CONTEXT - FREE EDGE REPLACEMENT GRAPH P SYSTEM

Thanga Murugeshwari V*, Emerald Princess Sheela J.D.,**

* Department of Mathematics, St.Joseph's College of Arts and Science for Women, Hosur
** Department of Mathematics, Queen Mary's College, Chennai

Abstract- The complicated networks in the nature are the recursive patterns which overlap with each other. These patterns can be captured by edge replacement graph grammars. In this paper, making use of edge replacement recursive graph grammar, we define context free edge replacement graph P system. The two variants of P system namely rewriting P system and P system with conditional communication are used with edge replacement recursive graph grammar. We exhibit the generation of some special kind of graphs such as double shell graph, shell butterfly graph and banana trees graph using context-free edge replacement graph P system, besides establishing some comparison results.

Index Terms- Context- free, Edge Replacement,Graph Grammar, P system

I. Introduction

Graph grammar which involves rewriting of graphs, is in fact an extension of the well-known string grammars. An account of several theoretical concepts and models relating to graph grammars can be found in handbook of graph grammars and graph transformation [8]. A graph grammar can be defined as a 3 –tuple (Mother graph, Daughter graph, Embedding Mechanism), with certain components of the mother graph being replaced by the daughter graph based on the embedding mechanism.

Natural computing is a field of research with different computational models which are based on the way nature computes. A new computational model based on the notion of membrane structure and functioning of the living cell was introduced by Paun [5, 7] and is called a P system.

The main aim of membrane computing is to define a computing model which is inspired by cell biology. P system is universal, since most of the classes of P system are Turing complete. There are various types of P systems namely cell like P system, tissue like P system, neural like P system, spiking neural P system and many others [5,7]. In a P system the computations are done in parallel[7].

In this paper, a new P system model called context-free edge replacement graph rewriting P system (CFEGRP) and Context-free edge replacement graph P system with conditional communication (CFEGPCC) are introduced and we discuss some comparison results.

II. Preliminaries

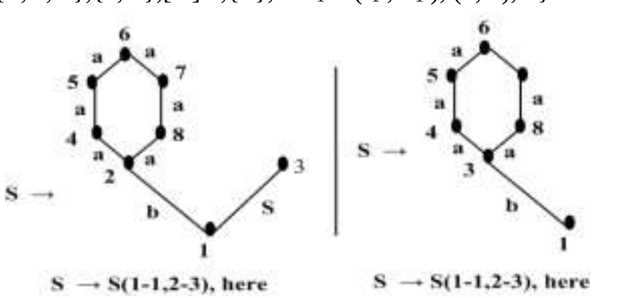
An *edge replacement recursive graph grammar*[4] is a 5 tuple $G_r = (N, T, E, \Omega, P)$ where N represents the alphabet set of node labels, T represents the alphabet set of terminal node labels, $T \subseteq V$, E represents the alphabet set of edge labels, Ω represents the alphabet set of terminal edge labels, $\Omega \subseteq E$, P represents finite set of productions of the form (r, G_p, C) where G_p is a graph, $r \in E\text{-}\Omega$ and there is at least one edge in G_p labelled r and C is an embedding mechanism with the set of connection instruction i.e., $C \subseteq (N\times N, N\times N)$ where N is the set of nodes of G. A connection instruction $(a, b ; c, d) \in C$ implies that derivation can take place by replacing a, c in one instance of G with b and c respectively, in another instance of G_p. All the edges incident to "a" are incident to "b". All the edges incident to "c" are incident to "d". All the edges incident with "b" and "d" remains unchanged. An *(n, k)− Banana tree* [3] is a graph obtained by connecting one leaf of each of n copies of a K star graph with the single root vertex that is distinct from all the stars. A *Shell graph* [3] is defined as Cycle Cn with (n − 3) chords sharing a common end point called the apex. A *Double Shell graph* [3] of two disjoint Shell with a common vertex. A *Shell Butterfly graph* [3] is Double Shell graph with exactly two pendent edges at the apex. A *Star graph* [3] is a complete bipartite graph $K_{1,n}$. A *context-free graph grammar CFGGr* [2] is 5− tuple CFGGr $= (\alpha_{NT}, \alpha_{TE}, \beta, E, S, CR)$ where α_{NT} represents the nonterminal node alphabets , α_{TE} represents terminal node alphabets, β represents link alphabets, E represents the terminal edge labels, S is the start label (axiom) and CR is the the context-free production rules. The context-free rules CR consists of a non-terminal in the left hand side and a connected graph over $\alpha_{NT} \cup \alpha_{TE}$ in the right hand side with input and output nodes. During the derivation the non-terminal node is replaced by the connected structure such that every edge entering the non-terminal node should enter the input node of the replacing structure. Similarly every edge exiting the non-terminal node should exit from the output node of the replacing structure. A *string rewriting P system* [1, 6] is a construct SRP = $(N, T, M, w_1, w_2, \cdots, w_n, R_1, R_2, \cdots, R_n, P_1, P_2, \cdots, P_n, (n, d), O_m)$ where N and T represent finite sets of non-terminals and terminals respectively, M represents a membrane structure with n membranes and depth d, which are labeled by numbers in the set $\{1, 2, \cdots, n\}$ with the skin membrane being labelled as 1. w_i' s are the strings over $N \cup T$ where $i = 1, 2, \cdots n$. R_i' s are the context-free rewriting rules in each membrane of the form $A \rightarrow \eta(trgt)$, $A \in N$ and $\eta \in (N \cup T)^*$, $trgt \in \{here, out\} \cup \{in_j \mid 1 \leq j \leq n\}$ where $i = 1, 2, \cdots n$. P_1, $P_2 \cdots P_n$ are partial ordered relations over the rules and O_m is the output membrane. The details on derivations can be found in [1, 6]. A *parallel rewriting P system with conditional communication* [9] is a construct $PRPCC = (N, T, M, w_1, w_2, \cdots, w_n, R_1, R_2, \cdots, R_n, (n, d))$ where N and T represent finite sets of non-terminals and terminals respectively; M represents a membrane structure with n membranes and depth d, which are labeled by numbers in the set $\{1, 2, \cdots, n\}$ with the skin membrane being labelled as 1, w_i' s is are the strings over $N \cup T$ where $i = 1, 2, \cdots, n$ and R_i's

III. CONTEXT-FREE EDGE REPLACEMENT GRAPH REWRITING P SYSTEM

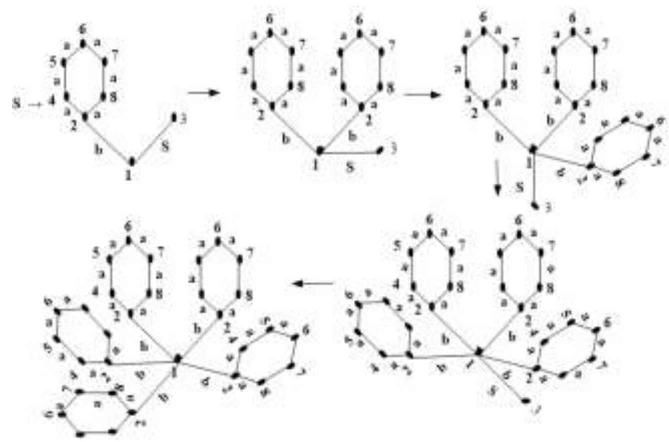
Definition 1: A ***Context-free edge replacement graph rewriting P system (CFEGRP)*** is a construct $\pi_{ed} = (N, T, E, M, E_1, E_2, \cdots, E_n, CR_1, CR_2, \cdots, CR_n, (n, d), O_m)$, where N represents the finite set of node labels, T represents the finite set of edge labels, E represents the set of terminal edges available in T, M represents the membrane structure with n membranes and depth d, which are labeled by numbers in the set $\{1, 2, \cdots, n\}$ with the skin membrane being labelled as 1. E_i is the finite set of non-terminal edges over T initially present in the region i, where $i = 1, 2, \cdots, n$. $CRi = (R_i, P_i)$ is the edge replacement recursive graph rules such that $R_i = (B \rightarrow A(trgt))$ where B is recursively replaced with the graph A with the help of connection instructions. $trgt \in \{here, out\} \cup \{in_j \mid 1 \leq j \leq n\}$. P_i represents the priority rule in the i^{th} membrane. Throughout this paper $P_i = \varphi$, node labels are given from the natural number set and all the node alphabets are terminal node alphabets. O_m is the output membrane.

$L(\pi_{ed})$ is the language generated by a system π_{ed}. The family of languages generated by context-free node replacement graph rewriting P system is denoted by $CFEGRP^d{}_n$ (nPri or Pri). Here n is the total number of membranes in the system, d is the depth, nPri denotes that there is no priority rules in the P system and Pri denotes that there is a priority in the application of rules.

Example 1 : Consider the example $\pi_{ed1} = (\{1, 2, 3, 4, 5, 6, 7, 8\}, \{S, a, b\}, \{a, b\}, [1\ ]1, \{S\}, CR_1 = (r_1, P_1), (1, 1), 1)$ with



$= (R_i, Pe_i, Fo_i)$ with R_i representing the context-free rewriting rules where $i = 1, 2, \cdots n$, Pe_i' s and Fo_i' s are permitting and forbidding conditions associated with the regions $1, 2, \cdots, n$. We refer to [9] for the notion of rewriting P systems with conditional communication in unique parallelism mode. But we restrict ourselves to only two modes of permitting and forbidding conditions, namely, empty and symbol checking conditions. Also, unique parallelism mode is used, in which exactly one symbol with respect to one rule is applied to all occurrences of that symbol. The notion of maximal parallelism can be referred from [7]



The above P system in single membrane generates the following graph which consists of a Star graph with each pendant vertices has Cycle of length 6.

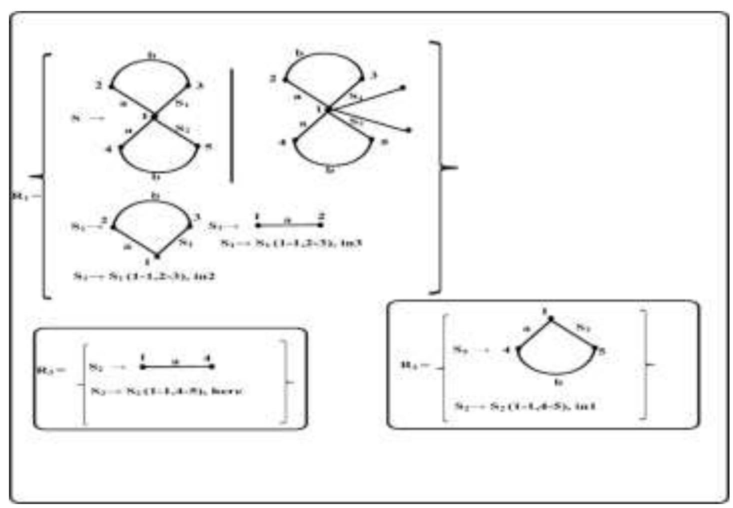


FIGURE. 1: The language $L(\pi_{ed1})$

Theorem 1: $CFEGRP^1{}_1(nP\ ri) \subset CFEGRP^2{}_3(nPri)$.

Proof : The inclusion is by definition. Proper inclusion is now established.

Consider a P system π_{ed2} = ({1, 2, 3, 4, 5},{S, S1 , S2 , a, b},{a, b},$[_1 [_2]_2 [_3]_3]_1$], {E_1 = S}, $CR_1 = (R_1 , P_1)$, $CR_2 = (R_2 , P_2)$, $CR_3 = (R_3 , P_3)$, (3, 2), 3) with rules diagrammatically represented in the following Fig. 2

FIGURE. 2: Representation of π_{ed2} in a P system

The above P system generates the class of Double Shell graph and Shell Butterfly graph with equal number of Shells on both sides as shown in Fig.3., which cannot be generated in one membrane.

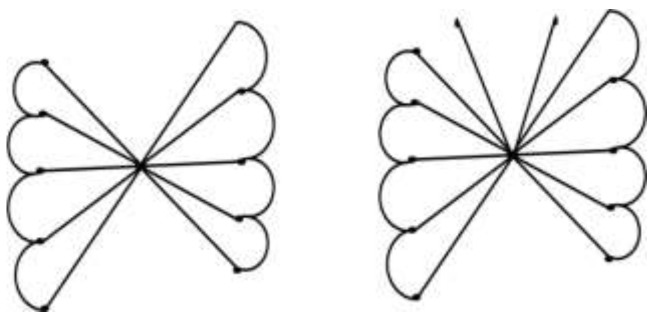


FIGURE. 3: L(π_{ed2}) Double Shell graph and Shell Butterfly graph

IV. CONTEXT-FREE EDGE REPLACEMENT GRAPH P SYSTEM WITH CONDITIONAL COMMUNICATION

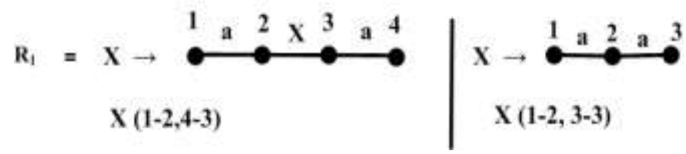
We introduce context-free edge replacement graph P system with conditional communication.

Definition 2: A ***Context-free edge replacement graph P system with conditional communication (CFEGPCC)*** is a construct π_{ed} = (N, T, E, M, E_1 , E_2 , ⋯, E_n, (R_1 , Per_1 , For_1), (R_2 , Per_2 , For_2), ⋯(R_n, Per_n, For_n), (n, d)) where N represents the finite set of node labels, T represents the finite set of edge labels and terminal edge labels, E represents the set of terminal edges available in T, M is a membrane structure with n membranes and depth d, which are labeled by numbers in the set {1, 2, ⋯, n} with the skin membrane is labelled as 1, E_i is the finite set of non-terminal edges over T initially present in the region i, i = 1, 2, ⋯, n of the system. R_i is the edge replacement recursive graph rules that consists of a non-terminal edge label along with the connection instructions. The conditional communication has empty and symbols checking, where empty has no restrictions in entering and existing graphs from one membrane to the other and the symbols checking has rules for entering the inner membrane and exiting the outer membrane separately.

L(π_{ed}) is the language generated by a system π_{ed} . The family of languages generated by context-free edge replacement graph P system with conditional communication is denoted by $CFEGPCC^d{}_n(\alpha, \beta)$. Here n is the total number of membranes in the system, d is the depth and α, β ∈ {empty, symbol} denote the permitting and forbidding conditions for communication.

Example 2 : Consider the example π_{ed3} = ({1, 2, , 3, 4},{X, a, b},{a, b},$[_1]_1$,{X}, (R_1 , Per_1 , For_1), (1, 1)) with the rules from context-free graph grammar

Per_1 = (True, out) and For_1 = (X, notout).



The above system generates the context-free string graph language, L(πe3) = {(a^n b^n) • /n ≥ 1} which is shown in the following Fig. 4.

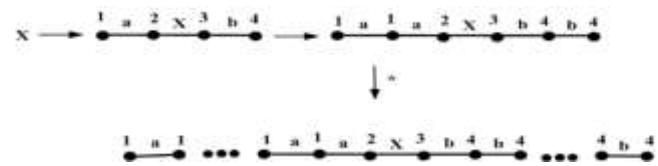


FIGURE. 4: The language L(π_{ed3})

Theorem 2. $CFEGPCC^1{}_1$ (empty, symbol) ⊂ CF $EGPCC^2{}_2$ (empty, symbol).

Proof The inclusion is by definition. Proper inclusion is now established.

Consider the language L_7 = B(n, k) (Banana trees).

Construct a P system in unique parallelism mode π_{ed4} = ({1, 2, 3, 4, 6, 7, 8},{S, D, a, b},{a, b},$[_1 [_2]_2]_1$] , {E_1 = S},{E_2 = φ}, (R_1 , P er_1 , For_1), (R_2 , Per_2 , For_2), (2, 1)) with the rules diagrammatically represented in the following Fig .5

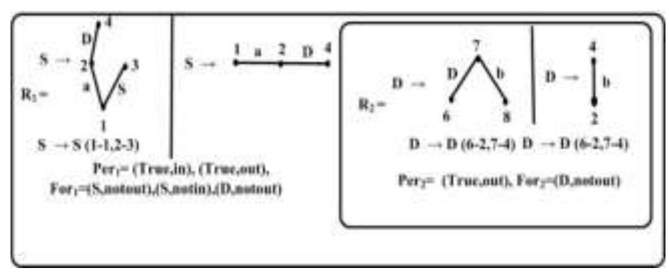


FIGURE 5: Representation of π_{ed4} in a P system

Initially, when the rule S in membrane one is applied, the edge (1, 3) is replaced by (1, 2) which increases the tree structure when used recursively. At any time the terminal rule can be applied which enables the graph to move from membrane 1 to 2. Then the rules for D is applied in membrane 2 where the branches are expanded by recursive application of the rule. Finally the terminal rule for D in membrane 2 is applied. The language generated by the above P system in two membrane is the class of Banana trees B(n, k). But the above class Banana tree cannot be generated in a single membrane. The generation of Banana tree B(4, 3) is shown in Fig 6.

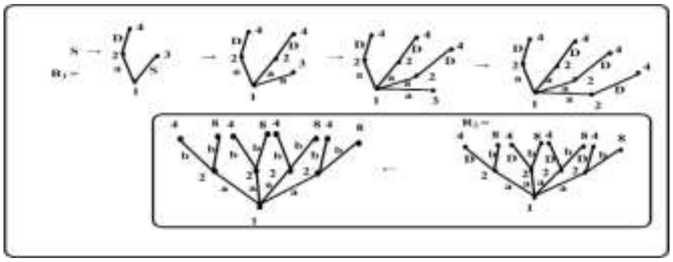


FIGURE. 6: The language L(π_{ed4})−Banana tree (4, 3)

V. CONCLUSION

A new P system model called context-free edge replacement graph P system is introduced and some comparison results are discussed. The generative power of edge replacement graph P systems considered in this paper is increased when the number of membranes is increased.

REFERENCES

1. P. Bottoni, A. Labella, C,Mart´ın-Vide , Gh . Paun, Rewriting P systems with Conditional Communication. In: W. Brauer, H. Ehrig , J. Karhum¨aki , A. Salomaa (eds) Formal and Natural Computing. Lecture Notes in Computer Science, Vol 2300. Springer, Berlin, Heidelberg, 2002.
2. P. Della Vigna and C. Ghezzi , "Context-free Graph Grammars", *Information and Control*, 37, 207 − 233, 1978.
3. J. A. Gallian, "A Dynamic Survey of Graph Labelling", *The electronic Journal of Combinatorics*, 16, 2013
4. J.P. Kukluk , L. Holder, D.J. Cook, "Inference of Edge Replacement Graph Grammars", *International Journal on Artificial Intelligence Tools – IJAIT*, 17, 496 − 501, 2007.
5. Gh. Paun, "A guide to membrane computing", *Theoretical Computer Science*, 287, 73 − 100, 2002.
6. Gh. Paun, "Computing with membranes", *Journal of computer and system sciences*,61(1), 108 − 143, 2000.
7. Gh. Paun, G. Rozenberg, A. Saloma, "The Oxford Handbook of Membrane Computing", *Oxford University Press*, 2010.
8. G. Rozenberg, "Handbook of Graph Grammars and Computing by Graph Transformation", *World Scientific*, 1997.
9. K.G. Subramanian, S. Hemalatha, C. Sri Hari Nagore, M.Margenstern , "On the Power of P systems with Parallel Rewriting and Conditional Communication", *Romanian Journal of Information Science and Technology*, 10(2), 137−144, 20

AUTHORS

First Author – Thanga Murugeshwari V, M.Sc., M.Phil.,Ph.D, Assistant Professor, St. Joseph's College of Arts and Science for Women, Hosur, Tamil Nadu, India , thangammathe05@gmail.
Second Author – Emerald Princess Sheela J.D.M.Sc.,M.Phil.,Ph.D, Assistant Professor, Queen Mary's College, Chennai, Tamil Nadu, India
Correspondence Author –Thanga Murugeshwari V, M.Sc., M.Phil.,Ph.D, Assistant Professor, St. Joseph's College of Arts and Science for Women, Hosur, Tamil Nadu, India , thangammathe05@gmail.