Material Science Research India Vol. **4**(2), 441-446 (2007)

A note on parallel communicating *T0L* array grammar systems

S. JEYA BHARATHI

Department of Mathematics, Thiagarajar College of Engineering, Madurai - 625 015 (India).

(Received: February 02, 2007; Accepted: May 22, 2007)

ABSTRACT

Parallel communicating L systems are systems having the communicating components as L-systems. In parallel communicating L-array systems we have 0L-(T0L) Array systems as communicating components. These models generates interesting pictures, some results on generative capacity are given.

Keywords: PC0L array systems, PCT0L-array systems.

INTRODUCTION

Parallelism is a very modern topic in Computer Science, important from the practical point of view and appealing from the theoretical one (Paun, 1992). L-Systems, introduced by Lindenmayer in connection with some problems in theoretical Biology, involves parallelism at every level of rewriting of the current sentential form (Rozenberg and Salomaa, 1980). Another type of parallelism occurs in the framework of grammar systems. There one does not have one typical generative device for producing a language, as in the classical formal language theory, but a system of generative devices, working together in some defined way for producing a language (Rozenberg and Paun). When the components of the system work sequentially, cooperating/distributed grammar systems exists (Paun, 1992). This topic was investigated in a series of papers (Csuhaj and Dassow 1993, Dassow and Paun 1990a, Dassow and Paun, 1990b, Paun 1992, Dassow and Paun 1996 from various points of view in Computer Science: artificial intelligence, and psychology etc. with the motivation that the components of a grammar system can also work in parallel and communicate in some way with each other. Thus the Parallel communicating grammar systems originate (Paun, 1992). In Parallel Communicating (PC) grammar system, the components of the grammar system work in parallel (each having its own sentential form) and communicate in some way (i.e when the query occurs, sending the currently generated string to other component). This Parallel Communicating grammar system whose components are L-systems (of various type), is mainly motivated by theoretical reasons, but seems to have biological applications in symbiosis, parasitism, simultaneous interrelated growing of plants and animals etc . Motivated by this we have proposed Parallel Communicating L-Array systems and Parallel Communicating Tabled L-Array system. We consider a specific type of L-Array systems defined in (Nirmal and Krithivasan, 1981) as the components of the system. This model is capable of generating interesting pictures. We give some hierarchy results in communicating array grammar and also examine some generative capacity.

Definition, examples and hirerarchy

In this section we define PC0LAS (Parallel Communicating 0L Array System), PCT0LAS

(Parallel Communicating Tabled 0L Array System). We assume the reader to be familiar with the theory of 2D languages. We illustrate the model with examples. We present some more results on hierarchy.

Definition

Parallel Communicating 0L Array System (PC0LAS) is a 3-tuple $G = (\Sigma, \omega, P)$ where

1. $\Sigma = \Sigma' \cup K$, Σ' is basic alphabet, $K = \{Q_1,Q_2,Q_3,\ldots, Q_n\}$ are query symbols.
2. $\omega = \{\omega_1,\omega_2,\ldots, \omega_n\}$ is the axiom of n-tuple and $w_i \in \Sigma^{**}$, $1 \le i \le n$.
3. $P = (P_1,P_2,P_3, \ldots, P_n)$ where $P_i = P_{i1} \cup P_{i2}$. Each P_{i1} is a nonempty finite subset of $\Sigma \times \Sigma^{**}$, is a finite set of pairs (a,x) with a in å and x in å** of dimensions r x s such that for each a in å at least one such pair is in P, $1 \le i \le n$. The pairs (a,x) are called the rules or productions and are written as $a \rightarrow x$. The communicating rules in P_{i2} are of the form

$$\begin{matrix} \alpha \\ \beta\ C\ \gamma \\ \delta \end{matrix} \rightarrow \begin{matrix} \alpha \\ \beta\ Qi\ \gamma \\ \delta \end{matrix} \quad \alpha,\beta,\gamma,\delta \in \Sigma^{**}$$, Qi is the

Query symbol, and $C \in \Sigma$.

The sets P_{i2} maybe empty, for $1 \le i \le n$. The derivation step is of two types as

i. $x_i \Rightarrow y_i$ if x_i has no query symbols $1 \le i \le n$. P_i
ii. $(x_1,x_2,\ldots,x_n) \Rightarrow (y_1,y_2,\ldots,y_n)$ where some x_i has query symbols

Q_j and communicates with jth component for $1 \le i,j \le n$.

In other words an n-tuple $(x_1,x_2,x_3,\ldots,x_n)$ directly yields $(y_1,y_2,\ldots,y_n)$ if either no query symbol appears in x_i and then we have a component wise derivation $x_i \Rightarrow y_i$ in G_i, $1 \le i \le n$, or x_i contains query symbols and then a communication step is performed ,as these symbols impose. Specifically each symbol Q_i is replaced by the corresponding component x_i, whereas the component G_i, resumes working from its axiom. The communication has priority over rewriting.

The Language generated by g is

$L(\gamma) = \{ x \in (\Sigma - K)^{++} / (\omega_1, \omega_2, \ldots, \omega_n) \Rightarrow^* (x,\alpha_2,\alpha_3,\ldots\alpha_n), \alpha_i \in \Sigma^{**}, 2 \le i \le n \}$.

In other terms, a derivation consists of repeated rewriting and communication steps, starting from the n- tuple of axioms. We retain in $L(\gamma)$ the array generated in this way on the first component, G_1 (Which is considered the master of the system) without containing query symbols. We shall denote by PC_nX be the family of languages generated by PC grammar having at most n components, $n \ge 1$.

i.e. To derive the configuration $(y_1,y_2,y_3,\ldots,y_n)$ either (i) No query symbol appears in $x_1,x_2,x_3,\ldots,x_n$ and then we have a component wise derivation, $x_i = y_i$, in each component P_{ij}, $1 \le i \le n$, $1 \le j \le n$ (one rule is used in each component P_i), except for the case when x_i is terminal, $x_i \in T^*$ then $x_i = y_i$.

or

(ii)Query symbols occur in some x_i. Then a communication step is performed. Each occurrence of Q_j in x_i is replaced by x_j, provided x_j does not contain query symbols. In a communication step, the communicated string x_j replaces the query symbol Q_j. Then the grammar G_j resumes rewriting beginning again from its axiom. The communication has priority over the effective rewriting: no rewriting is possible as long as at least one query symbol is present. If some query symbols are not satisfied at a given communication step, then they may be satisfied at the next step.

Definition

Parallel Communicating Tabled 0L Array System (PC0LAS) is a 3-tuple $G = (\Sigma, \omega, P)$ where Σ, ω are as in definition of PC0LAS. $P = (P_1,P_2,P_3,\ldots,P_n)$ where $P_i = P_{i1} \cup P_{i2}$. Each P_{i1} consists of a finite set $\{P_{i1}^{1},P_{i1}^{2},Pi1^3,\ldots,P_{i1}^{f}\}$, f ³ 1. Each P_{i1}^{t} is a finite subset of $\Sigma \times \Sigma^{**}$ called a table with the following two conditions.

i. $(\forall P)_{pi}\ (\forall a)_{\Sigma}\ (\exists a)_{\Sigma^{**}}\ (<a,\alpha> \in P)$;
ii $(\forall a)_{\Sigma} (\exists <a,\alpha>)_{P}$, a's are of the same dimension.

P_{i2} is defined as in definition of PC0LAS.

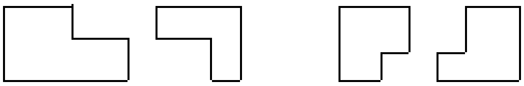
The derivation step is of two types as

i $x_i \underset{p_i}{\Rightarrow} y_i$ if x_i has no query symbols $1 \le i \le n$.
ii $(x_1,x_2,x_3,\ldots,x_n) \Rightarrow (y_1,y_2,y_3,\ldots,y_n)$ where some x_i has query symbols,

Q_j and communicates with j_{th} component for $1 \le i,j \le n$.

The derivations are defined as follows $P = P_r = \{P_1,P_2,\ldots,P_j\}$ where $P_i = P_{i1} \cup P_{i2}$, then we apply to ù the rules from the tables of Pr (Pc) row by row (column by column) i.e., we choose a rule in P_r and apply the rules to the first row (column). Then we choose another set of rules in P_r and apply the rules to the second row (column). Proceeding in this manner we apply the rules from a table to row (column) of ω. If the query symbol appears, the communication takes place so as to get a rectangular array and if no query symbol appears in the master system then the communication does not work and the derivation stops. If the resultant array is rectangular then the rules from the table Pr (Pc) can be applied again, otherwise the derivation comes to an end.

In PCT0L array system, the query symbols can occur anywhere by rule provided. The resultant after the communication step, is rectangular. The row or column rules are applied in such a way that the rectangular format is maintained. The following figures are the models to form the rectangle pattern after the communication.



2.3 Definition

Partial Parallel Communicating Tabled 0L Array System (Part CPCT0LAS)

is a five tuple G = (V, P, w, S, h) where

i) (V, P, ω) is a part PC0LAS

ii) Σ is a non-empty finite set called target symbol.

iii) h is a partial coding from V into Σ.

Starting from ω, arrays are derived by part PCTMLS and then coding h is applied to these arrays

$$h\begin{pmatrix} a_{11}a_{12}\ldots a_{1n} \\ \ldots\ldots\ldots\ldots \\ \ldots\ldots\ldots\ldots \\ \ldots\ldots\ldots\ldots \\ a_{m1}\ a_{m2}\ldots a_{mn} \end{pmatrix} = \begin{pmatrix} h(a_{11})\ldots h(a_{mn}) \\ \ldots\ldots\ldots\ldots \\ \ldots\ldots\ldots\ldots \\ \ldots\ldots\ldots\ldots \\ h(a_{11})\ldots h(a_{mn}) \end{pmatrix}$$

is undefined if there is at least one a_{ij} for which $h(a_{ij})$ is not defined.

Model

Let G = (V, P, ω, Σ, h) be a part CPCTAS (of degree four) $\gamma = (G_1,G_2,G_3,G_4)$.

where $V = \{X, Y, Z, a, b, c, Q_2, Q_3, Q_4\}$

$\Sigma = \{a, b, c\}$

$$P_{11}=\left\{\begin{array}{l} \#\quad\quad X \\ \#X\# \to b,\ \#X \to Q_2\# \end{array}\right\}$$

$$P_{12}=\left\{\begin{array}{ccc} & & X \\ & \# & b \\ \#X\# \to & & b \end{array}\right\}$$

$$P_{21}=\left\{\begin{array}{l} \#\quad\quad\quad\quad \# \\ Y\# \to bY,\ Y\# \to Q_3\# \end{array}\right\}$$

$$P_{22}=\left\{\begin{array}{l} \# \\ Y\# \to bbY \\ \# \end{array}\right\}$$

$$P_{31}=\left\{\begin{array}{l} \#Z\# \to c,\ \#Z\# \to Q_2\# \\ \#\quad\quad Z\quad \# \end{array}\right\}$$

$$P_{32}=\left\{\begin{array}{l} \#Z\# \to c, \\ \#\quad\quad c \\ \quad\quad\ \ Z \end{array}\right\}$$

$$P_{41}=\left\{\begin{array}{l} \quad\#\quad\quad\quad\#\quad\quad\quad\quad X \\ \#X\# \to Xb,\ \#X\# \to bb \end{array}\right\}$$

$$P_{42}=\left\{\begin{array}{l} \quad\# \\ \#X\# \to Xbb \\ \quad\# \end{array}\right\}$$

h(a) =, h(b)=, h(a)=h(b)=h(c) = x

$$\omega=\left\{\begin{array}{l} X \\ a,\ bY,c,Xb \\ \quad\quad Z \end{array}\right\}$$

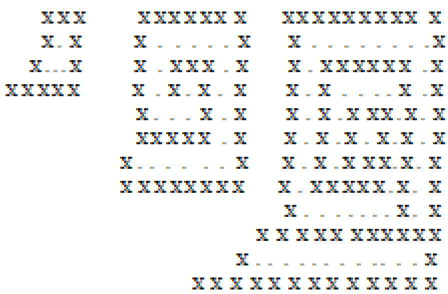
$$\Rightarrow\left\{\begin{array}{l} X \\ b, b\,b\ Y, c, Xb\,b\,b \\ a\quad\quad c \\ \quad\quad\ \ c \\ \quad\quad\ \ z \end{array}\right\}$$

$$\Rightarrow\left\{\begin{array}{l} Q\quad\quad X \\ b, bbQ, c, bbbbb \\ \quad\quad\ c \\ \quad\quad\ c \\ \quad\quad\ Q \end{array}\right\}$$

$$\Rightarrow\left\{\begin{array}{l} \quad b\,b\,b \\ \quad b\quad c \\ X\,a\quad c, bY, c, Xb \\ b\,b\,b\,b\,b\quad\quad Z \end{array}\right\}$$

$$\Rightarrow \left\{\begin{array}{l} X\,b\,b\,c \\ b\,b\ \ c \\ b\,a\,c, b\,b\,b\,b\,b\,Y, c, X\,b\,b\,b\,b\,b \\ b\,b\,b\,b\,b \qquad\quad c \\ \qquad\qquad\quad\ \ c \\ \qquad\qquad\quad\ \ c \\ \qquad\qquad\quad\ \ c \\ \qquad\qquad\quad\ \ Z \end{array}\right\}$$

Spirals of the following from are generated



RESULTS

Clearly $X = PC_1X \subseteq PC_2X \subseteq \ldots$ for all $X \in \{$ 0LAL ,T0LAL $\}$. We shall show that $X \subset PC_2X$ (two components are strictly powerful than one) for $X \in \{$ 0LAL ,T0LAL $\}$.

Theorem

(i) $PC_2$0LAL – E0LAL $\neq \varphi$.

(ii) $PC_2$0LAL – 0LAL $\neq \varphi$.

Proof: Follows from the definition of PC0LAL. In particular, model-1 cannot be generated by any E0LAL.

Corollary

(i) 0LAL $\subset$ PC_2 0LAL.

(ii) T0LAL $\subset$ PC_2 T0LAL.

Theorem

$$\text{The language } L = \left\{\begin{array}{cc} & a \\ a & a \\ a & a \end{array}\right\} \text{is not in } PC_n \text{ TOTAL}, n \geq 1$$

Proof :

Suppose $L = L(\gamma)$ for a PC_n TOLAL where $\gamma = (G_1, G_2, G_3, \ldots, G_n)$ be the TOLAL components , $G_i = (V_i, \omega_i, \{P_{1i}, P_{2i}, \ldots, P_{ki}\})$, $1 \leq i \leq n$. $V_i = \{a\}$.

Clearly

$$\omega_1 = \left\{\begin{array}{cc} & a \\ a & a \\ a & a \end{array}\right\} \quad \begin{array}{ccccccc} & & a & & a & & \\ & a & a & & a & & a \\ \text{if } a & \Rightarrow & a & \text{or} & a & \Rightarrow & a \end{array}$$

can be obtained from G_1, then $L(\gamma) \neq L$. Hence at least a communication step is necessary.

The rule,

$$\begin{array}{ccc} \alpha & & \alpha \\ \beta\ a\ \gamma & \to & \beta\ x\ \gamma \\ \delta & & \delta \end{array}, \; x \in \{a, Q_2, Q_3, \ldots Q_n\}$$

can be used in $\begin{array}{c} a \\ a \end{array} \Rightarrow \begin{array}{c} x \\ x \end{array}$ and $\begin{array}{cc} a & \\ a & a \end{array} \Rightarrow \begin{array}{cc} x & \\ x & x \end{array}$.

But the arrays in the communication must be ‘a’. If an array $(a)_t$, $t \neq 1$ is communicated then either λ or a_{2t}, a_{3t} are also generated. Hence from ω_1 we get only ω_1 and no other array. i.e., $L \notin PC_n$ T0LAL.

Corollary

E0LAL $\subset$ PC_n T0LAL $\neq \varphi$.

$$\text{Proof:} \quad L = \left\{\begin{array}{cc} & a \\ a & a \\ a & a \end{array}\right\} \in \text{E0LAL}$$

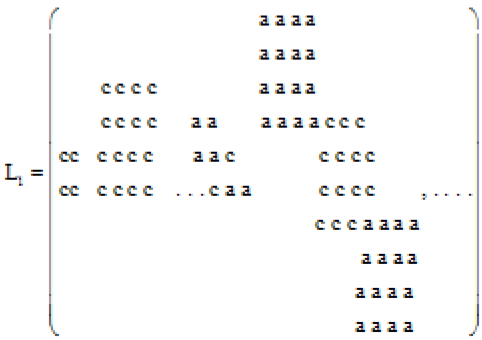
but not in PC_nT0LAL.

We present below some more hierarchy and the properties of the above family.

0LAS $\subset$ PCT0LAS

Proof

Inclusion follows from definition. For proper inclusion, consider a PCT0LAS



generated by PCT0LAS array system (of degree two) $\gamma = (G_1, G_2)$
let G = ({ a,b,c,Q_2 }, {c,a}, P) $P = P_1 \cup P_2$.

$$P_{11} = \left(c \to \begin{array}{c} c\,c \\ c\,c \end{array},\; b \to \begin{array}{c} b\,b \\ b\,b \end{array},\; a \to \begin{array}{c} a\,b \\ a\,b \end{array},\; \begin{array}{c} c \\ cc\,\# \\ \# \end{array} \to \begin{array}{c} c \\ cQ_2\# \\ \# \end{array},\; \begin{array}{c} \# \\ c\,c \\ c \end{array} \to \begin{array}{c} \# \\ Q_2c \\ c \end{array} \right)$$

$$P_{21} = \left(c \to \begin{array}{c} a\,a \\ a\,a \end{array} \right)$$

$$(ca) \Rightarrow \begin{pmatrix} cc & a\,a \\ aa & a\,a \end{pmatrix}$$

$$\Rightarrow \left(\begin{array}{ccc} & & a\,a\,a\,a \\ & & a\,a\,a\,a \\ Q_2 & c, & a\,a\,a\,a \\ c, & Q_2 & a\,a\,a\,a \end{array} \right)$$

$$\left(\begin{array}{l} a\,a\,a\,a \\ a\,a\,a\,a \\ a\,a\,a\,a \\ a\,a\,a\,a \\ \quad c\,a\,a\,a\,a\;,\; a \\ \qquad a\,a\,a\,a \\ \qquad a\,a\,a\,a \\ \qquad a\,a\,a\,a \end{array} \right)$$

$$L_1 = \left(\begin{array}{lllll} & & & a\,a\,a\,a & \\ & & & a\,a\,a\,a & \\ & c\,c\,c\,c & & a\,a\,a\,a & \\ & c\,c\,c\,c & a\,a & a\,a\,a\,a\,c\,c\,c & \\ cc & c\,c\,c\,c & a\,a\,c & \quad c\,c\,c\,c & \\ cc & c\,c\,c\,c & \ldots c\,a\,a & \quad c\,c\,c\,c & ,\ldots \\ & & & \quad c\,c\,c\,a\,a\,a\,a & \\ & & & \qquad a\,a\,a\,a & \\ & & & \qquad a\,a\,a\,a & \\ & & & \qquad a\,a\,a\,a & \end{array} \right)$$

is not an 0LAS.

Corollary

We can prove the following corollary trivially from the definition

i. PC0LAS $\subset$ PCT0LAS

ii. T0LAS $\subset$ PCT0LAS

The following hierarchy is true obviously, when the number of tuple is more any model can be done in fewer steps.

$PC_1T0LAS \subset PC_2T0LAS \subset PC_3T0LAS, \ldots$

Theorem

There exist some finite languages, which are not PC0L array languages.

Proof:

Let $L = \left\{ a\,,\; \begin{array}{cc} a & a \\ a & a \end{array}\,,\; \begin{array}{cc} a & a \\ a & a \\ a & a \end{array} \right\}$ be a finite language, with $G = (\Sigma, \omega, P)$ $\omega = a$

As $\lambda \notin L$, G should be a propagating system. Hence w = a. To get a b, we

$$P_1 = \left\{ \begin{array}{c} \# \\ a\,\# \end{array} \to Q_2,\; \begin{array}{c} \# \\ \#\,a \end{array} \to Q_2,\; a\,\# \to Q_2,\; \#a \to Q_2 \right\}$$

$$P_2 = \left\{ a \to \begin{array}{c} a\,a \\ a\,a \end{array} \right\}$$

This will generate the language

$$\left\{ a,\; \begin{array}{c} a\,a \\ a\,a \end{array},\; \begin{array}{c} a\,a\,a\,a \\ a\,a\,a\,a \\ a\,a\,a\,a \\ a\,a\,a\,a \end{array} \ldots \right\}$$

So arrays, which are not in L, will be generated by G. Hence L is not a PC0LAL.

Theorem

There exists an algorithm which for an arbitrary PCTAS G produces G° and a partial coding h such that L (G) $\equiv$ h (L (G°)) in the normal form

Proof:

Let G = (V,P,ω,Σ) be an PCTAS, then

Let G' =(V$\cup${S,F},P,S', Σ',h) be an part CPCTAS, where {S,F}$\notin$V and P' = $P_0 \cup \{P_i' \;\forall P_i \in P\}$

Let h be a partial coding h: V$\to\Sigma$', such that h is defined if h(a_{ij}) defined for all i,j h is not defined if atleast one a_{ij} is not defined.

The table in P' was constructed in such a way as to provide a possibility for the following simulation (in G°) of derivations in G.

Let $P_0 = \{S \to \omega,\ a \to M'r_0s_0\ ;\ a \in V \cup \{F\}\}$,
$P'_i = P_i \cup \{S \to M'r_is_i,\ F \to M'r_is_i\}$ $1 \le i \le n$,
Where $M'r_is_i$ are rejection arrays.
Hence it is clear that L (G) = L (G°).

If D is a derivation in G, then we construct a corresponding derivation D° in G° in such a way that h gives the partial coding. We keep track of all productive occurrences such that the partial coding h is defined for all.

Conclusion

A new type of communicating L-array model is formulated in this paper. This model generates interesting pictures. Some results on hierarchy are given. Seven other variation of this model can be made using the concept of E0LAL and ET0LAL.

REFERENCES

1. Csuhaj-Varju E.Dassow,J, On Cooperating/ distributed grammar systems, Elektr.Inform Kybern, (1993).
2. Dassow.J., Paun Gh, Regulatedrewriting in formal languages theory, Akademic Verlag, Berlin,Springer-Verlag, Berlin, 1990a (1989).
3. Dassow.J., Paun Gh: On some variants of cooperatin/distributed grammar systems, *Stud. Cerc. Matem,* **42**(2), 153-165 (1990b).
4. Gheoreghe Paun: Parallel Communicating Grammar Systems of L-systems, Springer-Verlag, 405-417 (1992).
5. N.Nirmal & K.Krithivasan, E0L and ET0L Array Languages, *Proc. Indian AcadSci* (*Math.Science*),**90**(3), 167-180 (1981).
6. G.Rozenberg and A.Salomaa, The Mathematical Theory of L-Systems, AcademicPress, New york, (1980).