International Journal of Emerging Trends in Science and Technology
IC Value: 76.89 (Index Copernicus) Impact Factor: 4.219 DOI: https://dx.doi.org/10.18535/ijetst/v4i9.43

# Contextual Array Kernel P System

Ajitha P S* and Dersanambika K.S †

*Department of Mathematics,

College Of Engineering,Kottarakara, Kerala 691531, India

† Department of Mathematics,

Fatima Mata National College, Kollam 691 001, India

**Abstract**

**In this paper we consider the contextual way of handling array objects in Kernel P Systems.We generate some languages in contextual array Kernel P systems.Hollow rectangles and squares are also generated using contextual array Kernel P Systems.**

**Keywords: Psystem, KP System, Contextual grammar ,Contextual arrray Kernel P System .**

## 1 Introduction

Membrane computing constitutes an active research topic with Natural computing introduced by Gh.Paun in 1998. It is a theoretical machine oriented model, where the computational devices called P Systems, aim at capturing

*email- psajithamurali@gmail.com
†email-dersanapdf@yahoo.com

relevent features of living cells.

A large number of different variants of P System have been introduced so far. The concept of Kernel P System (KP System) has been introduced inorder to include the most used concepts from P System into a single coherent setting which allows various solution to a certain problems to be specified, compared and formally verified.The rules selected against the multi set of objects available in each compartmet are executed in accordance with the well defined execution strategy. These rules are responsible for either transferring and moving objects between compatments or for changing the structure of the model.

Contextual grammars were introduced by S. Marcus as another model to describe natural languages. In this paper we recalls the definition of contextual array grammars with choice, without choice and contextual grammars with restricted selection.We defined the contextual array KP System and contextual array KP System with erased context.We constructed some examples of arrays with various selections and described the specialities of the languagge generated. Also discussed some properties of array contextual KP Systems and established the correspondance between external array contextual KP System and external array contextual KP System in normal form.

## 2 PREREQUISITES

Contextual Array Grammars

Let I be an alphabet, a finite non empty set of symbols. A string over I is a finite sequence of symbols from I. The set of all arrays over I ( including the empty array $\Lambda$ which has zero rows and zero columns) is denoted by $I^{**}$ and $I^{++} = I^{**} - \{\Lambda\}$

For strings x and y $x = a_1a_2...a_n$ $y = b_1b_2...b_n$ , the concatenation xy denotes $a_1a_2...a_nb_1b_2...b_n$, For arrays we define two types of concatenation, viz row and column catenation.

**Definition 2.1.** *For an array X of dimension $m \times n$ and an array Y of dimension $m' \times n'$ the column concatenation $X \oplus Y$ is defined only when $m = m'$ and the row catenation $X \ominus Y$ is defined only when $n = n'$*
*In row concatenation $X \ominus Y$ , Y is attached below X. In column concatenation $X \oplus Y$ , Y is attached to the right of X*

**Definition 2.2.** An external array contextual grammar with choice (EACGC) is a 5- tuple

$$G = (V, B, L\ominus, L_{\oplus}, \Psi) \ \ Where$$

1) V is a finite non empty set of symbols called the alphabet
2) B is a finite subset of $V^{**}$ whose elements are called axioms.
3) $L\ominus$ is a finite set $\{\rho_1, \rho_2, ..., \rho_n\}$ of array over $V^{**}$ \$ V** each $\rho_i$ having a bounded number of rows.
4) $L_{\oplus}$ is a finite set $\{\rho'_1, \rho'_2, ..., \rho'_n\}$ of array over $V^{**}$ \$ V** each $\rho^1_i$ having a bounded number of columns.
5) $\psi$ is a function $\psi : V^{**} \rightarrow_2 L \ominus \cup L_{\oplus}$
Case 1
$\beta = u \ominus \ \propto \ \ominus v$ ,where $u, v \in V^{**}, \phi(\alpha)$ contains $\rho_i, u \ \$ \ v \in \rho'_i, \rho_i \in L\ominus$ and $| \ u \ |_c = | \ v \ |_c = | \ \alpha \ |_c$
Case 11
$\beta = u \oplus \alpha \oplus v$ , where $u, v \in V^{**}$, $\phi(\alpha)$ contains $\rho'_i, u \ \$ \ v \in \rho'_i, \rho'_i \in L_{\oplus}$ and $| \ u \ |_r = | \ v \ |_r = | \ \alpha \ |_r$

**Definition 2.3.** The language generated by an EACGC G is defined by

$$L(G) = \{\omega \in v^{**} / \alpha \Longrightarrow^*_{ex} \omega, \alpha \in B\}$$

We denote by EACC the family of languages generated by EACGC

**Definition 2.4.** An internal array contextual grammar with choice ( IACGC) is a 5 tuple

$$G = (V, B, L_\ominus, L_\oplus\ , \phi) where$$

*1) V is a finite non empty set of symbols called alphabet.*
*2)B is a finite subset of $V^{**}$ whose elements are called axioms.*
*3) $L\ominus$ is a finite set$\{\rho_1, \rho_2, ...\rho_n\}$ of arrays over $V^{**}$ \$ $V^{**}$ each $\rho_i$ having a bounded number of rows.*
*4) $L_\oplus$ is a finite set $\{\rho_1^1, \rho_2', ..., \rho_n'\}$ of arrays over $V^{**}$ \$ $V^{**}$ each $\rho_i'$ having a bounded number of columns.*
*5) $\phi$ is a function $\phi : V^{**} \rightarrow 2^{L_\ominus\ U\ L_\oplus}$*
*the derivation with respect to G is defined as follows.*
*For $\alpha, \beta, \in V^{**}$ ,we say that $\alpha$ deverives $\beta$, and we write $\alpha \Rightarrow_{in} \beta$ if*
*case 1*
$\alpha = \alpha_1 \ominus \alpha_2 \ominus \alpha_3, \beta = \alpha_1 \ominus u \ominus \alpha_2 \ominus v \ominus \alpha_3$
*where $\alpha_1, \alpha_2, \alpha_3, u, v \in V^{**}$,*
*$\phi(\alpha_2)$ contains $\rho_i, u\ \&\ v \in \rho_i, \rho_i \in L_\ominus$*
*and $\mid u \mid_c = \mid v \mid_c \mid = \mid \alpha \mid_c$*
*Case 2*
$\alpha = \alpha_1 \oplus \alpha_2 \oplus \alpha_3\ , \beta\ = \alpha_1 \oplus u \oplus \alpha_2 \oplus v \oplus \alpha_3$
*where $\alpha_1, \alpha_2, \alpha_3, u, v \in V^{**}$,*
*$\phi(\alpha_2)$ contains $\rho_i', u\ \$\ v \in \rho_i', \rho_i' \in L_\oplus$*
*and $\mid u \mid_r = \mid v \mid_r \mid = \mid \alpha \mid_r$*

**Definition 2.5.** The language generated by an IACGC G is defined by
$L(G) = \{W \in V^{**}/\alpha \Rightarrow_{in}^* W, \alpha \in B\}$
The family of languages that can be generated by IACGC is defined by IACC
If we do not use a choice function, then the respective families are denoted by EAC and IAC

**Definition 2.6.**

A total array contextual grammar is a 5 tuple G=$(V, B, L_\ominus, L_\oplus, \psi)$

where
1) V is a finite non empty set of symbols called alphabet.
2) B is a finite subset of V** whose elements are called axioms.
3) $L_{\ominus}$ is a finite set$\{\rho_1, \rho_2, ..., \rho_n\}$ of arrays over $V^{**}$ \$ $V^{**}$ each $\rho_i$ having a bounded number of rows.
4) $L_{\oplus}$ is a finite set$\{\rho'_1, \rho'_2 ..., \rho'_n\}$ of arrays over $V^{**}$ \$ $V^{**}$ each $\rho'_i$ having a bounded number of columns.
5) $\phi$ is a function $\phi : V^{**} \times V^{**} \times V^{**} \to 2^{L_{\ominus} \cup L_{\oplus}}$
The definition with respect to G is defined as follows
For $\alpha, \beta \in V^{**}$ we say that $\alpha$ derives $\beta$ and we write $\alpha \Rightarrow_{in} \beta$ if
Case 1
$\alpha = \alpha_1 \ominus \alpha_2 \ominus \alpha_3$
$\beta = \alpha_1 \ominus \cup \ominus \alpha_2 \ominus V \ominus \alpha_3$
Where $\alpha_1, \alpha_2, \alpha_3, u, v \in V^{**}, \phi(\alpha_2)$ contains $\rho_i, u\$v \in \rho_i$
$\rho_i \in L_{\ominus}$ and $\mid u \mid_c = \mid v \mid_c = \mid \alpha \mid_c$
Case 2
$\alpha = \alpha_1 \oplus \alpha_2 \oplus \alpha_3$
$\beta = \alpha_1 \oplus u \oplus \alpha_2 \oplus v \oplus \alpha_3$
Where $\alpha_1, \alpha_2, \alpha_3, u, v \in V^{**}, \phi(\alpha_2)$ contains $\rho'_i, u\$v \in \rho'_i$
$\rho'_i \in L_{\oplus}$ and $\mid u \mid_r = \mid v \mid_r = \mid \alpha \mid_r$

Note
If we ignore the selection then the total array contextual grammar coincide with the internal array contextual grammars

**Definition 2.7.** The language generated by a Total Array Contextual grammars is defined by $L(G) = \{W \in V^{**}/\alpha \Rightarrow^* W, \alpha \in B\}$
The family of languages that can be generated by total array contextual grammar with choice and without choice is denoted by TACC and TAC respectively

**Definition 2.8.** An external array contextual grammar with restricted selection ( REACG ) is a 5 tuple

$G = (V, B, L\ominus, L\oplus, R)$
Where$V, B, L\ominus, L_{\oplus}$ are defined in the definition of EACGC and IACGC grammars and R is one of the families $REG$ , CF,CS over $L_{\ominus} \cup L_{\oplus}$
For $\alpha, \beta \in V^{**}$ , we say that $\alpha$ derives $\beta$ through the rule $\rho_i$ and we write $\alpha \Rightarrow_{e_x}^{\rho_i} \beta$ if
Case 1
$\rho_i \in L_{\ominus}, \beta = u \ominus \alpha \ominus v$ where
$u\$v \in \rho_i, \alpha \in v^{**}$ and
$| u |_c = | v |_c = | \alpha |_c$
*Case 2*
$\rho_i' \in L_{\oplus}$ , $\beta = v \oplus \alpha \oplus v$ *where*
$u\$v \in \rho_i, \alpha \in v^{**}$ *and*
$| u |_r = | v |_r = | \alpha |_r$ *The language generated by REACG G is defined as*
$L(G) = \{x \in V^{**} /$ *there exist a sequence of steps*
$\omega \Rightarrow_{\rho_{i1}} \omega_1 \Rightarrow ... \Rightarrow_{\rho_{in}} \omega_n = x$
*and* $\rho_{i1}, \rho_{i2}, ..., \rho_{in} \in R$ *where* $\rho_{ij} \in R$ *and* $\rho_{ij} \in L_{\ominus} \cup L_{\oplus}$ *and* $\omega \in B\}$
*Similarly we can define restricted internal array contextual grammars . Their class is defined by RIACG . The family of languages can be denoted by RIAC. There are two restricted versions are defined without the choice function*
*If we include the choice function then the two families are denoted respectively by REACC and RIACC. The language generated by REAC'S and RIACC'S is defined as follows. We apply the contextual rules arbitarly according to the choice function and pick up the sequence of rules we have applied.If this sequence belongs to the control set R, then we take the generated array in to the language set*
*A REAC grammar is defined by* $REAC(REG), REAC(CF), REAC(CS)$ *if R belongs to the families Regular ,Context free and Context sensitive respectively*

# 3 Kernel P System ( KP Systems)

A KP System is a formal model that uses some well known features of existing P System and includes some new elements and more importantly, it offers a coherent view on integrating them in to the same formalism. The system was introduced by Marian Gheorghe etal. Here a broad range of strategies to ues the rule against the multiset of objects available in each compartments is provided. Now we will see the definition of compartments and KP System

**Definition 3.1.** *Given a finite set $A$ called alphabet of elements, called objects and a finite set $L$ , of elements called labels , a compartment is a tuple $C = (l, \omega_0, R^\sigma)$ where $l \in L$ is the label of the comparments $\omega_0$ is the intial multiset over $A$ and $R^\sigma$ denotes the DNA code of $C$ , which comprises the set of rules, denoted by $R$ , applied in this comparments and a regular expression $\sigma, overLab(R)$ the labels of the rule of $R$*

**Definition 3.2.** *A kernel P System of degree $n$ is a tuple*
$K\Pi = (A, L, \mu, C_1, C_2, ..., C_n, i_0)$
*where $A$ and $L$ are as in definition 3.1, the alphabet and the set of labels respectively; $\mu$ defines the membrane structure which is a graph $(V, E)$ , where $V$ are vertices,*
*$V \subseteq L$ ( the nodes are labels of these comparments ) , and $E$ edges, $C_1, C_2, ..., C_n$ are $n$ compartments of the system - the inner part of each compartment is called region , which is delimited by a membrane , the labels of the comparments are from $L$ and initial multiset are over $A$ . $i_0$ is the output compartments where the result is obtained .*

# 4 Contextual Array Kernel P Systems

In this section ,we define external and internal array contextual KP Systems and give some examples

**Definition 4.1.**

An external array contextual KP system is a construct

$$K\Pi = (V, T, \mu, M_1, M_2, ..., M_n, L_{\ominus}, L_{\oplus}, R_1.R_2, ..., R_n, i_0)$$

where V is a finite nonempty set of symbols called alphabet
$T \subseteq V$ is the terminal alphabet
$\mu$ is the membrane srtucture
$M_i$ is a finite subset of $V^{**}$ called axioms, representing the array initially present in the region i,$1 \leq i \leq n$
$L_{\ominus}$ is a finite set of languages $\{\rho_{\prime}, \rho_2, ..., \rho_r\}$ each $\rho_i$ having bounded number of rows
$L_{\oplus}$ is a finite set of languages
$\{\rho_1', \rho_2', ..., \rho_s'\}$
each $\rho_i'$ having bounded number of columns

$$\phi_i : V^{**} \rightarrow 2^{R_i}$$

This is written in the form

$$\phi_i(\alpha) = \{(\rho_j \$ \rho_k, tar)\}$$

or

$$\phi_i(\alpha) = \{(\rho_j \$ \rho_k, tar)\delta\}$$

$R_i$ are finite set of rules of the form $(\rho_j \$ \rho_k, tar)$ or $(\rho_j \$\ \rho_k, tar)\delta$ , where $\rho_j, \rho_k \in L_{\ominus}$ or $\rho_j, \rho_k \in L_{\oplus}$ and tar $\in \{here, in, out\}$ is a target indication specifying the region where the result of the rewriting should be placed in the next step.
$\sigma$ is an execution strategy in KP System
Derivation is defined as follows
For $\alpha, \beta \in V^{**}$, we say $\alpha$ derives $\beta$

that is $\alpha \Rightarrow_{ex} \beta$ if
CASE 1
$\beta = u \ominus \alpha \ominus v$ where $u, v \in v^{**}$ $\phi_i(\alpha)$ contains $(\rho_j \$ \rho_k, tar)$ where
$\rho_j, \rho_k \in L_\ominus$ and $| u |_c = | v |_c = | \alpha |_c$ $u \in \rho_j, v \in \rho_k$
CASE 2
$\beta = u \oplus \alpha \oplus v$ where $u, v \in v^{**}$
$\phi_i(\alpha)$ contains $(\rho'_j \$ \rho'_k, tar)$ where
$\rho'_j, \rho'_k \in L_{\oplus}$ and $| u |_r = | v |_r = | \alpha |_r$ $u \in \rho^1_j, v \in \rho'_k$
If a rule $(\rho_j \$ \rho_k, tar)\delta$ is used after getting $\beta$ from $\alpha$ , the membrane is dissolved. The membrane structure and array in $\Pi$ constitute the initial configuration of the system. We can pass from one configuration to another one by using evolution rules. This is done in parallel, all arrays from all membranes, which can be the subject of local evolution rules should evolve simultatneously.
After the evolution, depending on the choice mapping a context is added to the string if there are choice strings that are free at the moment when we check its applicability
A sequence of transition between configuration of a P System $\Pi$ is called a computation with respect to $\Pi$ . A computation is successfull if and only if it halt , that is there is no rule applicable to the choice array present in the last configuration.The result of a successful computation is a set of arrays, which are send out of the system during the computation or the set of arrays present in the skin membrane.
The family of languages generated by external array contextual KP System of degee n denoted by $EACKP_n$ ,where the degree of the system denotes the total number of membranes in the system.

**Definition 4.2.** *An internal array of contextual KP System is a construct*
$K\Pi = (V, T, \mu, M_1, M_2, ..., M_n, L_\ominus, L_{\oplus}, R_1, R_2, ..., R_n, \phi_1, \phi_2, ..., \phi_n, i_0)$
*Where $V$ is the finite non empty set of symbols called total alphabet $T \subseteq V$ is the terminal alphabet, $\mu$ is the membrane structure $M_i$ is a finite subset of $V^{**}$ called axioms, representing the arrays initially present in the region $i, 1 \leq i \leq n$ $L_\ominus$ is a finite set of languages $\{\rho_1, \rho_2, ..., \rho_r\}$ each $\rho_i$ having bounded number*

*of rows.* $L_{\oplus}$ *is a finite set of languages* $\{\rho'_1, \rho'_2, ..., \rho'_s\}$ *each* $\rho'_i$ *having bounded number of columns*

$\phi_1 : V^{**} \to 2^{R_i}$ *This is written in the form*

$\Phi_i(\alpha) = (\rho_j \ \$ \ \rho_k, \ tar)$ *or* $\Phi_i(\alpha) = (\rho_j \ \$ \ \rho_k, \ tar)\delta$

$R_i$ *are the rules of the form*

$(\rho_j \$ \rho_k, tar) or (\rho_j \$ \rho_k, tar)\delta$

*Where* $\rho_j, \rho_k \in L_{\ominus} or \rho_j, \rho_k \in L_{\oplus}$ *and* $tar \in \{here, out, in\}$

$\sigma$ *is an execution strategy in KP System*

*Derivation is defined as follows for* $\alpha, \beta \in V^{**}$, *we say that* $\alpha$ *derives* $\beta$ *that is* $\alpha \Rightarrow_{in} \beta$ *If*

***case1***

$\alpha = \alpha_1 \ominus \alpha_2 \ominus \alpha_3$

$\beta = \alpha_1 \ominus u \ominus \alpha_2 \ominus v \ominus \alpha_3$

*Where* $\alpha_1, \alpha_2, \alpha_3, u, v \in V^{**}$, $\phi_i(\alpha)$ *contains* $(\rho_j \$ \rho_k, tar)$ *where* $\rho_j, \rho_k \in L_{\ominus}$ *and* $| u |_c = | v |_c = | \alpha_1 |_c = | \alpha_2 |_c = | \alpha_3 |_c, u \in \rho_j, v \in \rho_k$

***case2***

$\alpha = \alpha_1 \oplus \alpha_2 \oplus \alpha_3$

$\beta = \alpha_1 \oplus u \oplus \alpha_2 \oplus v \oplus \alpha_3$

*Where* $\alpha_1, \alpha_2, \alpha_3, u, v \in V^{**}$, $\phi_i(\alpha)$ *contains* $(\rho'_j \$ \rho'_k, tar)$ *where* $\rho'_j, \rho'_k \in L_{\oplus}$ *and* $| u |_r = | v |_r = | \alpha_1 |_r = | \alpha_2 |_r = | \alpha_3 |_r, u \in \rho'_j, v \in \rho'_k$

*if* $\delta$ *is used membrane is dissolved after the application of the rule. Like external array contextual KP System ,the internal array contextual KP System also works provided , in the evolution rules the contexts are internal.*

*Here also the result of a successfull computation is a set of arrays, which are sent out of the systems during the computation, or a set of arrays present in the skin membrane.*

*The family of languages generated by the internal array contextual KP System of degree n,* $n \geq 1$ *is denoted by* $IACKPC_n$

**Definition 4.3.** *A total array contextual KP System is a construct*

$$K\Pi = (V, T, \mu, M_1, M_2, ..., M_n, L_{\ominus}, L_{\bigcirc\!\!\!\!|}, R_1, R_2, ..., R_n, \phi_1, \phi_2, ..., \phi_n, i_0)$$

*Where*
*1) V is the finite non empty set of symbols called the total alphabet*
*2)* $T \subseteq V$ *the terminal alphabet*
*3)*$\mu$ *is a membrane structure*
*4)* $M_i$ *is a finite subset of* $V^{**}$ *called axioms representing the array initially present in the region i,*$1 \leq i \leq n$
$L_{\ominus}$ *is a finite set of languages* $\{\rho_1, \rho_2, ..., \rho_r\}$ *each* $\rho_i$ *having bounded number of rows*
$L_{\bigcirc\!\!\!\!|}$ *is a finite set of languages* $\{\rho'_1, \rho'_2, ..., \rho'_s\}$ *each* $\rho'_s$ *having bounded number of columns*
$\phi : V^{**} \to 2^{R_i}$
*This is written in the form*
$\phi_i(\alpha) = (\rho_j \ \$ \ \rho_k, tar)$ *or*

$\phi_i(\alpha) = (\rho_j \ \$ \ \rho_k, tar)\delta$
$R_i$ *are rules of the form* $(\rho_j \ \$ \ \rho_k, tar)\delta$
*where* $\rho_j, \rho_k \in L_{\ominus}$ *or* $\rho_j, \rho_k \in L_{\bigcirc\!\!\!\!|}$
*and* $tar \in \{here, out, in\}$

***case1***
$\alpha = \alpha_1 \ominus \alpha_2 \ominus \alpha_3$
$\beta = \alpha_1 \ominus u \ominus \alpha_2 \ominus v \ominus \alpha_3$

*Where* $\alpha_1, \alpha_2, \alpha_3, u, v \in V^{**}$
$\phi_i(\alpha)$ *contains* $(\rho_j \ \$ \ \rho_k, tar)$ *where*
$\rho_j, \rho_k \in L_{\ominus}$ *and*

$| u |_c = | v |_c = | \alpha_1 |_c = | \alpha_2 |_c = | \alpha_3 |_c$
$u \in \rho_j, v \in \rho_k$
***case2***
$\alpha = \alpha_1 \varobar \alpha_2 \varobar \alpha_3$
$\beta = \alpha_1 \varobar u\ \alpha_2 \varobar v \varobar \alpha_3$
*Where* $\alpha_1, \alpha_2, \alpha_3, u, v \in v^{**}$
$\phi_i(\alpha)$ *contains* $(\rho'_j\ \$\ \rho'_k, tar)$ *where*
$\rho^1_j, \rho'_k \in L_{\bigoplus}$ *and*
$| u |_r = | v |_r = | \alpha_1 |_r = | \alpha_2 |_r = | \alpha_3 |_r$
$u \in \rho'_j, v \in \rho'_k$
*If* $\delta$ *is used the membrane is dissolved after the application of the rules*

**Definition 4.4.** *One sided contextual KP System of degree* $n$ *is a construct*
$K\Pi = (V, T, \mu, M_1, M_2, ..., M_n, R_1, R_2, ..., R_n, i_0)$
*where* $V, T, \mu, M_i$ *are the same as defined in the above definitions and*
$R_i, 1 \leq i \leq n$ *are finite set of rules*
$(u\$\Lambda, tar)$ *or* $((\Lambda\$v), tar)$ *where*
$x, u, v \in V^{**}$ *and* $tar \in \{here, in, out\}$

### *Contextual Array KP System with erased Context*

*In a contextual KP System ,we adjoin context,in an erased contextual KP System ,we erase contexts. The two operations can be considered together in a KP System, which is able both to adjoin and erase contexts*
*We give the definition of erased contextual KP System as follows*

**Definition 4.5.** *An external array contextual KP System of degree* $n$ *with erased context is a construct*
$K\Pi = (V, T, \mu, M_1, M_2, ..., M_n, L_\ominus, L_{\bigoplus}, R_1, R_2, ..., R_n, \phi_1, \phi_2, ..., \phi_n, v_1, v_2, ..., v_n)$
*where* $V, T, \mu, M_i, L_\ominus, L_{\bigoplus}, R_i$*and* $\phi_i, 1 \leq i \leq n$ *are the same as defined for contextual array KP System.*
*Here* $v_i : V^{**} \to 2^{R_i}$

*This is written in the form*

$v_i(\alpha) = (\rho_j \$ \rho_k, tar)$, *where*

$\rho_j, \rho_k \in L_\ominus$ *or* $\rho_j, \rho_k \in L_{\oplus}$ *and*

$tar \in \{here, in_j, out\}$

*Where* $\rho_i$*'s are set of arrays over* $V^{**}$

*Derivation for erased contexts are defined as follows.*

*For* $\alpha, \beta \in v^{**}$, *we say* $\alpha$ *derives* $\beta$

$\alpha \Rightarrow_{ex} \beta$ *if*

***case1***

$\alpha = u \ominus \beta \ominus v$ *where* $u, v \in v^{**}$

$v_i(\alpha)$ *contains* $(\rho_j \$ \rho_k, tar)$ *where*

$\rho_j, \rho_k \in L_\ominus$ *and* $| u |_c = | v |_c = | \alpha |_c, u \in \rho_j, v \in \rho_k$

***case2***

$\alpha = u \oplus \beta \oplus v$ *where* $u, v \in v^{**}$

$\phi_i(\alpha)$ *contains* $(\rho_j^1 \$ \rho_k^1, tar)$ *where*

$\rho_j^1, \rho_k^1 \in L_{\oplus}$ *and* $| u |_r = | v |_r = | \alpha |_r$ $u \in \rho_j^1, v \in \rho_k^1$

*Like external or internal array contextual KP System, the external array contextual KP System with erased contexts also works provided in evolution rules, the contexts are either adjoined or erased.*

*Here also the result of a successful computation is a set of arrays that are send out of the system during the computation or the arrays present in the skin membrane.The family of languages generated by external array contextual KP System with erased contexts of degree n is denoted by* $EACPEC_n$

# 5 Some languages generated in Array KP Systems

**Example 5.1.** *Consider*
$K\Pi = (V, T, \mu, M_1, M_2, L_\ominus, L_{\bigcirc\!\!\!|}, R_1, R_2, i_o)$
$V = \{X, a\}, L_\ominus = \{\rho_1\}, L_{\bigcirc\!\!\!|} = \{\rho_2\}, T = \{X, a\}, i_o = skin\ membrane$
$M_1 = \{\ \}, M_2 = \{X\}, \mu = [_0[_1]_1]_0$
$\rho_1 = \{a^n \backslash n \geq 1\}\ \rho_2 = \{a_n \backslash n \geq 1\}$
$R_1 = \{r_1 : (\rho_1\$\ \rho_1\ , here)$
$r_2 = (\rho_2\$\ \rho_2\ , here)$
$r_3 : (\Lambda\ \$\ \Lambda\ , out)\delta\}$
$\sigma_1 = (r_1\ r_2)^n\ r_3$

*The grammar generates arrays of dimension* $(2n+1) \times (2n+1)$ *having boundaries of a's,with X in the centre. This language is with REG control .This is written as* $REAC(REG)$

*For n =2 Arrays of dimension* $5 \times 5$*having boundaries of a*

$$\begin{array}{ccccc} a & a & a & a & a \\ a & a & a & a & a \\ a & a & X & a & a \\ a & a & a & a & a \\ a & a & a & a & a \end{array}$$

**Example 5.2.**

$$K\Pi = (V, T, \mu, M_0, M_1, M_2, L_\ominus, L_{\bigcirc\!\!\!|}, R_1, R_2, i_0)$$

$V = \{X, a, b\}$ $\quad M_0 = \{\}$
$T = \{X, a, b\}$ $\quad M_1 = \{X\}$
$L_\ominus = \{\rho_1, \rho_1'\}$
$L_{\bigcirc\!\!\!|} = \{\rho_2, \rho_2'\}$
$\mu = [_0[_1]_1]_0$

$i_0 = skin\ membrane$
$\rho_1 = \{a^n \backslash n \geq 1\}$
$\rho_2 = \{a_n \backslash n \geq 1\}$
$\rho_1' = \{b^n \backslash n \geq 1\}$
$\rho_2' = \{b_n \backslash n \geq 1\}$

$R_1 = \{\ r_1 : (\rho_1 \$ \rho_1 here)$
$r_2 : (\rho_2 \$ \rho_2 here)$
$r_3 : (\rho_1'\ \$\ \rho_1', here)$
$r_4 : (\rho_2'\ \$\ \rho_2', here\ )$
$r_5 : (\Lambda\ \$\ \Lambda\ , out\ )\delta\ \}$
$\sigma_1\ = (r_1 r_2)^n\ (r_3 r_4)^n\ r_5$

This grammar generates arrays with an equal number of boundaries of a's and b's with X in the centre. This language is with context free control.
It is written as REAC(CF)

**Example 5.3.**

$$K\Pi = (V, T, \mu, M_0, M_1, L_{\ominus}, L_{\bigcirc\!\!\!\!|}, R_1, i_0)$$

$V = \{X, a, b, c\}$
$T = \{X, a, b, c\}$
$\mu = [_0[_1]_1]_0$
$M_0 = \{\ \}, M_1 = \{\ X\}$
$i_0 = skin\ membrane$
$L_{\ominus} = \{\rho_1, \rho_2, \rho_3\}$
$L_{\bigcirc\!\!\!\!|} = \{\rho_1^1, \rho_2^1, \rho_3^1\}$
$\rho_1 = \{a^n \backslash n \geq 1\}$
$\rho_2 = \{b^n \backslash n \geq 1\}$
$\rho_3 = \{c^n \backslash n \geq 1\}$
$\rho_1^1 = \{a_n \backslash n \geq 1\}$
$\rho_2^1 = \{b_n \backslash n \geq 1\}$

$\rho_3^1 = \{c_n \backslash n \geq 1\}$
$i_0 = M_0$
$R_1 = \{r_1 : (\rho_1 \ \$ \ \rho_1 \ , here)$
$r_2 : (\rho_1^1 \ \$ \ \rho_1^1 \ , here \ )$
$r_3 : (\rho_2 \ \$ \ \rho_2 \ , here)$
$r_4 : (\rho_2^1 \ \$ \ \rho_2^1 \ , here \ )$
$r_5 : (\rho_3 \ \$ \ \rho_3 \ , here)$
$r_6 : (\rho_3^1 \ \$ \ \rho_3^1 \ , here \ )$
$r_7 : (\Lambda \ \$ \ \Lambda \ , out \ )\delta \ \}$
$\sigma_1 \ = (r_1 \ r_2)^n \ (r_3 \ r_4)^n \ (r_5 \ r_6)^n \ r_7$

*Here we get an array with equal number of rows and columns. This KP System generates array of dimension* $(6n + 1) \times (6n + 1)$ *having equal number of boundaries of a's,b's and c's with X in the centre.This language is with context sensitive control. 'a' squares are added such that they are interior to squares of b's and 'b' squares are added such that they are interior to squares of of c's and exterior to squares of a's.It is denoted as REAC(CS)*
<u>*Remarks*</u>
*Examples 5.1, 5.2 and 5.3 holds good for the proper inclusion*
$REAC(REG) \subset REAC(CF) \subset REA(CS)$'

/

**Example 5.4.**

$$K\Pi = (V, T, \mu, M_0, M_1, M_2, L_\ominus, L_{\bigcirc\!\!\!\!|}, R_1, R_2, i_0)$$

$V = \{X, a, b\}$
$T = \{X, a, b\}$
$\mu = [_0[_1[_2]_2]_1]_0$
$M_0 = \{ \ \}$
$M_1 = \{ \ \}$

$M_2 = \{\ X\ \}$
$\rho_1 = \{a^n \backslash n \geq 1\}$
$\rho_2 = \{b^n \backslash n \geq 1\}$
$\rho_1' = \{a_n \backslash n \geq 1\}$
$\rho_2' = \{b_n \backslash n \geq 1\}$
$i_0 = M_1$
$R_2 = \{r_1 : (\rho_1\ \$\ \rho_1\ , here)$
$r_2 : (\rho_1^1\ \$\ \rho_1^1\ , here\ )$
$r_3 : (\Lambda\ \$\ \Lambda\ , out)\delta\}$
$\sigma_2\ = (r_1\ r_2)^* r_3$
$R_1 = \{r_1 : (\rho_2\ \$\ \rho_2\ , here)$
$r_2 : (\rho_2^1\ \$\ \rho_2^1\ , here\ )$
$r_3 : (\Lambda\ \$\ \Lambda\ , out\ )\delta\}$
$\sigma_1\ = (r_1\ r_2)^* r_3$
*We get array of the form $a^\dagger\ b^\dagger$, with $X$ in the centre*

**Example 5.5.**

$$K\Pi = (V, T, \mu, M_1, M_2, L_\ominus, L_{\bigcirc\!\!\!\vert}, R_2, i_0\}$$

$V = \{X, .\}$ $T = \{X, .\}$ $M_1 = \{\ \}$ , $M_2 = B$
$\mu = [_0[_1]_1]_0$

$$\left\{ \begin{array}{ccccccc} . & . & . & . & . & . & X \\ . & . & . & . & . & . & X \\ . & . & . & X & X & X & X \\ . & . & . & X & . & . & . \\ X & X & X & X & . & . & . \end{array} \right\}$$

$L_\ominus = \{\rho_2\}$
$L_{\bigcirc\!\!\!\vert} = \{\rho_1\}$
$\rho_1 = \{X\ (.)_{2n}/n \geq 2\}$
$\rho_2 = \{(.)_{3n+3}X \mid n \geq 2\}$
$i_0 = M_0$

$R_1 = \{r_1 \ : (\Lambda \ \$ \ \rho_1, here)$
$r_2 \ : (\Lambda \ \$ \ \rho_2, \ here)$
$r_3 \ : \ (\Lambda \ \$ \ \Lambda \ , out \ ) \ \delta\}$
$\sigma_1 = \{(r_1^3 \ r_2^2)^m r_3 / m \geq 1\}$
*This grammar generates the staircase of the same prportion and we get the required result in the skin membrane*

**Example 5.6.** $K\Pi = (V, T, \mu, M_1, M_2, M_3, M_4, R_1, R_2, R_3, R_4, i_0)$
$V = \{a, \#\}$
$T = \{a\}$
$\mu = [_1[_2[_3[_4]_4]_3]_2]_1$

$$M_1 = \left\{ \begin{array}{ccc} a & & \\ a & a & a \end{array} \right\}$$

$M_2 = \{ \ \}, \quad M_3 = \{ \ \}, \quad M_4 = \{ \ \}$
$i_0 = M_4$
$R_1 = \{r_1 : \# \begin{smallmatrix}\# \\ a\end{smallmatrix} \rightarrow \# \begin{smallmatrix}a \\ a\end{smallmatrix}(in)\}$
$\sigma_1 = r_1^*$
$R_2 = \{r_1 : \begin{smallmatrix}a \\ \#\end{smallmatrix} \ \# \rightarrow \begin{smallmatrix}a \\ \#\end{smallmatrix} \ a \ (out)$
$r_2 : \begin{array}{c} \# \\ a \\ \# \end{array} \# \rightarrow \begin{array}{c} a \\ a \\ \# \end{array} \# \quad (in)\}$

$\sigma_2 = r_1^* r_2$

$R_3 = \{r_1 : \begin{array}{ccc} & \# & \\ \# & a & \# \end{array} \rightarrow \begin{array}{ccc} & \# & \\ \# & a & a \\ \# & & \end{array} \ (in)\}$
$\sigma_3 = r_1^*$

$R_4 = \{r_1 : \begin{array}{c} \# \\ a \\ \# \end{array} \# \rightarrow \begin{array}{c} \# \\ a \\ \# \end{array} a$
$r_2 : \begin{array}{ccc} & \# & \\ \# & a & \# \end{array} \rightarrow \begin{array}{ccc} & a & \\ \# & a & \# \end{array} \ \}$

$\sigma_4 = (r_1 r_2)^*$

*This grammar generates a square .For a rectangle we start with*

$$M_1 = \left\{ \begin{array}{ccc} a & & \\ a & & \\ a & a & a \end{array} \right\}$$

# 6 Some properties of array contextual KP System

Certain properties are defined for one dimensional contextual grammars in Marcus [4]. In this section we extend these properties and theorem to the arrays and discuss certain behaviour of some array contextual languages

**Definition 6.1.** *A Language* $L \subseteq V^{**}$ *has the ERBS property if there is a constant* $P$ *such that for each* $x \in L$ , $\mid x \mid_r > p$ *there is* $Y \in L$ *such that* $x = uyv$ *and* $0 <\mid uv \mid_r \leq p$

**Definition 6.2.** *A language* $L \subseteq V^{**}$ *has the ECBS property if there is a constant* $P$ *such that for each* $x \in L$ , $\mid x \mid_c > p$ *there is* $y \in L$ *such that* $x = u \oplus y \oplus v$ *and 0* $<\mid u \oplus v \mid_c \leq p$

**Definition 6.3.** *A language* $L \subseteq V^{**}$ *has the internal rows bounded step* $(IRBS)$ *property if there is a constant* $P$ *such that for each* $x \in L \mid x \mid_r > p$ *there is* $y \in L$ *such that*
$x = x_1 \ominus u \ominus x_2 \ominus v \ominus x_3$
$y = x_1 \ominus x_2 \ominus x_3$ *and*
$0 <\mid u \ominus v \mid_r \leq p$

**Definition 6.4.** *A language* $L \subseteq V^{**}$ *has the internal column bounded step* $(ICBS)$ *property if there is a constant* $P$ *such that for each* $x \in L \mid x \mid_c > p$ *there is* $y \in L$ *such that*
$x = x_1 \oplus u \oplus x_2 \oplus v \oplus x_3$
$y = x_1 \oplus x_2 \oplus x_3$ *and*
$0 < \mid \ u \oplus v \ \mid_c \leq \ p$

**Definition 6.5.** *A language* $L \subseteq V^{**}$ *has the bounded row increases* $(BRI)$ *property if there is a constant* $P$ *such that for each* $x \in L \mid x \mid_r > p$ *there is* $y \in L$ *with*
$-p \ < \mid x \mid_r \ - \mid y \mid_r \leq p$

**Definition 6.6.** *A language* $L \subseteq V^{**}$ *has the bounded coloumn increases* $(BCI)$ *property if there is a constant* $P$ *such that for each* $x \in L \mid x \mid_c > p$ *there is* $y \in L$ *with*
$-p \ < \mid x \mid_c \ - \mid y \mid_c \leq p$
*clearly if a language has either the ERBS or the IRBS property then it also has the BRI property. Similarly if a language has either the ECBS or ICBS property ,then it also has the BCI property. But the converse need not be true.*

**Theorem 6.1.** *If a language has BRI property then it need not have ERBS and IRBS property*

**Theorem 6.2.** *If a language has BCI property then it need not have ECBS and ICBS property*

*Proof.* We can prove theorem 1 and 2 with a counter example

The rectangular arrays of the form

$\{ \ a^n b^n c^n d^n e^n \backslash n \geq 1\}$
has both BRI and BCI property. But it does not have the ERBS, IRBS,ECBS and ICBS properties with the constant $n = 5$

□

**Theorem 6.3.** *A language is in the total contextual array family iff it has the IRBS and ICBS property.*

**Theorem 6.4.** $L = \{a^n b^n c^n d^n e^n / n \geq 1\} \notin TAC$
*Because it does not satisfy the IRBS and ICBS property*

**Example 6.1.** $K\Pi = (V, T\ , \mu, M_1, M_2, L_{\ominus}, L_{\oplus}, R_1, i_0)$
$V = \{X, a\}$
$T = \{X\}$
$\mu = [_0[_1]_1]_0$
$L_{\ominus} = \{\rho_1\}$
$L_{\oplus} = \{\rho_2\}$
$M_1 = \{\ \}$
$M_2 = \{X\}$
$\rho_1 = \{a^n \backslash n \geq 1\}$
$\rho_2 = \{a_n \backslash n \geq 1\}$

$R_1 = \{r_1 : (\rho_1\ \$\ \rho_1\ , here)$
$r_2 : (\rho_2\ \$\ \rho_2\ , here\ )$
$r_3 : (\Lambda\ \$\ \Lambda\ , out\ )\}$
$\sigma_2\ = ((r_1\ r_2)^2)^n\ r_3$

*We get square arrays of the form $a^{2^n}$ with $X$ in the centre in the skin membrane.*

**Theorem 6.5.** $L = \{a^{2^n} / n \geq 1\} \notin TAC$
*Since it does not have the BCI and BRI property*

**Example 6.2.**

$$K\pi = (V, T\, \mu, M_0, M_1, M_2, L_{\ominus}, L_{\oplus}, R_0, R_1, R_2, R_3, i_0)$$

$V = \{X, a, b\}$
$T = \{X, a, b\}$
$\mu = [_0[_1[_2[_3]_3]_2]_1]_0$
$M_0 = \{\ \}, M_1 = \{\ \}\ M_2 = \{\ \}$
$M_3 = \{\ X\}$
$L_{\ominus} = \{\rho_1, \rho_2\}$
$L_{\oplus} = \{\rho'_1, \rho'_2\}$
$\rho_1 = \{a^n \backslash n \geq 1\}$
$\rho_2 = \{b^n \backslash n \geq 1\}$
$\rho'_1 = \{a_n \backslash n \geq 1\}$
$\rho'_2 = \{b_n \backslash n \geq 1\}$
$R_3 = \{r_, : (\rho_1\ \$\ \rho_1, here)$
$r_2 : (\rho'_1\ \$\ \rho'_1, here)$
$r_3 : (\Lambda\ \$\ \Lambda, out, \delta)$
$\sigma_3 = r_1 r_2 r_3$
$R_2 = \{r_1 : (\rho_2\ \$\ \rho_2\ , here)$
$r_2 : (\rho'_2\ \$\ \rho'_2\ , here\ )$
$r_3 : (\Lambda\ \$\ \Lambda\ , out\ )\delta\}$
$\sigma_2\ = (r_1\ r_2)^* r_3$
$R_1 = \{r_1 : (\rho_1\ \$\ \rho_1\ , here)$
$r_2 : (\rho'_1\ \$\ \rho'_1\ , here\ )$
$r_3 : (\Lambda\ \$\ \Lambda\ , out\ )\delta\}$
$\sigma_1\ = r_1\ r_2\ r_3$
*We get array of the form $ab^\dagger a$ with $X$ in the centre*

**Theorem 6.6.** $L = \{ab^\dagger a\} \notin EACGC$
*Because it does not have the ERBS and ECBS property*

# 7 Correspondance between external array contextual KP System and external array contextual KP System in normal form

*Genaralised definition of normal form for rewriting P System were given in [10]*
*We extended the definitions in the case of array KP Systems*

**Definition 7.1.** *The depth of a KP System is equal to height of the tree describing its membrane structure*

**Definition 7.2.** *A skin cell in which there are some elememntary cells floating, then it is a normal form membrane structure*

**Definition 7.3.** *A membrane structre is said to be in the normal form if it is of depth 2*

**Definition 7.4.** *Two KP Systems are equivalent if they generate the same language*

**Definition 7.5.** *An array KP System is in m-n normal form if its depth is exactly m and in each membrane , we have exactly n rules. If we put no restriction either on the depth or on the number of rules then we replace the corresponding form with* $*$

**Theorem 7.1.** *For any external array contextual KP System of degree m, there exist an equivalent external array contextual KP System of degree m in* $2 - *$ *normal form*

*Proof.* Let us consider an external array cvontextual KP System of degree m as
$K\Pi = (V, T, \mu, M_0, M_1, M_2, ..., M_{m-1}, L_{\ominus}, L_{\bigoplus}, R_1, R_2, ...R_{m-1}, i_0)$

$\mu = [_0[_1[_2[_3...]_3]_2[_1]_0$

We construct an equivalent KP System in $2-*$ normal form as

$K\Pi_1 = (V, T, \mu, M_0, M_1, M_2, ..., M_{m-1}, L'_{\ominus}, L'_{\oplus}, R'_1, R'_2, R'_3, ...R'_{m-1}, i_0)$

$\mu = [_0[1]_1, [2]_2...[m-1]_{m-1}]_0$

$V_1 = V \ \cup \rho_1 \ \cup \ \rho_2 \ \cup \ ... \cup \rho_{m-1} \ \cup \ \rho'_1 \ \cup \ \rho'_2 \ \cup \ ... \cup \ \rho'_{m-1}$

Where $\rho_1 , \rho_2 , \rho_3 ..., \rho_{m-1}$ are rows of $a_, , a_2 , a_3 , ...a_{m-1}$ and $\rho'_1 , \rho'_2 , \rho'_3 , ..., \rho'_{m-1}$ are columns of $a_1 , a_2 , a_3 , ...a_{m-1}$ respectively.

$R'_1 = \{r_1 \ : (u \ \$ \ v \ , here \ )/(\rho_1 \ \$ \ \rho_1 \ , here \ )$

$r_2 : (u \ \$ \ v \ , here \ /(\rho'_1 \ \$ \ \rho'_1 \ ), here)$

$r_3 : (u \ \$ \ v \ , here \ )/(\Lambda \ \$ \ \Lambda \ , in_2 \ \}$

$\sigma_1 = r_1 r_2 r_3$

$R'_2 \ = \{r_1 \ : (u \ \$ \ v \ , here \ )/(\rho_2 \ \$ \ \rho_2 \ , here \ )$

$r_2 : (u \ \$ \ v \ , here) \ /(\rho'_2 \ \$ \ \rho'_2 \ ), here)$

$r_3 : (u \ \$ \ v \ , here \ )/((\Lambda \ \$ \ \Lambda), in_3 \ )\}$

$\sigma_2 = r_1 r_2 r_3$

Proceeding like this

$R'_{m-1} \ = \{r_1 \ : (u \ \$ \ v \ , here \ )/((\rho_{m-1} \ \$ \ \rho_{m-1}) \ , here \ )$

$r_2 \ : \ (u \ \$ v \ , here \ /((\rho'_{m-1} \ \$ \ \rho'_{m-1} \ ), here)$

$r_3 \ : (u \ \$ \ v \ , here \ )/((\Lambda \ \$ \ \Lambda \ , \text{out} \ ) \ \}$

$\sigma_{m-1} = r_1 r_2 r_3$

We can easily see that the arrays that reached the skin membrane $\Pi_1$ , are the same as those in $\Pi$

Hence $L(\Pi \ ) = L(\Pi_1 \ )$ Hence the theorems. □

# 8 Conclusion

*In this paper we have considered contextual way of processing array objects in KP Systems. The examples and the results from the previous section proved the power of array KP Systems. Also we have established the theorem that the external array contextual KP System and that in the $2^*$ normal form generate the same languages with the same degree*

## References


[1] *Ajitha P S and Desanamabika K S,Regulated Contextual KP Systems,International Jounrnal Of Mathematics Trends and Techonologies,Vol 34,Part 11 , June 2016*

[2] *Ajitha P S and Desanamabika K S,Contextual KP Systems with minimal/maximal use of selectors,International Jounrnal Of Computer Trends and Techonologies,Vol 28,issue 1, October 2015*

[3] *Ajitha P S and Desanamabika K S,Contextual KP Systems,International Jounrnal Of Mathematics Trends and Techonologies,Vol 17,issue 1 January 2015*

[4] *Gh. Paun, Marcus Contextual Grammars ,Kluwer Academic publishers, 1997*

[5] *Gheorghe Păun, Membrane Computing An Introduction, Springer–Verlag, Berlin, 2002.*

[6] *Madhu Mutyam, Kamala Krithivasan, Contextual P Systems, Fundamenta Informaticae, 2002*

[7] *Marian Gheorghe, Florentin Ipate, Ciprian Dragomir, Laurentiu Mierla, LuisValencia-Cabrera, Manuel Garcia-Quismondo,and Mario J, Kernel P System - Version 1, International Confrerence On Membrane Computing, 2013*

[8] *Rodica Ceterchi,Madhu Mutyam,Gheorghe Paun and K G Subramanian, Array- rewriting P Systems K luwer Academic Publishers, Printed in the Netherlands, 2003*

[9] *Kamala Krithivvasan, M Sakti Balan, Array Contextual Grammars, recent topics in mathematical and computer linguistics, PP154-168, Jan 2000*

[10] K.S Dersanambika and K. Krithivasan Contextual Array P Systems, International Journal of Computer Mathematics Vol 81 ,No 8 , August 2004, PP 955-969.

[11] Mutyam Madhu, Kamala Krithivasan,Generalised form for rewriting P Systems, ActaInformatica 38,PP721-734 Springer -Verlag 2002

[12] I.Peter , A normal form for P Systems, Bulletin of EATCS67, 165-172(1999)

[13] C.Zandron,C.Ferretti,G.Mauri. Two normal forms for rewriting P Systems, Proceedings of third international conference on machines,Computations and Universality, MCU2001, Chisinau, Moldova, Volume 2055 of LNCS,PP. 153-164.Berlin Heidelberg New York: Springer 2001