Symbolic Dynamics and Finite Automata

Marie-Pierre Béal
Institut Gaspard Monge,
Université Paris 7 - Denis Diderot,
France

Dominique Perrin
Institut Gaspard Monge,
Université de Marne-la-Vallée,
France

December 17, 1999

1 Introduction

Symbolic dynamics is a field which was born with the work in topology of Marston Morse at the beginning of the twenties [44]. It is, according to Morse, an *"algebra and geometry of recurrence"*. The idea is the following. Divide a surface into regions named by certain symbols. We then study the sequences of symbols obtained by scanning the successive regions while following a trajectory starting from a given point. A further paper by Morse and Hedlund [45] gave the basic results of this theory. Later, the theory was developed by many authors as a branch of ergodic theory (see for example the collected works in [59] or [12]). One of the main directions of research has been the problem of the *isomorphism of shifts of finite type* (see below the definition of these terms). This problem is not yet completely solved although the latest results of Kim and Roush [35] indicate a counterexample to a long-standing conjecture formulated by F. Williams [61].

There are many links between symbolic dynamics and the theory of automata, as pointed out by R. Adler and B. Weiss [60]. Actually a very early reference on this connection can be found in a paper of A. Gleason published many years later [30] after a series of lectures given at the Institute for Defense Analysis in 1960. In this paper, based on notes by R. Beals and M. Spivak, methods of finite semigroups were introduced to obtain some of the results of G. Hedlund.

The idea of considering infinite words also appears, of course, in the framework of automaton theory, independently of symbolic dynamics. This theory was developed initially by R. Büchi and R. McNaughton. Since the beginning it has, however, taken a different direction and is connected with problems of logic rather than with the topological ones raised in symbolic dynamics.

In this chapter, we present some interconnections between automata and symbolic systems and discuss some of the new results that have been obtained in this direction together with some interesting open problems.

The material presented here does not cover all existing connections of this kind. There are, in particular, interesting links between symbolic dynamics and representation of numbers that are not presented here (see [28]). There are also important connections with cellular automata (see e.g. [16]). The applications of symbolic dynamics to coding are not treated (see [2] or the book of D. Lind and B. Marcus [38]).

The chapter is organized as follows. The first sections (Sections 2, 3, 4) constitute an introduction to symbolic dynamics. It is essentially self-contained although some proofs are only sketched. The concepts introduced include shift spaces, symbolic systems, minimal systems, sofic systems and systems of finite type.

In the next section (Section 5), we show how the notion of a minimal deterministic automaton translates in the framework of symbolic dynamics to the notion of a Fischer cover. Both notions essentially coincide but for the choice of an initial state.

The following section (Section 6) describes the relation between unambiguous automata and codes with a class of maps called finite-to-one. We show that some results on the completion of codes can be translated into ones on shifts of finite type.

Section 7 is an introduction to the technique of state splitting. We show in particular how this operation is related to automata minimization.

The next section (Section 8) deals with the notion of the isomorphism of shifts of finite type. We define shift equivalence and strong shift equivalence. We show that two shifts of finite type are isomorphic iff their matrices are strong shift equivalent (William's theorem).

Section 9 contains the definition of entropy. We show how this notion is related to well-known concepts in automata and coding theory such as the Kraft inequality. We also state without proof a recent result of D. Handelman that characterizes the entropies of systems generated by finite codes.

The following sections present various topics relating symbolic dynamics and finite automata, covering in particular zeta functions and circular codes.

This chapter is a survey of many results and concepts, not all of them presented with the same degree of detail. As far as the definitions are concerned, it is self-contained for a reader familiar with basic notions of automata theory. Concerning the proofs of the results, the situation is more variable: some of them are given completely, even if sometimes condensed. Some others are only sketched or not even given here, as not being in the scope of this survey.

The material presented is an extended version of a survey by the second author at the conference MFCS in September 1995 in Prague [51].

We would like to thank many people for their help during the preparation of this work and, in particular, Frédérique Bassino, Véronique Bruyère, Aldo De Luca, and Paul Schupp.

2 Symbolic dynamical systems

We present in this Section a short introduction to the concepts of symbolic dynamics. For a much more detailed and complete exposition, we refer to the new book of Doug Lind and Brian Marcus [38] which is the first exposition in book form of this theory. Our presentation aims especially at a public of computer scientists already familiar with such concepts as finite automata and transductions.

Let A be a finite alphabet. We denote by A^* the set of finite words on A. The empty word is denoted ϵ and the set of nonempty words is thus $A^+ = A^* - \epsilon$. We consider the set $A^{\mathbb{Z}}$ of two-sided infinite words as a topological space with respect to the usual product topology. An element of $A^{\mathbb{Z}}$ is a sequence

$$\begin{aligned} x &= (x_n)_{n\in\mathbb{Z}} \\ &= \ldots x_{-1}x_0x_1\ldots \end{aligned}$$

The topology is defined by the distance for which words are close if they coincide on a long interval centered at 0. Formally, we may define the distance of x, y as

$$d(x,y) = 2^{-e(x,y)}$$

where

$$e(x,y) = \max\{n \geq 0 \mid x_i = y_i,\ -n \leq i \leq n\}$$

using the convention $e(x,y) = \infty$ if $x = y$, and $e(x,y) = -1$ if $x_0 \neq y_0$.

The *shift* transformation σ acts on $A^{\mathbb{Z}}$ bijectively. It associates to $x \in A^{\mathbb{Z}}$ the element $y = \sigma(x) \in A^{\mathbb{Z}}$ defined for $n \in \mathbb{Z}$ by

$$y_n = x_{n+1}$$

and obtained by shifting all symbols one place left.

A *symbolic dynamical system* or *subshift* is a subset S of $A^{\mathbb{Z}}$which is both

1. topologically closed and,

2. shift-invariant, i.e. such that $\sigma(S) = S$.

Thus, and in more intuitive terms, a subshift is a set of bi-infinite words whose definition does not make reference to the origin and allows one passing to the limit.

The system is denoted S or (S, σ) to emphasize the role of the shift σ.

For example, $(A^{\mathbb{Z}}, \sigma)$ itself is a symbolic dynamical system, often called the *full shift* in contrast with the subshifts whose name refer to the embedding in the full shift $A^{\mathbb{Z}}$.

As a less trivial example, the set of sequences on $A = \{a, b\}$ such that a symbol b is always followed by a symbol a is a subshift often called the *golden mean* system. We shall use this example several times in the rest of the chapter.

Let G be a directed graph with E as its set of edges. We actually use multigraphs instead of ordinary graphs in order to be able to have several distinct edges with the same origin and end. Formally, a directed multigraph is given by two sets E (the edges) and V (the vertices) and two functions $\alpha, \beta : E \to V$. The vertex $\alpha(e)$ is the *origin* of the edge e and $\beta(e)$ is its *end*. We shall always say "graph" for "directed multigraph".

Let S_G be the subset of $E^{\mathbb{Z}}$ formed by all bi-infinite paths in G. It is clear that S_G is a subshift called the *edge shift* on G. Indeed S_G is closed and shift invariant by definition.

Figure 1 presents an example of a graph with three edges which defines an edge shift on three symbols.

Let $\mathcal{A} = (Q, E)$ be a *finite automaton* on an alphabet A given by a finite set Q of states and a set $E \subset Q \times A \times Q$ of edges but without initial or final states. The set of all labels $\cdots a_{-1} a_0 a_1 \cdots$ of the bi-infinite paths

$$\cdots p_{-1} \xrightarrow{a_{-1}} p_0 \xrightarrow{a_0} p_1 \cdots$$

is a subshift S. We say that S is the subshift *recognized* by the automaton $\mathcal{A}$.

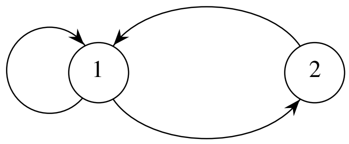

Figure 1: An edge shift

An automaton can be considered as a graph in which the set of edges is contained in $Q \times A \times Q$. Thus we may consider the subshift formed by all bi-infinite paths in $\mathcal{A}$, which is really the edge shift $S_{\mathcal{A}}$. The subshift recognized by $\mathcal{A}$ is the image of the edge shift $S_{\mathcal{A}}$ under the map assigning to a path its label. This map is continuous. Since the automaton is finite, the set S_G is compact and so is S which is thus closed. A subshift obtained in this way is called a *sofic* shift or a sofic system.

We will use in the sequel finite automata, either in the classical sense as a tuple (Q, E, I, T) with $I \subset Q$ as set of initial states and $T \subset Q$ as set of final states, to recognize a set of finite words, or, as above, without initial and final states, to recognize a subshift.

For a subset X of A^*, we say that X is *recognizable* if it can be recognized by a finite automaton.

A finite word v is said to be a *factor* (or also a *block*) of a finite or infinite word z if its symbols appear consecutively in z.

We may associate with a subshift $S \subset A^{\mathbb{Z}}$ the set

$$F_S = \{x_i \cdots x_j \mid x \in S,\ i \leq j\} \cup \epsilon$$

of factors of its elements.

We may also consider the complement $I_S = A^* - F_S$ of this set, which is the set of *forbidden blocks*.

The set $F = F_S \subset A^*$ satisfies the conditions of being

1. factorial: $uvw \in F \Rightarrow v \in F$.
2. extendible: $\forall v \in F,\ \exists\ a, b \in A :\ avb \in F$.

Conversely, such a set of finite words is the set of factors of a subshift as shown by the following proposition.

Proposition 1 *Let $F \subset A^*$ be a factorial and extendible set. The set*

$$S_F = \{x \in A^Z \mid x_i \cdots x_j \in F\ (i \leq j)\}$$

is a subshift and $F_{S_F} = F$. Conversely, if S is a subshift, then $S_{F_S} = S$.

Proof. Let $S = S_F$. It is clear that S is a subshift and that $F_S \subset F$. To show that $F \subset F_S$, consider $v \in F$. Since F is extendible, we can construct a two-sided infinite word x of the form $\cdots a_n \cdots a_1 v b_1 \cdots b_n \cdots$ such that $a_n \cdots a_1 v b_1 \cdots b_n \in F$ for all n. Then all factors of x are in F and thus $x \in S$, whence $v \in F_S$. Thus $F_S = F$.

For the second formula, let $F = F_S$. The inclusion $S \subset S_F$ is clear. The converse follows by compactness.

■

One may also define a subshift by a set of forbidden blocks. Indeed, for any set $I \subset A^+$ the set

$$S = \{x \in A^Z \mid x_i \cdots x_j \notin I \ (i \leq j)\}$$

is always a subshift whatever be the set I, but we only have the inclusion $I_S \supset I$ instead of an equality.

For instance, if S is the golden mean system, the set I_S of forbidden blocks is formed by all words containing bb.

The simple relation between a subshift and its set of factors shows that subshifts are closely related to ordinary sets of finite words, in contrast with more general sets of infinite words defined by Büchi automata. The latter are indeed not topologically closed in general and thus have a larger topological complexity in the Borel hierarchy (on Büchi automata, see [24], [58], or the chapter by W. Thomas in this Handbook).

We say that an automaton $\mathcal{A} = (Q, E)$ is *trim* if any state is on a bi-infinite path. We shall consider only here trim automata since we are interested in bi-infinite paths. For an automaton $\mathcal{A} = (Q, E, I, T)$ with initial and final states, we say that $\mathcal{A}$ is *trim* if any state is accessible from I and can access T.

Proposition 2 *Let $\mathcal{A} = (Q, E)$ be a finite trim automaton. The set F_S of factors of the subshift S recognized by $\mathcal{A}$ is recognized by the automaton (Q, E, Q, Q) in which all states are both initial and final. Conversely, if F_S is recognized by a trim finite automaton $\mathcal{A} = (Q, E, i, T)$, then S is recognized by the automaton (Q, E).*

In particular, S is sofic iff F_S is recognizable.

Proof. The first assertion is clear. For the second one, let S' be the subshift recognized by $\mathcal{A}$. Since $\mathcal{A}$ is trim, we have $F'_S \subset F_S$. The converse inclusion is also true by compactness of the set of paths in $\mathcal{A}$. Since $F_S = F_{S'}$ we have $S = S'$ and thus the conclusion. The third statement is a direct consequence of the first ones.

■

The notions introduced above can also be formulated in the context of one-sided infinite words. Indeed the set $A^{\mathbb{N}}$ of one-sided infinite words is also a topological space and the shift transformation is also defined upon it although it is no longer one-to-one. A *one-sided symbolic system*, or one-sided subshift, is a set $S \subset A^{\mathbb{N}}$ which is both closed and invariant. Equivalently, it is the set of right infinite sequences that appear in a shift. We shall usually work with two-sided subshifts because two-sided shifts take into account both the past and the future. An exception will be made in Section 3 concerning the notion of recurrence.

A subshift S is said to be *irreducible* if for any $x, y \in F_S$ there is a word u such that $xuy \in F_S$.

For example, the golden mean system is irreducible. In fact, if x and y avoid bb, then xay also does. On the contrary, the set of infinite words on $\{a, b\}$ avoiding ba is reducible since for all u, the word bua contains a factor ba.

Let S be the edge graph of a graph G. If G is strongly connected, then S is irreducible. The converse is true provided the graph satisfies the condition that each vertex has positive in- and out-degree.

An automaton with a strongly connected graph is said to be *transitive*.

In general, the definition of an irreducible system can be formulated in topological terms and it is related to the possibility of a decomposition into simpler elements.

A subshift S is said to be *primitive* if there is an integer $n > 0$ such that for all $x, y \in F_S$ there exists a word u of length n such that $xuy \in F_S$.

For example, the golden mean system is primitive. On the contrary, the system S formed by all words on $\{a, b\}$ avoiding aa and bb is not primitive. Indeed, for $x = a$, $y = b$ the only adequate u has to be in $(ba)^*$ and thus of even length, whereas for $x = a$, $y = a$ it has to be in $b(ab)^*$ and thus of odd length.

When S is the edge graph of a strongly connected graph G, then S is primitive if and only if the gcd of the cycle lengths is one.

A *morphism* between two subshifts (S, σ) and (T, τ) is a map $f : S \to T$ which is continuous and commutes with the shifts, i.e. such that $f\sigma = \tau f$.

$$\begin{tikzcd} S \arrow[r, "f"] \arrow[d, "\sigma"'] & T \arrow[d, "\tau"] \\ S \arrow[r, "f"] & T \end{tikzcd}$$

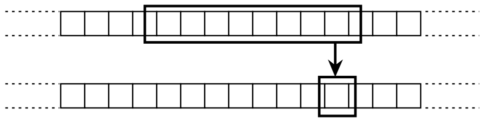

Figure 2: The sliding window

When f is a morphism from S onto T, it is said that T is a *factor* of S.

Let $S \subset A^{\mathbb{Z}}$ and $T \subset B^{\mathbb{Z}}$ be two subshifts and let $k \geq 1$ be an integer. A function $f : S \to T$ is said to be *k-local* or local if there exists a function $\overline{f} : A^k \to B$ and an integer $m \in \mathbb{Z}$ such that for all $x \in S$ the word $y = f(x)$ is defined for $n \in \mathbb{Z}$ by

$$y_{n+m} = \overline{f}(x_{n-(k-1)} \cdots x_{n-1} x_n) \tag{1}$$

Thus the value of a symbol in the image is a function of the symbols contained in a window of length k above it, called a *sliding window* (represented on Figure 2 in the case $m = 0$). A local function defined by a formula like Formula (1) with $m = 0$ is called *sequential.* Thus a sequential local function is one that writes below the right end of the window.

A local function is also called a *sliding block code* and a k-local function is also called a *k-block map.* A 1-local function or one-block map is nothing else than an alphabetic substitution (or very fine morphism in [24]).

The following result is well-known. In Hedlund's article [33], it is credited to M.L. Curtis, G. Hedlund and R.C. Lyndon [1].

Theorem 1 *Let $S \subset A^{\mathbb{Z}}, T \subset B^{\mathbb{Z}}$ be two symbolic systems defined over finite alphabets A and B. A function $f : S \subset A^{\mathbb{Z}} \to T \subset B^{\mathbb{Z}}$ is a morphism if it is k-local for some $k \geq 0$.*

Proof. It is clear that a local function is a morphism since it is continuous and commutes with the shift by definition. Conversely, let $f : S \to T$ be a continuous map. Since A is finite, $A^{\mathbb{Z}}$ is compact and so is S which is closed in $A^{\mathbb{Z}}$. Thus f is uniformly continuous. This implies that there is an integer k such that the symbol $f(x)_0$ is determined by the window $x_{-k} \cdots x_0 \cdots x_k$. Since f commutes with the shift the other symbols of $f(x)$ are also determined by the corresponding window of length $2k + 1$.

[1] Such an activity was supposedly related to cryptography and supported as such by the Institute for Defense Analysis. It was one of Roger Lyndon's favorite subjects of jokes on the notion of applications in mathematics.

■

An *isomorphism* (also called a *conjugacy*) is a bijective morphism. If f is an isomorphism from S onto T, then S and T are said to be *conjugate*. Since the alphabet A is finite, the space S is compact. The inverse of a continuous function $f : S \to T$ is also continuous when S is compact.Thus the inverse of a conjugacy is a conjugacy.

As a general rule, the concepts studied in symbolic dynamics are invariant under conjugacy and a lot of the attention is given to the search of *complete invariants*, i.e. invariants that characterize a subshift up to conjugacy.

3 Recurrence and minimality

In this section, we concentrate on a special kind of symbolic dynamical systems: the smallest system containing a given infinite word. It is more appropriate to present it in the one-sided case. For a one-sided infinite word $x \in A^{\mathbb{N}}$, we define $F(x)$ to be the set of factors of x. We also define $S(x) = \{y \in A^{\mathbb{N}} \mid F(y) \subset F(x)\}$. The set $S(x)$ is obviously the smallest subshift containing x.

A one-sided infinite word $x \in A^{\mathbb{N}}$ is said to be *recurrent* if any block occurring in x has an infinite number of occurrences. It is obviously enough for x to be recurrent that any prefix of x has a second occurrence.

It is easy to verify that x is recurrent if and only if the subshift $S(x)$ is irreducible. Indeed, if $S(x)$ is irreducible, then for any prefix u of x there is a v such that $uvu \in F(x)$ and thus u has a second occurrence. Conversely, if x is recurrent then for any $u, v \in F(x)$, v has an occurrence following any occurrence of u and thus there is a word w such that $uwv \in F(x)$.

The notion of a recurrent word is linked to the concept of a *sesquipower* (or $(1+\frac{1}{2})$-power). A word w is called a sesquipower of order n if it can be written $w = uvu$ with u a sesquipower of order $n-1$ and v any word. A sesquipower of order 0 is any nonempty word.

A recurrent word is one that can be written as an infinite *sesquipower*, i.e. as

$$\begin{aligned} x &= u_0 \cdots \\ &= u_0 u_1 u_0 \cdots \\ &= u_0 u_1 u_0 u_2 u_0 u_1 u_0 \cdots \end{aligned} \tag{2}$$

Indeed, we can choose u_0 to be the first symbol of x. Then, since u_0 has a second occurrence in x, we can find u_1 such that $u_0u_1u_0$ is a prefix of x and so on.

A word $x \in A^{\mathbb{N}}$ is said to be *uniformly recurrent* if every block of x appears infinitely often at bounded distance.

A periodic word is obviously uniformly recurrent. A simple way to construct a uniformly recurrent non periodic word is to use words u_i of bounded length in Eq. (2). We shall see examples in more detail below.

These notions are strongly related to that of a *minimal subshift*, i.e. a subshift $S \subset A^Z$ such that $T \subset S$ implies $T = \emptyset$ or $T = S$.

The following result is one of the earliest in symbolic dynamics ([15],[45]). It links a dynamical property of the orbit of a point with a property of the words representing the orbit.

Theorem 2 *Let $x \in A^{\mathbb{N}}$ be a one-sided infinite word. The following conditions are equivalent.*

1. *x is uniformly recurrent.*
2. *$S(x)$ is minimal.*

Proof. $1 \Rightarrow 2$ Let $S \subset S(x)$ be a subshift and let $y \in S$. Then $S(y) \subset S$. Since $y \in S(x)$, we have $F(y) \subset F(x)$ by the definition of $S(x)$. Let $w \in F(x)$. Since x is uniformly recurrent, w appears in every long enough block of x. Hence w appears in the long enough blocks of y. Whence $w \in F(y)$. This shows that $F(x) = F(y)$ and this implies that $S(y) = S = S(x)$.

$2 \Rightarrow 1$ Any block w of x appears in all $y \in S(x)$ since $S(x)$ is minimal. For a given block w of x, we define $i_w(y)$ to be the function assigning to $y \in S(x)$ the least integer i such that $y = uwz$ with $|u| = i$. Since i_w is continuous and $S(x)$ is compact, i_w is bounded. Let w be a block of $x = uwy$. Since $y \in S(x)$, $w \in F(y)$ and thus w has a second occurrence in x at a distance bounded by $i_w(y)$.

■

Example 1 The word of Thue-Morse is the infinite word obtained by iterating the substitution f defined by $f(a) = ab$, $f(b) = ba$. The word $m = f^{\omega}(a)$ is uniformly recurrent. Indeed, aaa or bbb are not in $F(m)$. Thus successive occurrences of a or b are separated by at most two symbols. It follows that any block of m appears at bounded distance since it has to appear in some $f^k(a)$ or $f^k(b)$. The system $S(m)$ is known as the *Morse minimal set.*

It is possible to generalize the example of the Morse minimal set as follows. Let $f : A \to A^*$ be a substitution such that for any symbols $a, b \in A$ there is at least an occurrence of b in $f(a)$. Then any infinite word x such that $f(x) = x$ is uniformly recurrent [43].

We used in the proof of Theorem 2 a possible variant of the definition of a uniformly recurrent word: for all $n > 0$ there is an $m > n$ such that any factor of length n appears in any factor of length m. This condition can be used as a definition for a uniformly recurrent two-sided infinite word. It also leads to the definition of a function $r_x(n)$ called the *recurrence index* of x. We let $r_x(n) = m$ if m is the smallest possible integer such that any factor of length n appears in any factor of length m. It is well defined for all integers n iff x is uniformly recurrent.

It is worth mentioning yet another equivalent definition of uniformly recurrent words. It uses the notion of a well-quasi-order. Recall (see [39],[37] or [23]) that a partial order on a set X is called a *well-quasi-order* if

1. there are no infinite descending chains

2. Any set of pairwise incomparable elements is finite

Well-quasi-orders are a generalization of well orders which are total orders satisfying condition (1) , or equivalently such that any nonempty subset has a smallest element.

We consider the *factor ordering* on sets of words. It is the partial order defined by $u < v$ if u is a factor of v. The first condition in the above definition is then automatically satisfied since the length of words cannot decrease indefinitely.

We have the following result.

Proposition 3 *A recurrent one-sided infinite word x is uniformly recurrent if and only if the factor ordering on the set $F(x)$ is a well-quasi-order.*

Proof. The condition is obviously necessary since the order considered on the set $F(x)$ of factors of a uniformly recurrent word x satisfies the stronger property that the set of elements incomparable with a given $u \in F(x)$ is finite. Conversely, if x is not uniformly recurrent, we can find a sequence $vu_nv \in F(x)$ of words of increasing length such that the only occurrences of v in vu_nv are the two ones as a prefix and a suffix. Then the words vu_nv are incomparable and thus the order is not a well partial order.

■

Note that the factor order on the set A^* itself is not a well-quasi-order (and in fact quite the opposite since it is a maximal set of factors instead of a

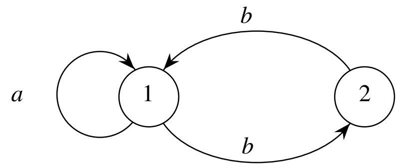

Figure 3: The even system

minimal one). By a classical theorem of Higman (see [39]), it is well-quasi-ordered by a different order, namely the *subword ordering* defined by $u < v$ if $u = u_1u_2 \dots u_n$ and $v = x_0u_1x_1 \cdots x_{n-1}u_nx_n$.

In this section, we have only touched the subject of minimal subshifts. There are many other interesting developments in this direction, such as the one of *Sturmian words* (see the Chapter by Aldo De Luca and Stefano Varricchio in this Handbook). Minimal subshifts are in a sense at the opposite of the subshifts that we are going to study now. To be more precise, minimal subshifts contain no periodic point unless they are finite, whereas in the sofic subshifts introduced below, the set of periodic points is dense.

4 Sofic systems and shifts of finite type

Recall from Section 2 that a sofic system is a subshift S recognized by a finite automaton. For instance the system given on Figure 3 is a sofic system. It consists of all binary sequences such that the length of the blocks of b's between two a's are of even length. This system is sometimes called the *even system*.

A *shift of finite type* is a subshift which is made of all infinite words avoiding a given finite set of blocks. For example, the golden mean system defined in Section 2 as formed by all words on $\{a, b\}$ avoiding bb is a shift of finite type.

Shifts of finite type are closely related to a particular and well-known class of finite automata called local automata. We first give their definition.

Proposition 4 *Let $\mathcal{A}$ be a finite automaton. The following conditions are equivalent.*

1. *The current state on a path is determined by a bounded number of labels in the past and in the future.*
2. *There is at most one infinite path with a given label.*

Moreover, if $\mathcal{A}$ is transitive, the previous conditions are equivalent to:

3. *There is at most one periodic infinite path with a given label.*

Proof. $1 \Rightarrow 2$. If

$$\cdots p_{n-1} \xrightarrow{a_{n-1}} p_n \xrightarrow{a_n} p_{n+1} \cdots$$

is an infinite path, the current state p_n is determined by (a bounded number of) the symbols $\cdots a_{n-1}a_n \cdots$ and thus the label determines the path.

$2 \Rightarrow 1$. If condition 1 is not true, there exist by König's lemma two distinct infinite paths $\cdots p_{n-1} \xrightarrow{a_{n-1}} p_n \cdots$ and $\cdots q_{n-1} \xrightarrow{a_{n-1}} q_n \cdots$ with the same label.

$2 \Rightarrow 3$ is clear.

$3 \Rightarrow 1$. If condition 1 is not true, there exist two distinct infinite paths $\cdots p_{n-1} \xrightarrow{a_{n-1}} p_n \cdots$ and $\cdots q_{n-1} \xrightarrow{a_{n-1}} q_n \cdots$ with the same label. As the paths are distinct, there is an index n such that $p_n \neq q_n$. Since the automaton is finite, both paths use the same pair of states (p, q) infinitely often before time n, and the same pair of states (r, s) infinitely often after time n. If $p \neq q$ or $r \neq s$, this defines two distinct periodic paths with the same label. If not, we have $p = q$ and $r = s$. As the automaton is transitive, there exists a path from r to q and this again defines two distinct periodic paths with the same label.

■

A finite automaton is said to be *local* if it satisfies condition 1 above (equivalent to 2 and also to 3 when the automaton is transitive). The bound on the window size corresponding to condition 1 above can actually be shown to be quadratic as a function of the number of states (see for instance [7] p.45).

We recall that an automaton is called *deterministic* if it admits at most one edge leaving a given state and with a given label.

Proposition 5 *Let $\mathcal{A}$ be a finite deterministic automaton. The following conditions are equivalent.*

1. *The current state on a path is determined by a bounded number of labels in the past.*

2. *The automaton is local.*

Proof. $1 \Rightarrow 2$ is clear.

$2 \Rightarrow 1$. By definition there exist n and m such that the current state on a path is determined by n symbols in the past and m symbols in the future. Any block of $n+m$ symbols determines the final state. Indeed, on a path

$$p_0 \xrightarrow{a_1} \cdots \xrightarrow{a_n} p_n \xrightarrow{a_{n+1}} \cdots \xrightarrow{a_{n+m}} p_{n+m}$$

the state p_n is determined. But then $p_{n+1}, \ldots, p_{n+m}$ are also determined because the automaton is deterministic. Thus $n+m$ symbols in the past determine the current state.

■

The basic example of a deterministic local automaton is the *standard* k-local automaton or De Bruijn graph. Its set of states is the set A^k of words of length k and its edges are the triples (au, b, ub) for $u \in A^{k-1}$ and $a, b \in A$.

The following result shows that shifts of finite type correspond to local automata.

Proposition 6 *A shift of finite type is a sofic system. More precisely, a subshift is of finite type if and only if it is recognized by a local automaton.*

Proof. Let S be a shift of finite type defined by a set of forbidden blocks of maximal length k. We use the standard $(k-1)$-local automaton, erasing all edges (au, b, ub) such that aub contains a forbidden block. In this way, we obtain a finite automaton recognizing S as a sofic system.

Conversely, let $\mathcal{A}$ be a local automaton. By definition, the current state on a path is determined by a bounded number n of symbols in the past and a bounded number m of symbols in the future. Let I be the set of words of length $n+m+1$ which are not labels of paths of $\mathcal{A}$. The sofic system recognized by $\mathcal{A}$ is then equal to the set of bi-infinite words whose blocks of length $n+m+1$ avoid I. As a consequence, it is of finite type.

■

We shall see in the next section how this result can be used to check effectively whether a sofic system is of finite type or not.

As an illustration of the above proposition, we may consider again the golden mean system. The set of allowed blocks of length 2 is $\{aa, ab, ba\}$ thus giving the automaton of Figure 4.

As a particular class of shifts of finite type, one may define a *Markov* shift as a shift of finite type defined by a set of forbidden blocks of length 2. It is clear that the edge shift S_G on a graph G is a Markov shift. The previous example is also a Markov shift since it can be defined by the set of words on $\{a, b\}$ avoiding bb.

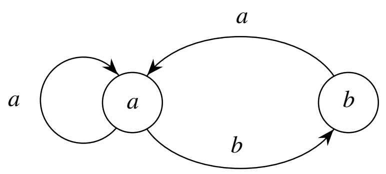

Figure 4: The golden mean system

Any shift of finite type S can be obtained, up to conjugacy, as the edge shift of some finite graph. Indeed, in a local automaton the map from edges to labels is a 1-block conjugacy from the edge shift onto S.

We will prove that both the notion of a shift of finite type as well as that of a sofic system, are invariant under conjugacy. Before proceeding to prove further properties of shifts of finite type and sofic systems, we define a useful way to realize a morphism between subshifts.

A (synchronous) *transducer* on $A \times B$ is a finite automaton (Q, E) with edges labeled by $A \times B$. If (p, a, b, q) is an edge, we say that a is the *input label* and b the *output label*. We will only consider here synchronous transducers instead of the more general notion of a transducer in which the edges are labeled by pairs of words on A, B. We shall also only use transducers such that if two distinct edges (p, a, b, q) and (p, a', b', q) have the same origin p and end q then $a \neq a'$ and $b \neq b'$.

The hypothesis made on transducers implies that, by removing the input labels or the output labels, we get an automaton. The automaton we get by removing the input labels is called the *output automaton* of the transducer and the automaton we get by removing the output labels is called the *input automaton* of the transducer.

Let $f : S \to T$ be a morphism from a subshift S into a subshift T. A transducer $\mathcal{T}$ is said to *realize* f if for all $x \in S$, there is a path with input label x and all of them have output label $y = f(x)$ (we admit the possibility of several paths with input label x).

A morphism $f : S \to T$ between two subshifts $S \subset A^{\mathbb{Z}}$ and $T \subset B^{\mathbb{Z}}$ can be realized by a transducer $\mathcal{T} = (Q, E)$ such that the input automaton is local. Let in fact k be such that f is k-local. Up to some composition with a power of the shift, we may assume for simplicity that f is sequential. The set Q of states of $\mathcal{T}$ is the set of factors of S of length $(k-1)$, and there is an edge between au and ub labeled $(b, f(aub))$.

If S is moreover of finite type, we may choose k large enough to ensure that S is recognized by the input automaton of $\mathcal{T}$.

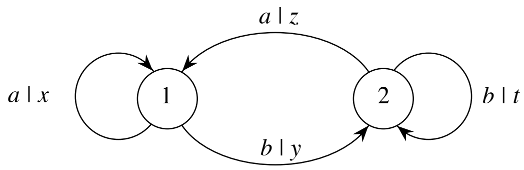

Figure 5: A 2-block map

For example, the transducer of Figure 5 realizes the morphism coding the overlapping blocks of length 2 over the binary alphabet $A = \{a, b\}$ by a symbol from the alphabet $B = \{x, y, z, t\}$.

The transducer that we have associated with a morphism is such that the input automaton is local since it is part of a De Bruijn graph. The following proposition gives a practical method to check whether a morphism between shifts of finite type is a conjugacy.

Proposition 7 *Let S be a shift of finite type and let $f : S \to T$ be a morphism from S onto a subshift T. Let $\mathcal{T}$ be a transducer realizing f and such that its input automaton is a local automaton recognizing S. Then f is a conjugacy iff the output automaton of $\mathcal{T}$ is local.*

Proof. If both the input and the output automaton are local, then f is one-to-one and therefore a conjugacy. Conversely, if the output automaton is not local, there exist two distinct paths with the same output label and thus f is not one-to-one.

■

We now prove that the notion of a shift of finite type is invariant under conjugacy.

Proposition 8 *Any conjugate of a shift of finite type is of finite type.*

Proof. Let $S \subset A^{\mathbb{Z}}$ be a shift of finite type conjugate by f to $T \subset B^{\mathbb{Z}}$. By Proposition 7, f is realized by a transducer whose output automaton is local and recognizes T. By Proposition 6, T is a shift of finite type.

■

The following proposition is analogous to the well-known result that the class of rational languages is the closure of local languages under substitutions (Medvedev's theorem, see [24] p.27 for instance).

Proposition 9 *The factors of shifts of finite type are the sofic systems.*

Proof. A sofic system is by definition recognized by a finite automaton. As such, it is a factor of the edge graph associated with the automaton.

Conversely, let $f : S \to T$ be a morphism from a shift of finite type S onto a subshift T. Up to some composition of f with a power of the shift, we may realize f by a transducer $\mathcal{T}$ which may also be chosen such that its input automaton recognizes S. Then the output automaton of $\mathcal{T}$ recognizes T which is therefore sofic.

∎

We finally obtain the desired result on the invariance under conjugacy of the class of sofic systems as a corollary of the above statement.

Proposition 10 *Any conjugate of a sofic system is sofic.*

Proof. Let S be a sofic system conjugate to T by f. The sofic system S is a factor of some system of finite type U by an onto morphism g. Then T is the image of U by the morphism $f \circ g$ which proves that it is sofic.

∎

5 Minimal automaton of a subshift

The close connection between a subshift S and the set F_S of its finite blocks leads to a possibility of studying the same objects from both the points of view of symbolic dynamics and of finite automata. As a first example of a result with equivalent formulations in terms of symbolic dynamics and in terms of finite automata, the existence of a unique minimal deterministic automaton takes the following form for sofic systems.

We recall that an automaton is said to be transitive if its graph is strongly connected. If $\mathcal{A} = (Q, E)$ is a deterministic automaton, we often denote by $p \cdot a$ the unique state q such that $(p, a, q) \in E$ if it exists. The notation is extended to words. Thus a deterministic automaton is transitive iff for any states p, q there exists a word x such that $p \cdot x = q$.

Proposition 11 *Any sofic system can be recognized by a deterministic automaton. The system is irreducible iff the automaton can be chosen transitive.*

Proof. By Proposition 2 a sofic system S can be recognized by any automaton recognizing the set F_S. It follows that this automaton can be chosen to be deterministic. If the automaton is transitive, the system is clearly irreducible. Conversely, we consider a deterministic automaton $\mathcal{A} = (Q, E, i, Q)$

recognizing the set F_S of factors of an irreducible sofic system S. Let $\mathcal{C}$ be a maximal connected component accessible from i of the automaton (here maximal means that any edge starting in $\mathcal{C}$ also ends in $\mathcal{C}$). Then $\mathcal{C}$ is a deterministic transitive automaton recognizing S. Indeed, any label of a finite path of $\mathcal{C}$ is in F_S. Conversely, let w be a word of F_S and u be the label of a path from i to a state of C. As S is irreducible, there exists a word v such that uvw belongs to F_S. Since the automaton $\mathcal{C}$ is transitive and $\mathcal{A}$ is deterministic, we get that w is the label of a path of C. The set of labels of finite paths in $\mathcal{C}$ is F_S and thus the automaton $\mathcal{C}$ recognizes S.

■

A *reduction* from an automaton $\mathcal{A} = (Q, E)$ onto an automaton $\mathcal{B} = (R, F)$ is a surjective mapping $\rho : Q \to R$ such that $(p, a, q) \in E$ iff $(\rho(p), a, \rho(q)) \in F$. We will show that an irreducible sofic system S has a unique minimal deterministic automaton $\mathcal{A}_S$, in the sense that for any transitive deterministic automaton $\mathcal{B}$ recognizing S, there is a reduction from $\mathcal{B}$ onto the minimal automaton $\mathcal{A}_S$. In particular, the automaton $\mathcal{A}_S$ has the minimum possible number of states. This result is due to Fischer [25] and the minimal automaton is also called the Fischer cover. It was also obtained independently by D. Beauquier [11].

Proposition 12 *Any irreducible sofic system has a unique minimal deterministic automaton.*

The proof of this result does not follow immediately from the corresponding well-known statement for ordinary finite automata because of the absence of an initial state. It relies on the notion of a synchronizing word which allows one to fix an initial state. Let S be a non empty irreducible sofic system and let F_S be its set of finite factors. A word x of F_S is a *synchronizing* word of S iff for all words u, v

$$ux, xv \in F_S \Rightarrow uxv \in F_S.$$

Let $\mathcal{A} = (Q, E)$ be a deterministic automaton. For a finite word $x \in A^*$, we define the *rank* of x as the cardinality of the set $Q \cdot x = \{q \cdot x \mid q \in Q\}$.

Proposition 13 *Any non-empty irreducible sofic system admits a synchronizing word. In fact any word of minimal nonzero rank is synchronizing.*

Proof. Let x be a word of F_S of minimal nonzero rank. If u is a word such that $ux \in F_S$, then $\emptyset \neq Q \cdot ux \subset Q \cdot x$. By minimality of the rank of x, this implies that $Q \cdot ux = Q \cdot x$. Let $ux, xv \in F_S$. Let $p \xrightarrow{x} q \xrightarrow{v} r$ be a path of label xv. Since $Q \cdot x = Q \cdot ux$, there is a path $s \xrightarrow{ux} q$ and thus a path

$$s \xrightarrow{ux} q \xrightarrow{v} r$$

We conclude that $uxv \in F_S$ and thus x is synchronizing.

■

Proof of Proposition 12. We choose a synchronizing word x of S. Let $\mathcal{A}'$ be the minimal automaton of the set of finite words $x^{-1}F_S = \{y \mid xy \in F_S\}$.

We denote by $\mathcal{A}$ the automaton obtained from $\mathcal{A}'$ by allowing all states to be both initial and terminal. The automaton $\mathcal{A}$ recognizes S. Indeed any label of a path in $\mathcal{A}$ is clearly in F_S. Conversely, let $y \in F_S$. Since S is irreducible, there exists a word u such that $xuy \in F_S$. Thus $uy \in x^{-1}F_S$ showing that y is the label of some path in $\mathcal{A}$.

Let now $\mathcal{B}$ be any transitive deterministic automaton recognizing S. The automaton $\mathcal{B}'$ obtained from $\mathcal{B}$ by choosing as initial state a state i in $Q \cdot x$ and all states as terminal states, recognizes the set $x^{-1}F_S$. Indeed, it is clear that the language L recognized by $\mathcal{B}'$ is included in $x^{-1}F_S$.

Conversely, let y be a word of $x^{-1}F_S$. Let z be a word which has nonzero minimal rank in $\mathcal{B}$. As $\mathcal{B}'$ is a transitive automaton, there exists a word u such that zux is a label of a path of $\mathcal{B}'$ leading to state i. Since x is a synchronizing word,

$$zux \in F_S, xy \in F_S \Rightarrow zuxy \in F_S.$$

Then the ranks of $zuxy$, zux, and z are the same. Thus y is the label of a path of $\mathcal{B}'$ beginning at i, since otherwise $zuxy$ would have a nonzero rank strictly smaller than the rank of zux.

We now use the known result that a recognizable set of finite words has a unique minimal automaton. Let $\mathcal{A}'$ be the minimal automaton of the set $x^{-1}F_S$. Since $\mathcal{B}'$ recognizes $x^{-1}F_S$, there is a reduction from $\mathcal{B}'$ onto $\mathcal{A}'$. Thus there is a reduction from $\mathcal{B}$ onto $\mathcal{A}$.

■

The minimal automaton is used to characterize different classes of sofic systems like aperiodic systems [7], almost-of-finite-type shifts [42], or shifts of finite type. We give the result for shifts of finite type.

Proposition 14 *A sofic system is of finite type if and only if its minimal automaton is local.*

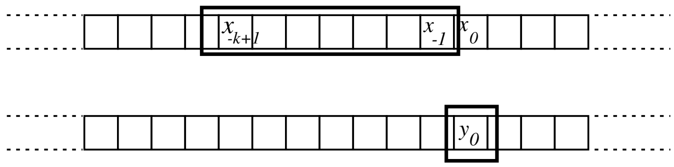

Figure 6: A right-resolving map

Proof. By Proposition 6, if the sofic system admits a local minimal automaton, it is of finite type. Conversely, by the same proposition, a shift of finite type is recognized by a deterministic local automaton $\mathcal{A}$. The minimal automaton $\mathcal{A}_S$ itself is obtained by a reduction from the local automaton $\mathcal{A}$. A reduction transforms a local automaton into a local one since a fixed number of symbols determine the current state in $\mathcal{A}$ and thus in $\mathcal{A}_S$.

■

There is a definition of deterministic automaton which is more abstract and which we introduce now.

Let S and T be two subshifts with S of finite type. Let $f : S \to T$ be a sequential local function and let $\mathcal{T}$ be a transducer realizing f with a local input automaton recognizing S. The morphism f is said to be *right-resolving* if the output automaton is deterministic.

The following statement, whose proof is straightforward, shows that one can define a right-resolving function directly, without reference to the transducer.

Proposition 15 *A sequential k-local function f is right-resolving iff for $y = f(x)$, the values of the block $x_{-k+1} \cdots x_{-1}$ and of the symbol y_0 determine the value of the symbol x_0.*

■

Right-resolving morphisms belong to a broader family of almost one-to-one morphisms called finite-to-one and introduced in Section 6.

A *right-resolving cover* of a sofic system S is a right-resolving morphism $f : T \to S$ from a shift of finite type T onto S. The minimal automaton of a sofic system S defines a right-resolving cover $f : T \to S$ of S which is minimal in the sense that for any other right-resolving cover $g : U \to S$ the subshift T is a factor of U.

6 Codes and finite-to-one maps

In this section, we are going to study the relationship between two notions: finite-to-one maps on the one hand and codes on the other hand (on codes, see also the chapter by Helmut Jürgensen in this Handbook). We will show that the close connection between both notions allows one to prove new results and also to give new and simpler proofs of old ones. We begin with the definition of a finite-to-one map.

A morphism $f : S \to T$ between two subshifts S and T is said to be *finite-to-one* if, for all $y \in T$, the set $f^{-1}(y)$ is finite.

We shall see below that when S is an irreducible shift of finite type, a finite-to-one map is actually *bounded-to-one* in the sense that there is a constant n such that each point of T has at most n pre-images.

We now come to the concepts of codes and unambiguous automata, which are related to the notion of finite-to-one maps.

For a set X of finite words, we denote by X^* the set of all concatenations $x_1x_2\dots x_n$ with $n \geq 0$ and $x_i \in X$.

A set X of finite words is called a *code* if no non-trivial equality holds between the words of X^*. In more precise terms, X is a code iff

$$u_1u_2\dots u_n = v_1v_2\dots v_m,$$

where $u_i, v_j \in X$, implies $n = m$ and $u_i = v_i$ for each index i.

As an example of a code, a *prefix code* is a set X such that no element of X is a prefix of another element of X.

An automaton is said to be *unambiguous* if two paths with the same origin state, the label, and the same final state, are equal. A deterministic automaton is unambiguous since the origin state and the label are sufficient to determine the path. As another particular case, a transitive local automaton is also unambiguous. Indeed, if two distinct paths have the same origin and end, and the same label, we can build two distinct cycles with the same label. This contradicts the hypothesis that the automaton is local.

Let $\mathcal{A} = (Q, E)$ be a transitive automaton and let $i \in Q$ be a particular state. A path from i to i is called *simple* if it does not use i between its endpoints. The set of labels of simple paths from i to i is called the *set of first returns to i*.

The following result establishes the connection between these concepts. It can be stated more generally for arbitrary morphisms between shifts of finite type. We state it, however, in the case of one-block maps for simplicity.

Proposition 16 *Let $\mathcal{A} = (Q, E)$ be a transitive automaton, $i \in Q$ and let X be the set of first returns to i. The following conditions are equivalent.*

1. *The automaton $\mathcal{A}$ is unambiguous.*
2. *X is a code and distinct simple paths from i to i are labeled by distinct elements of X.*
3. *The map going from bi-infinite paths in the automaton to their labels is finite-to-one.*

Proof. It is clear that 1 and 2 are equivalent.

Further, if the automaton is ambiguous, the map f going from paths in the automaton to their labels is not finite-to-one. Thus $3 \Rightarrow 1$.

Finally, if the map f is not finite-to-one, there exists an infinite number of infinite paths having the same image by f. As there is only a finite number of states, an infinity of these paths go through a same state p after the edge of index 0. We can assume that an infinity of them are distinct after the index 0. Let us take $n+1$ of them, where n is the number of states of the automaton. Let us assume that they can all be two by two distinguished before the edge of index m, where m is a positive integer. At least two of them go then through the same state q after this edge and we get two equally labeled paths going from p to q. Thus $1 \Rightarrow 3$.

■

An easy consequence of this result is that a finite-to-one map f from an irreducible shift of finite type S to T is really bounded-to-one.

Proposition 17 *Let $f : S \to T$ be a finite-to-one map realized by a transducer with n states. The number of pre-images of an element of T is bounded by n^2.*

Proof. Let x be an element of T. For each $m \geq 1$ there are at most n^2 paths $p_m \xrightarrow{w_m} q_m$ labeled by $w_m = x_{-m} \cdots x_m$ since such a path is determined by the pair (p_m, q_m). Thus x has at most n^2 pre-images.

■

The connection between codes and subshifts can be considered independently of automata. Let indeed X be a code and let S be the subshift $S_{F(X^*)}$ formed by all bi-infinite words having all its factors in the set $F(X^*)$. One then has the following statement.

Proposition 18 *Let $\mathcal{A}$ be an unambiguous automaton such that X is the code of first returns to some state of $\mathcal{A}$. Then S is the subshift recognized* *by $\mathcal{A}$. If X is finite, then S is the set of bi-infinite words having at least one factorization in words of X.*

■

It is of course still true that the set of bi-infinite words having a factorization in words of X is a subshift, even if X is a set of words which is not a code, provided it is finite. Such a system is sometimes called the *renewal system* generated by X.

Example 2 *If $X = \{a, ba\}$, then S is the golden mean system of Figure 4*

A code $X \subset A^+$ is said to be *maximal* if it is maximal for set-inclusion, that is if $X \subset Y$ for a code Y implies $X = Y$. It is known that a rational code $X \subset A^*$ is maximal iff it is *complete*, i.e. if the set $F(X^*)$ of factors of words of X^* is equal to A^* (see [13] p. 68). A result, due to Ehrenfeucht and Rozenberg, says that any rational code is included in a maximal one (see [13] p. 62).

An analogous proof can be used to obtain the following result [4].

Theorem 3 *If S, T are irreducible shifts of finite type, and $f : S \to T$ is a finite-to-one morphism, then there is an irreducible shift of finite type U containing S and a finite-to-one morphism from U onto T extending f.*

We make two comments about this statement before indicating its proof.

First, the fact that the larger subshift U is required to be irreducible is essential in the statement which would be otherwise trivial: it would be enough to take U to be a disjoint union of S and a copy of T, and to define f on U as being the identity on T.

Second, the link with the theorem of Ehrenfeucht and Rozenberg is the following. Consider the particular case of the map f: paths $\mapsto$ labels in a transitive unambiguous automaton $\mathcal{A}$. Thus f is a finite-to-one morphism from the edge shift S on $\mathcal{A}$ into the full shift $T = A^{\mathbb{Z}}$. Let X be the code defined by the first returns to some state q of $\mathcal{A}$. An embedding $X \subset Y$ of X into a maximal rational code Y can always be obtained by adding states and edges to $\mathcal{A}$ in such a way that Y is the set of first returns to q in the new automaton $\mathcal{B}$.

The set of labels in $\mathcal{B}$ is thus equal to A^* since it is the set of factors of the complete code Y. Thus embedding X into a rational maximal code Y corresponds to an extension of the finite-to-one map f to a larger subshift in such a way that it becomes surjective.

We now give an indication of the proof of Theorem 3.

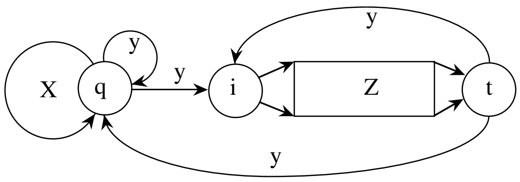

Figure 7: The resulting transducer

Proof of Theorem 3. We first make the hypothesis that $S \subset A^{\mathbb{Z}}$ and $T \subset B^{\mathbb{Z}}$ are Markov shifts. This is true up to conjugacy and thus we may make this hypothesis. We also suppose that f is a one-block map. This is true again up to a conjugacy. Under these assumptions, we may realize f as the map paths $\mapsto$ labels in an automaton $\mathcal{A}$ whose set of states is $Q = A$. Also, T is recognized by an automaton $\mathcal{B}$ with set of states equal to B.

Let $q \in Q$ be a particular state of $\mathcal{A}$. Let X be the image under f of the set of first returns to q in $\mathcal{A}$. Let $b = f(q)$ and Y be the set of first returns to b in $\mathcal{B}$. Thus $X \subset Y^*$. Moreover Y^* and X^* are rational subsets of B^*.

If $T = f(S)$, there is nothing to prove. Otherwise, there is a word $y \in Y^*$ such that

1. y is unbordered, i.e. y has no non-trivial prefix which is also a suffix.

2. $y \notin F_{f(S)} = f(F_S)$

The construction of such a word y follows the same lines as the analogous construction in [13] p. 64.

Let $Z \subset B^*$ be the set

$$Z = Y^* - X^* - B^* y B^*$$

We build an irreducible shift of finite type U containing S and an extension g of f by considering the automaton $\mathcal{C}$ of Figure 7 where the component called Z is actually a finite automaton recognizing the set Z (with i as initial state and t as final state). The subshift U is the set of infinite paths in G and the function g is the map paths $\mapsto$ labels in $\mathcal{C}$.

Then the extension g of f to U satisfies the following properties.

1. g maps U into T.

2. g is finite-to-one.

3. g is onto.

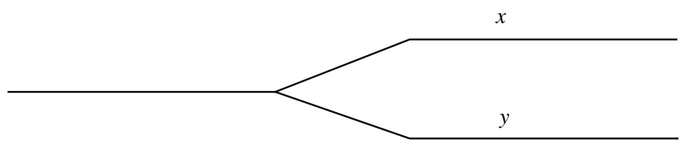

Figure 8: Two left-asymptotic words x and y

Thus g is a finite-to-one morphism from U onto T extending f. This completes the sketch of the proof of Theorem 3.

■

The theorem of Ehrenfeucht and Rozenberg corresponds to the case where T is the full shift. The more general case where a code X is constrained to be included in a fixed factorial set F has been studied by A. Restivo who has shown that the results known previously extend to this case [53].

The paper [4] contains other results of the same kind. One of them deals with right-closing morphisms, a notion intermediate between finite-to-one and right-resolving which is defined precisely as follows.

A function $f : S \to T$ is *right-closing* if whenever $x, y \in S$ have a common left-infinite tail and $f(x) = f(y)$ then $x = y$ (see Figure 8).

Right-closing morphisms correspond to automata with bounded delay in the same way as right-resolving morphisms correspond to deterministic automata. The result on right-closing morphisms proved in [4] corresponds to the result on codes with bounded deciphering delay proved by V. Bruyère, L. Wang and L. Zhang in [21]: any rational code with finite deciphering delay can be embedded into a rational maximal one (see also [20]).

Finally, the paper [4] contains an analogous result on extending morphisms belonging this time to the class of *biclosing* morphisms i.e. morphisms that are both left and right closing. This is related to a result proved recently by L. Zhang: any rational biprefix code can be embedded into a rational maximal one [62].

7 State splitting and merging

We have seen that one may associate with every irreducible sofic system a minimal automaton that recognizes it. The computation of such a minimal automaton may be performed using one of the standard algorithms for automaton minimization, such as Moore's algorithm or Hopcroft's algorithm (see [3] for example). All minimization algorithms consist in some kind of state identification since the states of the resulting minimal automaton are equivalence classes of states (equivalent states are those with the same future). In this section, we introduce an operation on symbolic systems, called state merging which allows one to identify states of the automaton recognizing a sofic system S in such a way that the resulting system is conjugate to S. The inverse operation is called state splitting. These concepts are due to F. Williams [61]. We first define the operation on the edge shift of a graph.

Let $G = (Q, E)$ be a graph. Let $q \in Q$ and let I (resp. O) be the set of edges entering q (resp. going out of q). Let $O = O' + O''$ be a partition of O. The operation of *(output) state splitting* relative to (O', O'') transforms G into the automaton $G' = (Q', E')$ where $Q' = Q \cup q'$ is obtained from Q by adding a new state q' and E' is defined as follows.

$$E' = E - O' + I' + U$$

with $I' = \{(p, q') \mid (p, q) \in I\}$ and $U = \{(q', q'') \mid (q, q'') \in O'\}$.

Thus G' is obtained from G by

1. leaving unchanged all edges not adjacent to q.
2. giving q' copies of the input edges of q (this is the set I').
3. distributing the output edges of q between q' and q according to the partition of O into O' and O''.

The operation of input state splitting is analogous. It uses a partition $I = I' + I''$ of the edges entering q instead of a partition of the edges going out of q.

Example 3 Let us consider the edge shift of the graph represented on Figure 9 on the left side. The graph on the right side is obtained by state splitting on vertex 1 with the effect that a vertex 1 before a 2 is transformed into a 3 . The edges going out of 1 are split into two parts: the loop on 1 on the first hand remains unchanged. The edge going from 1 to 2 becomes an edge from 3 to 2. The edges incoming at 1 and 3 are identical.

The operation of *(output) state merging* is the inverse of that of (output) state splitting. Formally, let $G = (Q, E)$ be a graph and let $q, q' \in Q$ be two states such that the edges coming into q and q' are the same (except for the end). We merge q and q' in a single state q having the same input edges as the former state q (with loops (q, q) for the edges of the form (q', q)). The output edges are obtained as the union of those of q and q'.

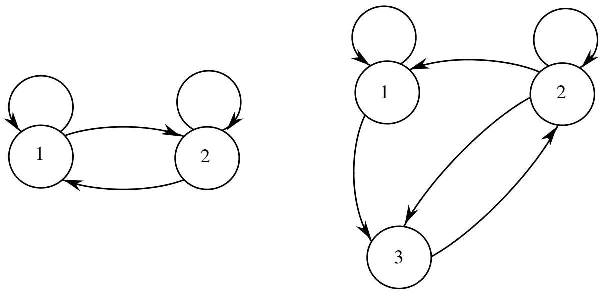

Figure 9: An output split of state 1

Finally, the operation of input state merging is the inverse of input state splitting. It is the operation linked with the minimization of automata, as we shall see shortly.

The following statement is easy to prove.

Proposition 19 *Let G' be obtained from G by state splitting. Then the edge graphs S_G and $S_{G'}$ are conjugate. More precisely, a state splitting is a 2-block map whose inverse (the merge) is a 1-block map.*

Proof. It is clear that the merge is a 1-block map. Consider now an output split at q according to a partition $O = O' + O''$ of the set O of edges going out of q. The image of an edge is itself unless it is in I or O'. In the first case it is transformed into an edge of I' if it is followed by an edge of O'. In the second case, it is transformed into an edge of U. Thus a sliding window of length 2 is enough to perform the transformation.

■

Let S be a sofic system and let $\mathcal{A} = (Q, E)$ be an automaton recognizing S. The operations of state splitting and merging on the graph transfer to operations on the automaton. In an output split, the labels of the edges coming into and going out of q are transferred to the edges incident to the new state q'. More precisely, we have

$$I' = \{(p, a, q') \mid (p, a, q) \in I\} \text{ and } U = \{(q', a, q'') \mid (q, a, q'') \in O'\}.$$

Example 4 Let $\mathcal{A}$ be the automaton with two states represented on the left of Figure 10. There are two edges going out of state 1: $(1, a, 1)$ and $(1, b, 2)$. We transfer the second one to a new state called 3 which receives the same input edges as state 1. The result is represented on the right side of Figure 10.

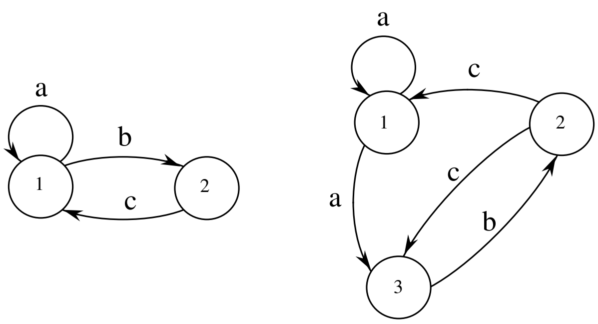

Figure 10: An output split

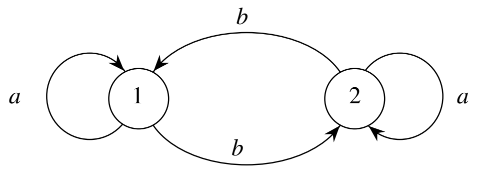

Figure 11: A non minimal automaton

Let $\mathcal{A}$ be a deterministic automaton. A sequence of input state mergings produces a deterministic automaton $\mathcal{B}$ with fewer states than $\mathcal{A}$ and still equivalent, i.e. recognizing the same subshift. However, it is not true in general that the minimal automaton $\mathcal{C}$ can be reached in this way. This is illustrated by the following example.

Example 5 Let us consider the automaton $\mathcal{A}$ of Figure 11. The two states cannot be merged since they have distinct output edges (taking the labels into account). Actually, the map from paths to labels is 2-to-1 and the subshift recognized is the full shift. If it were possible to reach the minimal automaton with splits and merges, the result would be a conjugacy between paths and labels.

There is however one interesting case where the minimization can be obtained by merges.

Proposition 20 *Let $\mathcal{A}$ be a transitive and deterministic automaton. If $\mathcal{A}$ is local, there is a sequence of state merges that transforms $\mathcal{A}$ into a minimal equivalent automaton.*

Proof. We recall from Section 5 that the minimal automaton can be computed by an identification of states having the same future i.e. the same set $F_q = \{w \in A^* \mid q \cdot w \text{ is well-defined}\}$.We suppose that $\mathcal{A}$ is not minimal and thus that there are distinct states q, q' with the same future. Let x be a word of maximal length such that $q \cdot x \neq q' \cdot x$. Then for all $a \in A$, we have $(q \cdot x) \cdot a = (q' \cdot x) \cdot a$ and thus $q \cdot x, q' \cdot x$ can be merged.

∎

The following result is due to F. Williams [61].

Theorem 4 *Any conjugacy between shifts of finite type can be obtained, up to a renaming of the symbols, as a composition of splits and merges.*

Proof. We first consider the particular conjugacy which is the coding by overlapping blocks of fixed length, say k. This particular map can certainly be obtained by a series of splits. This allows us to obtain the shift map itself since it can be obtained through a coding in blocks of length 2.

It is therefore enough to prove that we can obtain a 1-block map whose inverse is sequential as a composition of splits and merges. Such a map is the map from paths to labels in a deterministic local automaton $\mathcal{A}$. Let k be such that k symbols in the past determine the current state. We can, up to a coding by blocks of length $k+1$, make the labels all distinct. The result is a 1-block map whose inverse is also 1-block. Such a map is a renaming of the symbols.

∎

Theorem 4 shows in particular that the group of automorphisms of a shift of finite type is generated by splits and merges, but through possibly larger shifts. For a primitive shift of finite type, the automorphism group contains every finite group [33]. It is not known whether, on a finite alphabet, it is generated by the shift and its elements of finite order, although this has been conjectured (on automorphisms see [19] or [46]).

The operation of state splitting plays an important role in the applications of symbolic dynamics to coding (see [41],[2]). In the next section, we shall see how it is related to the isomorphism of shifts of finite type.

8 Shift equivalence

In this section, we discuss the problem of the conjugacy of shifts of finite type. In particular, we shall give an algebraic formulation of the equivalence in terms of matrices. In fact, R. F. Williams [61] introduced two equivalence relations on matrices allowing one to formulate the relation of conjugacy on the subshifts in algebraic terms.

Two square matrices M, N with nonnegative coefficients are said to be *elementary shift equivalent* if there exist two nonnegative integral matrices U, V such that

$$M = UV, \qquad N = VU \tag{3}$$

Note that M and N may have different dimensions.

Then M and N are called *strong shift equivalent* if there is a chain of elementary shift equivalences between M and N.

Let $S = S_G, T = T_H$ be two edge shifts given by the adjacency matrices M, N of the graphs G, H. We then have the following result.

Theorem 5 (Williams [61]) *Two shifts of finite type S and T given by the matrices M, N as above are conjugate iff the matrices M, N are strong shift equivalent.*

Proof. Let first S and T be conjugate. By Theorem 4, there exists a sequence of splits and merges transforming S into T. It is therefore enough to prove that splits and merges correspond to shift equivalences on matrices. We consider an output split of a state q. We suppose that q corresponds to the last index of M. Then

$$M = \begin{bmatrix} M' \\ x+y \end{bmatrix}, \qquad N = \begin{bmatrix} N' & z & z \end{bmatrix}$$

where x, y are row vectors, z is a column vector and the decomposition of the last row of M corresponds to the partition of the edges going out of q. We have $M = UV$ and $N = VU$ where

$$U = \begin{bmatrix} 1 & 0 & & & & 0 \\ 0 & \ddots & & & & 0 \\ & & 1 & & & 0 \\ & & & \ddots & 0 & 0 \\ & & & 0 & 1 & 1 \end{bmatrix}, \qquad V = \begin{bmatrix} M' \\ x \\ y \end{bmatrix} = \begin{bmatrix} N' & z \end{bmatrix}$$

Thus conjugate subshifts have strong shift equivalent matrices. We prove the converse in the case where the matrices M and N have coefficients 0 or 1 or, equivalently when G and H are ordinary graphs. The general case is not substantially more difficult but the notation is more cumbersome.

Since neither G or H has multiple edges, we may consider S and T as formed by sequences of vertices instead of sequences of edges. Let $f : S \to T$ be the function defined as follows. For $x \in S$ and $n \in Z$, if $x_n = i, x_{n+1} = j$ then (i, j) is an edge of the graph G. Since $M = UV$ there is exactly one vertex k of H such that $U_{ik} = V_{kj} = 1$. We define a 2-block map by $f(i, j) = k$. The inverse is obtained in the same way and S, T are thus conjugate.

■

Example 6 For instance, in Example 3, the adjacency matrices M and N of the graphs are elementary shift equivalent since

$$\begin{aligned} M &= \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 1 & 1 \\ 0 & 1 \end{bmatrix}, \\ N &= \begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 1 \\ 0 & 1 & 0 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 1 & 1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} \end{aligned}$$

The relation of strong shift equivalence is not easy to compute because in a chain $(M_1, M_2, \ldots, M_n)$ of elementary equivalences, the dimensions of the matrices M_i are not a priori bounded. It is not known whether it is recursively computable or not.

We now come to the second equivalence relation on square matrices. Two square matrices M and N are called *shift equivalent*, denoted $M \sim_k N$, if there exist two nonnegative integral matrices U, V and an integer k such that

$$\begin{aligned} MU &= UN, & NV &= VM \\ M^k &= UV, & N^k &= VU \end{aligned} \tag{4}$$

The relation $\sim_k$ is transitive. Let indeed $M \sim_k N$ and $N \sim_l P$, let $MU = UN, NV = VM, M^k = UV, N^k = VU$ and $NR = RP, PS = SN, N^l = RS, P^l = SR$. Then

$$\begin{aligned} MUR &= URP, & PSV &= SVM, \\ M^{k+l} &= (UR)(SV), & P^{k+l} &= (SV)(UR) \end{aligned}$$

The integer k is called the *lag* of the equivalence. It is clear that $k = 1$ corresponds to an elementary shift equivalence and thus that strong shift equivalence implies shift equivalence.

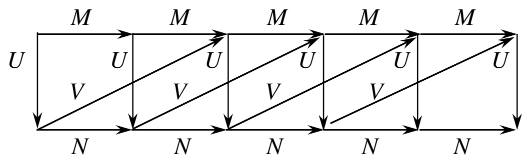

Figure 12: Shift equivalence (lag 2)

The converse was posed by Williams as a problem: does shift equivalence imply strong shift equivalence? It was shown by Kim and Roush in [35] that the answer is negative in general. However the subshifts of their counterexample are reducible and the conjecture is still pending for irreducible shifts of finite type.

The definition of shift equivalence given above asks for the existence of nonnegative integral matrices U, V such that Equations 4 are satisfied. If we only require that U, V have integer coefficients (possibly negative), we get the a priori weaker notion of *shift equivalence over* $\mathbb{Z}$.

Both notions coincide however for primitive matrices, i.e. matrices such that the associated shifts are primitive (Parry and Williams [48]).

Proposition 21 *Two primitive integral matrices are shift equivalent iff they are shift equivalent over* $\mathbb{Z}$.

It should be noted that the index k in Eq. (4) can be larger over $\mathbb{N}$ than over $\mathbb{Z}$ as shown in the following example.

Example 7 Let

$$M = \begin{bmatrix} 1 & 3 \\ 2 & 1 \end{bmatrix}, \qquad N = \begin{bmatrix} 1 & 6 \\ 1 & 1 \end{bmatrix}$$

Then M and N are similar over $\mathbb{Z}$ since if

$$P = \begin{bmatrix} 2 & 3 \\ 1 & 1 \end{bmatrix}, \qquad P^{-1} = \begin{bmatrix} -1 & 3 \\ 1 & -2 \end{bmatrix}$$

then $M = P^{-1}NP$. Thus M, N are shift equivalent over $\mathbb{Z}$. They are thus shift equivalent. We may indeed choose $k = 3$ and $U = P^{-1}N^3, V = P$.

It has been shown by Kirby Baker that the matrices M and N are indeed strong shift equivalent (see [38] page 238). However the least number of pairs of elementary shift equivalent matrices used to go from M to N is 7.

The matrices M and N are particular cases of the more general case:

$$M = \begin{bmatrix} 1 & k \\ k-1 & 1 \end{bmatrix}, \qquad N = \begin{bmatrix} 1 & k(k-1) \\ 1 & 1 \end{bmatrix}$$

It is easy to see that these matrices are shift equivalent over $\mathbb{Z}$ and thus over $\mathbb{N}$. However, it is not known whether they are strong shift equivalent.

It is interesting to note that shift equivalence has been proved decidable by Kim and Roush [34]. Also, the problem of comparing, inside a given semigroup, the various relations generalizing the group conjugacy has been studied by several authors (see [22]). It is however a different problem here since the relation is defined among square matrices of different dimensions and not inside a semigroup.

9 Entropy

The notion of entropy in information theory has its root in the work of Shannon. It is defined as a measure of uncertainty and depends on the use of probabilities. Its use in symbolic dynamics, under the name of *topological entropy*, is independent of probabilities. It is an invariant under conjugacy as we shall now see.

The *entropy* of a nonempty subshift S is the limit

$$h(S) = \lim_{n\to\infty} \frac{1}{n} \log s_n$$

where s_n is the number of blocks of length n appearing in the elements of S.

This limit is well defined. In fact, if S is a subshift, we have $s_{n+m} \leq s_n s_m$. Then $\log(s_{n+m}) \leq \log(s_n) + \log(s_m)$. We get that the sequence $(\log(s_n))_{n>0}$ is a subadditive sequence of strictly positive integers and, as a consequence of this fact, that the sequence $(\log(s_n)/n)_{n>0}$ converges.

The entropy of a set of finite words X is the superior limit

$$h(X) = \limsup_{n\to\infty} \frac{1}{n} \log \alpha_n$$

where α_n is the number of words of length n belonging to X.

The following statement gives a method to compute the entropy of a irreducible sofic system provided we can compute the entropy of a set of the form X^* where X is a code.

Proposition 22 *Let S be an irreducible sofic system recognized by a transitive unambiguous automaton $\mathcal{A}$. Let X be the code of first returns to some state of $\mathcal{A}$. The entropy of S is equal to the entropy of X^*.*

Proof. Let us denote by l_n the number of words of X^* of length n and let s_n be the number of blocks of length n in S. For any positive integer n, we have $l_n \leq s_n$ since any word of X^* is a block of an element of S. This proves that $h(X^*) \leq h(S)$.

Let k be the number of states of $\mathcal{A}$. Let w_n be a block of length n of an element of S. As the graph of $\mathcal{A}$ is strongly connected, there exist two words u and v, of lengths $|u|, |v|$ satisfying $|u| + |v| \leq k$, such that uw_nv belongs to X^*. This allows us to associate to each block of length n of a sequence of S, a word of length at most $(n+k)$ of X^*, which admits the block as factor. As the number of positions of the block in the word is at most $k+1$, we get $s_n \leq (k+1)(l_n + l_{n+1} + \cdots + l_{n+k})$. It follows from this that $h(S) \leq h(X^*)$.

■

Let L be a set of finite words over an alphabet A and let $f_n = \mathrm{card}(L \cap A^n)$ be the number of words of length n in L. Then

$$f_L(z) = \sum_{n \geq 0} f_n z^n$$

is the generating series of the sequence f_n. One can show (see [13] p. 42) that a set X is a code iff the following equality holds.

$$f_{X^*} = \frac{1}{1 - f_X}$$

The following result allows one to compute the entropy of a set of the form X^* where X is a code.

Theorem 6 *Let S be an irreducible sofic system. Let $\mathcal{A}$ be a transitive unambiguous automaton recognizing S and let X be the code of first returns to some state of $\mathcal{A}$. The entropy of S is $\log(1/r_X)$ where r_X is the unique positive root of $f_X(r) = 1$.*

Proof. By Proposition 22 we have $h(S) = h(X^*)$. The entropy of X^* is equal to $\log(1/r)$ where r is the convergence radius of f_{X^*}. Since X is a code, we have $f_{X^*} = (1 - f_X)^{-1}$. As any $\mathbb{R}^+$-rational series which is not a polynomial has its convergence radius as a pole, we get that r_X is the unique positive root of $f_X(z) = 1$.

■

Theorem 6 gives a method to compute the entropy of an irreducible sofic system. One may compute the number r by solving the equation $f_X(z) = 1$. This is effective when X is a rational code or equivalently when S is a sofic system.

An alternative method to compute r is to use the fact that $1/r$ is the maximal eigenvalue of the matrix associated to any unambiguous automaton recognizing S. In fact, let M be a matrix with real coefficients. The spectral radius ρ of M is the maximal modulus of its eigenvalues. One has (see [29] for example)

$$\rho = \limsup \sqrt[n]{\| M^n \|}$$

where $\| M \|$ is any norm of the matrix M. If we choose the particular norm equal to the sum of modulus of all coefficients, then the number of blocks of length n in the edge shift of the automaton is $\| M^n \|$. Thus the entropy of S is $\log \rho$.

This is true also when the automaton is not transitive. Thus, the entropy of S is the maximum of the entropies of the irreducible components of S.

Example 8 Let S be the even system represented on Figure 3. We have $X = \{a, bb\}$ and $f_X(z) = z + z^2$. Thus $r = 1/\varphi$ where φ is the golden mean. Accordingly, the maximal eigenvalue of the matrix

$$M = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}$$

is φ.

We prove the following result which implies in particular that the entropy is invariant under conjugacy (a fact that could also be proved directly).

Proposition 23 *Let S and T be two irreducible shifts of finite type and let $f : S \to T$ be a morphism. The following conditions are equivalent.*

1. *f is finite-to-one.*
2. *$h(S) = h(T)$*

Proof. $1 \Rightarrow 2$. Let $\mathcal{A}$ be a transducer realizing f with a local input automaton. By Proposition 16, the output automaton is unambiguous. Let q be a state of $\mathcal{A}$, let X be set of first returns to q in the input automaton and let Y be the set of first returns to q in the output automaton. Then X and

Y have the same number of words of each length and thus $h(X^*) = h(Y^*)$. Hence $h(S) = h(T)$ by Proposition 22.

$2 \Rightarrow 1$. Let $\mathcal{A}$ be a transducer realizing f. We suppose that the output automaton of $\mathcal{A}$ is ambiguous. If there exist two edges in $\mathcal{A}$ which only differ by the input, then we can remove one of these edges without changing the map realized. This removal decreases the entropy of the set of returns X^* in the input automaton since it increases strictly r_X.

To handle the general case, we consider two paths u, v of length k with the same label x. We shall consider the automaton $\mathcal{A}^k$ which has the same set of states as $\mathcal{A}$ but the set of words of length k as alphabet with the transitions induced by those of $\mathcal{A}$. Obviously, the entropies of the systems S^k, T^k recognized by the input and output automata of $\mathcal{A}^k$ satisfy $h(S^k) = k\,h(S), h(T^k) = k\,h(T)$. We may choose k to be prime to the gcd of the cycle lengths of the automaton. In this way the automaton $\mathcal{A}^k$ is still transitive. We are thus in the situation considered at the beginning.

■

If the alphabet A has k elements, then $r \geq 1/k$ or equivalently

$$f_X(1/k) \leq 1 \tag{5}$$

which is Kraft's inequality.

It is well-known that one has equality in (5) iff the code X is maximal (see [13]). This can be seen as equivalent to the fact that the sofic system S associated to X is equal to the full shift on k symbols.

There are actually several results for which one may indifferently use either the vocabulary of subshifts or that of codes and automata.

As an example, we have the following result, due to Hedlund [33].

Proposition 24 *Let S and T be irreducible sofic systems and let $f : S \to T$ be a morphism. Any two of the following conditions imply the third.*

1. *f is finite-to-one.*
2. *f is onto.*
3. *$h(S) = h(T)$*

This statement is the direct counterpart of the following one for codes (see [13] p. 69).

Proposition 25 *Let X be a recognizable subset of A^* and let $k = Card(A)$. Any two of the three following statements imply the third.*

1. *X is a code.*

2. $f_X(1/k) = 1$.

3. *X is complete.*

For any series f with positive coefficients satisfying (5), it is well-known that there exists a prefix code X on a k-symbol alphabet such that $f = f_X$.

Recall that a series $f = \sum_{k\geq 0} f_k z^k$ is said to be $\mathbb{N}$-rational if there exists a nonnegative integral $n \times n$ matrix M, and two vectors $i \in \mathbb{N}^{1\times n}, t \in \mathbb{N}^{n\times 1}$ such that identically $f_k = iM^k t$.

If X is a rational code, then f_X satisfies (5) and is additionally an $\mathbb{N}$-rational series. What can be said conversely? It is tempting to conjecture that for any $\mathbb{N}$-rational series $f = \sum_{n\geq 0} \alpha_n z^n$, such that $f(1/k) \leq 1$, there exists a rational prefix code X over a k-letter alphabet such that $f = f_X$.

A particular case of this is proved in [49].

We recall that an algebraic number is a root r of a monic polynomial whose coefficients are rational numbers. Among these polynomials there is a unique one $p(z)$ of minimal degree, called the *minimal polynomial* of r. The algebraic conjugates of r are the roots of $p(z)$.

When X is a recognizable code, $f_X(z)$ is a rational series and thus r_X is an algebraic number. It is indeed the largest root of the numerator of $1 - f_X(z)$. In the case of a finite code, one has additional properties of this algebraic number.

Proposition 26 *If X is a finite code, then r_X has no other real positive algebraic conjugate.*

Proof. Since $f_X(z)$ has positive coefficients, the function $z \mapsto f_X(z) - 1$ is strictly increasing from -1 to $+\infty$ for $z \in [0, +\infty[$. Thus there can be only one positive real number r such that $f_X(r) = 1$. Hence, the algebraic integer r_X has the property that it has no other positive real algebraic conjugate.

■

Thus r_X is the only real positive root of its minimal polynomial. It is also its root of minimal modulus since for any other root ρ of $1 - f_X(z)$, one has $f_X(r) = 1 \leq f_X(|\rho|)$ whence $r \leq |\rho|$.

The above property can be used to prove that some systems cannot be obtained as a renewal system generated by a finite code, as shown in the following example.

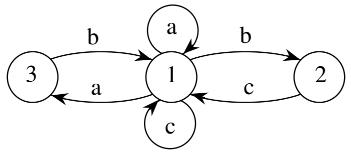

Figure 13: The renewal system S

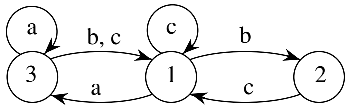

Figure 14: The minimal automaton of S

Example 9 Let $X = \{a, bc, ab, c\}$. The set X is not a code since abc has two factorizations. Let S be the renewal system generated by X. An automaton recognizing S is represented on Figure 13.

The determinization and further minimization of the automaton of Figure 13 gives the automaton of Figure 14. The minimal automaton of S is local and therefore S is a shift of finite type (although the automaton of Figure 13 is not local). Let M be the adjacency matrix of $\mathcal{A}$. We have

$$M = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & 0 \\ 2 & 0 & 1 \end{bmatrix}$$

The characteristic polynomial of M is $p(z) = z^3 - 2z^2 - 2z + 1 = (1 + z)(1 - 3z + z^2)$. The roots of $p(z)$ are $-1, \varphi^2, \hat{\varphi}^2$ where φ is the golden mean and $\hat{\varphi}$ its conjugate. The entropy of S is $\log \varphi^2$. Since φ^2 has a positive real conjugate which is $\hat{\varphi}^2$, the system S cannot be generated by a finite code. This example is due to J. Ashley (unpublished). It answers negatively a conjecture formulated by A. Restivo asserting that a finitely generated renewal system which is at the same time a shift of finite type can be generated by a finite code.

The following result, due to D. Handelman gives a converse to Proposition 26.

Theorem 7 (Handelman [32]) *Let r be an algebraic integer strictly less in modulus than any of its conjugates. The number r is a root of a polynomial* *of the form $1 - zp(z)$ with p a polynomial with non-negative coefficients iff it has no other real positive algebraic conjugate.*

The theorem does not cover the case where r has conjugates of the same modulus. In this case, the set of roots of modulus r is of the form $r\epsilon$ where ϵ is any of the p-th roots of 1. The generalization of Handelman's theorem to this case has been obtained by F. Bassino ([6],[5]).

The polynomials of the form $1-zp(z)$ with p non-negative are a particular case of *polynomials with one sign change*, i.e. such that, after deleting the zero coefficients, the sequence $(a_0, a_1, \dots)$ of coefficients has exactly one sign change. The result of D. Handelman is actually stated in this more general case.

The proof of Theorem 7 uses properties of *log concave* polynomials, also called unimodal, which are polynomials such that the sequence $(a_0, a_1, \cdots)$ of coefficients satisfies ${a_i}^2 > a_{i+1}a_{i-1}$. It is also related to a result of Poincaré according to which, if a polynomial p with real coefficients has exactly one positive real root, then there exists a polynomial P such that the product pP has one sign change.

Theorem 7 has been extended to study the star-height of one-variable rational series. It is known that the star-height of a one-variable $\mathbb{N}$-rational series is at most two (see [56]). F. Bassino has used Theorem 7 to obtain a characterization of the series of star-height one under the assumption that they have a unique pole of minimal modulus ([6], [5]).

Much of the study of shifts of finite type is linked to that of *positive matrices*. Indeed, a shift of finite type is given by a finite graph which in turn is given by its adjacency matrix. The shift of finite type itself corresponds to a class of equivalent matrices. The study of this equivalence has motivated a lot of research (see [18] for a survey).

One aspect of this research is the study of the cone of positive matrices inside the algebra of all integer matrices. The basic properties of positive matrices were obtained long ago by Perron and Frobenius. The theorem says essentially that a nonnegative real matrix has an eigenvalue of maximal modulus which is a positive real number. If it is irreducible, it has a corresponding eigenvector with positive coefficients. And if it is primitive, there is only one eigenvalue of maximal modulus.

In a more recent work, Handelman [31] has proved a kind of converse of the Perron Frobenius theorem. A matrix is said to be *eventually positive* if some power has only strictly positive coefficients. A *dominant eigenvalue* is an eigenvalue α such that $\alpha > |\beta|$ for any other eigenvalue.

Theorem 8 (Handelman [31]) *A matrix with integer coefficients is conju-*

gate to an eventually positive matrix iff it has a dominant eigenvalue of multiplicity one.

This result is very close to one due to M. Soittola (see [56]) characterizing $\mathbb{N}$-rational series in one variable among $\mathbb{Z}$-rational series. We quote it for series having a *minimal pole* i.e. a unique pole with minimal modulus.

Theorem 9 (Soittola) *A $\mathbb{Z}$-rational series with nonnegative integer coefficients*

$$f = \sum_{n \geq 0} \alpha_n z^n$$

having a minimal pole is $\mathbb{N}$-rational.

In [50] it is shown that both theorems can be deduced from the construction of a basis in which the matrices have the appropriate properties. It also gives at the same time a proof that a one-variable $\mathbb{N}$-rational series has at most star-height 2.

10 The road coloring problem

A classical notion in automata theory is that of a synchronizing word. We recall that, given a deterministic and complete automaton $\mathcal{A}$ on a state set Q, a word w is said to be synchronizing if the state reached from any state $q \in Q$ after reading w is independent of q. The automaton itself is called *synchronizing* if there exists a synchronizing word.

A maximal prefix code X is called *synchronizing* if it is the set of first returns to the initial state in a synchronizing automaton.

It is clear that a necessary (but not sufficient) condition for an automaton to be synchronizing is that the underlying graph is primitive (i.e. strongly connected and the gcd of the cycle lengths is 1). The same holds for a prefix code (which is called aperiodic if it is maximal and the gcd of the word lengths is 1).

The following problem was raised in [1]. Let G be a finite directed graph with the following properties:

1. All the vertices of G have the same outdegree.

2. G is primitive.

The problem is to find, for any graph G satisfying these hypotheses, a labeling turning G into a synchronizing deterministic automaton. The problem is called the *road-coloring problem* because of the following interpretation: a labeling (or coloring) making the automaton synchronizing allows a traveler lost on the graph G to follow a path which is a succession of colors leading back home regardless of where he actually started.

In terms of symbolic dynamics, such a labeling defines a right-resolving map $f : S_G \to A^{\mathbb{Z}}$ from the shift of finite type S_G onto a full shift which is 1-to-1 almost everywhere. In this context, a synchronizing word is called a *resolving block* and we say that f has a resolving block if there exists such a synchronizing word.

The road-coloring problem itself remains still open but some results have been obtained that we describe now.

The following result is proved in [1]. It shows that if the road coloring problem can perhaps not be solved on a given graph G, it can be solved on a subshift conjugate to S_G.

Theorem 10 *If G is a primitive graph with all vertices of the same outdegree, there exists a conjugate T of S_G and a right-resolving map $f : T \to A^{\mathbb{Z}}$ from T onto the full shift on k symbols having a resolving block.*

. The proof consists in considering the subshift $S_{G^{(n)}}$, for large enough n, where $G^{(n)}$ is the graph having as edges the paths of length n in G.

Actually, Theorem 10 can also be obtained using a result on codes that we now describe.

A set $X \subset A^*$ is called *thin* if there exists a word $w \in A^*$ such that $A^*wA^* \cap X = \emptyset$, that is to say that w does not appear as a block in the words of X. It is known that every rational code is thin (see [13] p. 69).

The following result is due to Schützenberger [57] (see also [52]).

Theorem 11 (Schützenberger) *Let $k \geq 2$ be an integer and let $\alpha = (\alpha_n)_{n\geq 1}$ be a sequence of integers. Let $f = \sum_n \alpha_n z^n$ and let ρ_α denote the radius of convergence of the series f.*

There exists a thin maximal and synchronizing prefix code X on a k-letter alphabet such that $f_X = f$ iff the following conditions are satisfied.

1. $\sum_{n\geq 1} \alpha_n k^{-n} = 1$,
2. $\rho_\alpha > 1/k$,
3. *the integers n such that $\alpha_n \neq 0$ are relatively prime,*

Actually, conditions (1) and (2) are equivalent to the existence of a thin maximal prefix code such that $f_X = f$ and condition (3) holds then iff X is aperiodic. It is shown in [57] that, under the hypotheses of the theorem, one may choose an integer n and two symbols $a, b \in A$ such $a^n \in X$ and that the set

$$Y = a^n \cup (X \cap a^* b a^*) \tag{6}$$

contains a synchronizing word for X.

Theorem 11 can be used to prove Theorem 10. Indeed, let us consider a particular vertex i of the graph G and let α_n be the number of simple paths from i to i in G. Then the sequence $\alpha = (\alpha_n)_{n \geq 1}$ satisfies the conditions of Theorem 11. We can use a state splitting to be able to label $n+1$ paths of the resulting graph by the words of the set Y of Eq (6).

The following result is proved in [52], providing a partial answer to the problem. For a prefix code X, we denote by T_X the usual (unlabeled) tree whose leaves correspond to the elements of X. Several prefix codes may thus correspond to the same tree according to the choice of a labeling of the sons of each node.

Theorem 12 *Given a finite aperiodic prefix code X, there is a synchronizing prefix code Y such that $T_X = T_Y$.*

The proof uses heavily the theorem of C. Reutenauer ([54]) on the noncommutative polynomial of a code.

In terms of symbolic dynamics, Theorem 12 solves positively the road-coloring problem for those graphs G satisfying the following additional assumption:

(3) All vertices except one have indegree 1.

Other results on the road coloring problem have been obtained and in particular, by G. O'Brien [47] and by J. Friedman [27].

11 The zeta function of a subshift

Besides the entropy, there is an invariant of symbolic dynamical systems which takes into account the number of elements of a given period.

Let (S, σ) be a subshift and let

$$P_n = \{x \in S \mid \sigma^n(x) = x\}$$

be the set of points of period dividing n.

The *zeta function* of a subshift S is the series

$$\zeta_S(z) = \exp \sum_{n>0} \frac{p_n}{n} z^n$$

with $p_n = \mathrm{card}(P_n)$.

Two subshifts have the same zeta function iff they have the same number of elements of each period. Since a conjugacy preserves the period of the points, the zeta function is an invariant under conjugacy. This information is useful for separating non equivalent systems. It is actually stronger than entropy for sofic systems.

In fact, let S be an irreducible sofic subshift recognized by an unambiguous automaton $\mathcal{A}$. Let X be the code of first returns to some state of $\mathcal{A}$. Let t_n be the number of words of length n in X^*. Then

$$t_n \leq p_n \leq s_n$$

By definition, we have $h(S) = \lim \frac{1}{n} \log s_n$ and $h(X^*) = \lim \frac{1}{n} \log t_n$. By Proposition 22 we have $h(X^*) = h(S)$ and thus

$$\lim \frac{1}{n} \log t_n = \lim \frac{1}{n} \log p_n = \lim \frac{1}{n} \log s_n$$

Hence $h(S)$ is determined by $\zeta(S)$.

Another invariant related to the number of minimal forbidden blocks of each length is studied in [10].

It was proved by R. Bowen and O. Lanford in [17] that the zeta function of a shift of finite type S is a rational series. They proved actually the following proposition.

Proposition 27 *Let S be the edge shift of a graph G. If M is the adjacency matrix G,*

$$\zeta_S(z) = det(I - Mz)^{-1}$$

Proof. As we can associate bijectively to each sequence x of S such that $\sigma^n(x) = x$, a cycle of the graph of length n, we get that $s_n = \mathrm{trace}(M^n)$. The computation of the zeta function of S can now be done as follows, where

I is the identity matrix of the same size as M.

$$\begin{aligned}
\zeta_S(z) &= \exp \sum_{n>0} \frac{1}{n} \operatorname{trace}(M^n) z^n = \exp \operatorname{trace}(\sum_{n>0} \frac{1}{n} (Mz)^n) \\
&= \det \exp(\sum_{n>0} \frac{1}{n} (Mz)^n) = \det \exp \log(I - Mz)^{-1} \\
&= \det(I - Mz)^{-1}
\end{aligned}$$

■

It was proved later by A. Manning [40] and also by R. Bowen [17] that the zeta function of a sofic system is also rational.

This has motivated further investigations in several directions. On one hand, J. Berstel and C. Reutenauer have extended the result of Manning to the case of a generalization of the zeta function to some formal series [14] and proved the rationality of the generalized zeta function for cyclic languages.

On the other hand, M. P. Béal has introduced an operation on finite automata, the *external power*, allowing one to obtain the generalized zeta function of a sofic system as a combination of the values obtained on the different external powers, (see [7] and [8]). This gives a proof of the formula of Bowen [17] and this proof can be extended to the case of cyclic languages [9].

More recently, C. Reutenauer [55] has obtained new results showing that the zeta function of a sofic system is not only rational but even $\mathbb{N}$-rational. He has also extended his results to more general symbolic systems, introduced by D. Fried under the name of *finitely presented systems* [26].

12 Circular codes, shifts of finite type and Krieger embedding theorem

There is a close connection between shifts of finite type and a particular class of codes called circular codes (see [13] for a more comprehensive introduction).

A set $X \subset A^+$ is called a *circular code* if any circular word over A has at most one decomposition as a product of words from X. More precisely, X is a circular code if for any $x_1, x_2, \dots, x_n, y_1, y_2, \dots, y_m \in X$ and $p \in A^*, s \in A^+$

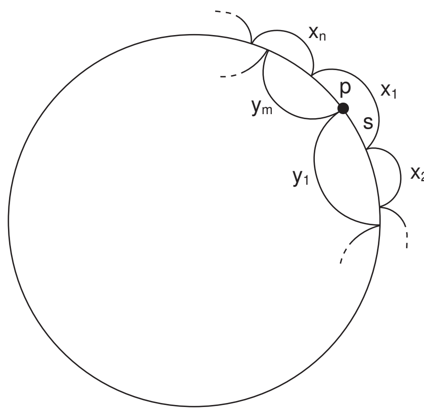

Figure 15: Two circular factorizations

the equalities

$$\begin{aligned} sx_2x_3\dots x_np &= y_1y_2\dots y_m, && (7)\\ x_1 &= ps && (8) \end{aligned}$$

imply $n = m$, $p = \epsilon$ and $x_1 = y_1, \dots, x_n = y_n$. Indeed, Equalities 7,8 corresponds to two decompositions of a word written on a circle as represented on Figure 15.

The following statement relates circular codes and local automata (see [7] p. 65).

Proposition 28 *Let X be a finite code and let $\mathcal{A}$ be an unambiguous strongly connected automaton such that X is the set of first returns to some state of $\mathcal{A}$. The following conditions are equivalent.*

1. *X is a circular code.*
2. *The automaton $\mathcal{A}$ is local.*

Proof. Two cycles in $\mathcal{A}$ with equal labels define two factorizations of a circular word and conversely. Thus the result follows from Proposition 4.

■

The next result relates circular codes and shifts of finite type.

Proposition 29 *The renewal system generated by a finite circular code is a shift of finite type.*

Proof. Let S be the renewal system generated by the circular code X. By the previous proposition, there exists a local automaton recognizing S. By Proposition 6, S is a shift of finite type.

■

Let, as in Section 9, $\alpha_n = \mathrm{Card}(X \cap A^n)$ and $f_X(z) = \sum_{n \geq 0} \alpha_n z^n$. Let S be the system generated by X, which is the set of all bi-infinite words having a factorization in words of X. The following statement shows that the zeta function of S can be easily computed.

Proposition 30 *Let X be a finite circular code. The zeta function of S is given by*

$$\zeta_S = (1 - f_X)^{-1} \tag{9}$$

Proof. Since X is circular, S is a shift of finite type (Proposition 28). Let M be the adjacency matrix of the graph of the flower automaton of X. By Proposition 27, we have

$$\zeta_S(z) = \det(I - Mz)^{-1}$$

It is well-known for any graph made of n cycles of lengths $(\alpha_1, \ldots, \alpha_n)$ with one common vertex that

$$\det(I - Mz) = 1 - f_X$$

The result follows from the two above equations.

■

The number p_n of points of S of period dividing n can be computed from Formula (9). Indeed, we have

$$\sum_n \frac{p_n}{n} z^n = \log(\zeta_S)$$

and thus, by Formula (9)

$$\begin{aligned}
\sum_n \frac{p_n}{n} z^n &= -\log(1 - f_X) \\
&= -\log(1 - \sum_n \alpha_n z^n) \\
&= \sum_n s_n z^n
\end{aligned}$$

with

$$s_n = \sum_{k=1}^{n} \frac{1}{k} \alpha_n^{(k)}, \qquad \alpha_n^{(k)} = \sum_{i_1 + \ldots + i_k = n} \alpha_{i_1} \cdots \alpha_{i_n}$$

Thus we have

$$p_n = \sum_{i=1}^{n} \frac{n}{i} \alpha_n^{(i)} \tag{10}$$

The number of points having period exactly n is denoted by $q_n(S)$ or simply q_n. Obviously p_n and q_n are related by $p_n = \sum_{d|n} q_d$. The following inequalities are then satisfied for all $n \geq 1$.

$$q_n \leq l_n(k) \tag{11}$$

where $l_n(k)$ is the number of points of period exactly n in the full shift over k symbols. Indeed, the number of points in S having period exactly n cannot exceed the total number of points of period n in the full shift on k symbols.

The numbers $l_n(k)$, sometimes called Witt numbers, satisfy $\sum_{d|n} d l_d(k) = k^n$ or equivalently, by Möbius inversion formula,

$$l_n(k) = \frac{1}{n} \sum_{d|n} \mu(d) k^{n/d}$$

The length distribution $(\alpha_n)_{n \geq 1}$ of a circular code satisfies inequalities stronger than (5) which are obtained after expressing in (11) the integers p_n in terms of the α_n using Formula (10).

The first inequalities are, in explicit form:

$$\begin{array}{rcl} \alpha_1 & \leq & k \\ \alpha_2 + \dfrac{1}{2}(\alpha_1^2 - \alpha_1) & \leq & \dfrac{1}{2}(k^2 - k) \\ \cdots & \leq & \cdots \end{array}$$

It was shown by Schützenberger (see [13] p. 343) that these inequalities characterize the length distributions of circular codes.

Theorem 13 (Schützenberger) *A sequence α_n of integers is the length distribution of a circular code over a k-letter alphabet iff it satisfies the above inequalities.*

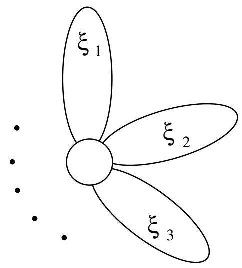

Figure 16: A renewal graph

This is linked in a very interesting way with a theorem of Krieger which gives a necessary and sufficient condition for the existence of a strict embedding of a shift of finite type into another one.

Theorem 14 (Krieger [36]) *Let S and T be two shifts of finite type. Then there exists an isomorphism f from S into T with $f(S) \neq T$ iff*

1. $h(S) < h(T)$.
2. *for each* $n \geq 1$, $q_n(S) \leq q_n(T)$

A proof of Krieger's theorem can be found in the book of D. Lind and B. Marcus [38].

We explain here the connection between Krieger's theorem and the theorem of Schützenberger on circular codes.

Given a finite sequence $\xi = (\xi_i)_{1 \leq i \leq n}$, let G be the *renewal graph* made of n simple cycles of lengths $\xi_1, \ldots, \xi_n$ with exactly one common point (see Figure 16). Any circular code on the alphabet A with length distribution ξ defines an isomorphism from the edge shift S_G into $A^{\mathbb{Z}}$. Indeed, there is a labeling of G which defines a flower automaton $\mathcal{A}$ for X. By Proposition 28, the subshift recognized by $\mathcal{A}$ is of finite type. The map from paths to labels is therefore an embedding of S_G into the full shift $A^{\mathbb{Z}}$.

Thus Theorem 13 gives a proof of Theorem 14 in the particular case where S is a the edge shift defined by a renewal graph and T is a full shift.

References

[1] Roy Adler, I. Goodwin, and Benjamin Weiss. Equivalence of topological Markov shifts. *Israel J. Math.*, 27:49–63, 1977.

[2] Roy L. Adler, D. Coppersmith, and M. Hassner. Algorithms for sliding block codes. *IEEE Trans. Inform. Theory*, IT-29:5–22, 1983.

[3] Alfred V. Aho, John E. Hopcroft, and Jeffrey D. Ullman. *The Design and Analysis of Computer Algorithms.* Addison Wesley, 1974.

[4] Jonathan Ashley, Brian Marcus, Dominique Perrin, and Selim Tuncel. Surjective extensions of sliding block codes. *SIAM J. Discrete Math.*, 6:582–611, 1993.

[5] Frédérique Bassino. Non-negative companion matrices and star-height of $\mathbb{N}$-rational series. *Theoret. Comput. Sci.*, 1996. (to appear).

[6] Frédérique Bassino. Star-height of an $\mathbb{N}$-rational series. In C. Puech and R. Reischuk, editors, *STACS 96*, volume 1046 of *Lecture Notes in Computer Science*, pages 125–135. Springer Verlag, 1996.

[7] Marie-Pierre Béal. *Codage Symbolique.* Masson, 1993.

[8] Marie-Pierre Béal. Puissance extérieure d'un automate déterministe, application au calcul de la fonction zêta d'un système sofique. *R.A.I.R.O-Informatique Théorique et Applications*, 29:85–103, 1995.

[9] Marie-Pierre Béal, Olivier Carton, and Christophe Reutenauer. Cyclic languages and strongly cyclic languages. In C. Puech and R. Reischuk, editors, *STACS 96*, volume 1046 of *Lecture Notes in Computer Science*, pages 49–59. Springer Verlag, 1996.

[10] Marie-Pierre Béal, Filippo Mignosi, and Antonio Restivo. Minimal forbidden words and symbolic dynamics. In C. Puech and R. Reischuk, editors, *STACS 96*, volume 1046 of *Lecture Notes in Computer Science*, pages 555–566. Springer Verlag, 1995.

[11] Danièle Beauquier. Minimal automaton for a factorial, transitive rational language. *Theoret. Comput. Sci.*, 67:65–73, 1989.

[12] Tim Bedford, Michael Keane, and Caroline Series, editors. *Ergodic Theory, Symbolic Dynamics and Hyperbolic Spaces.* Oxford University Press, 1991.

[13] Jean Berstel and Dominique Perrin. *Theory of codes*. Academic Press, 1985. (also available on http://www-litp.ibp.fr/berstel/LivreCodes).

[14] Jean Berstel and Christophe Reutenauer. Zeta functions of formal languages. *Trans. Amer. Math. Soc.*, 321:533–546, 1990.

[15] Garett D. Birkhoff. Quelques théorèmes sur le mouvement des systèmes dynamiques. *Bull. Soc. Math. France*, 40:305–323, 1912.

[16] François Blanchard and Alejandro Maass. On dynamical properties of generalized cellular automata. In Ricardo Baeza-Yates and Eric Goles, editors, *LATIN 95: Theoretical Informatics*, volume 911 of *Lecture Notes in Comput. Sci.*, pages 84–98, 1995.

[17] Rufus Bowen and O. Lanford. Zeta functions of restrictions of the shift transformation. *Proc. Symp. Pure Math.*, 14:43–50, 1970.

[18] Michael Boyle. Symbolic dynamics and matrices. In S. Friedland, V. Brualdi, and V. Klee, editors, *Combinatorial and Graph-Theoretic Problems in Linear Algebra*, volume 50 of *IMA Volumes in Mathematics and its Applications*. Springer Verlag, 1993.

[19] Michael Boyle, Douglas Lind, and D. Rudolph. The automorphism group of a shift of finite type. *Trans. of Amer. Math. Soc.*, 306:71–114, 1988.

[20] Véronique Bruyère and Michel Latteux. Variable-length maximal codes. (ICALP 96, Lecture Notes in Computer Sci., Springer Verlag, to appear), 1996.

[21] Véronique Bruyère, L. Wang, and Liang Zhang. On completion of codes with finite deciphering delay. *European J. Combin.*, 11:513–521, 1990.

[22] Christian Choffrut. Conjugacy in free inverse monoids. *Int. J. Alg. Comput.*, 3:169–188, 1993.

[23] Aldo De Luca and Stefano Varricchio. *Combinatorics on Words and Regularity Conditions*. Springer Verlag, 1996. (to appear).

[24] Samuel Eilenberg. *Automata,Languages and Machines*, volume A. Academic Press, 1974.

[25] R. Fischer. Sofic systems and graphs. *Monatshefte Math.*, 80:179–186, 1975.

[26] David Fried. Finitely presented dynamical systems. *Ergod. Th. Dynam. Sys.*, 7:489–507, 1987.

[27] Joel Friedman. On the road coloring problem. *Proc. Amer. Math. Soc.*, 110:1133–35, 1990.

[28] Christiane Frougny and Boris Solomyak. Finite beta-expansions. *Ergod. Th. & Dynam. Syst.*, 12:45–82, 1992.

[29] F. R. Gantmacher. *The Theory of Matrices*. Chelsea, 1960.

[30] Andrew Gleason. Semigroups of shift register matrices. *Mathematical Systems Theory*, 25:253–267, 1992. (notes of a course given at Princeton en 1960).

[31] David Handelman. Positive matrices and dimension groups affiliated to C^*-algebras and topological Markov chains. *J. Operator Theory*, 6:55–74, 1981.

[32] David Handelman. Spectral radii of primitive integral companion matrices and log-concave poynomials. In Peter Walters, editor, *Symbolic Dynamics and its Applications*, volume 135 of *Contemporary Mathematics*, pages 231–238. Amer. Math. Soc., 1992.

[33] George Hedlund. Endomorphisms and automorphisms of the shift dynamical system. *Math. Syst. Theory*, 3:320–375, 1969.

[34] K. H. Kim and F. W. Roush. Decidability of shift equivalence. In *Symbolic Dynamics*, volume 1342 of *Lecture Notes in Mathematics*, pages 374–424. Springer, 1988.

[35] K. H. Kim and F. W. Roush. William's conjecture is false for reducible matrices. *J. Amer. Math. Soc.*, 5:213–215, 1992.

[36] Wolfgang Krieger. On the subsystems of topological Markov chains. *Ergod. Th. & Dynam. Syst.*, 2:195–202, 1982.

[37] Joseph Kruskal. The theory of well-quasi-ordering: a frequently rediscovered concept. *J. Comb. Theory (ser. A)*, 13:297–305, 1972.

[38] Douglas Lind and Brian Marcus. *An Introduction to Symbolic Dynamics and Coding.* Cambridge University Press, 1996.

[39] M. Lothaire. *Combinatorics on Words.* Cambridge University Press, 1983.

[40] Anthony Manning. Axiom a diffeomorphisms have rational zeta function. *Bull. London Math. Soc.*, 3:215–220, 1971.

[41] Brian Marcus. Factors and extensions of full shifts. *Monats. Math.*, 88:239–247, 1979.

[42] Brian Marcus. Sofic systems and encoding data. *IEEE Trans. Inf. Theory*, IT-31:366–377, 1985.

[43] J. C. Martin. Substitution minimal sets. *Amer. J. Math.*, 93:503–526, 1971.

[44] Marston Morse. Recurrent geodesics on a surface of negative curvature. *Trans. Amer. Math. Soc.*, 22:84–110, 1921.

[45] Marston Morse and George Hedlund. Symbolic dynamics. *Amer. J. of Math.*, 3:286–303, 1936.

[46] Masakazu Nasu. *Textile Systems for Endomorphisms and Automorphisms of the Shift*, volume 546 of *Memoirs of Amer. Math. Soc.* Amer. Math. Soc., 1995.

[47] G.L. O'Brien. The road coloring problem. *Israel J. Math*, 39:145–154, 1981.

[48] William Parry and R. F. Williams. Block coding and a zeta function for finite markov chains. *Proc. London Math. Soc.*, 35:483–495, 1977.

[49] Dominique Perrin. Arbres et séries rationnelles. *C. R. Acad. Sci. Paris*, 309:713–716, 1989.

[50] Dominique Perrin. On positive matrices. *Theoretical Computer Science*, 94:357–366, 1992.

[51] Dominique Perrin. Symbolic dynamics and finite automata. In Jiri Wiedermann and Petr Hajek, editors, *Mathematical Foundations of Computer Science 1995*, volume 969 of *Lecture Notes in Computer Science*, pages 94–104. Springer Verlag, 1995.

[52] Dominique Perrin and Marcel-Paul Schützenberger. Synchronizing prefix codes and automata and the road coloring problem. In Peter Walters, editor, *Symbolic Dynamics and its Applications*, volume 135 of *Contemporary Mathematics*, pages 295–318. Amer. Math. Soc., 1992.

[53] Antonio Restivo. Codes and local constraints. *Theoret. Comput. Sci.*, 72:55–64, 1990.

[54] Christophe Reutenauer. Non commutative factorization of variable length codes. *J. Pure and Applied Algebra*, 36:157–186, 1985.

[55] Christophe Reutenauer. $\mathbb{N}$-rationality of zeta functions. *Advances in Mathematics*, 1996. (to appear).

[56] Artp Salomaa and Matti Soittola. *Automata-Theoretic Aspects of Formal Power Series.* New York, Springer-Verlag, 1978.

[57] Marcel-Paul Schützenberger. On synchronizing prefix codes. *Inform and Control*, 11:396–401, 1967.

[58] Wolfgang Thomas. Automata on infinite objects. In J. van Leeuwen, editor, *Handbook of Theoretical Computer Science*, volume vol. B, Formal models and semantics, pages 135–191. Elsevier, 1990.

[59] Peter Walters, editor. *Symbolic Dynamics and its Applications*, volume 135 of *Contemporary Mathematics.* Amer. Math. Soc., 1992.

[60] Benjamin Weiss. Subshifts of finite type and sofic systems. *Monats. Math.*, 77:462–474, 1973.

[61] Frank Williams. Classification of subshifts of finite type. *Ann. of Math.*, 98:120–153, 1973. (Errata ibid. **99**:380-381,1974).

[62] Liang Zhang and Zhonghui Shen. Completion of recognizable bifix codes. *Theoret. Comput. Sci.*, 145:345–355, 1995.

Index