# Algebraic Properties of Linear Cellular Automata

L. Le Bruyn
M. Van den Bergh*
University of Antwerp (UIA)
Dept of Math. and Comp. Science
Universiteitsplein 1
2610 Wilrijk, Belgium

90-15

**Abstract**

In this paper we present an algebraic formalism for dealing with cellular automata whose local transition rule satisfies an additivity property. We discuss the phenomen of self-replication; the connection with higher order cellular automata and the state transition graph.

## 1 Introduction

Cellular automata are structures which evolve on a finite dimensional lattice according to a deterministic local law. They were first introduced by J. Von Neumann [4] and S. Ulam [7] as examples of simple structures presenting some of the features of life. Recently, there is a strong impetus to reconsider these automata coming from artificial intelligence and parallel computing on the one hand and their suitability to simulate complex physical phenomena on the other hand. For more details and motivation we refer the reader to [8].

Some cellular automata have a simplifying additivity property, that is, their local transition function is linear. Some of the properties of these so called linear cellular automata were investigated in a paper by Martin, Odlyzko and Wolfram [5]. Although such cellular automata are rather special, they are expected to provide useful models for the understanding of more complex, nonlinear cellular automata.

In this paper we provide the natural algebraic setting for studying more general linear cellular automata. The main difference with the previous definition is that we allow the cellprocessor memory to be a finite dimensional vectorspace over a finite field $\mathbb{F}_q$ (rather than restricting to the one-dimensional case studied before). Not only this new definition gives an abundancy of new examples

*The authors are supported by an NFWO-grant

and new phenomena of linear cellular automata, but they occur also naturally in the study of higher order linear cellular automata as in [6]. This allows us to explain in a natural way the construction of E. Fredkin and N. Margolus of linear higher order reversible automata (and to create many new examples of reversible cellular automata) as well as to describe the state transition graph of these higher order cellular automata.

In section two we describe the formalism of linear cellular automata and how they can be described algebraically. As an application we explain the self-replication phenomen for linear cellular automata. In section three we prove that the class of linear cellular automata is closed under taking higher order cellular automata (i.e. the local transition rule depends not only on the present state, but also on a set of previous states) and we give a classification of reversible linear cellular automata. In section four we give a complete description of the state transition graph of a linear cellular automaton. Most of the information about it seems to be hidden in the characteristic polynomial of a matrix over a Laurent field extension over the finite field associated to the local state transition rule.

In this paper we have restricted attention to the case of infinite linear cellular automata. The particularly interesting case which occurs when we impose boundary conditions (null or periodic) will be treated in a forthcoming paper.

## 2 The formalism of linear cellular automata

In this section we will outline the formalism which enables us to determine the evolution of certain cellular automata satisfying superposition principles. Before giving the formal definition of such 'linear cellular automata' let us give a few easy examples which have attracted some interest :

**Example 2.1** *Consider the one-dimensional cellular automaton studied extensively in [8] in which each cellprocessor has one bit of memory and which performs the exclusive or operation on the previous state of its left and right neighbor synchronously on each time step*

$$\cdots \mid x_{i-1} \mid \quad x_i \quad \mid x_{i+1} \mid \cdots$$

$$x_i(t+1) \cong x_{i-1}(t) + x_{i+1}(t) \mod 2$$

*This cellular automaton corresponds to 'rule 90' of the S. Wolfram classification scheme, [8].*

**Example 2.2** *E. Franklin devised a very simple system capable of selfreproduction. He considered a two-dimensional cellular automaton such that each*

*cellprocessor has one bit of memory and performs the nim sum of its four orthogonal neighbors :*

| | | | | |
|---|---|---|---|---|
| $\cdots$ | $x_{i-1,j+1}$ | $x_{i,j+1}$ | $x_{i+1,j+1}$ | $\cdots$ |
| $\cdots$ | $x_{i-1,j}$ | $x_{i,j}$ | $x_{i+1,j}$ | $\cdots$ |
| $\cdots$ | $x_{i-1,j-1}$ | $x_{i,j-1}$ | $x_{i+1,j-1}$ | $\cdots$ |

$$x_{i,j}(t+1) \cong x_{i,j+1} + x_{i+1,j} + x_{i,j-1} + x_{i-1,j} \mod 2$$

These cellular automata are very special in that the logic rules are linear. Such cellular automata are, however, expected to provide useful models for the understanding of more complex, nonlinear cellular automata.

Although the formal definition given below can be readily extended to cellular automata defined over an arbitrary finite commutative ring, we restrict ourselves in this paper to the case that this ring is a finite field $\mathbb{F}_q$ on $q = p^m$ elements where $p$ is a prime number, the characteristic of the field. For more details on finite fields, the reader is referred to [2].

**Definition 2.3** *A linear cellular automaton $\Sigma$ of type $(k, l, m, n, p)$ is a cellular automaton such that*

1. *$k$ is the dimension of the cellular space, that is, each cell is uniquely determined by a $k$-tuple of integers $\alpha = (a_1, \ldots, a_k) \in \mathbb{Z}^k$.*
2. *$l$ is the number of neighbors, that is, a choice $\Delta = \{\delta_1, \ldots, \delta_l\}$ of $l$ elements from $\mathbb{Z}^k$ such that the neighborhood of a cell $\alpha$ are the cells $\{\alpha+\delta_1, \ldots, \alpha+\delta_l\}$.*
3. *$\mathbb{F}_q$ where $q = p^m$ is the field of definition, that is, an elementary unit of information is an element from $\mathbb{F}_q$ (which we will call a 'qit')*
4. *$n$ is the number of qits of cellprocessor memory, that is, the state of the cellprocessor $\alpha$ at time $t$ is an $n$-tuple of elements from $\mathbb{F}_q$ which we will represent by a column vector and denote by $x_\alpha(t)$*
5. *the local state transition function is linear, that is, there exist $l$ square $n$ by $n$ matrices over $\mathbb{F}_q$ say $\mathcal{A} = \{A_1, \ldots, A_l\}$ such that for each cell*

$$x_\alpha(t+1) = A_1 x_{\alpha+\delta_1}(t) + \cdots + A_l x_{\alpha+\delta_l}(t)$$

From now on, we will fix one linear cellular automaton $\Sigma$ of type $(k, l, m, n, p)$ (i.e. a particular choice of $\Delta$ and $\mathcal{A}$) and describe the formalism enabling us to calculate its evolution. Consider the ring

$$\mathbb{F}_q[X_1, X_1^{-1}, \ldots, X_k, X_k^{-1}] = \mathbb{F}_q[X_i, X_i^{-1}; i]$$

which is the localization of the ordinary polynomial ring $\mathbb{F}_q[X_1, \ldots, X_k]$ at the element $X_1 \cdots X_k$. That is, it consists of all elements $Y$ from the rational function field in $k$ variables $\mathbb{F}_q(X_1, \ldots, X_k)$ such that $(X_1 \cdots X_k)^u.Y \in \mathbb{F}_q[X_1, \ldots, X_k]$ for some natural number $u \in \mathbb{N}$.

For each cell $\alpha = (a_1, \ldots, a_k) \in \mathbb{Z}^k$ we define a unique monic monomial

$$X_\alpha = X_1^{a_1} \cdots X_k^{a_k} \in \mathbb{F}_q[X_i, X_i^{-1}; i]$$

In the rest of this section we aim to show that all information about the linear cellular automaton $\Sigma$ is contained in the $n$ by $n$ matrix

$$A_\Sigma = \sum_{i=1}^{l} A_i X_{-\delta_i} \in M_n(\mathbb{F}_q[X_i, X_i^{-1}; i])$$

Let $V$ be the standard $n$-dimensional vectorspace over $\mathbb{F}_q$ consisting of all 1 by $n$ column vectors, then any position can be described uniquely by an element in

$$V[X_i, X_i^{-1}; i] = V \otimes_{\mathbb{F}_q} \mathbb{F}_q[X_i, X_i^{-1}; i]$$

which is the standard free module of rank $n$ over $\mathbb{F}_q[X_i, X_i^{-1}]$. Namely, consider the finitely many cells $\alpha_1, \ldots, \alpha_z$ which are not in quiescent state (which we assume to be the zero vector) at time $t$, then the position of $\Sigma$ at time $t$ is fully described by the element

$$P(t) = \sum_{i=1}^{z} x_{\alpha_i}(t) X_{\alpha_i} \in V[X_i, X_i^{-1}]$$

Now, $n$ by $n$ matrices from $M_n(\mathbb{F}_q[X_i, X_i^{-1}])$ act on $V[X_i, X_i^{-1}]$ by left multiplication. The crucial observation to make is that the next generation is described by the element

$$P(t+1) = A_\Sigma.P(t) \in V[X_i, X_i^{-1}; i]$$

or, more generally, after $u$ clock pulses the configuration is described by the element

$$P(t+u) = A_\Sigma^u.P(t) \in V[X_i, X_i^{-1}; i]$$

Obviously, the formal simulation of the linear cellular automaton $\Sigma$ in the module $V[X_i, X_i^{-1}; i]$ reduces drastically the amount of computations required compared to direct simulation.

**Example 2.4** *Consider a one-dimensional linear cellular automaton $\Sigma$ such that each cellprocessor has two bits of memory with neighborhood $\Delta = \{-1, 1\}$ and the state transition function determined by the two 2 by 2 matrices*

$$A_1 = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} \qquad A_{-1} = \begin{pmatrix} 1 & 0 \\ 1 & 0 \end{pmatrix}$$

*That is, $\Sigma$ is of type $(1,2,1,2,2)$ and the corresponding matrix is*

$$A_\Sigma = \begin{pmatrix} X & 0 \\ X & X^{-1} \end{pmatrix} \in M_2(\mathbb{F}_2[X, X^{-1}])$$

*For example, the position*

| | $-2$ | $-1$ | $0$ | $1$ | $2$ | |
|---|---|---|---|---|---|---|
| $\cdots$ | $\begin{bmatrix}0\\0\end{bmatrix}$ | $\begin{bmatrix}1\\1\end{bmatrix}$ | $\begin{bmatrix}1\\0\end{bmatrix}$ | $\begin{bmatrix}1\\1\end{bmatrix}$ | $\begin{bmatrix}0\\0\end{bmatrix}$ | $\cdots$ |

*corresponds to the element*

$$\begin{pmatrix} X^{-1}+1+X \\ X^{-1}+X \end{pmatrix} \in \begin{pmatrix} \mathbb{F}_q[X_i, X_i^{-1}; i] \\ \mathbb{F}_q[X_i, X_i^{-1}; i] \end{pmatrix}$$

*Therefore, the next generation is obtained by multiplication*

$$\begin{pmatrix} X & 0 \\ X & X^{-1} \end{pmatrix} \begin{pmatrix} X^{-1}+1+X \\ X^{-1}+X \end{pmatrix} = \begin{pmatrix} 1+X+X^2 \\ X^{-2}+X+X^2 \end{pmatrix}$$

*which corresponds to the position*

| | $-2$ | $-1$ | $0$ | $1$ | $2$ | |
|---|---|---|---|---|---|---|
| $\cdots$ | $\begin{bmatrix}0\\1\end{bmatrix}$ | $\begin{bmatrix}0\\0\end{bmatrix}$ | $\begin{bmatrix}1\\0\end{bmatrix}$ | $\begin{bmatrix}1\\1\end{bmatrix}$ | $\begin{bmatrix}1\\1\end{bmatrix}$ | $\cdots$ |

As an application of the above formalism we give an explanation of the phenomen of self-replication that has been observed by various authors. See e.g. [8].

**Theorem 2.5** *Assume that $\Sigma$ has one bit of cel processor memory. Then for $k \gg 0$ the configuration of $\Sigma$ at $t = 2^k$ will consist of a number of (translated) copies of the original configuration ($t = 0$).*

**Proof** Recall that in a commutative ring with $p.1 = 0$

$$(a+b)^p = a^p + b^p$$

Since $\Sigma$ has only one bit of cell processor memory $A_i = 0, 1$. Therefore, by reducing $\Delta$ if necessary, we may assume that $A_i = 1$ for all $i$. Hence $A_\Sigma$ is the polynomial

$$\sum_i X^{-\delta_i}$$

The configuration of $\Sigma$ at time $t$ will be given by a polynomial

$$P(t) \in \mathbb{F}_q[X_i, X_i^{-1}; i]$$

Hence we obtain

$$P(2^k) = A_\Sigma^{2^k} P(0) = \sum_i X^{-\delta_i 2^k} P(0)$$

Now clearly $\sum_i X^{-\delta_i 2^k} P(0)$ corresponds to the sum of translated copies of the original configuration. If $2^k$ is large enough, these translated copies will have no living cells in common. This proves the theorem. ∎

# 3 Higher order and reversible linear cellular automata

Since the matrix $A_\Sigma \in M_n(\mathbb{F}_q[X_i, X_i^{-1}; i])$ satisfies its characteristic polynomial

$$\det(tI_n - A_\Sigma)$$

which is a monic polynomial of degree $n$ in $t$ with coefficients in $\mathbb{F}_q[X_i, X_i^{-1}; i]$, there is a fixed recursive relation between every $n$ succesive generations. For example 2.4 this relation is $P(t+2) - (X + X^{-1})P(t+1) + P(t) = 0$. One of the main motivations for studying linear cellular automata in the general setting of definition 2.3 rather than the special case when $n = 1$ is that this class is closed under higher order cellular automaton rules.

Normally, the rules for cellular automaton evolution take configurations to be determined solely from their immediate predecessors. One may, however, in general consider higher order cellular automaton rules, which allow dependence on say the $s$ preceding configurations. The state transition rule for such linear higher order cellular automata may be represented by the order $s$ recurrence relation

$$P(t) = \sum_{i=1}^{s} A_{\Sigma_j} P(t-j)$$

where all $A_{\Sigma_j} \in M_n(\mathbb{F}_q[X_i, X_i^{-1}; i])$. We then have the following

**Proposition 3.1** *A linear higher order cellular automaton can be simulated by a linear cellular automaton.*

**Proof** Let $\Sigma$ be the linear higher order cellular automaton determined by the above state transition function. We claim that we can represent it by a linear cellular automaton $\Gamma$ having $n.s$ qits of cellprocessor memory (i.e. each cell is capable of storing its $s$ previous states). A typical higher order cellular automaton configuration is therefore of the form

$$\begin{pmatrix} P(t-1) \\ P(t-2) \\ \cdots \\ P(t-s) \end{pmatrix} \in V^{\oplus s}[X_i, X_i^{-1}; i]$$

The state transition matrix $A_\Gamma$ of the linear cellular automaton $\Gamma$ is taken to be

$$A_\Gamma = \begin{pmatrix} A_{\Sigma_1} & A_{\Sigma_2} & \cdots & A_{\Sigma_{s-1}} & A_{\Sigma_s} \\ I_n & 0 & \cdots & 0 & 0 \\ 0 & I_n & \cdots & 0 & 0 \\ & & & & \\ 0 & 0 & \cdots & I_n & 0 \end{pmatrix} \in M_{ns}(\mathbb{F}_q[X_i, X_i^{-1}; i])$$

The linear cellular automaton $\Gamma$ simulates the linear higher order cellular automaton $\Sigma$. ∎

So, even if one is only interested in higher order linear cellular automata with one qit of cellprocessor memory one is naturally led to study the linear cellular automata as we defined them above.

Every configuration in a cellular automaton has a unique successor in time. If every configuration also has a unique predecessor, the cellular automaton is said to be reversible or invertible. Reversible systems are valuable models of computations since the information content of a patern of cells turns out to be a conserved quantity. In contrast to the linear cellular automata with one qit of cellprocessor memory which are irreversible except for trivial cases, there is an abundancy of reversible automata within our more general setting.

**Proposition 3.2** *A linear cellular automaton $\Sigma$ is reversible if and only if $A_\Sigma$ is an invertible matrix, that is $A_\Sigma \in GL_n(\mathbb{F}_q[X_i, X_i^{-1}; i])$ or equivalently* $\det(A_\Sigma)$ *is a monomial in $\mathbb{F}_q[X_i, X_i^{-1}; i]$.*

**Proof** If $A_\Sigma \in GL_n(\mathbb{F}_q[X_i, X_i^{-1}; i])$ then there is a matrix $B_\Sigma$ such that $A_\Sigma B_\Sigma = B_\Sigma A_\Sigma = I_n$. Given any configuration determined by its element $P(t) \in V[X_i, X_i^{-1}; i]$ one can find its direct predecessor by $P(t-1) = B_\Sigma.P(t) \in V[X_i, X_i^{-1}; i]$.

Conversely, suppose that $\Sigma$ is a reversible linear cellular automaton. Let $P_i$ be the configuration which is quiescent everywhere except for the zero cell where the state is $(0, \ldots, 0, 1, 0, \ldots, 0)^\tau$ with 1 on place $i$. Then, by assumption there exists a unique direct predecessor of $P_i$ say $Q_i$. Let $B_\Sigma$ be the $n$ by $n$ matrix whose $i$-th column is equal to $Q_i$ for all $i$. Then $A_\Sigma B_\Sigma = I_n$, that is, $A_\Sigma$ is invertible. ∎

Example 2.4 gives a onedimensional reversible linear cellular automaton. The matrix $B_\Sigma$ is in this case $\begin{pmatrix} X^{-1} & 0 \\ X & X \end{pmatrix}$ so the predecessor of the starting position can be found by

$$\begin{pmatrix} X^{-1} & 0 \\ X & X \end{pmatrix} \begin{pmatrix} X^{-1}+1+X \\ X^{-1}+X \end{pmatrix} = \begin{pmatrix} X^{-2}+X^{-1}+1 \\ X \end{pmatrix}$$

which corresponds to the position

| | −3 | −2 | −1 | 0 | 1 | 2 | |
|---|---|---|---|---|---|---|---|
| ⋯ | $\begin{bmatrix}0\\0\end{bmatrix}$ | $\begin{bmatrix}1\\0\end{bmatrix}$ | $\begin{bmatrix}1\\0\end{bmatrix}$ | $\begin{bmatrix}1\\0\end{bmatrix}$ | $\begin{bmatrix}0\\1\end{bmatrix}$ | $\begin{bmatrix}0\\0\end{bmatrix}$ | ⋯ |

An immediate consequence of this proposition is that the only reversible linear cellular automata with one bit of cellprocessor memory are the translations. A combination of the two foregoing results explains also the construction of E. Fredkin and N. Margolus of linear higher order reversible cellular automata as

described in [6, p 245]. Consider the second-order linear cellular automata with state transition function

$$P(t) = F.P(t-1) - P(t-2) \in \mathbb{F}_q[X_i, X_i^{-1}; i]$$

where $F$ is an arbitrary element of $\mathbb{F}_q[X_i, X_i^{-1}; i]$. By proposition 1 this second-order cellular automaton can be simulated by the linear cellular automaton with two qits of cellprocessor memory and with corresponding matrix

$$A_\Gamma = \begin{pmatrix} F & -1 \\ 1 & 0 \end{pmatrix} \in M_2(\mathbb{F}_q[X_i, X_i^{-1}; i])$$

which is invertible and so $\Gamma$ and hence the second-order linear cellular automaton is reversible.

On the other hand it is possible to describe reversible linear cellular automata which do not simulate higher-order linear cellular automata with one qit of cellprocessor memory.

# 4 The state transition graph

We will now describe the state transition graph of an arbitrary linear cellular automaton. A connected component of this graph will be called a lifecycle. A lifecycle is called a halfline if it is of the form

$$\circ \to \circ \to \circ \to \cdots$$

where the first configuration does not have a predecessor. Such configurations are usually called 'garden of Eden' configurations. A lifecycle is called a line if it is of the form

$$\cdots \to \circ \to \circ \to \circ \to \cdots$$

A lifecycle will be named senile if it is finite and repetitive, i.e. it goes in circles...

First we will consider the relatively easy case of linear cellular automata having just one qit of cellprocessor memory :

**Proposition 4.1** *Let $\Sigma$ be a non-reversible linear cellular automaton with one qit of cellprocessor memory, then the state transition graph consists of infinitely many halflines indexed by the garden of Eden configurations which are precisely those elements $F \in \mathbb{F}_q[X_i, X_i^{-1}; i]$ such that the defining polynomial $A_\Sigma$ does not divide $F$.*

**Proof** Since $\mathbb{F}_q[X_i, X_i^{-1}; i]$ is a unique factorization domain and $A_\Sigma$ is not a unit, every element $F \in \mathbb{F}_q[X_i, X_i^{-1}; i]$ can be written uniquely in the form $F = A_\Sigma^a.G$ where $a \geq 0$ and $A_\Sigma$ does not divide $G$. Therefore, $F$ corresponds to the configuration on place $a$ of the halfline lifecycle starting in the garden of Eden configuration corresponding to the polynomial $G$. ∎

In order to handle the reversible case, we define the $X_i$-degree of an element $F \in \mathbb{F}_q[X_i, X_i^{-1}; i]$ to be the highest power of $X_i$ occuring in a monomial of $F$. If $\Sigma$ is a reversible linear cellular automaton with one qit of cellprocessor memory, we know that its defining polynomial is a single monomial $A_\Sigma = f.X_1^{a_1} \cdots X_k^{a_k}$ where $f \in \mathbb{F}_q$. Consider first the case that all $a_i = 0$, then the state transition graph consists of infinitely many senile lifecycles of a fixed period which must be a divisor of $q-1$ (the group of units of a finite field is a cyclic group of order $q-1$) the period is the minimal value $a$ such that $f^a = 1$. If at least one of the $a_i \neq 0$, two configurations $P$ and $Q$ belong to the same lifecycle provided $P = A_\Sigma^a.Q$ for some $a \in \mathbb{Z}$. But then $X_i$-deg$(P) = X_i$-deg$(Q)$+$a.a_i$. This finishes the proof of :

**Proposition 4.2** *Let $\Sigma$ be a reversible linear cellular automaton with one qit of cellprocessor memory, i.e. $A_\Sigma = f.X_1^{a_1} \cdots X_k^{a_k}$ where $f \in \mathbb{F}_q$ then*

1. *If all $a_i = 0$ the state transition graph consists of infinitely many senile lifecycles with fixed period equal to the least $a$ such that $f^a = 1$ which is always a divisor of $q-1$*
2. *If there is a $k$ such that $a_k \neq 0$ then the state transition graph consists of infinitely many line lifecycles indexed by those elements $P \in \mathbb{F}_q[X_i, X_i^{-1}; i]$ s.t. $0 \leq X_k\text{-}deg(P) < a_k$*

For more qits of cellprocessor memory the situation is more complicated (and interesting). We will first handle the case that the determinant of the associated matrix $A_\Sigma$ is not equal to zero.

In this case the predecessor of a position $P \in V[X_i, X_i^{-1}; i]$ must be unique (if it exists). Hence there are three possible types of lifecycles : halflines, lines and seniles. Below will be concerned wih parametrizing these type of lifecycles. In order to get some grip on the various order of infinity that will occur, we will make the following definition.

**Definition 4.3** *Assume that $N$ is a $\mathbb{F}_q[X_i, X_i^{-1}; i]$-submodule of $V[X_i, X_i^{-1}; i]$ of rank $r$ then we say that there are $\infty^r$ configurations in $N$.*

It is clear that a configuration, whose lifecycle is a line or senile will lie in

$$N_\Sigma = \bigcap_n A_\Sigma^n.V[X_i, X_i^{-1}; i]$$

Furthermore, configurations whose lifecycle is senile will belong to

$$N'_\Sigma = \{m \in V[X_i, X_i^{-1}; i] \mid \exists n > 0 : A_\Sigma^n.m = 0\}$$

It is clear that both $N_\Sigma$ and $N'_\Sigma$ are submodules of $V[X_i, X_i^{-1}; i]$. Below we will determine the ranks of $N_\Sigma$ and $N'_\Sigma$.

To this end we will need some elementary commutative algebra. In particular we have to recall the definition of the characteristic polynomial of an endomorphism of a module.

Let $R$ be a commutative Noetherian integrally closed domain with quotient field $K$ and let $M$ be a finitely generated $R$-module equipped with an endomorphism $f$. Then the characteristic polynomial $F_f(\lambda)$ of $f$ is defined as the characteristic polynomial of $1 \otimes f$ acting on $K \otimes M$.

**Lemma 4.4** *Suppose that we have a commutative diagram of $R$-modules with exact rows*

$$\begin{array}{ccccccccc} 0 & \to & M & \to & M' & \xrightarrow{\rho} & M'' & \to & 0 \\ & & \downarrow f & & \downarrow f' & & \downarrow f'' & & \\ 0 & \to & M & \to & M' & \xrightarrow{\rho} & M'' & \to & 0 \end{array}$$

*Then*

$$F_{f'}(\lambda) = F_f(\lambda) F_{f''}(\lambda)$$

**Proof** Tensor this commutative diagram with $K$. Then $1 \otimes \rho$ will split and hence we may write $1 \otimes f'$ in matrix form as

$$\begin{pmatrix} A'' & * \\ 0 & A \end{pmatrix}$$

where $A$, $A''$ are the matrices corresponding to $1 \otimes f$ and $1 \otimes f''$. This proves what we want. ∎

**Lemma 4.5** *$F_f(\lambda)$ has coefficients in $R$.*

**Proof** Write $M$ as a quotient of a free module $R^n$. Then we may lift $f$ to an endomorphism $g$ of $R^n$ and hence we may construct a commutative diagram with exact rows

$$\begin{array}{ccccccccc} 0 & \to & N & \to & R^n & \to & M & \to & 0 \\ & & \downarrow g' & & \downarrow g & & \downarrow f & & \\ 0 & \to & N & \to & R^n & \to & M & \to & 0 \end{array}$$

Clearly $F_g(\lambda) = F_f(\lambda) F_{g'}(\lambda)$ has coefficients in $R$, but then by Gauss's lemma [3], $F_f(\lambda)$ will also have coefficients in $R$. ∎

**Lemma 4.6** *Assume that*

$$F_f(\lambda) = U(\lambda) V(\lambda)$$

*where $U(\lambda)$ and $V(\lambda)$ are monic polynomials. Define*

$$N = \ker U(f) = \{m \in M \mid U(f)m = 0\}$$

*and $f' = f \mid N$. Then $F_{f'}(\lambda) = U(\lambda)$ and hence* $\operatorname{rk} N = \deg U(\lambda)$.

**Proof** Using the definition of the characteristic polynomial, it suffices to prove this for a field and there the assertion is standard. ∎

According to Gaus's lemma, the monic irreducible factors of $F_f(\lambda)$ will have coeffients in $R$. Let $G_f(\lambda)$ be the product of all irreducible factors of $F_f(\lambda)$ whose constant term is a unit in $R$ (we put $G_f(\lambda) = 1$ if there are no such factors).

**Lemma 4.7** *Let* $N = \bigcap_n f^n(M)$. *Then*

$$\operatorname{rk} N = \deg G_f(\lambda)$$

**Proof** According to lemma 4.6 it is sufficient to show that $N = \ker G_f(f)$.

Let $m \in M$ such that $G_f(f)m = 0$. Write out $G_f(\lambda)$ as

$$\lambda^r + q_{r-1}\lambda^{r-1} + \cdots + q_0$$

and define for $n > 0$

$$m'_n = [-q_0^{-1}(f^{r-1} + q_{r-1}f^{r-2} + \cdots + q_1)]^n m$$

Then $f^n(m'_n) = m$ and hence $m \in f^n(M)$ for all $n$. Therefore $\ker G_f(f) \subset N$.

Now let $f' = f \mid N$. $f'$ is an isomorphism and hence $F_{f'}(\lambda) \mid G_f(\lambda)$. We obtain

$$\ker G_f(f) \subset N \subset \ker F_{f'}(f) \subset \ker G_f(f)$$

which proves what we want. ∎

**Lemma 4.8** *Define*

$$N' = \{m \in M \mid \exists n > 0 : f^n(m) = m\}$$

*Then* $\operatorname{rk} N'$ *is equal to the number of roots of* $F_f(\lambda)$ *that are roots of unity.*

**Proof** We may assume that $K = R$. Recall that a cyclotomic polynomial is a polynomial whose only roots are roots of unity.

Let $G'_f(\lambda)$ be the product of monic irreducible factors of $F_f(\lambda)$ that are cyclotomic polynomials with the restriction that we take every such factor only once. Then according to lemma 4.6 it is sufficient to show that $N' = \ker G'_f(f)$.

Let $m \in N'$. Then $(f^n - 1)m = 0$ for some $n > 0$. Hence by Bezout $(f^n - 1, F_f(f))m = 0$. But $(\lambda^n - 1, F_f(\lambda)) \mid G'_f(\lambda)$ and hence $m \in \ker G'_f(f)$.

Conversely let $m \in M : G'_f(f)m = 0$. Since $G'_f(f) \mid f^n - 1$ for some $n > 0$ we find that $m \in N'$. ∎

We may summarize the above results in the case of a linear cellular automaton.

**Theorem 4.9** *Let $F_\Sigma$ be the characteristic polynomial of $A_\Sigma$. Let $r$ be the degree of the monic factor of $F_\Sigma$ of highest degree, whose constant term is a monomial. Let $r'$ be the number of eigenvalues of $A_\Sigma$ that are roots of unity. Then*

1. *There are $\infty^r$ configurations that are senile or have a line as a lifecyle.*
2. *There are $\infty^{r'}$ senile configurations.*

**Example 4.10** *Let $\Sigma$ be a onedimensional linear cellular automaton with three bits of cellprocessor memory, $\Delta = \{-1, 1\}$ and the state transition function is determined by the matrices*

$$A_{-1} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix} \qquad A_1 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}$$

*Then, every configuration $P = (F_1, F_2, F_3)^\tau$ with $F_i \in \mathbb{F}_2[X, X^{-1}]$ s.t. $F_1 \neq 0$ lies on a halfline lifecycle starting from a garden of Eden configuration which are determined by the fact that their first component is a nonzero element $G_1 \in \mathbb{F}_2[X, X^{-1}]$ not divisible by $X^{-1} + X$. Every configuration $P = (0, F_2, F_3)$ is senile with period 2 except for the zero configuration.*

So, we are left to consider the remaining case : $\Sigma$ is a linear cellular automaton such that the associated matrix $A_\Sigma$ has zero determinant. In this case 0 will be an eigenvalue of $A_\Sigma$. So, let us first consider the case that 0 is the only eigenvalue of $A_\Sigma$. Over the algebraic closure $K$ of $\mathbb{F}_q(X_1, \ldots, X_k)$ the matrix $A_\Sigma$ is similar to a direct sum of Jordan matrices $J_i$ of size $u_i$ s.t. $\sum_{i=1}^{z} u_i = n$. Recall that $J_i$ is a square matrix of size $u_i$ of the form

$$\begin{pmatrix} 0 & 1 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 1 & \cdots & 0 & 0 \\ & & & & & \\ 0 & 0 & 0 & \cdots & 0 & 1 \\ 0 & 0 & 0 & \cdots & 0 & 0 \end{pmatrix}$$

The state transition graph of such a linear cellular automaton is a rooted tree (having the quiescent state as graveyard state) such that in each node there are $\infty^z$ direct predecessors and all lifecycles have length smaller or equal to $\sum_{i=1}^{z}(u_i - 1)$ which is always smaller than $n$. In particular after at most $n$ clock pulses every cell will be in the quiescent state. This is another property of linear cellular automata : mercifull dead (if something dies, it dies quickly).

In the general case we may consider the submodule of $V[X_i, X_i^{-1}; i]$ given by

$$N''_\Sigma = \{P \in V[X_i, X_i^{-1}; i] \mid \exists n \geq 0 : A_\Sigma^n . P = 0\}$$

Again the configurations in $N''_\Sigma$ form a connected component of the state transition graph of $\Sigma$. The rank $r''$ of $N''_\Sigma$ is the number of zero eigenvalues of $A_\Sigma$. Furthermore, as above, we show that the number $n$, occuring in the definition of $N''_\Sigma$ may be taken to be less than or equal to $r''$.

We may now construct a commutative diagram with exact rows

$$\begin{array}{ccccccccc} 0 & \to & N''_\Sigma & \to & V[X_i, X_i^{-1}; i] & \xrightarrow{\rho} & M_1 & \to & 0 \\ & & \downarrow A_\Sigma & & \downarrow A_\Sigma & & \downarrow A_\Sigma & & \\ 0 & \to & N''_\Sigma & \to & V[X_i, X_i^{-1}; i] & \xrightarrow{\rho} & M_1 & \to & 0 \end{array} \tag{1}$$

Below we will show that $\rho$ splits in a way that is compatible with the action of $A_\Sigma$ (but *not* with the $\mathbb{F}_q[X_i, X_i^{-1}; i]$-module structure).

Denote this splitting by $\psi$. Then we have

$$V[X_i, X_i^{-1}; i] = N''_\Sigma \oplus \psi(M_1)$$

We obtain that the state transition graph of $\Sigma$ is the product of the state transition graphs of configurations in $N''_\Sigma$ and of configurations in $\psi(M_1)$.

The state transition graph of configurations in $\psi(M_1)$ is a union of lines halflines and seniles. From this we obtain the following

**Theorem 4.11** *In general, the connected components of the state transition graph of a linear cellular automaton $\Sigma$ are either lines, halflines or seniles, with in every node a fixed rooted tree attached. If $r''$ is the number of zero eigenvalues of $A_\Sigma$ then there are $\infty^{r''}$ nodes in this rooted tree. The length of the braches is at most $r'' - 1$. The number of components that correspond to lines or seniles may be determined exactly as in Theorem 4.9*

**Theorem 4.12** *A linear cellular automaton can never be a universal Turing machine*

**Proof** If it were, the halting problem would be decidable since a configuration dies if and only if it is dead after $n$ clock pulses. ∎

By combining this fact with the observation that higher order linear cellular automata can always be simulated by linear ones we also have proved that higher order linear cellular automata cannot be universal Turing machines.

Now we are left with proving the splitting of (1). This follows from the lemma below if we let $k = \mathbb{F}_q$, and the action of $Y$ corresponds to the action of $A_\Sigma$.

**Lemma 4.13** *Let $k$ be a field and let*

$$0 \to N \to M \xrightarrow{\rho} M_1 \to 0 \tag{2}$$

*be a short exact sequence of $k[Y]$-modules. Assume that $Y^n N = 0$ and $M_1$ is $Y$-torsion free. Then this sequence is split as $k[Y]$-modules.*

**Proof** The problem is that we do not require $M_1$ to be finitely generated as a $k[Y]$-module. Otherwise $M_1$ would be projective and (2) would be trivially split. In our more general situation we only know that $M_1$ is flat. Nevertheless we can circumvent this difficulty in the following way : $M_1/Y^nM_1$ is a flat $k[Y]/(Y^n)$-module and for $k[Y]/(Y^n)$ it is true that every flat module is projective by a result of Bass [1]. Furthermore, using the fact that $M$ is $Y$-torsionfree we deduce that

$$\mathrm{Ext}^1_{k[Y]}(M,N) = \mathrm{Ext}^1_{k[Y]/(Y^n)}(M_1/Y^nM_1, N)$$

These two fact together show that $\mathrm{Ext}^1_{k[Y]}(M,N) = 0$ and hence (2) splits. ∎

# References


[1] H. Bass, Finitistic dimension and a homological generalization of semi-primary rings, Trans. Amer. Math. Soc. 95, 466-488 (1960).

[2] E. Berlekamp, Algebraic Coding Theory, (1968) McGraw-Hill.

[3] S. Lang, Algebra, (1965), Addison-Wesley, Massachusetts.

[4] J. von Neumann, Theory of self-reproducing automata, edited by A.W. Burks.

[5] O. Martin, A. Odlyzko, S. Wolfram, Algebraic properies of cellular automata, Bell Laboratories report (Jan. 1983).

[6] T. Toffoli, Computation and construction universality of reversible cellular automata, J. Comput. Sys. Sci. 15, 213.

[7] S. Ulam, Some ideas and prospects in biomathematics, Ann. Rev. Bio., 255.

[8] S. Wolfram, Statistical mechanics of cellular automata, Rev. of Mod. Physics, Vol. 55, No. 3.