RAIRO-Theor. Inf. Appl. 44 (2010) 59–77
DOI: 10.1051/ita/2010005 Available online at:
www.rairo-ita.org

HOW TO BUILD BILLIARD WORDS USING DECIMATIONS *

Jean-Pierre Borel[1]

Abstract. We present two methods based on decimation for computing finite billiard words on any finite alphabet. The first method computes finite billiard words by iteration of some transformation on words. The number of iterations is explicitly bounded. The second one gives a direct formula for the billiard words. Some results remain true for infinite standard Sturmian words, but cannot be used for computation as they only are limit results.

Mathematics Subject Classification. 68R15, 68Q68.

Introduction

Let u be the finite word of length $|u| = 24$, on the finite alphabet $\mathcal{A} := \{a, b\}$:

$$u := bbabbbabbbabbabbbabbbabb.$$

Rewrite u, using two stairs and alternatively putting, from the left to the right, the a's down and up at the same position as in u, and so for the b's, as follow:

u	=	b	b	a	b	b	b	a	b	b	b	a	b	b	a	b	b	b	a	b	b	b	a	b	b
u_1	=		b			b		a	b		b			b	a		b			b		b	a		b
u_2	=	b		a	b		b			b		a	b			b		b	a		b			b	

Keywords and phrases. Decimations.

* *Research partially supported by Région Limousin.*

[1] XLim, UMR 6172, Université de Limoges – CNRS, 123 avenue Albert Thomas, 87060 Limoges Cedex, France; borel@unilim.fr

Then $u = u_1.u_2$. This property remains true if we take three stairs:

u	$=$	b	b	a	b	b	b	a	b	b	b	a	b	b	a	b	b	b	a	b	b	b	a	b	b
v_1	$=$				b				b			a	b				b				b		a		b
v_2	$=$		b				b	a			b					b			a	b				b	
v_3	$=$	b		a		b				b				b	a			b				b			

Then we have again $u = v_1 v_2 v_3$.

These properties obviously fails for general words, and we discuss in this paper the algorithmic process corresponding to these transformations. In the example before, it works for two reasons. The main one is that u is a finite billiard word, or a finite cutting sequence, but we also need a technical property, that the numbers of letters are divisible by 2 and by 3, $|u|_a = 6$ and $|u|_b = 18$ respectively.

The paper is organized as follow. In Section 1, we define finite billiard words on finite alphabets, and decimations. The main results are stated in Section 2. They contain both an invariance property which characterizes billiard words, and some iterative process which can be used for computation of billiard words. Then we give a geometrical interpretation and some algebraic properties of this process in Sections 3 and 4 respectively. Section 5 is devoted to complete the proofs.

1. Some definitions

1.1. Digital lines and curves

1.1.1. *Digitization systems*

We consider a *digitization*, or a *pixellization*, of the whole space $\mathbb{R}^k$, *i.e.*, the union $\mathbf{P}$ of identic *pixels* translated by any integer vectors from an original pixel which is a (closed, convex and symmetric) part of the first unit k-cube, *i.e.*, the set of points with all coordinates between 0 and 1. These pixels are pairwise distinct, in the sense that the common part of two of them is included in a $(k-1)$-dimensional space. The classical examples in the plane are the unit squares, or unit spheres, or some others pixels as diamonds (see [11]). In this paper we only consider unit k-cubes.

The *ambiguous points* are the common points of at least three pixels. They correspond in this paper to k-dimensional points with two integer coordinates, at least.

1.1.2. *Coding a curve*

Let $\mathcal{C}$ be any curve in the k-dimensional space $\mathbb{R}^k$, *i.e.*, a continuous application from some interval of $\mathbb{R}$ into $\mathbb{R}^k$. Such a curve can be considered as the trajectory of a point, moving from the "beginning" of the curve up to the "end". In the following, we use the following hypothesis:

Hypothesis 1.1. *The curve $\mathcal{C}$ does not contain any ambiguous point, except maybe for its extremities.*

Consider the set of pixels crossed by this curve, and increasingly ordered by the moving point. Then we define:

- the *pixellized curve* $\mathcal{PC}$ of the curve, which is the union of the pixels crossed by $\mathcal{C}$;
- the *digitized curve* $d\mathcal{C}$, which is the sequence of the segments joining the centers of two consecutive pixels crossed by $\mathcal{C}$.

Such a digital curve can be encoded by a finite or infinite word on some alphabet $\mathcal{A}$, each letter of $\mathcal{A}$ representing the translation from some center to the next one. In dimension 2, we get a finite alphabet with 4 letters, corresponding to the four movements: up, down, left, right. It gives the classical *Freeman code*, see [9], which is also called the chain-code with 4-neighborhoods (see [14] or [8] for another equivalent approach, for two-letter alphabets: the Freeman code can also be obtained using the encoded sequence of the intersections of the curve and the sides of the unit squares, or of the facets in higher dimensions, see [6]). This coding word is called the **P**-code of the curve $\mathcal{C}$.

In a k-dimensional space, we need an alphabet with $2k$ letters, corresponding to "up" and "down" in each direction.

1.2. k-dimensional billiard words

1.2.1. *Finite billiard words*

Let M be a positive integer point in the k-dimensional space $\mathbb{R}^k$, *i.e.*, with positive integer coordinates $(m_1, m_2, \dots m_k)$. The finite billiard word associated to M is the Freeman code of the segment OM, and Hypothesis 1.1 can be written here:

Hypothesis 1.2. *The coordinates $(m_1, m_2, \dots, m_k)$, are pairwise coprime integers.*

This Freeman code is denoted by c_M. It is a finite word of length $m := \sum_{j=1}^{k} (m_j - 1)$ on the finite alphabet $\mathcal{A} := \{a_1, a_2, \dots, a_k\}$, where the letter a_j encodes the vector $\vec{e}_j$ of the canonical basis: as the point moves from O to M, the vector between two consecutive centers is one of the $\vec{e}_j$, always in the positive sense, and we only need k letters instead of $2k$. For simplicity, this alphabet is denoted by $\{a, b\}$, $\{a, b, c\}$ or $\{a, b, c, d\}$ in dimension $2, 3, 4$ respectively.

Notice that this finite word encodes also a billiard trajectory inside a k-dimensional cubic billiard, starting from one vertex and ending at another one: each letter in $\mathcal{A}$ corresponds to the reflexion of this trajectory on a given facet of the cubic billiard.

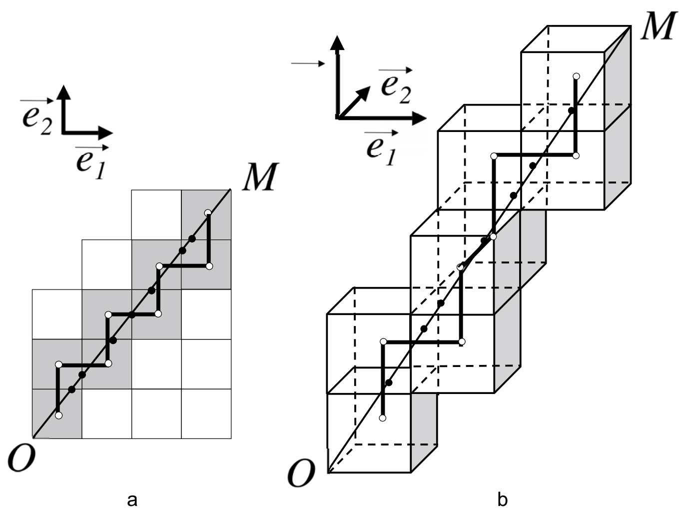

FIGURE 1. Two and three-dimensional finite billiard words.

In Figure 1, we give two examples with $k = 2$, $M(4, 5)$ and $c_M = bababab$ (Fig. 1a) and $k = 3$, $M(3, 2, 5)$ and $c_M = cacbcac$ (Fig. 1b). In these two cases the billiard word is a palindrome, *i.e.*, equal to its reversal. This property is well known, and very easy to prove by symmetry with respect to the midpoint of the segment OM.

1.2.2. *Infinite billiard words*

In the same way, let $\mathcal{D}$ be the half-line of origin O, in the k-dimensional space $\mathbb{R}^k$, whose direction is given by the positive vector $\vec{\alpha} := (\alpha_1, \alpha_2, \ldots, \alpha_k)$. Then we define the associated *standard billiard word* (standard means that $\mathcal{D}$ is starting from O), or *cutting sequence*, denoted by $c_{\vec{\alpha}} = c_{\alpha_1,\alpha_2,\ldots,\alpha_k}$, as the Freeman code of the half-line $\mathcal{D}$. We use the same alphabet $\mathcal{A} = \{a_1, a_2, \ldots, a_k\}$ where letters have the same signification as before. For this particular curve $\mathcal{D}$, Hypothesis 1.1 corresponds to:

Hypothesis 1.3. *All ratios* $\frac{\alpha_i}{\alpha_j}$ *are irrational numbers,* $1 \leq i < j \leq k$.

The same method can be used for any half-line $\mathcal{D}$, *i.e.*, starting from some point S in $\mathbb{R}^k$ and whose direction is given by the positive vector $\vec{\alpha} := (\alpha_1, \alpha_2, \ldots, \alpha_k)$ as before. Then we obtain the corresponding infinite billiard word, which is the Freeman code of $\mathcal{D}$, whenever $\mathcal{D}$ does not contain any point with at least two integer coordinates, except for its origin S. In the following, it is denoted by $c_{\vec{\alpha},S}$.

These words have been intensively studied, see for example [1] or [2] for general expositions.

1.3. Decimations

Decimations consist in periodic cancellation of letters in a word. Let v be a finite (or infinite) word on the finite alphabet $\mathcal{A} := \{a_1, a_2, \ldots, a_k\}$. We define the *position* and the *rank* of a letter a in v, as follow: for each letter a in v, we write $v = v_1 a v_2$, the position of this letter a is $|v_1| + 1$ and the rank is $|v_1|_a + 1$ (recall that the empty word ε is of length $|\varepsilon| = 0$, and that $|v|_a$ is the number of occurrences of the letter a in the word v). Then we define decimations:

- choose two integer values $1 \leq r_j \leq n_j$;
- withdraw the letters a_j in v, except those whose rank is congruent to r_j mod n_j;
- iterate this operation for each index $1 \leq j \leq k$.

Dv is the word consisting of the remaining letters. For a two-letters alphabet $\mathcal{A} = \{a, b\}$, this notion of n-decimation has been introduced by Rauzy [13], and used in [7] for $k = 2$, and then independently by Justin and Pirillo [10] (when $k = 2$ and for cutting sequences) and the author [3] (for $k = 2$ and for Christoffel words), see also [4] and [12] for some generalizations. For $k = 2$ and finite words, we have $C = aVb$, where C is the Christoffel word and V the cutting sequence, and the Christoffel approach is more convenient in this case.

In the following we consider four kinds of decimations.

(1) GD, for *generalized decimations*, defined as before for given integers $(n_1, n_2, \ldots, n_k)$ and $(r_1, r_2, \ldots, r_k)$;
(2) GSD, for *generalized standard decimations*, where $r_j = n_j$ for all j. It is denoted by $D_{n_1, n_2, \ldots, n_k}$;
(3) n-*decimation*, where $n_j = n$ for all j;
(4) n-*standard decimation*, where $n_j = r_j = n$ for all j. This decimation is denoted by D_n in the following.

Then Dv is a finite (resp. infinite) word when v is finite (resp. infinite). In the infinite case, if each letter a_j appears in v with the asymptotic frequency φ_j, then it appears in Dv with the asymptotic frequency:

$$\psi_j = \frac{\frac{\varphi_j}{n_j}}{\sum_{j'=1}^{k} \frac{\varphi_{j'}}{n_{j'}}}$$

which is equal to φ_j for n-decimations.

2. Main results

2.1. Image of billiard words by decimations

Image by decimations of standard billiard words have been already studied, on a two-letter alphabet ($k = 2$). These words are invariant by standard decimations, see [10] or [3] for a Christoffel version of this result. Roughly speaking, they are the only invariant words if we use two standard decimation with coprime values of n [10].

We prove:

Theorem 2.1. *The image of an infinite billiard word by any generalized decimation is a billiard word:*

$$Dc_{\vec{\alpha},S} = c_{\vec{\alpha'},S'}$$

for some α' and S' depending on α, S and D. Moreover, for n-decimations, the two billiard words are parallel, i.e., $\alpha' = \alpha$.

As a corollary, we obtain that the language of finite factors of billiards words with a given direction is invariant under n-decimations, and that the image of a finite billiard word is a finite billiard word.

2.2. Computation of finite standard billiard words

2.2.1. *A new transformation*

We define the *junction word* $w := a_1 a_2 \dots a_k$, and the following transformation on finite word u:

$$T_n(u) := D_n(u^{[w]n}) = D_n((uw)^{n-1}u).$$

Roughly speaking, we consider the n-decimation onto the word u^n, except that we need a junction factor between consecutive factors u. Then:

Proposition 2.1. *Each letter a_j has the same number of occurrences in the two words u and $T_n(u)$.*

Remark that, when $k = 2$, and if $u = V$ is a billiard word, then we have:

$$a.V^{[ba]n}.b = (aVb)^n = C^n$$

where C is the corresponding Christoffel word, and that $a.V^{[ab]n}.b$ can be viewed as a lower Christoffel word. This result immediately implies that u and $T_n(u)$ have the same length.

2.2.2. *Invariance and convergence properties*

We consider the set $\mathcal{A}_M$ of all finite words u on $\mathcal{A}$ such that $|u|_{a_j} = m_j - 1$, for a given fixed integer point $M = (m_1, m_2, \dots, m_k)$ with pairwise coprime coordinates. Thus this set contains the finite billiard word c_M, and is invariant by the transform T_n, using Proposition 2.1. Notice that in this case, at most one letter of each factor w in $u^{[w]n}$ still remains after n-decimation, so that the order of the letters in the junction word w does not effect on the result.

Theorem 2.2.

- *For any $n \geq 2$, the finite billiard word c_M is the only T_n-invariant word in $\mathcal{A}_M$.*

- *For any* $n \geq 2$ *and any* $u \in \mathcal{A}_M$*,* $T_n^r(u) = c_M$ *for large* r*.*
- $r = \left\lceil \frac{\ln(\sum_{j=1}^{k} m_j^2)}{\ln n} \right\rceil$ *is sufficient in the previous item.*

The bound r in Part 3 corresponds to a trivial majoration, it can be improved in some cases. However, it cannot be significatively improved in the general case: when $k = 2$ and $u = a^{p-1}b^{q-1}$ for example, $T_n^r(u)$ begins by the letter a and ends by the letter b as $n^r \leq \min(p, q)$. It is not a palindrome, hence it cannot be equal to the billiard word c_M.

However, this theorem can be used for computing finite standard billiard words on a k-letters alphabet: compute the successive iterates $T_n^r(u)$, and when two successive words are equal, we get c_M using Part 1. This can be used for direct computation, for words in small dimensional spaces, $k = 3$ or $k = 4$ for example.

When the hypothesis of "pairwise coprime coordinates" is not satisfied by M, the theorem still holds, but it must be expressed in a different way, and uses w as a new letter, corresponding to any of the $k!$ words of length k obtained by permutation of the letters of $\mathcal{A}$. The billiard word c_M does not exist, the "fixed" point is still attractive, but it may correspond to some cycles, if we only use the k original letters. We give only two examples of these cases when $k = 2$.

(1) For $M(6, 4)$, *abawaba* is the only fixed point of T_2, it corresponds to the two fixed points *abaababa* and *ababaaba* in the original alphabet $\mathcal{A}$.
(2) For $M(3, 3)$, *ww* is the only fixed point of T_2. In the alphabet $\mathcal{A}$, it corresponds to:
   - two fixed points *abab* and *baba*;
   - a cycle of length two, $\{abba, baab\}$.

   The limit value of $T^n(u)$ depends of the relative position of the trajectory of u and of the integer points $(1, 1)$ and $(2, 2)$ on the segment OM.

The situation is more complicate for $k \geq 3$.

2.2.3. *A concrete approach for* T_n

Let u be in $\mathcal{A}_M$, with odd coordinates m_j. Then $T_2(u)$ can be computed as follow:

(1) dispose the letters of u into two rows, the letters in the upper (resp. lower) row corresponding to those with even (resp. odd) rank;
(2) concatenate the upper word and the lower word.

Then we get the new word $T_2(u)$. For example:

$$u = a\ a\ a\ b\ b\ c\ c\ c\ a\ d\ d\ b\ b\ b\ c\ b\ c\ c\ c\ c\ c$$

$$u_1 = a\ b\ c\ a\ d\ b\ c\ b\ c\ c\ c$$
$$u_2 = a\ a\ b\ c\ c\ d\ b\ b\ c\ c\ c$$

$$T_2(u) = u_1.u_2 = a\ b\ c\ a\ d\ b\ c\ b\ c\ c\ c.\ a\ a\ b\ c\ c\ d\ b\ b\ c\ c\ c$$

By this way, we avoid to use words longer than the initial word u, and obtain the billiard word by a finite number of iterations, following Theorem 2.2. For the example above, the iteration gives successively:

$$\begin{array}{rcl} u & = & aaabbcccaddbbbcbcccccc \\ T_2(u) & = & abcadbcbcccaabccdbbccc \\ T_2^2(u) & = & abccabcdbccabcdbccacbc \\ T_2^3(u) & = & cabcbcdcabcabccdbcabcc \\ T_2^4(u) & = & cbcabcdcabccabcdbcacbc \\ T_2^5(u) & = & cbcacbdcabccbacdbcacbc \\ T_2^6(u) & = & cbcacbdcabccbacdbcacbc \end{array}$$

and the equality $T_2^5(u) = T_2^6(u)$ shows that this word, *i.e.*, *cbcacbdcabccbacdbcacbc*, is the finite billiard word associated to $M(5, 7, 11, 3)$, as $5 = |u|_a + 1$, $7 = |u|_b + 1$, $11 = |u|_c + 1$ and $3 = |u|_d + 1$. Note that we get this word using five iterations, and the upper bound in Part 3 of Theorem 2.2 is $r = \lceil 7,672\ldots \rceil = 8$.

More generally, we get:

Proposition 2.2. *Let u be in $\mathcal{A}_M$, with all coordinates m_j coprime with n. Dispose the letters of u in a rectangular matrix with n rows and m columns, such that the letter a_j in place ℓ and rank r in u is put in column ℓ and line $i \equiv 1 - rm_j^{-1} \pmod n$. Let u_i be the finite word corresponding to row i, read from left to right. Then $T_n(u) = u_1u_2\ldots u_n$.*

This method uses only one copy of the finite word u to compute $T_n(u)$, instead of n copies of u and $n-1$ copies of the junction word w. It uses the property that all letters of the junction words disappear by n-decimation, and that each letter in u remains exactly one time, in one of the n copies of u in $T_n(u)$. This property is false when some m_j is not coprime with n.

2.2.4. *A direct formula*

Denote by $\mathbf{m}$ the product of the m_j's, and by $\tilde{m_j}$ the complement of m_j in this product:

$$\tilde{m_j} := \frac{\mathbf{m}}{m_j} = \prod_{j' \neq j} m_{j'}.$$

Theorem 2.3.

$$c_M = D_{\tilde{m_1}, \tilde{m_2}, \ldots, \tilde{m_k}}(w^{\mathbf{m}-1}).$$

In this result, the factors u disappear, and we only take the junction factors w. However, the word $w^{\mathbf{m}-1}$ is rather long: in the example of Section 2.2.3, its length is equal to 4616, and this formula seems to be useless for an effective computation.

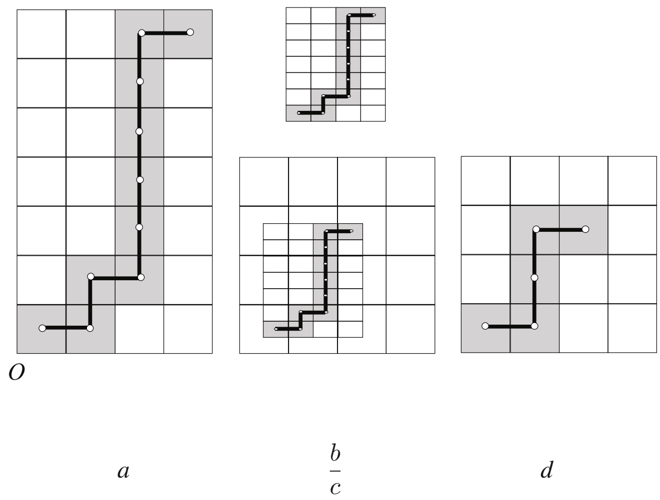

FIGURE 2. From u to $D(u)$: the geometric vision of decimation, $k = 2$, $r_1 = 1$, $n_1 = 2$, $r_2 = 2$, $n_2 = 3$, and $u = ababbbbba$.

3. Geometrical interpretations

3.1. Interpretation of decimations

Decimations have a geometrical interpretation, which is the main ingredient in the proofs in Section 5. We consider a generalized decimation D, associated to (r_j, n_j), $1 \le j \le k$, and a finite or infinite word u on the k-letter alphabet, each letter a_j corresponding to the vector $\vec{e}_j$. Figure 2 corresponds to $k = 2$, $r_1 = 1$, $n_1 = 2$, $r_2 = 2$, $n_2 = 3$, and $u = ababbbbba$.

The decimation operates as follows:

(1) consider the digitized curve (black path) and the pixellized curve (grey squares) associated with u (Fig. 2a);
(2) transform the whole space $\mathbb{R}^k$ by homothecies on coordinates $x_j \mapsto \frac{x_j}{n_j}$, $1 \le j \le k$ (Fig. 2b);
(3) put this new figure in the original grid of pixels, such in a way that the image of the origin O is put at the point whose coordinates are $\frac{n_j - r_j}{n_j}$, $1 \le j \le k$ (Fig. 2c);
(4) $D(u)$ is the Freeman code of the image of the initial digital curve by this transformation (Fig. 2d).

3.2. The transformations T_n

T_n also has a geometrical interpretation, as it is defined using decimations. First note that we use standard n-decimation, so that the origin O remains at the origin at step (3) above. Consider some finite word u in $\mathcal{A}_M$, then we get $T_n(u)$

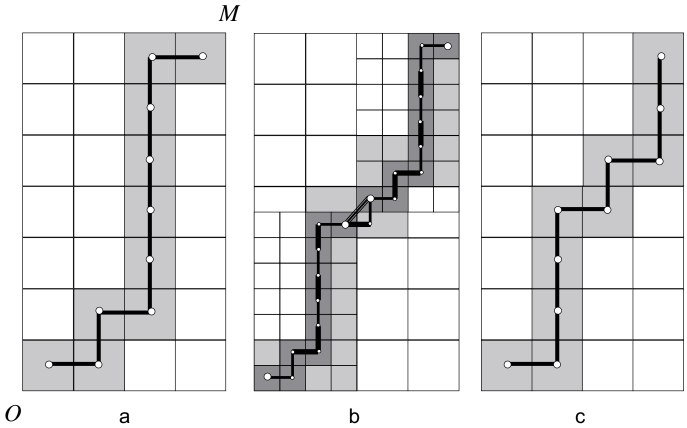

FIGURE 3. The geometrical representation of the transformation $T_2(ababbbbba)$.

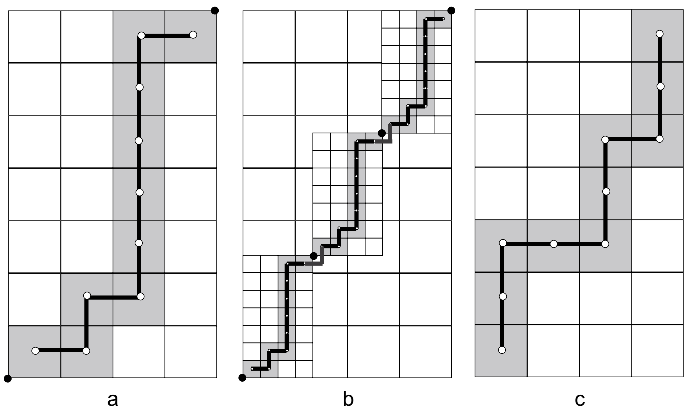

FIGURE 4. The geometrical representation of the transformation $T_3(ababbbbba)$.

by the following geometrical method, which corresponds to Figure 3 for $n = 2$ and $u = ababbbbba$:

(1) consider the digitized curve and the pixellized curve associated with u (Fig. 3a);
(2) make n copies of the rectangle parallelepiped whose main diagonal is OM, and transform them by the homothecy whose ratio is $\frac{1}{n}$: we get n "small" rectangle parallelepipeds, whose main diagonal is denoted by O_iM_i;
(3) put them in the original grid of k-units cubes, in such a way that $O_1 = O$, $O_{i+1} = M_i$, $1 \leq i \leq n-1$. Then $M_n = M$ (Fig. 3b or Fig. 4b), and $u^{[w]n}$ is the Freeman code, in the grid of "small" pixels, of the digital curve in Figure 3b;

(4) $T_n(u)$ is the Freeman code of this curve, in the original grid of pixels (Fig. 3c).

So we have $T_2(ababbbbba) = abbbababb$. Note that the junction word w is necessary to ensure continuity between two consecutive "small" copies of the initial digital curve, and this geometrical interpretation of T_n proves that the image $T_n(u)$ is still in $\mathcal{A}_M$, *i.e.*, Prop. 2.1. A direct algebraic proof is also immediate:

Proof. $|u|_a$ denotes the number of occurrences of the letter a in the word u. Then we have $|w|_a = 1$, and $|u^{[w]n}|_a = n|u|_a + (n-1)$. By n-decimation, the number of remaining letters a is equal to $|u|_a$. □

The word $w = ab$ can be replaced by ba: as the coordinates 4 and 7 are coprime, 2 cannot divide these two numbers, so that the two segments cannot be thick simultaneously. This property remains true in the general case, as the coordinates m_j are pairwise coprime integers.

In Figure 4 we give another illustration of these transformations, with $k = 2$, $n = 3$, $u = ababbbbba$ as before, Then we get

$$u^{[ab]3} = ababbbbba.ab.ababbbbba.ab.ababbbbba$$

and

$$T_3(ababbbbba) = bbaabbabb.$$

4. Algebraic properties

4.1. Composition of decimations

Proposition 4.1. *The set of generalized decimation is a monoid, the set of n-decimations is a submonoid, and the sets of GSD, resp. standard n-decimations, are commutative submonoids.*

Proof. Consider two generalized decimations $D^{(i)}$, $i = 1, 2$, associated with the integers $r_j^{(i)}, n_j^{(i)}$. Then the composition $D^{(2)}D^{(1)}$ is the decimation $D^{(3)}$, corresponding to:

$$\begin{aligned} n_j^{(3)} &= n_j^{(1)} n_j^{(2)} \\ r_j^{(3)} &= r_j^{(1)} + (r_j^{(2)} - 1) n_j^{(1)}. \end{aligned}$$

These formulae prove all the announced results, as $r_j^{(i)} = n_j^{(i)}$, $i = 1, 2$, implies that $r_j^{(3)} = n_j^{(1)} n_j^{(2)} = n_j^{(3)}$. □

Remark that we obtain $D_n D_m = D_{nm}$ as a special case.

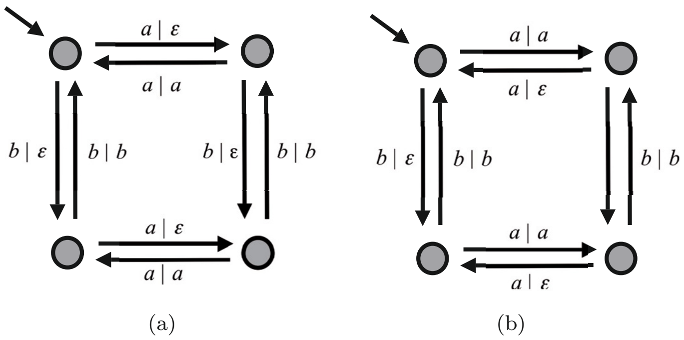

FIGURE 5. Transducers corresponding to decimations, $k = 2$ and $n = 2$, (a) standard 2-decimation; (b) $(1, 2); (2, 2)$-decimation.

4.2. DECIMATIONS AND TRANDUCERS

Any decimation can be realized using a transducer, more precisely a letter-to-letter transducer, with $\prod_{j=1}^{k} n_j$ different states. This transducer copies the configuration of a k-dimensional parallelepiped, whose sides have n_j states. It acts both on finite and infinite words. Figure 5 gives two examples of these transducers in the simplest cases, *i.e.*, $k = n = 2$.

4.3. COMPOSITION OF T-TRANSFORMATIONS

Proposition 4.2. *For any positive n and m, we have $T_m T_n = T_{mn}$.*

Proof. Recall that w is the junction word, *i.e.*, contains one copy of each letter of the alphabet. First, we prove two formulae. The first one is:

$$D_n((uw)^n) = D_n((uw)^{n-1}u)w$$

as $v := (uw)^n$ is such that $|v|_a$ is a multiple of n, hence the last occurrence of a, *i.e.*, the letter a in the last factor w, remains after n-decimation. This is true for each letter a, hence the whole last factor w remains, and it gives the formula. The second formula is true whenever the word v is such that $|v|_a$ is a multiple of n for each letter a. Then we obviously have:

$$D_n(vv') = D_n(v)D_n(v').$$

Then we have:

$$\begin{aligned} T_m(T_n(u)) &= T_m(D_n((uw)^{n-1}u)) \\ &= D_m(D_n((uw)^{n-1}u)wD_n((uw)^{n-1}u)w\dots D_n((uw)^{n-1}u) \\ &\quad \times wD_n((uw)^{n-1}u)) \\ &= D_m(D_n((uw)^n)D_n((uw)^n)\dots D_n((uw)^n)D_n((uw)^{n-1}u)) \\ &= D_m(D_n((uw)^n(uw)^n\dots(uw)^n)(uw)^{n-1}u) \\ &= D_m(D_n((uw)^{mn-1}u)) \\ &= D_{mn}((uw)^{mn-1}u) \\ &= T_{mn}(u) \end{aligned}$$

using the first formula and iteration of the second. □

We use in the following the immediate consequence of Proposition 4.2:

$$(T_n)^r = T_{n^r}.$$

5. Proofs of the theorems

5.1. On the image of billiard words by decimation

We consider a billiard word $c = c_{\vec{\alpha},S}$, such that the half-line $\mathcal{D}$ starting from S and parallel to $\vec{\alpha}$ satisfies Hypothesis 1.1, and a generalized decimation D associated with the numbers (r_j, n_j), $1 \le j \le k$. Denote by $\mathcal{C}$ the corresponding digital curve, then the pixellized curve $\mathcal{D}C$ is the set of pixels crossed by $\mathcal{D}$.

Then we transform the whole space by $\mathcal{T}$ as in Section 3.1 items (2) and (3), *i.e.*, by the relation $x_j \mapsto \frac{x_j+n_j-r_j}{n_j}$. The image of the initial grid of k-unit cubes is transformed in a new grid, whose elements are called the "small pixels". Small pixels are rectangle parallelepiped, whose sides are parallel to the axis and of length $\frac{1}{n_j}$ on the j-th axis (Fig. 6).

Then c is the Freeman code in the grid of small pixels of the half-line $\mathcal{TD}$, which is starting from $S' := \mathcal{T}S$ and parallel to the vector $\vec{\alpha'} := (\frac{\alpha_1}{n_1}, \frac{\alpha_2}{n_2}, \dots, \frac{\alpha_k}{n_k})$. Then S' is an interior point of the first pixel, and the geometrical interpretation of decimations, given in Section 3.1, implies that Dc is the Freeman code of half-line $\mathcal{TD}$.

It proves Theorem 2.1, and we get more precisely:

$$Dc_{\vec{\alpha},S} = c_{\vec{\alpha'},S'}$$

with:

$$\begin{cases} \alpha'_j = \frac{\alpha_j}{n_j} \\ s'_j = \frac{n_j-r_j+s_j}{n_j}. \end{cases}$$

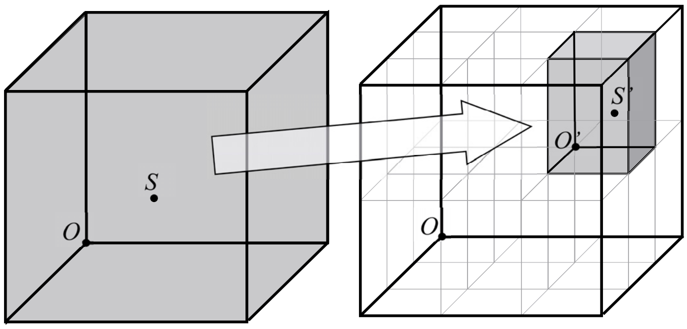

FIGURE 6. Homothetic transformation of the initial pixel, $k = 3$, $n_1 = n_2 = 3$, $r_2 = n_3 = 2$ and $r_1 = r_3 = 1$.

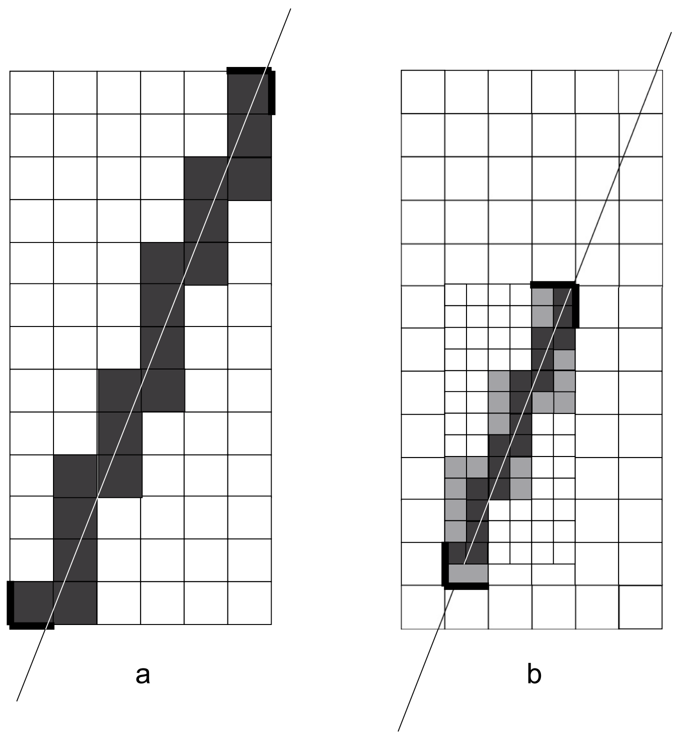

FIGURE 7. The image of a factor of some billiard word by 2-decimation is also a factor of this word.

This result improves some results of [12], Section 4.2. This theorem has easy consequences.

Corollary 5.1. *Infinite standard billiard words are invariant under standard decimations.*

Corollary 5.2. *The language of billiard words parallel to $\vec{\alpha}$ is invariant under n-decimations.*

Corollary 5.3. *The image by n-decimation of a finite billiard word is a prefix factor of an infinite standard billiard word.*

Proof. The first corollary is immediate, as $S = O$ implies $S' = O$ for a n-decimation. □

In general case, two infinite billiard words with same direction have the same language, and it implies Corollary 5.2. However, this property is false when there exists some linear relation over $\mathbb{Z}$ between the coefficients α_j, see [5]. In that case, we use the following general direct proof.

Let u be a factor of an infinite billiard word parallel to $\vec{\alpha}$. It implies that the pixellized curve whose Freeman code is u contains a line parallel to $\vec{\alpha}$ which enters the pixellized curve at the *left* and leaves at the *right* (Fig. 7a): the left (resp. right) part of the first pixel is the set of points in this pixel with at least one coordinate equal to 0 (resp. 1), we define the left part and the right part of any pixel by translation, and the left (resp. right) part of a finite pixellized curve is the left part of the first (resp. last) pixel of the curve (the black segments in Figs. 7a, 7b). Use now the geometric interpretation of n-decimation, then u is also the Freeman code of the black small pixelled curve (Fig. 7b, where $n = 2$), and $D_n(u)$ is the Freeman code of the grey pixelled curve. But a line parallel to $\vec{\alpha}$ enters this pixelled curve at the left and leaves at the right, so that $D_n(u)$ is a factor of an infinite billiard word parallel to $\vec{\alpha}$. This gives Corollary 5.2.

As a finite billiard word is a prefix factor of some infinite standard billiard word (take $\mathcal{D}$ close to M), we get Corollary 5.3.

5.2. Computation of billiard word by iteration

The main idea is that the curve whose Freeman code is $T_n(u)$ is very close to the main diagonal OM, for any word u in $\mathcal{A}_M$, and for large n. When this segment does not contain any ambiguous point, *i.e.*, the coordinates m_j of M are pairwise coprime numbers, then the digital curve and the segment OM have the same Freeman code, *i.e.*, $T_n(u) = c_M$.

We use "small grid" for the grid of k-cubes whose sides have length $\frac{1}{n}$.

5.2.1. *Preliminary lemmas*

Lemma 5.1. *The trajectory $\mathcal{T}$ corresponding in the small grid to the word $u^{[w]n}$ is such that the distance between any point $N \in \mathcal{T}$ and the segment OM is less than $\frac{1}{n}\sqrt{\sum_{j=1}^{k} m_j^2}$.*

Proof. Let C be the distance between OM and the farthest point of the parallelepiped $[OM]$. Then all the points in the small parallelepipeds are closer than $\frac{C}{n}$ from the main diagonal of the corresponding small parallelepiped, thus from OM. This is also true for the parts of trajectory corresponding to the junctions words.The lemma comes from $C^2 \leq \sum_{j=1}^{k} m_j^2$, this majoration is rather bad but is sufficient for the following. □

Lemma 5.2. *Consider a trajectory $\mathcal{T}$ starting from O and going to M such that each point N of the trajectory is such that the distance between N and the segment OM is less than $\frac{1}{\sqrt{\sum_{j=1}^{k} m_j^2}}$. Then the k-unit cubes crossed by $\mathcal{T}$ are exactly those crossed by the segment OM.*

Proof. For any point N with integer coordinates inside the rectangle parallelepiped $[OM]$, the distance between N and the segment OM is equal to:

$$\sqrt{\frac{\left(\sum_{j=1}^{k} m_j^2\right)\left(\sum_{j=1}^{k} n_j^2\right) - \left(\sum_{j=1}^{k} m_j n_j\right)^2}{\sum_{j=1}^{k} m_j^2}}$$

and the numerator inside the square root is a non-zero integer, except for N on the segment, *i.e.*, for $N = O$ or M using Hypothesis 1.2. Hence there is no point with integer coordinates inside the cylinder of axis OM and radius $\frac{1}{\sqrt{\sum_{j=1}^{k} m_j^2}}$, and any trajectory $\mathcal{T}$ inside this cylinder crosses the same set of k-unit squares. □

5.2.2. *End of the proof of Theorem 2.2*

For any finite word $u \in \mathcal{A}_M$, $(T_n)^r(u) = T_{n^r}(u)$ encodes a trajectory in the small grid, whose maximal distance to the diagonal OM is less than $\frac{C}{n^r}$ using Lemma 5.1. For large r:

$$\frac{C}{n^r} < \frac{1}{\sqrt{\sum_{j=1}^{k} m_j^2}}$$

so that $(T_n)^r(u) = T_{n^r}(u) = c_M$ by Lemma 5.2.

Replacing C by its majoration gives Part 3 of the theorem. □

When the initial word u is closed to the segment OM, then the constant C in Lemma 5.1 can be improved, and in the best cases we can choose $r = \left\lceil \frac{\mathrm{cst}}{\ln n} \right\rceil$.

5.3. On Theorem 2.3

5.3.1. *The "naive" computation of a finite standard billiard word*

Let M be a point whose coordinates m_j are pairwise coprime integers, and consider all the ratios $\frac{ON_i}{OM}$, where the N_i is the increasing sequence of the intersections of the segment $[OM]$ and the facets of the unit k-cubes (the black points in Figs. 1a & 1b). N_i has a unique integer coordinate, whose index is denoted by j and value by p. Then we have:

$$\frac{ON_i}{OM} = \frac{p}{m_j}$$

with $1 \leq p \leq m_j - 1$. Using the second construction of finite billiard word, we get the following method: consider all the rational numbers $\frac{p}{m_j}$, for $1 \leq j \leq k$ and $1 \leq p \leq m_j - 1$. These ratios have distinct values, due to Hypothesis 1.2. Encode each ratio by the corresponding letter a_j and order the letters by increasing values of the ratios. Then we get the finite billiard word.

5.3.2. *An example*

Consider the integer point $M = (3, 2, 7, 5)$ in the 4-dimensional space. The ratios are the following ones:

$$\frac{1}{2}, \frac{1}{3}, \frac{2}{3}, \frac{1}{5}, \frac{2}{5}, \frac{3}{5}, \frac{4}{5}, \frac{1}{7}, \frac{2}{7}, \frac{3}{7}, \frac{4}{7}, \frac{5}{7}, \frac{6}{7}$$

and the increasingly ordered ratios are:

$$\frac{1}{7}, \frac{1}{5}, \frac{2}{7}, \frac{1}{3}, \frac{2}{5}, \frac{3}{7}, \frac{1}{2}, \frac{4}{7}, \frac{3}{5}, \frac{2}{3}, \frac{5}{7}, \frac{4}{5}, \frac{6}{7}.$$

So we get the billiard word encoded by the sequence of denominators 7573572753757 encoded in the original alphabet $\mathcal{A}$ by:

$$cdcadcbcdacdc.$$

5.3.3. *Naive computation comes from decimation*

Now, write all the ratios $\frac{i}{\mathbf{m}}$ in the increasing order, $1 \leq i \leq \mathbf{m} - 1$, with $\mathbf{m} := \prod_{j=1}^{k} m_j$. We associate with each ratio the junction word w, so that the word $u := w^{\mathbf{m}-1}$ is associated with the whole sequence of ratios.

Recall that $\tilde{m_j} := \frac{\mathbf{m}}{m_j} = \prod_{j' \neq j} m_{j'}$, and consider the word $v := D_{\tilde{m_1}, \tilde{m_2}, \ldots, \tilde{m_k}}$ $(w^{\mathbf{m}-1})$. For a given index j, the letters a_j which appear in v come from the i-th factor w in u, when $\tilde{m_j}$ divides i, *i.e.*, $i = \ell \tilde{m_j}$, $1 \leq \ell \leq m_j - 1$. It implies both:

(1) each factor w in u gives at most one letter in v;

(2) a factor w in u gives letter a_j in v when it corresponds to a ratio $\frac{\ell}{m_j}$, $1 \leq \ell \leq m_j - 1$.

According to Section 5.3.1, we have $v = c_M$. It proves Theorem 2.3. □

5.4. Computing T_n using a matrix presentation of letters

Consider the image $v := T_n(u) = D_n((uw)^{n-1}u)$, with $u \in \mathcal{A}_M$. When n is prime to m_j, the letter a_j which is in the i-th junction word w has rank im_j and it disappears under n-decimation. It implies:

- for any factor w, at most one letter a_j remains in v. This implies that the order of the letters in the junction word does not affect the result v;
- when n is prime to all the coordinates m_j, all the letters of all factors w disappears by n-decimation.

In the last case, a letter a_j which remains in v and come from the i-th factor u in $(uw)^{n-1}u$ has rank $r \equiv (1-i)m_j \pmod n$. So we have $i \equiv 1 - rm_j^{-1} \pmod n$, and it gives the construction announced in Proposition 2.2.

5.4.1. *An example*

Take $\mathcal{A} = \{a, b, c\}$, $n = 3$, and $M = (2, 5, 7)$. Consider the word $u := abbbbcccccc$. We write this word on three rows. As in Section 2.2.3, the letters c are put on the rows 3-2-1 periodically, and the letters a and b on the rows 2-3-1 periodically, according to the remainings modulo $n = 3$ of the m_j's, *i.e.*, $(2, 2, 1)$:

1				b				c			c
2	a	b			b		c			c	
3			b			c			c		

Then the image word is obtained by reading the three rows from top to bottom:

$$T_3(abbbbcccccc) = bcc.abbcc.bcc.$$

By iteration, we get $T_3^2(u) = bcc.bcabc.cbc$, $T_3^3(u) = cbc.bcacb.cbc = T_3^4(U)$, which is the corresponding billiard word.

References

[1] J-P. Allouche and J. Shallit, *Automatic sequences: Theory and Applications*. Cambridge University Press (2003).

[2] J. Berstel and P. Séébold, Sturmian words, in *Algebraic combinatorics on words*, edited by M. Lothaire, Cambridge University Press (2002).

[3] J-P. Borel, Opérations sur les mots de Christoffel. *C.R. Acad. Sci. Paris* **325** (1997) 239–242.

[4] J-P. Borel, Image homographique de mots de Christoffel. *Bull. Belg. Math. Soc.* **8** (2001) 239–253.

[5] J-P. Borel, *Complexity of degenerated three dimensional billiard words*, in *DLT 2006*, edited by H. Ibarra and Z. Dang. Lect. Notes Comput. Sci. **4036** (2006) 386–396.

[6] J-P. Borel and C. Reutenauer, Palindromic factors of billiard words. *Theoret. Comput. Sci.* **340** (2005) 334–348.

[7] A.M. Bruckstein, Self-similarity properties of digitized straight lines. *Contemp. Math.* **119** (1991) 1–20.

[8] D. Crisp, W. Moran, A. Pollington and P. Shive, Substitution invariant cutting sequences. *J. Théor. Nombres Bordeaux* **5** (1993) 123–137.

[9] H. Freeman, On the encoding of arbitrary geometric configuration. *IRE Trans. Electron. Comput.* **10** (1961) 260–268.

[10] J. Justin and G. Pirillo, Decimations and Sturmian words. *RAIRO-Theor. Inf. Appl.* **31** (1997) 271–290.

[11] J. Koplowitz, On the performance of Chain Codes for Quantization of Line Drawings. *IEEE Trans Pattern Anal. Machine Intell.* **PAMI-3** (1981) 357–393.

[12] B. Parvaix, *Contribution à l'étude des suites sturmiennes*, Ph.D. Thesis, Univ. Limoges, France (1998).

[13] G. Rauzy, Mots infinis en arithmétique, in *Automata on Infinite Words*, edited by M. Nivat and D. Perrin. Lect. Notes Comput. Sci. **192** (1985).

[14] C. Series, The geometry of Markoff numbers. *Math. Intelligencer* **7** (1985) 20–29.