Information Sciences 608 (2022) 917–930

Contents lists available at ScienceDirect

Information Sciences

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

Virtual cyclic cellular automata, finite group actions and recursive properties

D. Hernández Serrano [a,*], A. Martín del Rey [b]

[a] Universidad de Salamanca, Institute of Fundamental Physics and Mathematics, Department of Mathematics, Plaza de la Merced 1, 37008 Salamanca, Spain
[b] Universidad of Salamanca, Institute of Fundamental Physics and Mathematics, Department of Applied Mathematics, Calle del Parque 2, 37008 Salamanca, Spain

ARTICLE INFO

Article history:
Received 6 February 2019
Received in revised form 27 January 2022
Accepted 2 July 2022
Available online 5 July 2022

2010 MSC:
68Q80
68Q70
20C40
68W30
94A08

Keywords:
Elementary cellular automata
Reversibility
Rule 150
Cyclic cellular automata
Recursive rule
Wolfram number

ABSTRACT

The aim of these notes is three fold. First we introduce virtual cyclic cellular automata and show that the inverse of a reversible $(2R+1)$-cyclic cellular automaton with periodic boundary conditions is a virtual cyclic cellular automaton. These virtual automata have two special characteristics: they have active and non active cells at specific steps of times and they reflect certain periodicity. We will relate these particularities with a finite cyclic group action on the cellular automaton, and prove that the inverse transition dipolynomial is an invariant dipolynomial under this action. Secondly, we use a recursive estimation of neighbours (REN) algorithm to produce direct examples of virtual cyclic cellular automata, which moreover generalize some of the cellular automata used in applications like collective control or traffic patterns. We also propose a new REN algorithm which allows us to reinterprete a $(2R+1)$-cyclic cellular automata as a recursive sequence originated from the elementary cellular automaton with base rule 150, and which motivate us to introduce a new notion of a recursive Wolfram number for a $(2R+1)$-cyclic cellular automaton. Finally we show that this recursive Wolfram number can be computed by the new REN algorithm applied to the base rule 150 and its complementary 105 rule.

1. Introduction

A one dimensional cellular automaton (CA) is a simple model of computation capable to simulate complex phenomena which can be described as a finite number of cells, endowed with a state in a finite state set, and uniformly arranged in a one dimensional grid; the states change in discrete steps of time according to a local transition rule depending on the states of the neighbour cells at the previous steps of time [19]. Elementary cellular automata (ECA for short) are CA where the cells are linearly arranged and the state set is given by the finite field $\mathbb{F}_2$. The state of each cell is ruled by a three-variable boolean function whose variables are the states of the main cell and its two nearest cells (neighbours) on each side, and they have been extensively studied (see [18]). The reversibility for these cellular automata and computing the inverse of a reversible cellular automaton are problems which have been treated in many works ([1,7,11,13,14]).

Local transition functions for one dimensional CA can be also defined using more than the two nearest cells by including the notion of neighborhood radius R, and a particular type of them are the $(2R+1)$-cyclic cellular automata with periodic

---

* Corresponding author.
E-mail addresses: dani@usal.es (D. Hernández Serrano), delrey@usal.es (A. Martín del Rey).

https://doi.org/10.1016/j.ins.2022.07.007
0020-0255/

boundary conditions over the finite field $\mathbb{F}_2$, $(2R+1)$-CCA with p.b.c. for short, studied in [5]. The most important issue in the reversibility problem, due to wide range of applications in signal and image processing, linear forecast or error correcting code theory (see [2]), is to explicitly compute the inverse cellular automaton. In [5], an explicit formula for the inverse transition polynomial of a reversible $(2R+1)$-CCA with p.b.c is given for any value of R (see [4,8–10] for the $R=1$ case, which corresponds to the ECA with rule number 150, ECA150). If we take a closer look the formula there contained, we can observe the inverse of a reversible $(2R+1)$-CCA has also a cyclic CA structure (with, in general, a different neighbourhood radius) but in this CCA not all the cells are active, some of them are not appearing in the explicit expression.

In these notes we are interested in this type of CCA where not all the cells are active at a specific time. We will define in Section 4 a new class of CCA called virtual CCA (VCCA) having both an standard neighbourhood radius and a new extra radius (which we call the virtual radius) which keeps track of the active cells. We will prove that the inverse of a $(2R+1)$-CCA is a VCCA. The specific formula for the inverse transition dipolynomial says, a priori, that we can compute the inverse, but that we can not determine the exact location or distribution of the active cells. Nonetheless, it reflects certain symmetry behaviour among the active cells. We will introduce a finite cyclic group action on a CCA and use it to prove that the virtual transition dipolynomial of the inverse can be shown to be invariant under a $\mathbb{Z}/N$ action $(N=2R+1)$ and proved to be equal to a dipolynomial with neighbourhood radius d but in a different weighted variable. That is, for this particular VCCA we know the exact distribution of active cells. This nice and clean formula (given in Section 5) presents the inverse as particular type of VCCA and simplifies the results of [5].

The notion of an extra perception radius is also treated in [6] from a different perspective, let us briefly introduce it here. In [12] the motion of a simulated flock of birds is artificially created by a distributed model, where each bird is an independent entity moving accordingly to its local perception, an extra radius (which have lead to other applications such collective control of robots, traffic patterns or 3D patterns in animation). To study the relationship between this and the information processing and fluctuation control in CA pattern formation, and using a recursive estimation of neighbours (REN) algorithm, an extended CA (having an extra radius used to define the local perception area) is defined in [6]. We will prove that the REN algorithm proposed in this paper and applied with base rule ECA150, recursively produces CCA rules which are VCCA, where the virtual radius is the extra radius which codifies the local perception. Moreover, we propose a different REN algorithm which allow us to prove that the $(2R+1)$-CCA can be thought of as a recursive sequence parametrised by R and originated from the base rule ECA150 (the $R=1$ case), thus we could have certain control on the fluctuation in this particular group formation. This suggests that the $(2R+1)$-cyclic cellular automata might be defined by a Wolfram type rule obtained recursively by the base rule ECA150. In fact, we will define a new recursive Wolfram number (RWN) which is recursively generated by the ECA150 and its complementary rule, the ECA105.

The paper is organized as follows: in Section 2 the mathematical background related to elementary cellular automata with neighborhood radius R and its subclass of $(2R+1)$-cyclic cellular automata is defined; Section 3 is devoted to show how to explicitly compute the transition dipolynomial of the inverse of a reversible $(2R+1)$-CCA with p.b.c. In Section 4 we introduce a new class of CCA, the virtual CCA, and prove that the inverse of a reversible $(2R+1)$-CCA is a virtual CCA. Section 5 starts by giving some basic notions on finite cyclic group actions in the context of CCA and follows by proving that the virtual transition dipolynomial of the inverse can be shown to be invariant under a $\mathbb{Z}/N$ action $(N=2R+1)$ and proved to be equal to a dipolynomial with standard neighbourhood radius but in a different weighted variable. This drastically simplifies the explicit expression of the inverse of a reversible CCA and some of the results of [5]. This section ends by illustrating the results of the last two sections working out the examples for the $R=1$ (corresponding to the ECA150) and the $R=2$ (which is the penta-cyclic rule) cases. The aim of Section 6 consists of showing that the REN algorithm of [6] recursively produce examples of non trivial VCCA, where the virtual radius is the extra radius which codifies the local perception area. This proves the existence of VCCA beyond being the inverses of certain reversible CCA. Moreover, we propose a different REN algorithm which allow us to prove that the $(2R+1)$-CCA can be shown to be a recursive sequence parametrised by R and originated from the base rule ECA150. We finish these notes by defining in Section 7 a recursive Wolfram number for a $(2R+1)$-CCA, which can be recursively computed by the new REN algorithm applied to the base rule ECA150 and its complementary ECA105 rule.

2. The basic theory of elementary cellular automata

Elementary cellular automata with neighborhood radius R are linear cellular automata characterized by the following features:

(1)The n cells constituting the cellular space are uniformly arranged in a one-dimensional grid.

(2)The state set is $\mathbb{F}_2=\{0,1\}$, such that s_i^t stands for the state of the i-th cell at the step of time t.

(3)The state of each cell at time $t+1$ depends on the states of R left and right neighbour cells and the cell itself at the previous step of time. Consequently, the local transition linear function is as follows:

$$s_i^{t+1}=f\left(s_{i-R}^t,\ldots,s_{i-1}^t,s_i^t,s_{i+1}^t,\ldots,s_{i+R}^t\right).$$

(4)As the number of cells is finite, boundary conditions must be taken into account. Usually periodic boundary conditions (p.b.c.) are assumed, that is, if $i\equiv j\ (\mathrm{mod}\,n)$ then $s_i^t=s_j^t$.

The $(2R+1)$-cyclic cellular automata with p.b.c, $(2R+1)$-CCA, are defined by the local transition function:

$$s_i^{t+1} = s_{i-R}^t \oplus \cdots \oplus s_{i-1}^t \oplus s_i^t \oplus s_{i+1}^t \oplus \cdots \oplus s_{i+R}^t, \quad 1 \leqslant i \leqslant n, \tag{1}$$

where $\oplus$ denotes the sum in $\mathbb{F}_2$. If $C^t = (s_1^t, s_2^t, \ldots, s_n^t) \in \mathbb{F}_2^n$ is the configuration at step of time t, its global dynamics can be represented in $\mathbb{F}_2[x]/(x^n - 1)$ as follows:

$$P^{t+1}(x) \equiv T_n(x) \cdot P^t(x) \mod x^n - 1, \tag{2}$$

where $P^t(x) = \sum_{i=1}^n s_i^t x^i$ is the configuration polynomial at step of time t, and

$$T_n(x) \equiv D_R(x) = 1 + \sum_{j=1}^R (x^j + x^{-j})$$

is the transition dipolynomial (of neighbourhood radius R).

In [5] the following properties for dipolynomials are proven (over $\mathbb{F}_2$):

(a)$D_m(x) = \frac{x^{-m}(1+x^{1+2m})}{1+x}$. In particular $D_R(x) = \frac{x^{-R}(1+x^N)}{1+x}$.
(b)$D_{n+i}(x) \equiv D_i(x) \pmod{x^n - 1}$. In particular $D_n(x) \equiv 1 \pmod{x^n - 1}$.
(c)$D_{n-i}(x) \equiv D_{i-1}(x) \pmod{x^n - 1}$. In particular $D_{n-1}(x) \equiv 1 \pmod{x^n - 1}$.
(d)$(x^{-i} + x^i)D_b(x) = D_{b-i}(x) + D_{b+i}(x)$.
(e)$D_a(x)D_b(x) = \sum_{i=-a}^a D_{b+i}(x)$.
(f)If we formally define $D_{\frac{b}{c}}(x^c) = \frac{x^{-b}\left(1+x^{c\left(1+2\frac{b}{c}\right)}\right)}{1+x^c} = \frac{x^{-b}(1+x^{c+2b})}{1+x^c}$, then:

$$D_a(x)D_{\frac{b}{2a+1}}(x^{2a+1}) = D_{a+b}(x).$$

Let us also recall here (since we shall use it in the last section) that the local transition functions for one-dimensional ECA are boolean three-variable functions (the neighborhood radius is 1), thus there are 2^{2^3} possible elementary rules leading to 256 elementary cellular automata. Each ECA is indexed by its rule number (the Wolfram number) $0 \leqslant w = \sum_{i=0}^7 \alpha_i \cdot 2^i \leqslant 255$ where:

$$f(0,0,0) = \alpha_0,\ f(0,0,1) = \alpha_1,\ f(0,1,0) = \alpha_2,\ f(0,1,1) = \alpha_3, \tag{3}$$

$$f(1,0,0) = \alpha_4,\ f(1,0,1) = \alpha_5,\ f(1,1,0) = \alpha_6,\ f(1,1,1) = \alpha_7. \tag{4}$$

In particular, the ECA150 is defined by means of the following local linear transition function:

$$s_i^{t+1} = f^{150}(s_{i-1}^t, s_i^t, s_{i+1}^t) = s_{i-1}^t \oplus s_i^t \oplus s_{i+1}^t, \quad 1 \leqslant i \leqslant n, \tag{5}$$

and its Wolfram number 150 is obtained by $w = 2 + 2^2 + 2^4 + 2^7$.

3. The inverse of a reversible $(2R+1)$-cyclic ECA

Let $N = 2R+1$ and $n = Nk + r$ where k is a natural number and $r \in \{1, \ldots, N-1\}$ denotes the remainder of n modulo N. A $(2R+1)$-cyclic CA with p.b.c has, modulo $x^n - 1$, the following transition dipolynomial:

$$T_n(x) \equiv D_R(x) = 1 + \sum_{j=1}^R (x^j + x^{-j}).$$

In [16] it is shown that a $(2R+1)$-cyclic ECA is reversible if N and n are coprime, and notice that if $N = \prod_{i=1}^m p_i^{n_i}$ is the prime factorization of N, we need $(r, p_i) = 1$ for all $i = 1, \ldots, m$ to achieve reversibility. In [5] an explicit and closed formula is given for the inverse of a reversible $(2R+1)$-cyclic ECA:

$$D_{\frac{b}{N}}(x^N) \equiv \frac{x^{rr'-tN} + x^{(t+1)N-rr'}}{1+x^N}, \quad \text{where } b = n + tN - rr'. \tag{6}$$

It is shown that the inverse a reversible CCA has also a CCA structure, but with a different neighborhood radius, and its inverse rule will be given by a particular dipolynomial which we shall study next in detail (and which shall be simplified in Section 5).

Expression (6) is no yet formulated as a dipolynomial and it depends on the additional numbers $r\prime$ and t. These numbers are computed by the CES algorithm we write below in detail, which determines a correct sequence of couples $(r\prime, t)$ for all the values of $r \in \{1, 2, \ldots, N-1\}$. Then we can introduce a particular number $n = Nk + r$ and obtain its unique pair $(r\prime, t)$.

The algorithm will be only written for the values $0 \leqslant r \leqslant R$, since the remaining values will be achieved later on by using $N - r$. The CES algorithm is the following:

1. We start by computing, in increasing order, the following sequence:

$$R = \frac{R}{1}, \lceil \frac{R}{2} \rceil, \lceil \frac{R}{3} \rceil, \dots, \lceil \frac{R}{R-1} \rceil, \frac{R}{R} = 1$$

   and store 1 and R.
2. In increasing order and for each $j \in \{2, \dots, R-1\}$ such that $\gcd(j, p_i) = 1$ (for all $i = 1, \dots, m$), we divide R by j and write $R = qj + h$, where q is a natural number and $h \in \{0, 1, \dots, R-1\}$.
   - If $j - h = 1$ we store $j' = \lceil \frac{R}{j} \rceil$ and, if and only if $j \neq j'$, we then also store $j = \lceil \frac{R}{j'} \rceil$. For the value j' we introduce two new variables s and t (that we will need later on) and define $(r', s, t) = (j, 1, 0)$; similarly, for the value j we define $(r', s, t) = (j', 1, 0)$. We continue this procedure for the values which do not produce already stored numbers.
   - If $j - h \neq 1$, remove $\lceil \frac{R}{j} \rceil$ and continue in increasing order with the sequence.
3. If $j \in \{2, \dots, R-1\}$ (with $\gcd(j, p_i) = 1$ for all $i = 1, \dots, m$) is the first number such that $\lceil \frac{R}{j-1} \rceil = \lceil \frac{R}{j} \rceil$, then compute $\lceil \frac{sR}{j} \rceil$ where $s := 2t + 1$ is the smallest odd integer number which is smaller or equal that j and such that $\lceil \frac{sR}{j} \rceil \neq \lceil \frac{R}{i} \rceil$ for all $i = 2, \dots, j-1$.
4. For each $2 \leqslant j \leqslant R-1$, let $j' = \lceil \frac{sR}{j} \rceil$ be the result of the previous step and let h be such that $sR = Qj + h$ (aso that $0 \leqslant h < j$).
   - If $j - h - t = 0, 1$ then we store $j' = \lceil \frac{sR}{j} \rceil$ and, in the case that $j \neq j'$, we also store $j = \lceil \frac{sR}{j'} \rceil$. For the value j' we now update the variables by setting $(r', s, t) = \left(j, s, \frac{s-1}{2}\right)$; and for the value j we set $(r', s, t) = \left(j', s, \frac{s-1}{2}\right)$.
   - If $j - h - t \neq 0, 1$, then remove $j' = \lceil \frac{sR}{j} \rceil$ and go back to step 3 starting from that j', moving in increasing order and not considering the already stored numbers.

Now we know how to compute the couple (r', t) for the values in $\{1, \dots, R\}$, let us show how to deal with the values $\{R+1, \dots, 2R\}$. Let us define $\bar{s} = 2r' - s$ and set $\bar{s} = 2\bar{t} + 1$. Once we apply the CES algorithm for the values in $\{1, 2, \dots, R\}$, the couple $(r', \bar{t})$ for the entries $\bar{r} \in \{R+1, \dots, 2R\}$ is computed as follows:

- Write $r = N - \bar{r} \in \{1, 2, \dots, R\}$ and apply the CES algorithm to look for its associated r' and $s = 2t + 1$.
- Use $\bar{s} = 2r' - s = 2\bar{t} + 1$ and apply the following result ([5, Lemma 4] to compute its associated $\bar{t}$: if $r = \lceil \frac{sR}{r'} \rceil$ then:

$$\bar{r} = N - r = \begin{cases} \lceil \frac{(2r'-s)R}{r'} \rceil & \text{if} \quad h' \neq 0 \\ \frac{(2r'-1)R}{r'} + 1 & \text{if} \quad h' = 0 \end{cases}$$

  where $h' = h - r + r'$.

Let us illustrate the CES algorithm by performing an example:

Example 1. We shall deal with the $N = 11$ case, that is, a $(2R+1)$-CCA with neighbourhood radius $R = 5$ (notice that since 11 is prime, coprime conditions in the algorithm are always satisfied). The possible values for the remainders r of n modulo N $(n = Nk + r)$ are in the set $\{1, 2, \dots 5, 6, \dots, 10\}$. Let us apply the CES algorithm for the first five entries of these values $\{1, 2, 3, 4, 5\}$:

- For the entry 1: we do $\lceil \frac{5}{1} \rceil = \frac{5}{1} = 5$ so that $j = 1$ and $j' = 5$ and $R = qj + h$ implies that $j - h = 0$. We then store $j' = \frac{5}{1} = \boxed{5}$ and $j = \frac{5}{5} = \boxed{1}$ (since $j \neq j$). Then, for the value $j' = 5$ we introduce s and t by defining $(r', s, t) = (1, 1, 0)$; for the value $j = 1$ we define $(r', s, t) = (5, 1, 0)$.
- For the entry 2: one has $\lceil \frac{5}{2} \rceil = 3$ and then $R = q2 + h$, so that $j - h = 1$. We then store $j' = \lceil \frac{5}{2} \rceil = \boxed{3}$ and $j = \lceil \frac{5}{3} \rceil = \boxed{2}$ (since $j \neq j'$ and none of them are already stored). That is, for the value $j' = 3$ we define $(r', s, t) = (2, 1, 0)$ and for the value $j = 2$ we define $(r', s, t) = (3, 1, 0)$.
- The entry 3 is already stored.
- For the entry 4: we firstly obtain that $\lceil \frac{R}{4} \rceil = 2$, which is already store, so we perform $\lceil \frac{3 \cdot 5}{4} \rceil = 4$, this one not being stored. Then we have that $j = j' = 4, s = 3$ (so that $t = 1$) and $h = 3$. Thus $j - h - t = 0$ and therefore we can store $j' = \lceil \frac{3 \cdot 5}{4} \rceil = \boxed{4}$. For the value $j' = 4$ we define $(r', s, t) = (4, 3, 1)$.
- The entry 5 is already stored.

Now we compute r' and $\bar{t}$ for the entries in $\{6, 7, 8, 9, 10\}$:

- If $\bar{r} = 6$ it is $r = N - \bar{r} = 5 = \frac{5}{1}$ so that we set $(r', s, t) = (1, 1, 0)$. Then we have $\boxed{6} = \bar{r} = N - r = \frac{5}{1} + 1$ and therefore $r' = 1$ and $\bar{s} = 1 \Rightarrow \bar{t} = 0$.
  Thus, for the value 6 we define $(r', \bar{s}, \bar{t}) = (1, 1, 0)$.
- If $\bar{r} = 7$ it is $r = N - \bar{r} = 4 = \lceil \frac{3 \cdot 5}{4} \rceil$ so that $(r', s, t) = (4, 3, 1)$. Then we have $\boxed{7} = \bar{r} = N - r = \lceil \frac{5 \cdot 5}{4} \rceil$ (since $s + \bar{s} = 2r'$). Thus, for the value 7 we define $(r', \bar{s}, \bar{t}) = (4, 5, 2)$.
- If $\bar{r} = 8$ it is $r = N - \bar{r} = 3 = \lceil \frac{5}{2} \rceil$ so that $(r', s, t) = (2, 1, 0)$. Then we have $\boxed{8} = \bar{r} = N - r = \lceil \frac{3 \cdot 5}{2} \rceil$ (since $s + \bar{s} = 2r'$). Thus, for the remainder 8 we define $(r', \bar{s}, \bar{t}) = (2, 3, 1)$.
- If $\bar{r} = 9$ it is $r = N - \bar{r} = 2 = \lceil \frac{5}{3} \rceil$ so that $(r', s, t) = (3, 1, 0)$. Then we have $\boxed{9} = \bar{r} = N - r = \lceil \frac{5 \cdot 5}{3} \rceil$ (since $s + \bar{s} = 2r'$). Thus, for the remainder 9 we define $(r', \bar{s}, \bar{t}) = (3, 5, 2)$.
- If $\bar{r} = 10$ it is $r = N - \bar{r} = 1 = \frac{5}{5}$ so that $(r', s, t) = (5, 1, 0)$. Then we have $\boxed{10} = \bar{r} = N - r = \frac{9 \cdot 5}{5} + 1$ (since $s + \bar{s} = 2r'$). Thus, for the remainder 10 we set $(r', \bar{s}, \bar{t}) = (5, 9, 4)$.

Finally we end up with the following table (see Table 1):

Now, let us show how to compute the inverse of a reversible $(2R + 1)$-CCA using the CES algorithm and giving its inverse rule by a transition dipolynomial:

Step 1. Apply the CES algorithm to compute the sequence of pairs (r', t) for the entries in $\{1, 2, \ldots, R\}$.
Step 2. In order, and for each entry $\bar{r}$ in $\{R + 1, \ldots 2R\}$:

- Do $r = N - \bar{r}$ (which belongs to $\{1, 2, \ldots, R\}$) and look for its associated pair (r', t) given by step 1.
- The associated pair for $\bar{r}$ is $(r', \bar{t})$ where $\bar{t} := r' - t + 1$.
- **Step 3.**Divide n by N so that $n = Nk + r$, where $1 \leqslant r < N$
- If $1 \leqslant r \leqslant R$, localise its associated pair (r', t) from step 1 and use the [5, Theorem 2] to compute the inverse transition dipolynomial:

$$\widetilde{T}_n(x) = 1 + \sum_{\substack{l=1,\ldots,k \\ j=0,1,\ldots,r'-1}} \left(x^{(lN-rj)} + x^{-(lN-rj)}\right) + + \sum_{l'=1,\ldots,t} \left(x^{((k+l')N-r(r'-1))} + x^{-((k+l')N-r(r'-1))}\right). \tag{7}$$

- If $R + 1 \leqslant \bar{r} \leqslant 2R$, localice its associated pair $(r', \bar{t})$ from step 2 and use the [5, Theorem 2] to compute the inverse transition dipolynomial:

$$\widetilde{T}_n(x) = 1 + \sum_{\substack{l=1,\ldots,k \\ j=0,1,\ldots,r'-1}} \left(x^{(lN-\bar{r}j)} + x^{-(lN-\bar{r}j)}\right) + + \sum_{\bar{l}'=1,\ldots,\bar{t}} \left(x^{\left(\left(k+\bar{l}'\right)N-\bar{r}(r'-1)\right)} + x^{-\left(\left(k+\bar{l}'\right)N-\bar{r}(r'-1)\right)}\right) \tag{8}$$

4. Virtual cyclic cellular automata

In this section we are interested in defining a particular class of cyclic cellular automata with p.b.c. having a transition dipolynomial of the form $1 + \sum_{j=1}^{b} a_j(x^j + x^{-j})$, where some a_j might be zero. The reason is motivated by the specific shape of the inverse transition dipolynomial of Eq. (7) (and also Eq. (8)), where there is a neighbourhood of radius $b = (k + t)N - r(r' - 1) = n + Nt - rr'$, but there are only $2d + 1$ active neighbours or cells, where $d = kr' + t$.

We shall define virtual cyclic cellular automata with p.b.c. (VCCA from now on) and we will prove that the inverse of a $(2R + 1)$-CCA is a VCCA.

Definition 1. Let $d \leqslant b$ be natural numbers. We define a one dimensional virtual $(2b + 1)$-cyclic cellular automaton with virtual neighbourhood radius d as a particular subclass of $(2b + 1)$-CCA with p.b.c. having the following local transition function:

$$s_i^{t+1} = s_i^t \oplus \sum_{j=1}^{b} a_j \left(s_{i-j}^t \oplus s_{i+j}^t\right), \text{ where } a_j \neq 0 \text{ only for } d \text{ elements in } \{a_1, \ldots, a_b\}, \tag{9}$$

where both $\sum$ and $\oplus$ denote the sum in $\mathbb{F}_2$. We'll refer to it as a (b, d)-VCCA.

Table 1
Applying the CES algorithm for $R = 5$.

r	1	2	3	4	5	6	7	8	9	10
ceilings	$\frac{5}{5}$	$\lceil \frac{5}{3} \rceil$	$\lceil \frac{5}{2} \rceil$	$\lceil \frac{3 \cdot 5}{4} \rceil$	$\frac{5}{1}$	$\frac{5}{1} + 1$	$\lceil \frac{5 \cdot 5}{4} \rceil$	$\lceil \frac{3 \cdot 5}{2} \rceil$	$\lceil \frac{5 \cdot 5}{3} \rceil$	$\frac{9 \cdot 5}{5} + 1$
r′	5	3	2	4	1	1	4	2	3	5
s or $\bar{\mathbf{s}}$	1	1	1	3	1	$\bar{1}$	$\bar{5}$	$\bar{3}$	$\bar{5}$	$\bar{9}$
t or $\bar{\mathbf{t}}$	0	0	0	1	0	$\bar{0}$	$\bar{2}$	$\bar{1}$	$\bar{2}$	$\bar{4}$

Remark 1. Since it is defined as a particular subclass of $(2b+1)$-cyclic cellular automata, we are assuming that $a_b \neq 0$.

Remark 2. Working over $\mathbb{F}_2$ only d coefficients are 1, so that the local transition function can be written as:

$$s_i^{t+1} = s_i^t \oplus \sum_{\substack{1 \leqslant m \leqslant d \\ j_m \in \{1,\dots,b\}}} \left(s_{i-j_m}^t \oplus s_{i+j_m}^t \right),$$

where $j_d = b$ (since we are assuming in the definition that $a_b \neq 0$) and thus $s_{i+j_d}^t = s_{i+b}^t$.

Remark 3. Notice that the virtual cyclic cellular automata are linear CA over a finite field, and that their associated transition matrix can be understood as a particular circulant matrix having a special diagonal band (having both 1's and 0's) determined by the virtual radius d. From this point of view, we shall be proving in this section that the inverse of a circulant matrix (coming from a reversible CCA with p.b.c.) is a matrix of this particular form (see [3,15,17,20] for the general theory of circulant matrices and how to compute their inverses).

Definition 2. Let $C^t = (s_1^t, s_2^t, \dots, s_n^t) \in \mathbb{F}_2^n$ be the configuration at step of time t. The global dynamics of a (b,d)-VCCA can be represented in $\mathbb{F}_2[x]/(x^n - 1)$ by:

$$P^{t+1}(x) \equiv D_b^d(x) \cdot P^t(x) \mod x^n - 1,$$

where $P^t(x) = \sum_{i=1}^n s_i^t x^i$ is the configuration polynomial at step of time t and

$$D_b^d(x) = 1 + \sum_{j=1}^{b} a_j \left(x^j + x^{-j} \right), \text{ where } a_j \neq 0 \text{ only for } d \text{ elements in } \{a_1, \dots, a_b\}.$$

We will refer to this transition dipolynomial as a (b,d)-virtual transition dipolynomial.

Remark 4. Working over $\mathbb{F}_2$ only d coefficients are 1 in $D_b^d(x)$, so that:

$$D_b^d(x) = 1 + \sum_{\substack{m=1,\dots,d \\ j_m \in \{1,\dots,b\}}} \left(x^{j_m} + x^{-j_m} \right),$$

where $x^{j_d} = x^b$.

Remark 5. The definition of virtual CCA can be extended to other finite fields, we shall study its properties elsewhere.

Notice that if $d = b = R$ then a (b,d)-VCCA is just the $(2R+1)$-CCA with p.b.c defined by Eq. (1):

$$s_i^{t+1} = s_{i-R}^t \oplus \cdots \oplus s_{i-1}^t \oplus s_i^t \oplus s_{i+1}^t \oplus \cdots \oplus s_{i+R}^t, \quad 1 \leqslant i \leqslant n,$$

and thus $D_b^d(x) = D_R(x) = 1 + \sum_{j=1}^R \left(x^j + x^{-j} \right)$.

Then, we can reformulate [5, Theorems 2 and 3] as:

Proposition 1. Let $N = 2R+1$ and $n = Nk + r$, where $r \leqslant R$ and $\gcd(n,N) = 1$. The inverse of a reversible $(2R+1)$-CCA with p.b.c. is a (b,d)-VCCA where $b = (k+t)N - r(r\prime - 1)$ and $d = kr\prime + t$. Its transition dipolynomial is given, modulo $x^n - 1$, by the (b,d)-virtual dipolynomial $D_b^d(x)$.

Proposition 2. Let $N = 2R+1$ and $n = Nk + \bar{r}$ where $R < \bar{r} < N$ and $\gcd(n,N) = 1$. The inverse of a reversible $(2R+1)$-CCA with p.b.c. is a $(\bar{b},\bar{d})$-VCCA where $\bar{b} = (k+\bar{t})N - \bar{r}(r\prime - 1)$ and $\bar{d} = kr\prime + \bar{t}$. Its transition dipolynomial is given, modulo $x^n - 1$, by the $(\bar{b},\bar{d})$-virtual dipolynomial $D_{\bar{b}}^{\bar{d}}(x)$.

Corollary 1. *For $r = R, R+1$, the inverse of a reversible $(2R+1)$-CCA with p.b.c. is a (b,d)-VCCA where $b = kN$ and $d = k$ and its transition dipolynomial is given virtual dipolynomial $D_b^d(x) = D_k\left(x^N\right) = D_{Nk}^k(x)$.*

5. Cyclic group actions and Virtual Cyclic Cellular Automata

The specific formula for the inverse transition dipolynomial has a complicated shape. We will prove next that the virtual transition dipolynomial $D_b^d(x)$ of the inverse of a reversible $(2R+1)$-CCA is invariant under a $\mathbb{Z}/N$ action ($N = 2R+1$), and we will prove that it is in fact equal to the dipolynomial $D_d(y)$ with neighbourhood radius d but in the weighted variable $y = x^N$. This nice and clean formula show the inverse as particular type of VCCA and simplifies the results of [5].

5.1. Cyclic group action on a CCA

Let the finite cyclic group $\mathbb{Z}/N$ act on the polynomial ring $k[x]$ by $x \mapsto \xi x$, where $\xi^N = 1$ (that is, ξ is a N-th root of unity, an element of the dual group $\mu_N \simeq (\mathbb{Z}/N)^*$).

Since we have that:

$$\xi^j x^j = x^j \quad \Longleftrightarrow \quad \xi^j = 1 \quad \Longleftrightarrow \quad j \equiv 0 \mod N,$$

the ring of invariants is $k[x]^{\mathbb{Z}/N} \simeq k[x^N] = k[y]$, where we are denoting $y = x^N$. Thus, we have a immersion $k[y] \hookrightarrow k[x]$ of rings which endows $k[x]$ with a $k[y]$-module structure:

$$k[x] \simeq k[y] \oplus x k[y] \oplus \cdots \oplus x^{N-1} k[y]$$

and presents a decomposition of the polynomial ring $k[x]$ into an invariant part $k[y]$, and $N-1$ parts $x^j k[y]$ which are usually referred to as anti-invariant parts or j-th twisted sectors.

Remark 6. From the point of view of the algebraic geometry we are working with a cyclic $N:1$ cover

$$\pi : X = \mathrm{Spec}\, k[x] \to Y = \mathrm{Spec}\, k[y],$$

where the space Y is identified with the quotient space

$$Y \simeq X/(\mathbb{Z}/N) \simeq \mathrm{Spec}\, k[x]^{\mathbb{Z}/N} \simeq \mathrm{Spec}\, k\left[x^N\right].$$

The direct image of the ring $k[x]$ has then a natural $k[y]$-module structure $\pi_* k[x] \simeq k[y] \oplus x k[y] \oplus \cdots \oplus x^{N-1} k[y]$.

We can do analogous considerations for the polynomial ring $k\left[x, x^{-1}\right]$ where the invariant part under the $\mathbb{Z}/N$ action $\left(x, x^{-1}\right) \mapsto \left(\xi x, \xi^{-1} x^{-1}\right)$ is

$$k\left[x, x^{-1}\right]^{\mathbb{Z}/N} \simeq k\left[y, y^{-1}\right].$$

It is worth to notice that, in general, the decomposition of a dipolynomial in $k\left[x, x^{-1}\right]$ could have an anti-invariant part living in $x^j k[y]$ but it might not have an anti-invariant part in $x^{-j} k\left[y^{-1}\right]$ for $j = 0, 1, \ldots, N-1$. Since in these notes we are only considering a particular type of elements in $k\left[x, x^{-1}\right]$ which are invariant under the substitution $x \mapsto x^{-1}$, which are the transition dipolynomials $D_m(x) = 1 + \sum_{i=1}^{m} \left(x^{-i} + x^i\right) \in k\left[x^{-1}, x\right]$, this situation can not happen (if there is a term $x^j y$ necessarily there is also the term $x^{-j} y^{-1}$). By abuse of language, we shall refer to these particular anti-invariants parts lying on $x^j k[y] \oplus x^{-j} k\left[y^{-1}\right]$ as the j-th twisted sectors of $k\left[x, x^{-1}\right]$.

5.2. The inverse VCCA as an invariant CCA under a cyclic group action

If we look at the formula for the inverse transition dipolynomial of Eq. (7), we observe that this virtual dipolynomial seems to have not only an invariant part in $k\left[y, y^{-1}\right]$, but also some anti-invariant parts living in the twisted sectors. We will see that this is not the case and that in fact, the inverse transition dipolynomial of a reversible $(2R+1)$-CCA can be expressed in terms only of the invariant part. This improves and simplifies the results of [5].

Let us note that a simple computation shows the following:

Lemma 1. *Let us consider the dipolynomial* $D_{mN}(x) = 1 + \sum_{i=1}^{mN} \left(x^{-i} + x^i\right) \in k\left[x^{-1}, x\right]$. *Then its invariant part under the* $\mathbb{Z}/N$ *action on* $k\left[x, x^{-1}\right]$ *defined by* $\left(x, x^{-1}\right) \mapsto \left(\xi x, \xi^{-1} x^{-1}\right)$ *is:*

$$D_{mN}(x)^{\mathbb{Z}/N} = D_m(y) = 1 + \sum_{i=1}^{m} \left(y^{-i} + y^i\right) \in k\left[y^{-1}, y\right],$$

where $y = x^N$.

Theorem 1. *Let* $N = 2R+1$ *and* $n = Nk + r$, *where* $r \leqslant R$ *and* $\gcd(n, N) = 1$. *The inverse of a reversible* $(2R+1)$*-CCA with p.b.c. is a* (b, d)*-VCCA where* $b = (k+t)N - r(r'-1)$ *and* $d = kr' + t$, *and its transition dipolynomial is given, modulo* $x^n - 1$, *by the* $\mathbb{Z}/N$*-invariant dipolynomial:*

$$D_b^d(x) = D_{dN}(x)^{\mathbb{Z}/N} = D_d(y),$$

where $y = x^N$.

Proof. Notice that by [5, Theorem 1] we have $D_b^d(x) \equiv D_{\frac{b}{N}}(x^N) \mod x^n - 1$, where the fractional dipolynomial is defined in Section 2. Thus, using Eq. (7) and Lemma 1, we have to prove that $D_{\frac{b}{N}}(x^N) = D_d(y)$. Since $n = Nk + r$ and we are working modulo $x^n - 1$, we have that $x^{Nk} \equiv x^{-r}$. Bearing in mind that $b = n + tN - rr\prime$, we then have:

$$\begin{aligned} D_{\frac{b}{N}}(x^N) &= \frac{x^{-b}\left(1 + x^{N+2b}\right)}{1 + x^N} \equiv \frac{x^{-(tN-rr\prime)}\left(1 + x^{N+2(tN-rr\prime)}\right)}{1 + x^N} \equiv \\ &\equiv \frac{x^{-(tN+kNr\prime)}\left(1 + x^{N+2(tN+kNr\prime)}\right)}{1 + x^N} = \frac{x^{-Nd}\left(1 + x^{N(1+2d)}\right)}{1 + x^N} = \\ &= D_d(y) = D_{dN}(x)^{\mathbb{Z}/N}. \end{aligned}$$

□

Similarly we can prove:

Theorem 2. *Let* $N = 2R + 1$ *and* $n = Nk + \bar{r}$ *where* $R < \bar{r} < N$ *and* $\gcd(n, N) = 1$. *The inverse of a reversible* $(2R + 1)$*-CCA with p.b.c. is a* $(\bar{b}, \bar{d})$*-VCCA where* $\bar{b} = (k + \bar{t})N - \bar{r}(r\prime - 1)$ *and* $\bar{d} = kr\prime + \bar{t}$, *and its transition dipolynomial is given, modulo* $x^n - 1$, *by the* $\mathbb{Z}/N$*-invariant dipolynomial:*

$$D_{\bar{b}}^{\bar{d}}(x) = D_{\bar{d}N}(x)^{\mathbb{Z}/N} = D_{\bar{d}}(y).$$

Corollary 2. *For* $r = R, R + 1$, *the inverse of a reversible* $(2R + 1)$*-CCA with p.b.c. is a* (b, d)*-VCCA where* $b = kN$ *and* $d = k$ *and its transition dipolynomial is given by the* $\mathbb{Z}/N$*-invariant dipolynomial:*

$$D_{Nk}^{k}(x) = D_{Nk}(x)^{\mathbb{Z}/N} = D_k(y).$$

Remark 7. In the context of algebraic geometry, if one consider the orbifold (stack quotient) $\mathcal{Y} = [X/(\mathbb{Z}/N)]$ defined in the weighted coordinate $y = x^N$, then the coordinate x is often referred to as an orbinate. From this point of view, we should have been called a (b, d)-VCCA as a d-orbifold $(2b + 1)$-CCA or a $(2b + 1)$-CCA with orbifold neighbourhood radius d, since it is defined using the transition dipolynomial in the orbinate x. Nonetheless, we have finally proved in 1 and 2 that the inverse of a reversible $(2R + 1)$-CCA (the (b, d)-VCCA) is in fact an ordinary cyclic cellular automaton with neighbourhood radius d in the weighted variable $y = x^N$.

5.3. Examples: the ECA150 and the penta cyclic cellular automaton

The ECA150 with p.b.c., defined by the local transition function $s_i^{t+1} = s_{i-1}^t \oplus s_i^t \oplus s_{i+1}^t$ $(1 \leqslant i \leqslant n)$, corresponds to the $(2R + 1)$-CCA with p.b.c for $R = 1$. Given $n = 3k + r$, its transition dipolynomial is $D_1(x) = x^{-1} + 1 + x$ and the transition dipolynomial corresponding to its inverse cellular automaton (when n and 3 are coprime) can be determined by the virtual dipolynomial:

$$D_{3k}^{k}(x) = D_{3k}(x)^{\mathbb{Z}/3} = D_k\left(x^3\right) = 1 + \sum_{i=1}^{k}\left(x^{3i} + x^{-3i}\right),$$

since $b = 3k$ and $d = k$. Almost everything here, was contained in [5], so let us study next the $R = 2$ case, where the virtual context and the cyclic group action really makes apparent.

The cellular automaton with penta cyclic rule, defined by the local transition function $s_i^{t+1} = s_{i-2}^t \oplus s_{i-1}^t \oplus s_i^t \oplus s_{i+1}^t \oplus s_{i+2}^t$ $(1 \leqslant i \leqslant n = 5k + r$ where $r < 5)$, corresponds to the $(2R + 1)$-CCA with p.b.c for $R = 2$ and its transition dipolynomial is $D_2(x) = x^{-2} + x^{-1} + 1 + x + x^2 \pmod{x^n - 1}$. If $\gcd(5, n) = 1$ it is reversible and the transition dipolynomial of the inverse can be described (modulo $x^n - 1$) as virtual dipolynomials and simplified using the $\mathbb{Z}/5$ cyclic action.

- For $r = 1$ by Proposition 1 we have that the inverse is the virtual dipolynomial

  $$D_{5k-1}^{2k}(x) = D_{\frac{5k-1}{5}}\left(x^5\right),$$

  so that the inverse has $2d = 4k$ symmetric active neighbours fitted in a radius of $b = 5k - 1$, but in order to know how these active neighbours are distributed we need, a priori, its explicit expression as a dipolynomial given by Eq. (7) (proved in [5, Theorem 1]):

$$D_{\frac{5k-1}{5}}\left(x^5\right) = 1 + \sum_{l=1}^{k}\left(x^{5l} + x^{-5l}\right) + \sum_{l=1}^{k}\left(x^{5l-1} + x^{-(5l-1)}\right). \tag{10}$$

Nonetheless, we have proved in Theorem 1 that in fact this is an ordinary $(2d+1)$-CCA in the variable $y = x^5$ given by:

$$D_{\frac{5k-1}{5}}\left(x^5\right) \equiv D_{10k}(x)^{\mathbb{Z}/5} \equiv D_{2k}\left(x^5\right) = 1 + \sum_{i=1}^{2k}\left(x^{5i} + x^{-5i}\right). \tag{11}$$

Let us work out two explicit cases to see the equivalence of Eqs. (10) and (11) modulo $x^n - 1$.
If $n = 6 = 5k+1$ then $k = 1$ and thus Eq. (11) gives

$$D_{\frac{4}{5}}\left(x^5\right) \equiv D_{10}(x)^{\mathbb{Z}/5} \equiv D_2\left(x^5\right) = 1 + x^5 + x^{-5} + x^{10} + x^{-10},$$

which is equivalent, modulo $x^6 - 1$, to

$$1 + x^5 + x^{-5} + x^4 + x^{-4} = D_{\frac{4}{5}}\left(x^5\right),$$

that is, the result of Eq. (10).
If $n = 11 = 5k+1$ then $k = 2$ and thus Eq. (11) gives

$$\begin{aligned} D_{\frac{9}{5}}\left(x^5\right) &\equiv D_{20}(x)^{\mathbb{Z}/5} \equiv D_4\left(x^5\right) = \\ &= 1 + x^5 + x^{-5} + x^{10} + x^{-10} + x^{15} + x^{-15} + x^{20} + x^{-20}, \end{aligned}$$

which is equivalent, modulo $x^{11} - 1$, to

$$1 + x^5 + x^{-5} + x^{10} + x^{-10} + x^4 + x^{-4} + x^9 + x^{-9} = D_{\frac{9}{5}}\left(x^5\right),$$

that is, the result of Eq. (10).

- The cases $r = 2$ and $\bar{r} = 3$ are easily described by Corollary 2:

$$D^k_{5k}(x) = D_k\left(x^5\right) = 1 + \sum_{i=1}^{k}\left(x^{5i} + x^{-5i}\right),$$

that is, the inverse has $2k$ symmetric active neighbours fitted in a radius of $5k$ and regularly distributed by jumps of 5 slots.

- For $\bar{r} = 4$ then (see [5, Section 5]) $r = 1, r\prime = 2$ and $\bar{t} = 1$, so that $\bar{b} = 5k+1$ and $\bar{d} = 2k+1$ and by Proposition 2 we have that the inverse is the virtual dipolynomial

$$D^{2k+1}_{5k+1}(x) = D_{\frac{5k+1}{5}}\left(x^5\right),$$

so that the inverse has $2\bar{d} = 4k+2$ symmetric active neighbours fitted in a virtual radius of $\bar{b} = 5k+1$, but in order to know how these active neighbours are located we need, a priori, its explicit expression as a dipolynomial given by Eq. (8) (proved in [5, Theorem 1]):

$$D_{\frac{5k+1}{5}}\left(x^5\right) = 1 + \sum_{l=1}^{k}\left(x^{5l} + x^{-5l}\right) + \sum_{l=1}^{k}\left(x^{5l-4} + x^{-(5l-4)}\right) + x^{5k+1} + x^{-(5k+1)}. \tag{12}$$

Nonetheless, we have prove in Theorem 2 that in fact this is an ordinary $(2\bar{d}+1)$-CCA in the variable $y = x^5$ given by:

$$D_{\frac{5k+1}{5}}\left(x^5\right) \equiv D_{(2k+1)5}(x)^{\mathbb{Z}/5} \equiv D_{2k+1}\left(x^5\right) = 1 + \sum_{i=1}^{2k+1}\left(x^{5i} + x^{-5i}\right). \tag{13}$$

Let us work out two explicit cases to see the equivalence of Eqs. (12) and (13) modulo $x^n - 1$.
If $n = 9 = 5k+4$ then $k = 1, \bar{b} = 6, \bar{d} = 3$ and thus Eq. (13) gives

$$D_{\frac{6}{5}}\left(x^5\right) \equiv D_{15}(x)^{\mathbb{Z}/5} \equiv D_3\left(x^5\right) = 1 + x^5 + x^{-5} + x^{10} + x^{-10} + x^{15} + x^{-15},$$

which is equivalent, modulo $x^9 - 1$, to

$$1 + x^5 + x^{-5} + x + x^{-1} + x^6 + x^{-6} = D_{\frac{6}{5}}\left(x^5\right),$$

that is, the result of Eq. (12).
If $n = 14 = 5k+4$ then $k = 2, \bar{b} = 11, \bar{d} = 5$ and thus Eq. (13) gives

$$\begin{aligned} &D_{\frac{11}{5}}\left(x^5\right) \equiv D_{25}(x)^{\mathbb{Z}/5} \equiv D_5\left(x^5\right) = \\ &= 1 + x^5 + x^{-5} + x^{10} + x^{-10} + x^{15} + x^{-15} + x^{20} + x^{-20} + x^{25} + x^{-25}, \end{aligned}$$

which is equivalent, modulo $x^{14} - 1$, to

$$1+x^5+x^{-5}+x^{10}+x^{-10}+x+x^{-1}+x^6+x^{-6}+x^{11}+x^{-11}=D_{\frac{11}{5}}(x^5),$$

that is, the result of Eq. (12).

6. Algorithms of recursive estimation of neighbours

In [12] an artificial life algorithm was developed to study the motion of a flock of birds. The motion of the simulated flock is created by a distributed model where each bird is an independent entity which moves according to its local perception, but there is no a central entity controlling the boid. Among other applications, this leads to the study of collective control of group motion like herds of animals or robots, traffic patterns or 3D patterns in animation programming. To study the relationship between this and the information processing and fluctuation control in CA pattern formation, in [6] an extended CA is introduced using a recursive estimation of neighbours (REN) algorithm. This extended CA have an extra radius used to define the local perception area and a base rule is used to recursively estimate the neighbours's states.

We will see that if one uses the REN algorithm proposed in [6] with base rule the ECA150, one can recursively produce CA rules which are examples of the virtual automata (b,d)-VCCA defined in Section 4. Thus, we are proposing the virtual radius d as a different extra radius to codify the local perception, and then Theorem 1 not only says that the inverse of a reversible $(2R+1)$-CCA is a (b,d)-VCCA, but also that (in the appropriated coordinate) the inverse can be written as a regular CCA having neighbourhood radius the virtual perception radius d.

Moreover, we will introduce a different REN algorithm which allow us to prove that the $(2R+1)$-CCA can be shown to be a recursive sequence parametrised by R and originated from the base rule ECA150, thus we could have certain control on the fluctuation in this particular group formation.

6.1. The REN algorithm

Let us denote the ECA150 by $\#150R1$ and call it the base rule. The recursive estimation of neighbors (REN) algorithm proposed in [6], and used with the ECA150 as a base rule, is:

$$\phi_{R,i}^{t+1}:=f^{150}\left(\phi_{R-1,i-1}^{t+1},s_i^t,\phi_{R-1,i+1}^{t+1}\right), \tag{14}$$

where $\phi_{R,i}^{t+1}$ is the i-th cell in a neighbourhood of radius R at evolution time $t+1$ and $\phi_{R-1,i\pm1}^{t+1}$ indicates the estimate states of neighbours at $t+1$ with an estimated radius $R-1$. They have:

$$\phi_{R-j,i-j}^{t+1}=f^{150}\left(\phi_{R-j-1,i-j-1}^{t+1},s_{i-j}^t,\phi_{R-j-1,i-j+1}^{t+1}\right)$$

$$\phi_{R-j,i+j}^{t+1}=f^{150}\left(\phi_{R-j-1,i+j-1}^{t+1},s_{i+j}^t,\phi_{R-j-1,i+j+1}^{t+1}\right)$$

for $j=1,2,\ldots,R-1$, and

$$\phi_{0,i\pm R}^{t+1}=x_{i\pm R}^t,\qquad \phi_{0,i\pm R\mp2}^{t+1}=x_{i\pm R\mp2}^t.$$

When $R=1$ this is just the 150ECA and using the formulas we can easily perform higher radius cases. For $R=2$ we end up with the penta-cyclic rule

$$\phi_{2,i}^{t+1}=s_{i-2}^t\oplus s_{i-1}^t\oplus s_i^t\oplus s_{i+1}^t\oplus s_{i+2}^t,$$

which is the 5-CCA or, analogously, the (b,d)-VCCA defined by $b=d=2$.

The case $R=3$ gives:

$$\phi_{3,i}^{t+1}=s_{i-3}^t\oplus s_{i-2}^t\oplus s_i^t\oplus s_{i+2}^t\oplus s_{i+3}^t,$$

which is the $(3,2)$-VCCA having a neighbourhood radius $b=3$ and a virtual radius $d=2$ (that is, $2d+1=5$ active, and symmetric, cells in a total of $2b+1=7$ cells). Notice that this is the xor summation of the 7-CCA plus the ECA90.

The case $R=4$ gives:

$$\phi_{4,i}^{t+1}=s_{i-4}^t\oplus s_{i-3}^t\oplus s_{i-2}^t\oplus s_i^t\oplus s_{i+2}^t\oplus s_{i+3}^t\oplus s_{i+4}^t,$$

which is the $(4,3)$-VCCA having a neighbourhood radius $b=4$ and a virtual radius $d=3$. Notice that this is the xor summation of the 9-CCA plus the ECA90.

The case $R=5$ gives:

$$\phi_{5,i}^{t+1}=s_{i-5}^t\oplus s_{i-4}^t\oplus s_{i-2}^t\oplus s_i^t\oplus s_{i+2}^t\oplus s_{i+4}^t\oplus s_{i+5}^t,$$

which is the $(5,3)$-VCCA having a neighbourhood radius $b=5$ and a virtual radius $d=3$.

6.2. A different REN algorithm

We shall show that, with a different REN algorithm to that of [6], we can use the ECA150 to recursively generate the $(2R+1)$-CCA rules parametrised by the neighbourhood radius R. The REN algorithm we propose is given by:

$$\varphi_{R,i}^{t+1} := f^{150}\left(s_{i-R}^{t}, \varphi_{R-1,i}^{t+1}, s_{i+R}^{t}\right), \tag{15}$$

where $\varphi_{R,i}^{t+1}$ is an estimated state of the i-th cell with radius R at evolution time $t+1$. Notice that $\varphi_{R,i}^{t+1} = s_i^{t+1}$ and that applying the definition for the $R = 1$ case (which is the ECA150, the base rule $\#150R1$):

$$\varphi_{1,i}^{t+1} = f^{150}\left(s_{i-1}^{t}, \varphi_{0,i}^{t+1}, s_{i+1}^{t}\right) = s_{i-1}^{t} \oplus \varphi_{0,i}^{t+1} \oplus s_{i+1}^{t}$$

we have that $\varphi_{0,i}^{t+1} = s_i^t$.

The $R = 2$ case corresponds to the 5-neighbour rule:

$$s_i^{t+1} = f^{150}\left(s_{i-2}^{t}, \varphi_{1,i}^{t+1}, s_{i+2}^{t}\right) = s_{i-2}^{t} \oplus s_{i-1}^{t} \oplus s_{i}^{t} \oplus s_{i+1}^{t} \oplus s_{i+2}^{t}. \tag{16}$$

Therefore, the $\#150R2$ recursive rule corresponds again to the penta-cyclic ECA. Now, straight forward computations shows that:

Proposition 3. The recursive sequence, represented by:

$$[150] = \{\#150R1, \#150R2, \ldots\},$$

is the equivalent to the family of $(2R+1)$-CCA with periodic boundary conditions.

7. Wolfram based classification for $(2R+1)$-cyclic ECA

The REN algorithm of Eq. (15) suggests that the $(2R+1)$-cyclic cellular automata might be defined by a Wolfram type rule obtained recursively by the base rule $\#150R1$. In this section we will define a Recursive Wolfram number (RWN) for $(2R+1)$-CCA and we will proof that this RWN is recursively generated by both the Wolfram number of the ECA150 and that of its complementary rule ECA105.

Recall that the ECA150 is defined by means of the following local transition function:

$$s_i^{t+1} = f^{150}(s_{i-1}^t, s_i^t, s_{i+1}^t) = s_{i-1}^t \oplus s_i^t \oplus s_{i+1}^t, \quad 1 \leqslant i \leqslant n. \tag{17}$$

and notice that its Wolfram number 150 is obtained by the binary expression:

$$w_{150} = 2 + 2^2 + 2^4 + 2^7 = (0,1,1,0,1,0,0,1).$$

The ECA150 has the ECA105 as its complementary rule (this late one beeing also a Class III rule), which is defined by the local transition function:

$$s_i^{t+1} = f^{105}(s_{i-1}^t, s_i^t, s_{i+1}^t) = 1 \oplus s_{i-1}^t \oplus s_i^t \oplus s_{i+1}^t, \quad 1 \leqslant i \leqslant n,$$

and its Wolfram number is obtained by:

$$w_{105} = 2^0 + 2^3 + 2^5 + 2^6 = (1,0,0,1,0,1,1,0).$$

Since ECA105 is the complementary rule of ECA150, let us use the notation $w_{105} = w_{150}^c$.

We start by analyzing the $R = 2$ case, which is the penta-cyclic CA defined by the local transition function of Eq. (16) and that we write here as:

$$s_i^{t+1} = f^{\#150R2}(s_{i-2}^t, s_{i-1}^t, s_i^t, s_{i+1}^t, s_{i+2}^t) = s_{i-2}^t \oplus s_{i-1}^t \oplus s_i^t \oplus s_{i+1}^t \oplus s_{i+2}^t.$$

There are 2^{2^5} possible elementary rules leading to 4.294.967.296 cellular automata. Each CA is indexed by its rule number $w = \sum_{i=0}^{31} \alpha_i \cdot 2^i$, where $\alpha_i = f^{\#150R2}(i_2)$ and i_2 stands the binary expression of $0 \leqslant i \leqslant 31$:

$$i = a_0 2^0 + a_1 2^1 + a_2 2^2 + a_3 2^3 + a_4 2^4 \quad \Longleftrightarrow \quad i_2 = (a_0, a_1, a_2, a_3, a_4).$$

That is, $\alpha_i = f^{\#150R2}(i_2) = a_0 \oplus a_1 \oplus a_2 \oplus a_3 \oplus a_4$. Thus, for the penta-cyclic CA $\#150R2$ we have that the Wolfram number is:

$$\begin{aligned} w_{\#150R2} &= \sum_{i=0}^{31} \alpha_i \cdot 2^i = \sum_{i=0}^{31} f^{\#150R2}(i_2) \cdot 2^i = \\ &= 2 + 2^2 + 2^4 + 2^7 + 2^8 + 2^{11} + 2^{13} + 2^{14} + 2^{16} + \\ &+ 2^{19} + 2^{21} + 2^{22} + 2^{25} + 2^{26} + 2^{28} + 2^{31} = 2.523.490.710 \end{aligned}$$

Notice that this big number can be grouped and rewritten as:

$$\begin{aligned} w_{\#150R2} &= 2 + 2^2 + 2^4 + 2^7 + 2^8\left(2^0 + 2^3 + 2^5 + 2^6\right) + \\ &+ 2^{16}\left(2^0 + 2^3 + 2^5 + 2^6\right) + 2^{24}\left(2 + 2^2 + 2^4 + 2^7\right) = \\ &= w_{150} + 2^8 w_{105} + 2^{16} w_{105} + 2^{24} w_{150} = \\ &= w_{150} + 2^{2^3} w^c_{150} + 2^{2 \cdot 2^3} w^c_{150} + 2^{3 \cdot 2^3} w_{150} \end{aligned}$$

For $R = 3$ we have the hepta-cyclic CA defined by the local transition function:

$$\begin{aligned} s_i^{t+1} &= f^{\#150R3}\left(s^t_{i-3}, s^t_{i-2}, s^t_{i-1}, s^t_i, s^t_{i+1}, s^t_{i+2}, s^t_{i+3}\right) = \\ &= s^t_{i-3} \oplus s^t_{i-2} \oplus s^t_{i-1} \oplus s^t_i \oplus s^t_{i+1} \oplus s_{i+2} \oplus s^t_{i+3} \end{aligned}$$

and it has the following RWN:

$$\begin{aligned} w_{\#150R3} &= \sum_{i=0}^{2^7-1} f^{\#150R3}(i_2) \cdot 2^i = w_{150} + 2^8 w^c_{150} + 2^{16} w^c_{150} + 2^{24} w_{150} + \\ &+ 2^{32} w^c_{150} + 2^{40} w_{150} + 2^{48} w_{150} + 2^{56} w^c_{150} + \\ &+ 2^{64} w^c_{150} + 2^{72} w_{150} + 2^{80} w_{150} + 2^{88} w^c_{150} + \\ &+ 2^{96} w_{150} + 2^{104} w^c_{150} + 2^{16} w^c_{112} + 2^{120} w_{150} = \\ &= \left(w_{150} + 2^{2^3} w^c_{150} + 2^{2 \cdot 2^3} w^c_{150} + 2^{3 \cdot 2^3} w_{150}\right) + \\ &+ 2^{32}\left(w^c_{150} + 2^{2^3} w_{150} + 2^{2 \cdot 2^3} w_{150} + 2^{3 \cdot 2^3} w^c_{150}\right) + \\ &+ 2^{2 \cdot 32}\left(w^c_{150} + 2^{2^3} w_{150} + 2^{2 \cdot 2^3} w_{150} + 2^{3 \cdot 2^3} w^c_{150}\right) + \\ &+ 2^{3 \cdot 32}\left(w_{150} + 2^{2^3} w^c_{150} + 2^{2 \cdot 2^3} w^c_{150} + 2^{3 \cdot 2^3} w_{150}\right) = \\ &= w_{\#150R2} + 2^{2^5} w^c_{\#150R2} + 2^{2 \cdot 2^5} w^c_{\#150R2} + 2^{3 \cdot 2^5} w_{\#150R2}. \end{aligned}$$

Where the RWN $w^c_{\#150R2}$ is given by:

$$w^c_{\#150R2} = w^c_{\#105R2} := \sum_{i=0}^{2^5-1} f^{\#105R2}(i_2) \cdot 2^i = w^c_{150} + 2^{2^3} w_{150} + 2^{2 \cdot 2^3} w_{150} + 2^{3 \cdot 2^3} w^c_{150}.$$

Let us now state the general case.

Definition 3. Given the local transition function:

$$s_i^{t+1} = f^{\#150RR}\left(s^t_{i-R}, \ldots, s^t_{i-1}, s^t_i, s^t_{i+1}, \ldots, s^t_{i+R}\right) = s^t_{i-R} \oplus \cdots \oplus s^t_i \oplus \cdots \oplus s^t_{i+R}$$

and its complementary rule:

$$s_i^{t+1} = f^{\#105RR}\left(s^t_{i-R}, \ldots, s^t_{i-1}, s^t_i, s^t_{i+1}, \ldots, s^t_{i+R}\right) = 1 \oplus s^t_{i-R} \oplus \cdots \oplus s^t_i \oplus \cdots \oplus s^t_{i+R},$$

let us write $N_R = 2R + 1$ and let $i_2 = (a_0, a_1, \ldots, a_{2R-1}, a_{2R})$ be the binary representation of a natural number i such that $0 \leqslant i \leqslant 2^{N_R} - 1$. We define the following Recursive Wolfram Numbers:

$$w_{\#150RR} := \sum_{i=0}^{2^{N_R}-1} f^{\#150RR}(i_2) \cdot 2^i; \qquad w^c_{\#150RR} := \sum_{i=0}^{2^{N_R}-1} f^{\#105RR}(i_2) \cdot 2^i$$

We have the following result.

Proposition 4. The recursive Wolfram number for the $(2R+1)$-cyclic CA given by the local transition function $f^{\#150RR}$ is recursively generated by the Wolfram number of ECA150 by the following formula:

$$w_{\#150RR} = w_{\#150R(R-1)} + 2^{2^{N_{R-1}}} w^c_{\#150R(R-1)} + 2^{2 \cdot 2^{N_{R-1}}} w^c_{\#150R(R-1)} + 2^{3 \cdot 2^{N_{R-1}}} w_{\#150R(R-1)}$$

where $N_{R-1} = 2(R-1) + 1 = 2R - 1$.

Proof. We have that

$$w_{\#150RR} := \sum_{i=0}^{2^{N_R}-1} f^{\#150RR}(i_2)\cdot 2^i = \sum_{i=0}^{2^{N_{R-1}}-1} f^{\#150RR}(i_2)\cdot 2^i + \\ + \sum_{i=2^{N_{R-1}}}^{2\cdot 2^{N_{R-1}}-1} f^{\#150RR}(i_2)\cdot 2^i + \overset{3\cdot 2^{N_{R-1}}-1}{\underset{i=2\cdot 2^{N_{R-1}}}{m}} f^{\#150RR}(i_2)\cdot 2^i + \sum_{i=3\cdot 2^{N_{R-1}}}^{2^{N_R}-1} f^{\#150RR}(i_2)\cdot 2^i,$$

Where $i_2 = (a_0, a_1, \ldots, a_{2R-1}, a_{2R})$ is the binary representation of $0 \leqslant i \leqslant 2^{N_R} - 1$. Let us study the following cases:

- If $0 \leqslant i \leqslant 2^{N_{R-1}} - 1 = 2^{2R-1} - 1$, we have that $a_{2R-1} = a_{2R} = 0$ and then $i_2 = (a_0, \ldots, a_{2R-2}, 0, 0)$. Therefore:
  $$f^{\#150RR}(i_2) = a_0 \oplus \cdots \oplus a_{2(R-1)} = f^{\#150R(R-1)}(i_2)$$
- If $2^{N_{R-1}} = 2^{2R-1} \leqslant i \leqslant 2\cdot 2^{N_{R-1}} - 1 = 2^{2R} - 1$, we have that $a_{2R-1} = 1$ and $a_{2R} = 0$ and then $i_2 = (a_0, \ldots, a_{2R-2}, 1, 0)$. Therefore:
  $$f^{\#150RR}(i_2) = 1 \oplus a_0 \oplus \cdots \oplus a_{2(R-1)} = f^{\#105R(R-1)}(i_2)$$
- If $2\cdot 2^{N_{R-1}} = 2^{2R} \leqslant i \leqslant 3\cdot 2^{N_{R-1}} - 1 = 2^{2R} + 2^{2R-1} - 1$, we have that $a_{2R-1} = 0$ and $a_{2R} = 1$ and then $i_2 = (a_0, \ldots, a_{2R-2}, 0, 1)$. Therefore:
  $$f^{\#150RR}(i_2) = 1 \oplus a_0 \oplus \cdots \oplus a_{2(R-1)} = f^{\#105R(R-1)}(i_2)$$
- If $3\cdot 2^{N_{R-1}} = 2^{2R} + 2^{2R-1} \leqslant i \leqslant 3\cdot 2^{N_R} - 1 = 2^{2R+1} - 1$, we have that $a_{2R-1} = 1$ and $a_{2R} = 1$ and then $i_2 = (a_0, \ldots, a_{2R-2}, 1, 1)$. Therefore:
  $$f^{\#150RR}(i_2) = 1 \oplus 1 \oplus a_0 \oplus \cdots \oplus a_{2(R-1)} = a_0 \oplus \cdots \oplus a_{2(R-1)} = f^{\#150R(R-1)}(i_2)$$

And thus we conclude:

$$\begin{aligned} w_{\#150RR} = & \sum_{i=0}^{2^{N_{R-1}}-1} f^{\#150R(R-1)}(i_2)\cdot 2^i + \sum_{i=2^{N_{R-1}}}^{2\cdot 2^{N_{R-1}}-1} f^{\#105R(R-1)}(i_2)\cdot 2^i + \\ & + \sum_{i=2\cdot 2^{N_{R-1}}}^{3\cdot 2^{N_{R-1}}-1} f^{\#105R(R-1)}(i_2)\cdot 2^i + \sum_{i=3\cdot 2^{N_{R-1}}}^{2^{N_R}-1} f^{\#150R(R-1)}(i_2)\cdot 2^i = \\ & = \sum_{i=0}^{2^{N_{R-1}}-1} f^{\#150R(R-1)}(i_2)\cdot 2^i + 2^{2^{N_{R-1}}} \sum_{i=0}^{2^{N_{R-1}}-1} f^{\#105R(R-1)}(i_2)\cdot 2^i + \\ & + 2^{2\cdot 2^{N_{R-1}}} \sum_{i=0}^{2^{N_{R-1}}-1} f^{\#105R(R-1)}(i_2)\cdot 2^i + 2^{3\cdot 2^{N_{R-1}}} \sum_{i=0}^{2^{N_{R-1}}-1} f^{\#150R(R-1)}(i_2)\cdot 2^i = \\ & = w_{\#150R(R-1)} + 2^{2^{N_{R-1}}} w^c_{\#150R(R-1)} + 2^{2\cdot 2^{N_{R-1}}} w^c_{\#150R(R-1)} + 2^{3\cdot 2^{N_{R-1}}} w_{\#150R(R-1)}. \end{aligned}$$

□

8. Conclusions

We have developed a mathematical framework to introduce virtual cyclic cellular automata as a certain class of cellular automata having active and non active cells at specific steps of times, and also reflecting certain periodicity. We have show that the inverse of a reversible $(2R+1)$-cyclic cellular automaton with periodic boundary conditions is a virtual cyclic cellular automaton, and using algebro-geometric tools, the inverse transition dipolynomial is characterised as an invariant dipolynomial under the action of a finite cyclic group, which moreover simplifies the main results of [5] and reduces its computational implementation. Secondly, we have used a recursive estimation of neighbours (REN) algorithm to produce other examples of virtual cyclic cellular automata, which generalize the CA used in [6,12] to study, among others, the relationship involving herds and information processing. We hope that this will open a door to using VCCA in applications similar to those given in these references, such as the collective control of entities or traffic patterns. This line of research will be conducted elsewhere. We have also proposed a new REN algorithm which presents the $(2R+1)$-cyclic cellular automata as a recursive family of cyclic cellular automata parametrised by R, and which is originated from the elementary cellular automaton with base rule 150. This reinterpretation of a $(2R+1)$-CCA allow us to define a new notion of recursive Wolfram number for a $(2R+1)$-CCA, and show that it can be recursively computed using the new REN algorithm applied to the base rule 150 and its complementary 105 rule.

CRediT authorship contribution statement

D. Hernández Serrano: Conceptualization, Methodology, Validation, Formal-analysis, Investigation, Resources, Writing-original-draft, Writing-review-editing, Visualization, Supervision, Project-administration. **A. Martín del Rey:** Validation, Formal-analysis, Investigation.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work has been supported by Ministerio de Economía y Competitividad (Spain) and the European Union through FEDER funds under grant MTM2017-86042-P, the project STAMGAD 18.J445/ 463AC03 supported by Consejería de Educación (GIR, Junta de Castilla y León, Spain) and the project PIC2-2021-10 supported by Universidad de Salamanca (Spain).

References

[1] P. Chaudhuri, D. Chowdhury, S. Nandi, S. Chattopadhyay, Additive cellular automata, Theory and Applications, vol. 1, IEEE Computer Society Press, Los Alamitos, 1997.
[2] I.B. Collings, I. Vaughan, L. Clarkson, A low-complexity lattice-based low-PAR transmission scheme for DSL channels, IEEE Trans. Commun. 52 (2004) 755–764.
[3] L. Fuyong, The inverse of circulant matrix, Appl. Math. Comput. 217 (1) (2011) 8495–8503.
[4] L. Hernández Encinas, A. Martín del Rey, Inverse rules of ECA with rule number 150, Appl. Math. Comput. 189 (2007) 1782–1786.
[5] D. Hernández Serrano, A. Martín del Rey, A closed formula for the inverse of a cellular automaton with $(2R+1)$-cyclic rule, to appear in Appl. Math. Comput (2019).
[6] Y. Kayama, Extension of cellular automata via the introduction of an algorithm for the recursive estimation of neighbors, Artif. Life Robot. 21 (3) (2016) 338–344.
[7] A. Martín del Rey, A note on the reversibility of the elementary cellular automaton with rule number 90, Rev. Union Mat. Argent. 56 (1) (2015) 107–125.
[8] A. Martín del Rey, G. Rodríguez Sánchez, On the reversibility of 150 Wolfram cellular automata, Int. J. Mod. Phys. C 17 (2006) 975–984.
[9] A. Martín del Rey, G. Rodríguez Sánchez, On the invertible cellular automata 150 over $\mathbb{F}_p$, Appl. Math. Comput. 219 (10) (2013) 5427–5432.
[10] A. Martín del Rey, G. Rodríguez Sánchez, Reversible cellular automaton with rule number 150 and periodic boundary conditions over Fp, Int. J. Mod. Phys. C 26 (11) (2015) 1550120.
[11] A. Nobe, F. Yura, On reversibility of cellular automata with periodic boundary conditions, J. Phys. A: Math. Gen. 37 (2004) 5789–5804.
[12] C.A. Reynolds, Flocks, herds and schools: a distributed behavioral model, SIGGRAPH Comput. Graph. 21 (4) (1987) 25–34.
[13] J.C. Seck-Tuoh-Mora, G.J. Martínez, R. Alonso-Sanz, N. Hernández-Romero, Invertible behavior in elementary cellular automata with memory, Inf. Sci. 199 (2012) 125–132.
[14] B. Sethi, N. Fatès, S. Das, Reversibility of Elementary Cellular Automata under Fully Asynchronous Update, in: T.V. Gopal, M. Agrawal, A. Li, S.B. Cooper (Eds.), Proceedings of the International Conference on Theory and Applications of Models of Computation, Lecture Notes in Computer Science, vol. 8402, Springer International Publishing, 2014, pp. 39–49.
[15] S.-Q. Shen, J.-M. Cen, Y. Hao, On the determinants and inverses of circulant matrices with fibonacci and lucas numbers, App. Math. Comput. 217 (23) (2011) 9790–9797.
[16] I. Siap, H. Akin, M.E. Koroglu, The reversibility of $(2r+1)$-cyclic rule cellular automata, TWMS J. Pure Appl. Math. 4 (2013) 215–225.
[17] J. Wang, C. Dong, Inverse matrix of symmetric circulant matrix on skew field, Int. J. Algebra 1 (2007) 541–546.
[18] S. Wolfram, Cellular Automata and Complexity, Westview Press, Collected Papers, 1994.
[19] S. Wolfram, A New Kind of Science, Wolfram Media Inc., 2002.
[20] Y. Yazlik, N. Taskara, On the inverse of circulant matrix via generalized k-horadam numbers, App. Math. Comput. 223 (2013) 191–196.