ZETA FUNCTIONS OF FINITE-TYPE-DYCK SHIFTS ARE $\mathbb{N}$-ALGEBRAIC

MARIE-PIERRE BÉAL, MICHEL BLOCKELET, AND CĂTĂLIN DIMA

Abstract. Constrained coding is a technique for converting unrestricted sequences of symbols into constrained sequences, *i.e.* sequences with a predefined set of properties. Regular constraints are described by finite-state automata and the set of bi-infinite constrained sequences are finite-type or sofic shifts. A larger class of constraints, described by sofic-Dyck automata, are the visibly pushdown constraints whose corresponding set of bi-infinite sequences are the sofic-Dyck shifts. An algebraic formula for the zeta function, which counts the periodic sequences of these shifts, can be obtained for sofic-Dyck shifts having a right-resolving presentation. We extend the formula to all sofic-Dyck shifts. This proves that the zeta function of all sofic-Dyck shifts is a computable $\mathbb{Z}$-algebraic series. We prove that the zeta function of a finite-type-Dyck shift is a computable $\mathbb{N}$-algebraic series, *i.e.* is the generating series of some unambiguous context-free language. We conjecture that the result holds for all sofic-Dyck shifts.

Zeta function, shift spaces, sofic-Dyck shifts, sofic shifts, visibly pushdown languages, $\mathbb{N}$-algebraic series

1. Introduction

Applications of constrained coding are often confined to sequences drawn from regular languages. They are usually described by finite sets of forbidden blocks (finite-type constraints) or by finite-state automata (finite labelled graphs) and are then called sofic constraints since the set of these sequences is a symbolic dynamical system called a sofic shift [33].

Nevertheless, there are classes of constrained sequences going beyond the sofic constraints (see for instance [38], [18]). A particular interesting one is the class of visibly pushdown sequences which corresponds to sofic-Dyck shifts and to visibly pushdown constraints. Visibly pushdown languages [1, 2] are a strict subclass of unambiguous context-free languages. They are rich enough to model many languages like XML languages. They form a natural and meaningful class in between the class of regular languages and the class of unambiguous context-free languages, extending the parenthesis languages [37], [28], the bracketed languages [20], and the balanced languages [10], [11].

Date: March 2, 2014.

This work is supported by the French National Agency (ANR) through "Programme d'Investissements d'Avenir" (Project ACRONYME n°ANR-10-LABX-58) and through the ANR EQINOCS.

The class of these languages is moreover tractable and robust. For instance the class of visibly pushdown languages over a same pushdown alphabet is stable by intersection and complementation.

In [23], [32], [24], Inoue, Krieger, and Matsumoto introduced and studied special classes of shifts of sequences characterized by a context-free language of factors called Markov-Dyck shifts. They generalize the Dyck shifts whose blocks are finite factors of well parenthesized words. In [24], [30] (see also [22] and [21]), the authors considered also extensions of Markov-Dyck shifts by constructing shifts from sofic systems and Dyck shifts, and shifts defined by $\mathcal{R}$-graphs. Synchronization properties of Markov-Dyck or Motzkin shifts are obtained in [29], [35], [36], [31].

In [9], we introduced the class of sofic-Dyck shifts which contains the above classes. We showed that they are characterized by a visibly pushdown language. An interesting property of sofic-Dyck constraints is the fact that they have finite-state labelled graph presentations, called Dyck automata, which are equipped with a set of so called matched edges encoding the context-free aspect of the constraints. A subclass of sofic-Dyck shift is the class of finite-type Dyck [8] shifts obtained by adding a finite set of forbidden blocks to some Dyck constraint.

In [9] we performed a computation of the zeta function of a sofic-Dyck shift having a right-resolving presentation. The zeta function is a conjugacy invariant of shifts which counts the periodic sequences (see [3] for an overview of zeta functions). The formula of the zeta function of a shift of finite type is due to Bowen and Lanford [15]. Formulas for the zeta function of a sofic shift were obtained by Manning [34] and Bowen [14]. Proofs of Bowen's formula can be found in [33] and [7, 6]. An $\mathbb{N}$-rational expression of the zeta function of a sofic shift has been obtained in [39] (see also [12] for zeta functions of formal languages and [25] for the zeta function of periodic-finite-type shifts). The zeta functions of the Dyck shifts were determined by Keller in [26]. For the Motzkin shifts, where some unconstrained symbols are added to the alphabet of a Dyck shift, the zeta function was determined by Inoue in [23]. In [32], Krieger and Matsumoto obtained an expression for the zeta function of a Markov-Dyck shift. The computation in [9] extends Krieger and Matsumoto's formula to sofic-Dyck shifts having a right-resolving presentation.

In this paper, we extend the formula of the zeta function to all sofic-Dyck shifts, getting rid of presentation restrictions. This proves that the zeta function of all sofic-Dyck shifts is a computable $\mathbb{Z}$-algebraic series. We also prove that the zeta function of a finite-type-Dyck shift is a computable $\mathbb{N}$-algebraic series, *i.e.* is the generating series of some unambiguous context-free language. Actually, we prove a more precise result. The zeta function of a finite-type Dyck shift is the generating series of a visibly pushdown language, a strict subclass of unambiguous context-free languages. Rational and algebraic formal power series are well-behaved objects with many interesting properties (see [5, 4], [13], [17], [40]). We conjecture that the zeta function of every sofic-Dyck shift is ℕ-algebraic.

The paper is organized as follows. In Section 2, we give the definition of the different types of constraints considered in this paper, including the notions of sofic-Dyck shifts and finite-type-Dyck shifts. In Section 3.1 we recall the general formula of the zeta function of a sofic-Dyck wich extends the formula of the zeta function of Markov-Dyck shifts from Krieger and Matsumoto [32]. In Section 3.2 we describe how to obtained reduced presentations for sofic-Dyck which have good properties for the computation of the zeta function. In Section 3.3, we recall the formula of the zeta function of a finite-type-Dyck shift obtained in [9]. It was proved in [9] for shifts having a right-resolving presentation and holds now for all finite-type-Dyck shifts. In Section 3.4, we prove the ℕ-algebraicity of the series. In Section 3.5, we recall the formula of the zeta function of a sofic-Dyck shift obtained in [9]. The restriction on the presentations is removed. We give an example of computation of the series.

2. Sofic-Dyck constraints

In the section we define the class of finite-type and sofic-Dyck shifts introduced in [9] and [8]. Basic notions of symbolic dynamics can be found in [33, 27].

Let A be a finite alphabet. The set of finite sequences or words over A is denoted by A^* and the set of nonempty finite sequences or words over A is denoted by A^+. More generally, if L is a set of words over an alphabet A, then L^* is the set of concatenation of words of L, the empty word included.

Let F be a set of finite words over the alphabet A. We denote by X_F the set of bi-infinite sequences of $A^{\mathbb{Z}}$ avoiding each word of F. The set X_F is called a *shift* (or *subshift*). When F can be chosen finite (resp. regular), the shift X_F is called a *shift of finite type* (resp. *sofic*). When F is a visibly pushdown language, the set X_F is called a sofic-Dyck shift [9]. The set of finite factors of a shift X is denoted $\mathcal{B}(X)$, its elements being called *blocks* of X.

Visibly pushdown languages are context-free languages of finite words accepted by visibly pushdown automata defined as follows.

We consider an alphabet A which is partitioned into three disjoint sets of symbols, the set A_c of call symbols , the set A_r of return symbols, and the set A_i of internal symbols. A *visibly pushdown automaton* on finite words over $A = (A_c, A_r, A_i)$ is a tuple $M = (Q, I, \Gamma, \Delta, F)$ where Q is a finite set of states, $I \subseteq Q$ is a set of initial states, Γ is a finite stack alphabet that contains a special bottom-of-stack symbol $\perp$, $\Delta \subseteq (Q \times A_c \times Q \times (\Gamma \setminus \{\perp\}) \cup (Q \times A_r \times Q \times \Gamma \times Q) \cup (Q \times A_i \times Q)$, and $F \subseteq Q$ is a set of final states. A transition (p, a, q, γ), where $a \in A_c$ and $\gamma \neq \perp$, is a push-transition. On reading a, the stack symbol γ is pushed onto the stack and the control changes from state p to q. A transition (p, a, γ, q) is a pop-transition. The symbol γ is read from the top of the stack and popped. If $\gamma = \bot$, the symbol is read but not popped. A transition (p, a, q) is a local action.

A stack is a nonempty finite sequence over Γ ending in $\bot$. A *run* of M labelled by $w = a_1 \mathinner{..} a_k$ is a sequence $(p_0, \sigma_0) \cdots (p_k, \sigma_k)$ where $p_i \in Q$, $\sigma_i \in (\Sigma \setminus \{\bot\})^* \bot$ for $1 \le i \le k$ and such that:

- If $a_i \in A_c$, then there are $\gamma_i \in \Gamma$ and $(p_{i-1}, a_i, p_i, \gamma_i) \in \Delta$ with $\sigma_i = \gamma_i \cdot \sigma_{i-1}$.
- If $a_i \in A_r$, then there are $\gamma_i \in \Gamma$ and $(p_{i-1}, a_i, \gamma_i, p_i) \in \Delta$ with either $\gamma_i \neq \bot$ and $\sigma_{i-1} = \gamma_i \cdot \sigma_i$ or $\gamma_i = \bot$ and $\sigma_i = \sigma_{i-1} = \bot$.
- If $a_i \in A_i$, then $(p_{i-1}, a_i, p_i) \in \Delta$ and $\sigma_i = \sigma_{i-1}$.

A run is *accepting* if $p_0 \in I$, $\sigma_0 = \bot$, and the last state is final, *i.e.* $p_k \in F$. A word over A is *accepted* if it is the label of an accepting run.

A sofic-Dyck shift is presented by a Dyck-automaton. A *Dyck automaton* over an alphabet $A = (A_c, A_r, A_i)$ is a pair $(\mathcal{A}, M)$ of an automaton (or a directed labelled graph) $\mathcal{A} = (Q, E, A)$ over A where Q is the finite set of states or vertices, $E \subset Q \times A \times Q$ is the set of edges, and with a set M of pairs of edges $((p, a, q), (r, b, s))$ such that $a \in A_c$ and $b \in A_r$. The set M is called the set of *matched edges*. We define the *graph semigroup* S associated to $(\mathcal{A}, M)$ as the semigroup generated by the set $E \cup \{x_{pq} \mid p, q \in Q\} \cup \{0\}$ with the following relations.

$$
\begin{aligned}
0s = s0 &= 0 && s \in S,\\
x_{pq}x_{qr} &= x_{pr} && p, q, r \in Q,\\
x_{pq}x_{rs} &= 0 && p, q, r, s \in Q, q \neq r,\\
(p, \ell, q) &= x_{pq} && p, q, \in Q, \ell \in A_i,\\
(p, a, q)x_{qr}(r, b, s) &= x_{ps} && (p, a, q), (r, b, s) \in M,\\
(p, a, q)x_{qr}(r, b, s) &= 0 && (p, a, q), (r, b, s) \in (A_c \times A_r) \setminus M,\\
(p, a, q)(r, b, s) &= 0, && q \neq r, a, b \in A,\\
x_{pp}(p, a, q) &= (p, a, q) && p, q \in Q, a \in A,\\
(p, a, q)x_{qq} &= (p, a, q) && p, q \in Q, a \in A,\\
x_{pq}(r, a, s) &= 0 && a \in A,\ q \neq r,\ s \neq t.\\
x(r, a, s)x_{tu} &= 0 && a \in A,\ q \neq r,\ s \neq t.
\end{aligned}
$$

If π is a finite path of $\mathcal{A}$, we denote by $f(\pi)$ its image in the graph semigroup S. A finite path π of $\mathcal{A}$ such that $f(\pi) \neq 0$ is said to be an *admissible path* of $(\mathcal{A}, M)$. A finite word is *admissible* for $(\mathcal{A}, M)$ if it is the label of some admissible path of $(\mathcal{A}, M)$. A bi-infinite path is *admissible* if all its finite factors are admissible. The *sofic-Dyck shift presented* by $(\mathcal{A}, M)$ is the set of labels of bi-infinite admissible paths of $(\mathcal{A}, M)$.

A word w labeling an admissible path π such that $f(\pi) = x_{pq}$ for some states p, q is called a *well-matched* word of $(\mathcal{A}, M)$. It is a *prime Dyck word* is any nonempty shorter prefix of w is not well-matched.

Example 1. Consider the sofic-Dyck shift over A presented by the graph shown in Fig. 1 and with the following matched edges. The (-edge is matched with the)-edge and the [-edge is matched with the]-edge. This shift is called the *Motzkin shift*. A sequence is a block (or is allowed) if it is a factor of a well-parenthesized word, the internal letters being omitted. For instance $(\, i \, [\, i \, i \,] \, [\, i \,) \, (\, ($ is a block while the patterns $(\,]$ or $(\, i \, i \,]$ are forbidden. The block $(\, i \, [\, i \,] \,)$ is a prime well-matched block of the Motzkin shift while $(\, [\,] \,) \, (\,)$ is a well-matched block which is not prime and $(\,) \, [\, [$ is a block which is not well-matched.

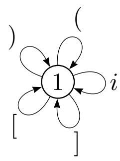

FIGURE 1. The Motzkin shift on the alphabet $A = (A_c, A_r, A_i)$ with $A_c = \{(, [\}$, $A_r = \{),]\}$ and $A_i = \{i\}$. The (-edge is matched with the)-edge and the [-edge is matched with the]-edge.

Example 2. Consider now the sofic-Dyck shift over A presented by the graph shown in Fig. 2 and with the following matched edges. The (-edge is matched with the)-edge and the [-edge is matched with the]-edge. This shift is called the *even-Motzkin shift*. The constraint is now stronger than the constraint described by the Dyck-automaton of Example 1. A even number of internal letters i is required between each parenthesis symbol. The even-Motzkin shift is a subshift of the Motzkin shift.

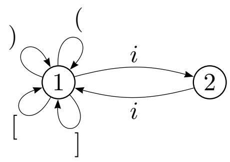

FIGURE 2. The even-Motzkin shift on the alphabet $A = (A_c, A_r, A_i)$ with $A_c = \{(, [\}$, $A_r = \{),]\}$ and $A_i = \{i\}$. The (-edge is matched with the)-edge and the [-edge is matched with the]-edge.

Let a, m be nonnegative integers (m for memory and a for anticipation). A Dyck-automaton $(\mathcal{A}, M)$ is (m, a)-*definite* if whenever two admissible paths $(p_i, a_i, p_{i+1})_{-m \le i \le a}$, $(q_i, a_i, q_{i+1})_{-m \le i \le a}$, of $\mathcal{A}$ of length $m + a$ have the same label, then $p_0 = q_0$. We say that $(\mathcal{A}, M)$ is *definite* (or *local*) if it is (m, a)-local for some nonnegative integers m and a. Note that this property is independent of the set M. It is shown in [8] that it is not necessary to assume that the paths are admissible. Sofic-Dyck shifts presented by definite Dyck automata are called *finite-type-Dyck shifts*. The Motzkin shift of Example 1 is a finite-type-Dyck shift while the even-Dyck shift of Example 2 is not.

A Dyck-automaton is *right-resolving* (or *deterministic*) if there is at most one edge starting in a given state and with a given label. Sofic shifts (see [33]) have right-resolving presentations. It is no more the case in general for sofic-Dyck shifts although visibly pushdown languages are accepted by deterministic visibly pushdown automata [1].

3. Zeta functions of sofic-Dyck shifts

In this section, we recall the formula of the zeta function of a sofic-Dyck shift obtained in [9]. This formula was established for shifts having a right-resolving Dyck automaton. We extend the formula to all sofic-Dyck shifts.

3.1. Definitions and general formula.

We first define the *multivariate zeta function* of a shift. Denote $\mathbb{Z}\langle\langle A\rangle\rangle$ the set of noncommutative formal power series over $\mathbb{Z}$ on the alphabet A. Each language L of finite words over A defines a series, its characteristic series defined by $\underline{L} = \sum_{u \in L} u$.

Let $\mathbb{Z}[[A]]$ be the usual commutative algebra of formal power series in the variables a in A and $\pi\colon \mathbb{Z}\langle\langle A\rangle\rangle \to \mathbb{Z}[[A]]$ be the natural homomorphism. Let $S \in \mathbb{Z}\langle\langle A\rangle\rangle$ be a noncommutative series. One can write $S = \sum_{n\geq 0}[S]_n$ where each $[S]_n$ is the homogeneous part of S of degree n. The *multivariate zeta function* of S is the commutative series in $\mathbb{Z}[[A]]$ is

$$Z(S) = \exp \sum_{n\geq 1} \frac{[S]_n}{n}.$$

The *(ordinary) zeta function* of a language L is

$$\zeta_L(z) = \exp \sum_{n\geq 1} a_n \frac{z^n}{n},$$

where a_n is the number of words of lengths n of L. Note that $\zeta_L(z) = \theta(Z(\underline{L}))$, where $\theta\colon \mathbb{Z}[[A]] \to \mathbb{Z}[[z]]$ is the homomorphism such that $\theta(a) = z$ for any letter $a \in A$.

Let X be a shift. The set X is invariant by the *shift transformation* σ defined by $\sigma((x_i)_{i\in\mathbb{Z}}) = (x_{i+1})_{i\in\mathbb{Z}}$. Call *periodic pattern* of X a word u such that the bi-infinite concatenation of u belongs to X and denote $P(X)$ the set of periodic patterns of X. These definitions are extended to σ-invariant sets of bi-infinite sequences which may not be shifts (*i.e.* which may not be closed subsets of sequences).

The *multivariate zeta function $Z(X)$ of the shift* (resp. *zeta function $\zeta_X(z)$ of the shift*) X is defined as the multivariate zeta function (resp. zeta function) of its set of periodic patterns. Hence

$$\zeta_X(z) = \exp \sum_{n \geq 1} p_n \frac{z^n}{n},$$

where p_n the number of points of X of period n, *i.e.* of points x such that $\sigma^n(x) = x$.

Let X be a sofic-Dyck shift presented by a Dyck automaton $(\mathcal{A}, M)$ over A, where Q is the set of states.

We denote by X_C the σ-invariant subset of X containing all orbits of points obtained by bi-infinite concatenation of prime Dyck words. We denote by X_{C_c} (resp. X_{C_r}) the σ-invariant subset of X containing all orbits of points obtained by bi-infinite concatenation of words of the form uv, where u is a prime Dyck word and $v \in A_c^*$ (resp. of words of the form vu, where u is a prime word and $v \in A_r^*$). We also denote by X_{Z_c} (resp. X_{Z_r}) the subshifts of X of points included in $A_c^{\mathbb{Z}}$ (resp. in $A_r^{\mathbb{Z}}$).

The following proposition is due to Krieger and Matsumoto [32] for Markov-Dyck shifts. It extends to sofic-Dyck shifts directly [9].

Proposition 1. *The zeta function $\zeta_X(z)$ of a sofic-Dyck shift satisfies*

$$\zeta_X(z) = \frac{\zeta_{\mathsf{X}_{C_c}}(z)\zeta_{\mathsf{X}_{C_r}}(z)\zeta_{\mathsf{X}_{Z_c}}(z)\zeta_{\mathsf{X}_{Z_r}}(z)}{\zeta_{\mathsf{X}_C}(z)}, \tag{1}$$

where X_C, X_{C_c}, X_{C_r}, X_{Z_c}, X_{Z_r} are the σ-invariants subsets of subshifts defined above.

3.2. **Reduced presentations of sofic-Dyck shifts.** In order to compute the zeta functions of Equation (1), we need to work with presentations of a sofic-Dyck shift for which some good properties hold, as for instance deterministic or unambiguous Dyck automata.

A Dyck automaton is said to be *unambiguous* if whenever two *admissible* paths with the same origin, end, and label are equal.

Sofic-Dyck shifts may not have either deterministic or unambiguous presentations in general. We show below that the determinization (or co-determinization) process of visibly pushdown automata [1] can be adapted to Dyck automata and performs a *reduction* of these automata.

Let $(\mathcal{A}, M)$ be a Dyck automaton accepting a sofic-Dyck shift X. Following the subset-based construction of [1] we define the Dyck automaton $(\mathcal{A}, N)$ as follows. The process has some similarity with the determinization of transducers. We refer to [1, Theorem 2] for the intuition of the construction.

If $\mathcal{A} = (Q, E, A)$, we first define the Dyck automaton $\mathcal{T} = (\mathfrak{P}(Q \times Q) \times \mathfrak{P}(Q), F, B)$ where $B = (A_c \times (\mathfrak{P}(Q \times Q) \times \mathfrak{P}(Q) \times A_c), A_r \times ((\mathfrak{P}(Q \times Q) \times \mathfrak{P}(Q) \times A_r) \cup \{\perp\}), A_i$, and $\mathfrak{P}(S)$ is the set of subsets of S for a set S.

A set S called the summary is associated to each state p. If p can be reached after reading aw, where a is a call letter and w a well-matched word, the summary of p contains pairs of states (q, q') such that there is an admissible path from q to q' in $(\mathcal{A}, M)$ labelled by w.

Let $\mathrm{Diag}(Q)$ denote the set of all pairs (p, p) for $p \in Q$.

The edges of $\mathcal{T}$ are defined as follows.

- For every $\ell \in A_i$, $((S, R), \ell, (S', R')) \in F$ if $S' = \{(p, p') \mid \exists\, q \in Q\colon (p, q) \in S, (q, \ell, p') \in E\}$ and $R' = \{p' \mid \exists\, p \in P\colon (p, \ell, p') \in E\}$.
- For every $a \in A_c$, $((S, R), a|(S, R, a), (\mathrm{Diag}(Q), R')) \in F$ if $R' = \{p' \mid \exists\, p \in R : (p, a, p') \in E\}$.
- For every $b \in A_r$, $((S, R), b|(S", R", a), (S', R')) \in F$ if $S' = \{(p, p') \mid \exists\, p_3\colon (p, p_3) \in S", (p_3, p') \in \mathrm{Update}\}$ is non empty and $R' = \{p' \mid \exists p \in R" : (p, p') \in \mathrm{Update}\}$, where Update is $\{(p, p') \mid \exists p_1, p_2\colon (p, a, p_1) \in E, (p_1, p_2) \in S, (p_2, b, p') \in E, ((p, a, p_1), (p_2, b, p')) \in M$.
- For every $b \in A_r$, $((S, R), b|\bot, (\emptyset, R')) \in F$ if (S', R') are as in the previous item and S' is empty.

The Dyck automaton $(\mathcal{B}, N)$ is obtained as follows. The automaton $\mathcal{B}$ is obtained from $\mathcal{T}$ by removing the second components of the labels of edges of $\mathcal{T}$. Every return edge $((S, R), b, (S', R'))$ such that $((S, R), b|(S", R", a), (S', R')) \in F$ in $\mathcal{T}$ is matched with every call edge labelled by a and starting at $(S", R")$. Note that there is at most one edge with a given label between two given states in $\mathcal{B}$. The Dyck automaton $(\mathcal{B}, N)$ is called the *left reduction* of $(\mathcal{A}, N)$ and is said to be *left reduced*. We similarly define the *right reduction* of $(\mathcal{A}, N)$ with a co-determinization of $(\mathcal{A}, N)$ and an inversion of the roles played by call and return edges.

The determinization process is shown in Fig. 3.

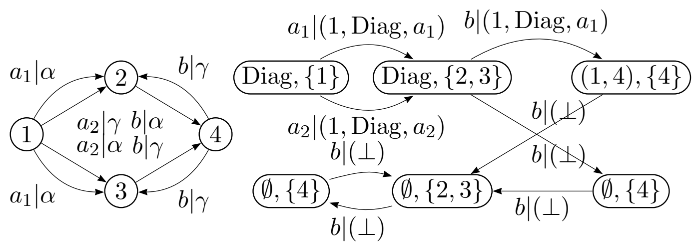

FIGURE 3. The left-reduction of the Dyck automaton $(\mathcal{A}, M)$ over $A = (\{(a\}, \{b\}, A_i)$ pictured on the left. The labels of edges of $(\mathcal{A}, M)$ are the first components of the edges of the picture. Two call and return edges are matched if they share the same second-component label. The Dyck automaton $\mathcal{T}$ is pictured on the right. The left-reduction of $(\mathcal{A}, M)$ is obtained by removing the second components of the labels of $\mathcal{T}$.

A Dyck automaton is said to be *well-matched deterministic* (resp. *L-deterministic*) is any two finite admissible paths labelled by a same well-matched word (resp. a word in a language L) and with the same origin have the same end.

Proposition 2. *A left-reduced Dyck automaton presenting a sofic-Dyck shift X with set of blocks $\mathcal{B}(X)$ is L-deterministic when L is either the code $C(X)$ of prime Dyck words of X, or the set $C(X)(A_c)^* \cap \mathcal{B}(X)$, or the set $(A_c)^* \cap \mathcal{B}(X)$. A right-reduced Dyck automaton presenting a sofic-Dyck shift X with set of blocks $\mathcal{B}(X)$ is L-codeterministic when L is either the set $(A_r)^*C(X) \cap \mathcal{B}(X)$, or $(A_r)^* \cap \mathcal{B}(X)$.*

Proof. Let $(\mathcal{A}, M)$ be a left-reduced Dyck automaton. By construction, $(\mathcal{A}, M)$ is $((A_c)^* \cap \mathcal{B}(X))$-deterministic.

Let us show that $(\mathcal{A}, M)$ is L-deterministic, where L is the set of well-matched words of $\mathcal{B}(X)$. Assume the contrary, let w be a shortest well-matched word for which there are two admissible paths of $(\mathcal{A}, M)$

$$p \xrightarrow{a} p_1 \xrightarrow{u} q_1 \xrightarrow{b} r_1 \xrightarrow{v} s_1,$$
$$p \xrightarrow{a} p_2 \xrightarrow{u} q_2 \xrightarrow{b} r_2 \xrightarrow{v} s_2,$$

such that $a \in A_c$, $b \in A_r$ and u, v are well-matched words.

We first have $p_1 = p_2$. By assumption, we get $q_1 = q_2$. If $r_1 = (S'_1, R'_1) \neq r_2 = (S'_2, R'_2)$, and $q_1 = (S, R)$, there are two edges $((S, R), b|(S"_1, R"_1, a_1), (S'_1, R'_1))$ and $((S, R), b|(S"_2, R"_2, a_2), (S'_2, R'_2)) \in \mathcal{T}$. Since the above paths are admissible, these two edges have to be matched with $(p = (S", R"), a, p_1)$. It follows that $(S"_1, R"_1, a_1) = (S"_2, R"_2, a_2)$ and $r_1 = r_2$. Again by assumption, $s_1 = s_2$.

The fact that $(\mathcal{A}, M)$ is L-deterministic, where L is the set $C(X)(A_c)^* \cap \mathcal{B}(X)$ follows easily. □

If $(\mathcal{A}, M)$ is definite, then its left and right reduction are also definite.

3.3. **Zeta function of finite-type-Dyck shifts.** In the rest of the paper, we will assume that $(\mathcal{A}, M)$ is a left-reduced Dyck automaton and $(\mathcal{A}', M')$ is a right-reduced Dyck automaton accepting a sofic-Dyck shift X.

We define the matrix $C = (C_{pq})_{p,q \in Q}$ in $(Q \times Q)^{\mathbb{Z}\langle\langle A \rangle\rangle}$ where C_{pq} is the characteristic series of the set of prime Dyck words labeling an admissible paths of $(\mathcal{A}, M)$ going from p to q. We denote by M_c the matrix in $(Q \times Q)^{\mathbb{Z}\langle\langle A_c \rangle\rangle}$ where $M_{c,pq}$ is the characteristic series of the set of labels of paths going from p to q and made only of consecutive edges of $\mathcal{A}$ labelled by call letters. We define C' and M_r similarly for $(\mathcal{A}', N')$. We denote by C_c (resp. C_r) the matrix $CM_c{}^*$ (resp. $M_r{}^*C'$). We denote by X_H be the σ-invariant set containing all orbits of points $x \in A^{\mathbb{Z}}$ which are labels of bi-infinite paths of $(p_i, c_i, p_{i+1})_{i \in \mathbb{Z}}$, where $c_i \in H_{p_i p_{i+1}}$. Note that $\mathsf{X}_{M_c} = \mathsf{X}_{Z_c}$ and $\mathsf{X}_{M_r} = \mathsf{X}_{Z_r}$.

By Proposition 2, each matrix $H = C, C_c, M_c$ has $0-1$ coefficients and the product of H by H is *unambiguous*, *i.e.* if for each word $uv \in H_{pq}$, there is a unique state r such that $u \in H_{pr}$ and $v \in H_{rq}$. This is equivalent to the fact that H^k has $0-1$ coefficients for any $k \geq 0$. The same results holds for $H = C', C_r, M_r$.

We denote by H_σ the matrix such that $H_{\sigma,pq} = \sum_{w \in H_{pq}} \sum_{1 \leq j \leq |w|} \sigma^j(w)$, where $\sigma(a_1 \, . \, . \, a_n) = a_2 \, . \, . \, a_n a_1$.

Proposition 3. *Let $(\mathcal{A}, M)$ be a left reduced definite Dyck automaton and H be one of the matrices C, C_c, M_c in $(Q \times Q)^{\mathbb{Z}\langle\langle A\rangle\rangle}$ defined above. We have*

$$\begin{aligned}\pi(\underline{P}_n(\mathsf{X}_H)) &= \mathrm{tr}[\pi H_\sigma (1 - \pi H)^{-1}]_n, \\ &= \mathrm{tr} \sum_{1 \le j \le n} j[\pi(H)]_j[(1 - \pi H)^{-1}]_{n-j}.\end{aligned}$$

where $P_n(\mathsf{X}_H)$ is the set of periodic pattern of X_H of length n.

Proof. Since $(\mathcal{A}, M)$ is definite, we have

$$\underline{P}_n(\mathsf{X}_H) = \sum_{\substack{u \neq \varepsilon, v, p, q \\ uv \in H_{pq}}} [v(\sum_{k \ge 1} H^k)_{qp} u]_n.$$

Thus

$$\begin{aligned}\pi(\underline{P}_n(\mathsf{X}_H)) &= \sum_{\substack{p, q, 1 \le j \le |uv| \\ uv \in H_{pq}}} [uv]_j [\pi H^k)_{qp}]_{n-j}, \\ &= \mathrm{tr}[\pi H_\sigma (1 - \pi H)^{-1}]_n, \\ &= \mathrm{tr} \sum_{1 \le j \le n} j[\pi H]_j[(1 - \pi H))^{-1}]_{n-j}.\end{aligned}$$

Note that the pseudo inverse $(1 - \pi H)$ of πH exists since πH belongs to $(Q \times Q)^{\mathbb{Z}\langle\langle A\rangle\rangle}$. □

Proposition 4. *Let $(\mathcal{A}, M)$ be a left-reduced definite Dyck automaton and H be one of the matrices C, C_c, M_c in $(Q \times Q)^{\mathbb{Z}\langle\langle A\rangle\rangle}$ defined above. We have*

$$\zeta(\mathsf{X}_C)(z) = \frac{1}{\det(I - H(z))},$$

where $H(z) = \theta \pi H$.

Proof. From Proposition 3, we get

$$\begin{aligned}&\sum_{n \ge 1} \frac{\theta \pi \underline{P}_n(\mathsf{X}_H)}{n} \\ &= \sum_{n \ge 1} \frac{1}{n} \mathrm{tr} \sum_{j=1}^{n} j[\theta \pi H]_j [(I - \theta \pi H)^{-1}]_{n-j}, \\ &= \sum_{n \ge 1} \frac{1}{n} \mathrm{tr} \sum_{j=0}^{n-1} (j+1)[\theta \pi H]_{j+1}[(I - \theta \pi H)^{-1}]_{n-j-1}, \\ &= \sum_{n \ge 1} \frac{1}{n} \mathrm{tr} \sum_{j=0}^{n-1} [\mathrm{d}\theta \pi H]_j [(I - \theta \pi H)^{-1}]_{n-j-1}, \\ &= \sum_{n \ge 1} \frac{1}{n} \mathrm{tr}[(\mathrm{d}\theta \pi H)(I - \theta \pi H^{-1})]_{n-1},\end{aligned}$$

$$
\begin{aligned}
&= \sum_{n\geq 1} \frac{1}{n} \operatorname{tr}[(\mathrm{d}\theta\pi H)(I-\theta\pi H^{-1})]_{n-1},\\
&= \sum_{n\geq 1} \operatorname{tr} \frac{1}{n}[(\mathrm{d}\theta\pi H)(I-\theta\pi H^{-1})]_{n-1},\\
&= \sum_{n\geq 1} \operatorname{tr}[-(\log(I-\theta\pi H]_n,\\
&= \operatorname{tr} - \log(I-\theta\pi H).
\end{aligned}
$$

Thus, using Jacobi's formula, we obtain

$$
\begin{aligned}
\zeta(\mathsf{X}_C)(z) &= \exp \operatorname{tr} - \log(I-\theta\pi H),\\
&= \det \exp - \log(I-\theta\pi H),\\
&= \frac{1}{\det(I-\theta\pi H)},
\end{aligned}
$$

where d denotes the derivative with respect to the variable z. □

We have a similar formula with a right reduced definite Dyck automaton and H one of the matrices C', C_c, M_r.

3.4. **ℕ-algebraicity of the zeta function of finite-type-Dyck shifts.** In this section, we prove that the zeta function of a finite-type Dyck shift is ℕ-algebraic, *i.e.* that it is the generating series of an unambiguous context-free language. Actually, we prove a more precise result. The zeta function is the generating series of a visibly pushdown language.

More precisely, a series $S(z)$ is ℕ-*algebraic* if and only if only if if it has its coefficients in ℕ and if $S(z) - S(0)$ is the first component of the solution of a proper positive system, *i.e.* a system of equations of the form $S_i = P_i(z, S_1, \ldots, S_k)$, for some polynomials $P_i(z, x_1, \ldots, x_k)$ having coefficients in ℤ. The system is said to be proper if P_i has no constant term ($P_i(0, ..., 0) = 0$) and does not contain any linear term x_i. It is positive if the coefficients of the P_i are nonnegative.

A series $S(z)$ is ℕ-algebraic if and only if $S(0) \in \mathbb{N}$ and there exists an unambiguous context-free language having generating function $S(z) - S(0)$ (Chomsky-Schützenberger theorem [16]).

There is no known criterion for a ℤ-algebraic series with coefficients in ℕ to be ℕ-algebraic but recent insights from Banderier and Drmota [5, 4] provide some necessary conditions. Banderier and Dmota solved a conjecture on the asymptotic behavior of the coefficients, known to be of the form $\kappa\lambda^n n^\alpha$, which says that the appearing critical exponents α belong to a subset of dyadic numbers. Their result extends the Drmota-Lalley-Woods theorem which assures that $\alpha = -3/2$ as soon as some dependency graph associated to the algebraic system defining the series is strongly connected.

Rational and algebraic formal power series are well-behaved objects with many interesting properties.

Theorem 1. *The zeta function of a finite-type-Dyck shift is* $\mathbb{N}$*-algebraic.*

Proof. From Section 3.3 we get,

$$\zeta_X(z) = \det(I - \theta\pi C_c)^{-1}) \det(I - \theta\pi C_r)^{-1}$$
$$\det(I - \theta\pi M_c)^{-1} \det(I - \theta\pi M_r)^{-1} \det(I - \theta\pi C),$$

where the matrices are defined as in Section 3.3. Thus

$$\begin{aligned}\zeta_X(z) = \det(I - \theta\pi C) \\ \det(I - \theta\pi C {M_c}^*)^{-1} det(I - \theta\pi M_c)^{-1} \\ \det(I - \theta\pi M_r)^{-1} \det(I - \theta\pi {M_r}^* C')^{-1}.\end{aligned}$$

Since

$$\begin{aligned}(I - C{M_c}^*)(I - M_c) &= (I - C\frac{1}{I - M_c})(I - M_c) \\ &= (I - C - M_c),\end{aligned}$$

and similarly $(I - M_r)(I - {M_r}^* C') = (I - M_r - C')$, we get that

$$\begin{aligned}\zeta_X(z) &= \det(I - \theta\pi C) \det(\theta\pi(I - C - M_r)(I - C' - M_r))^{-1}, \\ &= \det(\theta\pi(I - C^* M_c)(I - M_r - C'))^{-1}, \\ &= \det(I - \theta\pi C^* M_c)^{-1} \det(I - \theta\pi(M_r + C'))^{-1}.\end{aligned}$$

Now we use the classical formula for computing $\det(I - N)$ for an $R \times R$-matrix N with noncommutative entries (see for instance [19, Section 7.4]). The matrix defines a labelled graph $\mathcal{G}$ with R as set of vertices and (p, N_{pq}, q) as edges. Set $R = \{1, 2, .., |R|\}$ and denote by $Y_{N,i}$ the set of labels of first-return paths from i to i in $\mathcal{G}$ after removing the lines and columns $1, 2, .. i-1$ of the matrix N. We have

$$\det(I - N) = \prod_{i=1}^{|R|} (1 - Y_{N,i}).$$

It follows that, if C (resp. C') is a $Q \times Q$-matrix (resp. a $Q' \times Q'$-matrix),

$$\zeta_X(z) = \prod_{i=1}^{|Q|} {Y_{C^* M_c, i}}^* \prod_{i=1}^{|Q'|} {Y_{M_r + C', i}}^*,$$

which ends the proof. □

3.5. **Zeta function of sofic-Dyck shifts.** In the case of sofic-Dyck shifts, we recall the computation of the zeta function performed in [9] for sofic shifts having a deterministic Dyck automaton. It extends now to all sofic-Dyck shifts. This proves that all sofic-Dyck shifts have a $\mathbb{Z}$-algebraic zeta function.

We first need some machinery similar to the one used to count periodic points of sofic shifts (see for instance [33]).

Let $(\mathcal{A} = (Q, E, A), M)$ be a Dyck automaton. Let $(\mathcal{A}_{\otimes \ell}$ be the labelled graph whose set of states is the set $Q_{\otimes \ell}$ of all subsets of Q having ℓ elements. We fix an ordering on the states in each element of $Q_{\otimes \ell}$. Let $P = (p_1, \ldots, p_\ell)$, $P' = (p'_1, \ldots, p'_\ell)$ be two states of $Q_{\otimes \ell}$ with $p_1 < \cdots < p_\ell$ and $p'_1 < \cdots < p'_\ell$ according to this ordering. There is an edge labelled by a from P to P' in $\mathcal{A}_{\otimes \ell}$ if and only if there are edges labelled by a from p_i to q_i for $1 \leq i \leq \ell$ and $(q_1, \ldots, q_\ell)$ is an even permutation of P'. If the permutation is odd we assign the label $-a$. Otherwise, there is no edge with label $+a$ or a from P to P'. The labelled graph $\mathcal{A}_{\otimes \ell}$ is a Dyck automaton on the alphabet $A' = (A'_c, A'_r, A'_i)$ with $A'_c = A_c \cup \{-a \mid a \in A_c\}$, $A'_r = A_r \cup \{-a \mid a \in A_r\}$, and $A'_i = A_i \cup \{-a \mid a \in A_i\}$.

We define $M_{\otimes \ell}$ as the set of pairs of edges $((p_1, \ldots, p_\ell),\ a,\ (p'_1, \ldots, p'_\ell))$, $(r_1, \ldots, r_\ell), +/-b,\ (r'_1, \ldots, r'_\ell))$ of $\mathcal{A}_{\otimes \ell}$ such that each edge (p_i, a, p'_i) is matched with (r_i, b, r'_i) for $1 \leq i \leq n$. We say that a path of $\mathcal{A}_{\otimes \ell}$ is *admissible* if it is admissible when the signs of the labels are omitted, the sign of the label of a path being the product of the signs of the labels of the edges of the path.

Let $(\mathcal{A}, M)$ (resp. $(\mathcal{A}', M')$) be a left-reduced (resp. a right-reduced) Dyck automaton$X$. In the sequel H denotes one of the matrices C, C_c, M_c in $(Q \times Q)^{\mathbb{Z}\langle\langle A \rangle\rangle}$ defined in Section 3.3. We denote by $H_{\otimes \ell}$ the matrices defined from $(\mathcal{A}_{\otimes \ell}, M_{\otimes \ell})$ similarly. The matrices C', C_r, M_r $(Q' \times Q')^{\mathbb{Z}\langle\langle A \rangle\rangle}$ are defined similarly from a left-reduced Dyck automaton accepting X. We denote by $H'_{\otimes \ell}$ the matrices defined from $(\mathcal{A}'_{\otimes \ell}, M'_{\otimes \ell})$ similarly.

The following result was established in [9] for deterministic presentations. It holds for reduced presentations $(\mathcal{A}, M)$ and $(\mathcal{A}', M')$.

$$\zeta_{\mathsf{X}_H}(z) = \prod_{\ell=1}^{|Q|} \det(I - \theta\pi H_{\otimes \ell}(z))^{(-1)^\ell},$$

where H is either C, C_c or M_c. Similar formula holds for H' is either C', C_r or M_r.

Proposition 5. *The zeta function $\zeta_X(z)$ of a sofic-Dyck shift is equal to*

$$\prod_{\ell=1}^{|Q|} \det(I - \theta\pi C_{c,\otimes \ell}(z))^{(-1)^\ell} \det(I - \theta\pi M_{c,\otimes \ell}(z))^{(-1)^\ell}$$
$$\prod_{\ell=1}^{|Q'|} \det(I - \theta\pi C_{r,\otimes \ell}(z))^{(-1)^\ell} \det(I - \theta\pi M_{r,\otimes \ell}(z))^{(-1)^\ell}$$
$$\prod_{\ell=1}^{|Q|} \det(I - \theta\pi C_{\otimes \ell}(z))^{(-1)^\ell + 1},$$

where $H_{\otimes \ell}(z) = \theta\pi H_{\otimes \ell}$, for $H_{\otimes \ell} = C_{\otimes \ell}$, $C_{c,\otimes \ell}$, $C_{r,\otimes \ell}$, $M_{c,\otimes \ell}$, $M_{r,\otimes \ell}$, defined above.

Example 3. *Let X be the sofic-Dyck shift of Example 2. The Dyck automaton $\mathcal{A}_{\otimes 1}$ is the same as $\mathcal{A}$. The Dyck automaton $\mathcal{A}_{\otimes 2}$ is pictured on the right part of Figure 4. Let us compute the zeta function of $\mathsf{X}_{\mathcal{C}}$ for this*

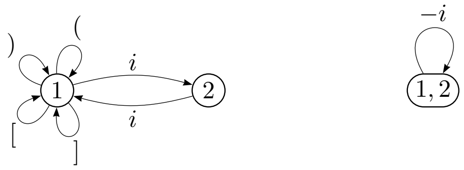

FIGURE 4. The Dyck automaton $(\mathcal{A}, M)$ over $A = (\{(, [\}, \{),]\}, \{i\})$ (on the left) and the Dyck automaton $\mathcal{A}_{\otimes 2}$ on the right.

automaton. We have

$$C = \begin{bmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{bmatrix},\ C_{\otimes 2} = \begin{bmatrix} C_{(1,2),(1,2)} \end{bmatrix}.$$

We have

$$\begin{aligned} C_{11} &= (\ D_{11}\ ) + [\ D_{11}\ ], \\ C_{22} &= 0, \\ C_{12} &= i, \\ C_{21} &= i, \end{aligned}$$

with $D_{11} = (\ D_{11}\ )\ D_{11} + [\ D_{11}\ ]\ D_{11} + iiD_{11} + \varepsilon$. Hence

$$2z^2 D_{11}^2(z) - (1 - z^2)D_{11}(z) + 1 = 0$$

Since $[z^0]D_{11}(z) = 1$, we get

$$D_{11}(z) = \frac{1 - z^2 - \sqrt{1 - 10z^2 + z^4}}{4z^2}.$$

Hence

$$C_{11}(z) = 2z^2 D_{11}(z) = \frac{1 - z^2 - \sqrt{1 - 10z^2 + z^4}}{2}.$$

We have $C_{22}(z) = 0$, $C_{12}(z) = C_{21}(z) = z$. We also have $C_{(1,2),(1,2)} = -i$ and thus $C_{(1,2),(1,2)}(z) = -z$. Thus

$$\begin{aligned} \zeta_{X_C}(z) &= \prod_{\ell=1}^{2} \det(I - \theta\pi C_{\otimes \ell}(z))^{(-1)^\ell} \\ &= (1+z) \begin{vmatrix} 1 - \frac{1-z^2-\sqrt{1-10z^2+z^4}}{2} & -z \\ -z & 1 \end{vmatrix}^{-1}, \\ &= \frac{1+z}{1 - z^2 - \frac{1-z^2-\sqrt{1-10z^2+z^4}}{2}}. \end{aligned}$$

For $H = M_c, M_r$, we have

$$\prod_{\ell=1}^{2} \det(I - \theta\pi H_{\otimes \ell}(z))^{(-1)^\ell} = \frac{1}{1-2z}.$$

We also have

$$C_c = C{M_c}^* = \begin{bmatrix} C_{11} & i \\ i & 0 \end{bmatrix} \begin{bmatrix} \{(,[\}^* & 0 \\ 0 & \varepsilon \end{bmatrix} = \begin{bmatrix} C_{11}\{(,[\}^* & i \\ i\{(,[\}^* & 0 \end{bmatrix},$$
$$C_r = {M_r}^* C = \begin{bmatrix} \{),]\}^* & 0 \\ 0 & \varepsilon \end{bmatrix} \begin{bmatrix} C_{11} & i \\ i & 0 \end{bmatrix} = \begin{bmatrix} \{),]\}^* C_{11} & \{),]\}^* i \\ i & 0 \end{bmatrix}.$$

$$\begin{aligned}\prod_{\ell=1}^{2} \det(I - \theta\pi C_{c,\otimes \ell}(z))^{(-1)^\ell} &= (1+z) \begin{vmatrix} 1 - \frac{C_{11}(z)}{(1-2z)} & -z \\ -\frac{z}{1-2z} & 1 \end{vmatrix}^{-1} \\ &= \frac{(1+z)(1-2z)}{1-2z-z^2-C_{11}(z)}.\end{aligned}$$

The same equality holds for C_r. We finally get

$$\begin{aligned}\zeta_X(z) &= \frac{(1+z)(1-z^2-C_{11}(z))}{(1-2z-z^2-C_{11}(z))^2}, \\ &= \frac{(1+z)}{(1-2z(z^2+C_{11}(z))^*)(1-2z-z^2-C_{11}(z))}, \\ &= (1+z)(2z(z^2+C_{11}(z))^*)^*(2z+z^2+C_{11}(z))^*.\end{aligned}$$

Since $C_{11}(z)$ is the generating series of a visibly pushdown language, $\zeta_X(z)$ is an ℕ-algebraic series. The asymptotic behavior of the coefficients z^n in $\zeta_X(z)$ is $\kappa\lambda^n n^{-3/2}$, where

$$\lambda = \frac{2}{\sqrt{13}-3}$$

This is a typical behavior for an ℕ-algebraic series (see [5, 4] *and* [17]*). The topological entropy of the sofic-Dyck shift is $\log \lambda \sim \log 3.3027$.*

We conjecture that the zeta function of a sofic-Dyck shift is always ℕ-algebraic.

Acknowledgements

The authors would like to thank Wolfgang Krieger for helpful discussions and Cyril Banderier for pointing us some mistakes in a preliminary version of this paper.

This work was supported by the French National Agency (ANR) through "Programme d'Investissements d'Avenir" (Project ACRONYME n°ANR-10-LABX-58) and through the ANR EQINOCS.

References

[1] R. Alur and P. Madhusudan. Visibly pushdown languages. In *Proceedings of the 36th Annual ACM Symposium on Theory of Computing*, pages 202–211 (electronic), New York, 2004. ACM.

[2] R. Alur and P. Madhusudan. Visibly pushdown languages. extended version, 2005.

[3] V. Baladi. Comment compter avec les fonctions zêta. *Gazette des mathématiciens*, 47:79–96, 1991.

[4] C. Banderier and M. Drmota. Coefficients of algebraic functions: formulae and asymptotics. *Combinatorics, Probability and Computing*. Special issue dedicated to Philippe Flajolet, to appear.

[5] C. Banderier and M. Drmota. Coefficients of algebraic functions: formulae and asymptotics. In *DMTCS Proc. AS, 25th International Conference on Formal Power Series and Algebraic Combinatorics (FPSAC 2013)*, pages 1065–1076, 2013.

[6] M.-P. Béal. *Codage symbolique*. Masson, 1993.

[7] M.-P. Béal. Puissance extérieure d'un automate déterministe, application au calcul de la fonction zêta d'un système sofique. *RAIRO Inform. Théor. Appl.*, 29, 1995.

[8] M.-P. Béal, M. Blockelet, and C. Dima. Finite-type-dyck shifts. *CoRR*, http://arxiv.org/1311.4223, 2013.

[9] M.-P. Béal, M. Blockelet, and C. Dima. Sofic-Dyck shifts. *CoRR*, http://arxiv.org/1305.7413, 2013.

[10] J. Berstel and L. Boasson. Balanced grammars and their languages. In *Formal and Natural Computing*, volume 2300 of *Lecture Notes in Computer Science*, pages 3–25. Springer, 2002.

[11] J. Berstel and L. Boasson. Formal properties of xml grammars and languages. *Acta Inf.*, 38(9):649–671, 2002.

[12] J. Berstel and C. Reutenauer. Zeta functions of formal languages. *Trans. Amer. Math. Soc.*, 321:533–546, 1990.

[13] M. Bousquet-Mélou. Rational and algebraic series in combinatorial enumeration. In *International Congress of Mathematicians. Vol. III*, pages 789–826. Eur. Math. Soc., Zürich, 2006.

[14] R. Bowen. Symbolic dynamics. In *On axiom A diffeomorphism*, number 35 in CBMS Reg. Conf. American Mathematical Society, 1978.

[15] R. Bowen and O. Lanford. Zeta functions of restrictions of the shift transformation. In *Proc. Sympos. Pure Math.*, volume 14, pages 43–50. American Mathematical Society, 1970.

[16] N. Chomsky and M.-P. Schützenberger. The algebraic theory of context-free languages. In P. Braffort and D. Hirschberg, editors, *Computer Programming and Formal Systems*, pages 118–161. North Holland, 1963.

[17] P. Flajolet and R. Sedgewick. *Analytic Combinatorics*. Cambridge University Press, Cambridge, 2009.

[18] H. Fujisaki. On embeddings of shifts of finite type into the golden-mean-dyck shift. In *Proceedings of the International Symposium on Information Theory and its Applications, ISITA 2010, 17-20 October 2010, Taichung, Taiwan*, pages 583–588, 2010.

[19] I. M. Gelfand, D. Krob, A. Lascoux, B. Leclerc, V. S. Retakh, and J.-Y. Thibon. Noncommutative symmetric functions. *Adv. Math.*, 112(2):218–348, 1995.

[20] S. Ginsburg and M. A. Harrison. Bracketed context-free languages. *J. Comput. Syst. Sci.*, 1(1):1–23, 1967.

[21] T. Hamachi and W. Krieger. A construction of subshifts and a class of semigroups. *CoRR*, 2013.

[22] T. Hamachi and W. Krieger. On certain subshifts and their associated monoids. *CoRR*, 2013.

[23] K. Inoue. The zeta function, periodic points and entropies of the Motzkin shift. *CoRR*, math/0602100, 2006.

[24] K. Inoue and W. Krieger. Subshifts from sofic shifts and Dyck shifts, zeta functions and topological entropy. *CoRR*, 2010.

[25] N. Kashyap and A. Manada. On the zeta function of a periodic-finite-type shift. In *IEEE International Symposium on Information Theory, ISIT 2009, June 28 - July 3, 2009, Seoul, Korea, Proceedings*, pages 1569–1573, 2009.

[26] G. Keller. Circular codes, loop counting, and zeta-functions. *J. Combin. Theory Ser. A*, 56(1):75–83, 1991.

[27] B. P. Kitchens. *Symbolic Dynamics*. Universitext. Springer-Verlag, Berlin, 1998. One-sided, two-sided and countable state Markov shifts.

[28] D. E. Knuth. A characterization of parenthesis languages. *Information and Control*, 11(3):269–289, 1967.

[29] W. Krieger. On g-functions for subshifts. In *Dynamics & stochastics*, volume 48 of *IMS Lecture Notes Monogr. Ser.*, pages 306–316. Inst. Math. Statist., Beachwood, OH, 2006.

[30] W. Krieger. On subshift presentations. *CoRR*, 2012.

[31] W. Krieger and K. Matsumoto. A notion of synchronization of symbolic dynamics and a class of C^*-algebras. *CoRR*, http://arxiv.org/abs/1105.3249, 2011.

[32] W. Krieger and K. Matsumoto. Zeta functions and topological entropy of the Markov-Dyck shifts. *Münster J. Math.*, 4:171–183, 2011.

[33] D. Lind and B. Marcus. *An Introduction to Symbolic Dynamics and Coding*. Cambridge University Press, Cambridge, 1995.

[34] A. Manning. Axiom A diffeomorphisms have rationnal zeta fonctions. *Bull. London Math. Soc.*, 3:215–220, 1971.

[35] K. Matsumoto. A certain synchronizing property of subshifts and flow equivalence. *CoRR*, http://arxiv.org/abs/1105.3249, 2011.

[36] K. Matsumoto. A notion of synchronization of symbolic dynamics and a class of C^*-algebras. *CoRR*, http://arxiv.org/abs/1105.4393, 2011.

[37] R. McNaughton. Parenthesis grammars. *J. ACM*, 14(3):490–500, 1967.

[38] O. Milenkovic. Constrained coding for context-free languages with applications to genetic sequence modelling. In *2007 IEEE International Symposium on Information Theory, ISIT 2007, Nice, France, June 24-19, 2007*, pages 1686–1690, 2007.

[39] C. Reutenauer. ℕ-rationality of zeta functions. *Adv. in Appl. Math.*, 18(1):1–17, 1997.

[40] A. Salomaa and M. Soittola. *Automata-Theoretic Aspects of Formal Power Series*. Springer-Verlag, New York, 1979.

Université Paris-Est, Laboratoire d'informatique Gaspard-Monge, UMR 8049 CNRS

E-mail address: beal@univ-mlv.fr

Université Paris-Est, Laboratoire d'Algorithmique, Complexité et Logique

E-mail address: michel.blockelet@univ-mlv.fr

Université Paris-Est, Laboratoire d'Algorithmique, Complexité et Logique

E-mail address: catalin.dima@u-pec.fr