

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.

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

1. Lemma

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

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

Az 1. Lemma bizonyítása:

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

Nulla számú lépés esetében az állítás triviálisan fennáll, hiszen ekkor $q = p$ kell legyen és így a 0 lépéses levezetés illetve $p' = q'$ megfelel.

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 $q_1 \in \delta(q, a)$.

$q \in q'$ 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 az 1. Lemma állítása teljesül.

Véges automata determinizálása

Az 1. Lemmából $L(A) \subseteq L(A')$:

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 az 1. Lemma 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 $u \in L(A')$ adódik.

Mivel $u \in L(A)$ tetszőleges volt, ezért $L(A) \subseteq L(A')$.

Véges automata determinizálása

2. Lemma

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

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

A 2. Lemma bizonyítása: A lépések száma szerinti indukcióval történik.

Nulla számú lépés esetén az állítás triviális. (A 0 lépéses levezetés és $q = p$ jó lesz.)

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

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 $p_1 v_1 = p_1 a v \Rightarrow_A p v$. (1)

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

$$q u \Rightarrow_A^* p_1 v_1. \quad (2)$$

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

A 2. Lemmából $L(A') \subseteq L(A)$:

Tegyük fel, $u \in L(A')$, azaz $q'_0 u \Rightarrow_{A'}^*$, p' és $p' \in F'$ teljesül.

Ekkor F' definíciója alapján van olyan $p \in p'$, melyre $p \in F$ teljesül.

A 2. Lemma alapján valamely $q_0 \in q'_0 = Q_0$ -ra $q_0 u \Rightarrow_A^* p$ teljesül.

Ez azt jelenti, hogy $u \in L(A)$.

Mivel u tetszőleges, így $L(A') \subseteq L(A)$, bizonyítva a tételt. \square

Véges automata determinizálása

Példa:

	<i>a</i>	<i>b</i>
$\rightarrow q_0$	$\{\}$	$\{q_1, q_2\}$
$\Leftrightarrow q_1$	$\{q_0\}$	$\{\}$
$\leftarrow q_2$	$\{q_1\}$	$\{q_2\}$

VNDA

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

<i>a</i>	<i>b</i>
$\{\}$	$\{\}$
$\{\}$	$\{q_1, q_2\}$
$\{q_0\}$	$\{\}$
$\{q_1\}$	$\{q_2\}$
$\{q_0\}$	$\{q_1, q_2\}$
$\{q_1\}$	$\{q_1, q_2\}$
$\{q_0, q_1\}$	$\{q_2\}$
$\{q_0, q_1\}$	$\{q_1, q_2\}$

VDA

\mathcal{L}_3 további zártsági tulajdonságai

Következmény

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

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 \bar{L} .

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

- ▶ Mivel $L_1 \cap L_2 = \overline{\bar{L}_1 \cup \bar{L}_2}$, ezért a metszetre való zártság következik az unióra és a komplementerre való zártságból.
- ▶ Mivel $L_1 \setminus L_2 = L_1 \cap \bar{L}_2$, ezért a különbségre való zártság következik a metszetre és a komplementerre való zártságból.

A reguláris nyelvosztályt leíró formális eszközök

Tehát az alábbi formális eszközök mind az \mathcal{L}_3 nyelvosztályt írják le egy adott ábécé felett:

- ▶ jobblineáris grammatikák
- ▶ ballineáris grammatikák
- ▶ 3-as normálformájú grammatikák
- ▶ reguláris kifejezések
- ▶ a komplementer, metszet, különbség, tükörkép, pozitív lezárt műveletekkel bővített ún. általánosított reguláris kifejezések
- ▶ véges determinisztikus automaták
- ▶ véges nemdeterminisztikus automaták

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

$$H_0 = \{q_0\},$$

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

Mivel $H_0 \subseteq H_1 \subseteq \dots \subseteq H_k \subseteq Q$ 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 $\delta' : H \times T \rightarrow H$ 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

Megjegyzés: Egy az átmenetdiagramon a kezdőállapotból indított szélességi keresés révén is meghatározható az elérhető állapotok H halmaza. H elemei ugyanis éppen azon állapotok, amelyek az algoritmus során valamikor benn voltak a **sorban**. Példa: VNDA-k determinizálásakor is gyakran csökkenthető az állapotok száma.

VNDA determinizálása – az összefüggő részautomata

Példa:

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

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

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.

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.

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 (és egy őt generáló grammatika algoritmikusan előállítható) 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.

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 $u \in T^*$. $u \stackrel{?}{\in} L(G)$.

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 részhalmazából álló H_i ($0 \leq i \leq n$) sorozatot. H_i -ből H_{i+1} (csak G -től függő) konstans időben számolható.

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

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

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

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

Lineáris algoritmus a 3-as típusú szóproblémára

Példa:

$$S \rightarrow aA \mid bS \mid \varepsilon$$

$$A \rightarrow aA \mid aS$$

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
$\Leftrightarrow \{S\}$	$\{A\}$	$\{S\}$
$\{A\}$	$\{S, A\}$	$\{\}$
$\leftarrow \{S, A\}$	$\{S, A\}$	$\{S\}$
$\{\}$	$\{\}$	$\{\}$

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, $\delta(H, a) = \{B \in N \mid \exists A \in H : A \rightarrow aB \in P\}$ állapotátmenet-függvénnel ($H \in \mathcal{P}(N)$ és $a \in T$) és $F' = \{H \subseteq N \mid H \cap F \neq \emptyset\}$ elfogadó állapothalmazzal, ahol $F = \{A \in N \mid A \rightarrow \varepsilon \in P\}$.
- ▶ Az algoritmus **lineáris** időben eldönti, $u \stackrel{?}{\in} L(G)$, hiszen minden egyes H_i halmaz kiszámításának ideje csak G -től ($|u|$ -től 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.