rekurzívan kiszámol egy N részhalmazaiból álló H_i $(0 \leq i \leq n)$ sorozatot. H_i-ból H_{i+1} (csak G-től függő) konstans időben számolható.

$H_0 := \{S\}$

Lineáris algoritmus a 3-as típusú szóproblémára

Szóprobléma: Adott a G grammatika és $u \in T^*$. $u \stackrel{?}{\in} L(G)$.

Tétel

A 3-as típusú grammatikák szóproblémája (hatékonyan) eldönthető.

Bizonyítás: Legyen $G = \langle N, T, P, S \rangle$ egy 3-as típusú grammatika normálformában adva és $u = t_1 \cdots t_n$ a levezetendő szó.

Az algoritmus rekurzívan kiszámol egy N részhalmazaiból álló H_i $(0 \leq i \leq n)$ sorozatot. H_i-ból H_{i+1} (csak G-től függő) konstans időben számolható.

$H_0 := \{S\} \quad H_{i+1} := \{A \in N \,|\, \exists\, B \in H_i \wedge B \rightarrow t_{i+1}A \in P\}$.

Lineáris algoritmus a 3-as típusú szóproblémára

Szóprobléma: Adott a G grammatika és $u \in T^*$. $u \overset{?}{\in} L(G)$.

Tétel

A 3-as típusú grammatikák szóproblémája (hatékonyan) eldönthető.

Bizonyítás: Legyen $G = \langle N, T, P, S \rangle$ egy 3-as típusú grammatika normálformában adva és $u = t_1 \cdots t_n$ a levezetendő szó.

Az algoritmus rekurzívan kiszámol egy N részhalmazaiból álló H_i ($0 \le i \le n$) sorozatot. H_i-ból H_{i+1} (csak G-től függő) konstans időben számolható.

$$H_0 := \{S\} \quad H_{i+1} := \{A \in N \mid \exists B \in H_i \wedge B \to t_{i+1}A \in P\}.$$

Könnyen látható, hogy H_i azon nemterminálosok halmaza, melyek pontosan i levezetési lépés után a mondatforma végén állhatnak.

Lineáris algoritmus a 3-as típusú szóproblémára

Szóprobléma: Adott a G grammatika és $u \in T^*$. $u \overset{?}{\in} L(G)$.

Tétel

A 3-as típusú grammatikák szóproblémája (hatékonyan) eldönthető.

Bizonyítás: Legyen $G = \langle N, T, P, S \rangle$ egy 3-as típusú grammatika normálformában adva és $u = t_1 \cdots t_n$ a levezetendő szó.

Az algoritmus rekurzívan kiszámol egy N részhalmazaiból álló H_i $(0 \leq i \leq n)$ sorozatot. H_i-ból H_{i+1} (csak G-től függő) konstans időben számolható.

$$H_0 := \{S\} \quad H_{i+1} := \{A \in N \,|\, \exists\, B \in H_i \wedge B \rightarrow t_{i+1}A \in P\}.$$

Könnyen látható, hogy H_i azon nemterminálosok halmaza, melyek pontosan i levezetési lépés után a mondatforma végén állhatnak.

Tehát ha $F = \{A \in N \,|\, A \rightarrow \varepsilon \in P\}$, akkor nyilván

Lineáris algoritmus a 3-as típusú szóproblémára

Szóprobléma: Adott a G grammatika és $u \in T^*$. $u \overset{?}{\in} L(G)$.

Tétel

A 3-as típusú grammatikák szóproblémája (hatékonyan) eldönthető.

Bizonyítás: Legyen $G = \langle N, T, P, S \rangle$ egy 3-as típusú grammatika normálformában adva és $u = t_1 \cdots t_n$ a levezetendő szó.

Az algoritmus rekurzívan kiszámol egy N részhalmazaiból álló H_i $(0 \leq i \leq n)$ sorozatot. H_i-ból H_{i+1} (csak G-től függő) konstans időben számolható.

$H_0 := \{S\} \quad H_{i+1} := \{A \in N \,|\, \exists\, B \in H_i \wedge B \rightarrow t_{i+1}A \in P\}$.

Könnyen látható, hogy H_i azon nemterminálosok halmaza, melyek pontosan i levezetési lépés után a mondatforma végén állhatnak.

Tehát ha $F = \{A \in N \,|\, A \rightarrow \varepsilon \in P\}$, akkor nyilván

$$u \in L(G) \Leftrightarrow H_n \cap F \neq \emptyset.$$

Lineáris algoritmus a 3-as típusú szóproblémára

Szóprobléma: Adott a G grammatika és $u \in T^*$. $u \overset{?}{\in} L(G)$.

Tétel

A 3-as típusú grammatikák szóproblémája (hatékonyan) eldönthető.

Bizonyítás: Legyen $G = \langle N, T, P, S \rangle$ egy 3-as típusú grammatika normálformában adva és $u = t_1 \cdots t_n$ a levezetendő szó.

Az algoritmus rekurzívan kiszámol egy N részhalmazaiból álló H_i $(0 \leq i \leq n)$ sorozatot. H_i-ból H_{i+1} (csak G-től függő) konstans időben számolható.

$H_0 := \{S\} \quad H_{i+1} := \{A \in N \,|\, \exists\, B \in H_i \wedge B \to t_{i+1}A \in P\}$.

Könnyen látható, hogy H_i azon nemterminálosok halmaza, melyek pontosan i levezetési lépés után a mondatforma végén állhatnak.

Tehát ha $F = \{A \in N \,|\, A \to \varepsilon \in P\}$, akkor nyilván

$$u \in L(G) \Leftrightarrow H_n \cap F \neq \emptyset.$$

Lineáris algoritmus a 3-as típusú szóproblémára

Példa:

$S \to aA \mid bS \mid \varepsilon$
$A \to aA \mid aS$

Lineáris algoritmus a 3-as típusú szóproblémára

Példa:

$S \to aA \mid bS \mid \varepsilon$
$A \to aA \mid aS$

abb-re $H_0 = \{S\}, H_1 = \{A\}, H_2 = \emptyset, H_3 = \emptyset,$

Lineáris algoritmus a 3-as típusú szóproblémára

Példa:

$S \to aA \,|\, bS \,|\, \varepsilon$
$A \to aA \,|\, aS$

abb-re $H_0 = \{S\}, H_1 = \{A\}, H_2 = \emptyset, H_3 = \emptyset$,

míg aab-re $H_0 = \{S\}$, $H_1 = \{A\}, H_2 = \{A, S\}, H_3 = \{S\}$

lesz a H_i-k sorozata. Mivel most $F = \{S\}$, ezért aab generálható, abb viszont nem.

Vegyük észre, hogy $H_i \in \mathcal{P}(N)$ és minden $1 \leq i \leq n-1$-re H_{i+1} csak H_i-től és t_{i+1}-től függ. Azaz a $\{H_i\}_{0 \leq i \leq n}$ halmazok meghatározása egy VDA segítségével automatizálható.

	a	b
$\leftrightarrows \{S\}$	$\{A\}$	$\{S\}$
$\{A\}$	$\{S, A\}$	$\{\}$
$\leftarrow \{S, A\}$	$\{S, A\}$	$\{S\}$
$\{\}$	$\{\}$	$\{\}$

Lineáris algoritmus a 3-as típusú szóproblémára

Megjegyzések:

- Ez a VDA nem más, mint a G-hez készített VNDA determinizáltja. $\mathcal{P}(N)$ állapothalmazzal, $H_0 = \{S\}$ kezdőállapottal, $\delta(H, a) = \{B \in N \mid \exists A \in H : A \to aB \in P\}$ állapotátmenet-függvénnyel ($H \in \mathcal{P}(N)$ és $a \in T$) és $F' = \{H \subseteq N \mid H \cap F \neq \emptyset\}$ elfogadó állapothalmazzal, ahol $F = \{A \in N \mid A \to \varepsilon \in P\}$.
- Az algoritmus **lineáris** időben eldönti, $u \overset{?}{\in} L(G)$, hiszen minden egyes H_i halmaz kiszámításának ideje csak G-től ($|u|$-tól nem) függő konstans.

Lineáris algoritmus a 3-as típusú szóproblémára

Megjegyzések:

- Ez a VDA nem más, mint a G-hez készített VNDA determinizáltja. $\mathcal{P}(N)$ állapothalmazzal, $H_0 = \{S\}$ kezdőállapottal, $\delta(H, a) = \{B \in N \mid \exists\, A \in H : A \to aB \in P\}$ állapotátmenet-függvénnyel ($H \in \mathcal{P}(N)$ és $a \in T$) és $F' = \{H \subseteq N \mid H \cap F \neq \emptyset\}$ elfogadó állapothalmazzal, ahol $F = \{A \in N \mid A \to \varepsilon \in P\}$.
- Az algoritmus **lineáris** időben eldönti, $u \overset{?}{\in} L(G)$, hiszen minden egyes H_i halmaz kiszámításának ideje csak G-től ($|u|$-tól nem) függő konstans.
- Amennyiben G méretéhez képest hosszú szóra vagy több szóra szeretnénk a kérdést eldönteni érdemes lehet előfeldolgozó lépésként a fenti automatát elkészíteni.

Lineáris algoritmus a 3-as típusú szóproblémára

Megjegyzések:

- Ez a VDA nem más, mint a G-hez készített VNDA determinizáltja. $\mathcal{P}(N)$ állapothalmazzal, $H_0 = \{S\}$ kezdőállapottal, $\delta(H, a) = \{B \in N \mid \exists\, A \in H : A \to aB \in P\}$ állapotátmenet-függvénnyel ($H \in \mathcal{P}(N)$ és $a \in T$) és $F' = \{H \subseteq N \mid H \cap F \neq \emptyset\}$ elfogadó állapothalmazzal, ahol $F = \{A \in N \mid A \to \varepsilon \in P\}$.
- Az algoritmus **lineáris** időben eldönti, $u \overset{?}{\in} L(G)$, hiszen minden egyes H_i halmaz kiszámításának ideje csak G-től ($|u|$-tól nem) függő konstans.
- Amennyiben G méretéhez képest hosszú szóra vagy több szóra szeretnénk a kérdést eldönteni érdemes lehet előfeldolgozó lépésként a fenti automatát elkészíteni.
- Másrészt, kevés, kisméretű szó és nagyméretű grammatika esetén hatékonyabb lehet a H_i halmazok közvetlen kiszámítása.