Synthesis of Non-uniform Cellular Automata Having only Point Attractors

Nazma Naskar[1], Sumit Adak[2], Pradipta Maji[3], and Sukanta Das[2]

[1] Department of Information Technology,
Seacom Engineering College, Howrah, 711302, India
naskar.preeti@gmail.com
[2] Department of Information Technology,
Indian Institute of Engineering Science and Technology, Shibpur
(formerly Bengal Engineering and Science University, Shibpur), 711103, India
maths.sumit@gmail.com, sukanta@it.becs.ac.in
[3] Machine Intelligence Unit, Indian Statistical Institute,
203 B. T. Road, Kolkata, India 700108
pmaji@isical.ac.in

Abstract. This paper studies a special class of non-uniform cellular automata (CAs) that contain only single length cycle (point) attractors in their state space. These CAs always converge to some point attractors. A number of theorems and lemmas are reported in this paper to characterize this class of CAs. *Reachability tree*, a discrete tool for characterizing 1-d CA, has been utilized to develop theories for these types of CAs. We finally report an algorithm that *synthesizes* a non-uniform cellular automaton having only point attractors.

Keywords: Single length cycle attractor (point attractor), multi state attractor, reachability tree, cyclic states.

1 Introduction

Synthesis of a non-uniform cellular automaton (CA) refers to a process that selects individual rules for cells. Aim of this paper is to synthesize a non-uniform CA that always converges to some point attractors. The boundary condition is assumed here as null. This type of CAs attracted the researchers due to their utility in several applications, like pattern classification and recognition [1,6,3], design of associative memory, etc. [4].

However, to ensure that the CA will always converge to point attractors, we have to explore the state space of the CA to see whether it contains any multi-length cycle attractor. To efficiently synthesize our required CA, we first characterize them. An introductory characterization has already been reported in [5]. We use *reachability tree* [2], a mathematical tool, in our characterization. The tool is further utilized to develop synthesis algorithm.

J. Wąs, G.C. Sirakoulis, and S. Bandini (Eds.): ACRI 2014, LNCS 8751, pp. 105–114, 2014.

Table 1. The rules 5, 73, 200 and 80

Present state :	111	110	101	100	011	010	001	000	*Rule*
(RMT)	(7)	(6)	(5)	(4)	(3)	(2)	(1)	(0)	
(i) Next state :	0	0	0	0	0	1	0	1	5
(ii) Next state :	0	1	0	0	1	0	0	1	73
(iii) Next state :	1	1	0	0	1	0	0	0	200
(iv) Next state :	0	1	0	1	0	0	0	0	80

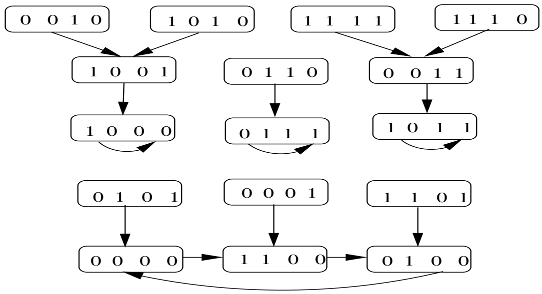

Fig. 1. State Transition Diagram of CA $\langle 5, 73, 200, 80 \rangle$

2 Preliminaries

The cellular automata (CAs) we consider here are the elementary cellular automata that use null boundary condition. The next state functions of these CAs, known as 'rule' [7], are commonly represented through a tabular form (Table 1). The first row of the Table 1 lists the possible 8 combinations of the present states of left, self and right neighbor of a cell. The last four rows indicate the next states of the cell for the rules 5, 73, 200 and 80 respectively.

Traditionally, the cells of an automaton follow same rule. Such a CA is *uniform CA*. In a *non-uniform CA*, the cells may follow different rules. We refer a rule vector $\mathcal{R} = \langle \mathcal{R}_0, \cdots \mathcal{R}_i, \cdots \mathcal{R}_{n-1} \rangle$ for an n-cell non-uniform CA, where the cell i follows $\mathcal{R}_i$. Obviously, uniform CAs are special case of non-uniform CAs.

The first row of Table 1 notes the combinations of the present states of three neighbors. Borrowing vocabulary from *Switching Theory*, we refer each combination as a Rule Min Term (RMT) [2]. Here we introduce a set Z_8^i that contains the valid RMTs of $\mathcal{R}_i$. That is, $Z_8^i = \{k \mid \text{RMT } k \text{ of } \mathcal{R}_i \text{ is valid}\}$. Since we have 8 RMTs (Table 1), generally $|Z_8^i| = 8$.

The state transition diagram (see Fig. 1) of an automaton may contain *cyclic* and *acyclic* states. The states 0000, 1100, 0100, 1000, 0111 and 1011 are the cyclic states, and they form attractors. The 0000, 1100 and 0100 form an attractor of length 3, whereas the rest three states form three attractors of length 1 (point attractors). In this work, we put our attention on those CA which contain only

Table 2. Relationship between i^{th} and $(i+1)^{th}$ RMTs

i^{th} RMT	$(i+1)^{th}$ RMTs
0, 4	0, 1
1, 5	2, 3
2, 6	4, 5
3, 7	6, 7

point attractors. However, the acyclic states can be of two types - *reachable* and *non-reachable*. A state is reachable if it has at least one predecessor. The acyclic states 1001 and 0011 are reachable, whereas the 0010, 1010, 0110, 1111 (Fig. 1) are non-reachable.

A CA state can also be viewed as a sequence of RMTs. For example, the state 1110 in null boundary condition can be viewed as $\langle 3764 \rangle$, where 3, 7, 6 and 4 are corresponding RMTs on which the transition of first, second, third and forth cells can be made. For an n-bit state, we get a sequence of n RMTs. However, two consecutive RMTs in an RMT sequence (RS) are related [2]. The relation is noted in Table 2. We call two RMTs r and s $(r \neq s)$ equivalent to each other if $2r \pmod 8 = 2s \pmod 8$. Therefore, RMTs 0 and 4 are equivalent to each other. Similarly, RMTs 1 and 5, RMTs 2 and 6, and RMTs 3 and 7 are equivalent to each other.

3 Reachability Tree (RT) and State Transition

Definition 1. *Reachability tree for an n-cell cellular automaton under null boundary condition is a rooted and edge-labeled binary tree with $n+1$ levels, where $E_{i.2j} = (N_{i.j}, N_{i+1.2j}, l_{i.2j})$ and $E_{i.2j+1} = (N_{i.j}, N_{i+1.2j+1}, l_{i.2j+1})$ are the edges between nodes $N_{i.j} \subseteq Z_8^i$ and $N_{i+1.2j} \subseteq Z_8^{i+1}$ with label $l_{i.2j} \subseteq N_{i.j}$, and between nodes $N_{i.j}$ and $N_{i+1.2j+1} \subseteq Z_8^{i+1}$ with label $l_{i.2j+1} \subseteq N_{i.j}$ respectively $(0 \leq i \leq n-1,\ 0 \leq j \leq 2^i - 1)$. Following relations are maintained in the tree:*

1. $l_{i.2j} \cup l_{i.2j+1} = N_{i.j}$
2. *$\forall r \in l_{i.2j}$ (resp. $\forall r \in l_{i.2j+1}$), RMT r of $\mathcal{R}_i$ is 0 (resp. 1) and RMTs $2r \pmod 8$ and $2r+1 \pmod 8$ of $\mathcal{R}_{i+1}$ are in $N_{i+1.2j}$ (resp. $N_{i+1.2j+1}$)*
3. *$\bigcup_{0 \leq j \leq 2^i - 1} N_{i.j} = Z_8^i$, $(0 \leq i \leq n-1)$*

Fig. 2 shows the reachability tree for the CA of Fig. 1. Under the null boundary condition, only 4 RMTs are valid for left most and right most cells, and $Z_8^0 = \{0,1,2,3\}$ and $Z_8^3 = \{0,2,4,6\}$. Hence, the root $N_{0.0} = Z_8^0$. The label of edge $E_{0.0}$ is $\{1, 3\}$, as RMTs 1 and 3 of rule 5 are 0. We write RMTs of a label on the edge within a bracket. However, the label of edge $E_{2.7}$ is empty, that is $l_{2.7} = \phi$. This edge is non-reachable. Fig. 2 marks such nodes as black. Since $Z_8^n = \phi$ for an n-cell CA, the leaf nodes are empty. Number of leaves (excluding black leaves as they don't exit) is 8, which is the number of reachable states. We call edge $E_{i.2j}$ as 0-edge and $E_{i.2j+1}$ as 1-edge where $0 \leq j \leq 2^i - 1$. A sequence of edges

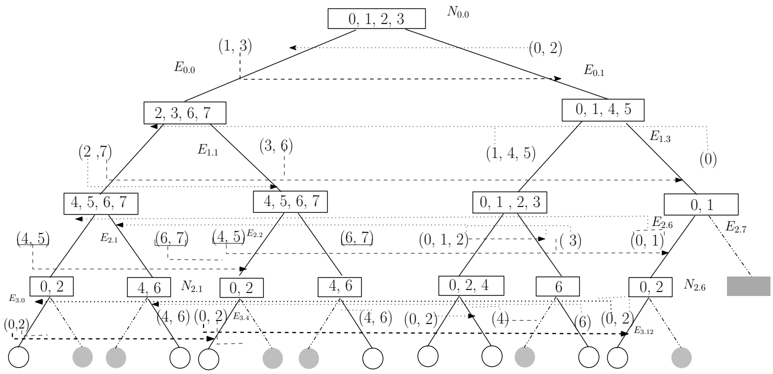

Fig. 2. Reachability Tree of CA $\langle 5, 73, 200, 80 \rangle$. The links are also shown.

from the root to a leaf node represents a reachable state, when 0-edge and 1-edge are replaced by 0 and 1 respectively. For example, 0011 is a reachable state in Fig. 2. On the other hand, the states 1110 and 1111 are non-reachable.

Reachability tree gives us information about reachable states. A sequence of edges $\langle E_{0.j_0} E_{1.j_1} \cdots E_{i.j_i} E_{i+1.j_{i+1}} \cdots E_{n-1.j_{n-1}} \rangle$ from root to a leaf associates a reachable state and at least one RS $\langle r_0 r_1 \cdots r_i r_{i+1} \cdots r_{n-1} \rangle$, where $r_i \in l_{i.j_i}$ and $r_{i+1} \in l_{i+1.j_{i+1}}$ $(0 \leq i < n-1, 0 \leq j_i \leq 2^i - 1$, and $j_{i+1} = 2j_i$ or $2j_i + 1)$. That is, the sequence of edges represents at least two CA states. Note that if RMT r_i is 0 (resp. 1) then $E_{i.j_i}$ is 0-edge (resp. 1-edge). Therefore, the reachable state is the next (resp. present) state of the state (resp. predecessor), represented as RMT sequence. Interestingly, there are 2^n RSs in the tree, but number of reachable states may be less than 2^n. However, a sequence of edges may associate m-number of RSs $(m \geq 1)$, which implies, this state is reachable from m-number of different states.

Since the RSs and the states, both of an automaton can be traced in the tree, which RS corresponds to what state can be identified. To identify this correspondence, we form *links* among edges. The links are formed for each RMT $r \in l_{i.j}$, present on edge $E_{i.j}$ $(0 \leq i \leq n-1, 0 \leq j \leq 2^i - 1)$. By the processing of reachability tree, we find the links among the edges for each individual RMT on the tree. The links are formed depending on whether the RMTs are self replicating (defined below) or not.

Definition 2. *An RMT $x0y$ (resp. $x1y$) is said to be self replicating if RMT $x0y$ (resp. $x1y$) is 0 (resp. 1).*

For example, RMT 2 (010) of rule 5 is self replicating, whereas all the RMTs except RMT 2 of rule 200 are self replicating (see Table 1). If an RMT $r \in l_{i.j}$ is not self replicating, then there is a link from the edge $E_{i.j}$ to $E_{i.k}$ $(j \neq k)$. Depending on the values of j and k, we can classify the links in the following way: forward link (when $j < k$), backward link (when $j > k$) and self link (when

$j = k$). The rules, followed to form links in a reachability tree, are noted below:

1) If RMT $r \in l_{0.j}$ is self replicating ($j = 0$ or 1),
The edge $E_{0.j}$ is self linked for RMT r.
Otherwise, if $j = 0$,
there is a forward link from $E_{0.0}$ to $E_{0.1}$ for RMT r;
else, there is a backward link from $E_{0.1}$ to $E_{0.0}$ for RMT r.

2) If $E_{i-1.j}$ is self linked for RMT $r \in l_{i-1.j}$,
and if s is self replicating
where $s \in l_{i.2j}$ (resp. $s \in l_{i.2j+1}$) is $2r \pmod 8$ or $2r+1 \pmod 8$,
then $E_{i.2j}$ (resp. $E_{i.2j+1}$) is self linked.
But if s is not self replicating,
then there is a forward link from $E_{i.2j}$ to $E_{i.2j+1}$ (resp. backward link from $E_{i.2j+1}$ to $E_{i.2j}$).

3) If there is a link from $E_{i-1.j}$ to $E_{i-1.k}$ ($j \neq k$) for RMT $r \in l_{i-1.j}$,
and $s \in l_{i.2j}$ (resp. $s \in l_{i.2j+1}$) is $2r \pmod 8$ or $2r+1 \pmod 8$,
then there is a link from $E_{i.2j}$ (resp. $E_{i.2j+1}$) to $E_{i.2k}$ while $s \in \{0,1,4,5\}$ or to $E_{i.2k+1}$ while $s \in \{2,3,6,7\}$.
[It is forward link if $j < k$, backward link if $j > k$]

Example 1. We apply the above rules on each RMT of each edge to get the links in the tree. Fig. 2 shows the links of edges caused by RMTs of the CA $\langle 5, 73, 200, 80\rangle$. There is a (forward) link from $E_{0.0}$ to $E_{0.1}$ for RMT 3 (forming links using rule 5), so a dotted line is drawn from RMT 3 of $E_{0.0}$ to $E_{0.1}$. Now, we get a forward link from $E_{1.1}$ to $E_{1.3}$ for RMT 6 (forming links using rule 73). However, no lines are shown in Fig. 2 for RMTs involved in self links. Now, we can get links from $E_{2.2}$ to $E_{2.6}$ for RMT 4 (forming links using rule 200), and from $E_{3.4}$ to $E_{3.12}$ for RMT 0 (forming links using rule 80). Therefore, for the RS $\langle 3640\rangle$, we can get a sequence of links, hence a sequence of edges $\langle E_{0.1}E_{1.3}E_{2.6}E_{3.12}\rangle$, which points to 1100. Note that the RS $\langle 3640\rangle$ corresponds to the state 1100. The sequence $\langle E_{0.0}E_{1.1}E_{2.2}E_{3.4}\rangle$ associates the state 0100, as well as the RS $\langle 3640\rangle$. The RS $\langle 3640\rangle$, hence the state 1100, is the predecessor of the state 0100. See Fig. 1 for verification. Therefore, the links establish relationships among the states.

Lemma 1. *There exist only two links to $E_{i.j}$, for any j,from $E_{i.k}$ and from $E_{i.l}$ when $0 \leq i \leq n-1$, and only one link to $E_{i.j}$ from $E_{i.m}$ when $i = n-1$ in a reachability tree ($0 \leq k, l, m \leq 2^i - 1$).*

As an example, edge $E_{0.0}$ of Fig. 2 has only two links, one from $E_{0.0}$ for RMT 1 (self link) and another from $E_{0.1}$ for RMT 0 (backward link). Same is true for $E_{0.1}$ for RMTs 2 and 3. In the next level, each edge has two links for two RMTs. To $E_{1.0}$ link from $E_{1.3}$ (for RMT 0) and $E_{1.2}$ (for RMT 1), to $E_{1.1}$ link from $E_{1.0}$ (for RMT 2) and $E_{1.1}$ (for RMT 3). In the leaf, however $E_{3.0}$ has only one link from $E_{3.12}$ for RMT 0.

Cross link: Applying the following rules of links, we can get a forward (or backward) link, like $E_{i.j_1}(r_1) \rightarrow E_{i.j_2}(r_2) \rightarrow \cdots \rightarrow E_{i.j_{q-1}}(r_{q-1}) \rightarrow E_{i.j_q}$. Now, for some values of i ($0 \leq i \leq n-1$), we may find forward links and backward links which combined form a loop. That is, a cycle of links like $E_{i.j_1}(r_1) \rightarrow \cdots \rightarrow E_{i.j_q}(r_q) \rightarrow \cdots \rightarrow E_{i.j_m}(r_m) \rightarrow E_{i.j_1}$ can be observed. We refer this link as *cross link*. We define the length of a cross link as the number of RMTs involved in the link (here, it is m).

Example 2. Let us consider Fig. 2 for the CA $\langle 5, 73, 200, 80 \rangle$. We get a cross link: $E_{0.0}(2) \rightarrow E_{0.1}(1) \rightarrow E_{0.0}$. It is noticed that at each level of Fig. 2, a cross link exists. Finally we get a cross link among $E_{3.j}$s: $E_{3.0}(0) \rightarrow E_{3.4}(0) \rightarrow E_{3.12}(0) \rightarrow E_{3.0}$ (Fig. 2). Length of the cross link is 3, and it can be noted that length of multi state attractor of the CA is also 3 (Fig. 1).

From Example 2, we can see that the cross link plays an important role in forming multi state attractors. In this part, we report some characteristics of cross link which affect multi state attractors.

Theorem 1. *An n-cell CA contains multi state attractor, if a cross link among $E_{n-1.k}$s exists.*

Example 3. There exists a cross link of length 3 in Fig. 2: $E_{0.0}(2) \rightarrow E_{0.1}(1) \rightarrow E_{0.0}$. Corresponding CA (Fig. 1) has a multi state attractor of length 3.

Theorem 2. *An RMT $r \in l_{i.j}$ can not be a part of a cycle, if the RMT is not involved in a self link or cross link ($0 \leq i \leq n-1$, $0 \leq j \leq 2^i - 1$).*

Corollary 1. *An n-cell CA contains m number of point attractors, if m number of self-linked $E_{n-1.k}$s exist.*

Example 4. According to Fig. 2, at leaf level there is 3 self linked at $E_{3.7}(6)$, $E_{3.8}(0)$ and $E_{3.11}(6)$, and also from state transition diagram (Fig. 1) we can see that there are three point attractors. Hence, only self links form point attractors at leaf level.

Theorem 3. *An n-cell CA contains at least one multi state attractor, if a set of RMTs form cross link among the edges $E_{i.j_1}$, $E_{i.j_2}$, $\cdots$, $E_{i.j_k}$ where $0 \leq j_k \leq 2^i - 1$, and the edges are not involved in any self link.*

Corollary 2. *An n-cell CA contains at least one multi state attractor if a RMTs $r_1, r_2, \cdots, r_s$ of $\mathcal{R}_i$ form cross link among edges $E_{i.j}$s and RMTs $2r_1 \pmod 8$, $2r_1 + 1 \pmod 8$, $2r_2 \pmod 8$, $2r_2 + 1 \pmod 8$, $\cdots$, $2r_s \pmod 8$, $2r_s + 1 \pmod 8$ also participate a cross link among $E_{i+1.k}$s.*

Based on the theories developed, we next report the synthesis of non-uniform CA that always converges to point attractors.

4 Synthesis of CA Having only Point Attractors

In this section, we discuss the procedure of getting a rule vector $\mathcal{R}$ of a non-uniform cellular automaton that contains only point attractors in its state space. According to Theorem 1, Theorem 2 and Corollary 1, we can identify following characteristics of reachability tree of an n-cell CA that contains only point attractors.

1. There exists at least one self-linked edge $E_{n-1.k}$ for any value of k.
2. There is no cross-link among the edges $E_{n-1.k}$s.

In the proposed synthesis scheme what we do is,
1. we first select $\mathcal{R}_0$, from root node of the reachability tree, get edges from the root, identify links between edges following rule 1 of link formation, and get the nodes of level 1,
2. then we select $\mathcal{R}_1$ and get the edges from the nodes and identify the links,
3. next we select $\mathcal{R}_2$ and repeat step 2, and so on. We finally get the tree and then verify if $E_{n-1.k}s$ contain only self links and no cross links.

4.1 Dealing with Self Link

It is obvious from the rules 1 and 2 of link formation that to get self linked edge $E_{n-1.k}$ for some k, there has to exist at least one self-linked edge $E_{i.k}$ for any value of k, where $0 \leq i \leq n-1$. To get the point attractors from reachability tree, we allow only self link. If we only allow self link, like as *rule* 204, where all the RMTs belongs to self link, then at leaf 2^n nodes (for an n-cell CA) are reachable and form 2^n point attractors.

Generally for a n-cell CA ($n > 2$), we observe that many of the nodes carry same property, as well as all the RMTs of the nodes are in self link. If we get two or more nodes with same property at any level, then we can consider only one for further processing. Two nodes are said to be *sub-node* of each other if all the RMTs of one node are same or equivalent RMTs of another node. As an example, in Fig. 3, $N_{2.0}$ and $N_{2.3}$ are sub-node of each other because the RMTs of $N_{2.0}$ ($N_{2.0} = \{0, 1, 2, 3\}$) are same or equivalent of $N_{2.3}$ ($N_{2.3} = \{4, 5, 6, 7\}$). Therefore, we only take one node for further processing. To maintain the characteristics (1), we use following condition.

Condition 1. *If RMTs $r_1, r_2, \cdots, r_k$ of $\mathcal{R}_i$ participate in self links, then either RMTs $2r_1, 2r_2, \cdots, 2r_k \pmod 8$ or RMTs $2r_1+1, 2r_2+1, \cdots, 2r_k+1 \pmod 8$ of $\mathcal{R}_{i+1}$ are self replicating.*

4.2 Dealing with Cross Link

To maintain the characteristics (2), we can deal with only self links and do not allow any cross link at any level. If we do not allow any cross link at intermediate levels, then only self link can exist. Therefore we get the all attractors as a point

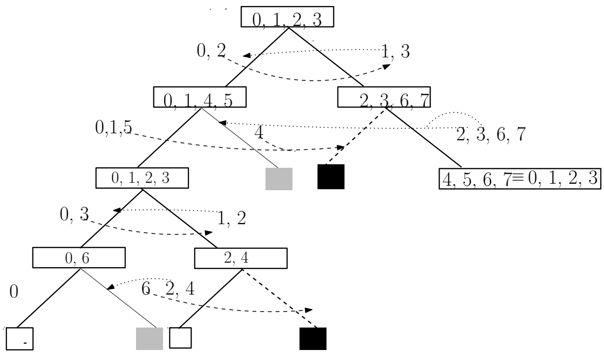

Fig. 3. Reachability tree $\langle 106, 220, 102, 96\rangle$ with links

attractor, but its a trivial case and then we will get very limited number of CA. So, we allow cross links at intermediate levels. But before the leaf level the cross links have to be disappeared.

Lemma 2. *If there is a cross link at level i of a reachability tree which trigger cross links at level $(i+1)$, level $(i+2)$, level $(i+3)$, then there always exists at least one cross links at level j of the tree where $0 \le i \le n-1$ and $i+3 < j \le n-1$.*

Since we want to synthesize a CA that does not have any multi state attractors, we select $\mathcal{R}_{i+1}$ such a way that the cross link of level i, if any, can not trigger any cross link at level $(i+3)$. To guarantee this, we select $\mathcal{R}_{i+1}$ in such a fashion that the RMTs of the rule follow Condition 2.

Condition 2. *If RMTs r_1 and r_2 of $\mathcal{R}_i$ participate in a cross link at level i, then RMTs $2r_1 \pmod 8$ and $2r_2+1 \pmod 8$ of $\mathcal{R}_{i+1}$ are self replicating (resp. non self replicating), and RMTs $2r_1+1 \pmod 8$ and $2r_2 \pmod 8$ of $\mathcal{R}_{i+1}$ are non self replicating (resp. self replicating).*

4.3 The Weight

A RMT r ($r \in l_{i.j}$) may be involved in more than one link, whether the links are forward or backward (part of cross link) or self. This situation comes, when an edge $l_{i.j} = \{0, 4\}$ where RMT 0 is self linked and RMT 4 makes a forward link to $E_{i.k}$. Then $l_{i+1.2j}$ or $l_{i+1.2j+1}$ contains RMTs 0 and 1 (as per Table 2). So, RMT 0 or 1 comes twice (one from RMT 0 and one from RMT 4), whether the link will be self (follow the link of RMT 0 of $E_{i.j}$) or forward (follow the link of RMT 4 of $E_{i.j}$). As an example, in Fig. 2, the RMT 2 of $E_{2.4}$ and RMT 3 of $E_{2.5}$ has two links.

To handle this situation, We introduce $w^r_{i.j}$ as the weight of RMT $r \in l_{i.j}$. The weight $w^r_{i.j}$ is the total number of links from $E_{i.j}$ to itself or to some other edges for RMT r. If $l_{i.j} = \phi$ (for some j), the edge $E_{i.j}$ is non-reachable. Now, for each $s \in l_{i.k}$, we decrease $w^s_{i.k}$ by 1 if $E_{i.k}(s) \to E_{i.j}$ for any value of k. After this decrement, if the weights of all RMTs of $l_{i.k}$ have become 0, we consider the edge as non-reachable. Note that $w^s_{i.k}$ can never be 0 if RMT s is involved in a self or cross link. In here, we will not consider those RMTs ($s \in l_{i.j}$) which weight is 0 ($w^s_{i.j} = 0$). So, weight of an RMT may be more than one in some cases. As an example, in Fig. 2, weight of all RMTs of $l_{0.j}s$ and $l_{1.j}s$ is 1. But in $l_{2.j}s$, we find the RMTs which weight is 2 ($w^2_{2.4} = w^3_{2.5} = 1$), therefore those RMTs have two links.

4.4 Algorithm

According to Theorem 2, only those RMTs can be part of cycle which participate in cross link or make a self link. Therefore, we deal with only those RMTs which either in self link or cross link. The algorithm deals with the labels of edges and we do not form whole tree at a time. Rather we deal with two labels – $\{l_{i.0}, l_{i.1}, \cdots l_{i.2^i-1}\}$ and $\{l_{i+1.0}, l_{i+1.1}, \cdots l_{i+1.2^{i+1}-1}\}$. We proceed with only non-empty labels, $l_0,\ l_1, \cdots$ and $l'_0,\ l'_1, \cdots$. Here, l_j corresponds to the label of $E_{i.j}$ and l'_k correspond to the label of $E_{i+1.k}s$ $(0 \le i \le n-1)$. We report the desired CA (rule vector) that only contain point attractors.

Algorithm 1. SynPointStateAttrCA
Input: *n (CA size).*
Output: *$\langle \mathcal{R}_0, \mathcal{R}_1, \cdots, \mathcal{R}_{n-1} \rangle$ (n-cell CA).*
Step 1: *Select $\mathcal{R}_0$ so that at least one RMT is self replicating. Put each valid RMT r of $\mathcal{R}_0$ in l_0 (resp. l_1) if RMT r is 0 (resp. 1).*
Step 2: *For $i = 1$ to $n-1$, repeat Step 3 to Step 11.*
Step 3: *If i equals to $n-1$, then goto Step 9.*
Step 4: *Find and store $2r \pmod 8$ and $2r+1 \pmod 8$ for all RMTs r, that are self linked at $(i-1)^{th}$ level and set RMTs at i^{th} level using Condition 1.*
Step 5: *Check whether a cross link exists for any RMTs of $l_j s$.*
If exists, goto Step 7.
Otherwise, goto Step 6.
Step 6: *Set all blank RMTs of $\mathcal{R}_i$ arbitrarily, goto Step 10 and discard all forward and backward links.*
Step 7: *For each cross link at $(i-1)^{th}$ level, set RMTs at i^{th} level using Condition 2 and fill remaining blank RMTs of $\mathcal{R}_i$ arbitrarily.*
Step 8: *Check whether any overlapping situation occur for any RMT of $\mathcal{R}_i$.*
If exists, goto Step 9.
Otherwise, goto Step 10.
Step 9: *Find a $\mathcal{R}_i$, that discard all cross link.*
Step 10: *For each non-empty labels l_k*
Find l'_{2k} and l'_{2k+1} so that, if $r \in l_k$ and $s = 2r \pmod 8$ or $2r+1 \pmod 8$, then $s \in l'_{2k}$ (resp. l'_{2k+1}) when RMT s of $\mathcal{R}_i$ is 0 (resp. 1).
Step 11: *Assign non-empty and unique l'_j to l_k so that the links among l'_j for each RMT is maintained.*
Step 12: *Report the CA $\langle \mathcal{R}_0, \mathcal{R}_1, \cdots, \mathcal{R}_{n-1} \rangle$.*

Theorem 4. *If cross links does not exist in i^{th} level of Algorithm 1, then at i^{th} level maximum number of self linked edges is 4 $(0 \le i \le n-1)$.*

Corollary 3. *Complexity of Algorithm 1 is $O(n)$.*

Proof. The complexity of the algorithm depends on n only (Step 2). We set rules within Step 4 to Step 8. In Step 4 we set Rule $\mathcal{R}_i$ for i^{th} level depending on self link of $(i-1)^{th}$. Step 5 checks existence of cross link. If cross link does not occur, then according to Theorem 4, there are maximum 4 unique edge. So, at the leaf level we can get maximum 4 unique edge (because in leaf level, there is no cross link). Therefore, complexity of the algorithm is $O(n)$.

Example 5. Let us consider the synthesis of a 4-cell CA. First, we select 102 as $\mathcal{R}_0$ randomly which has at least one self replicating RMT (RMTs 0 and 3 are self replicating). One cross link is formed between $E_{0.0}$ and $E_{0.1}$ $[E_{0.0}(2) \rightarrow E_{0.1}(1) \rightarrow E_{0.0}]$ (See Fig. 3 for verification). Now, using *Step 4* and *Step 7* rule 220 is selected as $\mathcal{R}_1$. In rule 220, most of the equivalent RMTs have same values, therefore, there is no cross link among $E_{1.j}$s. Number of unique non-empty set (node) is reduced to one. We now select rule 102 as $\mathcal{R}_2$ (using *Step 4* and *Step 6*). Finally, we select rule 96 as $\mathcal{R}_3$ (using *Step 9*). Therefore, the CA is $\langle 106, 220, 102, 96\rangle$, which contains only point attractors (Fig. 3 has no cross link at leaf level).

References

1. Bandini, S., Vanneschi, L., Wuensche, A., Shehata, A.B.: Cellular automata pattern recognition and rule evolution through a neuro-genetic approach. J. Cellular Automata 4(3), 171–181 (2009)
2. Das, S., Sikdar, B.K., Pal Chaudhuri, P.: Characterization of Reachable/Nonreachable Cellular Automata States. In: Sloot, P.M.A., Chopard, B., Hoekstra, A.G. (eds.) ACRI 2004. LNCS, vol. 3305, pp. 813–822. Springer, Heidelberg (2004)
3. Maji, P., Shaw, C., Ganguly, N., Sikdar, B.K., Pal Chaudhuri, P.: Theory and Application of Cellular Automata For Pattern Classification. Special Issue of Fundamenta Informaticae on Cellular Automata 58, 321–354 (2003)
4. Maji, P., Pal Chaudhuri, P.: Non-uniform cellular automata based associative memory: Evolutionary design and basins of attraction. Inf. Sci. 178(10), 2315–2336 (2008)
5. Naskar, N., Adak, S., Das, S.: Identification of non-uniform periodic boundary cellular automata having only point states. In: AUTOMATA 2013: 19th International Workshop on Cellular Automata and Discrete Complex Systems, pp. 67–75 (September 2013)
6. Raghavan, R.: Cellular automata in pattern recognition. Information Sciences 70(12), 145–177 (1993)
7. Wolfram, S.: Theory and Application of Cellular Automata. World Scientific (1986)