Rendiconti di Matematica, Serie VII
Volume 28, Roma (2008), 201–228

Sulla struttura combinatoria dei linguaggi bounded

FLAVIO D'ALESSANDRO – ALESSANDRA ZINNO PILO

ABSTRACT: *This paper concerns the studying of some algebraic and algorithmical problems on the growth function of formal languages.*

Proporremo una rassegna sintetica di alcuni argomenti relativi alla teoria delle funzioni di crescita dei linguaggi formali. Come suggerisce il titolo della nota, la sua redazione è essenzialmente legata ad una classe significativa di linguaggi, detti *bounded*, di cui analizzeremo le principali proprietà strutturali ed i legami con altre tematiche di interesse per la matematica e per l'informatica. Il taglio editoriale di questa nota può essere così rapidamente descritto: gli aspetti tecnici sono presentati in modo contenuto, privilegiando, nella misura maggiore possibile, le idee soggiacenti ai teoremi e alle loro dimostrazioni, nonché alle relazioni con altri aspetti interessanti della teoria. Alcune definizioni sono volutamente omesse, altre, invece, sinteticamente presentate. La speranza è che chi abbia dimistichezza con la matematica approfitti senza difficoltà di questo breve excursus in una tematica di così grande interesse e attualità per l'informatica.

1 – Introduzione

Sia L un linguaggio. La *funzione di conteggio* di L, detta anche *funzione di struttura*, è la funzione f_L che associa, ad ogni intero non negativo n, il numero $f_L(n)$ delle parole di L di lunghezza uguale ad n. La *funzione di crescita* di L è

KEY WORDS AND PHRASES: *Bounded language – Context-free language – Growth function.*
A.M.S. CLASSIFICATION: 68Q45

invece la funzione g_L che associa, ad ogni intero non negativo n, il numero $g_L(n)$ delle parole di L di lunghezza uguale o inferiore ad n.

Queste funzioni sono di grande interesse poiché lo studio del loro andamento asintotico fornisce preziose informazioni sulla struttura combinatoria del linguaggio considerato. Le nozioni di funzione di struttura e di funzione di crescita giocano un ruolo importante nello studio di problemi significativi formulati in altri ambiti distinti da quello della teoria dei linguaggi. Ad esempio, nella teoria matematica dell'informazione, due risultati fondamentali sui codici di parole a lunghezza variabile sono basati su tale nozione (*cfr.* [4]): la Disuguaglianza di Kraft-Mc Millan che fornisce una condizione necessaria affinché un insieme di parole sia codice oppure il teorema di Kraft relativo alla esistenza di codici prefissi ottimali per una data sorgente. Un altro notevole esempio ci è fornito dalla teoria dei gruppi dove un celeberrimo teorema dimostrato da M. Gromov in [20] permette di caratterizzare la struttura algebrica dei gruppi di crescita polinomiale: precisamente, un gruppo siffatto possiede un sottogruppo nilpotente di indice finito. Altri risultati profondi sono stati ottenuti nello studio della crescita delle algebre finitamente generate relativamente alla dimensione di Gelfand-Kirillov di tali algebre. Rimandiamo al testo [15] quale eccellente e completo riferimento bibliografico per questo argomento.

Nella teoria dei linguaggi formali, questa tematica è stata oggetto, anche recentemente, di un'attività di ricerca molto intensa con particolare riferimento ai linguaggi a crescita polimoniale (*cfr.* [6], [8], [9], [10], [13], [14], [16], [17], [19], [22], [23], [25], [26], [27], [29], [30], [32], [36]). Il punto di partenza dell'excursus che compiremo in questa nota è un quesito formulato da Flajolet in [16] relativo alla esistenza di linguaggi context-free di crescita intermedia. A tale quesito è stata data risposta negativa da Incitti in [30] e, qualche tempo dopo, indipendentemente, da Bridson e Gilman in [6], dimostrando la seguente proprietà di *gap*: *un linguaggio context-free ha crescita esponenziale oppure polinomiale.*

I linguaggi a crescita polinomiale, detti anche *sparsi*, svolgono un ruolo chiave sia nell'ambito della Teoria dei linguaggi formali, sia in quello della Complessità di calcolo. Questa circostanza è anche riconducibile al fatto che, nel caso context-free, la famiglia di tali linguaggi coincide con quella dei cosiddetti *linguaggi bounded* introdotti e studiati, per la prima volta, da S. Ginsburg nella seconda metà degli Anni 60 (*cfr.* [17]). Diversi risultati di questa ricca e ben sviluppata teoria sono stati riscoperti, anche in ambiti diversi da quello dei linguaggi, (e ripubblicati) più volte negli ultimi venti anni e uno degli scopi della nota è quello di mettere in evidenza alcuni aspetti, a nostro giudizio, rilevanti di questo affascinante argomento.

Eccoci infine, molto brevemente, a spiegare come si articolerà questa nota. Dopo aver presentato alcune definizioni preliminari, dimostreremo il teorema di gap e analizzeremo, sia dal punto di vista algebrico, sia da quello algoritmico, la struttura dei linguaggi context-free a crescita polinomiale. Ciò sarà fatto nelle Sezioni 2 e 3. Nella Sezione 4, invece, descriveremo un algoritmo il quale consente di calcolare in modo esatto la funzione di conteggio (e dunque anche quella di crescita) di un linguaggio context-free a crescita polinomiale. La Sezione 5 sarà invece dedicata alla descrizione del legame esistente fra queste tematiche ed un importante problema dell'Algebra: il *problema di Burnside per i semigruppi*. La Sezione 6 sarà dedicata ad un argomento "di frontiera": descriveremo una classe di linguaggi che pur avendo, nel senso che preciseremo poi, una struttura vicina a quella dei linguaggi context-free, sono a crescita intermedia. L'ultima sezione sarà invece dedicata alla presentazione di alcuni risultati ottenuti recentemente sulla funzione di conteggio delle relazioni razionali.

2 – Preliminari

Presenteremo molto brevemente, in questa sezione, un vocabolario minimo di definizioni soprattutto nell'intento di fissarne le corrispondenti notazioni. Quanto segue è in genere materia di un corso fondazionale di teoria dei linguaggi formali e può trovarsi nei testi, alcuni classici ed altri più recenti, di un'ampia rosa di scelta (*cfr.* [3], [12], [15], [17], [24], [33], [40]) alla quale rinviamo il lettore interessato.

2.1 – Monoidi liberi

Se A è un alfabeto, denoteremo con A^* il monoide libero delle parole, dette anche stringhe, sull'alfabeto A e con 1_{A^*}, o più semplicemente con 1, il suo elemento identità; tale elemento è detto *parola vuota*. L'insieme $A^* \setminus \{1\}$ è denotato A^+. Se u è una parola di A^*, la sua *lunghezza* è l'intero non negativo, denotato con il simbolo $|u|$, uguale a zero se $u = 1$ e, altrimenti, al numero dei simboli di cui si compone u. Fissata una lettera $a \in A$, denotiamo con $|u|_a$ il numero di occorrenze di a nella parola u. Siano u e w parole di A^*. La parola u si dice *fattore* di w se esistono λ, $\mu \in A^*$ tali che $\lambda u \mu = w$. L'insieme dei fattori di w è denotato con il simbolo $F(w)$. Similmente, se L è un insieme di parole di A^*, scriveremo $F(L) = \bigcup_{w \in L} F(w)$. u si dice poi *prefisso* di w se esiste $\mu \in A^*$ tale che $u\mu = w$. Infine u si dice *suffisso* di w se esiste $\lambda \in A^*$ tale che $\lambda u = w$. Una parola z è detta *primitiva* se $z = u^n$, con $n \geq 1$, implica $z = u$ e quindi $n = 1$. Ricordiamo infine che una congruenza di A^* è una relazione di equivalenza di A^* compatibile con l'operazione di prodotto di parole.

2.2 – Linguaggi context-free

Descriveremo ora in maggior dettaglio un'importante classe di linguaggi, quella dei *linguaggi context-free,* detti anche *liberi dal contesto*, introdotti in [11]. Questi linguaggi sono definiti tramite l'uso di certe strutture combinatorie dette *grammatiche context-free* di cui richiameremo la definizione anche al fine di fissare qualche utile notazione. Formalmente, una grammatica context-free è una quadrupla $G = (V, T, P, S)$, dove:

- V è un insieme finito di oggetti detti *variabili* della grammatica G;
- T è un insieme finito di simboli detti *terminali* della grammatica G, ed inoltre T è disgiunto da V;
- P è un sottoinsieme finito dell'insieme $V \times \{V \cup T\}^*$, detto delle *produzioni* della grammatica G; ogni produzione è usualmente denotata con la forma $A \to \alpha$ dove A è una variabile ed α è una stringa di simboli di $(V \cup T)^*$;
- S è una variabile detta *start symbol*, o *assioma della grammatica.*

Vogliamo ora definire il linguaggio generato da G. A tale scopo definiamo prima le seguenti due relazioni $\Rightarrow$ e $\stackrel{*}{\Rightarrow}$ tra stringhe in $(V \cup T)^*$. Se u, v sono parole in $(V \cup T)^*$, allora porremo

$$u \Rightarrow v$$

se esiste una produzione $A \to \beta$ della grammatica G tale che $u = \alpha A \gamma$ e $v = \alpha\beta\gamma$, dove $\alpha,\ \gamma \in (V \cup T)^*$. Le parole $\alpha,\ \gamma$ sono chiamate *contesti* di u e v e la relazione $\Rightarrow$ è detta di *derivazione atomica.* La relazione $\stackrel{*}{\Rightarrow}$ è poi definita quale chiusura transitiva e riflessiva di $\Rightarrow$, ovvero $\stackrel{*}{\Rightarrow} = \bigcup_{i \geq 0} \stackrel{i}{\Rightarrow}$. Questa relazione è detta di *derivazione* di G. In altre parole, se $\alpha,\ \beta \in (V \cup T)^*$ sono tali che $\alpha \Rightarrow^* \beta$, allora $\alpha = \beta$ oppure esistono parole $\alpha_0, \alpha_1, \ldots, \alpha_n, n \geq 1$ di $(V \cup T)^*$ tali che

$$\alpha = \alpha_0 \Rightarrow \alpha_1 \Rightarrow \cdots \alpha_{n-1} \Rightarrow \alpha_n = \beta.$$

Una qualsiasi stringa α tale che $S \stackrel{*}{\Rightarrow} \alpha$ si dirà *forma sentenziale* di G. Una variabile A si dice *utile* se esiste una parola α in T^* tale che $S \stackrel{*}{\Rightarrow} fAg \stackrel{*}{\Rightarrow} \alpha$, dove $f, g \in (V \cup T)^*$. Il motivo per cui queste grammatiche sono chiamate context-free, letteralmente "libere dal contesto", risiede nel fatto che, in un passo qualsiasi di una generica derivazione di G, l'applicazione di una produzione non dipende dalla scelta dei contesti della forma sentenziale a cui si applica. Infine il linguaggio generato da G, denotato con il simbolo $L(G)$, è l'insieme:

$$L(G) = \{w \in T^* \mid S \stackrel{*}{\Rightarrow} w\}.$$

Diremo che un linguaggio è *context-free* se esiste una grammatica context-free che lo genera. Diremo inoltre che due grammatiche context-free sono equivalenti se i linguaggi da esse generati coincidono. Ricordiamo infine che tali linguaggi sono anche detti *algebrici* poiché possono essere descritti quali componenti delle soluzioni di sistemi di equazioni polinomiali di parole, cioè di equazioni di parole in cui siano usati i soli operatori di unione e di concatenazione in A^*.

2.3 – Linguaggi regolari

Un'altra importante famiglia di linguaggi formali è quella dei cosiddetti *linguaggi regolari.* Tale famiglia di linguaggi corrisponde, rispetto alla classica gerarchia di Chomsky-Schützenberger, alla classe di linguaggi immediatamente precedente a quella dei linguaggi algebrici di cui ne costituisce una notevole sottoclasse. Un linguaggio su di un alfabeto A si dice *regolare* o *razionale* se può ottenersi a partire da linguaggi di cardinalità finita di A^*, tramite l'applicazione, in un numero finito di volte, delle operazioni razionali, cioè delle classiche operazioni di unione e di prodotto di linguaggi e di una terza operazione, questa più complessa, detta di *stella*, che associa ad ogni linguaggio L il sottomonoide L^* da esso generato. La famiglia di tali linguaggi è usualmente denotata con il simbolo $\mathrm{Rat}(A^*)$. In virtù della sua definizione, essa coincide con la più piccola famiglia di linguaggi di A^* contenente la famiglia dei linguaggi finiti e chiusa rispetto alle operazioni razionali. Due risultati fondamentali della teoria, entrambi dimostrati negli Anni '50, forniscono importanti caratterizzazioni dei linguaggi regolari. Il primo, dimostrato dal celebre logico americano Stephen Coole Kleene, è legato ad un modello di calcolo ben noto, quello di *automa a stati finiti*, e stabilisce che un linguaggio è regolare se e solo se esiste un automa a stati finiti che accetta (o riconosce) tale linguaggio. Il secondo teorema, dimostrato da John Myhill e Anil Nerode, costituisce una caratterizzazione algebrica di tali linguaggi basata sulle congruenze. A questo proposito, converrà ricordare che, dati una congruenza θ di A^* ed un linguaggio L, si dice che θ satura L, oppure che L è saturato da θ, se L è unione di classi di θ. Il teorema di Myhill e Nerode stabilisce che un linguaggio è regolare se e solo se esiste una relazione di congruenza di A^* di indice finito che satura il linguaggio. Come si è detto prima, i linguaggi regolari sono context-free: infatti essi sono generati da grammatiche context-free di tipo particolare, dette *lineari a destra* oppure *lineari a sinistra.* Ricordiamo che una grammatica context-free si dice lineare a destra (risp. lineare a sinistra) se ogni sua produzione $A \to \alpha$ è tale che α contiene, al più, una occorrenza di una variabile della grammatica e, se questo è il caso, tale variabile deve essere prefisso (risp. suffisso) di α. I teoremi ora ricordati mostrano dunque che la nozione di linguaggio regolare è robusta.

Ricordiamo infine un importante risultato, detto *Pumping Lemma*, che fornisce una condizione necessaria affinché un linguaggio sia regolare. Precisamente, se L è un linguaggio regolare su di un alfabeto A, esiste un intero non negativo n dipendente da L, tale che, per ogni parola $w \in L$ con $|w| \geq n$, w ammette una fattorizzazione del tipo $w = \alpha u \beta$ dove $\alpha, u, \beta \in A^*$, $u \neq \epsilon$ e, per ogni $i \geq 0$, $\alpha u^i \beta$ è una parola di L.

2.4 – Funzione di conteggio e funzione di crescita di un linguaggio

Siano A^* il monoide libero delle parole su di un dato alfabeto finito A ed L un sottoinsieme di A^*. La *funzione di struttura*, detta anche *funzione di conteggio* di L, è la funzione

$$f_L : \mathbb{N} \longrightarrow \mathbb{N},$$

così definita:

$$\forall\ n \in \mathbb{N}, \quad f_L(n) = \mathrm{Card}(L \cap A^n) = \mathrm{Card}(\{u \in L \mid |u| = n\}).$$

Si definisce poi *funzione di crescita* di L, la funzione

$$g_L : \mathbb{N} \longrightarrow \mathbb{N},$$

così definita:

$$\forall\ n \in \mathbb{N}, \quad g_L(n) = \mathrm{Card}(\{u \in L \mid |u| \leq n\}).$$

È facile osservare che, per ogni $n \in \mathbb{N}$, $g_L(n) = \sum_{i=0,\dots,n} f_L(i)$. È di grande utilità considerare una classificazione, riportata nella definizione seguente, dei linguaggi formali basata sull'andamento asintotico della funzione di crescita.

Definizione 1. Sia $L \subseteq A^*$. Si dice che L è *fino* o *thin* se:

$$\forall\ n \geq 0, \qquad f_L(n) \leq 1.$$

Si dice invece che L è *snello* o *slender* se esiste un intero $C \geq 0$ tale che:

$$\forall\ n \geq 0, \qquad f_L(n) \leq C.$$

Si dice che L ha *crescita polinomiale* o è *poly-slender* se esiste $k > 0$ tale che:

$$\forall\ n \geq 0, \qquad g_L(n) \leq n^k.$$

Un linguaggio a crescita polinomiale è anche detto *sparso*.

Si dice che L è *quasi-polinomiale* se:

$$\lim_{n \to +\infty} \frac{f_L(n)}{(\mathrm{Card}(A))^n} = 0.$$

Si dice che L ha *crescita esponenziale* se esistono $k > 1$ ed un intero $n_0 \geq 0$ tali che:

$$\forall\ n \geq n_0, \qquad g_L(n) \geq k^n.$$

Semplici esempi di linguaggi a crescita polinomiale ed esponenziale sono i seguenti. Sia

$$L = \{a^n b^n \mid n \geq 1\}.$$

È facile osservare che la funzione di conteggio di L è definita come:

$$\forall\ n \geq 0, \qquad f_L(n) = \begin{cases} 1 & \text{se n è pari} \\ 0 & \text{se n è dispari.} \end{cases} \tag{1}$$

Di conseguenza L è fino. In particolare, per ogni $n \geq 0$, $g_L(n) \leq n$, cioè L ha crescita lineare. Invece, come facilmente si verifica, i linguaggi

$$L = A^*, \qquad L' = \{ww \mid w \in A^*\}$$

sono tali che le rispettive funzioni di conteggio e di crescita sono esponenziali.

3 – Funzione di crescita dei linguaggi context–free

Intendiamo ora presentare alcuni notevoli risultati relativi alla struttura combinatoria dei linguaggi context-free: il primo riguarda una caratterizzazione dei linguaggi context-free a crescita polinomiale. Tale caratterizzazione è legata ad una importante definizione, quella di *linguaggio bounded*, che daremo nel seguito di questa sezione. Vedremo successivamente il teorema di *gap*, enunciato nell'Introduzione, per la funzione di crescita dei linguaggi context-free: la funzione di crescita di tali linguaggi è esponenziale oppure polinomiale. Mostreremo infine che il teorema di gap è effettivo: esiste un algoritmo, l'esecuzione del quale consente di decidere la natura, esponenziale oppure polinomiale, della funzione di crescita di un linguaggio siffatto. Nell'intento di dimostrare i risultati appena ricordati, riteniamo opportuno introdurre qualche definizione e presentare alcuni lemmi preliminari: cominceremo con la nozione, introdotta da Ginsburg in [17], di linguaggio dei cicli destri e sinistri associati ad una variabile della grammatica context-free.

Definizione 2. Siano $G = (V, T, P, S)$ una grammatica context-free, $L(G)$ il linguaggio da essa generato ed A una variabile di G. Denotiamo con L_A ed R_A i due linguaggi di parole sull'alfabeto T definiti come segue:

$$L_A = \{\alpha \in T^* \ \mid \ \exists\ \beta \in T^* \ \mid \ A \overset{*}{\Rightarrow} \alpha A \beta\},$$
$$R_A = \{\beta \in T^* \ \mid \ \exists\ \alpha \in T^* \ \mid \ A \overset{*}{\Rightarrow} \alpha A \beta\}.$$

L_A ed R_A prendono rispettivamente i nomi di *linguaggio dei cicli sinistri* e *linguaggio dei cicli destri della variabile* A.

È facile verificare che, per ogni variabile A di G, L_A ed R_A sono sottomonoidi di T^*. In effetti, se α, α' appartengono ad L_A, esistono $\beta, \beta' \in T^*$ tali che:

$$A \overset{*}{\Rightarrow} \alpha A \beta \text{ e } A \overset{*}{\Rightarrow} \alpha' A \beta'.$$

Poiché $A \overset{*}{\Rightarrow} \alpha A \beta \Rightarrow \alpha\alpha' A \beta\beta'$, si ha che $\alpha\alpha'$ appartiene ad L_A, cioè L_A è chiuso rispetto all'operazione di concatenazione di parole, e dunque L_A è un sottomonoide di T^*. Nello stesso modo, si dimostra che R_A è un sottomonoide di T^*. Il lemma seguente fornisce una proprietà combinatoria cruciale delle grammatiche, basata sulla definizione precedente, la quale assicura la crescita esponenziale dei linguaggi da esse generati.

Lemma 1. *Siano $G = (V, T, P, S)$ una grammatica context-free, $L = L(G)$ il linguaggio da essa generato ed A una variabile di G. Se d_1, $d_2 \in L_A$ (risp. $d_1, d_2 \in R_A$) sono parole tali che il sottomonoide $\{d_1, d_2\}^*$ è libero in T^*, allora L ha crescita esponenziale.*

Dim. Siano d_1, d_2 parole di L_A. Allora esistono parole $\alpha, \beta \in T^*$ tali che:

$$A \stackrel{*}{\Rightarrow} d_1 A\alpha \text{ e } A \stackrel{*}{\Rightarrow} d_2 A\beta.$$

Poiché la variabile A è utile, segue che esiste una derivazione della forma:

$$S \stackrel{*}{\Rightarrow} uAv \stackrel{*}{\Rightarrow} z, \text{ dove } u, v, z \in T^*.$$

Poniamo $|u| = N$, $|v| = M$, $|z| = Q$, $|d_1| = n_1$, $|d_2| = n_2$. Possiamo suppore che $n_1 = n_2 = n$ e $|\alpha| = |\beta| = m$, non essendo questa una restrizione. Infatti, nell'ipotesi in cui $n_1 \neq n_2$ (oppure $|\alpha| \neq |\beta|$), sarà sufficiente considerare il sottomonoide di $\{d_1, d_2\}^*$ generato dall'insieme $\{d_1d_2, d_2d_1\}$. Tale sottomonoide è ovviamente libero e le due parole che lo generano hanno la stessa lunghezza.

Si ha:

$$S \stackrel{*}{\Rightarrow} uAv \stackrel{*}{\Rightarrow} udA\gamma v \Rightarrow udz\gamma v, \tag{2}$$

dove $d \in \{d_1, d_2\}^*$ e $\gamma \in \{\alpha, \beta\}^*$.

Se $w = udz\gamma v$, si ha che $w \in L$ dato che $w \in T^*$ e $S \stackrel{*}{\Rightarrow} w$. Si ottiene:

$$|w| = |u| + |d| + |z| + |\gamma| + |v| = N + Q + M + |d| + |\gamma|.$$

Ponendo $N + Q + M = c$, possiamo scrivere:

$$|w| = c + |d| + |\gamma| = c + kn + km = c + k(n + m),$$

dove k è il numero di occorrenze delle parole d_1 e d_2 nella parola d ed è, ovviamente, anche il numero di occorrenze delle parole α e β nella parola γ. Indichiamo con L' l'insieme delle parole ottenute come nella Equazione (2) e osserviamo che $L' \subseteq L$. Sia g_L la funzione di crescita di L. Per ogni intero positivo i sufficientemente grande, si ha:

$$g_L(i) \geq g_{L'}(i) \geq 2^{\frac{i-c}{n+m}}.$$

Dalla condizione precedente segue che la funzione di crescita di L è esponenziale. La dimostrazione per R_A è simmetrica.

Nel seguito di queste pagine, dati due interi i e j, l'insieme degli interi compresi tra i e j sarà indicato col simbolo $[i, j]$.

Definizione 3. Sia $G = (V, T, P, S)$ una grammatica context-free e sia $V = \{S = S_1, S_2, \ldots, S_n\}$. Definiamo l'insieme V' ponendo $V' = \emptyset$ se $n = 1$, e $V' = \{S_2, \ldots, S_n\}$ se $n > 1$. Definiamo poi, per ogni $k = 1, \ldots, n$, la grammatica $G_k = (V', T, P_k, S_k)$ le cui produzioni sono quelle di G della forma:

$$A \to \alpha, \text{ con } A \in V' \cup \{S_k\} e\ \alpha \in (V' \cup T)^*.$$

Denotiamo con $L'(G)$ il sottoinsieme di $L(G)$ costituito dalle parole $w \in T^*$ che ammettono una derivazione:

$$S \Rightarrow w_1 \cdots \Rightarrow w_m = w,$$

tale che, per ogni $i = 1, \dots, m$, $w_i \in (V' \cup T)^*$. È facile osservare che, per $n = 1$, $L'(G)$ è il linguaggio finito delle parole di $L(G)$ che possono essere ottenute, a partire da S_1, mediante un solo passo di derivazione. Enunciamo ora, senza dimostrarlo, un lemma che utilizzeremo nella dimostrazione del prossimo teorema.

Lemma 2 ([30] Lemma 3.6). *$L(G) \subseteq L_S L'(G) R_S$.*

Siamo ora in grado di dimostrare una significativa caratterizzazione dei linguaggi context-free a crescita polinomiale. Tale caratterizzazione è legata ad una importante definizione, quella di *linguaggio bounded*.

Definizione 4. Siano A un alfabeto, L un linguaggio di A^* ed n un intero positivo. Il linguaggio L è detto *n-bounded* se esistono parole $u_1, \dots, u_n \in A^+$ tali che:

$$L \subseteq \{u_1\}^* \cdots \{u_n\}^*.$$

L è detto *bounded* se esiste un intero n tale che L è *n-bounded*.

È utile osservare che, dalla definizione precedente, si ha che la famiglia di tali linguaggi è chiusa rispetto alle operazioni di unione insiemistica e di prodotto di linguaggi.

La nozione di linguaggio bounded gioca un ruolo importante nello studio di diversi problemi di Matematica e di Informatica. In questo ultimo ambito l'interesse di tali linguaggi risiede, ad esempio, nel fatto che tutte le più significative proprietà dei linguaggi formali sono ricorsivamente decidibili rispetto alla famiglia dei linguaggi context-free bounded quando, in generale, non lo sono. La struttura combinatoria e le proprietà algoritmiche dei linguaggi context-free bounded sono state studiate approfonditamente da S. Ginsburg in [17]. Inoltre, in un lavoro del 1984 [38], A. Restivo e C. Reutenauer, nell'ambito dello studio del *problema di Burnside per i semigruppi*, hanno introdotto una condizione di finitezza, volta cioè ad assicurarne la finitezza del sostegno, per i semigruppi periodici, detta *proprietà di permutazione*, strettamente legata ai linguaggi bounded. Approfondiremo quest'ultimo argomento nella prossima sezione.

Teorema 1. *Sia G una grammatica context-free ed $L = L(G)$ il linguaggio da essa generato. Le seguenti condizioni sono equivalenti:*

1. *L ha crescita polinomiale*;
2. *Per ogni variabile A di G, L_A e R_A sono linguaggi* 1*-bounded*;
3. *L è un linguaggio bounded.*

Prima di dimostrare il Teorema 1, converrà ricordare che l'equivalenza delle condizioni (2) e (3) è stata dimostrata da S. Ginsburg nella prima metà degli anni sessanta (*cfr.* [17]), mentre l'equivalenza delle condizioni (1) e (3) è stata dimostrata da Latteux e Thierrin in [32] e ridimostrata successivamente in [27] e [36]. Per la dimostrazione dell'implicazione (2) $\Longrightarrow$ (3) utilizzeremo un argomento combinatorio descritto in [30].

DIM. ($1 \Rightarrow 2$).

Sia A una variabile di G e sia L_A il linguaggio dei cicli sinistri associato ad A.

Se $L_A = \{1_{T^*}\}$ allora L_A è banalmente un linguaggio 1-bounded. Siano ora u e z una parola non vuota di L_A e la sua radice primitiva. Supponiamo per assurdo che L_A non sia 1-bounded. Deve allora esistere una parola $v \in (L_A \setminus \{z\}^*)$. Consideriamo il sottomonoide $\{u, v\}^*$ di L_A. Per il *Teorema del difetto*, esso deve essere libero, e per il Lemma 1, L ha crescita esponenziale, così contraddicendo la condizione (1). Dunque L_A è un linguaggio 1-bounded. Lo stesso argomento consente di dimostrare che R_A è un linguaggio 1-bounded. L'implicazione ($1 \Rightarrow 2$) è dunque dimostrata.

($2 \Rightarrow 3$).

Dimostriamo il risultato per induzione sul numero n di variabili della grammatica G. Sia $n = 1$. Il linguaggio $L'(G)$ è finito e dunque bounded. Per ipotesi, L_S e R_S sono linguaggi 1-bounded e, dunque, si ha che $L_S L'(G) R_S$ è bounded. Per il Lemma 2, poiché $L \subseteq L_S L'(G) R_S$, ne segue infine che L è bounded. La base del procedimento induttivo è pertanto dimostrata.

Dimostriamo ora il passo induttivo. Osserviamo intanto che, per ipotesi induttiva, $L(G_k)$ è bounded, per ogni $k \in [2, n]$. In virtù del Lemma 2 e dal momento che, per ipotesi, L_S ed R_S sono linguaggi 1-bounded è sufficiente dimostrare che $L'(G)$ è un linguaggio bounded. Sia $W = \{\alpha_1, \ldots, \alpha_m\}$ l'insieme di parole sull'alfabeto $(V' \cup T)$ tali che, per ogni $i = 1, \ldots, m$, $S \to \alpha_i$ sono produzioni della grammatica G. Quindi, per ogni $i = 1, \ldots, m$, si avrà

$$\alpha_i = r_{1,i} S_{k_{1,i}} \cdots r_{x_i,i} S_{k_{x_i},i} r_{x_i+1,i},$$

dove $r_{1,i}, \ldots, r_{x_i+1,i} \in T^*$ e $S_{k_1,i}, \ldots, S_{k_{x_i},i} \in V'$.

Sia ora $w \in L'(G)$. Allora w sarà ottenuta tramite una derivazione del tipo:

$$S \Rightarrow w_1 \ldots \Rightarrow w_l = w,$$

dove, per ogni $j = 1, \ldots, l$, $w_j \in (V' \cup T)^*$.

Osserviamo che $w_1 \in W$ e che nelle stringhe $w_1, \ldots, w_l$ non occorre mai la variabile S. Di conseguenza si avrà:

$$w \in r_{1,i} L(G_{k_1,i}) \cdots r_{x_i,i} L(G_{k_{x_i},i}) r_{x_i+1,i},$$

e dunque:

$$L'(G) \subseteq \bigcup_{i=1}^{m} r_{1,i} L(G_{k_1,i}) \cdots r_{x_i,i} L(G_{k_{x_i},i}) r_{x_i+1,i}. \tag{3}$$

Osserviamo che, per ogni $k > 0$, l'insieme delle produzioni della grammatica G_k è un sottoinsieme dell'insieme delle produzioni della grammatica G; pertanto l'ipotesi definita dalla condizione (2) dell'enunciato del teorema vale ancora se in luogo di considerare la grammatica G si considera la grammatica G_k. Applicando l'ipotesi induttiva alla grammatica G_k si avrà allora che $L(G_k)$ è un linguaggio bounded. Dalla Equazione (3) segue dunque che $L'(G)$ è un linguaggio bounded. L'implicazione $(2 \Rightarrow 3)$ è pertanto dimostrata.

$(3 \Rightarrow 1)$.

Dal momento che L è bounded, esiste un intero $q \geq 1$ e parole non vuote $u_1, \ldots, u_q$ tali che $L \subseteq \{u_1\}^* \cdots \{u_q\}^*$. Sia $w \in L$ tale che $|w| \leq n$.

Allora

$$w = {u_1}^{t_1} \cdots {u_q}^{t_q},$$

dove per ogni $i \in [1, q]$, $t_i \geq 0$ e $t_1 + \cdots + t_q \leq n$.

Indichiamo con $\phi(q, n) = \mathrm{Card}(\{(t_1, \ldots, t_q) \mid t_1 + \cdots + t_q \leq n\})$.

Osserviamo ora che, per ogni $n \geq 0$, $g_L(n) \leq \phi(q, n)$. Poiché la funzione $\phi(q, n)$ è un polinomio nella variabile n assumendo che q sia costante, ne segue che la funzione g_L è limitata superiormente da una funzione polinomiale, ovvero L ha crescita polinomiale.

Come conseguenza del Teorema 1 si ha il corollario seguente.

Teorema 2. *La funzione di crescita di un linguaggio context-free è esponenziale oppure polinomiale.*

Dim. Sia L un linguaggio context-free la cui funzione di crescita non è polinomiale. In virtù del teorema precedente, esiste una variabile A della grammatica che genera il linguaggio tale che L_A oppure R_A non sono linguaggi 1-bounded. Supponiamo, per semplicità che ciò accada per L_A. Esistono allora parole non vuote $u, v \in L_A$ le cui radici primitive sono distinte. Per il *Teorema del difetto*, il sottomonoide $\{u, v\}^*$ è libero in L_A e, per il Lemma 1, L ha crescita esponenziale.

Converrà ricordare che, benché il Teorema 2 sia un corollario immediato del Teorema 1, il suo enunciato è stato dimostrato esplicitamente da Incitti in [30] e poco tempo più tardi da Bridson e Gilman in [6]. Esso in particolare fornisce la risposta negativa ad un quesito posto da Flajolet in [16] circa l'esistenza di linguaggi context-free a crescita intermedia.

Intendiamo chiudere questa sezione presentando un risultato dimostrato da T. Ceccherini e W. Woess in [7]. Se L è un linguaggio sull'alfabeto A, il *tasso di crescita* di L è il numero

$$\gamma(L) = \limsup_{n\to\infty} \mathrm{Card}(\{w \in L \mid |w| = n\})^{\frac{1}{n}}.$$

Se L è infinito allora $\gamma(L)$ è tale che $0 \le \gamma(L) \le \mathrm{Card}(A)$. Sia F un insieme di parole tale che $F \subseteq F(L)$ e sia

$$L_F = \{u \in L \mid F(u) \cap F = \emptyset\}.$$

L'insieme F è detto delle *parole proibite* e, nel caso in cui $\gamma(L_F) < \gamma(L)$, diremo che il tasso di crescita di L è sensibile rispetto ad F. Il problema studiato concerne le condizioni che assicurano che $\gamma(L_F) < \gamma(L)$. Converrà osservare che questo problema è interessante soltanto nel caso in cui L ha crescita esponenziale, ovvero nel caso in cui $\gamma(L) > 1$. Infatti se $\gamma(L) = 1$ allora $\gamma(L_F) = 1$ oppure $\gamma(L_F) = 0$ ed in tal caso L_F è finito. Il risultato principale ottenuto in [7] concerne i linguaggi context-free ergodici. Sarà conveniente, a tale proposito, presentare questa nozione. Consideriamo una grammatica context-free ridotta in cui, cioè, ogni variabile è utile. Ad essa è possibile associare un grafo orientato definito nel modo seguente: l'insieme dei suoi vertici è l'insieme delle variabili della grammatica mentre, dati due vertici V e W, la coppia (V, W) è un arco del grafo se esiste una produzione $V \to \alpha$ della grammatica tale che W è fattore di α. Una grammatica si dice *ergodica* se il grafo ad essa associato ha almeno un arco ed è fortemente connesso. Un linguaggio si dice poi *ergodico* se esiste una grammatica ergodica che lo genera. Il risultato principale dimostrato in [7] è il seguente: *ogni linguaggio context-free, non ambiguo, non lineare ed ergodico è tale che il suo il tasso di crescita è sensibile rispetto ad un qualsiasi insieme di parole.*

3.1 – Un algoritmo di decisione

In questo paragrafo presenteremo un algoritmo che permette di decidere se la funzione di crescita di un linguaggio context-free è esponenziale o polinomiale. In virtù dei due teoremi precedenti, si tratta di dimostrare la ricorsiva decidibilità della proprietà di essere bounded rispetto alla famiglia dei linguaggi context-free. Questo problema è stato affrontato e risolto da Ginsburg in [17]; descriveremo pertanto l'algoritmo da lui proposto. A tale proposito, converrà ricordare che, dato un linguaggio context-free, esso può essere sempre generato da una grammatica di tipo opportuno, detta *in forma normale di Chomsky*, nella quale tutte le produzioni sono della forma $A \to BC$ oppure $A \to a$ dove A, B e C sono variabili della grammatica ed a è un suo simbolo terminale. La procedura proposta da Ginsburg è essenzialmente basata sul lemma seguente con il quale si dimostra che, per ogni variabile A di una grammatica context-free data, i linguaggi L_A ed R_A sono context-free.

LEMMA 3. *Sia $G = (V, T, P, S)$ una grammatica context-free in forma normale di Chomsky. Per ogni variabile A di G esistono grammatiche context-free G_A e G'_A tale che $L_A = L(G_A)$ e $R_A = L(G'_A)$.*

TEOREMA 3. *È possibile decidere se la funzione di crescita di un linguaggio context-free è polinomiale o esponenziale.*

DIM. Siano $G = (V, T, P, S)$ una grammatica context-free, n il numero delle variabili di G, $L = L(G)$ il linguaggio da essa generato e g_L la funzione di crescita di L. Dal momento che g_L è polinomiale o esponenziale, è sufficiente verificare se g_L è polinomiale o meno. Inoltre, dal teorema di caratterizzazione dei linguaggi context-free a crescita polinomiale, segue che g_L è polinomiale se e solo se, per ogni variabile A di G, L_A ed R_A sono linguaggi 1-bounded. Quindi è sufficiente costruire una procedura che permetta di decidere se L_A (rispettivamente R_A) sia un linguaggio 1-bounded. Costruiamo allora tale procedura per il linguaggio L_A, essendo simmetrico il procedimento di decisione per il linguaggio R_A. Dal lemma precedente segue che, a partire dalla grammatica G, possiamo costruire una grammatica context-free G_A tale che $L_A = L(G_A)$. Utilizzando un argomento standard è possibile verificare se L_A contenga o meno una parola diversa dalla parola vuota. Se $L_A \subseteq \{1_{T^*}\}$ allora L_A è banalmente un linguaggio 1-bounded e la procedura termina.

Supponiamo, invece, che esista una parola non vuota $u \in L_A$ e sia z la radice primitiva di u. Chiaramente L_A è un linguaggio 1-bounded se e solo se $L_A \subseteq \{z\}^*$, cioè, se e solo se $M \neq \emptyset$ dove $M = L_A \cap (T^* \setminus \{z\}^*)$. Ora, dal momento che L_A è un linguaggio context-free e $T^* \setminus \{z\}^*$ è un linguaggio regolare, in virtù di un risultato classico dei linguaggi context-free, ovvero che l'intersezione di un linguaggio regolare e di uno context-free è un linguaggio context-free (si osservi, a tale proposito, che l'intersezione di due linguaggi context-free non è, in generale, context-free), segue che M è un linguaggio context-free. Poiché il problema di decidere se una grammatica context-free generi l'insieme vuoto è decidibile ed ogni passo relativo alla costruzione del linguaggio M è effettivo, ne segue che, applicando tale procedura ad M, è possibile decidere se $M = \emptyset$ o meno, cioè se L_A è 1-bounded o meno.

È interessante ricordare che una procedura decisionale più semplice di quella presentata nel Teorema 3 è stata proposta in [8] relativamente alla famiglia dei linguaggi context-free lineari non ambigui.

4 – Calcolo della funzione di conteggio di un linguaggio bounded

In [13] D'Alessandro, Intrigila e Varricchio hanno proposto una tecnica che consente, dato un linguaggio bounded context-free, di descrivere, in modo esatto, la sua funzione di conteggio. Più precisamente, tale procedura di calcolo permette, a partire dalla grammatica context-free che genera il linguaggio bounded, di costruire una famiglia finita di polinomi a coefficienti razionali utile per il calcolo della funzione sopraSu. Vale infatti il risultato seguente.

Teorema 4. *Sia L un linguaggio bounded context-free. Esistono un intero non negativo n_0 ed un insieme finito di polinomi a cofficienti razionali $p_0, \dots, p_{m-1}$ tali che, per ogni $n \geq n_0$,*

$$f_L(n) = p_l(n),$$

dove l è tale che:

$$l \equiv n \bmod m.$$

Nel semplice caso del linguaggio $L = \{a^n b^n \mid n \geq 0\}$, la cui funzione di conteggio è stata definita nella Equazione (1), si ha che tale funzione può essere descritta nel senso del teorema precedente, ponendo $p_0(x) = 1$, $p_1(x) = 0$, $m = 2$, $n_0 = 0$. La dimostrazione del Teorema 4 è effettiva e fornisce un algoritmo per la costruzione, a partire da una grammatica che genera il linguaggio, della famiglia di polinomi atti al computo della sua funzione di conteggio. Presenteremo ora alcuni risultati che derivano in modo immediato dal teorema precedente. A tale proposito, sarà opportuno richiamare alcune definizioni relative all'andamento asintotico di funzioni. Date $f, g : \mathbb{N} \longrightarrow \mathbb{N}$ funzioni definite sull'insieme $\mathbb{N}$, scriviamo

$$f \in \mathcal{O}(g) \quad (\text{risp. } f \in \Omega(g))$$

se esiste un intero positivo n_0 ed un numero reale C tali che, per ogni $n \geq n_0$,

$$f(n) \leq Cg(n) \quad (\text{risp. } f(n) \geq Cg(n)).$$

Corollario 1 ([29]). *Dati un linguaggio context-free L ed un intero positivo k, è decidibile verificare se la funzione di conteggio f_L appartiene alla classe $\mathcal{O}(n^k)$.*

Dim. Per il teorema precedente, partendo dal linguaggio L, è possibile costruire la famiglia dei polinomi $\{p_i\}$ che consente di descrivere la funzione f_L, cosicché si potrà porre

$$\delta = \max\{\delta_i\},$$

dove, per ogni polinomio p_i, δ_i è il suo grado. L'asserto deriva allora dal fatto che

$$f_L \in \mathcal{O}(n^k) \iff \delta \leq k.$$

Analogamente al caso precedente, se γ è il minimo grado dei polinomi della famiglia $\{p_i\}$ prima considerata, dal momento che, per ogni $k \geq 0$,

$$f_L \in \Omega(n^k) \iff \gamma \geq k,$$

si otterrà il corollario seguente:

Corollario 2. *Dati un linguaggio context-free L ed un intero positivo k, è decidibile verificare se la funzione di conteggio f_L appartiene alla classe $\Omega(n^k)$.*

Un'altra conseguenza immediata del teorema è data dal risultato seguente simile al Corollario 13 del lavoro [27].

Corollario 3. *Sia ϵ un numero razionale tale che $0 < \epsilon < 1$. Non esistono linguaggi context-free L tali che*

$$f_L \in \mathcal{O}(n^{k-\epsilon}) \;\; e \;\; f_L \in \Omega(n^{k-1+\epsilon}),$$

dove k è un intero non negativo.

In particolare, il Corollario 3 estende un risultato dimostrato in [29] relativo alla non esistenza di linguaggi context-free la cui funzione sia sub-lineare senza essere limitata superiormente da una costante.

La tecnica combinatoria utilizzata per dimostrare il Teorema 4 è non banale per diverse ragioni: prima di tutto, come è ben noto, esistono linguaggi context-free bounded ambigui per cui una tecnica di conteggio legata alla struttura della grammatica si rivela, in questo contesto, di scarsa utilità. Ricordiamo che una grammatica context-free G si dice *ambigua* se qualche parola in $L(G)$ è il prodotto di due alberi di derivazione distinti. Un linguaggio è poi detto *ambiguo* se ogni grammatica context-free che lo genera è ambigua. Un esempio noto di linguaggio context-free ambiguo è il seguente (*cfr.* [24]):

$$L = \{a^n b^m c^m d^n\}_{n,m \geq 0} \cup \{a^m b^n c^n d^m\}_{n,m \geq 0},$$

e, come si vede facilmente, L è bounded.

Il secondo motivo che rende interessante la tecnica dimostrativa del Teorema 4 è che essa permette di ottenere il seguente risultato particolarmente significativo.

Teorema 5. *Sia L un linguaggio context-free bounded. È possibile costruire un linguaggio regolare bounded L' la cui funzione di conteggio coincide con quella di L.*

Recentemente, in [35] è stata data una significativa applicazione del Teorema 5 nello studio della proprietà di context-freeness per alcune famiglie notevoli di parole. Se consideriamo il linguaggio $L = \{a^n b^n \mid n \geq 0\}$, possiamo osservare che il linguaggio $L' = \{ab\}^* = \{(ab)^n \mid n \geq 0\}$ è regolare ed ha, come si vede immediatamente dalla sua definizione, la stessa funzione di conteggio di L. Questo esempio ci consente di fare un'osservazione. In [13], non è stato possibile dimostrare che l'alfabeto del linguaggio L' del Teorema 5 è lo stesso del linguaggio L. Tuttavia, gli esempi considerati dagli autori sembrano indicare che, non soltanto questo è il caso ma, addirittura, il linguaggio L' è *commutativamente equivalente* a L. Ricordiamo che due linguaggi L, L' si dicono commutativamente equivalenti se possiedono lo stesso alfabeto, ovvero $L, L' \subseteq A^*$, ed esiste una biezione

$$\alpha : L \longrightarrow L'$$

di L in L' tale che, per ogni $u \in L$, si abbia:

$$\forall\, a \in A, \ \ |u|_a = |\alpha(u)|_a.$$

In questo contesto è allora naturale formulare il problema seguente:

Problema aperto: Dato un linguaggio context-free bounded, è sempre possibile ottenere un linguaggio regolare commutativamente equivalente ad esso?

Concludiamo infine questa sezione con un'ultima considerazione (*cfr.* [2]) di interesse nello studio della proprietà di ambiguità per i linguaggi context-free. Un risultato significativo dimostrato da Flajolet in [16] permette di dimostrare che se la funzione di crescita di un linguaggio context-free è trascendente, ovvero non è soluzione di alcuna equazione algebrica, allora il linguaggio è ambiguo. Questo criterio è dunque di aiuto nello studio dell'ambiguità di un linguaggio. Tuttavia, se il linguaggio è bounded context-free, per il Teorema 5, la sua funzione di conteggio e quella di crescita sono sempre razionali e quindi algebriche. Pertanto il criterio di Flajolet non è di utilità nello studio del problema predetto per i linguaggi bounded.

5 – Linguaggi bounded e problema di Burnside

Come si è detto nell'introduzione di questa nota, i linguaggi bounded giocano un ruolo significativo nello studio di un problema ben noto dell'Algebra: il *problema di Burnside per i semigruppi.* L'argomento è importante e vogliamo, per questa ragione, compiere un rapido *excursus* in questo ambito. Rimandiamo al testo [15] quale eccellente e completo riferimento bibliografico per questo argomento. Un semigruppo S si dice *periodico* o *di torsione* se, per ogni elemento s di S, esistono interi non negativi i, j tali che $i < j$ e $s^i = s^j$. Il semigruppo S si dirà poi *finitamente generato* se esiste un sottoinsieme finito di elementi di

S, detti *generatori*, tale che ogni elemento di S possa scriversi come prodotto di un numero finito di generatori. Come facilmente si verifica, un semigruppo di cardinalità finita è periodico e finitamente generato. È allora naturale porsi la domanda seguente: *È un semigruppo finitamente generato e periodico finito*?

Questo problema fu studiato per la prima volta da W. Burnside nel 1902 nel caso dei gruppi e tale studio fu successivamente esteso al caso dei semigruppi. La risposta al problema di Burnside è in generale negativa. Infatti, nel caso dei semigruppi, nel 1944, Hedlund e Morse fornirono in [21] un esempio di semigruppo infinito, costruito a partire dalla *parola di Thue*, che è 3-generato e periodico. Nel caso dei gruppi, invece, Golod, utilizzando un risultato di Shafarevitch relativo alla non finitezza della dimensione di un' opportuna algebra definita su un campo, nel 1964 in [18], dimostrò l'esistenza di un p-gruppo infinito 3-generato.

Se un semigruppo finitamente generato S è tale che, per ogni s di S, $s^m = s^n$, dove m ed n sono interi fissati tali che $0 \leq m < n$, allora il problema della finitezza di S è detto *problema di Burnside limitato.* Anche in questo caso, esistono semigruppi e gruppi che sono infiniti ed un risultato di grande interesse riguarda il gruppo libero $G(k, n)$ con k generatori nella varietà dei gruppi soddisfacenti alla identità $x^n = 1$. Un contributo di grande rilievo è infatti la dimostrazione data da Adjan e Novikov nel 1968 della infinitezza del gruppo $G(k, n)$ qualora $k > 1$ e n sia dispari con $n \geq 665$ (*cfr.* [1]). Gli autori dimostrarono inoltre che in tale caso il problema della parola è ricorsivamente decidibile, ovvero che esiste un algoritmo l'esecuzione del quale consente, date due parole sull'alfabeto dei generatori, di decidere se le due parole rappresentano il medesimo elemento del gruppo.

Un problema strettamente correlato al problema di Burnside per i semigruppi può essere formulato, nella teoria dei linguaggi formali, al modo seguente. Sia L un linguaggio su di un dato alfabeto A. Come si è detto in precedenza, in virtù del teorema di Myhill e Nerode, L è regolare se e solo se il suo monoide sintattico $M(L)$ è finito. Ricordiamo che il monoide $M(L)$ è il monoide quoziente $A^*/\equiv_L$ ove $\equiv_L$ è la relazione di congruenza definita come segue: per ogni $u, v \in A^*$, $u \equiv_L v$ se e solo se

$$\forall\, f,\ g \in A^*,\ (fug \in L \iff fvg \in L).$$

La relazione $\equiv_L$ è detta congruenza sintattica associata ad L. Supponiamo ora di aver scelto L per modo che il suo monoide sintattico sia periodico; d'ora in avanti, chiameremo semplicemente *periodici* tali linguaggi. Poiché $M(L)$ è (ovviamente) finitamente generato, nell'ipotesi in cui L sia periodico, ogni condizione che assicura una risposta positiva al problema di Burnside per i semigruppi, implica la finitezza di $M(L)$ e, dunque, la regolarità di L. Lo studio delle condizioni di finitezza per i monoidi sintattici dei linguaggi periodici è chiamato *problema di Burnside per i linguaggi* (*si veda anche* [39]). Tale problema consiste quindi nello studio delle condizioni che assicurano che un linguaggio periodico sia regolare.

Queste condizioni sono anche dette di *regolarità*. Vale la pena di osservare che ogni condizione di regolarità è una condizione di finitezza per il monoide sintattico di un linguaggio periodico e tuttavia esistono monoidi finitamente generati e periodici che non sono monoidi sintattici di nessun linguaggio. Vediamo ora come i linguaggi bounded si inseriscono nel quadro concettuale dei problemi che abbiamo ora succintamente descritto. A tale proposito, converrà ricordare le definizioni di semigruppo permutabile e debolmente permutabile.

DEFINIZIONE 5. Siano M un monoide ed n un intero positivo tale che $n \geq 2$. M è detto n-permutabile se, per ogni sequenza $m_1, \dots, m_n$ di n elementi di M, esiste una permutazione non banale ρ dell'insieme $\{1, 2, \dots, n\}$ tale che:

$$m_1 m_2 \cdots m_n = m_{\rho(1)} m_{\rho(2)} \cdots m_{\rho(n)}.$$

Si dice che M è permutabile se esiste un intero $n \geq 2$ tale che M è n-permutabile.

La definizione precedente può essere generalizzata nel modo seguente.

DEFINIZIONE 6. Siano M un monoide ed n un intero positivo tale che $n \geq 2$. M è detto n-debolmente permutabile se, per ogni sequenza $m_1, \dots, m_n$ di n elementi di M, esistono due permutazioni ϱ e τ dell'insieme $\{1, 2, \dots, n\}$, $\varrho \neq \tau$, tali che:

$$m_{\varrho(1)} m_{\varrho(2)} \cdots m_{\varrho(n)} = m_{\tau(1)} m_{\tau(2)} \cdots m_{\tau(n)}.$$

Si dice che M è debolmente permutabile se esiste un intero $n \geq 2$ tale che M è n-debolmente permutabile.

La proprietà di permutazione costituisce una generalizzazione di quella commutativa. Nel 1984, A. Restivo e C. Reutenauer nel lavoro [38] introdussero questa proprietà nell'ambito del problema di Burnside dimostrando che essa costituisce una condizione di finitezza per i semigruppi finitamente generati e periodici. Vale infatti il seguente teorema.

TEOREMA 6. *Sia M un monoide finitamente generato. Allora M è finito se e solo se M è periodico e permutabile.*

Alla luce di questo teorema è oppurtuno osservare che, in generale, la proprietà di debole permutazione non è una condizione di finitezza per i semigruppi esistendo, come dimostrato da A. Restivo in [37], monoidi finitamente generati, periodici, debolmente permutabili, che non sono di cardinalità finita. Vogliamo, a questo punto, fornire una dimostrazione molto sintetica del Teorema 6 anche e soprattutto per mettere in luce il ruolo svolto dai linguaggi bounded nello studio del problema di Burnside. A tale proposito è opportuno ricordare alcune definizioni. Ogni monoide finitamente generato M è immagine di un opportuno epimorfismo

$$\psi : A^* \longrightarrow M,$$

dove la cardinalità di A è uguale a quella dell'insieme dei generatori di M. Questo morfismo è detto *epimorfismo canonico* di M. Supponiamo ora che A sia totalmente ordinato. Possiamo ordinare totalmente A^* definendo una nuova relazione di ordine $<_a$, detta di *ordine alfabetico*, come:

$$u <_a v \Longleftrightarrow (|u| < |v|) \text{ o } (|u| = |v| \text{ e } u < v),$$

dove $<$ è l'ordinamento lessicografico. Dalla sua definizione segue che $<_a$ è un buon ordinamento. Una parola v si dice *riducibile* se esiste $u \in A^*$ tale che

$$u <_a v \text{ e } \psi(u) = \psi(v).$$

Una parola che non è riducibile si dice *irriducibile*. Sia ora m un elemento di M. Nell'insieme $\psi^{-1}(m)$ esisterà una sola parola irriducibile che chiameremo *rappresentante canonico di* m. Se L è un sottoinsieme qualsiasi del sostegno di M, indicheremo con il simbolo C_L l'insieme dei rappresentanti canonici degli elementi di L. Ovviamente gli insiemi L e C_L sono equipotenti. Queste definizioni ci consentono di estendere la nozione di funzione di crescita ad un insieme di un qualsiasi monoide finitamente generato. Come prima, sia M un monoide e sia $\psi : A^* \longrightarrow M$ il morfismo canonico di A^* in M dove la cardinalità di A è uguale a quella dell'insieme dei generatori di M. Se $m \in M$ è un generico elemento di M, la *lunghezza di* m è definita come

$$|m| = \min\{n \geq 0 \mid \psi^{-1}(m) \cap A^n \neq \emptyset\}.$$

Ad ogni sottoinsieme $L \subseteq M$, associamo allora la sua *funzione di conteggio* $f_L : \mathbb{N} \to \mathbb{N}$ definita come

$$f_L(n) = \mathrm{Card}(\{m \in L \mid |m| = n\}).$$

La *funzione di crescita di* L $g_L : \mathbb{N} \to \mathbb{N}$ si definisce a partire da quella di conteggio nello stesso modo visto nel monoide libero. Siamo ora in grado di enunciare la seguente importante proposizione (*cfr.* [15], Capitolo 3).

Proposizione 1. *Sia M un monoide permutabile e finitamente generato. Allora l'insieme dei suoi rappresentanti canonici è un linguaggio bounded.*

Possiamo alla luce della Proposizione 1 dimostrare il teorema di Restivo e Reutenauer.

Dim. Per la Proposizione 1, esistono parole $u_1, \ldots, u_k$ tali che

$$C_M \subseteq u_1^* \cdots u_k^*.$$

Ogni elemento c di C_M è rappresentato nella forma

$$u_1^{r_1} \cdots u_k^{r_k}.$$

Dunque si ottiene

$$|c| = \sum_{i=1,\dots,k} r_i |u_i| \leq r \cdot \left(\sum_{i=1,\dots,k} |u_i| \right),$$

dove $r = \max\{r_i \mid i = 1, \dots, k\}$. Sia $m_i = \phi(u_i)$, con $i = 1, \dots, k$. Poiché M è periodico, esiste un intero p tale che, per ogni $i = 1, \dots, k$, $m_i^p = m_i^{q_i}$, con $q_i < p$. Se M è infinito lo è anche C_M. Quindi la lunghezza di c e l'intero r saranno arbitrariamente grandi. Qualora $r > p$, si ha che c è riducibile e ciò contraddice l'aver assunto c quale rappresentante canonico di un elemento di M.

Si è detto prima che un linguaggio è periodico se il suo monoide sintattico è un monoide periodico. Similmente, diremo che un linguaggio è *permutabile* (risp. *debolmente permutabile*) se il suo monoide sintattico è un monoide debolmente permutabile. Il teorema che segue, la cui dimostrazione è essenzialmente basata sul Teorema 6, fornisce una notevole condizione di regolarità per i linguaggi, proposta da A. De Luca e S. Varricchio in [15]:

TEOREMA 7. *Un linguaggio L è regolare se e solo se L è permutabile e periodico.*

I risultati che seguono, proposti in [14] da F. D'Alessandro e S. Varricchio, sono di particolare interesse in quanto forniscono una condizione di regolarità per i linguaggi context-free bounded.

LEMMA 4. *Il monoide sintattico di un linguaggio bounded è permutabile.*

DIM. Siano A^* il monoide libero delle parole su di un dato alfabeto finito A, L un sottoinsieme di A^*, $\equiv_L$ la congruenza sintattica di L ed M il monoide sintattico di L. Poiché L è un linguaggio bounded, esistono parole $u_1, \dots, u_n$ di A^+ tali che $L \subseteq \{u_1\}^* \cdots \{u_n\}^*$. È lecito supporre che le parole $u_1, \dots, u_n$ siano primitive. Sia γ la massima lunghezza di una parola dell'insieme $\{u_1, \dots, u_n\}$. Proviamo che M è k-permutabile per ogni intero k tale che:

$$k - 1 > n(1 + (\gamma + 1)^2) + (n - 1).$$

Dimostriamo che, per ogni sequenza $m_1, \dots, m_k$ di k elementi di M, esiste una permutazione non banale ρ dell'insieme $\{1, \dots, k\}$ tale che $m_1 m_2 \cdots m_k = m_{\rho(1)} m_{\rho(2)} \cdots m_{\rho(k)}$.

Supponiamo che ogni $m_i \neq 1_M$. Per ogni $i = 1, \ldots, k$, sia w_i la parola di A^* che rappresenta m_i, cioè, $m_i = [w_i]_{\equiv_L}$. Ovviamente, per ogni $i = 1, \ldots, k$, w_i è una parola non vuota. Sono possibili i seguenti due casi.

Caso 1. Supponiamo che la parola $w_1 \cdots w_{k-1}$ non sia un fattore di alcuna parola di L, cioè, per ogni $\lambda, \mu \in A^*$, $\lambda w_1 \cdots w_{k-1} \mu \notin L$. Da questo segue che, per ogni $\lambda, \mu \in A^*$, $\lambda w_k w_1 \cdots w_{k-1} \mu \notin L$ e $\lambda w_1 \cdots w_{k-1} w_k \mu \notin L$. Da cui si ha: $w_1 \cdots w_{k-1} w_k \equiv_L w_k w_1 \cdots w_{k-1}$, cioè, $m_1 m_2 \cdots m_{k-1} m_k = m_k m_1 \cdots m_{k-1}$. Il primo caso è pertanto dimostrato.

Caso 2. Supponiamo ora che la condizione del Caso 1 non sia verificata. Dal momento che $k-1 > n(1+(\gamma+1)^2)+(n-1)$, possiamo trovare nella parola $w_1 \cdots w_{k-1}$ una parola della forma:

$$w_i \cdots w_j,\ 1 \leq i < j \leq k-1,$$

tale che

$$j - i \geq 1 + (\gamma+1)^2,\ \text{e}\ \alpha w_i \cdots w_j \beta \in \{u_l\}^+, \tag{4}$$

dove $1 \leq l \leq n$ e $\alpha, \beta \in A^*$. Consideriamo la sequenza di parole:

$$w_i, w_i w_{i+1}, \cdots, w_i \cdots w_{i+s}, \cdots, w_i \cdots w_j.$$

Sia v_l la coniugata di u_l. Dalla condizione (4), ogni parola della sequenza è un prefisso di una parola di $\{v_l\}^+$. Dal momento che $j - 1 \geq 1 + (\gamma+1)^2$, e il numero dei prefissi distinti di v_l sono minori o uguali a γ, esistono almeno tre interi i_1, i_2, i_3, dove $1 \leq i_1 < i_2 < i_3 \leq j$, tali che:

$$w_i \cdots w_{i_1}, w_i \cdots w_{i_2}, \cdots, w_i \cdots w_{i_3} \in \{v_l\}^+ p,$$

dove p è un prefisso di v_l. Si ottiene:

$$w_{i_1+1} \cdots w_{i_2}, w_{i_2+1} \cdots w_{i_3} \in \{sp\}^+, v_l = ps,$$

e quindi

$$w_{i_1+1} \cdots w_{i_2} w_{i_2+1} \cdots w_{i_3} = w_{i_2+1} \cdots w_{i_3} w_{i_1+1} \cdots w_{i_2}.$$

Vale allora la seguente identità:

$$\begin{aligned} m_1 \cdots m_k &= m_1 \cdots m_{i_1} (m_{i_1+1} \cdots m_{i_2} m_{i_2+1} \cdots m_{i_3}) m_{i_3+1} \cdots m_k = \\ &= m_1 \cdots m_{i_1} (m_{i_2+1} \cdots m_{i_3} w_{i_1+1} \cdots m_{i_2}) m_{i_3+1} \cdots m_k. \end{aligned}$$

Il Caso 2 è così dimostrato.

COROLLARIO 4. *Sia L un linguaggio context-free a crescita polinomiale. L è periodico se e solo se L è regolare.*

DIM. Sia M il monoide sintattico di L. Dal Teorema 1, L è un linguaggio bounded. Dal lemma precedente, il monoide M è permutabile, cosicché L è permutabile. Poiché, per ipotesi, L è periodico, dal Teorema 7 segue allora che L è regolare. Viceversa, sia L un linguaggio regolare. Per il teorema di Myhill e Nerode, il monoide M è finito e, di conseguenza permutabile e periodico. Il linguaggio L è pertanto permutabile e periodico.

Il Corollario 4 è effettivo poiché (si veda ancora [14]) esiste un algoritmo l'esecuzione del quale consente, a partire da una grammatica context-free che generi un linguaggio bounded, di decidere se il linguaggio è periodico o meno. Mostriamo infine che il corollario è non banale poiché esistono linguaggi context-free che sono periodici ma non regolari. A tale scopo introduciamo la definizione seguente.

DEFINIZIONE 7. Siano w una parola di A^* e p un intero positivo tale che $p > 1$. w è detta *p-power-free* se, per ogni $u \in A^+$, u^p non è un fattore di w.

Vale il seguente seguente risultato (*cfr.* [5], [34]).

LEMMA 5. *Sia f una parola infinita su di un alfabeto A generata da un morfismo $\phi : A^* \to A^*$ e sia* $\mathrm{Pref}(f)$ *l'insieme dei suoi prefissi finiti. Allora l'insieme complementare di* $\mathrm{Pref}(f)$ *in A^* è un linguaggio context-free.*

Utilizzando il Lemma 5, possiamo costruire un linguaggio context-free periodico non regolare. A tale proposito, siano $A = \{a, b, c\}$ un alfabeto di tre lettere, A^* il monoide libero delle parole sull'alfabeto A e sia f la parola infinita ottenuta iterando il morfismo $\phi : A^* \longrightarrow A^*$ definito come segue:

– $\phi(a) = abc$;
– $\phi(b) = ac$;
– $\phi(c) = b$.

Ad esempio, la parola abcacbabcbacabcacb è il prefisso di lunghezza 18 di f. Siano $F = \mathrm{Pref}(f)$ ed $L = A^* \backslash F$. Per il Lemma 5, L è un linguaggio context-free. È possibile dimostrare che f è una parola 3-power-free, cioè, per ogni $u \in A^*$ e per ogni intero positivo n, con $n \geq 3$, u^n non è un fattore della parola f. Quindi $u^n \in L$. Da questo segue che, per ogni intero positivo n, con $n \geq 3$, e per ogni $u \in A^*$, $[u^n]_{\equiv_L} = [u^{n+1}]_{\equiv_L}$. Dunque il monoide sintattico $A^*/ \equiv_L$ di L è periodico, ovvero L è periodico. Dimostriamo infine che L non è regolare. Per assurdo, supponiamo che L lo sia. Poiché la famiglia dei linguaggi regolari è chiusa rispetto alle usuali operazioni di intersezione e di unione insiemistica, F è un linguaggio regolare. Applicando il Pumping Lemma per i linguaggi regolari (*cfr.* Sezione 2.3) ad F, segue che esiste una costante n e una parola w di F, con $|w| \geq n$, tale che $w = \lambda u \mu$ ed inoltre, per ogni $i \geq 0$, $\lambda u^i \mu \in F$. Quindi, per ogni $i \geq 0$, $\lambda u^i \in F$. Di conseguenza f ammetterebbe parole della forma u^3 come suoi fattori, cioè f non sarebbe 3-power-free, e ciò è una contraddizione. Quindi L non è regolare.

6 – Linguaggi a crescita intermedia

In questa sezione ci interesseremo ad un risultato proposto da Grigorchuk e Machì in [19] relativo ai cosiddetti linguaggi di crescita intermedia. Diamone subito la definizione. Un linguaggio L si dice di *crescita intermedia* se la sua funzione di crescita è subesponenziale, ovvero se è definitivamente limitata superiormente da una funzione esponenziale k^n, con $k > 1$ ma non da alcuna funzione polinomiale di grado fissato. Abbiamo visto che un linguaggio context-free ha crescita polinomiale o esponenziale. Non esistono dunque linguaggi context-free a crescita intermedia. In [19], si fornisce un esempio di linguaggio a crescita intermedia la cui struttura combinatoria è, nel senso che preciseremo dopo, vicina a quella di un linguaggio context-free. Questa costruzione presuppone la conoscenza di un modello di calcolo, detto *automa stack*, il quale risulta, rispetto all'operazione di riconoscimento di linguaggi di parole, più potente di quello degli automi push-down, cioè delle macchine in grado di riconoscere i linguaggi context-free. Gli automi stack sono stati introdotti da Ginsburg, Greibach, ed Harrison (*cfr.* [24], Capitolo 14) alla fine degli Anni 60 nello studio di una possibile estensione dei linguaggi context-free. Non si ha intenzione di presentare in modo rigoroso la loro definizione ma, semplicemente, di fornire alcuni elementi descrittivi essenziali di questa struttura, al fine di illustrare l'esempio predetto. Essenzialmente, un automa stack è un automa push-down che, oltre a poter manipolare la pila con le stesse modalità di un automa push-down, può, in un qualsiasi passo della computazione, accedere, leggere e modificare il simbolo memorizzato in una locazione qualsiasi della pila, senza essere costretto (come accade negli automi push-down) a rimuovere tutti i caratteri memorizzati nelle locazioni che lo precedono. Quest' ultima modalità di accesso alla pila sarà detta *in modalità stack*. Gli automi stack possono essere deterministici, non deterministici, ad una via ("one way"), ovvero con la possibilità di leggere la parola di ingresso una sola volta (da sinistra verso destra) oppure a due vie ("two way"), ovvero con la possibilità di leggere, attraverso una oppurtuna testina di lettura, più volte la parola di input su di un apposito nastro, percorribile in entrambe le direzioni. In questo capitolo, siamo interessati ad automi 1-DNESA ("One-way, Deterministic, Non Erasing, Stack Automata"), cioè ad automi stack, deterministici, ad una via, in grado di leggere, ma non cancellare, il simbolo memorizzato in una locazione della pila.

Ricordiamo che se n un intero positivo, la sequenza di t interi positivi, $(n_1, n_2, \ldots, n_t)$ tali che $n_1 \geq n_2 \geq \ldots \geq n_t$, costituisce una partizione dell'intero

n se:

$$n = n_1 + n_2 + \cdots + n_t.$$

Indicando con $P(n)$ il numero di partizioni dell'intero n, si ha asintoticamente:

$$P(n) \approx \frac{e^{\pi\sqrt{\frac{2n}{3}}}}{4n\sqrt{3}}.$$

Teorema 8. *Sia $L \subseteq \{a, b\}^*$ il linguaggio sull'alfabeto $\{a,b\}$ costituito dall'insieme di tutte e sole le parole*:

$$ab^{i_1}ab^{i_2}\cdots ab^{i_k},$$

dove $0 \leq i_1 \leq i_2 \leq \ldots \leq i_k$ è una sequenza monotona non decrescente di $k \geq 1$ interi non negativi. Allora L è un linguaggio a crescita intermedia ed è accettato da un automa 1-DNESA.

Dim. Verifichiamo, prima di tutto, che L è un linguaggio a crescita intermedia. Se $n \in \mathbb{N}$ e $(l_1, l_2, \cdots, l_k)$ è una partizione di n, possiamo ad essa associare in modo univoco la parola di L $ab^{l_1}ab^{l_2}\ldots ab^{l_k}$. Di conseguenza, per ogni n, il numero delle parole di L di lunghezza n è uguale al numero delle partizioni distinte di n. Quindi, la funzione di struttura di L è subesponenziale ed in base alla definizione di g_L, un semplice argomento di conteggio permette di mostrare che anche g_L è subesponenziale. Quindi L ha crescita intermedia. Per il teorema di gap, L non è un linguaggio context-free. Mostriamo ora che L è accettato da un automa 1-DNESA. La struttura ed il funzionamento di questo automa possono essere descritti nel modo seguente. Se una parola w è una stringa (eventualmente vuota) di sole occorrenze del simbolo a, allora la parola w è accettata. Se la parola w comincia con una stringa di occorrenze del simbolo a seguita da una stringa s_1 di occorrenze del simbolo b allora tali occorrenze vengono inserite nella pila. A questo punto della computazione, i casi possibili sono i seguenti:

- Se, dopo aver letto la stringa s_1 di occorrenze del simbolo b, la lettura sul nastro d'ingresso termina, la parola è accettata;
- Supponiamo, invece, che la parola w sia della forma:

$$w = aa\cdots a\underbrace{bb\cdots b}_{s_1}a\underbrace{bb\cdots b}_{s_2}w',\ w' \in \{a,b\}^*$$

  Perché la parola w venga accettata deve necessariamente essere, per definizione del linguaggio L, $|s_1| \leq |s_2|$. Usando la testina della pila, l'automa, leggendo la pila in modalità stack, è in grado di confrontare il numero di b presenti nella pila, cioè $|s_1|$, con il numero di b nella stringa s_2, cioè $|s_2|$.

Più precisamente, leggendo in modo sincronizzato i simboli di s_1 ed s_2, uno per volta, se l'operazione di lettura del fattore s_2 (sul nastro d'ingresso) termina prima di quella di s_1 (nella pila), allora ciò significa che $|s_1| > |s_2|$. La computazione termina e la parola è rifiutata. Se ciò non accade, allora $|s_1| \leq |s_2|$. A questo punto, le eventuali occorrenze rimanenti del simbolo b vengono inserite nella pila e la computazione, nel caso in cui la parola w' sia non vuota, prosegue secondo la modalità prima descritta.

La descrizione sintetica ora data è quella di un modello di calcolo deterministico "one-way". Infine, poiché durante la computazione, l'operazione di cancellazione di simboli dalla pila non è mai effettuata, l'automa è "non erasing". Abbiamo quindi fornito la descrizione di un automa 1-DNESA che è in grado di accettare il linguaggio considerato.

7 – Sviluppi recenti

Chiudiamo questo articolo descrivendo molto sinteticamente il contenuto di un lavoro recente [10], relativo alla estensione dei teoremi presentati nelle Sezioni 3 e 4, al caso dei sottoinsiemi razionali dei monoidi di relazioni di parole. Rimandiamo il lettore alla consultazione dei testi [3] e [40], referenze ormai classiche sull'argomento, per una piana ed efficace introduzione ai concetti fondamentali della tematica delle relazioni razionali. Qui, ci limiteremo a richiamare un vocabolario minimo di concetti al solo fine di presentare i risultati a cui abbiamo accennato. Sia $M = A_1^* \times \cdots \times A_k^*$ il prodotto diretto di monoidi liberi generati da alfabeti $A_1, \ldots, A_k$. I sottoinsiemi di M sono chiamati k-relazioni.

Una k-relazione si dice *razionale* se si ottiene, a partire da relazioni finite, tramite l'applicazione, in un numero finito di volte, delle operazioni razionali di M, ovvero delle operazioni di unione insiemistica e di prodotto di due relazioni e della operazione di stella che associa ad ogni relazione il sottomonoide di M da essa generato. In virtù di un ben noto teorema di caratterizzazione, una relazione è k-razionale se i suoi elementi, cioè k-uple di parole su alfabeti fissati, sono accettati da uno specifico modello di calcolo detto *automa a k nastri.* Un automa a k nastri è in essenza un automa a stati finiti, non deterministico, dotato di k nastri, ognuno dei quali in grado di memorizzare una parola data. Al generico istante di computazione, l'automa è in grado di leggere, e di elaborare, uno per volta, i caratteri di una qualsiasi delle k parole registrate sui nastri. La k-upla è poi accettata se, una volta completata la lettura di tutte le k parole, l'automa si trovi in uno stato scelto nell'ambito di un insieme particolare di stati detti *accettanti.* Un automa siffatto costituisce una estensione del modello classico di automa a stati finiti. Come si è visto nella Sezione 5, i concetti di funzione di conteggio, di funzione di crescita (e la relativa classificazione dei linguaggi vista nella Definizione 1) e di insieme bounded possono essere definiti in un monoide finitamente generato qualsiasi in modo simile a quanto visto nei monoidi di parole. Vale allora il teorema seguente.

TEOREMA 9 ([10]). *Sia* $M = A_1^* \times \cdots A_k^*$ *il prodotto diretto di monoidi liberi generati da alfabeti* $A_1, \ldots, A_k$. *Valgono le condizioni seguenti*:

- *Una* k*-relazione razionale di* M *ha crescita esponenziale oppure polinomiale.*
- *Una* k*-relazione razionale di* M *ha crescita polinomiale se e solo se è bounded in* M.
- *È possibile decidere se una* k*-relazione razionale ha crescita esponenziale oppure polinomiale.*

È infine interessante ricordare che in [10] il teorema precedente è stato dimostrato nel caso più generale dei monoidi parzialmente commutativi di cui il prodotto di monoidi liberi costituisce una istanza particolare.

Ringraziamenti

Un "grand merci" da parte degli autori a Arturo Carpi e Tullio Ceccherini-Silberstein per le interessanti discussioni ed i suggerimenti numerosi che hanno preceduto e accompagnato la redazione di questa nota.

REFERENCES

[1] S. I. ADJAN: *The Burnside problem and identities in groups*, Springer, Berlin, 1985.

[2] A. BERTONI: *comunicazione privata agli autori*, 2006.

[3] J. BERSTEL: *Transduction and Context-Free Languages*, Teubner, Stuttgart, 1979.

[4] J. BERSTEL – D. PERRIN: *Theory of codes*, Academic Press, New York, 1985.

[5] J. BERSTEL: *Every iterated morphism yields a co-CFL, Information Processing Letters*, **22** (1986), 7-9.

[6] M. R. BRIDSON – R. H. GILMAN: *Context-free languages of sub-exponential growth*, J. Comput. System Sci., **64** (2) (1999), 308-310.

[7] T. CECCHERINI-SILBERSTEIN – W. WOESS: *Growth and ergodicity of context-free languages*, Trans. Amer. Math. Soc., **354** (11) (2002), 4597-4625.

[8] T. CECCHERINI-SILBERSTEIN: *On the growth of linear languages*, Advances in Appl. Math., **35** (2005), 243-253.

[9] C. CHOFFRUT – F. D'ALESSANDRO – S. VARRICCHIO: *On the separability of sparse context-free languages and of bounded rational relations*, Theoret. Comput. Sci., **381** (1-3) (2007), 274-279.

[10] C. CHOFFRUT – F. D'ALESSANDRO – S. VARRICCHIO: *On bounded rational trace languages*, in corso di pubblicazione su Theory of Computing Systems.

[11] N. Chomsky – M. P. Schützenberger: *The algebraic theory of context-free languages*, in: "Computer Programming and Formal System", P. Braffort, D. Hirschberg (eds.), North-Holland, Amsterdam,1963, 118-161.

[12] F. D'Alessandro – A. de Luca: *Teoria degli automi*, in preparazione.

[13] F. D'Alessandro – B. Intrigila – S. Varricchio: *On the structure of the counting function of context-free languages*, Theoret. Comput. Sci., **356** (2006), 104-117.

[14] F. D'Alessandro – S. Varricchio: *On the growth of context-free languages*, to appear on Journal of Automata, Languages and Combinatorics, **13** (2) (2008).

[15] A. de Luca – S. Varricchio: *Finiteness and Regularity in Semigroups and Formal Languages*, Springer, Berlin, 1999.

[16] P. Flajolet: *Analytic models and ambiguity of context-free languages*, Theoret. Comput. Sci., **49** (1987), 283-309.

[17] S. Ginsburg: *The mathematical theory of context-free languages*, McGraw-Hill Book Co., 1966.

[18] E. S. Golod: *On nil-algebras and finitely approximable p-groups*, Izv. Akad. Nauk. SSSR, Ser. Mat., **28** (1964), 273-276.

[19] R. I. Grigorchuk – A. Machì: *An example of an indexed language of intermediate growth*, Theoret. Comput. Sci., **215** (1999), 325-327.

[20] M. Gromov: *Groups of polynomial growth and expanding maps*, Inst. Hautes Études Sci., Publ. Math., **53** (1981), 53-73.

[21] G. Hedlund – M. Morse: *Unending chess, symbolic dynamics and a problem in semigroups*, Duke Math. J., **11** (1944), 1-7.

[22] J. Honkala: *Decision problems concerning thinness and slenderness of formal languages*, Acta Informatica, **35** (1998), 625-636.

[23] J. Honkala: *On Parikh slender context-free languages*, Theoret. Comput. Sci., **255** (2001), 667-677.

[24] J. Hopcroft – J. Ullman: *Introduction to Automata Theory, Languages and Computation*, Addison-Wesley Pub. Co., 1979.

[25] O. Ibarra – B. Ravikumar: *On sparseness, ambiguity and other decision problems for acceptors and transducers*, Lecture Notes in Computer Science, **210** Springer-Verlag, Berlin, 1986, 171-179.

[26] L. Ilie: *On a conjecture about slender context-free languages*, Theoret. Comput. Sci., **132**, (1994), 427-434.

[27] L. Ilie – G. Rozenberg – A. Salomaa: *A characterization of poly-slender context-free languages*, Theoretical Informatics and Applications, **34** (2000), 77-86.

[28] L. Ilie: *On length of words in context-free languages*, Theoret. Comput. Sci., **242** (2000), 327-359.

[29] L. Ilie: *On generalized slenderness of context-free languages*, in: "Words, semigroups, and transductions", World Sci. Publishing, River Edge, NJ, 2001, 189-202.

[30] R. Incitti: *The growth function of context-free languages*, Theoret. Comput. Sci., **255** (2001), 601-605.

[31] G. Krause – T. H. Lenagan: *Growth of Algebras and Gelfand-Kirillov Dimension*, Research Notes in Math. **116** Pitman, London, 1985.

[32] M. Latteux – G. Thierrin: *On bounded context-free languages*, Elektron. Informationsverarb. Kybernet., **20** (1984), 3-8.

[33] Lothaire: *Combinatorics on words*, World Science Division, Addison-Wesley, Reading, MA, 1983.

[34] M. G. Main – W. Bucher – D. Haussler: *Applications of an infinite co-CFL*, Proc. 12th ICALP 1985, Lecture Notes in Computer Science, Vol. 194, 1985, 404-412.

[35] N. Rampersad: *On the context-freeness of the set of words containing overlaps*, Information Processing Letters, **102** (2007), 74-78.

[36] D. Raz: *Length considerations in context-free languages*, Theoret. Comput. Sci., **183** (1997), 21-32.

[37] A. Restivo: *Permutation properties and the Fibonacci Semigroup*, Semigroup Forum, **38** (1989), 337-345.

[38] A. Restivo – C. Reutenauer: *On the Burnside problem for semigroups*, J. of Algebra, **89** (1984), 102-104.

[39] A. Restivo – C. Reutenauer: *Rational languages and the Burnside problem*, Theoret. Comput. Sci., **40** (1985), 13-30.

[40] J. Sakarovitch: *Éléments de théorie des automates*, Vuibert, Paris, 2003.

Lavoro pervenuto alla redazione il 25 giugno 2008
ed accettato per la pubblicazione il 30 giugno 2008.
Bozze licenziate il 30 settembre 2008

Indirizzo dell'Autore:

Flavio D'Alessandro – Alessandra Zinno Pilo – Dipartimento di Matematica – Università di Roma "La Sapienza" – Piazzale Aldo Moro 2, 00185 Roma – Italy.
E-mail: dalessan@mat.uniroma1.it
http: www.mat.uniroma1.it/people/dalessandro