Applied Mathematical Modelling 40 (2016) 8017–8032

Contents lists available at ScienceDirect

Applied Mathematical Modelling

journal homepage: www.elsevier.com/locate/apm

On the irreversibility of Moore cellular automata over the ternary field and image application

Selman Uguz[a,*], Hasan Akin[b], Irfan Siap[c], Ugur Sahin[d]

[a] Department of Mathematics, Arts and Sciences Faculty Harran University, Şanliurfa 63120, Turkey
[b] Hasan Akin, Department of Mathematics, Faculty of Education, Zirve University, Gaziantep 27260, Turkey
[c] Irfan SIAP, Department of Mathematics, Arts and Science Faculty, Yıldız Technical University, Istanbul, Turkey
[d] Ugur Sahin, Rochester Institute of Technology, Multi Agent Biorobotic Laboratory, Rochester, NY, USA

ARTICLE INFO

Article history:
Received 12 July 2012
Revised 13 February 2016
Accepted 26 April 2016
Available online 7 May 2016

Keywords:
Two-dimensional cellular automata
Moore neighborhood
Ternary fields
Reversibility
Error correcting codes
Image processing

ABSTRACT

Cellular automata have rich computational properties and provide many models in mathematical and physical processes. In this paper, one of the most commonly used neighborhood types of two dimensional (2D) cellular automata which is called the Moore neighborhood in two dimensional integer lattice is considered. We study the characterization of 2D linear cellular automata defined by the Moore neighborhood with periodic and null boundary conditions over ternary fields. Furthermore, we analyze some results of 2D cellular automata defined by the rule number *9840NB* and finally also present applications to error correcting codes and image processing field.

© 2016 Elsevier Inc. All rights reserved.

1. Introduction

A cellular automaton (plural cellular automata, abbreviated to CA) is a discrete model studied and applied in many areas of science. Cellular automata (CAs) have rich computational properties and provide different amazing models in computation. CAs were first used for modeling various physical and biological processes and especially in computer science. The study of CAs has received remarkable attention in the last few years [1–7], because CAs have been widely investigated in many disciplines with different purposes (e.g., simulation of natural phenomena, pseudo-random number generation, image processing, analysis of universal model of computations, coding theory, cryptography).

CAs provide a good source in parallel processing which is used in coding, cryptography and network applications in general. For instance, CAs computations with secret key is discussed in [8]. CA is also used to design a symmetric key cryptography system based on Vernam cipher [8]. Furthermore, CA (especially hybrid CA) is used in generating pseudo-random numbers sequence (PNS) which are applied in the encryption processes. The quality of PNSs highly depends on the set of applied CA rules and the system is very resistant to attempts of breaking the cryptography key.

Most of the studies and applications for CA is done for one-dimensional (1D) CA. “The Game of Life” developed by John H. Conway in the 1960s is an example of a two-dimensional (2D) CA. John von Neumann in the late 40’s and early 50’s studied CA singular [9]. 2D CA with von Neumann neighborhood has found many applications and been explored in the literature [10,11]. Recently, 2D CA have attracted much of the interest [7,10,12,13]. Some basic and precise mathematical

* Corresponding author. Tel.: +90 5322217013.
E-mail addresses: selmanuguz@gmail.com (S. Uguz), hasanakin69@gmail.com (H. Akin), isiap@yildiz.edu.tr (I. Siap), useee@rit.edu (U. Sahin).

http://dx.doi.org/10.1016/j.apm.2016.04.027
S0307-904X(16)30228-1/© 2016 Elsevier Inc. All rights reserved.

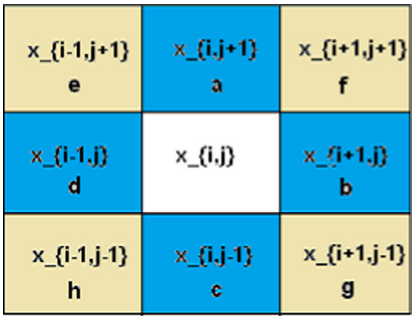

Fig. 1. 8 elements of the Moore neighborhood surround the center x_{ij}.

models using matrix algebra built on field $\mathbb{Z}_2$ were reported for characterizing the behavior of two-dimensional nearest neighborhood linear CAs with null or periodic boundary conditions [2–4,6,11,14,15].

In this paper, we investigate the characterization of 2D finite linear CA (shortly, 2D FLCA) with the Moore neighborhood by using matrix algebra built on $\mathbb{Z}_3 = \{0, 1, 2\}$ (The set of integers modulo 3). We present and study 2D linear CA with a new local rule called rule 9840 over ternary fields. This local rule is one of the important rules since it includes all possible neighbors of radius one. Here, we analyze some results about the rule numbers 9840NB (null boundary) and 9840PB (periodic boundary). We construct the rule matrix corresponding to the Moore CA. In order to compute the rank of the rule matrix the corresponding 2D finite Moore CA, by applying elementary row and column operations to the matrix, we obtain recurrences equations. In the present paper, the properties of two dimensional rules 9840NB and 9840PB over ternary fields are investigated and the dimension of the kernel is established via some recurrence relations. We examine whether 2D FLCA with the Moore neighborhood is reversible. Finally, we present two applications of this family to error correcting codes by extending the original method from binary fields to ternary fields and image processing area.

The present paper is organized as follows. Section 2 present basic technical details about 2D Moore CA over the ternary field $\mathbb{Z}_3$. Reversibility of mathematical model for Moore CA with null boundary case (Rule 9840NB) is given Section 3. In Section 4, an application of error correcting codes of CA is given in details. Application of 2D linear Moore CA in image processing (for the rules 9840NB, 9840PB) is presented in Section 5. Finally conclusions of the present study are briefly summarized in Section 6.

2. Two-dimensional CA over the field $\mathbb{Z}_3$

The 2D finite CA consists of $m \times n$ cells arranged in m rows and n columns, where each cell takes one of the values of the field $\mathbb{Z}_3$. From now on, we will denote 2D finite CA order to $m \times n$ by 2D CA$_{m \times n}$. A configuration of the system is an assignment of the states to all cells. Every configuration determines a next configuration via a linear transition rule that is local in the sense that the state of a cell at time $(t+1)$ depends only on the states of some of its neighbors at the time t using modulo 3 algebra. For 2D CA nearest neighbors, there are nine cells arranged in a 3×3 matrix centering that particular cell (see [2] for details).

2.1. Moore neighborhood

In 2D CAs theory, there are some classic types of neighborhoods, but in this paper we only restrict ourselves to the special neighbors which is called Moore neighborhood. The Moore neighborhood comprises the eight cells surrounding the central cell on a two-dimensional square lattice. It is one of the two most commonly used neighborhood types, the other one is the 4-cell von Neumann neighborhood (see [11]). The Moore neighborhood was used in the well known Conway's Game of Life. It is similar to the notion of 8-connected pixels in computer graphics [12,13,15].

In Fig. 1, we show the Moore neighborhood which comprises eight square cells which surround the center cell $x_{(i,j)}$. The state $x_{(i,j)}^{(t+1)}$ of the cell (i, j)th at time $(t+1)$ is defined by the local rule function $f : (\mathbb{Z}_3)^9 \to \mathbb{Z}_3$ as follows:

$$\begin{aligned} x_{(i,j)}^{(t+1)} &= f\left(x_{(i,j)}^{(t)}, x_{(i+1,j)}^{(t)}, x_{(i+1,j-1)}^{(t)}, x_{(i,j-1)}^{(t)}, x_{(i-1,j-1)}^{(t)}, x_{(i-1,j)}^{(t)}, x_{(i-1,j+1)}^{(t)}, x_{(i,j+1)}^{(t)}, x_{(i+1,j+1)}^{(t)}\right) \\ &= a_0 x_{(i,j)}^{(t)} + a_1 x_{(i+1,j)}^{(t)} + a_2 x_{(i+1,j-1)}^{(t)} + a_3 x_{(i,j-1)}^{(t)} + a_4 x_{(i-1,j-1)}^{(t)} \\ &\quad + a_5 x_{(i-1,j)}^{(t)} + a_6 x_{(i-1,j+1)}^{(t)} + a_7 x_{(i,j+1)}^{(t)} + a_8 x_{(i+1,j+1)}^{(t)} \pmod 3, \end{aligned} \tag{2.1}$$

where $a_0, a_1, \ldots, a_8 \in \mathbb{Z}_3$. The value of each cell for the next state may not depend upon all nine neighbors. Regarding the neighborhood of the boundary cells (see [3] for details), two approaches exist:

- If the boundary cells are connected to 0-state (see Table 1), then CA are called NB CA.
- If the boundary cells are adjacent to each other (see Table 3), then CA are called PB CA.

Table 1
Null boundary condition in a 2D finite $CA_{3\times3}$.

0	0	0	0	0
0	$x_{(i-1,j+1)}$	$x_{(i,j+1)}$	$x_{(i+1,j+1)}$	0
0	$x_{(i-1,j)}$	$x_{(i,j)}$	$x_{(i+1,j)}$	0
0	$x_{(i-1,j-1)}$	$x_{(i,j-1)}$	$x_{(i+1,j-1)}$	0
0	0	0	0	0

Table 2
Numbering of rule conventions with respect to neighbors over ternary field (i.e. modulo 3).

$a_6 = e \quad (3^6 = 729)$	$a_7 = a \quad (3^7 = 2187)$	$a_8 = f \quad (3^8 = 6561)$
$a_5 = d \quad (3^5 = 243)$	$a_0 \quad (3^0 = 1)$	$a_1 = b \quad (3^1 = 3)$
$a_4 = h \quad (3^4 = 81)$	$a_3 = c \quad (3^3 = 27)$	$a_2 = g \quad (3^2 = 9)$

Table 3
Periodic boundary condition in a 2D finite $CA_{3\times3}$.

$z_{(i+1,j-1)}$	$z_{(i-1,j-1)}$	$z_{(i,j-1)}$	$z_{(i+1,j-1)}$	$z_{(i-1,j-1)}$
$z_{(i+1,j+1)}$	$z_{(i-1,j+1)}$	$z_{(i,j+1)}$	$z_{(i+1,j+1)}$	$z_{(i-1,j+1)}$
$z_{(i+1,j)}$	$z_{(i-1,j)}$	$z_{(i,j)}$	$z_{(i+1,j)}$	$z_{(i-1,j)}$
$z_{(i+1,j-1)}$	$z_{(i-1,j-1)}$	$z_{(i,j-1)}$	$z_{(i+1,j-1)}$	$z_{(i-1,j-1)}$
$z_{(i+1,j+1)}$	$z_{(i-1,j+1)}$	$z_{(i,j+1)}$	$z_{(i+1,j+1)}$	$z_{(i-1,j+1)}$

If the same rule is applied to all of the cells in ever evaluation, then those CA are called uniform or regular. This paper deals with uniform CA with both null and periodic boundary conditions.

In general, the value of each cell for the next state may not depend upon all its nine neighbors. The sum of the neighboring cells on which each cell value is dependent is called the rule number of 2D CA over the field $\mathbb{Z}_3$. The conventional method of defining a rule number for a linear rule in 2D CA can be explained by Tables 1 and 2.

where $x_{(k,h)} \in \mathbb{Z}_3$. We can write the following table.

where for $i = 0, \dots, 8$, $a_i \in \mathbb{Z}_3^* = \mathbb{Z}_3/\{0\}$, that is $a, b, c, d, e, f, g, h \in \mathbb{Z}_3^*$. In Table 2, the center box represents the current cell and other boxes are eight neighborhoods of that cell. While extending and developing the theory for ternary field CA, we adopt similar notations as in binary CA in order to make the literature fluent and our paper more readable (see [3–5]). If the next state of a cell depends on the current state of itself and its left neighbor, it is referred to Rule 244 (=Rule 1 + Rule 243), since it is dependent only on two cells. If the next state depends on all its Moore neighbors, it is referred to $Rule\ 9840 = Rule\ 3^1 + Rule\ 3^2 + \cdots + Rule\ 3^8$ as seen in Table 2, since $9840 = 3^1 + 3^2 + \cdots + 3^8$. Hence we obtain that linear CA rule of Moore (Rule 9840) is summation of all nearest rules. All other linear rules can be obtained in the similar way. Periodic boundary condition in a 2D finite $CA_{3\times3}$ is obtained by means of the following table:

Now, we can characterize the Rule 9840 null boundary condition. In order to characterize the corresponding rule, first we represent each state matrix of size $m \times n$ as a column vector of size $mn \times 1$.

Hence, the problem of finding a rule matrix of the corresponding rule is taken from the space of $m \times n$ matrices to the space of $\mathbb{Z}_3^{mn}$. In order to accomplish this goal we define the following map

$$\Phi : \mathbb{M}_{m\times n}(\mathbb{Z}_3) \to \mathbb{M}_{mn\times 1}(\mathbb{Z}_3),$$

which takes the *tth* state $X^{(t)}$ given by

$$\begin{pmatrix} x_{11}^{(t)} & x_{12}^{(t)} & \cdots & x_{1n}^{(t)} \\ x_{21}^{(t)} & x_{22}^{(t)} & \cdots & x_{2n}^{(t)} \\ \vdots & \vdots & \cdots & \vdots \\ x_{m1}^{(t)} & x_{m2}^{(t)} & \cdots & x_{mn}^{(t)} \end{pmatrix} \longrightarrow X^{(t)} := \left(x_{11}^{(t)}, x_{12}^{(t)}, \dots, x_{1n}^{(t)}, \dots, x_{m1}^{(t)}, \dots, x_{mn}^{(t)}\right)^T, \tag{2.2}$$

where the superscript T denotes the transpose and $\mathbb{M}_{m\times n}(\mathbb{Z}_3)$ is the set of matrices with entries {0, 1, 2}.

Thus, local rules will be assumed to act on $\mathbb{Z}_3^{mn}$ rather than $\mathbb{M}_{m\times n}(\mathbb{Z}_3)$. The matrix

$$C^{(t)} = \begin{pmatrix} x_{11}^{(t)} & \cdots & x_{1n}^{(t)} \\ \vdots & \ddots & \vdots \\ x_{m1}^{(t)} & \cdots & x_{mn}^{(t)} \end{pmatrix},$$

is called the configuration matrix (or information matrix) of the 2-D finite CA at time t and $C^{(0)}$ is initial information matrix of the 2-D finite CA. Therefore, one can conclude that $\Phi(C^{(t)}) = X^{(t)}$.

Using the identification (2.2), we can define

$$(M_{9840NB})_{mn\times mn}X^{(t)} = X^{(t+1)}.$$

We do not include a detailed proof of the following theorem which gives the rule matrix of CA. The proof is obtained by determining the image of the basis elements of the space $\mathbb{Z}_3^{mn}$ under the CA. These images contribute to the columns of the rule matrix. We illustrate this in Example 2.3.

2.2. *Transition rule matrix for null boundary:* M_{9840NB}

Let $a, b, c, d, e, f, g, h \in \mathbb{Z}_3^* = \mathbb{Z}_3 \setminus \{0\}$, $m \geq 2$ and $n \geq 2$. Then, the rule matrix M_{9840NB} corresponding to the 2D finite Moore CA$_{m \times n}$ with NB is given by:

$$(M_{9840NB})_{mn\times mn} = \begin{pmatrix} S & P & 0 & \cdots & \cdots & 0 & 0 \\ K & S & P & \cdots & \cdots & 0 & 0 \\ 0 & K & S & \cdots & \cdots & 0 & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & \cdots & \cdots & 0 & K & S & P \\ 0 & 0 & \cdots & \cdots & 0 & K & S \end{pmatrix}, \tag{2.3}$$

where each submatrix is of order $n \times n$, and

$$S_{n\times n} = \begin{pmatrix} 0 & b & 0 & 0 & \cdots & 0 & 0 \\ d & 0 & b & 0 & \cdots & 0 & 0 \\ 0 & d & 0 & b & \cdots & 0 & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & \cdots & \cdots & 0 & d & 0 & b \\ 0 & 0 & \cdots & \cdots & 0 & d & 0 \end{pmatrix} \quad P_{n\times n} = \begin{pmatrix} c & g & 0 & 0 & \cdots & 0 & 0 \\ h & c & g & 0 & \cdots & 0 & 0 \\ 0 & h & c & g & \cdots & 0 & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & \cdots & \cdots & 0 & h & c & g \\ 0 & 0 & \cdots & \cdots & 0 & h & c \end{pmatrix}$$

$$K_{n\times n} = \begin{pmatrix} a & f & 0 & 0 & \cdots & 0 & 0 \\ e & a & f & 0 & \cdots & 0 & 0 \\ 0 & e & a & f & \cdots & 0 & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & \cdots & \cdots & 0 & e & a & f \\ 0 & 0 & \cdots & \cdots & 0 & e & a \end{pmatrix}. \tag{2.4}$$

Remark 2.1. In M_{9840NB}, if the submatrix $S_{n \times n}$ is considered as $S_{n \times n}(b, d)$, then the submatrices P and K can be written as $P_{n\times n} = cI_{n\times n} + S_{n\times n}(g, h)$, $\quad K_{n\times n} = aI_{n\times n} + S_{n\times n}(f, e)$. Hence the rule matrix of M_{9840NB} is expressed in terms of $S_{n \times n}(.,.)$ type submatrices.

Remark 2.2. If we just consider the von Neumann neighborhood CAs in Table 2, in other words when we choose $e = f = g = h = 0$ in our model, then the characterization of CA over ternary fields was given in [16].

Example 2.3. If we take $m = 3$ and $n = 3$, then we get the rule matrix M_{NB} of order 9. We consider a configuration of size 3×3 with null boundary:

$$\begin{matrix} 0 & 0 & 0 & 0 & 0 \\ 0 & x_{11} & x_{12} & x_{13} & 0 \\ 0 & x_{21} & x_{22} & x_{23} & 0. \\ 0 & x_{31} & x_{32} & x_{33} & 0 \\ 0 & 0 & 0 & 0 & 0 \end{matrix}$$

Now, we are going to apply rule 9840*NB*. We get a new configuration under this transformation which is

$$\begin{aligned}
y_{11} &= cx_{21} + bx_{12} + gx_{22} \\
y_{12} &= dx_{11} + cx_{22} + bx_{13} + gx_{23} + hx_{21} \\
y_{13} &= dx_{12} + cx_{23} + hx_{22} \\
y_{21} &= ax_{11} + cx_{31} + bx_{22} + fx_{12} + gx_{32} \\
y_{22} &= ax_{12} + dx_{21} + cx_{32} + bx_{23} + ex_{11} + fx_{13} + gx_{33} + hx_{31} \\
y_{23} &= ax_{13} + dx_{22} + cx_{33} + ex_{12} + hx_{32} \\
y_{31} &= ax_{21} + bx_{32} + fx_{22} \\
y_{32} &= ax_{22} + dx_{31} + bx_{33} + ex_{21} + fx_{23} \\
y_{33} &= ax_{23} + dx_{32} + ex_{22}.
\end{aligned}$$

Hence, we obtain the rule matrix M_{NB} of order 9 which is

$$M_{9840NB}=\left(\begin{array}{c|c|c} S_3(b,d) & cI+S_3(g,h) & 0_3 \\ \hline aI+S_3(f,e) & S_3(b,d) & cI+S_3(g,h) \\ \hline 0_3 & aI+S_3(f,e) & S_3(b,d) \end{array}\right)_{9\times 9},$$

where each submatrix is of order 3×3, and

$$S_{3\times3}(b,d)=\begin{pmatrix}0&b&0\\d&0&b\\0&d&0\end{pmatrix},\quad S_{3\times3}(g,h)=\begin{pmatrix}0&g&0\\h&0&g\\0&h&0\end{pmatrix},\quad S_{3\times3}(f,e)=\begin{pmatrix}0&f&0\\e&0&f\\0&e&0\end{pmatrix}.$$

2.3. Transition rule matrix for periodic boundary: M_{9840PB}

Let $a,b,c,d,e,f,g,h\in\mathbb{Z}_3^*$. For any $m>2$ and $n>2$ the rule matrix corresponding to the 2D finite Moore $\mathrm{CA}_{m\times n}$ with periodic boundary is given by:

$$(M_{9840PB})_{mn\times mn}=\begin{pmatrix} S^* & P^* & 0 & \cdots & \cdots & 0 & K^* \\ K^* & S^* & P^* & \cdots & \cdots & 0 & 0 \\ 0 & K^* & S^* & \cdots & \cdots & 0 & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & \cdots & \cdots & 0 & K^* & S^* & P^* \\ P^* & 0 & \cdots & \cdots & 0 & K^* & S^* \end{pmatrix}, \tag{2.5}$$

where each block submatrix is of order $n\times n$, and

$$S^*_{n\times n}=\begin{pmatrix} 0 & b & 0 & \cdots & \cdots & 0 & d \\ d & 0 & b & \cdots & \cdots & 0 & 0 \\ 0 & d & 0 & \cdots & \cdots & 0 & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & \cdots & \cdots & 0 & d & 0 & b \\ b & 0 & \cdots & \cdots & 0 & d & 0 \end{pmatrix}\quad {P^*}_{n\times n}=\begin{pmatrix} c & g & 0 & 0 & \cdots & 0 & h \\ h & c & g & 0 & \cdots & 0 & 0 \\ 0 & h & c & g & \cdots & 0 & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & \cdots & \cdots & 0 & h & c & g \\ g & 0 & \cdots & \cdots & 0 & h & c \end{pmatrix}$$

$${K^*}_{n\times n}=\begin{pmatrix} a & f & 0 & 0 & \cdots & 0 & e \\ e & a & f & 0 & \cdots & 0 & 0 \\ 0 & e & a & f & \cdots & 0 & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & \cdots & \cdots & 0 & e & a & f \\ f & 0 & \cdots & \cdots & 0 & e & a \end{pmatrix}. \tag{2.6}$$

Furthermore, if we define the new auxiliary matrices T_1^* and T_2^*, respectively, as follows:

$$(T_1^*)_{n\times n}=\begin{pmatrix} 0 & 1 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 1 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 0 & 1 & \cdots & 0 & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & \cdots & \cdots & \cdots & 0 & 0 & 1 \\ 1 & 0 & \cdots & \cdots & \cdots & 0 & 0 \end{pmatrix}, \tag{2.7}$$

and

$$(T_2^*)_{n\times n}=\begin{pmatrix} 0 & 0 & 0 & \cdots & \cdots & 0 & 1 \\ 1 & 0 & 0 & \cdots & \cdots & 0 & 0 \\ 0 & 1 & 0 & \cdots & \cdots & 0 & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & \cdots & \cdots & 0 & 1 & 0 & 0 \\ 0 & 0 & \cdots & \cdots & 0 & 1 & 0 \end{pmatrix}, \tag{2.8}$$

then $S^*=dT_2^*+bT_1^*$.

Remark 2.4. In M_{9840PB}, let us consider the submatrix $S^*_{n\times n}$ as $S^*_{n\times n}(b,d)$, and then the submatrices $P^*_{n\times n}$ and $K^*_{n\times n}$ can be written as

$$P^*_{n\times n}=cI_{n\times n}+S^*_{n\times n}(g,h),\quad K^*_{n\times n}=aI_{n\times n}+S^*_{n\times n}(f,e).$$

Hence the rule matrix of M_{9840PB} can be obtained in terms of $S^*_{n\times n}(.,.)$ type submatrices.

Example 2.5. In order to illustrate the previous theorem, we take $m = 3$ and $n = 3$ and consider a periodic boundary matrix of size 3×3.

$$\begin{array}{c|ccc|c} x_{33} & x_{31} & x_{32} & x_{33} & x_{31} \\ \hline x_{13} & x_{11} & x_{12} & x_{13} & x_{11} \\ x_{23} & x_{21} & x_{22} & x_{23} & x_{21} \\ x_{33} & x_{31} & x_{32} & x_{33} & x_{31} \\ \hline x_{13} & x_{11} & x_{12} & x_{13} & x_{11}. \end{array}$$

After the application of rule 9840*PB*, we get a new configuration under a transformation denoted by M_{PB} which is

$$\begin{aligned} y_{11} &= ax_{31} + dx_{13} + cx_{21} + bx_{12} + ex_{33} + fx_{32} + gx_{22} + hx_{23} \\ y_{12} &= ax_{32} + dx_{11} + cx_{22} + bx_{13} + ex_{31} + fx_{33} + gx_{23} + hx_{21} \\ y_{13} &= ax_{33} + dx_{12} + cx_{23} + bx_{11} + ex_{33} + fx_{31} + gx_{21} + hx_{22} \\ y_{21} &= ax_{11} + dx_{23} + cx_{31} + bx_{22} + ex_{13} + fx_{12} + gx_{32} + hx_{33} \\ y_{22} &= ax_{12} + dx_{21} + cx_{32} + bx_{23} + ex_{11} + fx_{13} + gx_{33} + hx_{31} \\ y_{23} &= ax_{13} + dx_{22} + cx_{33} + bx_{21} + ex_{12} + fx_{11} + gx_{31} + hx_{32} \\ y_{31} &= ax_{21} + dx_{33} + cx_{11} + bx_{32} + ex_{23} + fx_{22} + gx_{12} + hx_{13} \\ y_{32} &= ax_{22} + dx_{31} + cx_{12} + bx_{33} + ex_{21} + fx_{23} + gx_{13} + hx_{11} \\ y_{33} &= ax_{23} + dx_{32} + cx_{13} + bx_{31} + ex_{22} + fx_{21} + gx_{11} + hx_{12}. \end{aligned}$$

Hence, we get the rule matrix M_{PB} of order 9 as follows:

$$M_{9840PB} = \left(\begin{array}{c|c|c} S_3^*(b,d) & cI + S_3^*(g,h) & aI + S_3^*(f,e) \\ \hline aI + S_3^*(f,e) & S_3^*(b,d) & cI + S_3^*(g,h) \\ \hline cI + S_3^*(g,h) & aI + S_3^*(f,e) & S_3^*(b,d) \end{array}\right)_{9\times 9},$$

where each submatrix is of order 3×3, and

$$S_{3\times3}^*(b,d) = \begin{pmatrix} 0 & b & d \\ d & 0 & b \\ b & d & 0 \end{pmatrix}, \quad S_{3\times3}^*(g,h) = \begin{pmatrix} 0 & g & h \\ h & 0 & g \\ g & h & 0 \end{pmatrix}, \quad S_{3\times3}^*(f,e) = \begin{pmatrix} 0 & f & e \\ e & 0 & f \\ f & e & 0 \end{pmatrix}.$$

A special CA not included in the previous theorem is the CA $m = 2$ and $n = 2$ which gives the rule matrix M_{PB} of order 4. In order to determine this particular CA, we consider the following 2×2 matrix:

$$\begin{array}{c|cc|c} x_{22} & x_{21} & x_{22} & x_{21} \\ \hline x_{12} & x_{11} & x_{12} & x_{11} \\ x_{22} & x_{21} & x_{22} & x_{21} \\ \hline x_{12} & x_{11} & x_{12} & x_{11}. \end{array}$$

By applying rule 9840*PB*, we get a new configuration under a transformation denoted by M_{PB} which is given by

$$\begin{aligned} y_{11} &= (a+c)x_{21} + (b+d)x_{12} + (e+f+g+h)x_{22} \\ y_{12} &= (a+c)x_{22} + (b+d)x_{11} + (e+f+g+h)x_{21} \\ y_{21} &= (a+c)x_{11} + (b+d)x_{22} + (e+f+g+h)x_{12} \\ y_{22} &= (a+c)x_{12} + (b+d)x_{21} + (e+f+g+h)x_{11}. \end{aligned}$$

We get the rule matrix M_{PB} of order 4 as follows:

$$\begin{aligned} M_{9840PB} &= \left(\begin{array}{cc|cc} 0 & b+d & a+c & e+f+g+h \\ b+d & 0 & e+f+g+h & a+c \\ \hline a+c & e+f+g+h & 0 & b+d \\ e+f+g+h & a+c & b+d & 0 \end{array}\right)_{4\times4} \\ &= \begin{pmatrix} (b+d)I_2^* & (a+c)I_2 + (e+f+g+h)I_2^* \\ (a+c)I_2 + (e+f+g+h)I_2^* & (b+d)I_2^* \end{pmatrix}_{4\times4}, \end{aligned}$$

where $I_2^* = \left(\begin{smallmatrix} 0 & 1 \\ 1 & 0 \end{smallmatrix}\right)$. Hence, we have the following result.

Lemma 2.6. *Assuming the settings of rule matrices 9840PB with $m = 2$ and $n = 2$, we have*

$$(M_{9840PB})_{4\times4} = \begin{pmatrix} (b+d)I_2^* & (a+c)I_2 + (e+f+g+h)I_2^* \\ (a+c)I_2 + (e+f+g+h)I_2^* & (b+d)I_2^* \end{pmatrix}.$$

The remaining CA not considered above are listed below as special cases.

Lemma 2.7. *Assuming the settings of rule matrices 9840PB with $m \geq 3$ and $n = 2$, we have*

$$(M_{9840PB})_{m2\times m2} = \begin{pmatrix} (b+d)I^* & cI+(g+h)I^* & \cdots & 0 & aI+(e+f)I^* \\ aI+(e+f)I^* & (b+d)I^* & \cdots & 0 & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & \cdots & \cdots & (b+d)I^* & cI+(g+h)I^* \\ cI+(g+h)I^* & 0 & \cdots & aI+(e+f)I^* & (b+d)I^* \end{pmatrix}, \tag{2.9}$$

where I and I^* are matrices of size 2.

Lemma 2.8. *Assuming the settings of rule matrices 9840PB with $m = 2$ and $n \geq 3$, we have*

$$(M_{9840PB})_{2n\times 2n} = \begin{pmatrix} S^*(b,d) & (a+c)I+S^*(f+g,e+h) \\ (a+c)I+S^*(f+g,e+h) & S^*(b,d) \end{pmatrix}. \tag{2.10}$$

We note that having established all primary rules, by using linearity we can easily obtain the rule matrices of all other rules in a similar way.

3. Reversibility of Moore CA with null boundary

In this section, we characterize 2D linear CA (LCA) with NB determined by Moore rule (i.e. Rule 9840NB). For finite CA, in order to obtain the inverse of a 2D finite LCA many authors [3–5,16,17] have used the rule matrices. Since we already have found the rule matrix M_{9840NB} corresponding to the 2D finite CA, by using (2.3), we can state the following relation between the column vectors $X^{(t)}$ and the rule matrix M_{9840NB}:

$$X^{(t+1)} = M_{9840NB}X^{(t)} \pmod 3.$$

If the rule matrix M_{9840NB} is non-singular, then we have

$$X^{(t)} = (M_{9840NB})^{-1}X^{(t+1)} \pmod 3.$$

Thus, in this paper our main aim is to study whether the rule matrix M_{9840NB} in (2.3) is invertible or not. It is well known that the 2D finite CA is reversible if and only if its rule matrix M_{9840NB} is non-singular (see [3–5,16,17] for details). If the rule matrix M_{9840NB} has full rank, then it is invertible, so the 2D finite CA is reversible, otherwise it is irreversible.

Remark 3.1. For simplicity, we let

$$S_n(x,y,z) = \begin{pmatrix} y & x & 0 & \cdots & 0 & 0 \\ z & y & x & \cdots & 0 & 0 \\ 0 & z & y & x & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & \cdots & 0 & 0 & z & y \end{pmatrix}_{n\times n}.$$

Then the rule matrix becomes

$$(M_{9840NB})_{mn\times mn} = \left(\begin{array}{cccccc|c} S & P & 0 & \cdots & \cdots & 0 & 0 \\ \hline K & S & P & \cdots & \cdots & 0 & 0 \\ 0 & K & S & \cdots & \cdots & 0 & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & \cdots & \cdots & 0 & K & S & P \\ 0 & 0 & \cdots & \cdots & 0 & K & S \end{array}\right),$$

i.e. $S = S_n(b,0,d)$, $K = S_n(f,a,e)$ and $P = S_n(g,c,h)$.

The following theorem presents an algorithm for computing the rank of the rule matrix (2.3) which the case $K = S_n(f,a,e)$ is an invertible matrix.

Theorem 3.2. *If $K = S_n(f,a,e)$ is an invertible matrix, then the rank of the rule matrix (2.3) equals to $(m-1)n + rank(q_{2m-2})$ where*

$$q_0 = S, \quad q_1 = P$$
$$q_{2k} = -q_{2k-1}K^{-1}S + 2efq_{2k-2}, \quad \text{for} \quad k \geq 1$$
$$q_{2k+1} = -q_{2k-2}K^{-1}P, \quad \text{for} \quad k \geq 1.$$

Proof. For $m = 4$ we give a sketch of the proof. Let us consider the matrix

$$M_{9840NB} = \left(\begin{array}{ccc|c} S & P & 0 & 0 \\ \hline K & S & P & 0 \\ 0 & K & S & P \\ 0 & 0 & K & S \end{array}\right).$$

Since the matrix K is chosen as an invertible one, we multiply the second row block by $-SK^{-1} = -q_0K^{-1}$ and add it to the first row block. Then, we have

$$\approx \left(\begin{array}{ccc|c} 0 & q_2 = -SK^{-1}S + P & q_3 = -SK^{-1}P & 0 \\ \hline K & S & P & 0 \\ 0 & K & S & P \\ 0 & 0 & K & S \end{array}\right).$$

Next, multiply the third row block by $-(-SK^{-1}S + P)SK^{-1}$ and add it to the first row block. Then,

$$\approx \left(\begin{array}{ccc|c} 0 & 0 & q_4 = -(-SK^{-1}S + P)K^{-1}S - SK^{-1}P & q_5 = -(-SK^{-1}S + P)K^{-1}P \\ \hline K & S & P & 0 \\ 0 & K & S & P \\ 0 & 0 & K & S \end{array}\right)$$

$$= \left(\begin{array}{ccc|c} 0 & 0 & q_4 = -q_2K^{-1}S + q_3 & q_5 = -q_2K^{-1}P \\ \hline K & S & P & 0 \\ 0 & K & S & P \\ 0 & 0 & K & S \end{array}\right).$$

Finally, multiply the last row block by $-q_4K^{-1}$ and add it to the first row block. Then,

$$\approx \left(\begin{array}{ccc|c} 0 & 0 & 0 & -q_4K^{-1}S + q_5 = q_6 \\ \hline K & S & P & 0 \\ 0 & K & S & P \\ 0 & 0 & K & S \end{array}\right).$$

Since we consider that the matrix K is an invertible one, the sub matrix $\left(\begin{smallmatrix} K & S & P \\ 0 & K & S \\ 0 & 0 & K \end{smallmatrix}\right)$ has full rank, hence the desired result is obtained. □

Remark 3.3. Similar theorem can be written for the case P is an invertible matrix as stated in Theorem 3.2. In other words, if P is an invertible matrix, then by considering the transpose matrix of M_{9840NB} we easily see that the formula holds as in Theorem 3.2.

Now, the theorem does not say anything whenever both K and P are non-invertible. In order to determine this CA, we are going to study the problem of invertibility of these type of trigonal matrices.

For short we adopt $K_n = S_n(f, a, e)$. Let $K_1 = a, K_2 = \left(\begin{smallmatrix} a & f \\ e & a \end{smallmatrix}\right)$, and $K_3 = \left(\begin{smallmatrix} a & f & 0 \\ e & a & f \\ 0 & e & a \end{smallmatrix}\right)$, and so on. Here we compute the determinant of $K_n = S_n(f, a, e)$ as follows.

Theorem 3.4. *Let us consider the following trigonal matrix K_n,*

$$K_n = S_n(f, a, e) = \begin{pmatrix} a & f & 0 & \cdots & 0 & 0 \\ e & a & f & \cdots & 0 & 0 \\ 0 & e & a & f & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & \cdots & 0 & 0 & e & a \end{pmatrix}.$$

For $i \geq 0$, then we have the following determinant identities.

$$\begin{array}{ccc}
\det(K_{24i+1}) = |K_{24i+1}| = a & |K_{24i+9}| = 2efa & |K_{24i+17}| = 2a + efa \\
|K_{24i+2}| = 1 + 2ef & |K_{24i+10}| = 2 & |K_{24i+18}| = ef \\
|K_{24i+3}| = a + efa & |K_{24i+11}| = 0 & |K_{24i+19}| = 2efa + 2a \\
|K_{24i+4}| = 2 & |K_{24i+12}| = ef & |K_{24i+20}| = 1 + 2ef \\
|K_{24i+5}| = 2efa + a & |K_{24i+13}| = efa & |K_{24i+21}| = 2a \\
|K_{24i+6}| = 1 & |K_{24i+14}| = ef + 2 & |K_{24i+22}| = 2ef \\
|K_{24i+7}| = 2efa + 2a & |K_{24i+15}| = a + efa & |K_{24i+23}| = 0 \\
|K_{24i+8}| = ef + 2 & |K_{24i+16}| = 2ef & |K_{24i}| = 1.
\end{array}$$

Proof. By observing that $|K_1| = a,\ |K_2| = 1 + 2ef,\ |K_3| = a + aef,$ also

$$|K_4| = \begin{vmatrix} a & f & 0 & 0 \\ e & a & f & 0 \\ 0 & e & a & f \\ 0 & 0 & e & a \end{vmatrix} = a\begin{vmatrix} a & f & 0 \\ e & a & f \\ 0 & e & a \end{vmatrix} - f\begin{vmatrix} e & f & 0 \\ 0 & a & f \\ 0 & e & a \end{vmatrix} = a|K_3| + 2ef|K_2|.$$

Then in general we obtain the following identity

$$|K_n| = a|K_{n-1}| + 2ef|K_{n-2}| \quad for \quad n \geq 3.$$

Thus we get the desired result iteratively using the above identity. □

Remark 3.5. In the computations above theorem, we study the CA in the ternary case $\mathbb{Z}_3$. Since $a, e, f \neq 0$, the fact $a^2 = e^2 = f^2 = 1$ is used. It is also see from the theorem that some values of a, e, f, the determinant $|K_n|$ is zero. For example, let us choose $a = 1, e = 2, f = 1$; then $|K_{24i+3}| = a + efa = 0(mod\ 3)$. The open problem is finding the rank of M_R whenever K_i's are not invertible. So we have a partial answer so far. This will be studied later on.

Finally we obtain the following result for the reversibility of K_n over the ternary field $\mathbb{Z}_3$.

Lemma 3.6. *The trigonal matrix*

$$K_n = S_n(f, a, e) = \begin{pmatrix} a & f & 0 & \cdots & 0 & 0 \\ e & a & f & \cdots & 0 & 0 \\ 0 & e & a & f & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & \cdots & 0 & 0 & e & a \end{pmatrix}$$

is an invertible matrix for all values $a, e, f \in \mathbb{Z}_3^*$, *if* $n \equiv j \mod 24$, *where* $j \in \{0, 1, 4, 6, 9, 10, 12, 13, 16, 18, 21, 22\}$.

Remark 3.7. An important note that Lemma 3.6 states that K_n is an invertible matrix for some special n and any values of a, e, f. But of course some values n not listed in Lemma 3.6 and specific a, e, f; the trigonal matrix K_n can be a invertible one. For example, choose $n = 5$ and $a = 1, e = 2, f = 1$; then $|K_{24i+5}| = a + 2efa = 2(mod\ 3)$ which is different from zero. This shows that one can construct invertible K_n some special a, e, f and unless $n = 24i + 11, 24i + 23$ for $i \geq 0$ (see Theorem 3.4).

By using the result given in Lemma 3.6, we are able to determine the rank of the rule matrix given in (2.3) corresponding to two dimensional ternary CAs with null boundary. Periodic boundary case can be analyzed in the same way. Here we omitted this for giving two applications.

4. An application to error correcting codes

In this section, an application of this family of CAs to error correcting codes (ECC) is presented. In order to present the coding algorithm, we first introduce some well known basics on algebraic coding theory. Next, we present the encoding and decoding method which is a generalization of the original application over binary CA [18]. Finally, we build a bit error correcting code from this family of CA.

1D CAs that provide bit error correcting codes was first proposed by Chowdhury in 1994 [18]. Chowdhury et al. studied CA based ECC over binary fields. In this study we extend CA based bit error correcting codes (CA-ECC) over ternary fields. CA based bit error correcting codes have two main advantages if compared to standard linear coding. The first advantage is that without listing all syndromes as in the classical CA decoding is possible only in two steps. The second advantage is that the structure of CAs allows parallel computing which leads to faster computations and simpler hardware implementations.

It is well-known that $V = \mathbb{Z}_3^n$ is a $\mathbb{Z}_3$-vector space.

Definition 4.1. A subspace C of V is called a linear code of length n. The matrix with rows consisting of the basis of C is called a generator matrix of the code. The elements of C are called codewords.

Error correcting codes are applied in digital media. An information is encoded in order to be able to detect errors or even correct them. An information of length k is encoded to an n tuple regarded as an element (codeword) of C. After the transmission process if a linear code can detect and correct one error then the code is called one error correcting code. The number of nonzero entries of a vector v in V is called the Hamming weight of v. The smallest nonzero weight among all codewords of a linear code C is called the minimum Hamming weight of C. A linear code C of length n, dimension k and minimum Hamming weight d is represented by $[n, k, d]$. For a detailed information the reader can refer to [19]. These three parameters play an important role in error correcting codes. Especially the minimum Hamming weight of a code is very crucial.

Theorem 4.2 [19]. *If C is a linear code with minimum Hamming weight d, then C can correct up to* $\lfloor \frac{d-1}{2} \rfloor$ *errors.*

Let M_R be $n \times n$ nonsingular transition matrix of 2D Moore CA. Suppose that there exists $1 \le k \le s$, ($k, s \in \mathbb{Z}^+$ and $M_R^s = I_n$) such that $G = [I_n | M_R^k]$ (I_n, $n \times n$ identity matrix) generates a linear code that corrects at least one error.

First we present the encoding process and the decoding process via CA of a single bit error correcting code:

Encoding. Let $I = (i_1, i_2, \ldots, i_n) \in \mathbb{Z}_3^n$ be information vector to be encoded, where n is the rank of nonsingular transition matrix.

Then the information vector is encoded as the codeword $CW = (I, M_R^k[I]) = (i_1, i_2, \ldots, i_n, c_{n+1}, c_{n+2}, \ldots, c_{2n})$. ($C = M_R^k[I] = (c_{n+1}, c_{n+2}, \ldots, c_{2n})$ is the check vector).

Decoding. Let $CW' = (I', M_R^k[I]) = (i'_1, i'_2, \ldots, i'_n, c'_{n+1}, c'_{n+2}, \ldots, c'_{2n})$ be the received word, then we compute the syndrome vector as follows:

$S = 2M_R^k[I'] \oplus C'$.

Case 1. One error occurs in the information bits, then the syndrome of information part is:

$S_n = 2M_R^k[I'] \oplus C'$ and $M_R^{-k}[S_n]$ is the error vector of information part. In this CA the syndrome vector of check part S_c is all zero vector. Then the error vector is

$E = (M_R^{-k}[S_n], S_c) = (M_R^{-k}[S_n], 00\ldots0)$.

Case 2. If one error occurs in the check part, then the syndrome of the check part (at the same time error vector of check part) is

$S_c = M_R^k[I'] \oplus 2C'$. In this case, the syndrome vector of information part S_n is the all zero vector, then the error vector is

$E = (M_R^{-k}[S_n], S_c) = (00\ldots0, M_R^k[I'] \oplus 2C')$.

Now we present an example of this family CA:

Example 4.3. Let $a = c = 2$ and $b = d = e = g = f = 1 \in \mathbb{Z}_3^*$, then the rule matrix of M_{9840PB} is

$$M_{9840PB} = \left(\begin{array}{ccc|ccc|ccc} 0 & 1 & 1 & 2 & 1 & 1 & 2 & 1 & 1 \\ 1 & 0 & 1 & 1 & 2 & 1 & 1 & 2 & 1 \\ 1 & 1 & 0 & 1 & 1 & 2 & 1 & 1 & 2 \\ \hline 2 & 1 & 1 & 0 & 1 & 1 & 2 & 1 & 1 \\ 1 & 2 & 1 & 1 & 0 & 1 & 1 & 2 & 1 \\ 1 & 1 & 2 & 1 & 1 & 0 & 1 & 1 & 2 \\ \hline 2 & 1 & 1 & 2 & 1 & 1 & 0 & 1 & 1 \\ 1 & 2 & 1 & 1 & 2 & 1 & 1 & 0 & 1 \\ 1 & 1 & 2 & 1 & 1 & 2 & 1 & 1 & 0 \end{array}\right)_{9\times 9}.$$

For $k = 2$, the matrix $G = \left[I_9 | M_R^2\right]$ generates a $[18, 9, 4]_3 - code$ with $d(C) = 4$. Hence this code can correct only one error.

$$I = 111111111 \Rightarrow C = M_R^2[I] = 111111111 \Rightarrow CW = 111111111111111111.$$

Case 1. One error occurs in the information bits. Let the received word be $CW' = 1\hat{0}1111111111111111 = \left(I' | C'\right)$. Then the syndromes are computed as:

$$S = 2M_R^2\left[I'\right] \oplus C' = 111101101 \oplus 111111111 = 222212212$$
$$S_9 = S_9 \oplus S_c = 222212212.$$

Therefore, the error pattern of information part is

$$I_e = M_R^{-2}[S_9] = 010000000.$$

The correct information and check bits are computed as:

$$I = I' \oplus I_e = 101111111 \oplus 010000000 = 111111111$$
$$C = C' = 111111111.$$

Then, the error vector is $E = 010000000000000000$.

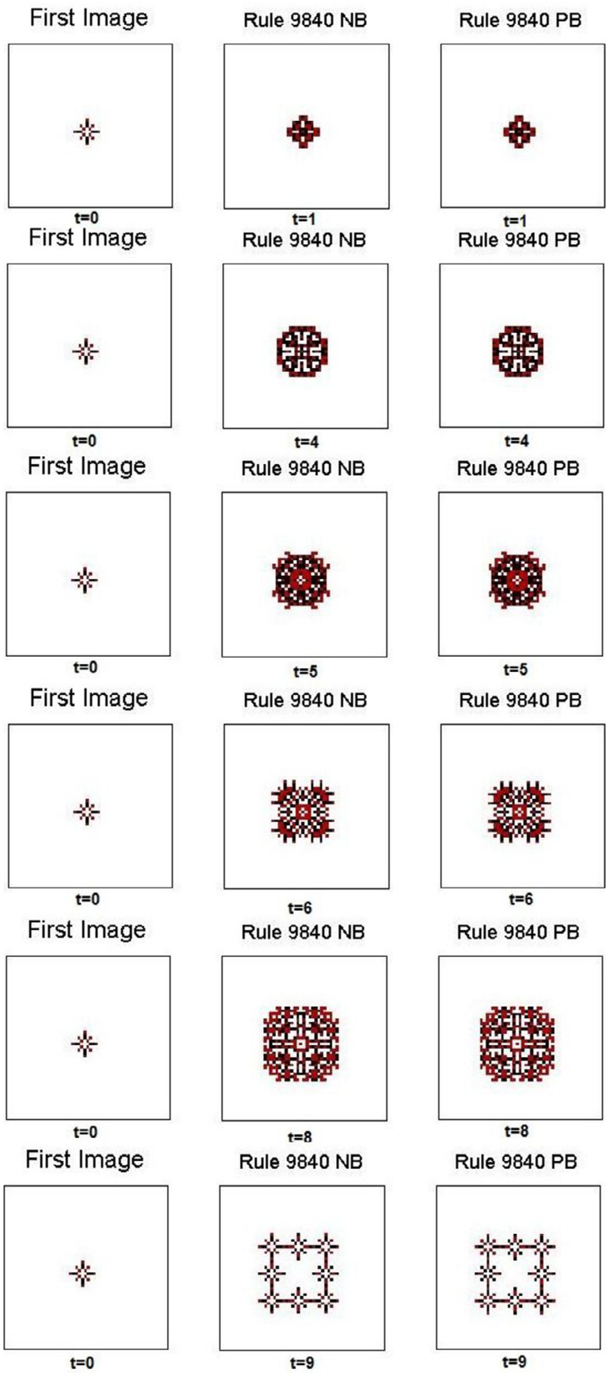

Fig. 2. Application of Moore (Rule 9840) rule over null and periodic boundaries (NB, PB) after 9 iterations of the first image.

Case 2. One error occur in the check part. Let the received word be $CW' = 1111111111111111\widehat{0}1 = \left(I'|C'\right)$. Then the syndromes are computed as:

$$S_c = M_R^2\left[I'\right] \oplus 2C' = 111111111 \oplus 222222202 = 000000010$$
$$S_9 = 000000000.$$

The error pattern of information and check parts are

$$I_e = M_R^2[S_9] = 000000000$$
$$C_e = S_c = 000000010.$$

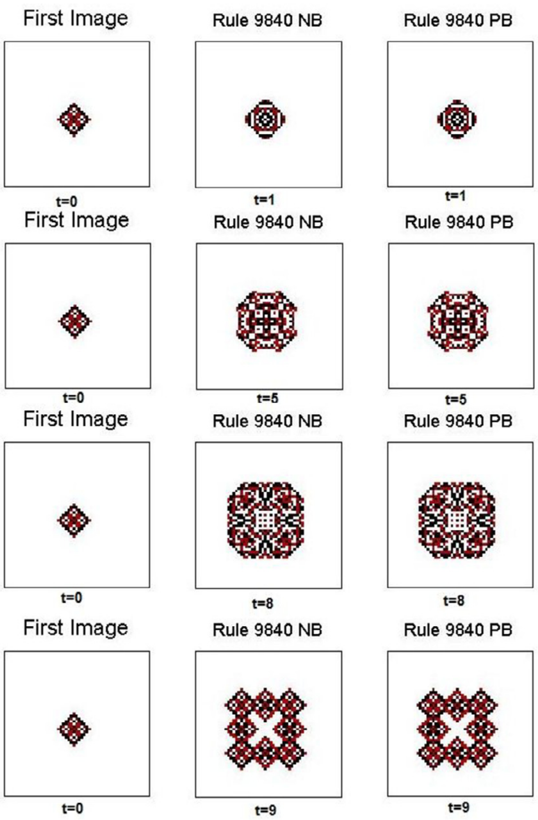

Fig. 3. Application of Moore (Rule 9840) rule over null and periodic boundaries (NB, PB) after 9 iterations of the first image.

respectively. The correct check bits are computed as:

$$C = C' \oplus C_e = 111111101 \oplus 000000010 = 111111111.$$

Hence the error vector is $E = 000000000000000010$.

5. An application to image processing area

It is known that cellular automata (CA) can be applied to several basic image processing problems due to their importance of lattice structure [7]. It is also known that the advantage of using CA is its simplicity of structure that enables fast implementation on various kinds of devices such as VLSI, FPGA and GPU devices. Hence it is seen that CA should be well-suited theory for image processing areas and applications. The typical pattern generation in the sense of image processing algorithm models using 2D CA can be shown in Figs. 2–6. These algorithms given as a matrix rules in the preceding section can be used for several applications in addition to edge detection, including noise suppression, thinning, and convex hull construction [7,12,15,20–23]. As mentioned with details in the introduction, CA provides simple discrete mathematical models for many physical, biological, and computational systems. Even though it has simple construction, CA are shown to be capable of complicated behavior and to generate complex patterns with some universal features (see Figs. 2–6). From the simple initial configurations, it will be seen that the generated patterns could be self replicating [21] (Figs. 2–6).

Generation of self replicating patterns are one of the interesting topics and research areas in non-linear science and technology. Pattern generation study is an important process of transforming copies of the seed image on the array (1D), plane (2D) or space (3D) to create the complete repeating pattern model with no overlaps and blank [24–26]. These generated patterns have some significant mathematical properties that give a way generating algorithm possible (such as 9-iteration in Figs. 2–6). Hence a cellular automaton present a good algorithmic approach used for pattern generation models in the literature.

Self replicating pattern programs begin with John von Neumann's studies [27]. Though it is not apparent at the first glance, cellular automata exhibit a wide variety of behaviors, i.e. self organization, chaotic behavior, pattern generation and fractals. Here one can notice from the visual appearance of the figures, the cases of self organization, pattern formation, and fractals are observed (see Rule 9840 in Figs. 2–5). Automaton program devised by von Neumann is an example of trivial self replication program. These kind of self replication procedures are easily implemented in many different programming

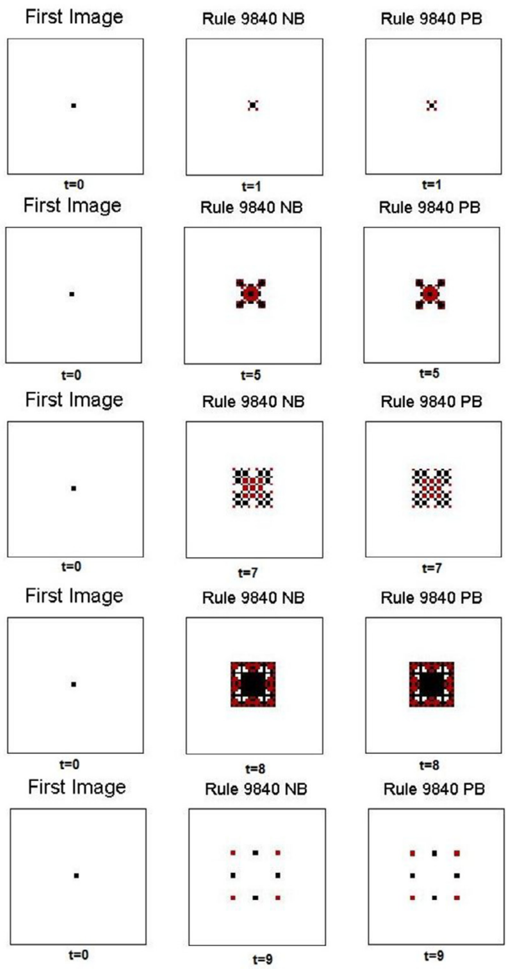

Fig. 4. Application of Moore (Rule 9840) rule over null and periodic boundaries (NB, PB) after 9 iterations of the first image.

language. In mathematics fractals are typically self replicating patterns, where self replicating means they are the same from near as from far. Let us look closely at these types of patterns in Figs. 2–6.

Producing algorithmic approach for self replicating patterns of digital images is sometimes difficult procedure. However one can face with many difficulties in producing and developing tiling algorithms, i.e. providing simple and applicable algorithm to designate very complicated complex patterns model. Evolution from simple seed image in 2D cellular automata may provide self replicating patterns with complicated boundaries (null and periodic). The approach mentioned here give to accurate algorithm for producing many different patterns.

5.1. Moore (9840NB and 9840PB) CA over ternary field $\mathbb{Z}_3$

In the present paper we use uniform 2D linear CA with the nearest eight neighborhoods (see Fig. 1) to generate self replicating patterns of digital images. After constructing 2D linear CA rules with null and periodic boundaries in image processing process, we take a ternary matrix of size 100×100. Note that the size of image matrix can be chosen to be larger, but due to the computation limits the sizes are reasonable, we encode each element of the matrix to a unique pixel on the screen (using MATLAB codes), and it is colored a pixel *white for 0, red for 1, black for 2* for the each matrix elements.

Also we choose another initial image whose size is less than (30×30) for which pattern is to be generated and put it in the center of the ternary matrix of size 100×100. Here we called the initial image within the first image as a *seed image*. This is the way, how the image is put within the ternary area of (100×100) matrix pixels. To compare with respect to the different boundary conditions, it is applied null and periodic boundary cases to the image. After then it is applied 2D linear CA rules on (100×100) image matrix and each time the rule is implemented using the varied image matrix and a new image is obtained again. In the present model, there exists three states, i.e. black, white and red, are used to represent the state of cells. However the pattern can be seen as the developing and evolving black, red and white patterns. Hence it can be given different colors to the patterns by using varied type graphics programme packages. From the figures, it is seen that the rules other than the basic fundamental rules produce many alternative patterns of the given seed image. Figs. 2–6 shows

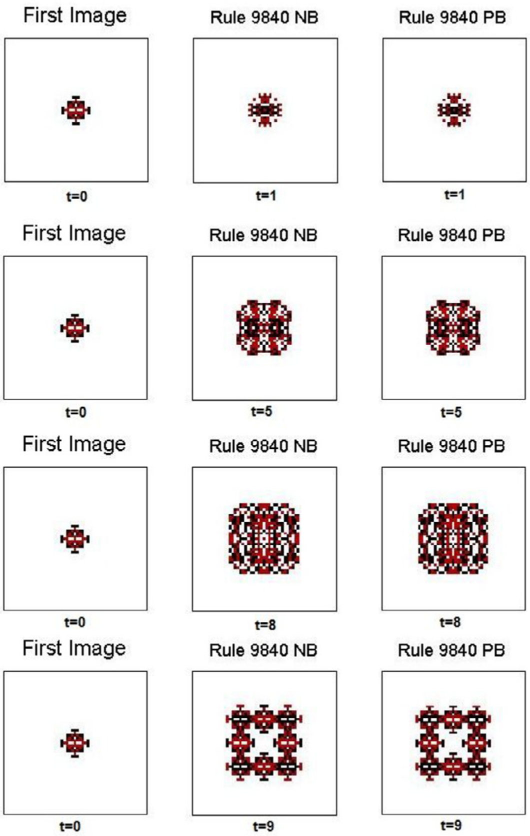

Fig. 5. Application of Moore (Rule 9840) rule over null and periodic boundaries (NB, PB) after 9 iterations of the first image.

some typical examples of patterns generated by the evolution of 2D linear cellular automata from initial image containing a single nonzero seed image.

This is the case in which the seed image is not exceeding 30 percentage of the display matrix. One can seen from the figures that the self replicating and self similar patterns could generate only when number of repetition is $t = 9$. This gives how the image is put inside in area of (100×100) matrix pixels. Here we apply 2D linear CA rules with null and periodic boundaries on (100×100) image matrix and every time the rule is implemented to obtain a new image. Hence we get the Figs. 2–6.

Remark 5.1. From observation of the Figs. 2–6, it can be seen that the boundary conditions has not enough effect (i.e. the evolving images are all same with respect to null and periodic boundaries) for the special case seed image smaller than 30 percentage of the displayed matrix. When we take big number of iterations and big seed image larger than 30 percentage of the display matrix should be considered for the next works. Our forthcoming efforts will be this direction.

Remark 5.2. It is seen that Fig. 5 is quite interesting to comment about the replication of seed image. It seems as emergence of fractal structure after some fixed iterations. An important note that the 8-fold replicated pattern at $t = 9$ Fig 5 can be produced by using a seed pattern for further replication on a larger lattice, just considered for the case seed image smaller than 30 percentage of the displayed matrix. Then we can conjecture that the same fixed number of replicates would be achieved again in the same number of iterations (i.e. $t = 9$ iterations). This is an important result of the present paper and it will need an extra research effort to reveal why only $t = 3^2 = 9$ times repetition of the rules produce the self replicating 8-fold fractal patterns of any arbitrary chosen seed image. This would need to be the case if a fractal-like structure were to emerge in the limiting case of an infinite lattice.

6. Conclusions

The work on understanding the behavior of 2D CA over ternary fields and studying their structure presented here is an ongoing research. Some properties of the rule numbers 9840NB and 9840PB are established in this paper. Theorem 3.2 depends on computation of q_{2m-2} which is important. Hence, an easier way of computing the rank of q_{2m-2} is an important issue. Further properties of these rules and properties of different number rules with respect to different neighboring relations are still being investigated by the authors. Similar computations and explorations of CA over different rings (see

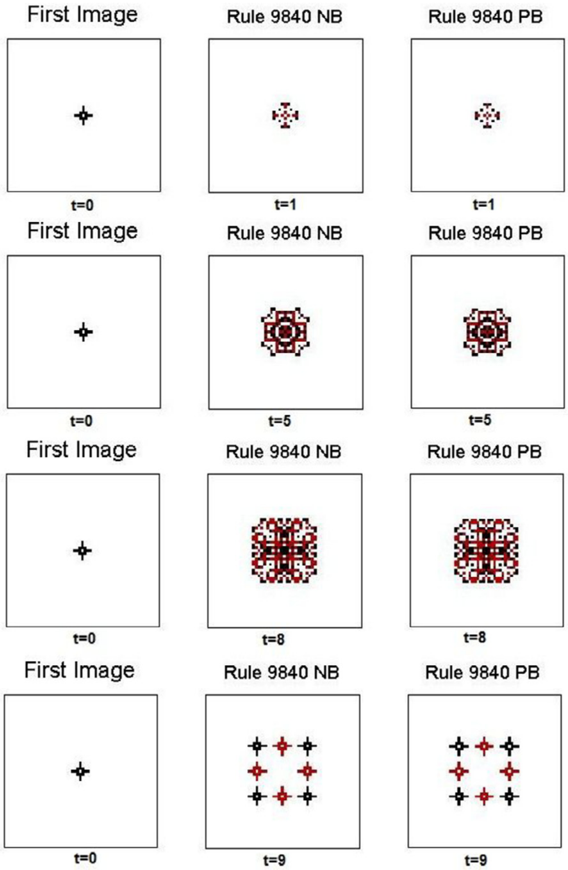

Fig. 6. Application of Moore (Rule 9840) rule over null and periodic boundaries (NB, PB) after 9 iterations of the first image.

[28] for details) remain to be of interest. The number of iterations taken for the 2D CA to create perfect non-overlapped copies of a shape depends on the size of the seed image shape. It is provided that maximum length of the image arranged in all directions not exceeding 30 percentage of row or column lengths (due to the rule matrix is a square matrix) of the display matrix (present case is 100). It can be seen that extra research effort is needed to reveal why only $t = 3^2 = 9$ times repetition of the rules produce the self replicating patterns of any arbitrary chosen seed image. Finally the hybrid cellular automaton (HCA) [29] and CA with memory [24–26] are also applied for Moore and different neighborhood configurations. In the present study, examples that connects CA with error correcting codes and image processing sample are presented. Further applications remain to be of great interest for future studies [12,13,30–36].

Acknowledgments

The authors thank the associate editor and the anonymous referees for their valuable remarks (especially Remark 5.2 for the associate editor) that led to the improvement of the presentation of the paper and give some directions for the future research. The work is supported by The Scientific and Technological Research Council of Turkey (TUBITAK), Project Number: 110T713. The first author (SU) thanks Prof. Dr. Ferat Sahin and Dr. Ugur Sahin for kind hospitality during the visit (supported by HUBAK Project Number: 13139) of the Rochester Institute of Technology (RIT).

References

[1] H. Akın, On the directional entropy of $\mathbb{Z}^2$-actions generated by additive cellular automata, Appl. Math. Comp. 170 (1) (2005) 339–346.
[2] P. Chattopadhyay, P.P. Choudhury, K. Dihidar, Characterization of a particular hybrid transformation of two-dimensional cellular automata, Comp. Math. Appl. 38 (1999) 207–216.
[3] A.R. Khan, P.P. Choudhury, K. Dihidar, S. Mitra, P. Sarkar, VLSI architecture of a cellular automata machine, Comp. Math. Appl. 33 (5) (1997) 79–94.
[4] A.R. Khan, P.P. Choudhury, K. Dihidar, R. Verma, Text compression using two dimensional cellular automata, Comp. Math. Appl. 37 (6) (1999) 115–127.
[5] Z. Ying, Y. Zhong, D. Pei-min, On behavior of two-dimensional cellular automata with an exceptional rule, Inf. Sci. 179 (5) (2009) 613–622.
[6] K. Dihidar, P.P. Choudhury, Matrix algebraic formulae concerning some special rules of two-dimensional cellular automata, Inform. Sci. 165 (2004) 91–101.
[7] U. Sahin, S. Uguz, H. Akin, The transition rules of 2d linear cellular automata over ternary field and self-replicating patterns, Int. J. Bifurc. Chaos 25 (2015) 1550011.

[8] F. Seredynski, P. Bouvry, A.Y. Zomaya, Cellular automata computations and secret key cryptography, Neural Parallel Comput. 30 (2004) 753–766.
[9] J. von Neumann, Collected Works, Design of Computers Theory of Automata and Numerical Analysis, vol. 5, Pergamon Press, 1951.
[10] U. Sahin, S. Uguz, H. Akin, I. Siap, Three-state von neumann cellular automata and pattern generation, Appl. Math. Model. 39 (2015) 2003–2024.
[11] N.Y. Soma, J.P. Melo, On irreversibility of von neumann additive cellular automata on grids, Discrete Appl. Math. 154 (5) (2006) 861–866.
[12] U. Sahin, S. Uguz, F. Sahin, Salt and pepper noise filtering with fuzzy-cellular automata, Comput. Electr. Eng. 40 (2014) 59–69.
[13] S. Uguz, U. Sahin, F. Sahin, Edge detection with fuzzy cellular automata transition function optimized by PSO, Comput. Electr. Eng. 43 (2014) 180–192.
[14] I. Siap, H. Akın, S. Uguz, 2d reversible cellular automata with nearest and prolonged next nearest neighborhoods, Eur. J. Pure App. Math. 6 (11) (2013) 315–334.
[15] S. Uguz, U. Sahin, I. Siap, H. Akin, 2d cellular automata with an image processing application, Acta Phys. Polonica A 125 (2014) 435–438.
[16] I. Siap, H. Akin, F. Sah, Characterization of two dimensional cellular automata over ternary fields, J. Frankl. Inst. 348 (2011a) 1258–1275.
[17] I. Siap, H. Akın, S. Uguz, Structure and reversibility of 2-dimensional hexagonal cellular automata, Comput. Math. Appl. 62 (11) (2011b) 4161–4169.
[18] D.R. Chowdhury, S. Basu, I.S. Gupta, P.P. Chaudhuri, Design of CACC-cellular automata based error correcting code, IEEE Trans. Comput. 43 (1994) 759–764.
[19] F.J. MacWilliams, N.J. Sloane, The Theory Of ErrorCorrecting Codes, North-Holland Pub. Co., 1977.
[20] H. Akın, S. Uguz, I. Siap, Characterization of 2d cellular automata with Moore neighborhood over ternary fields, AIP Conf. Proc. 1389 (2011) 2008–2011.
[21] S. Uguz, U. Sahin, H. Akın, I. Siap, Self-replicating patterns in 2d linear cellular automata, Int. J. Bifurc. Chaos 24 (2014) 1430002.
[22] S. Uguz, U. Sahin, F. Sahin, Uniform cellular automata linear rules for edge detection, in: Proceedings of the IEEE International Conference Systems, Man, and Cybernetics (SMC), 2013, 2013, pp. 2945–2950.
[23] U. Sahin, F. Sahin, S. Uguz, Hybridized fuzzy cellular automata thresholding algorithm for edge detection optimized by PSO, in: Proceedings of the 10th International Conference IEEE High Capacity Optical Networks and Enabling Technologies (HONET-CNS), 2013, pp. 228–232.
[24] R. Alonso-Sanz, Cellular Automata with Memory, Old City Publishing, 2009.
[25] R. Alonso-Sanz, Discrete Systems with Memory, World Scientific Publishing, 2011.
[26] R. Alonso-Sanz, Reversible cellular automata with memory: two-dimensional patterns from a single site seed, Phys. D: Nonlinear Phenom. 175 (2003) 1–30.
[27] J. von Neumann, in: A.W. Burks (Ed.), The theory of self-reproducing automata, Univ. of Illinois Press, Urbana, 1966.
[28] H. Akın, I. Siap, On cellular automata over Galois rings, Inf. Proc. Lett. 103 (1) (2007) 24–27.
[29] C.L. Penninger, A. Tovar, L.T. Watson, J.E. Renaud, KKT conditions satisfied using adaptive neighboring in hybrid cellular automata for topology optimization, Int. J. Pure Appl. Math. 66 (2011) 245–262.
[30] G.A. Hedlund, Endomorphisms and automorphisms of full shift dynamical system, Math. Syst. Theory 3 (1969) 320–375.
[31] G.A. Maranon, L.H. Encinas, A.H. Encinas, A.M. del Rey, G.R. Sanchez, Graphic Cryptography with Pseudorandom Bit Generators and Cellular Automata, Lecture Notes in Computer Science,, vol. 2773, Springer Berlin Heidelberg, 2003, pp. 1207–1214.
[32] G.A. Maranon, L.H. Encinas, A.M. del Rey, A New Secret Sharing Scheme for Images Based on Additive 2-Dimensional Cellular Automata, Lecture Notes in Computer Science, vol. 3522, Springer Berlin Heidelberg, 2005, pp. 411–418.
[33] C.F. Rubio, L.H. Encinas, S.H. White, A.M. del Rey, G.R. Sanchez, The use of linear hybrid cellular automata as pseudorandom bit generators in cryptography, Neural Parallel Sci. Comput. 12 (2004) 175–192.
[34] P.L. Rosin, Image processing using 3-state cellular automata, Comput. Vis. Image Underst. 114 (2010) 790–802.
[35] I. Siap, H. Akın, F. Sah, Garden of Eden configurations for 2-d cellular automaton with rule 2460n, Inform. Sci. 180 (2010) 3562–3571.
[36] S. Uguz, H. Akın, I. Siap, Reversibility algorithms for 3-state hexagonal cellular automata with periodic boundaries, Int. J. Bifurc. Chaos 23 (2013) 1350101.