Egy „kicserélési lemma" környezet-független nyelvekre

Pontos bibliográfiai adatok:

William Ogden, Rockford J. Ross, and Karl Winklmann. An „interchange lemma" for context-free languages. *SIAM Journal on Computing*, 14(2):410–415, May 1985.

Rövid cím: Kicserélési lemma.

A cikk egy új szükséges feltételt ad környezet–független nyelvekre, majd példát is ad az alkalmazására: a lemma segítségével belátja, hogy nem környezet–független egy olyan nyelvről, amely sokáig ellenállt minden hasonló bizonyítási kísérletnek.

Tétel. *Legyen L egy T abc-jű környezet–független nyelv. Ekkor létezik olyan $c > 0$ konstans, hogy minden $n \geq 2$ egészre, minden $Q \subseteq L \cap T^n$ halmazra és minden olyan m egészre, melyre $n \geq m \geq 2$ létezik $k \geq |Q|/(cn^2)$ egész és $z_1, \dots, z_k \in Q$ szavak a következő tulajdonságokkal:*

1. $z_i = w_i x_i y_i$, $i = 1, \dots, k$,
2. $|w_1| = |w_2| = \cdots = |w_k|$,
3. $|y_1| = |y_2| = \cdots = |y_k|$,
4. $m \geq |x_1| = |x_2| = \cdots = |x_k| > m/2$, *és*
5. $w_i x_j y_i \in L$ *minden* $i, j \in \{1, 2, \dots, k\}$ *esetén.*

A bizonyítás Chomsky normálformában levő nyelvtanokhoz tartozó levezetési fák levelei számlálásának segítségével történik.

A cikk az alkalmazásra is példát ad: megmutatja, hogy a „dadogó" szavak (azaz az $xyyz$ alakú szavak, ahol y nem üres) nyelve hárombetűs abc felett nem környezet–független.

Szavak kombinatorikus tulajdonságai és a Chomsky osztályok kapcsolata

Pontos bibliográfiai adatok:
Dömösi Pál et al. Some combinatorial properties of words, and the Chomsky hierarchy. In *Proc. Internat. Conf. „Words, Languages and Combinatorics, II", Kyoto, Japan; eds.: M. Ito and H. Jürgensen*, pages 105–123. Singapore: World Scientific, 1992.

Rövid cím: Kombinatorikus tulajdonságok.

Jelöljük Q-val a primitív szavak nyelvét, azaz azokat a szavakat, amelyek nem állnak elő más szó hatványaként. (Formálisan $Q = T^+ \setminus \{w^m \mid w \in T^*, m \geq 2\}$, ahol T egy legalább kétbetűs abc.) Sokat kutatott téma, hogy vajon Q környezet–független-e, a sejtés az, hogy nem. Jelen cikk nem dönti el a kérdést, de belátja, hogy Q teljesít három olyan feltételt, az Ogden, az erős Bader–Moura és a Sokolowsky feltételt, amelyet minden környezet–független nyelv teljesít. A cikk egy negyedik feltételt is vizsgál, ami szükséges feltétele minden környezet–független nyelvnek, a Parikh szemilinearitást. Megmutatja, hogy $Q \cap (a^*b)^n$ (ahol $n \geq 1$ egész) Parikh szemilineáris, tehát ez a támadási kísérlet is kudarcot vall. A dolgozat utolsó része általánosan vizsgál bizonyos nyelvi tulajdonságokat, különös tekintettel azok eldönthetőségére és elhelyezkedésükre a Kleene féle aritmetikai hierarchiában.

Definíció. *Azt mondjuk, hogy egy $L \subseteq T^*$ nyelv teljesíti a* Ogden feltételt, *ha létezik olyan $n \geq 2$ egész, melyre $z \in L$ esetén, ha legalább $n+1$ pozíciót z-ben kitüntetettnek tekintünk, akkor z felbontható $z = uvwxy$ alakban, ahol teljesülnek a következők:*

1. *vagy u, v és w, vagy w, x és y mindegyike tartalmaz kitüntetett pozíciót,*
2. *vwx legfeljebb n kitüntetett pozíciót tartalmaz,*
3. *minden $m \geq 0$ számra $uv^mwx^my \in L$.*

Tétel. *Minden környezet–független nyelv teljesíti az Ogden feltételt.*

Definíció. *Azt mondjuk, hogy egy $L \subseteq T^*$ nyelv teljesíti a* Sokolowsky feltételt, *ha minden $Y \subseteq T^*$, $|Y| \geq 2$, és minden $u_1, u_2, u_3 \in T^*$ esetén, ha $\{u_1xu_2xu_3 \mid x \in Y^+\} \subseteq L$, akkor létezik olyan $x', x'' \in Y^+$, $x' \neq x''$, melyre $u_1x'u_2x''u_3 \in L$.*

Tétel. *Minden környezet–független nyelv teljesíti a Sokolowsky feltételt.*

Környezet–független nyelvekre adott kicserélési és pumpáló lemmák

Pontos bibliográfiai adatok:
R. Boonyavatana and G. Slutzki. The interchange or pump (DI)lemmas for context-free languages. *Theoretical Computer Science*, 56(3):321–338, March 1988.

Rövid cím: Feltételek összehasonlítása.

A cikk összehasonlítja a kicserélési lemmát különböző pumpáló lemmákkal: a Bar–Hillel félével, Ogdenével, ennek általánosított változatával, a fentiek lineáris változatával, valamint a Sokolowsky féle feltételekkel. Ezen felül definiál egy kicserélési feltételt lineáris környezet–független nyelvekre, és összehasonlítja a többi feltétellel. Az eredmények megmutatják, hogy a kicserélési feltételek nem összehasonlíthatóak a pumpáló feltételekkel és a kiterjesztett Sokolowsky feltétellel, de határozottan erősebbek a Sokolowsky feltételnél.

Definíció. *Azt mondjuk, hogy egy $L \subseteq T^*$ nyelv teljesíti a* lineáris kicserélési feltételt, *ha létezik olyan $c > 0$ konstans, hogy minden $n \geq m \geq 2$ egészekre, és minden $Q \subseteq L \cap T^n$ halmazra létezik $R \subseteq Q$ a következő tulajdonságokkal:*

1. *$|R| \geq |Q|/(cn)$, és*
2. *valamilyen $i \geq 0$ számra $\{xwy \mid \exists u, v, z \in T^* : |x| = |u| = i, |w| = |z| = m, xzy, uwv \in R\} \subseteq L \cap T^n$.*

Definíció. *Azt mondjuk, hogy egy $R \subseteq T^* \times T^*$ reláció* nem korlátos, *ha minden m számra léteznek olyan $x, y \in T^*$ szavak, melyre $|x|, |y| \geq m$, és $R(x, y)$ fennáll.*

Definíció. *$x, y \in T^*$ esetén $x < y$ azt jelenti, hogy x-et úgy kaphatjuk y-ból, hogy néhány betűjét kitöröljük. Ha a törölt szimbólumok y utolsó (első) m szimbóluma közül kerülnek ki, azt így jelöljük: $x <^m y$ ($x \ {}^m\!< y$). Szavak párjaira definiáljuk a következő relációt:*

$$(x', y') <_m (x, y) \Leftrightarrow (x' < x \wedge y' = y) \vee (x' = x \wedge y' < y) \vee (x' <^m x \wedge y' \ {}^m\!< y).$$

Definíció. *Azt mondjuk, hogy egy $L \subseteq T^*$ nyelv teljesíti a* kiterjesztett Sokolowsky feltételt, *ha minden $u_1, u_2, u_3 \in T^*$ szóra és minden $R \subseteq T^* \times T^*$ nem korlátos relációra, ha $\{u_1 x u_2 y u_3 \mid R(x, y)\} \subseteq L$, akkor létezik olyan m egész, hogy minden $x, y \in T^*$ esetén*

$$|x|, |y| > m \ \wedge \ R(x, y) \implies \exists x', y' : (x', y') <_m (x, y) \ \wedge \ u_1 x' u_2 y' u_3 \in L.$$

A rekurzívan felsorolható nyelvek jellemzése beszúró–törlő rendszerek segítségével

Pontos bibliográfiai adatok:
Lila Kari, Gheorghe Păun, Gabriel Thierrin, and Sheng Yu. At the crossroads of dna computing and formal languages: Characterizing recursively enumerable languages using insertion-deletion systems. In *Proceedings of the 3rd DIMACS Workshop on DNA Based Computers, held at the University of Pennsylvania, June 23 – 25, 1997*, pages 318–333, June 1997.

Rövid cím: Beszúró–törlő rendszerek.

A formális nyelvek elméletében a leggyakoribb generáló mechanizmus az újraírás. A természetes nyelvek vizsgálata során merült fel a gondolat, hogy az újraírás mellett, vagy helyett szövegrészek beszúrásával, vagy törlésével képezzünk új szövegeket. E módszerek a molekuláris számítások szempontjából is rendkívül fontosak, mert különböző enzimek segítségével DNS, illetve RNS molekulákon megvalósíthatóak bizonyos (a környezettől is függő) beszúrások és törlések. A cikk a beszúró–törlő rendszerek több alosztályáról belátja, hogy az általuk generált nyelv megegyezik a rekurzívan felsorolható nyelvekkel.

Definíció. *Beszúró–törlő rendszeren a következő konstrukciót értjük:*

$$\gamma = (V, T, A, I, D),$$

ahol V *egy abc,* $T \subseteq V$*,* $A \subseteq V^*$*,* $I, D \subseteq V^* \times V^* \times V^*$ *véges halmazok.* T a rendszerünk terminális abc-je, A az axiómák, I a beszúró, D a törlő szabályok halmaza. $(u, z, v) \in I$ azt jelenti, hogy ha egy szóban előfordul az uv részszó, akkor u és v közé beszúrhatjuk a z szót. $(u, z, v) \in D$ azt jelenti, hogy ha egy szóban előfordul az uzv részszó, akkor u és v közül kitörölhetjük a z szót.

$x, y \in V^*$ esetén $x \Longrightarrow y$ pontosan akkor áll fenn, ha a következők egyike teljesül:

- $x = x_1uvx_2$, $y = x_1uzvx_2$, $x_1, x_2 \in V^*$ és $(u, z, v) \in I$
- $x = x_1uzvx_2$, $y = x_1uvx_2$, $x_1, x_2 \in V^*$ és $(u, z, v) \in D$

Jelölje $\Longrightarrow^*$ a $\Longrightarrow$ reláció reflexív, tranzitív lezártját. A γ által generált nyelv:

$$L(\gamma) = \{w \in T^* \mid x \Longrightarrow^* w, x \in A\}.$$

Nyelvek generálása kétfajta zárójelpár segítségével

Pontos bibliográfiai adatok:
Viliam Geffert. How to generate languages using only two pairs of parentheses. *EIK: Journal of Information Processing and Cybernetics EIK (formerly Elektronische Informationsverarbeitung und Kybernetik)*, 27, 1991.

Rövid cím: Geffert normálforma.

A szerző két jól használható normálformát ad a 0. típusú nyelvtanokra.

Tétel. *Minden G nulladik típusú nyelvtanhoz megadható egy vele ekvivalens G' nyelvtan, melynek idő– és térbonyolultsága aszimptotikusan egyenlő a G nyelvtanéval, mindössze* 5 *nemterminális szimbólumot használ (ezek: $S', (,), [,]$), szabályai kettő kivételével környezet–függetlenek. (A két kivétel: $() \to \epsilon$, és $[] \to \epsilon$.)*

Tétel. *Minden G nulladik típusú nyelvtanhoz megadható egy vele ekvivalens G' nyelvtan, melynek idő– és térbonyolultsága aszimptotikusan egyenlő a G nyelvtanéval, mindössze* 5 *nemterminális szimbólumot használ (ezek: S', X, A, B, C), szabályai egy kivételével környezet–függetlenek. (A kivétel: $ABC \to \epsilon$.)*

Fair párosító rendszerek

Pontos bibliográfiai adatok:
H. J. Hoogeboom and N. van Vugt. Fair Sticker Languages. *Technical Report, L.I.A.C.S., University of Leiden, NL*, Number 00-01, 2000.

Rövid cím: Fair párosító rendszerek.

1994–ben nagy vihart kavart Leonard M. Adleman cikke, amiben leírta egy kísérletét, melyben a molekuláris biológia eszközeivel megoldotta a Hamilton–útkeresés feladatának egy példáját. Az érdeklődést fokozta, hogy a köztudomásúan NP–teljes problémát sikerült polinomiális idő alatt megoldani. Módszerének formális modelljét Lila Kari, Gheorghe Păun, Grzegorz Rozenberg, Arto Salomaa és Sheng Yu adta meg 1996–ban, párosító rendszerek (matching systems, vagy sticker systems) néven. Jelen cikk ezen modellek egy speciális esetével, az ún. fair számítási módú párosító rendszerekkel foglalkozik. Megoldja azt a múlt évig megoldatlan problémát, hogy az ilyen rendszerek által generált nyelvek mindig környezet–függetlenek–e. Sőt: bebizonyítja, hogy az így generált nyelvek kódolásainak osztálya pontosan megegyezik az ún. vak egy–számlálójú gépek által elfogadott nyelvosztállyal.

Definíció. *Párosító rendszeren a következő ötöst értjük:*

$$\gamma = (V, \rho, D_u, D_l, A),$$

ahol V egy véges, nem üres abc, $\rho \subseteq V \times V$ egy szimmetrikus reláció V felett, $D_u, D_l \subseteq V^+$ felső és alsó nyúlványok véges halmaza, végül $A \subseteq V^* \times V^*$ az axiómák véges halmaza.

Az általános és a fair számítási módú rendszerek által generált nyelvek:

$$\begin{aligned} L(\gamma) = \{x \in V^* \mid\ & \exists y \in V^* : (x,y) \in \rho, x = x_0x', y = y_0y', \\ & (x_0, y_0) \in A, x' \in D_u^*, y' \in D_l^*\}, \\ L_f(\gamma) = \{x \in V^* \mid\ & \exists y \in V^*, n \geq 0 : (x,y) \in \rho, x = x_0x', y = y_0y', \\ & (x_0, y_0) \in A, x' \in D_u^n, y' \in D_l^n\}. \end{aligned}$$

Definíció. Vak egy–számlálójú gép *alatt a következő ötöst értjük: $B = (Q, \Sigma, \delta, q_{in}, F)$, ahol Q az állapotok véges halmaza, Σ az input abc, $q_{in} \in Q$ a kezdőállapot, $F \subseteq Q$ a végállapotok halmaza, és $\delta \subseteq Q \times \Sigma \times \{-1, 0, 1\} \times Q$ az utasítások véges halmaza.*

A vak egy–számlálójú gép működése hasonló a véges, nem determinisztikus automatáéhoz, csak még tartalmaz egy számlálót is. A számláló kezdetben nulla, az egyes állapot–átmenetek módosíthatják az értékét legfeljebb eggyel, és akkor fogad el a gép egy szót, ha végigolvasása után végállapotba jut, és a számláló értéke is nulla. Működése közben gépünk semmilyen módon nem függ a számláló értékétől, így azt sem tudja ellenőrizni, hogy az érték éppen nulla–e.

Betűnkénti keverés

Pontos bibliográfiai adatok:
B. Bérard. Literal shuffle. *Theoretical Computer Science*, 51(3), pp. 281-299, 1987.

Rövid cím: Betűnkénti keverés.

Két szó keverését a legkönnyebben a következőképp képzelhetjük el: az abc betűi kártyalapok. Az abc feletti szavak lapsorozatok. Két szó keverése úgy áll elő, hogy a lapsorozatokat (lazán a kézben tartva) egymásba csúsztatjuk. Így az egyes szavak betűi nem keverednek össze, de a két szó betűi igen. A betűnkénti keverés annyival kötöttebb, hogy a két lapsorozat „közös részén" a lapoknak szigorúan felváltva kell érkezniük a két pakliból. Így keverni már csak egy ügyes kártyás tud. A keverés és a betűnkénti keverés műveletét kézenfekvő általánosítani nyelvekre is.

A keverés művelete jól használható bizonyos párhuzamos feldolgozási, illetve többpontú kommunikációs problémák modellezésében. A betűnkénti keverés ezen modellek szinkronizált változatát adja.

Ez a cikk pótolja azt a hiányosságot, hogy bár a keverés műveletét többben alaposan tanulmányozták, a betűnkénti keverést eddig nem. Közli a betűnkénti keverés pontos definícióját, alaptulajdonságait. Foglalkozik a Chomsky nyelvosztályoknak a betűnkénti keverés műveletére való zártságának kérdésével. Vizsgálja továbbá a betűnkénti keverés műveletével kapott nyelvek bizonyos tulajdonságait.

Definíció. *A* keverés *(shuffle) műveleten a következőt értjük: $x, y \in X^*$ esetén $x \sqcup\!\sqcup\, y = \{x_1y_1 \ldots x_ny_n \mid n \geq 1, x = x_1 \ldots x_n, y = y_1 \ldots y_n, x_i, y_i \in X^*(1 \leq i \leq n)\}$.*

Tétel. *A keverés művelet kommutatív, asszociatív. Az $\mathcal{L}_0, \mathcal{L}_1, \mathcal{L}_3$ nyelvosztályok zártak a keverés műveletére nézve.*

Képnyelvek tulajdonságainak eldönthetetlenségéről

Pontos bibliográfiai adatok:
D. Robilliard et D. Simplot. Undecidability of existential properties in picture languages. *Theoretical Computer Science*, 233, 2 (1999), 51-74.

Rövid cím: Képnyelvek.

A cikk három módszert is leír, hogy hogy jelenthetnek meghatározott abc feletti szavak síkbeli képeket. Ilyen módon szavak nyelvéből képnyelveket kapunk. A dolgozat bevezeti a képek bizonyos geometriai tulajdonságait, és vizsgálja, hogy mikor eldönthető reguláris nyelvekről, hogy tartalmaznak–e olyan képrészletet, aminek meghatározott tulajdonsága van. Az eredmény a legtöbb esetben negatív: eldönthetetlen problémák sorát kapjuk ilyen módon.

Absztrakt nyelvosztályok

Pontos bibliográfiai adatok:
Arto Salomaa. *Abstact Families of Languages, In: Formal Languages*, chapter IV, pages 123–141. Academic Press, New York, 1973.

Rövid cím: AFL–elmélet.

A formális nyelvek elméletében gyakori feladat bizonyos konkrét nyelvosztályok jellemzése különböző szempontokból. Megfigyelhetjük, hogy amikor bizonyítjuk, hogy egy adott nyelvosztály teljesít valamilyen adott tulajdonságot, általában a nyelvosztálynak csak néhány alaptulajdonságát használjuk ki. Ez vezet ahhoz a gondolathoz, hogy ne konkrét nyelvosztályokat vizsgáljunk, hanem nyelvosztályok olyan gyűjteményét, melyben minden egyes elem teljesít bizonyos alaptulajdonságokat.

Definíció. *Egy nyelvosztály AFL, ha tartalmaz legalább egy nem–üres nyelvet, és zárt a következő műveletekre nézve: unió, ϵ–mentes lezárás, ϵ–mentes homomorfizmus, inverz homomorfizmus, reguláris nyelvvel való metszet.*

Valamennyi Chomsky–osztály AFL. Ezek után, ami tételeket ki tudunk mondani egy AFL–re, az automatikusan igaz lesz az összes Chomsky osztályra, így meg tudjuk spórolni a bizonyítások ismétlését.

A könyvrészlet definiálja az AFL fogalmát, bemutatja néhány tulajdonságát. Definiál további nyelvosztály–gyűjteményeket, és azok tulajdonságait is vizsgálja.

Logikai formulákkal definiálható naylevek

Pontos bibliográfiai adatok:
Arto Salomaa. *Rudimentary predicates, In: Formal Languages*, chapter III/12, pages 109–119. Academic Press, New York, 1973.

Rövid cím: Logikai formulák.

A formális nyelvek elméletében nyelvek definiálására a leggyakoribb eszközök vagy generálják, vagy elfogadják az adott nyelvet. A könyvrészlet egy ezektől különböző módszert mutat be: nyelvek definiálását logikai formulák segítségével. Ezek a formulák sokszor nagyon kényelmesen használhatóak, mivel a nyelvek intuitív definícióját formális leírássá fordítják. Gyakran könnyen és természetesen alkalmazhatóak adott nyelvekből újak definiálására is.

A szerző pontosan definiálja, hogy milyen típusú logikai formulákat használ nyelvek definiálására, és belátja, hogy az íly módon definiálható nyelvek osztálya megegyezik $\mathcal{L}_0$-al.

Lindenmayer rendszerek

Pontos bibliográfiai adatok:
Arto Salomaa. *Lindenmayer systems, In: Formal Languages*, chapter VII/13, pages 235–252. Academic Press, New York, 1973.

Rövid cím: Lindenmayer rendszerek.

A Lindenmayer rendszerek a nyelvtanokban megszokott újraíráson alapulnak, mégis három jelentős eltérést mutatnak a nyelvtanokhoz képest. Nincsenek benne külön terminális és nyelvtani jelek, így valamennyi levezetett mondatforma generált szó is egyben. Az ebből adódó hatóerő–csökkenést némiképp ellensúlyozza, hogy nem a mondatforma egy jelét írjuk egy lépésben át, hanem egyszerre, szimultán valamennyi jelét, illetve, hogy nem kezdő jelünk (axiómánk), hanem kezdő szavunk adott. Első hallásra talán meglepő, hogy bizonyos algák növekedését olyan pontosan le lehet írni L–rendszerek segítségével, hogy az elmélet a kidolgozását is némiképp az algáknak köszönheti.

A szerző definiálja az L–rendszerek néhány fajtáját, majd példát ad rájuk. Megvizsgálja a rendszerek néhány tulajdonságát, pl. zártsági tételeket bizonyít, összehasonlítja az így definiálható nyelvosztályokat a Chomsky osztályokkal, majd ejt néhány szót az L–rendszerek ún. növekedési függvényeinek érdekességeiről.

Az eldönthetőség és az el nem dönthetőség határmezsgyéjén I.

Pontos bibliográfiai adatok:

Maurice Margenstern Frontier between decidability and undecidability: a survey *Theoretical Computer Science*, 231(2), pp. 217-251, January 2000.

Rövid cím: Eldönthetőség I..

A tanulmány pontosan definálja az eldönthetőség és az el nem dönthetőség fogalmát, majd áttekintést ad a lehető legélesebb határokról, ami az eldőnthető problémák és a hozzájuk tartozó eldönthetetlen problémák között fekszenek. A diszkrét számítások széles köréből veszi a problémákat, az érintett modellek: diofantoszi egyenletek, a szóprobléma, Post–féle rendszerek, molekuláris számítások, regiszter–gépek, neurális hálózatok, sejtautomaták, síklefedések, síkbeli Turing-gépek. Ezek után rátér a klasszikus Turing gépek elemzésére, és egy példával fejezi be a problémák bemutatását: az ún. $3x + 1$ problémát elemzi.

Az első (I.) részhez az első két fejezet: a bevezetés és a klasszikus Turing gépen kívüli modellek tartoznak.

Az eldönthetőség és az el nem dönthetőség határmezsgyéjén II.

Pontos bibliográfiai adatok:

Maurice Margenstern Frontier between decidability and undecidability: a survey *Theoretical Computer Science*, 231(2), pp. 217-251, January 2000.

Rövid cím: Eldönthetőség II..

A tanulmány pontosan definálja az eldönthetőség és az el nem dönthetőség fogalmát, majd áttekintést ad a lehető legélesebb határokról, ami az eldőnthető problémák és a hozzájuk tartozó eldönthetetlen problémák között fekszenek. A diszkrét számítások széles köréből veszi a problémákat, az érintett modellek: diofantoszi egyenletek, a szóprobléma, Post–féle rendszerek, molekuláris számítások, regiszter–gépek, neurális hálózatok, sejtautomaták, síklefedések, síkbeli Turing-gépek. Ezek után rátér a klasszikus Turing gépek elemzésére, és egy példával fejezi be a problémák bemutatását: az ún. $3x + 1$ problémát elemzi.

A második részhez a 3, 4, 5 fejezetek tartoznak: a klasszikus Turing gépek elemzése, a $3x + 1$ probléma és a befejezés.

Kooperáló nyelvtan–rendszerek I.

Pontos bibliográfiai adatok:
E. Csuhaj-Varjú, J. Dassow, J. Kelemen and Gh. Paun. Grammar systems. A grammatical approach to distribution and cooperation. Topics in Computer Mathematics 8. Gordon and Breach Science Publishers, Yverdon, 1994.

Rövid cím: Nyelvtan–rendszerek I.

A klasszikus formális nyelvi elméletek olyan eszközökkel (pl. automata, nyelvtan) dolgoznak, melyek elfogadnak, vagy generálnak egy–egy nyelvet. Ez a megközelítés a hagyományos olyan területeken, mint pl. természetes nyelvek elemzése, programozási nyelvek fordítása. Az utóbbi időkben viszont egyre elterjedtebbek azok a megközelítési módok, amikben nem egy eszköz old meg egy feladatot, hanem több, viszonylag egyszerűbb eszköz együttműködve dolgozik a probléma megoldásán. A könyvrészlet a nyelvtanok fogalmát terjeszti ki ebben az értelemben: definálja a kooperáló nyelvtan–rendszerek fogalmát és megvizsgálja azok tulajdonságait.

Az első (I.) részhez a 3.1 és a 3.2 fejezetek tartoznak.

Definíció. *CD grammatika rendszer alatt a következő $n+2$-est értjük:*

$$\Gamma = (T, G_1, \ldots, G_n, S), \text{ahol}$$

- $G_i = (N_i, T_i, P_i)$ $(1 \leq i \leq n)$ *szokásos, axióma nélüli környezet–független grammatikák,* N_i *a nemterminálisok halmaza,* T_i *a terminálisoké,* P_i *a szabályhalmaz,*
- $T \subseteq \bigcup_{i=1}^{n} T_i$,
- $S \in \bigcup_{i=1}^{n} N_i = N$.

Levezetés $x \stackrel{t}{\Longrightarrow}_{G_i} y$, *ha létezik olyan* $x_1, \ldots, x_k$,*melyre*

- $x = x_1, y = x_k$,
- $x_j \Longrightarrow_{G_i} x_{j+1}$ $(1 \leq j \leq k-1)$,
- *nincs olyan* z, *melyre* $y \Longrightarrow_{G_i} z$

Generált nyelv

$L(\Gamma) = \{w \in T^* | \exists w_1, \ldots, w_k, i_1, \ldots, i_k : S \Longrightarrow_{G_{i_1}} w_1 \Longrightarrow_{G_{i_2}} \cdots \Longrightarrow_{G_{i_k}} w_k = w\}$

Elfogadási stílus

Azt mondjuk, hogy a Γ *CD grammatika rendszer elfogadási stílusa*

- *arb, ha* $T \subseteq \bigcup_{i=1}^{n} T_i$
- *one, ha* $T = T_i$ *valamely* $1 \leq i \leq n$*-re.*
- *ex, ha* $T = \bigcup_{i=1}^{n} T_i$
- *all, ha* $T = \bigcap_{i=1}^{n} T_i$

Kooperáló nyelvtan–rendszerek II.

Pontos bibliográfiai adatok:
E. Csuhaj-Varjú, J. Dassow, J. Kelemen and Gh. Paun. Grammar systems. A grammatical approach to distribution and cooperation. Topics in Computer Mathematics 8. Gordon and Breach Science Publishers, Yverdon, 1994.

Rövid cím: Nyelvtan–rendszerek II.

A klasszikus formális nyelvi elméletek olyan eszközökkel (pl. automata, nyelvtan) dolgoznak, melyek elfogadnak, vagy generálnak egy–egy nyelvet. Ez a megközelítés a hagyományos olyan területeken, mint pl. természetes nyelvek elemzése, programozási nyelvek fordítása. Az utóbbi időkben viszont egyre elterjedtebbek azok a megközelítési módok, amikben nem egy eszköz old meg egy feladatot, hanem több, viszonylag egyszerűbb eszköz együttműködve dolgozik a probléma megoldásán. A könyvrészlet a nyelvtanok fogalmát terjeszti ki ebben az értelemben: definálja a kooperáló nyelvtan–rendszerek fogalmát és megvizsgálja azok tulajdonságait.

A második (II.) részhez a 3.3, 3.4 fejezetek, valamint a befejezés tartozik.

Definíció. *CD grammatika rendszer alatt a következő* $n+2$*-est értjük:*

$$\Gamma = (T, G_1, \ldots, G_n, S), \text{ahol}$$

- $G_i = (N_i, T_i, P_i)$ $(1 \le i \le n)$ *szokásos, axióma nélüli környezet–független grammatikák,* N_i *a nemterminálisok halmaza,* T_i *a terminálisoké,* P_i *a szabályhalmaz,*
- $T \subseteq \bigcup_{i=1}^n T_i$,
- $S \in \bigcup_{i=1}^n N_i = N$.

Levezetés $x \overset{t}{\Longrightarrow}_{G_i} y$*, ha létezik olyan* $x_1, \ldots, x_k$*,melyre*

- $x = x_1, y = x_k$,
- $x_j \Longrightarrow_{G_i} x_{j+1}$ $(1 \le j \le k-1)$,
- *nincs olyan* z*, melyre* $y \Longrightarrow_{G_i} z$

Generált nyelv

$L(\Gamma) = \{w \in T^* | \exists w_1, \ldots, w_k, i_1, \ldots, i_k : S \Longrightarrow_{G_{i_1}} w_1 \Longrightarrow_{G_{i_2}} \cdots \Longrightarrow_{G_{i_k}} w_k = w\}$

Elfogadási stílus

Azt mondjuk, hogy a Γ *CD grammatika rendszer elfogadási stílusa*

- *arb, ha* $T \subseteq \bigcup_{i=1}^n T_i$
- *one, ha* $T = T_i$ *valamely* $1 \le i \le n$*-re.*
- *ex, ha* $T = \bigcup_{i=1}^n T_i$
- *all, ha* $T = \bigcap_{i=1}^n T_i$

Sejtautomaták

Pontos bibliográfiai adatok:
Jozef Gruska. Foundations of Computing. International Thomson Computer Press, 1997

Rövid cím: Sejtautomaták.

Az ötvenes években a kutatókat foglalkoztató egyik legizgalmasabb kérdés az volt, hogy képesek–e gépek önmaguk reprodukálására. Egyfajta választ erre a kérdésre Neumann János adott, aki megalkotta 1953–ban a sejtautomatákat, biológiai motivációk alapján. Ezzel megmutatta, hogy léteznek ön–reprodukáló gépek. A sejtautomaták később a párhuzamos számítógépek egy modelljének matematikai alapját is képezték.

A sejtautomaták olyan véges állapotú automatáknak valamilyen d–dimenziós elrendezésű, minden irányban végtelen hálózata, melyben minden automata azonos működésű, és az állapot–átmenetek csak az adott automata „szomszédjainak" állapotától függenek. A hálózat párhuzamosan működik, szinkronizáltan és diszkrét lépésekben.

A könyvrészlet pontosan definiálja a sejtautomaták fogalmát, példákat ad rá. Részletesebben foglalkozik két speciális esettel: a sokak által ismert életjátékkal és a lövész–osztagok szinkronizációs problémájával. Végül megmutatja azt a rendkívül fontos elméleti tételt, hogy a sejtautomaták és a Turing–gépek képesek egymás hatékony szimulációjára, illetve bevezeti a reverzibilis sejtautomaták fogalmát. A fejezetet feladatok zárják, néhány érdekesebb feladat megoldásának ismertetése hozzátartozik a dolgozat bemutatásához.