Regulated Re-writing

In a given grammar , re-writing can take place at a step of a derivation by the usage of any applicable rule in any desired place. That is if A is a nonterminal occurring in any sentential form say $\alpha A \beta$, the rules being

$$A \to \gamma$$

$$A \to \delta$$

then any of these two rules are applicable for the occurrence of A in $\alpha A \beta$. Hence, one encounters nondeterminism in its application. One way of naturally restricting the nondeterminism is by regulating devices, which can select only certain derivations as correct in such a way that the obtained language has certain useful properties. For example, a very simple and natural control on regular rules may yield a non regular language .

While defining the four types of grammars, we put restrictions in the form of production rules to go from type 0 to type 1 , then to type 2 and type 3 . In this chapter we put restrictions on the manner of applying the rules and study the effect . There are several methods to control re-writing , some of the standard control strategies are as follows

Matrix Grammar

A matrix grammar is a quadruple G = (N,T,P,S) where N , T and S are as in any Chomsky grammar. P is a finite set of sequences of the form :

$$m = [\alpha_1 \to \beta_1, \alpha_2 \to \beta_2, \ldots\ldots\ldots\ldots, \alpha_n \to \beta_n]$$

$n \geq 1$, with $\alpha_i \in (N \bigcup T)^+, \beta_i \in (N \bigcup T)^*, 1 \leq i \leq n.$ m is a member of P and a 'matrix' of P.

G is a matrix grammar of type i, where $i \in \{0,1,2,3\}$, if and only if the grammar $G_m = (N,T,m,S)$ is of type i for every $m \in P.$

Similarly , G is $\varepsilon - free$ if each G_m is $\varepsilon - free$

Definition 1

Let $G=\left(N,T,P,S\right)$ be a matrix grammar. For any two strings $u,v\in\left(N\bigcup T\right)^{+}$, we write $u\underset{G}{\Rightarrow}v$ (or $u\Rightarrow v$ if there is no confusion on G), if and only if there are strings $u_0,u_1,u_2,\ldots\ldots\ldots,u_n$ in $\left(N\bigcup T\right)^{+}$ and a matrix $m\in M$ such that $u=u_0,u_n=v$ and

$$u_{i-1}=u^{'}_{i-1}x_iu^{''}_{i-1},u_i=u^{'}_{i-1}y_iu^{''}_{i-1}$$

for some $u^{'}_{i-1},u^{''}_{i-1}$ for all $0\le i\le n-1$ and $x_i\rightarrow y_i\in m,1\le i\le n.$

Clearly , any direct derivation in a matrix grammar G corresponds to an n-step derivation by $G_m=\left(N,T,P,S\right)$. That is, the rules in m are used in sequence to reach v. $\overset{*}{\Rightarrow}$ is the reflexive, transitive closure of $\Rightarrow$ and

$$L\left(G\right)=\left\{w/w\in T^{*},S\overset{*}{\Rightarrow}w\right\}$$

Definition 2

Let $G = (N, T, P, S)$ be a matrix grammar . Let F be a subset of rules of M . We now use the rules of F such that , the rules in F can be passed over if they cannot be applied , whereas the other rules in any matrix $m \in P$ not in F must be used. That is, for

$$u, v \in (N \cup T)^{+}, u \underset{m}{\Rightarrow} v,$$

if and only if there are strings $u_0, u_1, \ldots\ldots\ldots, u_n$ and a matrix $m \in M$ with rules $\{r_1, r_2, \ldots\ldots\ldots, r_n\}$ (say) , with r_i :

$$x_i \to y_i \qquad 1 \le i \le n.$$

Then , either $u_{i-1} = u'_{i-1} x_i u''_{i-1}$, $u_i = u'_{i-1} y_i u''_{i-1}$ or the rule $x_i \to y_i \in F$. Then $u_i = u_{i-1}$

This restriction by F on any derivation is denoted as $\underset{ac}{\Rightarrow}$, where 'ac' stands for 'appearance checking' derivation mode. Then ,

$$L(G,F)=\left\{w / S \overset{*}{\underset{ac}{\Rightarrow}} w, w \in T^{*}\right\}$$

Let $M\left(M_{ac}\right)$ denote the family of matrix languages without appearance checking (with appearance checking) of type 2 without $\varepsilon - rules.$

Let $M^{\lambda}\left(M_{ac}^{\lambda}\right)$ denote the family of matrix languages without appearance checking (with appearance checking) of type 2 with $\varepsilon - rules.$

Example 1

Let $G = (N, T, P, S)$ be a matrix grammar where

$$N = \{S, A, B, C, D\}$$

$$T = \{a, b, c, d\}$$

$$P = \{P_1, P_2, P_3, P_4\},$$ where

$$P_1 : [S \to ABCD]$$

$$P_2 : [A \to aA, B \to B, C \to cC, D \to D]$$

$$P_3 : [A \to A, B \to bB, C \to C, D \to dD]$$

$$P_4 : [A \to a, B \to b, C \to c, D \to d]$$

Some sample derivations are :

$$S \underset{p_1}{\Rightarrow} ABCD \underset{p_2}{\Rightarrow} aABcCD \underset{p_4}{\Rightarrow} aabccd$$

$$S \underset{p_1}{\Rightarrow} ABCD \underset{p_2}{\Rightarrow} aABcCD \underset{p_3}{\Rightarrow} aAbBcCdD \underset{p_4}{\Rightarrow} aabbccdd$$

We can see that the application of matrix P_2 produces an equal number of a's and c's , application of P_3 produces an equal number of b's and d's . P_4 terminates the derivation . Clearly

$$L(G) = \left\{ a^n b^m c^n d^m \mid n, m \geq 1 \right\}.$$

The rules in each matrix are context free , but the language generated is context-sensitive and not context-free.

Example 2

Let $G = (N, T, P, S)$ be a matrix grammar with

$$N = \{S, A, B, C,\}$$

$$T = \{a, b\}$$

$$P = \{P_1, P_2, P_3, P_4, P_5\},$$ where

$$P_1 : [S \to ABC]$$

$$P_2 : [A \to aA,\ B \to aB,\ C \to aC]$$

$$P_3 : [A \to bA,\ B \to bB,\ C \to bC]$$

$$P_4 : [A \to a,\ B \to a,\ C \to a]$$

$$P_5 : [A \to b,\ B \to b,\ C \to b]$$

Some sample derivations are :

$$S \underset{p_1}{\Rightarrow} ABC \underset{p_2}{\Rightarrow} aAaBaC \underset{p_3}{\Rightarrow} abAabBabC \underset{p_4}{\Rightarrow} abaabaaba$$

$$S \underset{p_1}{\Rightarrow} ABC \underset{p_3}{\Rightarrow} bAbBbC \underset{p_2}{\Rightarrow} baAbaBbaC \underset{p_5}{\Rightarrow} babbabbab$$

clearly

$$L(G) = \left\{ www \mid w \in \{a,b\}^{+} \right\}.$$

Programmed Grammar

A Programmed Grammar is a 4-tuple $G=(N,T,P,S)$ where N , T and S are as in any Chomsky grammar . Let r be a collection of re-writing rules over $N\bigcup T$, lab (R) being the labels of R . σ and φ are mappings from lab(R) to $2^{lab(R)}$

$$P=\{(r,\sigma(r),\varphi(r))\mid r\in R\}$$

Here , G is said to be type i, or $\varepsilon-free$ if the rules in R are all type i , where i = 0,1,2,3 or $\varepsilon-free$, respectively.

Definition 3

For any x, y over $\left(N \bigcup T\right)^{*}$, we define derivation as below :

(i) $\left(u, r_1\right) \Rightarrow \left(v, r_2\right)$ if and only if $u = u_1 x u_2, v = u_1 y u_2$ for u_1, u_2 are over $N \bigcup T$ and $\left(r_1 : x \to y, \sigma\left(r_1\right), \varphi\left(r_1\right)\right) \in P$ and $r_2 \in \sigma\left(r_1\right)$ and

(ii) $\left(u, r_1\right) \underset{ac}{\Rightarrow} \left(v, r_2\right)$ if and only if $\left(u, r_1\right) \Rightarrow \left(v, r_2\right)$ holds , or else u=v if $r_1 : \left(x \to y, \sigma\left(r_1\right), \varphi\left(r_1\right)\right)$ is not applicable to u , i.e., x is not a sub word of u and $r_2 \in \varphi\left(r_1\right)$. Thus, $\underset{ac}{\Rightarrow}$ only depends on φ

Here, $\sigma\left(r\right)$ is called the success field as the rule with label r is used in the derivation step . $\varphi\left(r\right)$ Is called the failure field as the rule with label r cannot be applied and we move on to a rule with label in $\varphi\left(r\right)$.

$\overset{*}{\Rightarrow}$, $\underset{ac}{\overset{*}{\Longrightarrow}}$ are the reflexive and transitive closures of $\Rightarrow$ and $\underset{ac}{\Longrightarrow}$, respectively.

The language generated is defined as follows :

$$L(G,\sigma)=\left\{w \mid w\in T^{*},(S_1,r_1)\overset{*}{\Rightarrow}(w,r_2)\ for\,some\ \ r_1,r_2\in lab(P)\right\}$$

$$L(G,\sigma,\varphi)=\left\{w \mid w\in T^{*},(S_1,r_1)\underset{ac}{\overset{*}{\Longrightarrow}}(w,r_2)\ for\,some\ r_1,r_2\in lab(P)\right\}$$

Let $P\left(P_{ac}\right)$ denote the family of programmed languages without (with) appearance checking of type 2 without $\varepsilon-rules$.

Let $P^{\lambda}\left(P_{ac}^{\lambda}\right)$ denote the family of programmed languages without (with) appearance checking of type 2 with $\varepsilon-rules$.

Example 3

Let $G = (N, T, P, S)$ be a programmed grammar with

$$N = \{S, A, B, C, D\}$$

$$T = \{a, b, c, d\}$$

$$P:$$

	r	$\sigma(r)$	$\varphi(r)$
1.	$S \longrightarrow ABCD$	2,3,6	ϕ
2.	$A \longrightarrow aA$	4	ϕ
3.	$B \longrightarrow bB$	5	ϕ
4.	$C \longrightarrow cC$	2,3,6	ϕ

	r	$\sigma(r)$	$\varphi(r)$
5.	D $\longrightarrow$ dD	2,3,6	ϕ
6.	A $\longrightarrow$ a	7	ϕ
7.	B $\longrightarrow$ b	8	ϕ
8.	C $\longrightarrow$ c	9	ϕ
9.	D $\longrightarrow$ d	ϕ	ϕ

Let $lab(F) = \{1,2,3,4,5,6,7,8,9\}$.

Some sample derivations are

$$S \underset{1}{\Rightarrow} ABCD \underset{6}{\Rightarrow} aBCD \underset{7}{\Rightarrow} abCD \underset{8}{\Rightarrow} abcD \underset{9}{\Rightarrow} abcd$$

$$S \underset{1}{\Rightarrow} ABCD \underset{2}{\Rightarrow} aABCD \underset{4}{\Rightarrow} aABcCD \underset{6}{\Rightarrow} aaBcCD$$

$$\underset{7}{\Rightarrow} aabcCD \underset{8}{\Rightarrow} aabccD \underset{9}{\Rightarrow} aabccd$$

$$L(G) = \{a^n b^m c^n d^m \mid n, m \geq 1\}$$

Example 4

Let $G=(N,T,P,S)$ be a programmed grammar with

$N=\{S,A,B,C\}$

$T=\{a,b\}$

$P:$

	r	σ	φ
1.	$S \longrightarrow ABC$	2,5,8,11	ϕ
2.	$A \longrightarrow aA$	3	ϕ
3.	$B \longrightarrow aB$	4	ϕ
4.	$C \longrightarrow aC$	2,5,8,11	ϕ

	r	σ	φ
5.	$A \longrightarrow bA$	6	ϕ
6.	$B \longrightarrow bB$	7	ϕ
7.	$C \longrightarrow cB$	2,5,8,11	ϕ
8.	$A \longrightarrow a$	9	ϕ
9.	$B \longrightarrow a$	10	ϕ
10.	$C \longrightarrow a$	ϕ	ϕ
11.	$A \longrightarrow b$	12	ϕ
12.	$B \longrightarrow b$	13	ϕ
13.	$C \longrightarrow b$	ϕ	ϕ

$$L(G) = \left\{ www \mid w \in \{a,b\}^{+} \right\}.$$

Random Context grammar

A Random context grammar has two sets of nonterminals X , Y where the set X is called the permitting context and Y is called the forbidding context of a rule $x \rightarrow y$.

Definition 4

$G=(N,T,P,S)$ is a random context grammar where N ,T and S are as in any Chomsky grammar , where

$$p=\left\{\left(x\rightarrow y,X,Y\right)\mid x\rightarrow y\, is\, a\, rule\, over\; N\bigcup T,X,Y\; are\, subsets\, of\; N\right\}$$

We say $u\underset{G}{\Rightarrow}v$ if and only if $u=u^{'}xu^{''},v=u^{'}yu^{''}$ for $u^{'},u^{''}$ over $N\bigcup T$ $\left(x\rightarrow y,X,Y\right)$ such that all symbols X appear in and appears in $u^{'}\; u^{''}$ and no symbol of Y appears in $u^{'},u^{''}$. $\overset{*}{\Rightarrow}$ is the reflexive transitive closure of $\Rightarrow$.

$$L(G)=\left\{w:S\overset{*}{\Longrightarrow}w,w\in T^{*}\right\}.$$

As before, L is of type i, whenever G with rules $x\rightarrow y$ in P are of type i , i=0,1,2,3, respectively.

Example 5

Consider the random context grammar $G=\left(N,T,P,S\right)$ where

$$N=\left\{S,A,B,C\right\}$$

$$T=\left\{a\right\}$$

$$p=\left\{\begin{matrix}\left(S\to AA,\phi,\left\{B,D\right\}\right),\left(A\to B,\phi,\left\{S,D\right\}\right),\\ \left(B\to S,\phi,\left\{A,D\right\}\right),\left(A\to D,\phi,\left\{S,B\right\}\right),\\ \left(D\to a,\phi,\left\{S,A,B\right\}\right),\end{matrix}\right\}.$$

Some sample derivations are

$$S\Rightarrow AA\Rightarrow DA\Rightarrow DD\Rightarrow aD\Rightarrow aa$$

$$S\Rightarrow AA\Rightarrow BA\Rightarrow BB\Rightarrow SB\Rightarrow SS$$

$$\Rightarrow AAS\Rightarrow AAAA\Rightarrow a^4$$

$$L\left(G\right)=\left\{a^{2^n}\mid n\geq 1\right\}.$$

Time varying Grammar

Given a grammar G , one can think of applying a set of rules only for a particular period . That is, the entire set of P is not available at any step of a derivation . Only a subset of P is available at any time ‘t’ or at any i-th step of a derivation.

Definition 5

A time-varying grammar of type i , $0 \le i \le 3,$ is an ordered pair (G, ϕ) where $G = (N, T, P, S)$ is a type i grammar , and ϕ is a mapping of the set of natural numbers into the set of subsets of P . $(u, i) \Rightarrow (v, j)$

holds if and only if:

1. $j = i + 1 \ and$

2. There are words u_1, u_2, x, y over $N \bigcup T$ such that $u = u_1 x u_2$, $v = u_1 y u_2$ and $x \to y$ is a rule over $N \bigcup T$ in $\varphi(i)$.

$\overset{*}{\Longrightarrow}$ be the reflexive , transitive closure of $\Rightarrow$ and

$$L(G,\phi) = \left\{ w \mid (S,1) \overset{*}{\Rightarrow} (w, j) \right\} \text{ for some } j \in N, w \in T^*$$

A language L is time varying of type i if and only if for some time varying grammar (G,ϕ) is of type i with $L = L(G,\phi)$.

Definition 6

Let (G,ϕ) be a time varying grammar . Let F be a subset of the set of productions P . A relation $\underset{ac}{\Rightarrow}$ on the set of pairs (u , j) , where u is a word over $N \bigcup T$ and j is a natural number which is defined as follows :

$(u, j_1) \underset{ac}{\Rightarrow} (v, j_2)$ holds , if

$(u, j_1) \Rightarrow (v, j_2)$ holds , or else ,

$j_2 = j_1 + 1 , u = v,$ and for no production

$x \rightarrow y$ in $F \bigcap \phi(j_1)$,x is a subword of u.

$\overset{*}{\underset{ac}{\Longrightarrow}}$ is the reflexive , transitive closure of $\underset{ac}{\Longrightarrow}$. Then , the language generated by (G,ϕ) with appearance checking for productions in F is defined as :

$$L_{ac}(G,\phi,F)=\left\{w \mid w\in T^{*} \mid (S,1)\overset{*}{\underset{ac}{\Rightarrow}}(w,j)\ \ for\,some\ j\right\}$$

The family of languages of this form without appearance checking when the rules are context free (context-free and $\varepsilon - free$) and ϕ is a periodic function are denoted as τ^{λ} and τ , respectively. With appearance checking, they are denoted as τ_{ac}^{λ} and τ_{ac} , respectively.

Example 6

Let (G, ϕ) be a periodically time varying grammar with

$$G = (N, T, P, S) \text{ where}$$

$$N = \{S, X_1, Y_1, Z_1, X_2, Y_2, Z_2\}$$

$$T = \{a, b\}$$

$$P = \phi(1) \cup \phi(2) \cup \phi(3) \cup \phi(4) \cup \phi(5) \cup \phi(6) \text{ where}$$

$$\phi(1) = \{S \to aX_1aY_1aZ_1, S \to bX_1bY_1bZ_1, X_1 \to X_1, Z_2 \to Z_2\}$$

$$\phi(2) = \{X_1 \to aX_1, X_1 \to bX_2, X_2 \to aX_1, X_2 \to bX_2, X_1 \to \varepsilon, X_2 \to \varepsilon\}$$

$$\phi(3) = \{Y_1 \to aY_1, Y_1 \to bY_2, Y_2 \to aY_1, Y_2 \to bY_2, Y_1 \to \varepsilon, Y_2 \to \varepsilon\}$$

$$\phi(4) = \{Z_1 \to aZ_1, Y_1 \to bZ_2, Z_2 \to aZ_1, Z_2 \to bZ_2, Z_1 \to \varepsilon, Z_2 \to \varepsilon\}$$

$$\phi(5) = \{X_2 \to X_2, Y_1 \to Y_1\}$$

$$\phi(6) = \{Y_2 \to Y_2, Z_1 \to Z_1\}$$

Some sample derivations are a

$$(S,1) \Rightarrow (aX_1aY_1aZ_1,2) \Rightarrow (aaY_1aZ_1,3) \Rightarrow (aaaZ_1,4) \Rightarrow (aaa,5)$$

$$(S,1) \Rightarrow (bX_1bY_1bZ_1,2) \Rightarrow (baX_1bY_1bZ_1,3) \Rightarrow (baX_1baY_2bZ_1,4)$$

$$\Rightarrow (baX_1baY_1baZ_1,5) \Rightarrow (baX_1baY_1baZ_1,6)$$

$$\Rightarrow (baX_1baY_1baZ_1,7) \Rightarrow (baX_1baY_1baZ_1,8)$$

$$\Rightarrow (babaY_1baZ_1,9) \Rightarrow (bababaZ_1,10)$$

$$\Rightarrow (bababa,11)$$

$$L(G,\phi) = \left\{www \mid w \in \{a,b\}^+\right\}$$

Example 7

Let (G,ϕ) be a periodically time varying grammar with

$$G=(N,T,P,S)$$

$$N=\{A,B,C,D,S,A_1,A_2,B_1,B_2,C_1,C_2,D_1,D_2\}$$

$$T=\{a,b,c,d\}$$

$P:\bigcup_{i=1}^{8}\phi(i),$ where

$$\phi(1)=\{S\to aAbBcCdD,D_1\to D,A_2\to A\}$$

$$\phi(2)=\{A\to aA_1,A_1\to A_2,A\to\varepsilon\}$$

$$\phi(3)=\{B\rightarrow B_1, B\rightarrow bB_2, B\rightarrow \varepsilon\}$$

$$\phi(4)=\{C\rightarrow cC_1, C\rightarrow C_2, C\rightarrow \varepsilon\}$$

$$\phi(5)=\{D\rightarrow D_1, D\rightarrow dD_2, D\rightarrow \varepsilon\}$$

$$\phi(6)=\{A_1\rightarrow A, B_2\rightarrow B\}$$

$$\phi(7)=\{B_1\rightarrow B, C_2\rightarrow C\}$$

$$\phi(8)=\{C_1\rightarrow C, D_2\rightarrow D\}$$

$$L(G,\phi)=\{a^n b^m c^n d^m \mid n,m\geq 1\}.$$

Regular Control Grammars

Let G be a grammar with production set P and lab(P) be the labels of productions of P. To each derivation D , according to G , there corresponds a string over lab(P) (the so called control string) . Let C be a language over lab(P) . We define a language L generated by a grammar G such that every string of L has a derivation D with a control string from C. Such a language is said to be a controlled language .

Definition 7

Let $G = (N, T, P, S)$ be a grammar . Let lab(P) be the set of labels of productions in P . Let F be a subset of P . Let D be a derivation of G and K be word over lab(P) . K is a control word of D , if and only if the following conditions are satisfied :

1. For some string u, v, u_1, u_2, x, y over $N \bigcup T$, $D: u \Rightarrow v$ and K=f, where $u = u_1 x u_2$, $v = u_1 y u_2$ and $x \rightarrow y$ has a label f.

2. For some u, x, y , D is a derivation of a word 'u' only and $K = \varepsilon$ or else $K = f$, where $x \rightarrow y$ has a label $f \in F$ and x is not a sub word of u.

3. For some $u, v, w, K_1, K_2,$ D is a derivation $u \overset{*}{\Rightarrow} v \overset{*}{\Rightarrow} w$,where $K = K_1 K_2$ and $u \overset{*}{\Rightarrow} v$ uses K_1 as control string and $v \overset{*}{\Rightarrow} w$ uses K_2 as control string.

Let C be a language over the alphabet lab(P). The language generated by G with control language C with appearance checking rules F is defined by :

$$L_{ac}(G, C, F) = \left\{ w \in T^* \mid D : S \overset{*}{\Rightarrow} w, D\, has\, a\, control\, word\, K\, of\, C \right\}$$

If $F = \phi$ the language generated is without appearance checking and denoted by L(G,C)

Whenever C is regular and G is of type i , where i = 0 , 1, 2 , 3 , then G is said to be a regular control grammar of type i.

Let $\mathcal{L}(i, j, k)$ denote a family of type i languages with type j control with k=0 , 1. k=0 denotes without appearance checking; k=1 denotes with appearance checking.

Example 8

Let $G=(N,T,P,S)$ be a regular control grammar where

$$N=\{A,B,C,D,S\}$$

$$T=\{a,b,c,d\}$$

$P:$

1. $S\to ABC$
2. $A\to aA$
3. $B\to bB$
4. $C\to cC$
5. $D\to dD$

6. $A \rightarrow a$

7. $B \rightarrow b$

8. $C \rightarrow c$

9. $D \rightarrow d$

Then, $lab(P) = \{1, 2, 3, 4, 5, 6, 7, 8, 9\}$

Let, $K = 1(24)^*(35)^* 6789.$ Clearly , K is regular. Then

$$L(G, K) = \{a^n b^m c^n d^m \mid n, m \geq 1\}$$

Some sample derivations are :

for $u = 124356789 \in K,$

$$S \underset{1}{\Rightarrow} ABCD \underset{2}{\Rightarrow} aABCD \underset{4}{\Rightarrow} aABcCD \underset{3}{\Rightarrow} aAbBcCD$$

$$\underset{5}{\Rightarrow} aAbBcCdD \underset{6}{\Rightarrow} aabBcCdD \underset{7}{\Rightarrow} aabbcCdD$$

$$\underset{8}{\Rightarrow} aabbccdD \underset{9}{\Rightarrow} aabbccdd$$

If $u = 124246789 \in K$

$$S \underset{1}{\Rightarrow} ABCD \underset{2}{\Rightarrow} aABCD \underset{4}{\Rightarrow} aABcCD \underset{2}{\Rightarrow} aaABcCD$$

$$\underset{4}{\Rightarrow} aaABccCD \underset{6}{\Rightarrow} aaaBccCD \underset{7}{\Rightarrow} aaabccCD$$

$$\underset{8}{\Rightarrow} aaabcccD \underset{9}{\Rightarrow} aaabcccd$$

Example 9

Let $G = (N, T, P, S)$ be a grammar with

$N = \{S, A, B, C\}$

$T = \{a, b\}$

$P:$

1. $S \to ABC$
2. $A \to aA$
3. $B \to aB$
4. $C \to aC$
5. $A \to bA$
6. $B \to bB$

7. $c \to bC$

8. $A \to a$

9. $B \to a$

10. $C \to a$

11. $A \to b$

12. $B \to b$

13. $C \to b$

and $lab(P) = \{1, 2, ...13\}$

$K = 1(234 + 567)^* (89(10) + (11)(12)(13))$ be a regular control on G.

$$L(G,K) = \{www \mid w \in \{a,b\}^+\}$$

Indian Parallel Grammars

In the definition of matrix , programmed , time-varying , regular control , and random context grammars , only one rule is applied at any step of derivation . In this section , we consider parallel application of rules in a context -free grammars (CFG).

Definition 8

An Indian parallel grammar is a 4-tuple $G = (N, T, P, S)$ where the components are as defined for a CFG . We say that $x \Rightarrow y$ holds in G for strings x, y over $N \bigcup T$, if

$$x = x_1 A x_2 A \ldots A x_n A x_{n+1}, \quad A \in N, \quad x_i \in (N \bigcup T) - \{A\}^*$$

for $1 \leq i \leq n+1$

$$y = x_1 w x_2 w \ldots w x_n w x_{n+1}, \qquad A \to w \in P.$$

i.e., if a sentential form x has an occurrences of the nonterminal A, and if $A \to w$ is to be used it is applied to all A's in x simultaneously. $\overset{*}{\Rightarrow}$ is the reflexive , transitive closure of $\Rightarrow$

$$L(G) = \left\{ w \mid w \in T^*, S \overset{*}{\Rightarrow} w \right\}$$

Example 10

We consider the Indian parallel grammar:

$$G = \left(\{S\},\{a\},\{S \to SS, S \to a\}, S\right).$$

Some sample derivations are

$$S \Rightarrow a$$

$$S \Rightarrow SS \Rightarrow aa,$$

$$S \Rightarrow SS \Rightarrow SSSS \Rightarrow aaaa$$ and

$$L(G) = \left\{a^{2^n} / n \geq 0\right\}.$$

It is clear from this example that some non-context free languages can be generated by Indian parallel grammars.

The other way round , the question is : can all context free languages (CFL) be generated by Indian parallel grammars ? Since the first attempt to solve this was made in (Siromoney and Krithivasan . 1974) , this type of grammar is called an Indian parallel grammar.