On Invertible Three Neighborhood Null-Boundary Uniform Cellular Automata

Soumyabrata Ghosh*
Nirmalya S. Maiti
P. Pal Chaudhuri
Cellular Automata Research Lab (CARL)
Alumnus Software Limited, Sector-V
Kolkata, West Bengal, India 700091

Biplab K. Sikdar

Department of Computer Science and Technology
Bengal Engineering and Science University, Shibpur
West Bengal, India 711103
**soumyabrata@alumnux.com*

This paper solves the problem of determining the number of cells in an invertible three neighborhood null-boundary uniform cellular automaton (CA) by using its rule vector graph (RVG). The RVG represents an efficient data structure designed to characterize CA evolution and is derived out of its rule vector (RV). The concept of a horizontal rule vector subgraph (HRVS) is introduced to formulate the analytical framework of the solution. The RVG of a CA is partitioned into a number of identical HRVSs. It has been shown that invertible CA size depends on the size of the HRVS.

1. Introduction

The theory and applications of cellular automata (CAs) were initiated in [1] and carried forward by a large number of authors [2–27]. The rule vector (RV) of an n cell hybrid CA is denoted as $\langle R_0\, R_1\, R_2 \ldots R_i \ldots R_{(n-1)}\rangle$ where rule R_i is employed on the i^{th} cell. If the same rule is employed for each of the cells, it is referred to as a uniform CA, where $R_0 = R_1 = R_2 = \cdots = R_i = \cdots = R_{(n-1)}$. This paper deals with uniform CAs unless mentioned otherwise. Uniform null-boundary three neighborhood CAs are referred to as simply CAs in the rest of this paper. We address the problem of determining the number of cells in an invertible CA.

A linear time algorithm is reported in [27] to identify the invertibility of a CA (uniform or hybrid). All 256 rules of three neighborhood CAs, as per [26], can be divided into 88 groups of elementary rules. The value of n for which a rule group generates invertible CAs has been reported in Table 3 [27] without any formal proof. Section 4 of this paper presents a formal proof of the correctness of those results.

https://doi.org/10.25088/ComplexSystems.20.1.47

The proof has been derived by employing the horizontal rule vector subgraph (HRVS) introduced in Section 3. A brief introduction to the RVG follows in Section 2.

2. Rule Vector Graph of a Cellular Automaton

The RVG construction from the RV of a CA (Figure 1) has been detailed in [27]. For the sake of completeness of the current paper, a brief overview of the RVG is introduced along with an explanation for a few basic terminologies.

The eight minterms of the three-variable Boolean function f_i, corresponding to the rule R_i employed on the i^{th} cell (Figure 1(b)) are referred to as *rule minterms* (RMTs) (Table 1). The three Boolean variables are a_{i-1}, a_i, a_{i+1}, the current state values of cells $(i-1)$, i, and $(i+1)$ respectively, whereby the minterm $m = \langle a_{i-1}\, a_i\, a_{i+1} \rangle$. The symbol T($m$) denotes a single RMT in the text and is noted simply as m for clarity in the figures. {T} represents the set of all eight RMTs, whereby {T} = {T(0), T(1), T(2), T(3), T(4), T(5), T(6), T(7)} = {T(m)}.

A CA rule divides the RMTs into two subsets referred to as *0-RMT* and *1-RMT*, denoted as $\{T_0^i\}$ and $\{T_1^i\}$ respectively, where $\{T_0^i\} \cap \{T_1^i\} = \phi$, $\{T_0^i\} \cup \{T_1^i\} = \{\mathrm{T}\}$.

The derivation of RMT T^{i+1} for cell $(i+1)$ out of $T^i \in \{\mathrm{T}(m)\}$ is noted in Table 2. The RVG of an n cell CA with the RV $\langle R_0\, R_1 \ldots R_{n-1} \rangle$ has n levels 0 to $(n-1)$.

A *node* in a RVG represents a subset of RMTs. An output node of level i is derived from its input node through the RMT transitions given in Table 2. The output node of level i is the input node of level $(i+1)$ corresponding to the rule R_{i+1}.

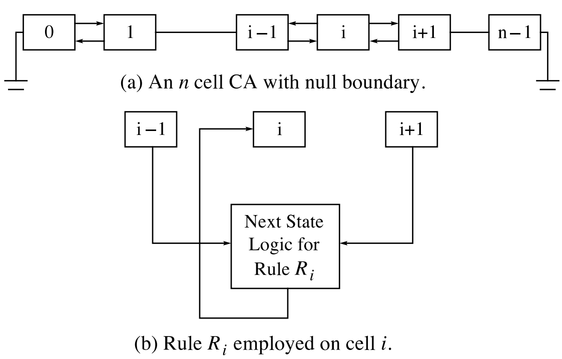

Figure 1. General structure of a CA employing the RV $\langle R_0\, R_1 \ldots R_i \ldots R_{(n-1)} \rangle$ of an n cell CA.

https://doi.org/10.25088/ComplexSystems.20.1.47

Next State Value b_i for Present States $\langle a_{i-1}\, a_i\, a_{i+1}\rangle$ Represented by the RMTs								
T(7)	T(6)	T(5)	T(4)	T(3)	T(2)	T(1)	T(0)	Rule Number
7	6	5	4	3	2	1	0	
1	1	0	0	1	0	1	0	202
1	0	1	0	0	1	1	0	166
0	1	0	1	1	0	1	0	90
0	0	0	1	0	1	0	0	20
0	1	1	1	1	0	0	0	120

Table 1. The RMT and CA rules. Note 1: (a) The left column represents the next state value b_i of cell i for the present state values $\langle a_{i-1}\, a_i\, a_{i+1}\rangle$ of the $(i-1)$, i, and $(i+1)$ cells. (b) The eight minterms $\langle a_{i-1}\, a_i\, a_{i+1}\rangle = 000$ to 111 are represented as T(0) to T(7) in the text and 0 to 7 in the figures. Note 2: The decimal value of the 8-bit binary pattern in the left columns is referred to as the rule number.

RMT $T^i\, \langle a_{i-1}\, a_i\, a_{i+1}\rangle$	RMT $T^{i+1}\, \langle a_i\, a_{i+1}\, a_{i+2}\rangle$
$T(0)$ (000) and $T(4)$ (100)	$T(0) = 000\,(0)$ $T(1) = 001\,(1)$
$T(1)$ (001) and $T(5)$ (101)	$T(2) = 010\,(2)$ $T(3) = 011\,(3)$
$T(2)$ (010) and $T(6)$ (110)	$T(4) = 100\,(4)$ $T(5) = 101\,(5)$
$T(3)$ (011) and $T(7)$ (111)	$T(6) = 110\,(6)$ $T(7) = 111\,(7)$

Table 2. RMT transition. The left column refers to the RMTs of cell i while the right column refers to the corresponding RMTs of the $(i+1)$ cell (derived by deleting a_{i-1} and appending 0 and 1 as a_{i+2}).

A RVG *edge* represents the RMT transition from the input to an output node of a level.

The *0-edge* and *1-edge* refer to the edges from an input node of level i corresponding to the rule R_i employed on cell i ($i = 0$ to $(n-1)$). The b_i edge ($b_i \in \{0, 1\}$) is assigned the *edge weight* $\left\{T^i_{b_i}\right\} \big/ b_i$, where $\left\{T^i_{b_i}\right\}$ represents the set of RMTs for rule R_i for which the next state value is b_i.

Generating RVG(i) (i.e., the i^{th} level RVG for rule R_i) is noted in Algorithm 1 [27]. For deriving the RVG of an n cell CA, Algorithm 2 [27] calls Algorithm 1 for each level i ($i = 0$ to $(n-1)$). Figure 2 shows the RVG and state transition graph (STG) for the noninvertible CA with RV $\langle 202\,202\,202\rangle$. Figure 3 shows the RVG and STG of an invertible CA.

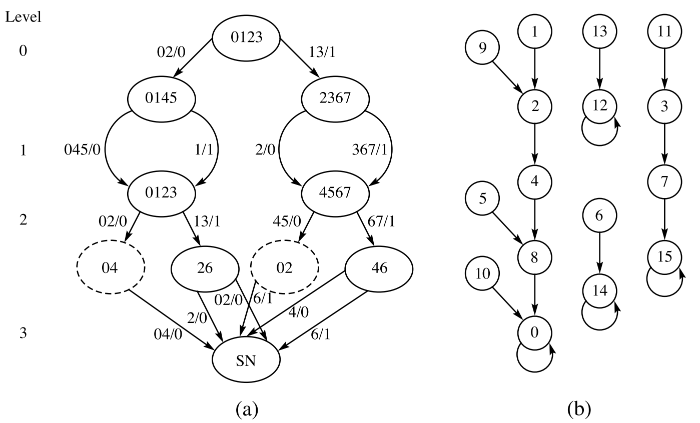

Figure 2. (a) RVG and (b) STG of a four cell uniform CA with RV ⟨202 202 202 202⟩.

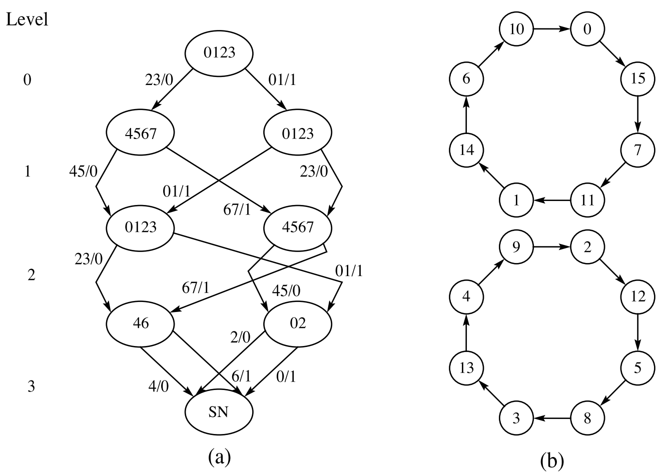

Figure 3. (a) RVG and (b) STG of a four cell uniform invertible CA with RV ⟨195 195 195 195⟩.

For a null-boundary CA, the leftmost cell (i.e., cell 0) can have RMTs T(0), T(1), T(2), T(3). Consequently, the input node for level 0 {T(0), T(1), T(2), T(3)} is referred to as the *root node* (RN).

For a null-boundary n cell CA, the $(n-1)$ cell can have RMTs T(0), T(2), T(4), T(6). Consequently, the input nodes and edges on level $(n-1)$ can have only even valued RMTs. The output node of level $(n-1)$ is marked as the *sink node* (SN).

Let V and V' be a pair of output nodes generated by Algorithm 1 [27], where each node covers a subset of RMTs. The node V' gets *merged* with the node V if $V' \subseteq V$. The resulting node V is referred to as a *merged* node. The two output nodes {T(0) T(1) T(2) T(3)} and {T(4) T(5) T(6) T(7)} (shown within bold line enclosures in Figure 2(a)) are merged nodes.

A node V is marked with the "Type 1" tag if there is a missing 0- or 1-edge outgoing from the node. The level 3 input nodes {T(0) T(4)} and {T(0) T(2)} are two Type 1 nodes (within dotted line enclosures in Figure 2(a)) due to missing 1-edges.

A merged node V is marked with the "Potential Type 2" tag if the merging has occurred for two nodes V' and V (where $V' \subset V$). The nodes {T(0) T(1) T(2) T(3)} and {T(4) T(5) T(6) T(7)} (within bold line enclosures in the level 1 output nodes) are the Potential Type 2 nodes in Figure 2(a). The subset V' is generated out of RMTs $\left\{T'^{i}_{b'_i}\right\}$ of the edge having weight $\left\{T'^{i}_{b'_i}\right\} \big/ b'_i$ and V is generated out of RMTs $\left\{T^{i}_{b_i}\right\}$ of the edge having weight $\left\{T^{i}_{b_i}\right\} \big/ b_i$, ($b_i \in \{0, 1\}$, $b'_i \in \{0, 1\}$), where $\left|\left\{T^{i}_{b_i}\right\}\right| > \left|\left\{T'^{i}_{b'_i}\right\}\right|$.

A Potential Type 2 node is noted with reference to the edge having weight $\left\{T'^{i}_{b'_i}\right\} \big/ b'_i$ that has fewer RMTs. The node {T(4), T(5), {T(6) T(7)} (Figure 2(a)) is a Potential Type 2 node with reference to the edge having weight T(2) / 0 that has fewer RMTs than the other incoming edge with weight {T(3), {T(6) T(7)} / 1.

A Potential Type 2 node V (at output level i) is marked as a "Type 2" node if:

1. A subpath can be identified from the node to the SN starting with a RMT $T^{i+1} \in \{V - V'\}$; and
2. no parallel subpath exists starting with a RMT $T'^{i+1} \in V'$, where T'^{i+1} is derived out of T'^{i} employing Table 2.

The Potential Type 2 nodes {T(0) T(1) T(2) T(3)} and {T(4) T(5) T(6) T(7)} (Figure 2(a)) are the Type 2 nodes.

The presence of a Type 1 and/or a Type 2 node in the RVG of a CA confirms the presence of nonreachable states (NRSs). Figure 2(a) illustrates the RVG of a noninvertible four cell CA with Type 1 and 2 nodes. The states 1, 5, 9, 13, and 6, 10, 11, as shown in Figure 2(b), are the NRSs due to Type 1 and 2 nodes, respectively.

The RVG of a CA can be traversed to locate the Type 1 and 2 nodes along with the identification of NRSs. If no Type 1 or 2 nodes exist, the CA is invertible (Theorem 1). Figure 3(a) shows the RVG of a four cell CA with the RV ⟨195 195 195 195⟩. Since there are no Type 1 or 2 nodes (Figure 3(a)), the four cell CA ⟨195 195 195 195⟩ is invertible. The value of n for such an n cell invertible CA is identified

https://doi.org/10.25088/ComplexSystems.20.1.47

in Section 4 subsequent to the introduction of the horizontal rule vector subgraph in Section 3.

3. Horizontal Rule Vector Subgraphs

The level i ($i = 0, 1, \ldots (n-1)$) rule vector subgraph RVS(i) of the RVG of an n cell CA refers to the input and output nodes of level i connected with weighted edges.

RVS(i, j) ($j \geq i$) refers to a subgraph that covers RVS(i), RVS($i+1$), … , RVS(j) where the output nodes of RVS(k) are the input nodes of RVS($k+1$), $k = (i), (i+1), \ldots, (j)$. In view of enforcing the horizontal partitioning of a RVG, a RVS(i, j) in subsequent discussions is referred to as a *horizontal rule vector subgraph* HRVS(i, j). For $i = j$, RVS(i, j) is the same as RVS(i).

3.1 HRVS(i, j) and the Root and Sink Horizontal Rule Vector Subgraphs

A HRVS(i, j) is a horizontal partition of the RVG from level i to j such that it repeats after every x levels ($x = (j+1) - i$).

The HRVS(i, j) derived from the RVG of the seven cell CA ⟨90 90 90 90 90 90 90⟩ is shown in Figure 4. The HRVS(i, j) ($i = 1$, $j = 2$) covers RVS(1) and RVS(2) and repeats after every two levels. The first HRVS covers level 1 to 2, while the second covers level 3 to 4.

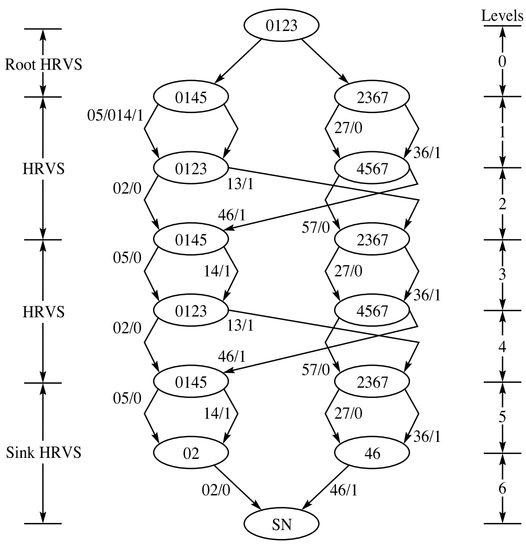

Figure 4. The RVG of a seven cell uniform CA with RV ⟨90 90 90 90 90 90 90⟩.

https://doi.org/10.25088/ComplexSystems.20.1.47

In general, the *root* HRVS covers level 0 to the level prior to the first HRVS, while the *sink* HRVS covers the levels from the last HRVS to level $(n-1)$. For the example CA of Figure 4, the root HRVS covers level 0, while the sink HRVS covers levels 5 and 6.

4. Size of an Invertible Cellular Automaton

The necessary and sufficient conditions for the RVG of an invertible CA (both hybrid and uniform) reported in Theorem 3 of [27] are reproduced below.

Theorem 1. Necessary and sufficient conditions for the RVG of an invertible CA are that no Type 1 or 2 nodes exist.

Theorem 1 leads to Theorem 2 stated next.

Theorem 2. The RVG of an invertible CA satisfies these conditions:

(a) For $i = 0$ to $(n-2)$ there are four RMTs in each node and the number of RMTs in the edge weight $\left\{T^i_{b_i}\right\} \Big/ b_i$ is two.

(b) For $i = (n-1)$, the number of RMTs in each input node is two and the number of RMTs in the edge weight is one.

Proof. The proof of necessity is given by contradiction. A RMT at the level i edge, as per Table 2, generates one odd and one even valued RMT at the level i output node, which is an $(i+1)$ level input node. The presence of a Type 1 node at the input of level i demands that either a 0- or 1-edge from the node is missing. If the edge is missing from a node having four RMTs, then the number of RMTs in the edge weight must be four, not two. On the other hand, if an output node at level i (having two outgoing edges) has k ($k \neq 4$) RMTs, the number of RMTs in at least one edge with weight $\{T^i_{b_i}\} \Big/ b_i$ must not be two.

Similarly, for the $(n-1)$ level having only even valued RMTs, the presence of a Type 1 node with a 0- or 1-edge missing demands the presence of two RMTs in the edge weight. Hence, the contradiction arises that ensures the absence of a Type 1 node if both stated conditions are true simultaneously.

The presence of a Type 2 node ensures that it is a merged node and the merging has occurred for two nodes V' and V ($V' \subset V$). If the number of RMTs in the edge weight of each edge is two with each node having four RMTs, then merging of two nodes V' and V, where V' is a subset of V ($V' \subset V$) cannot occur. The contradiction ensures that no Potential Type 2 node exists in the RVG. If there are no Potential Type 2 nodes, a Type 2 node cannot exist.

Proof of the sufficiency directly follows from the fact that the stated conditions (a) and (b) of the theorem lead to the results of Theorem 1. Hence the proof. □

https://doi.org/10.25088/ComplexSystems.20.1.47

Invertibility of a CA (hybrid or uniform) can be checked from the RVG employing the result of Theorems 1 or 2. Section 4.1 reports the analytical framework to identify the value of n of an n cell uniform CA that is invertible.

4.1 Invertible Cellular Automata

Lemmas 1 through 3 identify 10 CA rules that are invertible. The remaining 246 $(256 - 10)$ rules generate noninvertible n cell CAs for any value of n. The following terminologies are introduced to facilitate subsequent discussions in regards to the next state bit string (NSBS). Table 1 illustrates the next state bit b_i for each of the eight RMTs.

The NSBS $\langle b_7\, b_6\, b_5\, b_4\rangle$ for the RMTs T(7), T(6), T(5), T(4) is denoted as 1-NSBS since the most significant bit of each RMT has bit 1; the NSBS $\langle b_3\, b_2\, b_1\, b_0\rangle$ for the RMTs T(3), T(2), T(1), T(0) is denoted as 0-NSBS.

A 0- or 1-NSBS is marked as *balanced* if two 1s and two 0s exist in the string.

A 0-NSBS is the mirror image of its 1-NSBS and vice versa if $b_7 = b_0, b_6 = b_1, b_5 = b_2, b_4 = b_3$.

On the other hand, if $b_3 = b'_4$, $b_2 = b'_5$, $b_1 = b'_6$, $b_0 = b'_7$, where b'_j is the inverse of b_j $(j = 7, 6, 5, 4)$, $b_j = 0$ or 1, the 0-NSBS is the *complementary mirror image* of its 1-NSBS and vice versa.

A rule R_i is *complementary* to a rule R_j if both the 0- and 1-NSBSs of R_i have strings that are complementary to R_j.

For example, the rule 204 (1 1 0 0 1 1 0 0) is the complementary rule for 51 (0 0 1 1 0 0 1 1).

Lemma 1. If the RVG has no Type 1 or 2 nodes, the CA rule has balanced 0- and 1-NSBSs.

Proof. As per Theorem 2, each node at each level i $(i = 0$ to $(n-2))$ has four RMTs and the number of RMTs in the edge weight is two. For level $(n-1)$, the number of RMTs in the input node is two and the number of RMTs in the edge weight is one. The RMTs at the level i output node are derived out of the 0- and 1-RMTs on the level i edges. While the 0-RMTs have their next state bit 0, the 1-RMTs have their next state bit 1. Further, as per Table 2, a RMT on a level i edge generates one odd and one even valued RMT resulting in four RMTs (two odd and two even valued) in an output node. This situation can be true only if each of the 1- and 0-NSBSs of the rule are balanced strings with two 1s and two 0s. □

Lemma 2. There are only 18 CA rules whose RVG has no Type 1 or 2 nodes.

https://doi.org/10.25088/ComplexSystems.20.1.47

Proof. As per Lemma 1, the 0- and 1-NSBSs of the rule of an invertible CA should be balanced. Consequently, there can be ${}^4C_2 \times {}^4C_2 = 6 \times 6 = 36$ rules that satisfy the specified condition. The $(n-1)$ level input node has a pair of even valued RMTs derived out of T(0), T(2), T(4), T(6). In order to avoid Type 1 nodes at level $(n-1)$, the next state value for one RMT of the pair is 0, while for the other it is 1 or vice versa. Hence, out of 36 rules 50% are discarded. Hence, only $36/2 = 18$ rules exist whose RVG has no Type 1 or 2 nodes. □

The 18 rules that are probable candidates for generating invertible CAs can be divided into two classes.

Class A: (51, 204), (60, 195), (90, 165), (102, 153), (105, 150).

Class B: (54, 201), (57, 198), (99, 156), (108, 147).

The second rule in the pair (1^{st}, 2^{nd}) is complementary to the other. Consequently, the properties of the RVG of one rule are valid for the other with a 0- and 1-edge interchanged. The 0- and 1-NSBS of each rule of the Class A rule pairs is either a mirror image or a complementary mirror image of the other. While for Class B, the NSBS of each rule of the four rule pairs is neither a mirror image nor a complementary mirror image of the other. Figure 5 illustrates the RVG of two class B rules. Lemma 3 analyzes the Class B rules based on the HRVS.

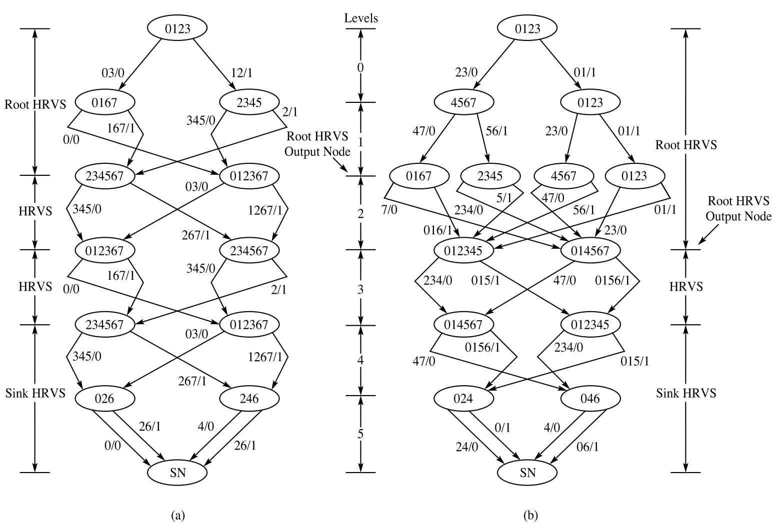

Figure 5. RVG of two Class B CA rules: (a) RV ⟨198 198 198 198 198 198⟩; (b) RV ⟨99 99 99 99 99 99⟩.

Lemma 3. The root HRVS of a CA with Class B rules has k RMTs $(k \neq 4)$ in a node and hence is marked as noninvertible.

https://doi.org/10.25088/ComplexSystems.20.1.47

Proof. The mirror or complementary mirror imaged 0- or 1-NSBS of a rule satisfies the conditions stated in Theorem 2. The 0- or 1-NSBS of each rule of the rule pairs of Class B is neither a mirror image nor a complementary mirror image of the other. As per Table 2, one odd and one even valued RMT is generated at the level i ($i = 0, 1, \ldots$) output node out of a RMT on the level i edge. As a result, merged nodes are generated in the root HRVS. Hence, the conditions noted in Theorem 2 get violated and the CA is noninvertible. □

The result of Lemma 3 is illustrated in the output nodes of the root HRVS that are merged nodes (Figure 6). While for rules 198 and 54 (Figure 5), merged nodes with six RMTs get generated at the output of level 1, for rules 99 and 147 it occurs at the output of level 2. Each merged node is a Potential Type 2 node with k RMTs, where $k \neq 4$. This violates the results of Theorem 2.

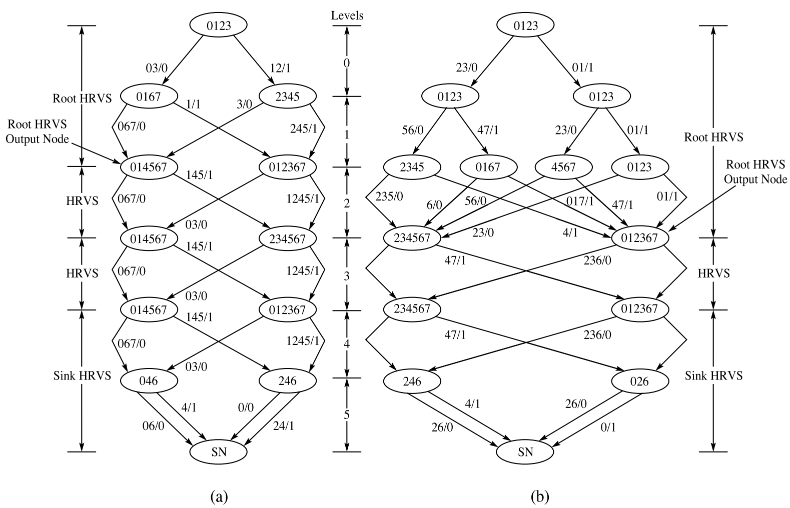

Figure 6. RVG of two Class B CA rules: (a) RV ⟨54 54 54 54 54 54⟩; (b) RV ⟨147 147 147 147 147 147⟩.

The earlier results and discussions can be summarized in the following theorem.

Theorem 3. The necessary and sufficient conditions for a rule to generate an invertible CA is that it has balanced 0- and 1-NSBSs with one NSBS as the mirror image or complementary mirror image of the other.

Proof. The proof directly follows from the results of Lemmas 1, 2, and 3. □

https://doi.org/10.25088/ComplexSystems.20.1.47

4.2 Size of Invertible Cellular Automata

Each of the rules in Class A (noted in Section 4.1) satisfies the condition of Theorem 3. These rules, as noted in Table 3, can be divided into six groups. An analysis of the HRVS of these six elementary rule groups leads to three different subgroups:

Subgroup 1: The rules 51, 60, 102, 153, 195, and 204.

Subgroup 2: The rules 90 and 165.

Subgroup 3: The rules 105 and 150.

Lemmas 4 through 7 characterize the HRVS for the rules of these subgroups. Such a characterization identifies the value of n of an n cell CA that is invertible.

Elementary Rule Group	CA Size for which the CA is Invertible
(51, 51, 51, 51) (51 = 00110011)	for all values of n
(60, 195, 102, 153) (60 = 00111100) (195 = 11000011) (102 = 01100110) (153 = 10011001)	for all values of n
(90, 165, 90, 165) (90 = 01011010) (165 = 10100101)	for even values of n
(105, 105, 105, 105) (105 = 01101001)	for all values of n excepting $n = 2 + 3y$ ($y = 0, 1, 2, 3 \ldots$)
(150, 150, 150, 150) (150 = 10010110)	for all values of n excepting $n = 2 + 3y$ ($y = 0, 1, 2, 3 \ldots$)
(204, 204, 204, 204) (204 = 11001100)	for all values of n

Table 3. The Class A invertible elementary rules with subgroup 1: (51, 60, 102, 153, 195, 204), subgroup 2: (90, 165), and subgroup 3: (105, 150).

Lemma 4. The nodes at a level i ($i = 0$ to $(n-2)$) of the RVG of an invertible CA have any one of these three node pair combinations:

1. {T(0), T(1), T(2), T(3)} and {T(4), T(5), T(6), T(7)};
2. {T(0), T(1), T(4), T(5)} and {T(2), T(3), T(6), T(7)}; and
3. {T(0), T(1), T(6), T(7)} and {T(2), T(3), T(4), T(5)}.

Proof. As per Theorem 2, the output nodes (with four RMTs) at any level i ($i = 0$ to $(n-2)$) are derived out of the RMT pair available in the edge weight at level i. (a) The 0- or 1-NSBS of invertible CA rules satisfy the necessary and sufficient conditions noted in Theorem 3. (b) As per Table 2, four RMTs of an output node are derived. Two of the four RMTs are even valued while the other two are odd valued.

In view of (a) and (b) only $2 \times 3 = 6$ possible nodes can exist: {T(0), T(1), T(2), T(3)}, {T(0), T(1), T(4), T(5)}, {T(0), T(1), T(6), T(7)}, {T(2), T(3), T(4), T(5)}, {T(2), T(3), T(6), T(7)}, and {T(4), T(5), T(6), T(7)}. A pair of such nodes cover all eight RMTs. Hence, for each level, a pair of such nodes are generated out of the three possible node pair combinations noted in the lemma. □

Lemma 5. The HRVS(i, j) in the RVG of each rule of Class A has length x where $x = (j+1) - i \leq 3$.

Proof. As per Lemma 4, only three different combinations are possible. So a node pair of HRVS(i, j) that appears at level i can reappear at level j, where $\text{Max}((j+1) - i) = 3$. Hence, $x \leq 3$. □

The length of HRVS(i, j) for each subgroup of Class A is specified in Lemmas 6 through 8 based on an analysis of the 0- and 1-NSBSs. In subsequent discussions, the symbol b_m ($m = 0, 1, 2, 3, 4, 5, 6, 7$) refers to the next state bit of RMT T(m).

Lemma 6. The length of the HRVS is 1 if the 0- and 1-NSBSs of a CA rule have the following properties:

1. one is the mirror image or complementary mirror image of the other; and
2. either $b_0 = b_1$ or $b_0 = b_3$.

Proof. For a rule with $b_0 = b_1$: Since the number of 1s in each NSBS is two and one is the mirror image or complementary mirror image of the other, we have $b_2 = b_3$, $b_4 = b_5$, and $b_6 = b_7$. Hence, level 0 output nodes are {T(0), T(1), T(2), T(3)} and {T(4), T(5), T(6), T(7)}. Also, the RMT pairs {T(0), T(1)}, {T(2), T(3)}, {T(4), T(5)}, and {T(6), T(7)} appear on level 1 edges. Consequently, as per Table 2, level 1 input nodes {T(0), T(1), T(2), T(3)} and {T(4), T(5), T(6), T(7)} reappear as its output node, that is, as the input node of level 2 (Figure 7). This situation continues for each level. Hence, the length of the HRVS is $2 - 1 = 1$.

For a rule with $b_0 = b_3$: For this case we have $b_1 = b_2$ with {T(0), T(1), T(6), T(7)} and {T(2), T(3), T(4), T(5)} as the level 0 output nodes. From the mirror image property we get $b_0 = b_7$, $b_1 = b_6$, $b_2 = b_5$, and $b_3 = b_4$. Consequently, the level 1 input nodes {T(0), T(1), T(6), T(7)} and {T(2), T(3), T(4), T(5)} reappear as its output nodes. Hence, the length of the HRVS is again 1. □

https://doi.org/10.25088/ComplexSystems.20.1.47

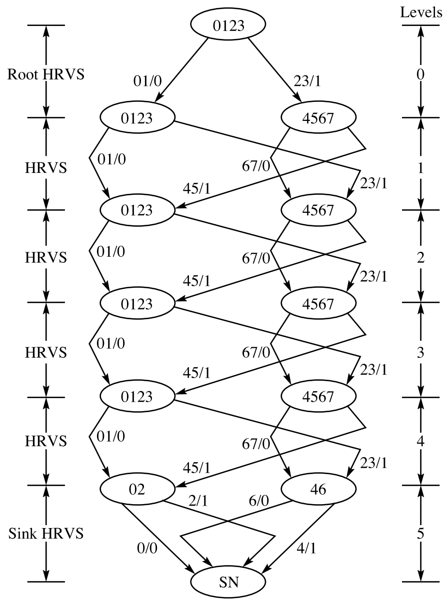

Figure 7. The RVG of a six cell uniform CA with RV ⟨60 60 60 60 60 60 60⟩.

Each of the rules of subgroup 1 (51, 60, 102, 153, 195, 204) (Table 3) satisfies the conditions of Lemma 6 and generate HRVSs of length 1. Hence, the next results follow.

Theorem 4. The CA with subgroup 1 rules (51, 60, 102, 153, 195, and 204) is invertible for all values of n.

Proof. As per Lemma 6, the length of the HRVS of each rule is 1. Also, the length of the root HRVS is 1. For the case $b_0 = b_1$, the node pair {T(0), T(1), T(2), T(3)} and {T(4), T(5), T(6), T(7)} repeats at each level i. Since for this case $b_0 \neq b_2$ and $b_4 \neq b_6$, no Type 1 nodes exist if level i becomes level $(n-1)$. Similarly, for the $b_0 = b_3$ case, the node pair {T(0), T(1), T(6), T(7)} and {T(2), T(3), T(4), T(5)} repeats at each level i. Since for this case $b_0 \neq b_6$ and $b_2 \neq b_4$, no Type 1 nodes exist if level i becomes level $(n-1)$. So there are no Type 1 or 2 nodes in any HRVS of the RVG of the rules covered by subgroup 1. Hence, such a rule generates an n cell invertible CA for any value of n. □

Lemma 7 characterizes subgroup 2 rules 90 and 165.

Lemma 7. The length of the HRVS is 2 if the 0- and 1-NSBSs of a CA rule have the following properties:

1. one is the mirror image of the other; and
2. $b_0 = b_2$.

https://doi.org/10.25088/ComplexSystems.20.1.47

Proof. With {T(0), T(1), T(2), T(3)} as the level 0 input node, the relation $b_0 = b_2$ generates {T(0), T(1), T(4), T(5)} and {T(2), T(3), T(6), T(7)} as the level 1 input nodes. Further, the mirror imaged 0- and 1-NSBSs lead to the following two situations:

S1: $b_0 = b_2, b_1 = b_3, b_5 = b_7, b_4 = b_6$;

S2: $b_0 = b_5, b_1 = b_4, b_2 = b_7, b_3 = b_6$ with $b_0 \neq b_4, b_2 \neq b_6$.

Either of these two situations appear in the RVG generated with an input node {T(0), T(1), T(4), T(5)} and {T(2), T(3), T(6), T(7)} at level 1. For each situation, the output node of a level i is one of the three node pair combinations noted in Lemma 4. The situation S2 leads to the level 1 output node pairs {T(0), T(1), T(2), T(3)} and {T(4), T(5), T(6), T(7)}, which are input node pairs of level 2. Next, the situation S1 with level 2 input node pair {T(0), T(1), T(2), T(3)} and {T(4), T(5), T(6), T(7)} generates the output nodes {T(0), T(1), T(4), T(5)} and {T(2), T(3), T(6), T(7)}. Thus, the level 1 input node pair reappears as the level 3 input nodes. This situation continues for any pair of levels i and $(i+1)$ $(i = 1, 3, 5, \ldots)$. Consequently, the level i input nodes reappear at level j, where $(j+1) - i = x = 2$. □

The rules 90 and 165 covered by subgroup 2 satisfy the properties noted in Lemma 7 and generate RVGs having HRVSs of length 2 (Figure 8). Hence, the next result follows.

Theorem 5. An n cell CA with rules of subgroup 2 (90 or 165) is invertible for even values of n.

Proof. The RVG of such a CA has root and sink HRVSs of length 1, while the length of each HRVS is 2. The input nodes at level i (for odd values of i) are {T(0), T(1), T(4), T(5)} and {T(2), T(3), T(6), T(7)}, while the $(i+1)$ level nodes are {T(0), T(1), T(2), T(3)} and {T(4), T(5), T(6), T(7)}. The situation repeats for two successive levels i and $(i+1)$ $(i = 1, 3, 5, \ldots)$. For an n cell CA with odd values of n, the $(n-1)$ level input nodes are {T(0), T(1), T(2), T(3)} and {T(4), T(5), T(6), T(7)}. Due to the null boundary, only even valued RMTs can exist at the $(n-1)$ level input nodes generating two nodes {T(0), T(2)} and {T(4), T(6)}. Since for such a CA $b_0 = b_2$ and $b_4 = b_6$ (situation S1, Lemma 7), the $(n-1)$ level input nodes are Type 1 nodes with either a 0- or 1-edge missing.

However, this condition is not true for even values of n, while the $(n-1)$ level input nodes are {T(0), T(1), T(4), T(5)} and {T(2), T(3), T(6), T(7)}. On deleting odd valued RMTs the input nodes at level $(n-1)$ are {T(0), T(4)} and {T(2), T(6)}. Because of the mirror image of the 0- and 1-NSBSs (situation S2, Lemma 7), $b_0 \neq b_4$ and also $b_2 \neq b_6$. Hence, the $(n-1)$ level input nodes are not Type 1 nodes. Hence, for all even values of n the CA is invertible. □

https://doi.org/10.25088/ComplexSystems.20.1.47

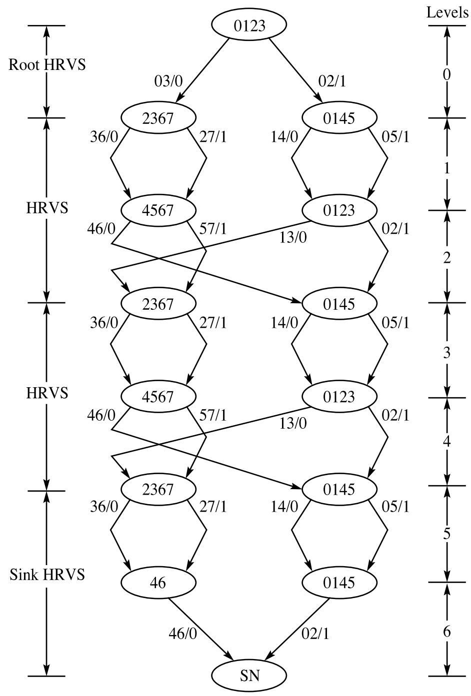

Figure 8. The RVG of a seven cell uniform CA with RV ⟨165 165 165 165 165 165 165⟩.

Lemma 8 characterizes the length of the HRVSs of subgroup 3 rules 105 and 150. The RVG of rule 150 is shown in Figure 9.

Lemma 8. The length of the HRVS is 3 if the 0- and 1-NSBSs of a CA rule have the following properties:

1. one is the complementary mirror image of the other; and
2. $b_0 = b_3$.

Proof. Let the level i input nodes of the RVG for the rules of this subgroup be {T(0), T(1), T(6), T(7)} and {T(2), T(3), T(4), T(5)}. These are also the output nodes of level 0 for rules 105 and 150. The balanced 0- and 1-NSBSs and the specified conditions of the lemma lead to the following situation:

S3: $b_0 = b_3$, $b_1 = b_2$, $b_4 = b_7$, $b_5 = b_6$ with $b_0 \neq b_2$, $b_4 \neq b_6$.

Situation S3, in turn, leads to one of the following two situations due to the complementary mirror imaging of the 0- and 1-NSBSs:

S4: $b_0 = b_5$, $b_1 = b_4$, $b_2 = b_7$, $b_3 = b_6$ with $b_0 \neq b_4$, $b_2 \neq b_6$.

S5: $b_0 = b_6$, $b_1 = b_7$, $b_2 = b_4$, $b_3 = b_5$.

https://doi.org/10.25088/ComplexSystems.20.1.47

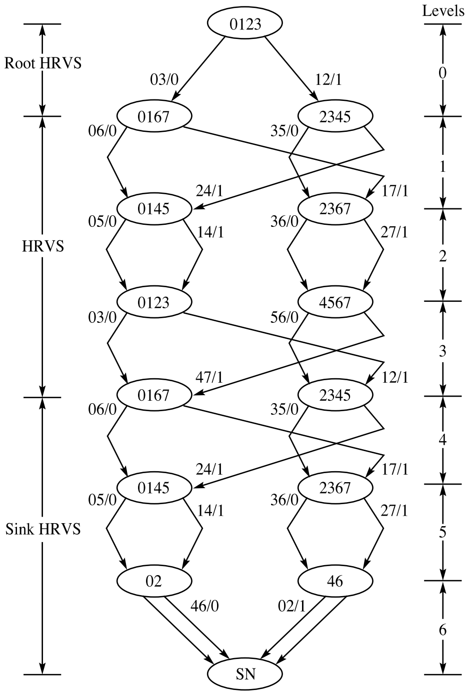

Figure 9. The RVG of a seven cell uniform CA with RV ⟨150 150 150 150 150 150 150⟩.

The situation S5 for level i input nodes {T(0), T(1), T(6), T(7)} and {T(2), T(3), T(4), T(5)} generate the output nodes {T(0), T(1), T(4), T(5)} and {T(2), T(3), T(6), T(7)}. With these nodes as the level $(i+1)$ input nodes, the output nodes due to situation S4 are {T(0), T(1), T(2), T(3)} and {T(4), T(5), T(6), T(7)}. These nodes are the input nodes for level $(i+2)$. Next, due to situation S3, level $(i+2)$ input nodes generate the output nodes {T(0), T(1), T(6), T(7)} and {T(2), T(3), T(4), T(5)}, which are the input nodes of level $(i+3)$ and identical to the level i input nodes. Thus, level i and level $(i+3)$ input nodes are identical with the length of the HRVS as $(i+3)-i=3$. □

Theorem 6. An n cell CA with rules of subgroup 3 (105 or 150) is invertible for all values of n excepting the cases where $n=2+3y$, where $y=0, 1, 2, \ldots$.

Proof. The input node pair {T(0), T(1), T(6), T(7)} and {T(2), T(3), T(4), T(5)} at level i generates the output nodes {T(0), T(1), T(4), T(5)} and {T(2), T(3), T(6), T(7)} which are the input nodes for level $(i+1)$. Consequently, the $(i+2)$ level input nodes are {T(0), T(1), T(2), T(3)} and {T(4), T(5), T(6), T(7)}.

https://doi.org/10.25088/ComplexSystems.20.1.47

The proof concentrates on analyzing each of the levels i, $(i+1)$, and $(i+2)$ constituting a HRVS and the associated situations S3, S4, and S5 noted in the proof of Lemma 8. For any level i with input nodes {T(0), T(1), T(6), T(7)} and {T(2), T(3), T(4), T(5)}, the situation S5 leads to the conditions $b_0 = b_6$ and $b_2 = b_4$. Consequently, if level i becomes the $(n-1)$ level, the nodes {T(0), T(6)} and {T(2), T(4)} (on deletion of odd valued RMTs) become Type 1 nodes with missing 0- or 1-edges. The length of the root HRVS is 1. So, an n cell CA with $n = 2$, $(2+x)$, $(2+2x)$, … , (x = length of HRVS = 3) is noninvertible due to the presence of the Type 1 node. Hence, for $n = 2 + 3y$ ($y = 0, 1, 2, \ldots$), the n cell CA is noninvertible.

The situation as noted for level i is not true if level $(i+1)$ or $(i+2)$ is considered as level $(n-1)$. For level $(i+1)$ with {T(0), T(1), T(4), T(5)} and {T(2), T(3), T(6), T(7)} as input nodes, situation S4 leads to the conditions $b_0 \neq b_4$, and $b_2 \neq b_6$. Consequently, if level $(i+1)$ becomes the $(n-1)$ level, the resulting input nodes {T(0), T(4)} and {T(2), T(6)} do not become Type 1 nodes since there are no missing 0- or 1-edges. Similarly, for level $(i+2)$ with {T(0), T(1), T(2), T(3)} and {T(4), T(5), T(6), T(7)} as input nodes, situation S3 leads to the conditions $b_0 \neq b_2$, and $b_4 \neq b_6$. Consequently, if level $(i+2)$ becomes the $(n-1)$ level, the resulting input nodes (with deletion of odd valued RMTs) {T(0), T(2)} and {T(4), T(6)} do not become Type 1 nodes. Considering all three cases, an n cell CA with a rule from subgroup 3 is invertible for all values of n other than $n = 2 + 3y$ ($y = 0, 1, 2, \ldots$). □

5. Conclusion

This paper solves the problem of identifying the value of n of an n cell three neighborhood null-boundary uniform cellular automaton (CA) that is invertible. The rule vector graph (RVG) of a CA derived out of its rule vector (RV) represents an efficient data structure to characterize CA evolution. The solution is based on an analysis of the subgraph referred to as a horizontal rule vector subgraph (HRVS). The HRVS is generated out of the RVG. The analytical framework of the HRVS presented in this paper identifies 10 CA rules that generate invertible CAs and specifies the value of n for which each of these rules generates an invertible CA.

References

[1] J. von Neumann, *Theory of Self-Reproducing Automata* (A. W. Burks, ed.), Urbana, IL: University of Illinois Press, 1966.

[2] S. Wolfram, *Theory and Application of Cellular Automata*, Singapore: World Scientific Publishing Company, 1986.

https://doi.org/10.25088/ComplexSystems.20.1.47

[3] M. Sipper, "Co-evolving Non-Uniform Cellular Automata to Perform Computations," *Physica D*, **92**(3–4), 1996 pp. 193–208. doi:10.1016/0167-2789(95)00286-3.

[4] K. Sutner, "Additive Automata on Graphs," *Complex Systems*, **2**(6), 1988 pp. 649–661.

[5] K. Sutner, "De Bruijn Graphs and Linear Cellular Automata," *Complex Systems*, **5**(1), 1991 pp. 19–30.

[6] T. Toffoli, "CAM: A High Performance Cellular Automata Machine," *Physica D*, **10**(1–2), 1984 pp. 195–204. doi:10.1016/0167-2789(84)90254-9.

[7] T. Toffoli and N. H. Margolus, "Invertible Cellular Automata: A Review," *Physica D*, **45**(1–3), 1990 pp. 229–253. doi:10.1016/0167-2789(90)90185-R.

[8] P. Pal Chaudhuri, D. R. Chowdhury, S. Nandi, and S. Chattopadhyay, *Additive Cellular Automata: Theory and Applications Volume 1*, Los Alamitos, CA: Wiley-IEEE Computer Society Press, 1997.

[9] P. Maji, C. Shaw, N. Ganguly, B. K. Sikdar, and P. Pal Chaudhuri, "Theory and Application of Cellular Automata for Pattern Classification," *Fundamenta Informaticae*, **58**(3–4), 2003 pp. 321–354.

[10] N. Ganguly, P. Maji, B. K. Sikdar, and P. Pal Chaudhuri, "Design and Characterization of Cellular Automata Based Associative Memory for Pattern Recognition," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, **34**(1), 2004 pp. 672–678. doi:10.1109/TSMCB.2002.806494.

[11] K. Cattell and J. C. Muzio, "Synthesis of One Dimensional Linear Hybrid Cellular Automata," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, **15**(3), 1996 pp. 325–335. doi:10.1109/43.489103.

[12] A. K. Das and P. Pal Chaudhuri, "Vector Space Theoretic Analysis of Additive Cellular Automata and Its Application for Pseudo-Exhaustive Test Pattern Generation," *IEEE Transactions on Computers*, **42**(3), 1993 pp. 340–352. doi:10.1109/12.210176.

[13] K. Cattell, J. C. Muzio, "Analysis of One Dimensional Linear Hybrid Cellular Automata over GF(q)," *IEEE Transactions on Computers*, **45**(7), 1996 pp. 782–792. doi:10.1109/12.508317.

[14] K. Paul, D. R. Chowdhury, and P. Pal Chaudhuri, "Theory of Extended Linear Machines," *IEEE Transactions on Computers*, **51**(9), 2002 pp. 1106–1110. doi:10.1109/TC.2002.1032629.

[15] B. K. Sikdar, N. Ganguly, and P. Pal Chaudhuri, "Fault Diagnosis of VLSI Circuits with Cellular Automata Based Pattern Classifier," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, **24**(7), 2005 pp. 1115–1131. doi:10.1109/TCAD.2005.850902.

[16] K. Cattell, S. Zhang, M. Serra, and J. C. Muzio, "2-by-n Hybrid Cellular Automata with Regular Configuration: Theory and Application," *IEEE Transactions on Computers*, **48**(3), 1999 pp. 285–295. doi:10.1109/12.754995.

[17] M. Serra. "Hybrid Cellular Automata." (Jan 8, 2009) http://webhome.cs.uvic.ca/~mserra/CA.html.

https://doi.org/10.25088/ComplexSystems.20.1.47

[18] S. Amoroso and Y. N Patt, "Decision Procedures for Surjectivity and Injectivity of Parallel Maps for Tessellation Structures," *Journal of Computer and System Sciences*, **6**(5), 1972 pp. 448–464. doi:10.1016/S0022-0000(72)80013-8.

[19] J. Kari, "Theory of Cellular Automata: A Survey," *Theoretical Computer Science*, **334**(1–3), 2005 pp. 3–33.

[20] J. Kari, "Reversibility of 2D Cellular Automata Is Undecidable," in *Cellular Automata: Theory and Experiment* (H. Gutowitz, ed.), Cambridge, MA: The MIT Press, 1991 pp. 379–385. doi:10.1016/j.tcs.2004.11.021.

[21] S. Das, B. K. Sikdar, and P. Pal Chaudhuri, "Characterization of Reachable/Non-reachable Cellular Automata States," in *Cellular Automata: 6th International Conference on Cellular Automata for Research and Industry Proceedings (ACRI '04)*, Amsterdam, The Netherlands (P. M. A. Sloot, B. Chopard, and A. G. Hoekstra, eds.), Berlin: Springer, 2004 pp. 813–822.

[22] N. S. Maiti, S. Munshi, and P. Pal Chaudhuri, "An Analytical Formulation for Cellular Automata (CA) Based Solution of Density Classification Task (DCT)," in *Cellular Automata: 7th International Conference on Cellular Automata for Research and Industry Proceedings (ACRI '06)*, Perpignan, France (S. El Yacoubi, B. Chopard, and S. Bandini, eds.), Berlin: Springer, 2006 pp. 147–156.

[23] S. Munshi, N. S. Maiti, D. Ray, D. R. Chowdhury, and P. Pal Chaudhuri, "An Analytical Framework for Characterizing Restricted Two Dimensional Cellular Automata Evolution," in *Proceedings of the 3rd Indian International Conference on Artificial Intelligence (IICAI '07)*, Pune, India (B. Prasad, ed.), 2007 pp. 1383–1402.

[24] K. Cattell and J. C. Muzio, "Synthesis of One Dimensional Linear Hybrid Cellular Automata," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, **15**(3), 1996 pp. 325–335. doi:10.1109/43.489103.

[25] Z. Kohavi, *Switching and Finite Automata Theory*, 2nd ed., New York: McGraw-Hill, 1978.

[26] S. Wolfram, *A New Kind of Science*, Champaign, IL: Wolfram Media, Inc., 2002.

[27] N. S. Maiti, S. Ghosh, S. Munshi, and P. Pal Chaudhuri, "Linear Time Algorithm for Identifying the Invertibility of Null-Boundary Three Neighborhood Cellular Automata," *Complex Systems*, **19**(1), 2010 pp. 89–113.

https://doi.org/10.25088/ComplexSystems.20.1.47