Linear Complexity Words and Surface Laminations

Luis-Miguel Lopez   Philippe Narbel

Abstract

We introduce here a way of describing and generating words and languages with linear subword complexities. The method relies on surface laminations and their codings into languages of words by using graph descriptions (train-tracks). The complexity of these languages is shown to be always linear and computable in terms of these graphs. This result leads to constructions of full semi-groups of substitutions whose limit words have a given linear complexity.

Résumé

Le but de cet article est d'étudier et d'exploiter une représentation géométrique de mots et de langages ayant une complexité linéaire. Cette représentation est basée sur la notion de laminations de surfaces qui sont des ensembles de courbes parallèles pouvant se décrire sous forme de graphes plongés (les "réseaux de chemins de fer"). Lorsque ces graphes sont étiquetés, ils induisent une représentation symbolique – un codage – des courbes qu'ils décrivent. Nous montrons que ce codage est toujours un langage de complexité linéaire dont les coefficients ne dépendent que du nombre de sommets et d'arcs du graphe. En utilisant le fait que certaines laminations sont des points fixes d'automorphismes de surfaces qui eux-mêmes peuvent se traduire en substitutions, il est alors possible de construire explicitement des mots et des langages de complexité linéaire. En particulier, nous exhibons des semi-groupes de substitutions dont les mots limites sont tous d'une complexité linéaire donnée.

1   Introduction

The complexity of an infinite word w is defined as the counting function $\mathbb{N}^* \to \mathbb{N}^*$ giving the number of subwords (factors) of length n occuring in w. This is a mean of estimating how intricate is an infinite word in a more precise way than entropy (see e.g. [29, 6]) which becomes effective only when complexity grows exponentially. The lowest complexities are related to periodicity: for one-way (resp. two-way) infinite words, the complexity is ultimately constant iff the word is ultimately periodic (resp. periodic). If the word is not ultimately periodic (resp. periodic), then its complexity at n is at least $n+1$. The words having that complexity have been studied for a long time [25, 5], mainly under the name of *Sturmian* words. Other linear complexities have also been investigated (see e.g. [4, 32, 9, 1, 18, 21], and the surveys [2, 13, 6]). Complexity can be extended to languages by counting all the subwords occuring in the words of a language.

Surface laminations are specific sets of infinite simple curves running on a surface (see e.g. [36, 10, 23] and definitions below). These can be represented by directed graphs embedded in the surface, often called *train-tracks* [36, 17, 14, 28]). When their edges are labelled, such graphs induce a symbolic coding of the laminations into languages of infinite words [38, 19, 20]. Our first result can then be informally stated as follows (see Theorem 3.2 in the text):

Theorem: *Let $\Gamma = (V, E)$ be a graph embedded into a surface M where V is its set of vertices and E its set of edges. Then the coding language of a lamination represented by Γ has complexity $(|E| - |V|)n + |V|$, $n \in \mathbb{N}^*$, where $|.|$ denotes cardinality.*

From the lamination point-of-view, this improves the polynomial estimation obtained by Weiss in [38]. From the formal language theory point-of-view, this gives an easy way of computing complexity as soon as one knows that a language is the coding of a lamination. Moreover, if every word in the language has the same set of finite subwords, the complexity goes to individual words too. One way of obtaining languages of this kind makes use of generic graphs coming from so-called *systems of curves* [37, 27, 19, 20] in surfaces (see Figures 1 to 3 in the text) which have the property that corresponding laminations are fixed points as sets of explicit automorphisms which in turn are conjugated to substitutions (morphisms) on words [19, 20]. As a result, coding languages with given linear complexity can be produced by iterating substitutions. With respect to this we obtain the following result (see Proposition 4.3 in the text):

Proposition: *For every $\alpha, \beta \in \mathbb{N}^*$ with $\beta \geq \alpha$, there are full semi-groups of substitutions giving words of complexity $\alpha n + \beta$ when iterated to infinity.*

Note that constructions of linear complexity words with substitutions have already been obtained based on different tools: for the $n+1$ Sturmian case (see e.g. [34, 11, 24]), for the $2n+1$ case (see [3, 4, 12]), and more generally for the $\alpha n + 1$ case with $\alpha \in \mathbb{N}^*$ (see [31, 9, 35, 18, 21] and also the discussion at the end of the paper).

Finally, by using *graph moves* (see Section 3.3 in the text), we also show how to transform the complexity functions obtained by applying the above theorem, so

that we mainly get the following result (see Proposition 4.5 in the text):

Proposition: *For every $\alpha, \beta \in \mathbb{N}^*$, there are semi-groups of substitutions giving words of complexity $\alpha n + \beta$, $n \in \mathbb{N}^*$, when iterated to infinity and projected to subalphabets.*

2 Definitions

Let A be a finite alphabet and $w = w_1 w_2...$ (resp. $w = ...w_i w_{i+1}...$) be a one-way (resp. indexed two-way) infinite word over A. For $n > 0$, let $F_w(n) = \{w_i w_{i+1}...w_{i+n-1}, \ i \in \mathbb{N}\}$ (resp. $F_w(n) = \{w_i w_{i+1}...w_{i+n-1}, \ i \in \mathbb{Z}\}$) be the set of subwords (also called factors) of length n occuring in w. The **word complexity function** $p_w : \mathbb{N}^* \to \mathbb{N}^*$ is defined by $p_w(n) = |F_w(n)|$; it is the counting function of the subwords occuring in w. If L is a language of one-way or two-way infinite words over A, the **language complexity function** p_L is the counting function of the subwords occuring in all the words in L, i.e. $p_L(n) = |\bigcup_{w \in L} F_w(n)|$. Clearly, $p_w \le p_L$ for every $w \in L$. Equality holds if a language L is **minimal**, i.e. the set of shifts of each word in L is dense in L for the corresponding classical Cantor metric: in this case all the words in L have the same sets of subwords.

Let M be a closed oriented surface of genus $g \ge 0$, with $m \ge 0$ punctures so that its Euler-Poincaré characteristic is negative. Such surfaces have metrics for which the universal covering is the hyperbolic plane; from now on we shall assume M has been given such a metric. Let $\Gamma = (V, E)$ be a finite directed graph embedded in M with set of vertices V and set of edges E. An **admissible path** in Γ is an indexing map from an interval of $\mathbb{Z}$ towards E, such that the end of an edge is the origin of the following one. It inherits in an obvious way an orientation. It is said to be **closed** if it can be defined by a two-way infinite periodic indexing map. A curve γ in M uniformly homotopic to an admissible path in Γ is said to be **carried by** Γ and is called a **leaf**. Uniform homotopy (i.e. homotopy with uniformly continuous maps) is equivalent to demand two homotopic curves have lifts to M's universal covering at bounded distance from each other. Carrying graphs like Γ are often called *train-tracks*[1] (see e.g. [17, 14, 28]).

A **geodesic lamination** $\mathcal{L}$ of M is a closed set of complete simple geodesics (see e.g. [10]). Up to homotopy there is another equivalent definition based on embedded graphs Γ: A **lamination** is a set $\mathcal{L}$ of pairwise disjoint, pairwise non-uniformly homotopic, simple infinite curves carried by some graph Γ, such that $\mathcal{L}$ is maximal with respect to inclusion. Given Γ and the lamination, we say that the lamination is **maximal rel.** to Γ. In general, a given graph Γ may carry many maximal laminations, and a lamination $\mathcal{L}$ may be maximal rel. to many different graphs. We denote by $\{\mathcal{L}_\Gamma\}$ the set of all the maximal laminations carried by Γ, and by $\{\Gamma_\mathcal{L}\}$

---

[1]We consider here only globally oriented graphs, so that in particular all the carried leaves inherit this orientation. Classical train-tracks need not have such an orientation. Nevertheless, to every train-track one may associate a *locally oriented* graph, i.e. a graph where each edge is oriented only near its vertices. Every such graph has a unique orientations double cover which is a globally oriented graph. This double cover is trivial iff the graph is globally orientable. If not, leaves lift to this cover where their codings record not only the edge visited by the carrying paths, but also the direction in which they visit it; so this just doubles the alphabet.

the set of all the graphs which carry $\mathcal{L}$ in a maximal way. A graph Γ is said to be **recurrent** if for every edge of Γ there is at least one closed admissible path going through it. It is a sufficient condition to ensure that $\{\mathcal{L}_\Gamma\} \neq \emptyset$.

Assuming that the edges of a carrying graph Γ are labelled over an alphabet A, the **coding of an admissible path** of Γ is the word obtained by concatenating the edges labels according to the indexing map. We shall always assume that the labelling is such that one distinct letter is assigned to each edge. An embedded graph Γ is said to be **free** if it does not carry distinct admissible paths which are homotopic on M. A necessary and sufficient condition to be free for a recurrent graph is that no disk or annulus component in $M \setminus \Gamma$ has its boundary made of two admissible paths. So for a free graph, the **coding of a carried leaf** ℓ, denoted by $cod(\ell)$, is defined as the coding of its unique homotopic carrying admissible path. By extension, the **coding of a carried lamination** $\mathcal{L}$ is the language of the codings of all the leaves of $\mathcal{L}$.

3   Linear Complexity and Laminations

3.1   Systems of Curves

A **system of curves** after [37, 27, 19] in a surface M is a pair of sets C and D, each consisting of finitely many pairwise non-homotopic and disjoint oriented simple closed curves, so that no disk component of $M \setminus C \cup D$ has less than three vertices on its boundary (i.e. C **hits** D **efficiently**), such that the orientation given by (c, d) at any point of $c \cap d$, $c \in C$ and $d \in D$, agrees with that of M. A system of curves $C \cup D$ can be considered as a recurrent directed graph Γ embedded in M: its set of vertices V is $C \cap D$ and its set of oriented edges E is the set of segments of $C \cup D$ between consecutive intersection points. In general, one requires [37, 27, 19] that each component of $M \setminus C \cup D$ is a topological disk. Here, we just demand that $C \cup D$ seen as a directed graph Γ must be connected. We also say that a system of curves is **free** if its associated graph Γ is. In the following constructions, if disks occur in $M \setminus C \cup D$ so that the initial system is not free we just puncture them. Generic instances of free systems of curves are shown in Figures 1 to 3.
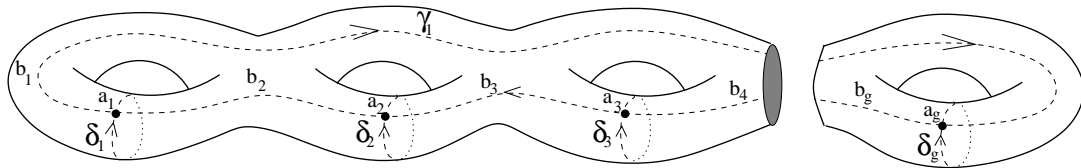

Figure 1: *A system of curves where $C = \{\gamma_1\}$ and $D = \{\delta_1, ..., \delta_g\}$.*

Lemma 3.1. *Let $\Gamma = (V, E)$ be a graph coming from a free system of curves $C \cup D$. Let $\mathcal{L} \in \{\mathcal{L}_\Gamma\}$. Then the coding language of $\mathcal{L}$ has complexity*

$$p_{cod(\mathcal{L})}(n) = |V|n + |V|, \quad n \in \mathbb{N}^*,$$
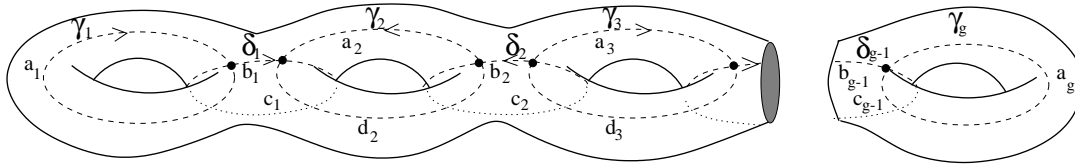
over an alphabet of $2|V| = |E|$ letters.

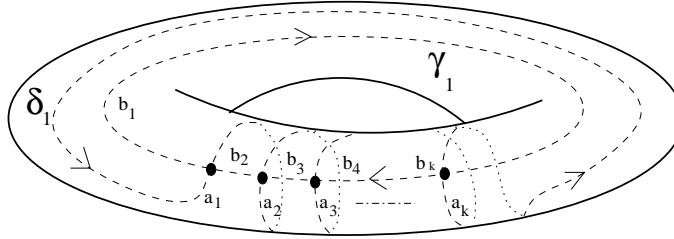Figure 2: *A system of curves where $C = \{\gamma_1, ..., \gamma_g\}$ and $D = \{\delta_1, ..., \delta_{g-1}\}$.*

Figure 3: *A system of curves where $C = \{\gamma_1\}$ and $D = \{\delta_1\}$.*

Proof. For each vertex v of Γ and $n > 0$, let $tree_n(v)$ be the set of subwords occurring in $cod(\mathcal{L})$ with length n and corresponding to carrying paths of $\mathcal{L}$ starting at v . This tree has subtrees of the form $tree_n(v; a)$ which denotes the set of subwords starting by the letter a. Since Γ comes from a system of curves, there are only two edges starting at v, say a and b, so that $tree_n(v) = tree_n(v; a) \cup tree_n(v; b)$. At each v, we put $a < b$ where a is the first outgoing arc following the natural cyclic order around v, called the **horizontal edge**, and b is the second one, called the **vertical edge** (recall that M is assumed oriented). This order relation is extended by lexicographic order, making a totally ordered set of $tree_n(v)$ for all $n > 0$ and $v \in \Gamma$. Since every edge in Γ is assumed to have a distinct label, the set of length-n subwords in $cod(\mathcal{L})$ is equal to $\bigsqcup_{v \in \Gamma} tree_n(v)$.

Let v be a vertex in Γ and v_{hori} (resp. v_{vert}) the origin of the horizontal edge a (resp. of the vertical edge b) ending at v. Let $n > 0$ and let f_a and f_b be the smallest, resp. the greatest length-$n+1$ subword beginning at v_{hori} and v_{vert}. Then the length-n suffixes of f_a and f_b are equal. Indeed, assume it is not the case and let v' be the first vertex from which they differ. This means that one would have a finite carried curve γ from v to v' inserted in between the existing leaves of $\mathcal{L}$. But then γ could be extended in both directions to become a leaf which is not already in $\mathcal{L}$, that is a contradiction with maximality. We call this common length-n suffix $f_a = f_b$ the n-**separating word** of v and denote it by $f_{n,v}$. By construction $f_{n,v}$ is the prefix of $f_{n+1,v}$, and the one-way infinite **separating word** f is the corresponding limit word, i.e. the word whose prefixes are the $f_{n,v}$'s.

Now, still considering the same vertex v, all the subwords of $tree_n(v)$ greater than $f_{n,v}$ can be extended backwards with the letter a to give subwords in $tree_{n+1}(v_{hori}; a)$. All the subwords of $tree_n(v)$ smaller than $f_{n,v}$ can be extended backwards with b to give subwords in $tree_{n+1}(v_{vert}; b)$. By maximality of $\mathcal{L}$, the subword $f_{n,v}$ can be extended with both a and b to give subwords in both of $tree_{n+1}(v_{hori}; a)$ and $tree_{n+1}(v_{vert}; b)$. In other words, $f_{n,v}$ is a *special* factor (see e.g. [9]). It is the only

one starting at v, since no crossings between curves in $\mathcal{L}$ are allowed: no other subword can occur in $cod(\mathcal{L})$. Thus the number of length-$(n+1)$ subwords whose corresponding carrying paths have v as second vertex is equal to one plus the number of length-n subwords starting from v.

Since Γ is recurrent and $\mathcal{L}$ maximal rel. to Γ, every edge is used to carry $\mathcal{L}$. So, using the facts that each edge of Γ is labelled by a distinct letter and that Γ comes from a system of curves, we first have that $p_{cod(\mathcal{L})}(1) = |E| = 2|V|$, i.e. the number of different letters used in $cod(\mathcal{L})$. Next, by the above discussion, $|V|$ new subwords appear each time the length of the subwords is increased by one. Therefore, $p_{cod(\mathcal{L})}(n) = 2|V| + (n-1)|V|$, for $n \in \mathbb{N}^*$, hence the result. $\Diamond$

For example, the graphs coming from systems of curves shown in Figure 1 and 3 lead to laminations with coding languages of complexity $mn + m$, for every $m > 0$ depending respectively on the genus of the surface, and on the number of turns of δ_1 around the meridian of the torus.

3.2   The General Case

An embedded graph Γ in M is said to be **coherent** if all the incoming edges incident to a vertex are adjacent (so all the incident outgoing edges are as well). In other words, let $In(v)$ (respect. $Out(v)$) be the set of the incoming (respect. outgoing) edges at a vertex v, and let $S_U(v)$ (respect. $T_U(v)$) be the convex hull in M containing v and $In(v) \cap U$ (respect. $Out(v) \cap U$), for U a small neighbourhood of v. Then a graph Γ is coherently embedded in M iff for every $v \in V$ there is a U so that either $S_U(v)$ meets $Out(v)$ only at v, or $T_U(v)$ meets $Int(v)$ only at v. Graphs coming from systems of curves $C \cup D$ and classical train-tracks (which are trivalent) are coherent graphs. Since train-tracks suffice to carry all the laminations of a given surface (see e.g. [14]), coherence is not a restrictive assumption from the laminations viewpoint.

Theorem 3.2. *Let* $\Gamma = (V, E)$ *be a free recurrent graph coherently embedded in* M. *Let* $\mathcal{L} \in \{\mathcal{L}_\Gamma\}$. *Then the coding language of* $\mathcal{L}$ *has complexity*

$$p_{cod(\mathcal{L})}(n) = (|E| - |V|)n + |V|, \quad n \in \mathbb{N}^*$$

over an alphabet of $|E|$ *letters.*

Proof.   For each $v \in \Gamma$, we can consider $tree_n(v)$, $n > 0$, like in the proof of Lemma 3.1. Similarly, for every pair of outgoing edges of v, say a and b, we put $a < b$ iff a occurs before b following the natural cyclic order around v (recall that M is oriented). This is well-defined since Γ is coherent, and recurrent so that for every $v \in V$, $In(v) \neq \emptyset$ and $Out(v) \neq \emptyset$. This order can also be extended to $tree_n(v)$ by the lexicographic order. Now, for each adjacent pair of incoming edges of v, there is a separating word. This gives $|In(v)| - 1$ of them (not necessarily distinct, i.e. they are counted with multiplicity). Each induces an increase by one of the number of length-$(n+1)$ subwords, with respect to the number of length-n subwords (for a given length-n subword w starting at some vertex v, the number of distinct extensions is equal to the number plus one of the n-separating words of v which are equal to w). Thus, since again $p_{cod(\mathcal{L})}(1) = |E|$, we have that:

$$p_{cod(\mathcal{L})}(n) = |E| + (n-1) \sum_{v \in V} (|In(v)| - 1).$$

And since $\sum_{v \in V} |In(v)| = |E|$, then $p_{cod(\mathcal{L})}(n) = |E| + (n-1)(|E| - |V|)$, hence the result. $\diamondsuit$

The above result shows that the complexities of the leaves of a lamination rel. to a carrying graph is quite less that the upper polynomial bound given by Weiss in [38] who proved that $p_w(n) \leq cn^r$ where $r = \binom{12g-6}{2} + 2$, and g is the minimum genus of the surface M which embeds Γ such that components of $M \setminus \Gamma$ are made only of disks.

3.3 Graph Moves

We have seen that complexity only depends on the graph Γ, i.e. it is an invariant of $\{\mathcal{L}_\Gamma\}$. But given some $\mathcal{L}$ it is clearly not an invariant of $\{\Gamma_\mathcal{L}\}$. With respect to this, we say that a transformation τ of a graph is a **graph move** iff $\{\mathcal{L}_\Gamma\} \subseteq \{\mathcal{L}_{\tau(\Gamma)}\}$. In other words, a lamination $\mathcal{L} \in \{\mathcal{L}_\Gamma\}$ is still maximal rel. to $\tau(\Gamma)$. So graph moves allow one to navigate in $\{\Gamma_\mathcal{L}\}$, and accordingly to manipulate the complexity of the coding of a lamination. In particular, if coherence is preserved by a graph move τ, Theorem 3.2 can still be applied after having performed τ.

The first transformation we consider, denoted by Mov_1, consists in collapsing one edge e of Γ linking two vertices v_1, v_2, i.e. e is replaced by a unique vertex whose incident edges are the ones of v_1 and v_2 in the same cyclic order (see Figure 4).
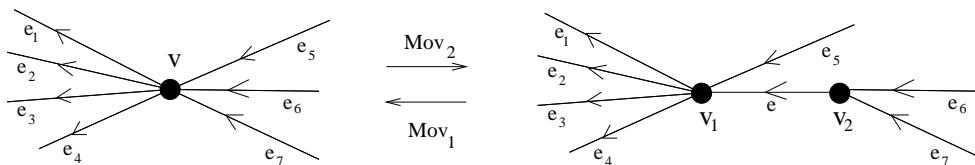

Figure 4: *Generic graph moves.*

Lemma 3.3. *Mov_1 is a graph move on Γ if the two vertices linked by the collapsed edge are distinct in Γ.*

Proof. Let $\mathcal{L} \in \{\mathcal{L}_\Gamma\}$. A vertex of Γ corresponds to a crossway for the leaves of $\mathcal{L}$ which may be contained into a ϵ-disk of M as small as one wants. Collapsing an edge between two distinct vertices amounts in agglomerating two crossways to make another one, which can also be contained in a ϵ-disk: the collapsed edge is just integrated inside the new crossway. So, $\mathcal{L}$ is still carried by $\Gamma' = Mov_1(\Gamma)$. Moreover, if one curve could be added to $\mathcal{L}$ and be carried by Γ', it could clearly be also carried by Γ, a contradiction to maximality. Hence, $\mathcal{L} \in \{\mathcal{L}_{\Gamma'}\}$. $\diamondsuit$

So in case Mov_1 leads to a coherent carrying graph (which is not always the case), the complexity of the coding language is given by Theorem 3.2. For instance, for the graphs shown in Figures 1 and 3, every edge b_i can be collapsed by Mov_1 while preserving coherence. These correspond in fact to a case where one can maximally collapse a graph Γ by a sequence of Mov_1 into a coherently embedded **bouquet of circles**, i.e. a graph with $|V| = 1$ and $|E| = k, k > 0$. The complexities of the coding langages becomes $(k-1)n + 1$ where k equals the number of remaining edges. As a

matter of fact, laminations carried by a coherent bouquet of circles Γ are suspensions of interval exchange transformations (see e.g. [22]) whose permutation is fixed by the embedding of Γ in M. Consistently, complexities $(k-1)n+1$ match the known one computed for the natural symbolic coding of the orbits of irrational interval exchange transformations on k intervals (see e.g. [16, 4, 18]).

There exists an inverse of Mov_1, denoted by Mov_2, which consists in splitting a vertex v of Γ into two vertices v_1 and v_2 linked by a new edge e. Namely Mov_2 at the vertex v is characterized by a partition of the set of edges adjacent to v into two non-empty sets E_1 and E_2 of cyclically consecutive edges such that either E_1 contains all the outcoming edges from v or E_2 contains all the ingoing edges to v. To perform the move itself remove v, link the pending edges of E_i to the new vertices v_i, $i = 1, 2$, and add a new edge e from v_2 to v_1. The condition on E_i insures that the resulting graph can still be embedded in M as a coherent graph. For instance in Figure 4, $E_1 = \{e_1, ..., e_5\}$ and $E_2 = \{e_6, e_7\}$.

Lemma 3.4. *Mov_2 is a graph move on Γ.*

Proof. Reverse the one of Lemma 3.3. $\diamondsuit$

Mov_2 always preserves coherence, so that Theorem 3.2 can always be applied. A useful particular case of Mov_2, called **subdivision**, is when one of the two sets of the partition contains exactly one edge. Subdivision amounts to replacing a single edge by two edges in a row linked by one new vertex. It is a convenient tool to modify complexity.

Remark 3.5. *For every $\alpha, \beta \in \mathbb{N}^*$, there is a graph Γ such that for every $\mathcal{L} \in \{\mathcal{L}_\Gamma\}$, $p_{cod(\mathcal{L})}(n) = \alpha n + \beta$, for $n \in \mathbb{N}^*$.*

Proof. Consider a complexity $\alpha_0 n + \beta_0$ for some $\alpha_0, \beta_0 \in \mathbb{N}^*$. Let Γ be an embedded coherent bouquet of circles with $\alpha_0 + 1$ edges: for instance the one obtained after collapsing the edges b_i with $i \in \{2..g\}$ of a graph as in Figure 1 embedded into a surface of genus $g = \alpha_0$. Next, apply $\beta_0 - 1$ subdivisions to the edges of Γ. $\diamondsuit$

4   Construction of Words and Languages with Linear Complexity

We apply now the above results to effectively construct linear complexity languages and words. The system of curve case has been already discussed in [19, 20] to obtain pseudo-Anosov surface automorphisms together with their fixed laminations. We recall here the method.

A **substitution** θ over an alphabet A is a transformation which sends the letters of A to words over A, and which extends to any word $w = ...w_i w_{i+1}...$ with $w_i \in A$ by sending it to $...\theta(w_i)\theta(w_{i+1})...$. Let $L_\theta = \{\theta^n(s), s \in A, n \in \mathbb{N}\}$ be the language associated to the substitution θ. We then consider the closure rel. to the Cantor metric of the words of L_θ indexed and padded to both infinities (see e.g. [26]). This gives a set of one-way and two-way infinite words. We denote by $Bi(L_{\theta_h})$ any set of representatives of the two-way infinite ones quotiented by the shift operation.

An automorphism h of a surface M sends any embedded graph Γ to another one, via a graph map up to homotopy under certain conditions. If Γ is invariant (up

to homotopy) by h, then h is a conjugate of a group morphism (a Thurston's idea used e.g. in [7]). The results in [19, 20] are based on cases where this conjugacy holds towards free monoid morphisms, i.e. substitutions. More precisely, let Γ be free, labelled over A, and let $\mathcal{C}$ be a set of curves carried by Γ. Then h induces a unique substitution θ_h over A such that $cod(h(\mathcal{C})) = \theta_h(cod(\mathcal{C}))$ (examples are given in the next pages). A **Dehn twist** is a basic automorphisms of a surface M (see e.g. [10, 30]) which is built from a non-null homotopic simple closed curve γ: it consists in cutting M along γ and in applying one (or more) turn(s) to one of the two separated parts before repasting them back[2]. The advantage of working with a system of curves $C \cup D$ is that one can associate a Dehn twist to each curve by putting a copy of $C \cup D$ in **general position** with respect to M's orientation and the original $C \cup D$, either slightly **over** it or slightly **under** (see Figure 5). This makes a set of $|C| + |D|$ Dehn twists denoted by $\mathcal{T}_{C \cup D}$.
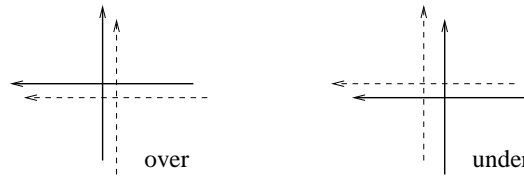

over   under

Figure 5: *The two possible general positions we consider, in a neighborhood of a vertex: twists are in bold style and $C \cup D$ in dashed style.*

It can be seen [37, 27, 19] that every $C \cup D$ considered as a graph Γ is invariant under every automorphism which is a composition of positive twists in $\mathcal{T}_{C \cup D}$. Corresponding substitutions are the compositions of the substitutions conjugated to the individual twists of $\mathcal{T}_{C \cup D}$. In [27, 19], the following semigroup of automorphisms of M has been defined:

$$H^+(C, D) = \{t_{i_1} \circ ... \circ t_{i_n}, t_{i_j} \in \mathcal{T}_{C \cup D}, \text{ where each twist of } \mathcal{T}_{C \cup D} \text{ occurs at least once}\}$$

One of the nice properties about this semi-group is that the corresponding substitutions are all **primitive** [19, 20], i.e. for each such substitution θ there exists a power $K > 0$ such that all the letters of the alphabet A occur at the K-th iterate of θ on any letter. This is the case mainly because systems of curves $C \cup D$ have been assumed connected. The following result has been proved in [20]:

Proposition 4.1. *Let $\Gamma = (V, E)$ come from a free system of curves $C \cup D$ embedded in M. Let $h \in H^+(C, D)$ and let θ_h be its corresponding substitution over A. Then $Bi(L_{\theta_h})$ is the coding language of a maximal lamination rel. to Γ and invariant under h.*

Corollary 4.2. *Let $\Gamma = (V, E)$ come from a free system of curves $C \cup D$ embedded in M. Let $h \in H^+(C, D)$ and let θ_h be its corresponding substitution. Then:*

$$p_{Bi(L_{\theta_h})}(n) = p_w(n) = |V|n + |V| \text{ for all } w \in Bi(L_{\theta_h}).$$

---

[2]More formally, a Dehn twist applies to a parametrized annulus $\{(r, e^{i\alpha}) \mid r \in [r_0, r_1], 0 < r_0 < r_1, \alpha \in [0, 2\pi)\}$ and is homotopic to $f_n : (r, e^{i\alpha}) \to (r, e^{2i\pi n \frac{r-r_0}{r_1-r_0}} e^{i\alpha})$ for some $n \in \mathbb{Z}$. According to n's sign, a twist is said **positive** or **negative**.

Proof.   By applying Lemma 3.1, we have the specified language complexity function. Since θ_h is primitive, $Bi(L_{\theta_h})$ is a minimal language. Thus each word in it has the same complexity. $\diamondsuit$

Note that since θ_h above is a primitive substitution, it must be **prefix-preserving** (resp. **suffix-preserving**) for some power $k > 0$ and some letter $a \in A$, i.e. $\theta_h^k(a) = av$ (resp. $\theta_h^k(a) = va$), where v is a non-empty word over A. By minimality and primitivity of θ_h, the corresponding one-way infinite fixed point $\theta_h^{\omega k}(a)$ has the same complexity as the words in $Bi(L_{\theta_h})$. Now, let us compute the basic substitutions corresponding to the examples shown above:

1. Figure 1 shows a surface M of genus $g \geq 1$ with a system of curves $C \cup D = \Gamma$ with $|C \cup D| = g + 1$, thus yielding $g + 1$ basic substitutions. We fix the general position of $\mathcal{T}_{C \cup D}$ to be under Γ. So for instance, the curve of the twist δ_1 intersects the edge b_2 near its extremity. The effect of one positive turn of δ_1 is therefore to drag the edge b_2 once along a_1. Accordingly it has the same effect on all the curves of a carried lamination. As a result the substitution conjugated to δ_1 – denoted also by δ_1 to save notation – sends b_2 to $b_2 a_1$ while being the identity on the other letters of the alphabet. Applying this argument to all the twists gives us the substitutions conjugated to the elements of $\mathcal{T}_{C \cup D}$, hence of $H^+(C, D)$ too (in the next formulas we only write down the images of the letters not sent to themselves):

$$\delta_1(b_2) = b_2 a_1, \quad \delta_2(b_3) = b_3 a_2, \quad ... \quad \delta_g(b_1) = b_1 a_g,$$
$$\gamma_1(a_1) = a_1 b_1 b_g b_{g-1}...b_2, \gamma_1(a_2) = a_2 b_2 b_1 b_g b_{g-1}...b_3, ..., \gamma_1(a_g) = a_g b_g b_{g-1}...b_1.$$

   According to Corollary 4.2, every composition containing at least one occurrence of each of these basic substitutions leads to a minimal language of complexity $gn + g$. Since it must be prefix-preserving on all letters, one can readily obtain one-way infinite words with the same complexity. Note that when $g = 1$, this corresponds to the Sturmian case.

   The form taken by the substitutions depends on the chosen general position of $\mathcal{T}_{C \cup D}$ rel. to $C \cup D$. If we take the other one indicated in Figure 5, i.e. over Γ, we get suffix-preserving substitutions instead. For instance, the curve of δ_1 intersects this time the edge b_1 only near its origin, and its effect is to drag it along a_1. Thus, the associated substitution sends b_1 to $a_1 b_1$ while being the identity on the other letters of the alphabet. This gives the following set of substitutions associated to $\mathcal{T}_{C \cup D}$:

$$\delta_1(b_1) = a_1 b_1, \quad \delta_2(b_2) = a_2 b_2, \quad ... \quad \delta_g(b_g) = a_g b_g,$$
$$\gamma_1(a_1) = b_1 b_g b_{g-1}...b_2 a_1, \gamma_1(a_2) = b_2 b_1 b_g b_{g-1}...b_3 a_2, ..., \gamma_1(a_g) = b_g b_{g-1}...b_1 a_g.$$

   Note however that for a substitution θ_h with $h \in H^+(C, D)$, the language $Bi(L_{\theta_h})$ does not depend on the general position. A way of proving that is to remark that the fixed stable lamination of a given automorphism of M is unique (see e.g. [10]).

2. Figure 2 shows a surface M of genus $g \geq 2$ with a system of curves $C \cup D = \Gamma$ with $|C \cup D| = 2g - 1$. We fix the general position of $\mathcal{T}_{C \cup D}$ to be under Γ. Figure 6 shows the positions of the twist curves rel. to Γ.
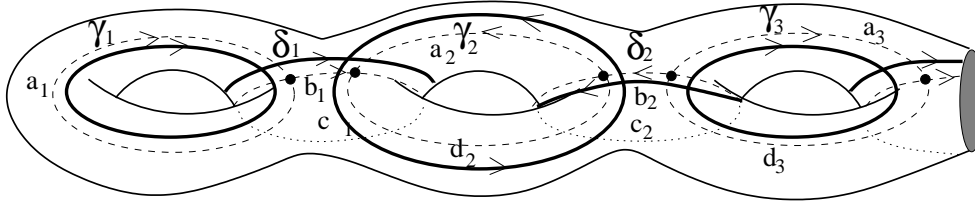

Figure 6: *A zoom at Figure 2 where twists curves in $\mathcal{T}_{C \cup D}$ are indicated in bold style.*

It explains why the associated substitutions depends on the parity of g:

$$\gamma_1(c_1) = c_1 a_1.$$

$$\begin{array}{llll} \gamma_g(b_{g-1}) = & b_{g-1}a_g & \text{if} & g \equiv 0 \mod 2. \\ \gamma_g(c_{g-1}) = & c_{g-1}a_g & \text{if} & g \equiv 1 \mod 2. \end{array}$$

$$\begin{array}{ll} \delta_{g-1}(a_{g-1}) = a_{g-1}b_{g-1}c_{g-1}, & \delta_{g-1}(a_g) = a_g c_{g-1}b_{g-1}, \text{ if } g \equiv 0 \mod 2. \\ \delta_{g-1}(d_{g-1}) = d_{g-1}c_{g-1}b_{g-1}, & \delta_{g-1}(a_g) = a_g b_{g-1}c_{g-1}, \text{ if } g \equiv 1 \mod 2. \end{array}$$

For $i \equiv 0 \mod 2$, ($1 < i < g$ for the γ_i's, $1 \leq i < g - 1$ for the δ_i's):

$$\begin{array}{llll} \gamma_i(b_{i-1}) & = & b_{i-1}d_i a_i, & \gamma_i(b_i) & = & b_i a_i d_i. \\ \delta_i(d_i) & = & d_i c_i b_i, & \delta_i(d_{i+1}) & = & d_{i+1}b_i c_i. \end{array}$$

For $i \equiv 1 \mod 2$, ($1 < i < g$ for the γ_i's, $1 \leq i < g - 1$ for the δ_i's):

$$\begin{array}{llll} \gamma_i(c_{i-1}) & = & c_{i-1}a_i d_i, & \gamma_i(c_i) & = & c_i d_i a_i. \\ \delta_i(a_i) & = & a_i b_i c_i, & \delta_i(a_{i+1}) & = & a_{i+1}c_i b_i. \end{array}$$

So, since the number of vertices is $2(g - 1)$, every composition containing at least one occurence of each of these basic substitutions leads to a minimal language of complexity $2(g - 1)n + 2(g - 1)$.

3. Figure 3 shows a surface M of genus 1 with $k + 1$ punctures needed to make free the system of curves $C \cup D = \Gamma$, with $|C \cup D| = 2$. Fixing again the general position of $\mathcal{T}_{C \cup D}$ under Γ, we obtain the following pair of substitutions:

$$\delta_1(b_1) = b_1 a_1 a_2...a_k, \ \delta_1(b_2) = b_2 a_2 a_3...a_k a_1, \ ..., \ \delta_1(b_k) = b_k a_k a_1...a_{k-1}.$$
$$\gamma_1(a_1) = a_1 b_1 b_k b_{k-1}...b_2, \ \gamma_1(a_2) = a_2 b_2 b_1 b_k...b_3, \ ..., \ \gamma_1(a_k) = a_k b_k b_{k-1}...b_1.$$

Every composition containing at least one occurence of each of these basic substitutions leads to a minimal language of complexity $kn + k$.

Proposition 4.3. *For every $\alpha, \beta \in \mathbb{N}^*$ with $\beta \geq \alpha$, there are semi-groups of substitutions giving words of complexity $\alpha n + \beta$, $n \in \mathbb{N}^*$ when iterated to infinity.*

Proof. Consider a complexity $\alpha_0 n + \beta_0$ for some $\alpha_0, \beta_0 \in \mathbb{N}^*$, with $\beta_0 \geq \alpha_0$. The generic curve system $C \cup D$ in Figure 1 gives languages and words of complexity $gn + g$ where $g \geq 1$ is the genus of the surface. So let $\Gamma = C \cup D$ be the graph such that $g = \alpha_0$, and apply $\beta_0 - \alpha_0$ subdivisions to its edges. This yields a graph Γ' carrying maximal laminations whose coding languages complexity is $\alpha_0 n + \beta_0$. Since the effect of the twists can be restricted to annuli ϵ-neighbourhoods of their curves, each can be taken so that it intersects only one edge. Thus invariance still holds, and associated substitutions can be computed.

To have equality between word and language complexity one must ensure minimality of the language. However subdivisions produce edges which are no more in intersection with the twist curves, so that resulting substitutions are no more primitive. Nevertheless, let θ be a substitution corresponding to an automorphism in $H^+(C, D)$, and θ' be the corresponding substitution on Γ'. The language $Bi(L_{\theta'})$ is equal to $\theta''(Bi(L_\theta))$, where θ'' is the substitution which replaces the label of each edge e in Γ by the label of the edge-path given by the subdivisions applied to e. So, if $Bi(L_\theta)$ is minimal, $\theta''(Bi(L_\theta))$ is also minimal. Hence Corollary 4.2 applies. $\diamondsuit$

For instance let the subdivisions in the above proof be all applied to the edge b_1 of Γ in Figure 1. If $W = d_1...d_k$ is the label of the edge-path made of the k new edges after the application of k subdivisions to b_1, the coding of the initial edge b_1 can be transformed into Wb_1. Fixing the genus g of the surface and the general position to be under $C \cup D$, the substitutions become (see the above example (1)):

$$\delta_1(b_2) = b_2 a_1, \quad \delta_2(b_3) = b_3 a_2, \quad ... \quad \delta_g(b_1) = b_1 a_g,$$

$$\gamma_1(a_1) = a_1 W b_1 b_g b_{g-1}...b_2, \ \gamma_1(a_2) = a_2 b_2 W b_1 b_g b_{g-1}...b_3, \ ...,$$

$$\gamma_1(a_g) = a_g b_g b_{g-1}...b_2 W b_1.$$

Every composition containing at least one occurence of each of these substitutions leads to a minimal language of complexity $gn + (g + k)$.

Remark 4.4. *Systems of curves $C \cup D$ can be enumerated, and so are their associated substitutions.*

Proof. Consider a non-empty set of pairwise disjoint cycles $C_1, C_2, ..., C_k$, in the graph sense, with respective sets of vertices $V_1, V_2, ..., V_k$. Let π be a permutation on $V = \bigcup_{i=1}^k V_i$ and define an oriented edge for each $v \in V$ going from v to $\pi(v)$. The resulting graph Γ is quadrivalent and covered by two sets of pairwise disjoint cycles which makes a system of curves on a surface M. Indeed $C = \bigcup_{i=1}^k C_i$, D comes from the cycles of π (in the permutation sense), and the surface M can be constructed as follows: replace each vertex $v \in V$ of Γ by a disk B^2, and each oriented edge by a rectangle with two marked sides: the side corresponding to the origin of its associated edge, and the opposite side corresponding to the end. Next, glue the rectangles on their corresponding disks along their marked sides according to the incidence relations in Γ, but in such a way that the resulting surface is orientable. This gives a surface M' where C and D are obviously embedded and intersect as required by the definition of a system of curves. Capping off the boundary components of M' with disks yields a closed orientable surface M with possibly some punctures, such that $C \cup D$ as a graph is free and carries laminations whose coding languages have

complexity $|V|n + |V|$. Clearly, every system of curves $C \cup D$ can be built by using this construction. $\diamondsuit$

As a last illustration of the method, we show how to drop the constraint $\beta \geq \alpha$ of Proposition 4.3 by applying Mov_1 graph moves to carrying graphs. Here however, in contrast to Proposition 4.3 where Mov_2 graph moves were applied, the recoding cannot be directly read from the initial substitutions. One must apply one final substitution to obtain the desired language (the process of generation becomes then a so-called *HD0L-system* [33, 9]):

Proposition 4.5. *For every $\alpha, \beta \in \mathbb{N}^*$, there are semi-groups of substitutions giving words of complexity $\alpha n + \beta$, $n \in \mathbb{N}^*$, when iterated to infinity and projected to subalphabets.*

Proof. Consider a complexity $\alpha_0 n + \beta_0$ for some $\alpha_0, \beta_0 \in \mathbb{N}^*$, with $\beta_0 < \alpha_0$ (if $\beta_0 \geq \alpha_0$ apply Proposition 4.3). Consider again the generic curve system $C \cup D$ in Figure 1 and a substitution θ leading to a language $Bi(L_\theta)$ of complexity $\alpha_0 n + \alpha_0$ via Corollary 4.2. Next, apply $\alpha_0 - \beta_0$ times graph moves Mov_1 to $C \cup D$ so that the edges $b_2, ..., b_{\alpha_0 - \beta_0 + 1}$ are collapsed: this gives a coherent graph inducing from $Bi(L_\theta)$ a language of complexity $\alpha_0 n + \beta_0$. With respect to the coding languages, applying a move Mov_1 amounts to get rid of the letter corresponding to the collapsed edge, i.e. to apply a projective substitution. Minimality of $Bi(L_\theta)$ is not impaired by projection on single letters, so that Corollary 4.2 applies too. $\diamondsuit$

For instance in case $g = 3$, the graph Γ of Figure 1 leads to the following substitutions (see the above example (1)):

$$\delta_1(b_2) = b_2 a_1, \quad \delta_2(b_3) = b_3 a_2, \quad \delta_3(b_1) = b_1 a_3,$$

$$\gamma_1(a_1) = a_1 b_1 b_3 b_2, \quad \gamma_1(a_2) = a_2 b_2 b_1 b_3, \quad \gamma_1(a_3) = a_3 b_3 b_2 b_1.$$

Every composition θ containing at least one occurence of each of these basic substitutions leads to a minimal coding language $Bi(L_\theta)$ of complexity $3n + 3$. Getting rid by projection of the occurences of the letter b_2 (resp. both b_2 and b_3) in $Bi(L_\theta)$ leads to a language of complexity $3n + 2$ (resp. $3n + 1$).

5   Discussion

The method presented here gives a generic way of obtaining linear complexity words by using graphs and iterating substitutions. The existence of substitutions heavily relies on the invariance of a system of curves $C \cup D$ under the compositions of the associated twist maps. So the first open question is to ask whether there are other invariant generic graphs. To find out some of them could allow to drop the condition $\beta \geq \alpha$ in Proposition 4.3. Note however that our geometric method seems not adapted to construct words with complexity $\alpha n + \beta$ for the cases where $\beta \leq 0$ (as in [9] where HD0L-systems are mainly used). Note also that from surface automorphisms viewpoint $C \cup D$ systems are not so restrictive, since every automorphism of a surface is known to be generated by compositions of twists belonging to systems of curves (see e.g. [8]).

With respect to the constraint $\beta \geq \alpha$ again, we already noted that words with complexity $\alpha n + 1$ with $\alpha \in \mathbb{N}^*$ generated by iterating substitutions have been studied. As a matter of fact, this can be done by using the idea of *Rauzy induction* [31] on interval exchange transformations. This has been exploited for the particular case of *Rauzy substitutions* (also called *generalized Fibonacci substitutions*) used in [3, 4, 9, 35], and for the general case in [18, 21]. In fact, this can be extended to complexities $\alpha n + \beta$ with $\alpha, \beta \in \mathbb{N}^*$: if one takes a symbolic coding of the orbits of an interval exchange transformation T depending on a finer partition than the one classically given by the initial subintervals, the constant term β can be increased at will. This partition is given by subdividing the initial subintervals by preimages of the discontinuity points of T. However, we do not know at this time how to relate this construction to the invariance of embedded graphs under automorphisms of surfaces.

In Section 3.3, we have quoted a relationship between carrying graphs and interval exchange transformations, in particular when the former are coherent bouquets of circles. In fact, this relationship is much more general since laminations can always be considered as suspensions of interval exchange transformations (see e.g. [15]). This amounts to say that for every lamination $\mathcal{L}$ there is always at least one coherent bouquet of circles in $\{\Gamma_{\mathcal{L}}\}$. With respect to this, a third question is to know whether there is a complete set of local graph moves like Mov_1 and Mov_2 which exhaust every graph in $\{\Gamma_{\mathcal{L}}\}$, so that in particular one could always transform a carrying graph Γ into a coherent bouquet of circles.

Finally, there still remain two general open problems to be solved. The first one is to characterize which linear words can be generated by iterating substitutions. This has been solved for the $n + 1$ complexity case in [11]. The second problem is to characterize which linear complexity languages can be obtained as coding languages of laminations by the method described here.

References

[1] P. Alessandri, *Codage de rotations et basses complexités*, Ph.D. thesis, University Aix-Marseille 2, 1996.

[2] J.-P. Allouche, *Sur la complexité des suites infinies.*, J. Bull. Belg. Math. Soc **1** (1994), no. 2, 133–143.

[3] P. Arnoux, *Un exemple de semi-conjugaison entre un échange d'intervalles et une translation sur le tore*, Bull. Soc. math. France **116** (1988), 489–500.

[4] P. Arnoux and G. Rauzy, *Représentation géométrique de suites de complexité $2n + 1$.*, Bull. Soc. Math. Fr. **119** (1991), 199–215.

[5] J. Berstel and P. Séébold, *Sturmian words*, Combinatorics on words. 2nd ed., Encyclopedia of Mathematics and Its Applications, no. 17, Cambridge University Press, 1997, Lothaire Editor, pp. 30–93.

[6] V. Berthé, *Sequences of low complexity: Automatic and sturmian sequences*, Symbolic dynamics and its applications, Lecture Note Series 279, Cambridge University Press, 2000, F. Blanchard, A. Maas, A. Nogueira, Editors, pp. 1–34.

[7] M. Bestvina and M. Handel, *Train-tracks for surface homeomorphisms.*, Topology **34** (1995), no. 1, 109–140.

[8] J.S. Birman, *Mapping class groups of surfaces*, Braids, AMS, 1988, Contemporary Math., Volume 78, pp. 13–43.

[9] J. Cassaigne, *Complexité et facteurs speciaux*, Bull. Belg. Math. Soc **1** (1997), no. 4, 67–88.

[10] A.J. Casson and S. Bleiler, *Automorphisms of surfaces after Nielsen and Thurston*, Student Text, no. 9, London Mathematical Society, 1988.

[11] D. Crisp, W. Moran, A. Pollington, and P. Shiue, *Substitution invariant cutting sequences*, Journal de Théorie des Nombres de Bordeaux **5** (1993), no. 1, 123–138.

[12] G. Didier, *Echanges de trois intervalles et suites Sturmiennes*, J. de Th. des Nombres de Bordeaux **9** (1997), 463–478.

[13] S. Ferenczi, *Complexity of sequences and dynamical systems*, Discrete Math. **1-3** (1999), 145–154.

[14] A.E. Hatcher, *Measured laminations spaces for surfaces from the topological viewpoint*, Topology and its Applications **30** (1988), 63–88.

[15] A. Kapovich, *Hyperbolic manifolds and discrete groups. lectures on Thurston's hyperbolization*, Birkäuser, 2000.

[16] M. Keane, *Interval exchange transformations*, Math. Zeit. **141** (1975), 25–31.

[17] S.P. Kerckhoff, *Simplicial systems for interval exchange maps and measured foliations*, Ergodic Th. and Dynamical Sys. **5** (1985), 257–271.

[18] ———, *Substitutions and interval exchange transformations of rotation class*, Theoretical Computer Science **1-2** (2001), no. 255, 323–344.

[19] ———, *D0L-systems and surface automorphisms*, Math. Foundations in Computer Science, Lecture Notes in Computer Science 1450, Springer-Verlag, 1998, pp. 522–532.

[20] ———, *Languages, D0L-systems, sets of curves and surface automorphisms*, To appear in Information and Computation, 2001.

[21] ———, *Substitutions from Rauzy induction*, Developments in Language Theory, Foundations, Applications, and Perspectives (G. Rozenberg and W. Thomas, eds.), World Scientific, 2000, pp. 200–210.

[22] R. Mañé, *Ergodic theory and differentiable dynamics*, Springer-Verlag, 1983.

[23] K. Matsuzaki and M. Taniguchi, *Hyperbolic manifolds and Kleinian groups*, Clarendon Press, Oxford, 1998.

[24] F. Mignosi and P. Séébold, *Morphismes Sturmiens et règles de Rauzy*, Journal de Théorie des Nombres de Bordeaux **5** (1993), no. 2, 221–233.

[25] M. Morse and G.A. Hedlund, *Symbolic dynamics II. Sturmian trajectories*, American Journal of Mathematics **62** (1940), 1–42.

[26] Ph. Narbel, *The boundary of iterated morphisms on free semi-groups*, Intern. J. of Algebra and Computation **6** (1996), no. 2, 229–260.

[27] R.C. Penner, *A construction of pseudo-Anosov homeomorphisms*, Transc. of the Amer. Math. Soc. **310** (1988), no. 1, 179–197.

[28] R.C. Penner and J.L. Harer, *Combinatorics of train tracks*, Princeton University Press, 1993, Annals of Math. Studies, Study 125.

[29] M. Quéffelec, *Substitution dynamical systems. spectral analysis*, Lecture Notes in Mathematics, no. 1294, Springer-Verlag, 1987.

[30] J.G. Ratcliffe, *Foundations of hyperbolic manifolds*, Springer Verlag, 1994.

[31] Rauzy, G., *Échanges d'intervalles et transformations induites*, Acta Arith. **34** (1979), 315–328.

[32] G. Rote, *Sequences with subword complexity 2n*, J. of Number Theory **46** (1994), 196–213.

[33] G. Rozenberg and A. Salomaa, *The mathematical theory of L systems*, Academic press, 1980.

[34] P. Séébold, *Fibonacci morphisms and Sturmian words*, Theoretical Computer Science **88** (1991), 365–384.

[35] V. Sirvent, *Relationships between the dynamical systems associated to the Rauzy substitutions*, Theoretical Computer Science **164** (1996), 41–57.

[36] W. Thurston, *The geometry and topology of 3-manifolds*, Chapter 9, Princeton University Lecture Notes, 1978.

[37] William P. Thurston, *On the geometry and dynamics of diffeomorphisms of surfaces*, Bull. Am. Math. Soc., New Ser. **19** (1988), no. 2, 417–431.

[38] H. Weiss, *The geometry of measured geodesic laminations and measured train tracks*, Ergod. Th. and Dynam. Sys. **9** (1989), 587–604.

Luis-Miguel Lopez
TAC College,2229 Iwasaki,
Yoshii-machi, Gunma-ken, 370-2131 Japan.
e-mail: lopez@tacc.ac.jp

Philippe Narbel
LABRI, Université Bordeaux I.
351, Cours de la Libération 33405 Talence, France
e-mail: narbel@labri.u-bordeaux.fr