

IX. Előadás (2020. Ápr. 23.)

(A) Reguláris és környezetfüggetlen nyelvek kapcsolata

Figyeljük meg hogy egy tetszőleges véges determinisztikus automatát felfoghatunk úgy, mint egy olyan (determinisztikus) veremautomatát aminek a működése nem függ a verem tartalmától - abban az értelemben hogy a működés során a verem tartalma nem változik (mindig Z_0) marad. Tekintsük az 1. Ábrán látható automatát, amely esetén minden egyes utasítás

$$\delta(q_i, a, Z_0) = \{(q_j, Z_0)\}$$

alakú, ε -os utasítás nincs értelmezve, viszont a δ leképezés minden $(q_i, a) \in Q \times \Sigma$ párra értelmezett (Z_0 veremállapottal). Tehát a véges automata egy speciális veremautomatának tekinthető. Ezért a véges det. automaták által elfogadott nyelvek egyúttal veremautomaták által elfogadott nyelvek is, ami az előző órán tanultak értelmében azt jelenti hogy környezetfüggetlen nyelvek. Mivel minden reguláris nyelv előállítható egy véges determinisztikus automatával, az alábbi eredményt kapjuk:

Következmény: Minden reguláris nyelv egyben környezetfüggetlen nyelv is, még hozzá egy determinisztikus veremautomata nyelve.

Megjegyzés 1. Fontos megemlíteni hogy a programozási nyelvek mind determinisztikus veremautomatákkal előállítható nyelvek. Ezért úgy tekinthetők mint a reguláris nyelvek általánosításai.

Vannak azonban olyan környezetfüggetlen nyelvek is amelyek nem állíthatók elő determinisztikus veremautomatákkal. Ezért a determinisztikus és nondeterminisztikus veremautomaták, a véges automatáktól eltérően, **nem** ekvivalens automata típusok!

(B) Környezetfüggetlen nyelvek zártsági tulajdonságai

Tétel 1.(i) A környezetfüggetlen nyelvek családja zárt a $\cup$, $\cdot$ és $*$ műveletekre.
(ii) A környezetfüggetlen nyelvek családja zárt a homomorfizmusra és a behelyettesítésre.

Biz.: Csak néhány tulajdonságot bizonyítunk.

(i) Lássuk be például, hogy ha $L_1, L_2 \subseteq \Sigma^*$ környezetfüggetlen nyelvek, akkor $L_1 \cup L_2$ is az.

Valóban, ekkor léteznek olyan $G_1 = (V_1, P_1, \Sigma, S_1)$ és $G_2 = (V_2, P_2, \Sigma, S_2)$ környezetfüggetlen nyelvtanok amelyekre $L_1 = \mathcal{L}(G_1)$ és $L_2 = \mathcal{L}(G_2)$ és nyilvánvalóan $L_1 \cup L_2 \subseteq \Sigma^*$.

Tekintsük most azt a $G^* = (V_1 \cup V_2 \cup \{S\}, P, \Sigma, S)$ környezetfüggetlen nyelvtant amire $P = P_1 \cup P_2 \cup \{S \rightarrow S_1 \mid S_2\}$. Igazolni fogjuk hogy $\mathcal{L}(G^*) = L_1 \cup L_2$, ez pedig azt jelenti hogy $L_1 \cup L_2$ egy környezetfüggetlen nyelv. Valóban, egy $w \in \Sigma^*$ szóra $w \in \mathcal{L}(G^*)$ pontosan akkor teljesül, ha létezik egy $S \xRightarrow{*} w$ deriváció a P -beli szabályok felhasználásával. De P definíciója szerint ez csak $S \xRightarrow{*} S_1 \xRightarrow{*} w$ vagy $S \xRightarrow{*} S_2 \xRightarrow{*} w$ alakú lehet. Az első esetben, az $S_1 \xRightarrow{*} w$ deriváció során csak P_1 -beli szabályokat alkalmazhatunk mert $S_1 \in V_1$ és S_1 -ből csak V_1 -beli változók érhetők el deriválással. A második esetben, az $S_2 \xRightarrow{*} w$ deriváció során csak P_2 -beli szabályokat használhatunk mert $S_2 \in V_2$ és S_2 -ből csak V_2 -beli változók érhetők el. Ezért az 1. esetben $w \in \mathcal{L}(G_1)$, míg a 2. esetben $w \in \mathcal{L}(G_2)$. Tehát $w \in \mathcal{L}(G_1) \cup \mathcal{L}(G_2) = L_1 \cup L_2$. Ez azt jelenti hogy $\mathcal{L}(G^*) = L_1 \cup L_2$.

(ii) Elég csupán azt bizonyítani hogy ha $L \subseteq \Sigma^*$ egy környezetfüggetlen nyelv, $s: \Sigma \rightarrow \mathcal{K}$ pedig egy behelyettesítés (ahol $\mathcal{K}$ a környezetfüggetlen nyelvek családját jelenti), akkor $s(L) \in \mathcal{K}$, ugyanis minden homomorfizmus származtatható egy sajátos behelyettesítéssel ahol $s(a)$ egy egyetlen w_a szóból álló nyelv. Így egy L k. független nyelv homomorf képe is k. független lesz.

A bizonyítás alapgondolatát egy szemléltető példán mutatjuk be: Legyen $L = \{a^i b^i \mid i \geq 0\}$, $L_a = \{0^n 1^n \mid n \geq 1\}$, $L_b = \{w w^R \mid w \in (0+2)^*\}$ és

$$s(a) = L_a, s(b) = L_b.$$

Nilván $s(L) \subseteq (0+1+2)^*$ és $s(L) = \{w^i z^i \mid w \in L_a, z \in L_b, i \geq 0\}$, definíció szerint. $s(L)$ -t úgy fogjuk pontosítani, hogy megkonstruálunk egy G^s környezetfüggetlen nyelvtant ami éppen $s(L)$ -t állítja elő. Ebből nyilván az is következni fog, hogy $s(L)$ környezetfüggetlen. Könnyen belátható hogy:

$$\begin{aligned} L \text{ előállítható a köv. } G \text{ nyelvtannal: } & S \rightarrow aSb \mid \varepsilon \\ L_a \text{ előállítható a } G_a \text{ nyelvtannal: } & S_a \rightarrow 0S_a1 \mid 01 \\ L_b \text{ előállítható a } G_b \text{ nyelvtannal: } & S_b \rightarrow 0S_b0 \mid 2S_b2 \mid \varepsilon \end{aligned}$$

Most $s(L)$ -t a következő k. független nyelvtannal adhatjuk meg.

$$P : \begin{cases} S \rightarrow S_a S S_b \mid \varepsilon \\ S_a \rightarrow 0S_a1 \mid 01 \\ S_b \rightarrow 0S_b0 \mid 2S_b2 \mid \varepsilon \end{cases}$$

□

Megjegyzés 2. A környezetfüggetlen nyelvek családja általában nem zárt a komplementerképzésre és metszett műveletre. Igaz viszont az alábbi:

Állítás 1. Ha $L \subseteq \Sigma^*$ egy környezetfüggetlen nyelv és $R \subseteq \Sigma^*$ egy reguláris nyelv, akkor $L \cap R$ is egy környezetfüggetlen nyelv.

Az állítást itt nem igazoljuk. Arra szoktuk felhasználni hogy egy nyelvről megmutassuk hogy nem környezetfüggetlen, mint ahogy azt az alábbi példa is mutatja:

Példa 1. Legyen $L = \{ww \mid w \in (a+b)^*\}$. Igazoljuk hogy L nem környezetfüggetlen. Tekintsük az $\{a^+b^+a^+b^+\}$ reguláris nyelvet. Ha L környezetfüggetlen lenne, akkor az Állítás 1. szerint az $L \cap \{a^+b^+a^+b^+\} = \{a^ib^ja^ib^j \mid i, j \geq 1\}$ nyelv is környezetfüggetlen lenne, holott ez szerepelt a nem környezetfüggetlen nyelvekre adott példák között. Tehát L sem lehet környezet független nyelv.

(C) Bar-Hillel lemma

A következő lemmát szintén arra szoktuk használni hogy megmutassuk hogy egy nyelv nem lehet környezetfüggetlen. A lemma azt mondja ki hogy a környezetfüggetlen nyelvek, a reguláris nyelvekhez hasonlóan rendelkeznek egyfajta „pumpálhatósági tulajdonsággal”.

Lemma (Bar-Hillel). *Legyen L egy környezetfüggetlen nyelv. Akkor létezik egy olyan $n \in \mathbb{N}_+$ (L -től függő) küszöbszám, úgy hogy minden $z \in L \subseteq \Sigma^*$ szó amire $|z| \geq n$, mindig feírható $z = uvwxy$ alakban (ahol $u, v, w, x, y \in \Sigma^*$), méghozzá úgy, hogy teljesülnek az alábbi feltételek.*

- 1) $|vx| \geq 1$;
- 2) $1 \leq |vwx| \leq n$;
- 3) $uv^iwx^iy \in L$, minden $i \geq 0$ egész számra.

Alkalmazás: Mutassuk meg hogy az $L = \{a^ib^ic^i \mid i \geq 1\}$ nyelv **nem** környezetfüggetlen.

- Tegyük fel hogy L környezetfüggetlen és jussunk ellentmondásra!

A fenti Lemma szerint ekkor L -re létezik egy, a fenti feltételeket teljesítő n küszöbszám. Tekintsük a $z = a^n b^n c^n \in L$ szót (ahol $n \geq 1$). Mivel $|z| = 3n \geq n$, ezért $z = uvwxy$ alakban írható és a z szóra a B-H lemma három feltételének teljesülnie kell. Most vizsgáljuk meg a vwx részszt:

Ez biztosan nem tartalmazhatja mind a három betűt (lásd 2. Ábra), mert ha v csak egyetlen a -betűt és x csak egyetlen c betűt is tartalmazna, akkor mind az n db. b -betű ebben a részsztóban lenne így vwx hossza legalább $n + 2$ lenne ami ellentmond a 2) feltételnek.

Megmutatjuk hogy vwx nem tartalmazhat csak egyetlen betűt. Valóban, ha ez mondjuk az a betű lenne, akkor minden b és c betű az y részsztóban lenne. Mivel $|vx| \geq 1$ (Lemma 1) feltétel, ezért vx legalább egy a betűt tartalmaz. De akkor a $z^0 = uwy$ szóban az a betűk száma legalább 1-el kisebb lenne, vagyis legfeljebb $n - 1$, míg a b , illetve a c betűk száma továbbra is n -maradna. ez ellentmond annak, hogy a 3) feltételbe $i = 0$ -t írva $z^0 = uwy \in L$ -t kapunk! (z^0 nem lehet L -ben mert nem tartalmaz ugyanannyit mindhárom betűből). Hasonlóan bizonyítjuk azt is, hogy L nem tartalmazhat csak b betűt, vagy csak c betűt.

Nyilvánvaló, hogy vwx nem állhat csupán a és c betűkből, hiszen a b betűk ezek között helyezkednek el a z szóban. Tehát vwx -ben vagy csak a és b betű, vagy csak b és c betű lehet. Megmutatjuk hogy mindkét eset ellentmondáshoz vezet. Elegendő csak az első esetben ezt bizonyítani, mert a második esetben az bizonyítás az elsővel teljesen szimmetrikus.

Tegyük fel tehát hogy a vwx részsóban csak a és b betű van. Akkor szükség szerint mind az n drb. c betű z -ből az y részsóban van. A 3) feltétel szerint $z^0 = uv^0wx^0y = uwy \in L$. Az elmondottak miatt z^0 -ban továbbra is n drb. c betű van, azonban a $|vx| \geq 1$ feltétel teljesülése miatt, vagy az a vagy a b betűből z^0 -ban legalább eggyel kevesebb van, vagyis legfeljebb $n-1$. De ez azt jelenti hogy $z^0 = uwy \notin L$, ami ellentmondás. Mivel minden lehetséges részletben ellentmondáshoz jutunk, L -re nem alkalmazható a B-H lemma, így nem lehet környezetfüggetlen nyelv. $\square$

- Hasonlóan, a Bar-Hillel lemma felhasználásával igazoljuk hogy az $L = \{a^i b^j a^i b^j \mid i, j \geq 1\}$ nyelv nem környezetfüggetlen.

(D) Eldönthetőségi kérdések

Ezek a kérdések környezetfüggetlen nyelvek esetén ugyanúgy felmerülnek mint a reguláris nyelvek esetén és amint azt az alábbi eredmény mutatja, a válaszok is hasonlóak.

>

Tétel 1.(i) Létezik algoritmus annak eldöntésére hogy egy $L \subseteq \Sigma^*$ környezetfüggetlen nyelv

- (1) üres-e;
- (2) véges-e
- (3) végtelen-e.

(ii) Létezik algoritmus annak az eldöntésére hogy egy $w \in \Sigma^*$ szó egy adott $L \subseteq \Sigma^*$ környezetfüggetlen nyelvhez tartozik-e vagy sem.

A fenti tételt itt nem bizonyítjuk.

Megjegyzés 3. Általában ezek az algoritmusok nem polinomiális lépésszámúak, ellenkezőleg, a lépésszámuk az adatok méretével exponenciálisan nő.

Az (ii) pontbeli algoritmus gyakorlati megvalósítása az ún. *Lexikális elemzés* alapfeladata. A környezetfüggetlen nyelvek osztályán belül különböző sajátos nyelvosztályokat különíthetünk el -ezek elemzésére különböző speciális algoritmusokat használnak, amelyek lépésszáma már polinomiális nagyságrendű, pl. lásd az irodalomban és az utolsó előadáson az ún LR(k) nyelveket.

(E) Környezetfüggetlen nyelvek és veremautomaták alkalmazásai

(1) Már említettük hogy programozási nyelvek is környezetfüggetlen nyelvek, még hozzá determinisztikus veremautomatákkal előállítható környezetfüggetlen nyelvek. Ezen az osztályon belül is, látni fogjuk, hogy nagyon közel állnak a reguláris nyelvekhez, bár azoknál általánosabbak.

(2) A mai computerek is determinisztikus veremautomatáknak tekinthetők, bár a valóságban az ún. *regiszter automaták* jözzé szokták besorolni őket, amelyek úgy képzelhetők el mint véges számú véges veremmel rendelkező determinisztikus automaták. Igaz viszont az, hogy minden regiszter gép ekvivalens egy veremautomatával.

(3) Az első, valamilyen kezdetleges memóriával rendelkező, mechanikai számológépek is veremautomaták voltak- a honvédségnél alkalmazták őket.

(F) Turing gépek

A fogalmat amit később róla neveztek el, Alain Turing vezette be 1936-ban és nagyon nevezetes matematikai előzményei vannak. Ellentétben a veremautomatákkal amelyet gyakran fizikailag is megvalósítanak, a *Turing gépek* valójában elméleti számítási modellek, az algoritmus fogalmának a szemléltetésére és tanulmányozására szolgál.

Matematikai előzmények: D.Hilbert, aki a XX század matematikai kutatásainak egy meghatározó, programadó személyisége volt, vetette fel azt a problémát, hogy létezik-e olyan a természetes számokra vonatkozó állítás (predikátum) hogy annak sem igaz sem hamis voltát nem lehet a természetes számok Peano-féle axiomarendszeréből levezetni? K. Gödel osztrák matematikus 1932-ben konstruktív módon bizonyította hogy vannak ilyen állítások. Felmerült a kérdés, hogy esetleg algoritmikusan nem lehet-e az ilyen állítások igaz vagy hamis voltát eldönteni. A. Turing, a róla elnevezett gép segítségével egy újabb bizonyítást adta Gödel válaszána és megmutatta hogy vannak olyan $\mathbb{N}$ feletti $P(x_1, \dots, x_k)$ predikátumok amelyek igaz vagy hamis volta algoritmikus úton sem dönthető el.

A Modell.

Úgy lett megkonstruálva hogy elvben egy korlátlan memóriával rendelkező „egyszerű gépet” kapjunk:

A Turing gépnek tehát

- egyetlen cellákra osztott (input) szallagja van. Ez kezdetben csak a Σ input ábécé betűit és blank-et (B) - vagyis üres cellát tartalmazhat. Erre a Γ ún. output ábécé betűivel írhatunk. Itt $\Sigma \subseteq \Gamma$ és $B \in \Gamma$.

- A fej mozoghat jobbra és balra is egy lépést a szallagon és átírhatja annak egy celláját. Állapotai egy véges Q halmazhoz tartoznak, aminek kötelezően tartalmaznia kell egy q_0 kezdőállapotot és egy vagy több végállapotot. Ezek halmazát F -el jelöljük.

Működése: A gép a q_i állapot és a beolvasott a_k jel függvényében az aktuális cella helyére egy másik jelet ír és eggyel jobbra vagy balra mozdul el a szallagon miközben állapotát q_j -re változtatja.

Matematikai alakban egy Turing gépet a köv. képpen adhatunk meg:

$$\mathcal{M} = (Q, \Sigma, \Gamma, \delta, q_0, B, F)$$

ahol

Q az automata állapotainak véges nem üres halmaza;

Σ input ábécé (véges, nem üres);

Γ output ábécé (véges, nem üres) és $\Sigma \subseteq \Gamma$, $B \in \Gamma$.

$q_0 \in Q$ az automata kezdőállapota (Start állapot);

$F \subseteq Q$ a végállapotok (véges) halmaza, $F \neq \emptyset$

$B \in \Gamma$ a blank (üres cella) szimbóluma.

$\delta: Q \times \Sigma \longrightarrow Q \times \Gamma \times \{L, R\}$ ún. tranzíciós függvény, ahol L -jelentése: *left* (egy lépés balra a szallagon), R jelentése: *right* (egy lépés jobbra)

δ egy parciális (nem mindenütt értelmezett) függvény amit általában egy táblázattal adunk meg. Egy tranzíciós lépés $(q_i, a_k) \longrightarrow (q_j, X_k, D)$ alakú, ahol $D \in \{L, R\}$ és $X_k \in \Gamma$.

Példa 2.: Tekintsük azt az $\mathcal{M}$ Turing gépet amelyre:

$Q = \{q_0, q_1, q_2, q_3, \mathbf{q}_4\}$, $F = \{\mathbf{q}_4\}$, $\Sigma = \{0, 1\}$, $\Gamma = \{0, 1, X, Y, B\}$, δ -t pedig az alábbi táblázat adja meg:

δ	0	1	X	Y	B
q_0	(q_1, X, R)	—	—	(q_3, Y, R)	—
q_1	$(q_1, 0, R)$	(q_2, Y, L)	—	(q_1, Y, R)	—
q_2	$(q_2, 0, L)$	—	(q_0, X, R)	(q_2, Y, L)	—
q_3	—	—	—	(q_3, Y, R)	$(\mathbf{q}_4, B, R)$
$\mathbf{q}_4$	—	—	—	—	—

$\mathbf{q}_4$ végállapotban szándékosan nincs értelmezve utasítás, azért hogy ebben az állapotban a gép megálljon.