

Picture Generation of Rectangular Blocks using Tetrahedra

A. Dharani

*Research Scholar, Department of Mathematics,
Queen Mary's College, Chennai, India.
E-mail: dharaniclement@gmail.com*

R. Stella Maragatham

*Assistant Professor, Department of Mathematics
Queen Mary's College, Chennai, India.
E-mail: rstellamar@yahoo.com*

Rani Siromoney

*Adjunct Professor, Chennai Mathematical Institute, India.
E-mail: ranisiro@gmail.com*

Abstract

3D -Tile Pasting system is a new theoretical picture generative model to tile the three dimensional space. It allows picture generation by various production rules with usual techniques including vertex to edge and edge to vertex. In this paper we propose a new theoretical picture generative model namely CT-Picture Generative Grammar to generate a rectangular block by congruent tetrahedra. A generalised formula has been derived to identify the coordinates of the tetrahedra which tiles the rectangular block.

Keyword: congruent tetrahedra, cube, rectangular block

I. INTRODUCTION

In formal languages, a string is a sequence of symbols called alphabets. A formal language which is a set of strings can be formed in many ways using a grammar, automaton etc. A two dimensional string is called array or a two dimensional picture. Various theoretical picture generative grammars are introduced to generate two dimensional pictures using matrix grammars, rewriting rules array grammars etc. The extension of two dimensional picture to three dimension has been studied as well. Some of the works are mentioned here.

Three dimensional matrix models and array models are introduced by Siromoney et al in [9]. It is mentioned that, regular solids such as cube, tetrahedron, octahedron etc needs a more powerful tool compared to matrix models. Three dimensional array models are introduced with the advantage of generating different sizes of polyhedra by a single array grammar.

Then a different approach was made to generate cubes by hexagonal arrays using Hexagonal Kolam Array grammar [4]. For hexagon a new kind of catenation called arrow head catenation is introduced. A three dimensional object like a cube is depicted using hexagonal arrays where the blocks are arranged one over the other, step by step.

The generation of cubes of any size is also studied by Katsunobu Imai et al in [6] by constructing Uniquely Parsable array grammar with rewriting rules. To cope up the complex structure of rewriting rule they form a tool to design Isometric Array Grammar (IAG).

Researchers of number theory have been dealing with three dimensional objects and the dense packing over the years. In number theory, filling the three dimensional space with polyhedra such that no two interior points are in contact is called packing. A wide range of research work is being carried in finding out different possibilities of packing and its corresponding packing density.

In our previous work [2] we have introduced 3D-Tile pasting system and proved that grammatical formalism is possible for such packing. This system enables tiling of three dimensional space by polyhedra using different types of catenations like vertex to edge, edge to vertex, edge to face, face to edge, face to face, vertex to vertex.

In this paper, we have defined CT-Picture Generative Grammar which involves congruent tetrahedra in 3D Tile pasting System. We have derived a formula to find out the coordinates of the tetrahedra which is tiling the rectangular block. The packing of a rectangular block made of eight cubes is studied in detail.

II. PRELIMINARIES

Definition:

Sequential Space Filling Grammar (SSFG) [2]

A Sequential Space Filling Grammar is a 4-tuple (S, S_0, L, P) where S is the set of three dimensional polyhedra, $S_0 \in S$ is the initial polyhedron. L is the set of alphabets which represent vertices ' v ', edge positions ' e_p ', face positions ' f_p ' or any combination of these in a three dimensional polyhedra. An edge position ' e_p ' or a face position ' f_p ' is a position in which the neighbouring polyhedron is in contact.

For any two alphabets $\alpha, \beta \in L$, $P = \{\alpha \rightarrow \beta\}$ is the set of production rules. One polyhedron is catenated with another polyhedron depending on the production rules. Catenation of polyhedra is nothing but, ' α ' of one polyhedron coincides with ' β ' of the neighbouring polyhedron.

Depending on the representation of α and β , there are 9 types of production rules as follows:

- | | | | |
|--------------------------|---------------------------|----------------------------|-----------------------------|
| i) $v \rightarrow v$ | ii) $e_p \rightarrow e_p$ | iii) $f_p \rightarrow f_p$ | iv) $v \rightarrow e_p$ |
| v) $f_p \rightarrow e_p$ | vi) $v \rightarrow f_p$ | vii) $e_p \rightarrow v$ | viii) $e_p \rightarrow f_p$ |
| ix) $f_p \rightarrow v$ | | | |

where v is the vertex, e_p is the edge position and f_p is the face position.

When the production rules are applied, different kinds of pictures are generated. A picture is a finite collection of polyhedra formed by catenating the polyhedra from the set S in the grammar. To derive one picture from another we use the symbol \Rightarrow .

- If A and B are three dimensional pictures formed using a three dimensional polyhedra then $A \Rightarrow B$ indicates the usual meaning that picture A 'directly derives' picture B .
- If we have, for instance $P = \{\alpha \rightarrow \beta\}$ and $A \Rightarrow^1 B$,

then \Rightarrow^1 indicates the fact that A directly derives B using the production rule P once and $A \Rightarrow^{(1)^2} B$ means A derives B using the rule P twice.

- Suppose $P = \{\alpha_1 \rightarrow \beta_1, \alpha_2 \rightarrow \beta_2\}$ then $A \Rightarrow^{1,2} B$, means A derives B using either the first rule $\alpha_1 \rightarrow \beta_1$ or using the second rule $\alpha_2 \rightarrow \beta_2$.

And also $A \Rightarrow^{12} B$ means A derives B using the first rule $\alpha_1 \rightarrow \beta_1$ first and then the second rule $\alpha_2 \rightarrow \beta_2$ next to it.

The set of all possible pictures generated by SSFG is denoted by L^* -SSFG.

The language L-SSFG is a subset of L*-SSFG.

III. PICTURE GENERATION OF A RECTANGULAR BLOCK USING CONGRUENT TETRAHEDRA

In this section we introduce CT-Picture Generative Grammar (CT-PGG). CT stands for Congruent Tetrahedra. We have proved that a cube and a rectangular block can be generated using this grammar.

Definition

Sequential CT- Picture Generative Grammar

A Sequential CT- Picture Generative Grammar (CT-PGG) is $G=(C, C_0, V, P)$

where C is the set of congruent tetrahedra, C_0 is the initial congruent tetrahedron. V is the set of all labels. P is the set of production rules i.e, $P= \{ T_x(f_a) \rightarrow T_y(f_a) \}$ T_x and T_y are the names of the tetrahedra. f_a is the face of the tetrahedra. $T_x(f_a)$ represents the face f_a of tetrahedron T_x . $T_y(f_a)$ represents the face f_a of the tetrahedron T_y . The production rule $T_x(f_a) \rightarrow T_y(f_a)$ represents a face to face catenation where the face f_a of tetrahedra T_y gets catenated with the face f_a of tetrahedra T_x . The tetrahedra in the set C are labelled as such that the faces to be catenated are given the same name.

The set of all possible pictures generated by CT-PGG is denoted by $\mathcal{L}(CTPGG)$.

The language generated by CT-PGG is denoted by CTPL and it is a subset of $\mathcal{L}(CTPGG)$.

Theorem

A cube can be generated by six congruent tetrahedra.

Proof

Let the six tetrahedral tiles be T_1, T_2, T_3, T_4, T_5 and T_6 with the following vertices:

$$T_1 = (V_1, V_2, V_3, V_4)$$

$$T_2 = (V_1, V_2, V_4, V_5)$$

$$T_3 = (V_1, V_2, V_5, V_6)$$

$$T_4 = (V_1, V_2, V_6, V_7)$$

$$T_5 = (V_1, V_2, V_7, V_8)$$

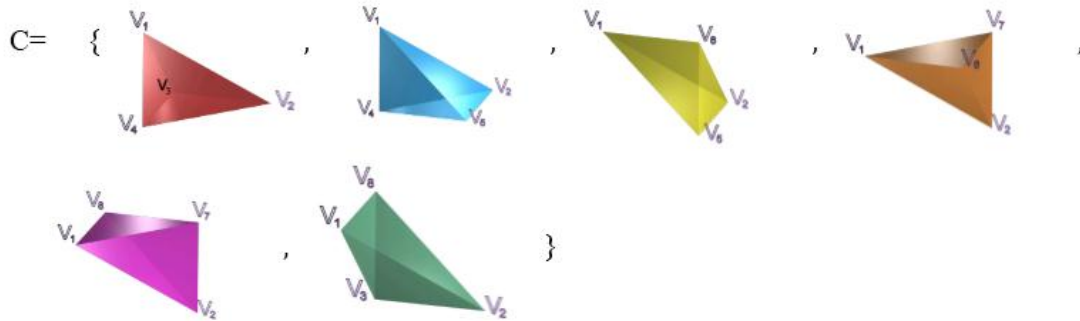
$$T_6 = (V_1, V_2, V_8, V_3)$$

where $V_1, V_2, V_3, V_4, V_5, V_6, V_7, V_8$ represents vertices of tetrahedra. The faces of the tetrahedra are labelled as given in the below table.

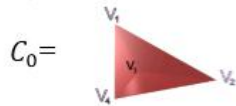
Table 1

Tetrahedra	Four vertices of tetrahedra	Vertices of the faces of tetrahedra	Labels of the faces
T_1	(V_1, V_2, V_3, V_4)	(V_1, V_2, V_3)	f_1
		(V_1, V_2, V_4)	f_2
		(V_1, V_3, V_4)	f_3
		(V_2, V_3, V_4)	f_4
T_2	(V_1, V_2, V_4, V_5)	(V_1, V_2, V_4)	f_2
		(V_1, V_2, V_5)	f_5
		(V_1, V_4, V_5)	f_6
		(V_2, V_4, V_5)	f_7
T_3	(V_1, V_2, V_5, V_6)	(V_1, V_2, V_5)	f_5
		(V_1, V_2, V_6)	f_8
		(V_1, V_5, V_6)	f_9
		(V_2, V_5, V_6)	f_{10}
T_4	(V_1, V_2, V_6, V_7)	(V_1, V_2, V_6)	f_8
		(V_1, V_2, V_7)	f_{11}
		(V_1, V_6, V_7)	f_{12}
		(V_2, V_6, V_7)	f_{13}
T_5	(V_1, V_2, V_7, V_8)	(V_1, V_2, V_7)	f_{11}
		(V_1, V_2, V_8)	f_{14}
		(V_1, V_7, V_8)	f_{15}
		(V_2, V_7, V_8)	f_{16}
T_6	(V_1, V_2, V_8, V_3)	(V_1, V_2, V_8)	f_{14}
		(V_1, V_2, V_3)	f_1
		(V_1, V_8, V_3)	f_{17}
		(V_2, V_8, V_3)	f_{18}

Consider a sequential CT- Picture Generative Grammar $G=(C, C_0, V, P)$
 Where C is the set of congruent tetrahedral



C_0 is the initial congruent tetrahedron



$V = \{T_1(f_1), T_1(f_2), T_1(f_3), T_1(f_4), T_2(f_2), T_2(f_5), T_2(f_6), T_2(f_7), T_3(f_5), T_3(f_8), T_3(f_9), T_3(f_{10}), T_4(f_8), T_4(f_{11}), T_4(f_{12}), T_4(f_{13}), T_5(f_{11}), T_5(f_{14}), T_5(f_{15}), T_5(f_{16}), T_6(f_{14}), T_6(f_1), T_6(f_{17}), T_6(f_{18})\}$

$P = \{T_1(f_2) \rightarrow T_2(f_2), T_2(f_5) \rightarrow T_3(f_5), T_3(f_8) \rightarrow T_4(f_8), T_4(f_{11}) \rightarrow T_5(f_{11}), T_5(f_{14}) \rightarrow T_6(f_{14}), T_6(f_1) \rightarrow T_1(f_1)\}$

The catenation of congruent tetrahedral tiles has been explained below.

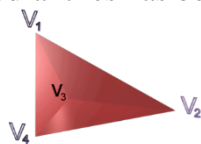


Figure 1: Tetrahedron T_1

Consider the initial tetrahedron $T_1 = (V_1, V_2, V_3, V_4)$. The four faces of T_1 are (V_1, V_3, V_4) , (V_2, V_3, V_4) , (V_1, V_2, V_4) and (V_1, V_3, V_2) .

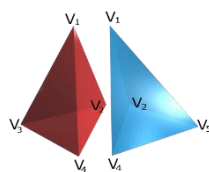


Figure 2: Catenation of Tetrahedron T_2

Consider the tetrahedron $T_2 = (V_1, V_2, V_4, V_5)$. The four faces of T_2 are (V_1, V_2, V_4) , (V_1, V_4, V_5) , (V_1, V_2, V_5) and (V_4, V_5, V_2) . The face (V_1, V_2, V_4) of T_1 and T_2 are catenated together.

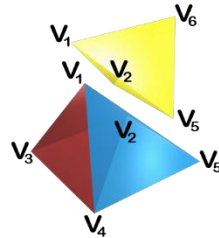


Figure 3: Catenation of Tetrahedron T_3

Consider the tetrahedron $T_3 = (V_1, V_2, V_5, V_6)$. The four faces of T_3 are (V_1, V_5, V_6) , (V_5, V_2, V_6) , (V_2, V_6, V_1) and (V_1, V_5, V_2) . The face (V_1, V_5, V_2) of T_2 and T_3 are catenated together.

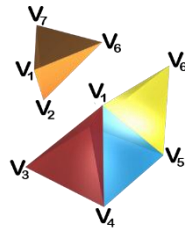


Figure 4: Catenation of Tetrahedron T_4

Consider the tetrahedron $T_4 = (V_1, V_2, V_6, V_7)$. The four faces of T_4 are (V_1, V_2, V_6) , (V_6, V_2, V_7) , (V_7, V_1, V_6) and (V_1, V_2, V_8) . The face (V_1, V_2, V_6) of T_3 and T_4 are catenated together.

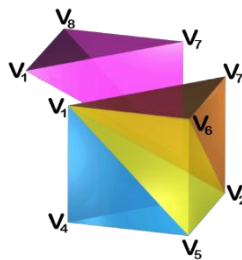


Figure 5: Catenation of Tetrahedron T_5

Consider the tetrahedron $T_5 = (V_1, V_2, V_7, V_8)$. The four faces of T_5 are (V_2, V_7, V_8) , (V_1, V_7, V_8) , (V_1, V_2, V_8) and (V_1, V_2, V_7) . The faces (V_1, V_6, V_7) of the tetrahedron T_4 and T_5 is catenated face to face.

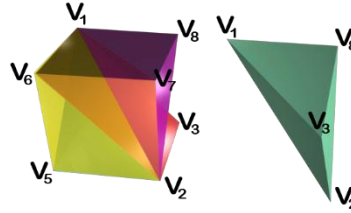


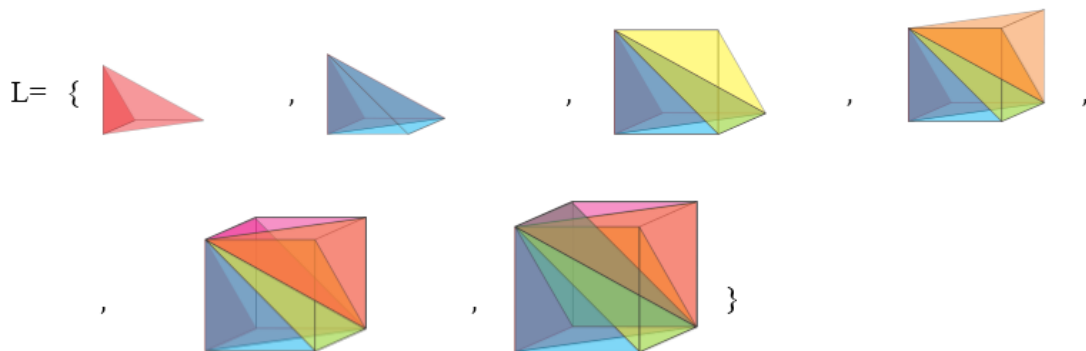
Figure 6: Catenation of Tetrahedron T_6

Consider the tetrahedron $T_6 = (V_1, V_3, V_2, V_8)$. The four faces of T_6 are (V_1, V_3, V_8) , (V_1, V_2, V_8) , (V_1, V_3, V_2) and (V_3, V_2, V_8) . The faces (V_1, V_2, V_8) of tetrahedra T_5 and T_6 are catenated face to face.



Figure 7: A cube formed by Six Congruent Tetrahedra

Hence we obtain a cube catenated by six congruent tetrahedra. The language is given by



Note: Apart from a cube, we can also generate a rectangular block as follows:

To begin with we form a cube using the previous grammar. Then catenate the face f_4 of the tetrahedron T_1 to the face f_{15} of the tetrahedron T_5 . The corresponding production rule is

$$Q = \{T_1(f_4) \rightarrow T_5(f_{15})\}$$

So as a result we get



Figure 8: A cube catenated with Tetrahedra T_1

As we continue applying the production rules in P, a rectangular block is generated.



Figure 9: Generation of rectangular block using congruent tetrahedral

IV. PICTURE GENERATION OF A RECTANGULAR BLOCK USING COORDINATE SYSTEM

Let us consider a unit cube with labels ($V_1, V_2, V_3, V_4, V_5, V_6, V_7, V_8$) in coordinate system by fixing the vertex V_3 as 111. Since it is a unit cube the other coordinates ($V_1, V_2, V_4, V_5, V_6, V_7, V_8$) are formed as (212, 121, 211, 221, 222, 122, 112).

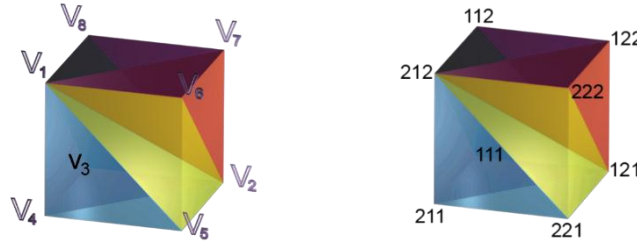


Figure 10: Labeling of a cube in coordinate system

When we apply the tiling process mentioned in the proof of the Theorem, in section 3, we get the following six tetrahedra

$$T_1 = (212, 121, 111, 211)$$

$$T_2 = (212, 121, 211, 221)$$

$$T_3 = (212, 121, 221, 222)$$

$$T_4 = (212, 121, 222, 122)$$

$$T_5 = (212, 121, 122, 112)$$

$$T_6 = (212, 121, 112, 111)$$

Now we try to extend the idea of tiling a cube by tetrahedra to a rectangular block of size (l, m, n) . A unit cube is of size $(2, 2, 2)$. Considering x, y, z coordinates as i, j, k respectively and fixing the value of i, j and k as 1, 1, 1 we obtain the following formula

Table 2

1. $[(i + 1)j(k + 1), i(j + 1)k, ijk, (i + 1)jk]$
2. $[(i + 1)j(k + 1), i(j + 1)k, (i + 1)jk, (i + 1)(j + 1)k]$
3. $[(i + 1)j(k + 1), i(j + 1)k, (i + 1)(j + 1)k, (i + 1)(j + 1)(k + 1)]$
4. $[(i + 1)j(k + 1), i(j + 1)k, (i + 1)(j + 1)(k + 1), i(j + 1)(k + 1)]$
5. $[(i + 1)j(k + 1), i(j + 1)k, i(j + 1)(k + 1), ij(k + 1)]$
6. $[(i + 1)j(k + 1), i(j + 1)k, ij(k + 1), ijk]$

Where $i = 1, 2, \dots, l - 1$

$j = 1, 2, \dots, m - 1$

$k = 1, 2, \dots, n - 1$

Now let us see how the above formula works for a cube of size (3,3,3).

Here $l = 3, m = 3$ and $n = 3$. So $i = 1,2$

$$j = 1,2$$

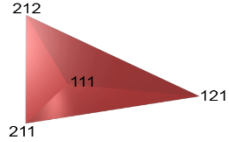
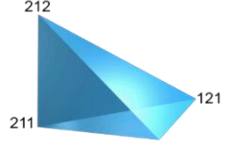
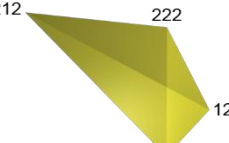
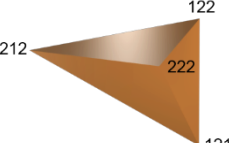
$$k = 1,2$$

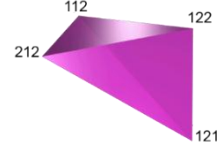
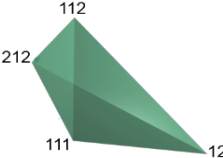
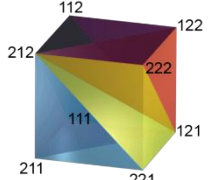
when we substitute the values of i, j and k we get eight possibilities (1,1,1), (1,2,1), (2,2,1), (2,1,1), (2,1,2), (1,1,2), (1,2,2), (2,2,2)

When we apply the above combinations one by one we obtain tetrahedral tiling the cubes of size(2,2,2). So for a cube of size(3,3,3), totally there are $8 \times 6=48$ tetrahedra which generates the cube. The following are the details of the coordinates of each tetrahedra which are tiling the cubes of size(2,2,2)

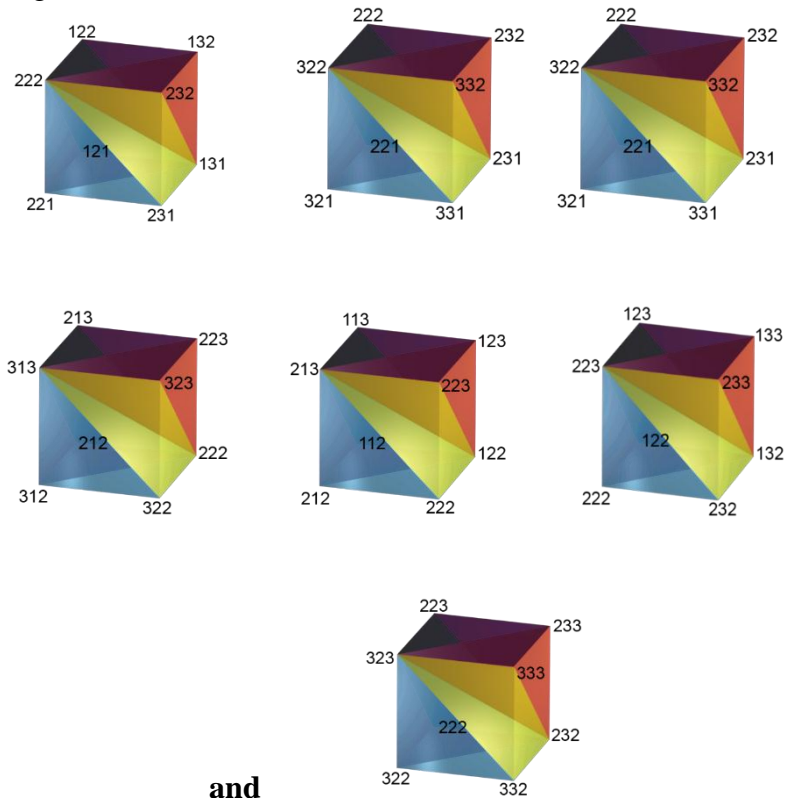
To begin with we consider the coordinate (1,1,1) and substitute it in the formula. So we get six congruent tetrahedral, which is named as $T_1, T_2, T_3, T_4, T_5, T_6$. Then we label each of the tetrahedra starting from a_1, a_2, \dots as such that the repeated coordinates of the faces are given the same label.

Table 3:

Values of i, j, k	Coordinates of six tetrahedra		Faces of the tetrahedra		
(1,1,1)	(212,121,111,211)		T_1	(212,121,111)	a_1
				(212,121,211)	a_2
				(212,111,211)	a_3
				(121,111,211)	a_4
	(212,121,211,221)		T_2	(212,121,211)	a_2
				(212,121,221)	a_5
				(212,211,221)	a_6
				(121,211,221)	a_7
	(212,121,221,222)		T_3	(212,121,221)	a_5
				(212,121,222)	a_8
				(212,221,222)	a_9
				(121,221,222)	a_{10}
(212,121,222,122)		T_4	(212,121,222)	a_8	
			(212,121,122)	a_{11}	
			(212,222,122)	a_{12}	
			(121,222,122)	a_{13}	

	(212,121,122,112)		T_5	(212,121,122)	a_{11}
				(212,121,112)	a_{14}
				(212,122,112)	a_{15}
				(121,122,112)	a_{16}
	(212,121,112,111)		T_6	(212,121,112)	a_{14}
				(212,121,111)	a_1
				(212,112,111)	a_{17}
				(121,112,111)	a_{18}
$P = \{T_1(a_2) \rightarrow T_2(a_2), T_2(a_5) \rightarrow T_3(a_5), T_3(a_8) \rightarrow T_4(a_8), T_4(a_{11}) \rightarrow T_5(a_{11}), T_5(a_{14}) \rightarrow T_6(a_{14}), T_6(a_1) \rightarrow T_1(a_1)\}$					

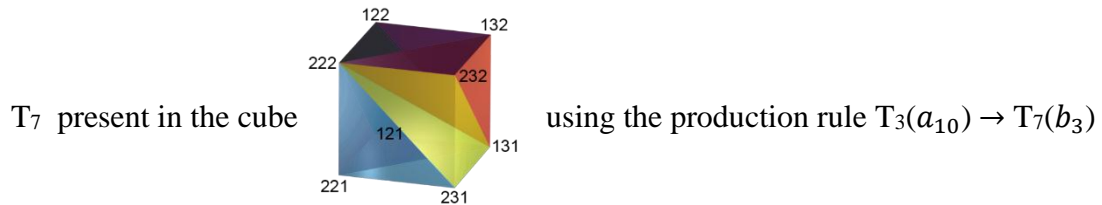
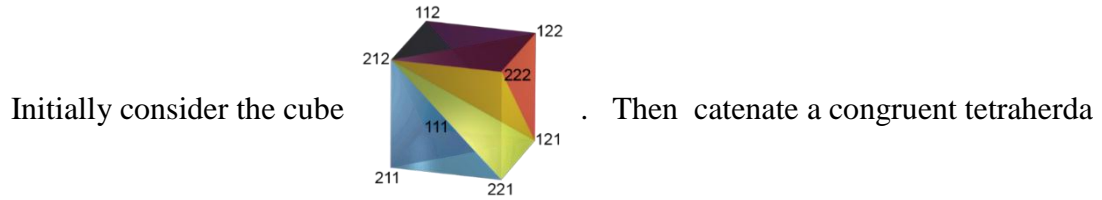
Similarly we consider the coordinates (1,2,1), (2,2,1), (2,1,1), (2,1,2), (1,1,2), (1,2,2), (2,2,2) and follow the procedure just as we had followed earlier for(1,1,1). As a result we get the cubes



and

Figure 11: Cubes generated using tables 2 and 3

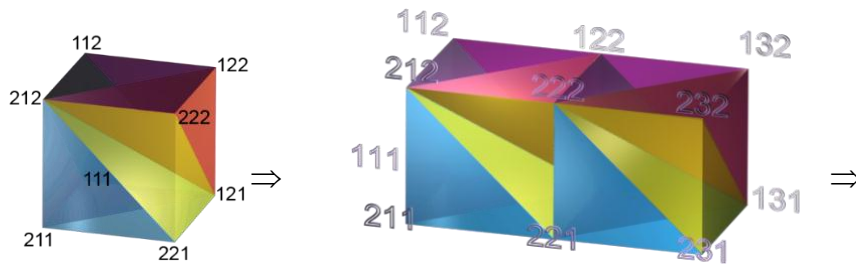
Considering all the eight unit cubes, we catenate its faces to form a cube of two units such that the congruent faces of the tetrahedra coincide.



and $T_4(a_{13}) \rightarrow T_{12}(b_{17})$. Similarly catenate the congruent faces of the tetrahedra using the production rules given below.

$$\begin{aligned}
 P = \{ & T_3(a_{10}) \rightarrow T_7(b_3), T_4(a_{13}) \rightarrow T_{12}(b_{17}), \\
 & T_9(b_9) \rightarrow T_{17}(c_{16}), T_8(b_6) \rightarrow T_{18}(c_{18}), \\
 & T_{18}(c_{17}) \rightarrow T_{22}(d_{13}), T_{13}(c_3) \rightarrow T_{21}(d_{10}), \\
 & T_{22}(d_{12}) \rightarrow T_{26}(e_7), T_{23}(d_{15}) \rightarrow T_{25}(e_4), \\
 & T_{29}(e_{16}) \rightarrow T_{33}(f_9), T_{30}(e_{18}) \rightarrow T_{32}(f_6), \\
 & T_{34}(f_{13}) \rightarrow T_{42}(g_{17}), T_{33}(f_{10}) \rightarrow T_{37}(g_3), \\
 & T_{39}(g_9) \rightarrow T_{47}(h_{16}), T_{38}(g_6) \rightarrow T_{48}(h_{18}), \\
 & T_{48}(h_{17}) \rightarrow T_{28}(e_{13}), T_{43}(h_3) \rightarrow T_{27}(e_{10}) \}
 \end{aligned}$$

Figurative explanation is given below.



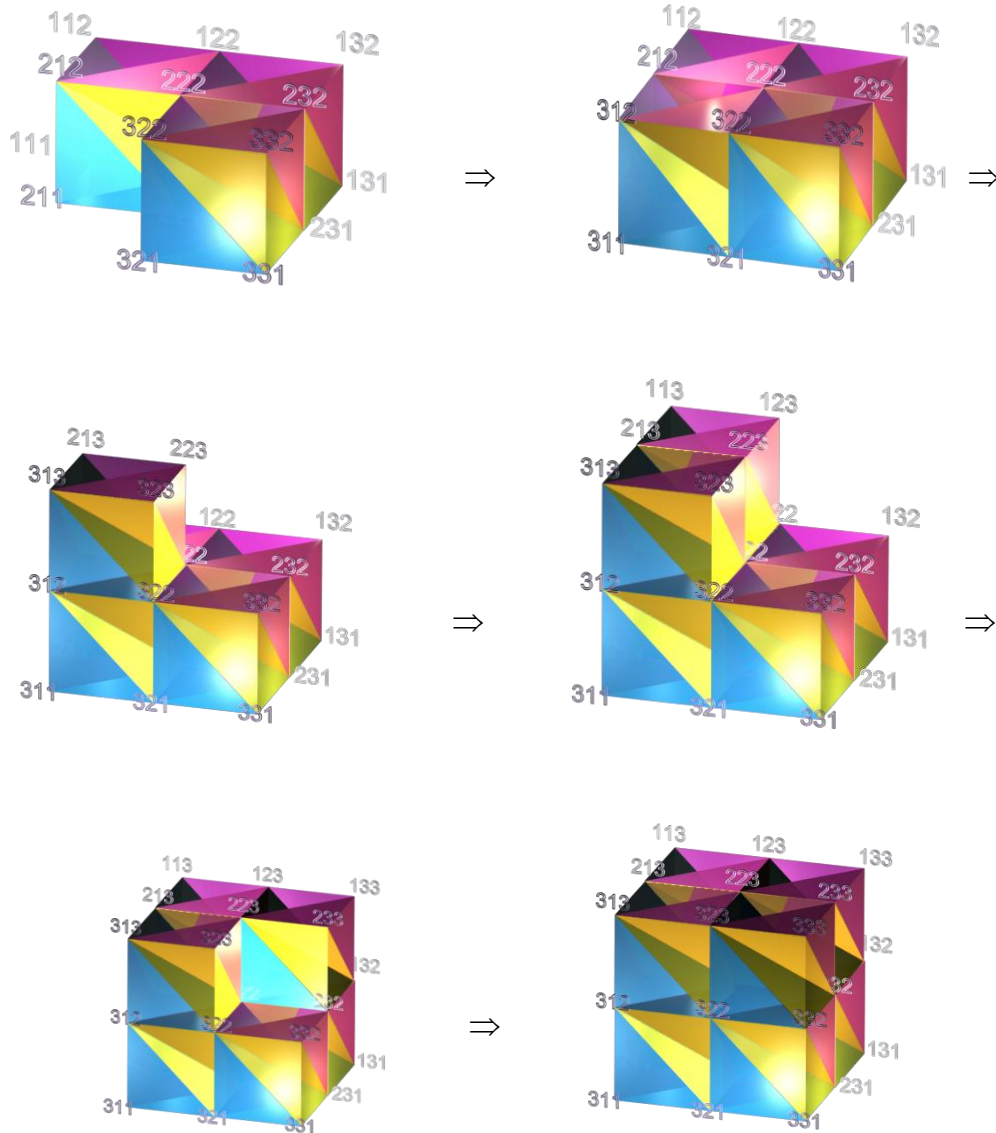


Figure 12: Generation of a cube of size two units using congruent tetrahedral

Each cube is catenated with another cube in a unique way in which the congruent faces of the tetrahedra are attached with each other. We can create cubes or rectangular blocks of any sizes following a similar procedure.

V. CONCLUSION

In this paper we have shown that grammatical formalism for the picture generation of rectangular blocks by congruent tetrahedra is possible. CT-Picture Generative Grammar has been defined. A formula has been introduced to obtain the coordinates of the tetrahedra using which we can generate rectangular blocks of any size.

REFERENCES

- [1] D.G.Thomas, F.Sweety,V.R.Dare,T.Kalyani,3D Rectangular Array Acceptors and Learning, Image Analysis from Theory of Applications, Research Publishing, Singapore,205-214,2008
- [2] A.Dharani, R.Stella Maragatham, R.Siromoney, Packing Regular Tetrahedra using 3D-Tile Pasting System, Communicated to International Journal of Pattern Recognition and Artificial Intelligence.
- [3] F.Sweety, D.G.Thomas, T.Kalyani, Recognizability of 3D Domino System, in Proceedings of Indian Conference on Intelligent System,265-271,2007
- [4] G.Siromoney, R.Siromoney, Hexagonal Arrays and Rectangular Blocks, Computer Graphics and Image Processing, 5, 353-381(1976)
- [5] Jefferey C.Lagarias and Chuanming Zong, Mysteries in Packing Regular Tetrahedra , Notices of the AMS, Dec 2012,1540-1549
- [6] Katsunobu Imai, Yukio Matsuda, Chuzo Iwamoto,and Kenichi Morita, A Three-Dimensional Uniquely Parsable Array Grammar that Generates and Parses Cubes, Electronic Notes in Theoretical Computer Science, 46, 2001,339-354.
- [7] De Floriani L., Lee M. (2004) Selective Refinement on Nested Tetrahedral Meshes. In: Brunnett G., Hamann B., Müller H., Linsen L. (eds) Geometric Modeling for Scientific Visualization. Mathematics and Visualization. Springer, 329-344, 2013.
- [8] R.Siromoney, G.Siromoney,Extended Controlled Table L-Arrays, Information and Control, 35, 1977, 119-138
- [9] R.Siromoney, Kamala Krithivasan, G.Siromoney, n-Dimensional Array Languages and Description of Crystal Symmetry-II, Proc. Ind. Acad. Scie 78A, 1973,130-139.

