International Journal of Pure and Applied Mathematics
Volume 117 No. 21 2017, 769-778
ISSN: 1311-8080 (printed version); ISSN: 1314-3395 (on-line version)
url: http://www.ijpam.eu
Special Issue

ijpam.eu

Hexagonal Online Tessellation Acceptor

[1]Pawan Kumar Patnaik, [2]Venkata Padmavati Metta and [3]Jyoti Singh

[1]Department of Computer Science and Engineering,
Bhilai Institute of Technology, Durg, India.
pawanpatnaik37@gmail.com

[2]Department of Computer Science and Engineering,
Bhilai Institute of Technology, Durg, India

[3]Chhattisgarh Professional Examination Board,
Raipur, India.

Abstract

The purpose of this research is to propose a new type acceptor called the "Hexagonal Online Tessellation acceptor" and in this paper based on the ideas due to Inoue and A. Nakamura [5] an application of the two dimensional tessellation acceptor is presented for very rapid online detection of hexagonal pattern in a text. This paper includes the proposed concept of hexagonal online tessellation automata recognize hexagonal picture languages.

Key Words: Finite automata, hexagonal array, hexagonal picture languages, online acceptor, tessellation automata.

1. Introduction

In recent years there has been unceasing interest in two and more dimensional pattern matching problems. Such interest is substantiated by the growing computational strength of our computers allowing multi-dimensional data.

Picture languages generated by grammars or recognized by automata have been introduced since the seventies for problems arising in the framework of pattern recognize and image analysis [3, 4, 8, 9]. Hexagonal patterns are known to occur in the literature on picture processing and scene analysis. Siromoney *et al* [10, 11] constructed grammars for generating hexagonal arrays and hexagonal patterns.

Recently Dersanambika et al [2] have introduced two interesting classes of hexagonal picture languages, viz., local hexagonal picture languages and recognizable hexagonal picture languages and studied their properties. In this paper we develop a recognizing device called hexagonal online tessellation automata to recognize these languages and provide examples.

This paper is organized as follows. Section II describes hexagonal array, hexagonal online tessellation acceptor and notions of hexagonal online tessellation acceptor. Section III deals with the design of the array matching problem for hexagonal on-line tessellation acceptor (h-dota). Finally, the conclusions have been drawn in section IV.

2. Preliminaries

In this section, we review some notions of hexagonal pictures and hexagonal array introduced in [2]. Let $\sum$ be a finite alphabet of symbols. A hexagonal picture p over $\sum$ is a hexagonal array of symbols of $\sum$. The set of all hexagonal arrays of the alphabet $\sum$ is denoted by $\sum^{***H}$

Example. 1 A hexagonal picture over the alphabet {a, b, c, d} is shown in Fig. 1

a a a
a b d c
b c c a c
a b d a
a a d

Fig. 1: Hexagonal Picture

The set of all hexagonal arrays over the alphabet $\sum$ is denoted by $\sum^{**H.}$ With respect to triad of triangular axes, x, y, z, the co-ordinates of each element of the hexagonal picture in Fig. 1 are shown in Fig. 2

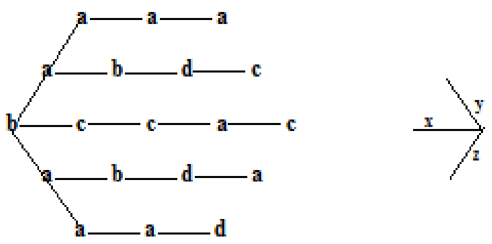


Fig. 2

Let p be a hexagonal picture in $\sum^{**H}$; p(i, j, k) is the symbol in the position (i, j, k); we denote the number of elements in the x, y, z directions of p by x(p), y(p) and z(p) respectively. The triplet (x(p), y(p), z(p)) is called the size of the hexagonal picture p. The size of the empty picture € is obviously (0, 0, 0)

Definition 2.1:

If x Є $\sum^{**H}$, then x is the hexagonal array obtained by surrounded by * is shown in Fig. 3

```
        *   *   *   *
      *   a   a   a   *
    *   a   b   d   c   *
  *   b   c   c   a   c   *
    *   a   b   d   a   *
      *   a   a   d   *
        *   *   *   *
```

Fig. 3: A Hexagonal Array Surrounded by *

Definition 2.2: A Hexagonal online tessellation acceptor (h-ota) is a 7- tuple

M = (K, E^3, $\sum$, U{*}, δ, q_e, q_0, q_0, F) where,

1. K is a finite set of states;
2. E^3 is a set of all 3 tuples of integers;
3. $\sum$ is a finite set of input symbols and (*) is the boundary symbol not in $\sum$.
4. $\delta : K^3 \times (\sum \cup \{*\}) \rightarrow 2^{K'} \cup \{\{q_0\}\}$ where, (K' = K – {q_e , q_0 , q_0} is the cell state transition function.
5. q_eЄ K is the motive state;
6. q_0Є K is the quiescent state;
7. F $\subseteq$ K – {q_e, q_0, q_0} is a set of final states.

The cell transition function δ prescribes state transition of cells in $E^{(3)}$. M is called deterministic if the image under δ of every element in $K^3 \times (\sum \cup \{*\})$ is a singleton; otherwise non-deterministic.

Example 2.

Let M = ({q_e , q_0 , q_1 , q_2 }, E^3 , {a,*}, δ, q_e , q_0 , q_0 F = {q_3}) be deterministic H-ota. where:

$\delta(q_e, q_0, q_0, q_0, a) = \delta(q_0, q_0, q_0, q_1, a) = \delta(q_0, q_0, q_1, q_0, a) = \delta(q_0, q_2, q_2, q_2, a) = \delta(q_0, q_2, q_0, q_1, a) = \delta(q_0, q_2, q_1, q_0, a) = q_1$And $\delta(q_0, q_1, q_1, q_1, a) = \delta(q_0, q_1, q_0, q_2, a) = \delta(q_0, q_1, q_2, q_0, a) = q_2$

when the pattern h over {a} as shown in the below figure. is presented to M, M enters the state configurations as shown in Fig.5(a), …, Fig. 5(h) respectively at time t=0,… t=7.

The run of M on h is the tape corresponding to the highlighted area enclosed by the heavy solid line. The halting state of the machine is shown in Fig. 5(h). The final state q_2 in the cell (2,4) indicates that M accepts the inputh on tape x.

*	*	*	*	*	*	*	*	*	*
*	*	a		a		a		*	*
*	a		a		a		a		*
*	*	a		a		a		*	*
*	*	*	*	*	*	*	*	*	*

Fig. 4: 3X4 tape x over {a}

q_0	q_0	q_0	q_0	q_0	q_0	q_0	q_0	q_0	q_0
q_0	q_0	q_0		q_0		q_0		q_0	q_0
q_0	q_e		q_0		q_0		q_0		q_0
q_0	q_0	q_0		q_0		q_0		q_0	q_0
q_0	q_0	q_0	q_0	q_0	q_0	q_0	q_0	q_0	q_0

(a) t = 0

q_0	q_0	q_0	q_0	q_0	q_0	q_0	q_0	q_0	q_0
q_0	q_0	q_0		q_0		q_0		q_0	q_0
q_0	q_1		q_0		q_0		q_0		q_0
q_0	q_0	q_0		q_0		q_0		q_0	q_0
q_0	q_0	q_0	q_0	q_0	q_0	q_0	q_0	q_0	q_0

(b) t = 1

q_0	q_0	q_0	q_0	q_0	q_0	q_0	q_0	q_0	q_0
q_0	q_0	q_1		q_0		q_0		q_0	q_0
q_0	q_1		q_0		q_0		q_0		q_0
q_0	q_0	q_1		q_0		q_0		q_0	q_0
q_0	q_0	q_0	q_0	q_0	q_0	q_0	q_0	q_0	q_0

(c) t = 2

q_0	q_0	q_0	q_0	q_0	q_0	q_0	q_0	q_0	q_0
q_0	q_0	q_1		q_0		q_0		q_0	q_0
q_0	q_1		q_2		q_0		q_0		q_0
q_0	q_0	q_1		q_0		q_0		q_0	q_0
q_0	q_0	q_0	q_0	q_0	q_0	q_0	q_0	q_0	q_0

(d) t = 3

q_0	q_0	q_0	q_0	q_0	q_0	q_0	q_0	q_0	q_0
q_0	q_0	q_1		q_2		q_0		q_0	q_0
q_0	q_1		q_2		q_0		q_0		q_0
q_0	q_0	q_1		q_2		q_0		q_0	q_0
q_0	q_0	q_0	q_0	q_0	q_0	q_0	q_0	q_0	q_0

(e) t = 4

q_0	q_0	q_0	q_0	q_0	q_0	q_0	q_0	q_0	q_0
q_0	q_0	q_1		q_2		q_0		q_0	q_0
q_0	q_1		q_2		q_1		q_0		q_0
q_0	q_0	q_1		q_2		q_0		q_0	q_0
q_0	q_0	q_0	q_0	q_0	q_0	q_0	q_0	q_0	q_0

(f) t =5

q_0	q_0	q_0	q_0	q_0	q_0	q_0	q_0	q_0	q_0
q_0	q_0	q_1		q_2		$\mathbf{q_1}$		q_0	q_0
q_0	q_1		q_2		q_1		q_0		q_0
q_0	q_0	q_1		q_2		$\mathbf{q_1}$		q_0	q_0
q_0	q_0	q_0	q_0	q_0	q_0	q_0	q_0	q_0	q_0

(g) t = 6

q_0	q_0	q_0	q_0	q_0	q_0	q_0	q_0	q_0	q_0
q_0	q_0	q_1		q_2		q_1		q_0	q_0
q_0	q_1		q_2		q_1		$\mathbf{q_2}$		q_0
q_0	q_0	q_1		q_2		q_1		q_0	q_0
q_0	q_0	q_0	q_0	q_0	q_0	q_0	q_0	q_0	q_0

(h) t = 7

Fig. 5: State Configurations of M (when the tape x as shown in Fig. 4 is presented to M)

The transition functions at different times (t=0,1,2…) for the above figures can be stated as follows:

At t = 1:

$\delta\ (q_e , q_0 , q_0 , \ q_0 , a) = q_1$

At t = 2:

$\delta\ (q_0 , q_0 , q_0 , \ q_1 , a) = q_1 \delta\ (q_0 , q_2 , q_0 , \ q_1 , a) = q_1$

$\delta\ (q_0 , q_0 , q_1 , \ q_0 , a) = q_1 \delta\ (q_0 , q_2 , q_1 , \ q_0 , a) = q_1$

At t = 3:

$\delta\ (q_0 , q_1 , q_1 , \ q_1 , a) = q_2$

At t = 4:

$\delta\ (q_0 , q_1 , q_0 , \ q_2 , a) = q_2 \delta\ (q_0 , q_1 , q_1 , \ q_1 , a) = q_2$

$\delta\ (q_0 , q_1 , q_2 , \ q_0 , a) = q_2$ **(Final State)**

At t = 5:

$\delta\ (q_0 , q_2 , q_2 , \ q_2 , a) = q_1$

At t = 6:

At t = 7:

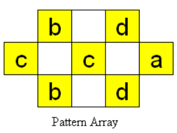
3. Design of Hexagonal (H-ota) as Array Matching

This section describes a procedure for designing a H-ota which solves the array matching problem for the set consisting of only one pattern array.

Definition 3.1:

Let $\sum$ be a finite alphabet. A hexagonal array over $\sum$ is a hexagonal pattern of elements of $\sum$ with the right edge adjusted as shown in Fig. 6. Let $\sum^3$ denote the set of all arrays over $\sum$ and for each y in $\sum^3$, let r(y) denote the number of rows of y.

For each hexagonal array, x in $\sum^3$ and each (i, j, k): $1 \leq i \leq r(x)$, $1 \leq j \leq c(x)$, where c(x) denotes columns of x.



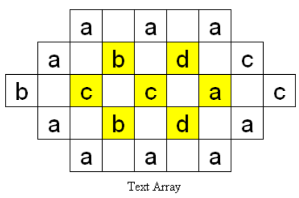
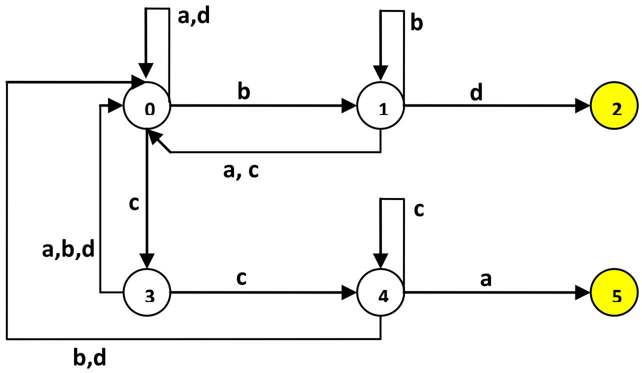


Fig. 6 Fig.7

Construction of Finite Automata for Pattern Array

A Pattern Matching Machine (PMM) has been developed as a useful device for solving the string matching problem. A PMM for pattern array is a machine which takes as input the text string in which keywords of pattern array appear as substrings. The PMM consists of a set of states. Each state is represented by a number. The machine processes the text string by successively reading the symbols in text string, making state transitions and occasionally emitting output. The behaviour of the PMM is dictated by two functions: a goto function g, a failure function f.



(A) Goto FUNCTION of M

Node No.	0	1	2	3	4	5
Failure Function f	-	0	0	0	3	0

(B) Failure Function of M

Fig. 8

A Pattern matching for a finite automaton whose next move function is outputted when algorithm 1 described below is applied to the goto function g and failure function f.

Algorithm1: Construction of the next move function δ.

Input: Goto Function g and failure function f for W

Output: Next move function δ
Method:
Begin
queue ← *empty*
For all a Є ∑ **do**
  begin
  δ (0,a) := g(0,a)
  If g(0,a) ≠ 0 **then** queue ← q(0,a)
  end;
  whilequeue ≠ empty **do**
    begin

r ← queue;

forall a Є ∑ **do**
begin
s := g(r,a)
if s ≠ fail **then**
begin
queue ← s; δ (r,a) := s
end
else δ (r,a) := δ (f(r),a)
end
end
end.

3.1. Construction of the Desired Hexagonal Online Tessellation automata (H-ota)

To design a H-ota M_y which solves the array matching problem for the set consisting of only one pattern array y over Σ have been proposed. The H-ota M_y obtained by using algorithm 2 .Ley 'y' be a given pattern array over an alphabet ∑ then, We design an H-ota M_y which acts as follows: When a text array x in $\sum^3$ is presented to M_y each of the (i, j, k) cell of M_y: x(i, j, k) ~ y enters an accepting state.

Algorithm 2: Construction of h-ota M.

Input: Finite automata with failure function and next move function

Output: The H-ota M = (K, E^3, ∑ U {*}, δ, q_e, q_0, q_0, q_0, F)

Method:

1. Let K = N(δ) U { q_e, q_0, q_0 }
2. F ⊆ K – { q_e, q_0, q_0 }
3. For each a Є ∑ and each [p_1, p_2 … p_l]
[q_1, q_2, …q_l] Є K – { q_e, q_0, q_0}

δ (q_e, q_0, q_0, q_0, a) = [δ_1 (q_{e01}, q_{001}, q_{001}, q_{001}, a), δ_2(q_{e02}, q_{002}, q_{002}, q_{002}, a) … δ_l(q_{e0l}, q_{00l}, q_{00l}, q_{00l}, a)]

δ (q_0[p_1, p_2 … p_l], q_0, q_0, a) = [δ_1(q_{001},p_1, q_{001}, q_{001}, a), δ_2(q_{002},p_2, q_{002}, q_{002}, a)] … δ_l(q_{00l}, p_l, q_{00l}, q_{00l}, a)]

δ (q_0, q_0,q_0[q_1, q_2 …q_l], a) = [δ_1(q_{001},q_{001}, q_{001}, q_1, a), δ_2(q_{002},q_{002}, q_{002}, q_2, a)] … δ_l(q_{00l},q_{00l}, q_{00l}, q_l, a)

And

δ (q_0,q_0[p_1, p_2…p_l] [q_1, q_2 … q_l], a) = δ_1(q_{001}, q_{001}, p1, q_1, a), δ_2(q_{002}, q_{002}, p_2, q_2, a) … δ_l(q_{00l}, q_{00l}, p_l, q_l, a)

Example 3: Let y be the pattern array over $\sum = \{a, b, c, d\}$ as shown in fig.6. The H-otaM_y obtained.

$M_y = (K, E^3, \sum, U \{*\}, \delta, q_e, q_0, q_0, q_0, F)$

where,

(1). $K = \{0,1,2,3,4,5\}$ U $\{q_e, q_0, q_0\}$

(2). $F \subseteq K - \{ q_e, q_0, q_0 \}$

$\delta (q_e, q_0, q_0, q_0 , b) = [1, 0, 0]$

$\delta (q_0, q_0, [1,0,0], q_0 , a) = [0, 0, 0]$

$\delta (q_0, q_0, [0,0,0], q_0 , a) = [0, 0, 0]$

$\delta (q_0, [0,0,0], [0,0,0], [3,0,0] , b) = [3, 1, 0]$

$\delta (q_0, [3,0,0], [3,1,0], [3,0,1] , c) = [4, 1, 1]$

$\delta (q_0, [3,1,0], [4,1,1], q_0 , d) = [3, 2, 1]$

$\delta (q_0, [3,1,2], [0,0,0], [5,2,2] , c) = [4,2,1]$

$\delta (q_0, [5,2,2], [4,2,1], [4,1,2] , c) = [0, 0, 0]$

$\delta (q_0, q_0, q_0, [1,0,0] , a) = [0, 0, 0]$

$\delta (q_0, q_0, q_0, [0,0,0] , a) = [0, 0, 0]$

$\delta (q_0, [1,0,0], [0,0,0], [0,0,0] , c) = [3, 0, 0]$

$\delta (q_0, [0,0,0], [3,0,0], q_0 , b) = [3, 0, 1]$

$\delta (q_0, [3,0,1], [0,0,0], [4,1,1], d) = [3,1,2]$

$\delta (q_0, [4,1,1], [3,1,2], [3,2,1] , d) = [5, 2, 2]$

$\delta (q_0, [3,2,1], [5,2,2], [0,0,0] , a) = [4, 1, 2]$

Let x be the text array as shown in fig 7. Fig.9 shows the states of M_y. Note that the cell [3,7] has entered an accepting state.

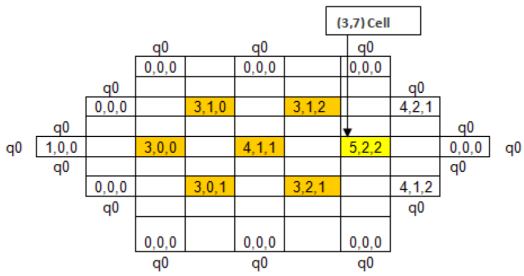


Fig. 9: State Configuration of H-ota My After My has Read the Text

4. Conclusion

It has been concluded that the array matching problem can be efficiently solved by using a h-ota. This research was motivated from the concepts of online tessellation acceptor and two- dimensional pattern matching by two dimensional online tessellation acceptors [12]. Using this concept we defined hexagonal tessellation acceptors. Syntactic methods play an important role in pattern generation and detection. One of the significant work of the research is the recognition of three-dimensional pattern using hexagonal online tessellation acceptor. It has applications in pattern matching and objects recognition which will enableto develop more efficient algorithms for recognizing three dimensional objects. This in-turn has applications in modeling proteins - antibody reactions which are used in drug designing.

Acknowledgement

I am highly indebted to the **INAE,** who have given me very useful opportunity by approving my mentorship program and enabling me to study the subjects of my interest. I would like to express my deep sense of gratitude to Prof. Kamala

Krithivasan for selecting me for mentorship program and spending her valuable time in guiding me throughout the period of my mentorship. I would also like to thank Dr. M.V Padmavati for their continuous guidance and support during the program. I am also indebted to the management, Bhilai Institute of Technology, without whose unstinting support and encouragement I could not have availed this rare opportunity.

References

[1] Bird R.S., Two dimensional pattern matching, Information Processing Letters 6 (1977), 168-170.

[2] Dersanambika K.S., Krithivasan K., Martin-vide C., Subramanian, K.G., Local and recognizable hexagonal picture languages. International Journal of Pattern Recognition 19 (2005), 853-871.

[3] Fu K.S., Syntactic Pattern Recognition and Applications, Prentice Hall Inc (1982).

[4] Giammarresi D., Restivo A., Two-dimensional finite state recognizability, Fundamenta Informaticae 25(3,4) (1996), 399-422.

[5] Inoue K., Nakamura A., Some properties of two dimensional online tessellation acceptors, Information Science 13 (1977), 95-121.

[6] Krithivasan K., Sitalakshmi R., An efficient two dimensional pattern matching algorithm, Technical report, CS-TR-1724, Centre for Automation Research, University of Maryland (1986).

[7] Polcar T., Melichar B., Two-dimensional pattern matching by two-dimentional online tessellation automata. Lecture Notes in Computer Science (2005), 327-328.

[8] Rozenberg G., Salomaa A.(eds.), Handbook of Formal Languages, Springer, Berlin (1997).

[9] Siromoney G., Siromoney R., Subramanian K.G., Stochastic table arrays, Computer Graphics and Image Processing 18(2) (1982), 202-211.

[10] Siromoney G., Siromoney R., Hexagonal arrays and rectangular blocks. Computer Graphics and Image Processing 5(3) (1976), 353-381.

[11] Subramanian K.G., Computer Graphics and Image Processing (1979).

[12] Toda M., Inoue K., Takanami I., Two-dimensional pattern matching by two-dimensional on-line tessellation acceptors, Theoretical Computer Science 24(2) (1983), 179-194.