



Theoretical Computer Science 164 (1996) 165–183

**Theoretical
Computer Science**

Propriétés combinatoires des suites définies par le billard dans les triangles pavants

Pascal Hubert

Institut de Mathématiques de Luminy, 163 avenue de Luminy, Case 930, 13288 Marseille cedex 09, France

Received February 1995; revised October 1995

Communicated by M. Nivat

Résumé

Nous étudions certaines propriétés combinatoires de la dynamique symbolique du billard dans le triangle rectangle isocèle et le “demi-triangle équilatéral”. Dans le cas du triangle rectangle isocèle, nous développons un algorithme de construction des suites obtenues comme codage et nous calculons la complexité de ces suites lorsqu’elles ne sont pas périodiques. Nous donnons des règles de construction des mots infinis qui codent les trajectoires de billard dans le “demi-triangle équilatéral”. Ceci nous a conduit à établir un algorithme de construction du codage d’un point sous une rotation munie d’une partition en intervalles.

Abstract

We study some combinatorial properties of the symbolic dynamics of billiard in the isosceles right angle triangle and in the “half equilateral triangle”. In the case of the isosceles right angle triangle, we develop an algorithm to build the coding sequences and we compute the complexity of these sequences when they are not periodic. Rules for building the infinite words coding the billiard trajectories in an “half equilateral triangle” are also given. Finally, this leads to an algorithm that gives the coding of a point under a rotation with a partition by intervals.

1. Introduction

Le problème de billard le plus simple est celui du billard carré. On peut coder une trajectoire d’un tel billard par une suite infinie m sur l’alphabet $\{0, 1\}$ selon les côtés qu’elle rencontre: les côtés verticaux sont codés par 0 et les côtés horizontaux par 1. On sait, depuis les travaux de Hedlund et Morse [10], que, lorsque la pente de la droite d’incidence est irrationnelle, la complexité du mot m vaut $p(n) = n + 1$, pour tout nombre entier n .

La complexité du codage des trajectoires de billards polygonaux ou polyédraux a été beaucoup étudiée ([2, 3] pour le billard dans le cube, [13] pour les billards polygonaux rationnels). Dans le cas particulier du triangle équilatéral, nous avons donné une caractérisation complète de la dynamique symbolique associée au flot du billard [11]. D'autre part, les seuls polygones qui pavent le plan de façon compatible avec le flot du billard sont le carré, le triangle équilatéral, le triangle rectangle isocèle et le “demi triangle-équilatéral” d'angle aux sommets $(\pi/2, \pi/3, \pi/6)$ [20].

Nous étudions donc ici la dynamique symbolique du billard dans le triangle rectangle isocèle et le “demi-triangle équilatéral”.

Nous donnons un algorithme de construction des mots infinis obtenus comme codage de trajectoires de billard dans un triangle rectangle isocèle et nous calculons la valeur exacte de la complexité de ces suites en fonction de l'angle d'incidence (Théorème 1).

Dans le cas du “demi-triangle équilatéral”, l'obtention d'un algorithme de construction des mots infinis est plus difficile. Pour obtenir cet algorithme, il est nécessaire de pouvoir déterminer explicitement le codage d'un point sous une rotation munie d'une partition en intervalles. Le théorème 2 répond à cette question:

Théorème 2. *Soit α un nombre irrationnel, $\alpha \in [0, 1]$, $x_0, \beta \in]0, 1[$ tels que $\beta \neq 1 - \alpha$. Soit w le mot infini qui code l'itinéraire de x_0 sous \mathcal{R}_α rotation d'angle α pour la partition $[0, \beta[, [\beta, 1[, u$ la suite sturmienne associée à l'itinéraire de x_0 sous la rotation \mathcal{R}_α et v la suite sturmienne associée à l'itinéraire de $x_0 - \beta$ sous la rotation \mathcal{R}_α . Alors, pour tout nombre entier n , le mot w vérifie: $w_{n+1} = w_n + v_n - u_n$. De plus, w_0 s'obtient explicitement en fonction des premiers termes des suites u et v .*

2. Définitions, notations et résultats utiles

Soit K un mot infini sur un alphabet fini \mathcal{A} , le langage d'ordre n de K , noté $L_n(K)$, est l'ensemble des facteurs (ou sous mots) de longueur n de K . La complexité de K est l'application p qui, à un nombre entier n , associe le cardinal de $L_n(K)$. On la note $p(n, K)$ ou plus simplement $p(n)$ lorsque le contexte est clair.

Une suite infinie u sur un alphabet fini à deux lettres est dite sturmienne si $p(n) = n + 1$, pour tout nombre entier n .

Soient α est un nombre irrationnel compris entre 0 et 1, x un point de $[0, 1[$ et \mathcal{R}_α la rotation d'angle α , définissons la suite $u(\alpha, x)$ par:

$$u(\alpha, x)_n = 0 \quad \text{si } \mathcal{R}_\alpha^n(x) \text{ appartient à } [0, 1 - \alpha[$$

$$u(\alpha, x)_n = 1 \quad \text{si } \mathcal{R}_\alpha^n(x) \text{ appartient à } [1 - \alpha, 1[.$$

La suite $u(\alpha, x)$ est une suite sturmienne. Le théorème de Hedlund et Morse [10] affirme que toutes les suites sturmiennes sont construites par ce procédé.

Les suites sturmiennes ont été beaucoup étudiées tant au niveau géométrique ([17] par exemple) que pour leurs propriétés combinatoires (voir, par exemple: [4, 8, 9, 15, 19]). En particulier, il existe divers algorithmes de construction du mot $u(\alpha, x)$ en fonction du point x et de du développement en fractions continues de l'angle α ([16] ou [14] par exemple).

On s'intéresse ici au problème de billard dans un triangle T de sommets ABC . Sur le bord de T les trajectoires suivent la règle suivante: l'angle d'incidence est égal à l'angle de réflexion ([6] ou [5] pour une définition précise du flot du billard). On repère les angles par rapport à une direction donnée, par exemple la droite AB .

Fixons une direction initiale φ . Soit Q un point du bord de T dont la trajectoire de billard de direction initiale φ ne rencontre pas de sommet de T , on code cette trajectoire par la suite des côtés qu'elle rencontre. On obtient une suite infinie K sur un alphabet fini à 3 lettres, dites couleurs, dont nous allons étudier les propriétés.

Définition. Soit M point du bord de T et ψ un angle, la trajectoire dépliée de (M, ψ) est la demi-droite $\mathcal{D}_{(M, \psi)}$ d'origine M qui fait un angle ψ avec AB .

Dire que le triangle T pave le plan de façon compatible avec le flot du billard signifie exactement que l'on peut déplier simultanément toutes les trajectoires de billard ou encore que, dans une direction donnée, le flot du billard peut se représenter comme un flot à pente constante sur un tore invariant. Nous renvoyons à [12] pour plus de détails.

Définitions. On dit qu'un triangle T est pavant s'il pave le plan de façon compatible avec le flot du billard. Un pavage du plan par symétrie par des triangles isométriques à T est un T -pavage et l'union des arêtes d'un T -pavage est une T -triangulation du plan.

On notera Φ_t le flot du billard et Φ l'application de premier retour du flot sur la T -triangulation.

On code les arêtes du T -pavage par symétrie (deux arêtes symétriques par rapport à une même troisième sont codées de la même façon). Par conséquent, le codage des intersections de la trajectoire dépliée de (Q, φ) et de la T -triangulation est égal à la suite K .

3. Billard dans le triangle rectangle isocèle

Soit T un triangle rectangle isocèle, nous donnons, dans ce paragraphe, un algorithme de construction des mots infinis obtenus comme codage de trajectoires de billard dans T ainsi que le calcul exact de la complexité de ces mots infinis.

3.1. Algorithme de construction des mots infinis

3.1.1. Notations

Un domaine fondamental D pour l'action du flot du billard dans un triangle rectangle isocèle T est une union de 8 triangles isométriques à T formant un carré de côté 2. Fixons un angle φ dont la tangente est irrationnelle et supposons que φ est compris entre $\pi/4$ et $\pi/2$, ce qui est toujours possible par symétrie du domaine D .

Les droites correspondant à l'hypoténuse de T seront codées par V , celles correspondant aux côtés de l'angle droit par R ou B selon leur nature (cf Fig. 1).

Soit Q un point appartenant à un côté horizontal du T -pavage isocèle tel que la trajectoire dépliée de Q (notée Δ) ne rencontre pas de sommet de la T -triangulation on cherche à déterminer la suite $K = (k_n)_{n \geq 0}$ de couleurs qui code la trajectoire de billard de Q de direction initiale φ . Pour cela, considérons f l'application de premier retour du flot linéaire de pente $\text{tg}(\varphi)$ sur les côtés horizontaux du domaine D . Utilisons des coordonnées adaptées au problème du codage. Le domaine D est divisé en 4 carrés de côtés 1, on définit la nature d'un de ces carrés selon qu'il intersecte une droite de la T -triangulation de pente 1 ou -1. La nature du carré est égale à 0 s'il intersecte une droite de pente 1, elle est égale à 1 sinon. La première coordonnée j est la nature du carré dans lequel se trouve le point Q , la seconde k est la couleur de l'horizontale qui contient le point Q , la troisième est un nombre réel x compris entre 0 et 1 qui mesure la position de Q sur l'horizontale à laquelle il appartient.

Soit \mathcal{R}_φ la rotation d'angle $1/\text{tg}(\varphi)$. Exprimons les coordonnées (j', k', x') de $f(Q)$ en fonction de (j, k, x) .

Lemme 1. Les coordonnées (j', k', x') vérifient:

$$(j', k', x') = (j + \zeta(x) \bmod 2, \bar{k}, \mathcal{R}_\varphi(x))$$

$$\text{où } \zeta(x) = \begin{cases} 0 & \text{si } x \text{ appartient à } [1 - 1/\text{tg}(\varphi), 1[, \\ 1 & \text{si } x \text{ appartient à } [0, 1 - 1/\text{tg}(\varphi)[. \end{cases}$$

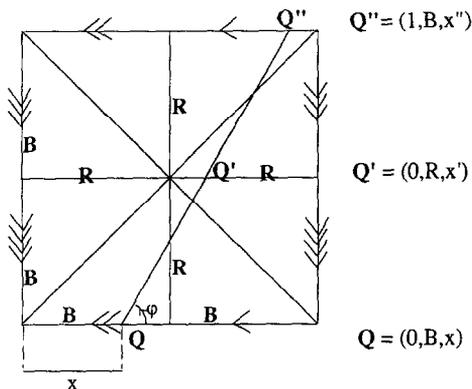


Fig. 1.

Démonstration du lemme 1. Il est clair que la couleur k' est égale à \bar{k} , où \bar{k} est égale à B si k est R , égale à R si k est B . L'application f agit sur la coordonnée x comme une rotation. Le point x' est l'image de x sous la rotation \mathcal{R}_φ , plus précisément:

$$\text{si } x \in [0, 1 - (1/\text{tg}(\varphi))[\Rightarrow x' = x + (1/\text{tg}(\varphi)),$$

$$\text{si } x \in [1 - (1/\text{tg}(\varphi)), 1[\Rightarrow x' = x + (1/\text{tg}(\varphi)) - 1.$$

La nature j' dépend de x , on vérifie que:

$$\text{si } x \in [0, 1 - (1/\text{tg}(\varphi))[\Rightarrow j' = 1 + j \bmod 2,$$

$$\text{si } x \in [1 - (1/\text{tg}(\varphi)), 1[\Rightarrow j' = j \bmod 2.$$

Ceci permet d'obtenir la formule du lemme 1. \square

Notons $(j_n)_{n \geq 0}$ la suite des natures des carrés associée à la trajectoire de Q et $K = (k_n)_{n \geq 0}$ la suite des couleurs. Etant donné la suite $(j_n)_{n \geq 0}$ et la couleur k_0 , grâce au Lemme 1 et au dessin ci-dessus, on obtient immédiatement la suite K par l'algorithme 1.

Algorithme 1.

Initialisation:

$K := k_0, k := k_0$

Incrémentatation:

Pour $n = 0, 1, \dots, \infty$ faire:

$$\text{si } j_n = 0 \text{ et } j_{n+1} = 0 \Rightarrow [K := K\bar{k}V\bar{k}, k := \bar{k}]$$

sinon

$$\text{si } j_n = 0 \text{ et } j_{n+1} = 1 \Rightarrow [K := KV\bar{k}, k := \bar{k}]$$

sinon

$$\text{si } j_n = 1 \text{ et } j_{n+1} = 0 \Rightarrow [K := KV\bar{k}, k := \bar{k}]$$

sinon

$$\text{si } j_n = 1 \text{ et } j_{n+1} = 1 \Rightarrow [K := KV\bar{k}k, k := \bar{k}]$$

Sortie:

Ecrire K .

3.2. Calcul de la complexité

Théorème 1. Soit φ un angle compris entre $\pi/4$ et $\pi/2$ tel que $\text{tg}(\varphi)$ est un nombre irrationnel, $n_0 = 2[\text{tg}(\varphi)] + 1$ et $n_1 = 3[\sin(\varphi)/\sqrt{2} \sin(\varphi - \pi/4)]$ alors la complexité

du mot K est donnée par les formules suivantes:

(1) si $[\text{tg}(\varphi)] = 1$, alors:

$$p(1) = 3,$$

$$p(2) = 6,$$

$$p(3) = 10,$$

$$p(n) = 4n + 2 \quad \text{pour } 4 \leq n \leq n_1,$$

$$p(n_1 + 1) = 4(n_1 + 1) + 6,$$

$$p(n) = 4n + 8 \quad \text{pour } n \geq n_1 + 2,$$

(2) si $[\text{tg}(\varphi)] = 2k$ avec $k \geq 1$, alors:

$$p(1) = 3,$$

$$p(2) = 6,$$

$$p(3) = 10,$$

$$p(4) = 18,$$

$$p(n) = 4n + 4 \quad \text{pour } 5 \leq n \leq n_0 - 1,$$

$$p(n_0) = 4n_0 + 6,$$

$$p(n) = 4n + 8 \quad \text{pour } n \geq n_0 + 1.$$

(3) si $[\text{tg}(\varphi)] = 2k + 1$ avec $k \geq 1$, alors:

$$p(1) = 3,$$

$$p(2) = 6,$$

$$p(3) = 10,$$

$$p(4) = 18,$$

$$p(n) = 4n + 4 \quad \text{pour } 5 \leq n \leq n_0,$$

$$p(n) = 4n + 8 \quad \text{pour } n \geq n_0 + 1.$$

Démonstration du théorème 1. Dans la première partie de la démonstration, nous introduisons un nouveau codage S des intersections de la demi-droite Δ avec le T -pavage. Dans la deuxième partie, nous calculons la complexité de la suite S et dans la troisième partie, nous montrons comment déduire la complexité de la suite K à partir de celle de la suite S .

3.2.1. Codage non orienté

On définit un autre codage de la triangulation du domaine fondamental D : on code les arêtes horizontales par 0 les verticales par 1, les droites parallèles à la première bissectrice par V_1 , celles parallèles à la deuxième bissectrice par V_2 . Nous dirons que ce codage est le codage non orienté des intersections de la demi-droite Δ avec les arêtes de la triangulation du domaine D .

Soit $S = (s_n)_{n \geq 0}$ le codage non orienté de la demi-droite Δ , nous donnons un algorithme de construction de cette suite en fonction de la suite $(j_n)_{n \geq 0}$. Cet algorithme se démontre de façon analogue à l'algorithme 1.

Algorithme 2.

Initialisation:

$S := 0$

Incrémentation:

Pour $n = 0, 1, \dots, \infty$ faire:

si $j_n = 0$ et $j_{n+1} = 0 \Rightarrow S := S1V_20$

sinon

si $j_n = 0$ et $j_{n+1} = 1 \Rightarrow S := SV_10$

sinon

si $j_n = 1$ et $j_{n+1} = 0 \Rightarrow S := SV_20$

sinon

si $j_n = 1$ et $j_{n+1} = 1 \Rightarrow S := SV_210$

Sortie:

Ecrire S .

Etant donné un mot $W = w_1 \dots w_n$ appartenant à $L(S)$ commençant 0, l'algorithme 3 calcule le mot $W' = w'_1 \dots w'_n$ de $L(K)$ qui code les mêmes intersections de la demi-droite Δ avec les droites du pavage isocèle. Cet algorithme nécessite de connaître w'_1 première couleur de W' . Dans l'algorithme 3, on se donne W sous la forme $W = w_1 \dots w_n \$$ où $\$$ est un symbole annexe différent de 0, 1, V_1 , V_2 .

Algorithme 3.

Initialisation:

$W' := w'_1, k := w'_1, i := 0$

Incrémentation:

Tant que $i < n$ faire:

si $w_i \neq 0 \Rightarrow i := i + 1$

sinon

si $w_i = 0 \Rightarrow$

si $w_{i+1} = V_1$ et $w_{i+2} = 0 \Rightarrow [W' := W'V\bar{k}, k := \bar{k}, i := i + 1]$

sinon

si $w_{i+1} = V_2$ et $w_{i+2} = 0 \Rightarrow [W' := W'V\bar{k}, k := \bar{k}, i := i + 1]$

sinon

si $w_{i+1} = 1$ et $w_{i+2} = V_2$ et $w_{i+3} = 0 \Rightarrow [W' := W'\bar{k}V\bar{k}, k := \bar{k}, i := i + 1]$

sinon

si $w_{i+1} = V_2$ et $w_{i+2} = 1$ et $w_{i+3} = 0 \Rightarrow [W' := W'Vk\bar{k}, k := \bar{k}, i := i + 1]$

sinon

si $w_{i+1} = V_1$ et $w_{i+2} = \$ \Rightarrow [W' := W'V, i := n]$

sinon

si $w_{i+1} = V_2$ et $w_{i+2} = \$ \Rightarrow [W' := W'V, i := n]$

sinon

si $w_{i+1} = V_1$ et $w_{i+2} = \$ \Rightarrow [W' := W'\bar{k}, i := n]$

sinon

si $w_{i+1} = 1$ et $w_{i+2} = V_2$ et $w_{i+3} = \$ \Rightarrow [W' := W'\bar{k}V, i := n]$

sinon

si $w_{i+1} = V_2$ et $w_{i+2} = 1$ et $w_{i+3} = \$ \Rightarrow [W' := W'Vk, i := n]$

Sortie:

Ecrire W'

L'algorithme 3 se vérifie simplement en utilisant l'algorithme 2 car tout mot de $L(S)$ est une concaténation de blocs de la forme: $0V_1, 0V_2, 01V_2, 0V_21$.

3.2.2. Complexité du codage non orienté

L'intérêt d'introduire le codage non orienté résulte de la proposition suivante. Notons p_0 la complexité du mot infini S .

Proposition. Pour tout nombre entier n supérieur ou égal à 2, $p_0(n) = 2n + 4$.

Démonstration de la proposition. Le raisonnement qui suit est de nature géométrique.

Le domaine fondamental D est invariant par translation de vecteur $(1, 1)$ donc un domaine fondamental compatible avec le codage non orienté est la réunion de 2 carrés de côtés 1, l'un de nature 0 l'autre de nature 1, avec des identifications sur le bord (voir Fig. 2). La surface \mathcal{C} obtenue est un tore, le flot est un flot linéaire de pente $\text{tg}(\varphi)$ sur le tore; ce flot est minimal puisque $\text{tg}(\varphi)$ est un nombre irrationnel.

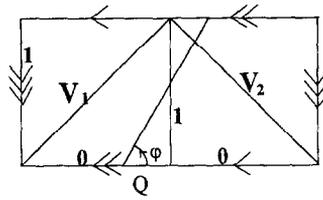


Fig. 2.

Définition. Soit W un mot fini, on note $C(W)$ l'ensemble des points des arêtes de la triangulation de D dont le codage commence par W . On dit que $C(W)$ est l'ensemble associé à W . Réciproquement, soit \mathcal{E} un sous ensemble des arêtes de la triangulation de D , on dit que \mathcal{E} est codé par W si \mathcal{E} est inclus dans $C(W)$.

On remarque que l'ensemble associé à un mot de longueur 2 est un segment contenu dans une seule arête (voir Fig. 3). Par conséquent, si W est un mot du langage de S de longueur n supérieure ou égale à 2, alors W est biprolongeable si et seulement si $\Phi^{n+1}(C(W))$ contient un sommet de la T -triangulation. Pour tout nombre entier n supérieur ou égal à 2, $p_0(n + 1) - p_0(n)$ est donc égal au nombre de sommets de la triangulation du tore \mathcal{C} c'est-à-dire égal à 2. Or il est facile de vérifier que $p_0(2) = 8$ d'où $p_0(n) = 2n + 4$ pour tout nombre entier n plus grand que 2. Ceci achève la démonstration de la proposition.

Notations. nous désignons, ci-dessous, par $\{1\} \dots \{8\}$, les segments associés respectivement à $0V_1, 01, 0V_2, 10, V_10, 1V_2, V_20, V_21$.

3.2.3. *Obtention de la complexité du mot K à partir de celle du mot S*

Notons p la complexité du mot K et σ la permutation sur les lettres B, V, R qui échange les lettres B et R .

Lemme 2. *Le langage $L(K)$ est invariant par la permutation σ . En particulier, le nombre de mots de $L_n(K)$ commençant par B est égal au nombre de mots commençant par R ; le nombre de mots commençant par VB est égal au nombre de mots commençant par VR .*

Démonstration du lemme 2. Soit W un mot de $L_n(K)$ et M un point de la triangulation de D dont le codage orienté commence par W . Le domaine D est invariant par translation de vecteur $(1, 1)$. Soit T_0 le triangle de D contenant M , T_1 le triangle translaté de T_0 de vecteur $(1, 1)$ et N le translaté de M de vecteur $(1, 1)$. La trajectoire de billard de N dans le triangle T_1 intersecte les mêmes côtés que celle de M dans T_0 à une permutation près des couleur B et R . Le codage de la trajectoire de N est donc l'image du codage de la trajectoire de M par la permutation σ . En particulier le codage de la trajectoire de N commence par le mot $\sigma(W)$. Comme le flot du billard est minimal dans D , il existe un point de l'orbite de Q dont le codage commence aussi par $\sigma(W)$. Ceci achève la démonstration du Lemme 2.

Notons $L''_n(K)$ le sous ensemble de $L_n(K)$ formé des mots commençant par B ou par VB et $p_1(n)$ le cardinal de $L''_n(K)$. Il suffit de calculer $p_1(n)$ puisque, d'après le Lemme 2, on a $p(n) = 2p_1(n)$.

Nous allons calculer $p_1(n)$ en fonction de $p_0(n)$. On définit l'application Π de $L_n(S)$ dans $L''_n(K)$ de la façon suivante: soit W appartenant à $L_n(S)$, $\Pi(W)$ est le codage sur l'alphabet $\{B, R, V\}$ de $C(W)$ si ce mot appartient à $L''_n(K)$ sinon c'est le codage sur l'alphabet $\{B, R, V\}$ du translaté de $C(W)$ par le vecteur $(1, 1)$. L'application Π est surjective par construction donc $p_1(n) \leq p_0(n)$. Nous déterminons ci-dessous, par une étude de cas, quels sont les mots de $L''_n(K)$ qui ont plusieurs antécédents par Π .

Les segments $\{1\}$, $\{3\}$ et $\{6\}$ sont codés par BV , les segments $\{2\}$ et $\{4\}$ sont codés par BR , les segments $\{5\}$, $\{7\}$ et $\{8\}$ sont codés par VB (Fig. 3).

Définition. Soit $X = x_1 \dots x_k$ un mot fini sur un alphabet fini, soit $X^\infty = (x_1 \dots x_k)^\infty$ le mot infini périodique de période $x_1 \dots x_k$. On note X^* l'ensemble des préfixes de X^∞ .

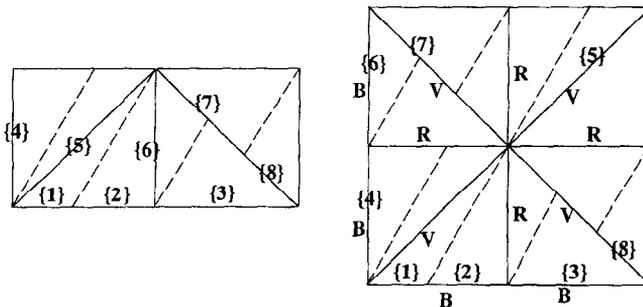
3.2.4. Etude des mots commençant par BV

Considérons les mots de longueur 3 commençant par BV , le segment $\{1\}$ est codé par BVR , le segment $\{6\}$ par BVB . Notons $\{3_1\}$ le sous segment maximal de $\{3\}$ codé par BVR , $\{3_2\}$ le sous segment maximal codé par BVB .

3.2.5. Mots commençant par BVR

Pour tout nombre entier n , les préfixes de longueur n de $(0V_10V_2)^\infty$ et de $(0V_20V_1)^\infty$ ont la même image par Π à savoir le préfixe de longueur n de $(BVRV)^\infty$, d'après l'algorithme 3.

Lemme 3. Soit $W = w_1 \dots w_k$ un mot de $L(K)$ commençant par BVR , n'appartenant pas à $(BVRV)^*$, alors W a un unique antécédent par Π .



segments $\{1\} \dots \{8\}$ dans D et \mathcal{C}

Fig. 3.

Démonstration du lemme 3. Soit i le plus petit entier tel que $W' = w_1 \dots w_i$ n'appartient pas à $(BVRV)^*$, il suffit de montrer l'affirmation pour W' .

Par minimalité de W' , d'après l'algorithme 2, on vérifie facilement que $w_i = \Pi(1)$ sinon W' serait élément de $(BVRV)^*$. Si $w_i = R$ alors $C(W')$ est un segment contenu dans $\{1\}$, si $w_i = B$, $C(W')$ est un segment contenu dans $\{3_1\}$. Ainsi W' a un unique antécédent par Π . Ceci démontre le Lemme 3. \square

La longueur maximale n_0 des mots de $L(K)$ commençant par BVR et qui ont plusieurs antécédents par Π a une interprétation géométrique simple (voir Fig. 4). $n_0 = 2[FH] + 1$ où $FH = \text{tg}(\varphi)$.

3.2.6. Etude des mots commençant par BVB

Les mots de même longueur appartenant à $(1V_20)^*$ et $(0V_21)^*$ ont la même image par Π : c'est un mot appartenant à $(BVBVR)^*$ d'après l'algorithme 3.

Lemme 4. Soit $W = w_1 \dots w_k$ un mot de $L(K)$ commençant par BVB , n'appartenant pas à $(BVBVR)^*$ alors W a un unique antécédent par Π .

Démonstration du lemme 4. Comme précédemment, il suffit de démontrer l'affirmation pour un mot minimal $W' = w_1 \dots w_i$.

Par minimalité de $W' = w_1 \dots w_i$ a deux prolongements appartenant à l'union de $L(K)$ et de $(BVBVR)^*$. En utilisant cette remarque, montrons que si i est congru à 1 modulo 3 alors $C(W')$ est contenu dans $\{6\}$, si i est congru à 0 modulo 3, $C(W')$ est contenu dans $\{3_1\}$ ce qui suffira pour conclure.

D'après l'algorithme 2, un mot de la forme $1V_2(01V_2)^k$ est suivi par $01V_2$ ou par $0V_2$. Donc un mot appartenant à $(1V_20)^*$ peut être biprolongeable seulement s'il se termine par 0. En particulier, sa longueur doit être congrue à 0 modulo 3. Si $w_1 \dots w_{i-1}$ a un antécédent dans $(1V_20)^*$, comme c'est un mot biprolongeable on doit avoir i congru

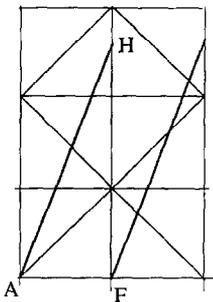


Fig. 4.

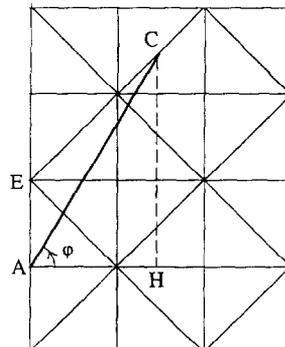


Fig. 5.

à 1 modulo 3. De même, si $w_1 \dots w_{i-1}$ a un antécédent dans $(0V_21)^*$ alors i est congru à 0 modulo 3. Ceci achève la démonstration du Lemme 4.

La longueur maximale n_1 des mots de $L(K)$ commençant par BVB et qui ont plusieurs antécédents par Π a aussi une interprétation géométrique simple (voir Fig. 5): $n_1 = 3[CH]$ où la longueur CH se calcule en fonction de l'angle φ par la formule:

$$CH = \frac{\sin(\varphi)}{\sqrt{2} \sin(\varphi - \pi/4)}.$$

3.2.7. Etude des mots commençant par BR

Les segments $\{2\}$ et $\{4\}$ codés par BR . Les mots de même longueur de $(01V_2)^*$ et $(10V_2)^*$ ont la même image par Π qui appartient à $(BRVRBV)^*$.

Lemme 5. Soit $W = w_1 \dots w_k$ un mot de $L(K)$ commençant par BR , n'appartenant pas à $(BRVRBV)^*$ alors W a un unique antécédent par Π .

La démonstration est similaire à celle du Lemme 4. On vérifie de plus que la longueur maximale commune des mots de $L(K)$ commençant par BR et qui ont plusieurs antécédents par Π est égale à n_1 .

3.2.8. Etude des mots commençant par VB

Les segments $\{5\}$, $\{7\}$ et $\{8\}$ sont codés par VB . Considérons les mots de longueur 3 commençant par VB : $\{5\}$ est codé par VBV , $\{8\}$ est codé par VBR , on définit $\{7_1\}$ le sous segment maximal de $\{7\}$ codé par VBV et $\{7_2\}$ le sous segment de $\{7\}$ codé par VBR .

3.2.9. Mots commençant par VBR

On vérifie que $\Phi(\{7_2\}) = \{2\}$ et $\Phi(\{8\}) = \{4\}$. Par conséquent, la situation pour les mots commençant par VBR est similaire à celle des mots commençant par BR quitte à décaler d'une lettre. La longueur maximale des mots de $L(K)$ commençant par VBR et qui ont plusieurs antécédents par Π est égale à $n_1 + 1$.

3.2.10. Mots commençant par VBV

Les segments $\{7_1\}$ et $\{5\}$ sont codés par VBV , or $\Phi(\{7_1\})$ et $\Phi(\{5\})$ sont codés par 0 sur l'alphabet $\{0, 1, V_1, V_2\}$ donc, d'après le lemme 3, les seuls mots codant des trajectoires issues de $\{7_1\}$ et $\{5\}$ qui peuvent avoir même image par Π appartiennent à $(V_10V_20)^*$ et à $(V_20V_10)^*$ respectivement. Soit n_2 la longueur maximale de ces mots ayant la même projection. Par les méthodes que l'on a développé précédemment, on montre que:

$$n_2 = n_0 - 1 \quad \text{si } n_0 = 4k + 1$$

$$n_2 = n_0 \quad \text{si } n_0 = 4k + 3.$$

Vu les interprétations géométriques de n_0 et n_1 , il est facile de vérifier que, si $[\text{tg}(\varphi)] = 1$ alors $n_0 = 3$ et $n_1 \geq 6$, par contre, si $[\text{tg}(\varphi)] > 1$, $n_1 = 3$ et $n_0 \geq 5$.

Le raisonnement établi ci-dessus, permet de donner des formules exactes pour la complexité p selon les valeurs de $[\text{tg}(\varphi)]$ et donc de démontrer le Théorème 1.

4. Billard dans le “demi-triangle équilatéral”

Nous donnons, dans ce paragraphe, des règles de construction des suites infinies obtenues comme codage de trajectoires de billard dans un “demi-triangle équilatéral”. Contrairement au cas précédent, dans le cas du “demi-triangle équilatéral”, la construction algorithmique de telles suites nécessite plus d’informations que la connaissance d’une seule suite sturmienne.

Dans un polygone pavant, l’application de premier retour du flot du billard sur une courbe fermée dans une direction donnée est une rotation. Dans le cas du triangle rectangle isocèle comme dans celui du triangle équilatéral, on peut trouver une courbe fermée pour laquelle la rotation associée est codée par la partition naturelle (partition qui correspond à une suite sturmienne) tandis que dans le cas du “demi-triangle équilatéral” les courbes provenant de la triangulation d’un domaine fondamental d’un tore invariant ne possèdent pas cette propriété. La difficulté supplémentaire résulte dans le fait de pouvoir trouver un algorithme permettant de construire l’itinéraire d’un point x sous une rotation d’angle α pour la partition $[0, \beta[$, $[\beta, 1[$, où x et β sont des points de $[0, 1[$ et α est un angle irrationnel. Nous donnons un tel algorithme dans le paragraphe 5.

Notations.

Soit T un demi triangle équilatéral, notons BC l’hypoténuse, AB le petit côté de l’angle droit et AC le grand côté de l’angle droit. Codons le côté BC par 0, le côté AC par 1 et le côté AB par 2. Fixons comme repère orthonormé $(A, 2AB, (2/\sqrt{3})AC)$. On repère les angles par rapport au côté AB . Soit φ un angle, compris entre $\pi/3$ et $\pi/2$ et Q un point de AB dont la trajectoire de billard de direction initiale φ ne rencontre pas de sommets du triangle T .

Un domaine fondamental d’un tore invariant par le flot du billard dans T est une réunion de 12 triangles isométriques à T formant un hexagone H . Soit f l’application de premier retour du flot du billard de direction initiale φ dans l’hexagone H sur les segments horizontaux de la triangulation de H à savoir: $I_0 = [-1, 0] \times \{\sqrt{3}/2\}$, $I_1 = [0, 1] \times \{\sqrt{3}/2\}$, $I_2 = [-\frac{1}{2}, \frac{1}{2}] \times \{0\}$ (voir Fig. 6).

Munissons ces intervalles de coordonnées adaptées au problème. Un point M appartenant à $I_0 \cup I_1 \cup I_2$ est repéré par un élément j qui indique dans quel segment se trouve le point M et un nombre réel x compris entre 0 et 1 qui donne la position de M à l’intérieur de ce segment (la position du point M est repérée par rapport

à l'extrémité gauche du segment du segment considéré). A partir de la donnée du couple (x, j) , on peut obtenir explicitement les coordonnées (x', j') du point $f(x, j)$:

x' est l'image de x sous la rotation \mathcal{R}_a d'angle a , où $a = \frac{1}{2}(\sqrt{3}/\text{tg}(\varphi)) + 1$,

$$j' = j + \zeta(x) \text{ mod } 3 \text{ où}$$

$$\zeta(x) = \begin{cases} 1 & \text{si } x \text{ appartient à } [0, 1 - a[, \\ 2 & \text{si } x \text{ appartient à } [1 - a, 1[. \end{cases}$$

Pour tout nombre entier n , notons Q_n l'image de Q par l'application f^n et (x_n, j_n) ses coordonnées adaptées. L'étude précédente permet d'obtenir la suite $J = (j_n)_{n \geq 0}$ mais ces données sont insuffisantes pour construire la suite des couleurs K à partir de règles de codage comme on l'a fait dans le cas du triangle rectangle isocèle. Pour déterminer la suite K , il faut connaître, pour tout nombre entier n , la position de $f^n(x)$ par rapport à certaines images réciproques de sommets de la triangulation de l'hexagone H à savoir $X^{(1)}$ de coordonnées adaptées $(2, \frac{1}{2})$, $X^{(2)}$ de coordonnées adaptées $(0, (1 - a)/2)$, $X^{(3)}$ de coordonnées adaptées $(0, \frac{3}{2} - a)$, $X^{(4)}$ de coordonnées adaptées $(1, 1 - \frac{1}{2}a)$ (voir Fig. 7).

A partir de ces données, on obtient la suite $(k_n)_{n \geq 0}$ par des règles de codage. Supposons que l'on a déterminé les 1 premières lettre de la suite $(k_n)_{n \geq 0}$, que k_1 code le segment contenant Q_n et que Q_{n+1} appartient au côté codé par la lettre k_n , nous donnons ci-dessous les valeurs des lettres $k_1 \dots k_h$.

Codage

si $j_n = 0$

si $x_n \in [0, (1 - a)/2[$

$$\Rightarrow [k_l = 0, k_{l+1} = 2, k_{l+2} = 1, k_{l+3} = 0 = k_h \text{ et } j_{n+1} = 1],$$

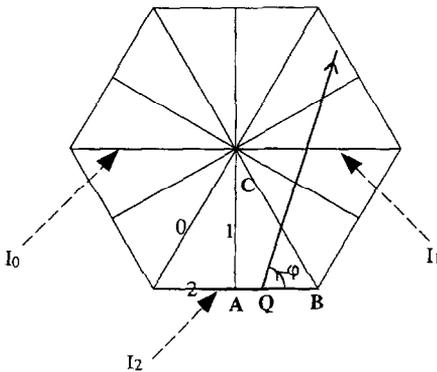


Fig. 6.

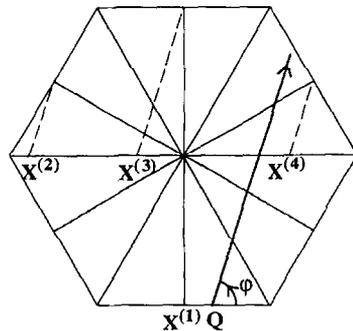


Fig. 7.

si $x_n \in [(1 - a)/2, 1 - a[$

$$\Rightarrow [k_l = 0, k_{l+1} = 1, k_{l+2} = 2, k_{l+3} = 0 = k_h \text{ et } j_{n+1} = 1],$$

si $x_n \in [1 - a, \frac{3}{2} - a[$

$$\Rightarrow [k_l = 0, k_{l+1} = 1, k_{l+2} = 0, k_{l+3} = 2 = k_h \text{ et } j_{n+1} = 2],$$

si $x_n \in [\frac{3}{2} - a, 1[$

$$\Rightarrow [k_l = 0, k_{l+1} = 1, k_{l+2} = 0, k_{l+3} = 1, k_{l+4} = 2 = k_h \text{ et } j_{n+1} = 2],$$

si $j_n = 1$

si $x_n \in [0, 1 - a[$

$$\Rightarrow [k_l = 0, k_{l+1} = 1, k_{l+2} = 0, k_{l+3} = 2 = k_h \text{ et } j_{n+1} = 2],$$

si $x_n \in [1 - a, 1 - \frac{1}{2}a[$

$$\Rightarrow [k_l = 0, k_{l+1} = 1, k_{l+2} = 2, k_{l+3} = 0 = k_h \text{ et } j_{n+1} = 0],$$

si $x_n \in [1 - \frac{1}{2}a, 1[$

$$\Rightarrow [k_l = 0, k_{l+1} = 2, k_{l+2} = 1, k_{l+3} = 0 = k_h \text{ et } j_{n+1} = 0],$$

si $j_n = 2$

si $x_n \in [0, 1 - a[$

$$\Rightarrow [k_l = 2, k_{l+1} = 0, k_{l+2} = 1, k_{l+3} = 0 = k_h \text{ et } j_{n+1} = 0],$$

si $x_n \in [1 - a, 1[$

$$\Rightarrow [k_l = 2, k_{l+1} = 1, k_{l+2} = 0, k_{l+3} = 1, k_{l+4} = 0 = k_h \text{ et } j_{n+1} = 1],$$

si $x_n \in [\frac{1}{2}, 1[$

$$\Rightarrow [k_l = 2, k_{l+1} = 0, k_{l+2} = 1, k_{l+3} = 0 = k_h \text{ et } j_{n+1} = 1].$$

Pour obtenir la suite $(k_n)_{n \geq 0}$, à partir de x et φ , il faut pouvoir placer algorithmiquement les itérés de Q sous la rotation \mathcal{R}_a par rapport aux points $X^{(1)}, X^{(2)}, X^{(3)}, X^{(4)}$ qui interviennent dans les règles de décodage. Le problème revient donc à la détermination, par un algorithme, du codage d'un point x sous une rotation d'angle α pour une partition en intervalles. Nous traitons ce problème au paragraphe suivant.

5. Codage d'une rotation sous une partition en intervalles

5.1. Partition en deux intervalles

Soit α un nombre irrationnel, $\alpha \in [0, 1]$, $x_0, \beta \in]0, 1[$ tels que $\beta \neq 1 - \alpha$. Nous donnons une formule explicite pour obtenir le infini w qui code l'itinéraire de x_0 sous la rotation d'angle α pour la partition $[0, \beta], [\beta, 1[$.

Etant donné les deux suites sturmiennes: $u = u(\alpha, x_0)$ et $v = u(\alpha, x_0 - \beta)$, le mot w est donné par le Théorème 2:

Théorème 2. *Le mot w vérifie: $w_{n+1} = w_n + v_n - u_n$, pour tout nombre entier n . De plus, la connaissance d'un nombre fini de termes des suites u et v permet de déterminer w_0 .*

Démonstration du théorème 2. Pour montrer la première partie du théorème, il suffit de montrer que, pour tout nombre entier n , $w_{n+1} - w_n = 1$ si et seulement si $v_n - u_n = 1$ et que $w_{n+1} - w_n = -1$ si et seulement si $v_n - u_n = -1$.

Fixons n un nombre entier positif, notons $x_n = \mathcal{R}_\alpha^n(x_0)$ et plaçons nous sur le cercle \mathbb{R}/\mathbb{Z} orienté. Tout ce qui est écrit dans ce paragraphe doit être considéré modulo 1 car les ensembles envisagés sont des sous ensembles du cercle, même si ce n'est pas précisé pour ne pas alourdir les notations.

On a les équivalences suivantes:

$$\begin{aligned} w_n = 1 \text{ et } w_{n+1} = 0 &\Leftrightarrow x_n \in \mathcal{R}_\alpha^{-1}([0, \beta[) \cap [\beta, 0[, \\ &\Leftrightarrow x_n \in [1 - \alpha, \beta - \alpha[\cap [\beta, 0[. \end{aligned}$$

$$\begin{aligned} w_n = 0 \text{ et } w_{n+1} = 1 &\Leftrightarrow x_n \in \mathcal{R}_\alpha^{-1}([\beta, 0[) \cap [0, \beta[, \\ &\Leftrightarrow x_n \in [\beta - \alpha, 1 - \alpha[\cap [0, \beta[. \end{aligned}$$

$$u_n = 1 \text{ et } v_n = 0 \Leftrightarrow x_n \in [1 - \alpha, 0[\cap [\beta, \beta - \alpha[.$$

$$u_n = 0 \text{ et } v_n = 1 \Leftrightarrow x_n \in [0, 1 - \alpha[\cap [\beta - \alpha, \beta[.$$

Notons $I_1 = [1 - \alpha, \beta - \alpha[\cap [\beta, 0[$, $I_2 = [\beta - \alpha, 1 - \alpha[\cap [0, \beta[$, $J_1 = [1 - \alpha, 0[\cap [\beta, \beta - \alpha[$, $J_2 = [0, 1 - \alpha[\cap [\beta - \alpha, \beta[$, montrons que $I_1 = J_1$ et $I_2 = J_2$.

Les ensembles I_1 et I_2 sont connexes [1] (ce n'est pas forcément le cas pour les ensembles correspondant à $w_n = 0$ et $w_{n+1} = 0$ et à $w_n = 1$ et $w_{n+1} = 1$).

La fin de la démonstration repose sur le lemme suivant:

Lemme 6. *Soient X_1, X_2, X_3, X_4 quatre points distincts du cercle \mathbb{R}/\mathbb{Z} tels que l'intersection de $[X_1, X_2[$ et $[X_3, X_4[$ est connexe non vide alors:*

$$[X_1, X_2[\cap [X_3, X_4[= [X_1, X_4[\cap [X_3, X_2[.$$

Le Lemme 6 se vérifie facilement sur une figure, de plus l'hypothèse de connexité est indispensable (voir Fig. 8):

Exemple. Fin de la démonstration du Théorème 2:

D'après le Lemme 6,

$$I_1 = [1 - \alpha, \beta - \alpha[\cap [\beta, 0[= [1 - \alpha, 0[\cap [\beta, \beta - \alpha[= J_1,$$

$$I_2 = [\beta - \alpha, 1 - \alpha[\cap [0, \beta[= [0, 1 - \alpha[\cap [\beta - \alpha, \beta[= J_2.$$

Ceci démontre que $w_{n+1} = w_n + v_n - u_n$.

Exemple

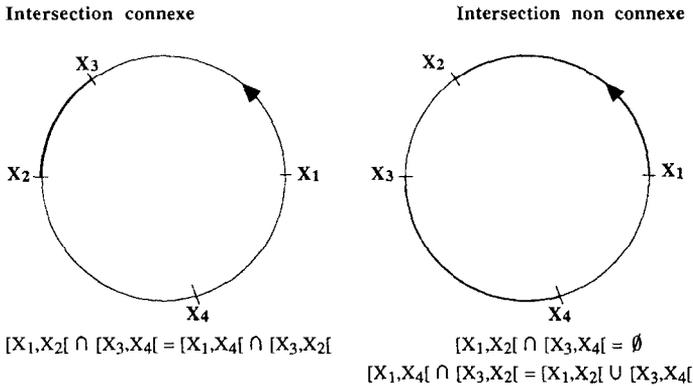


Fig. 8.

Déterminons la valeur de w_0 en fonction des suites u et v . Soit k le plus petit entier tel que $u_k \neq v_k$. D'après la relation de récurrence que l'on vient de démontrer, pour j plus petit que k , $w_j = w_k$. En particulier, $w_0 = w_k$, par conséquent il suffit de connaître la valeur de w_k . D'après le raisonnement précédent, si $u_k = 1$ et $v_k = 0$ alors x_k appartient à I_1 et $w_k = 0$, si $u_k = 0$ et $v_k = 1$ alors x_k appartient à I_2 et $w_k = 1$. La connaissance de $u_0 \dots u_k$ et $v_0 \dots v_k$ permet donc de déterminer explicitement w_0 . Par contre le nombre de termes de u et v nécessaires pour calculer w_0 n'est pas borné car les suites u et v peuvent avoir un début commun arbitrairement long selon les valeurs de α et β . Ceci achève la démonstration du Théorème 2.

6. Conséquences

6.1. Conséquence arithmétique

Soient α un nombre irrationnel $\alpha \in [0, 1]$, x et y deux points distincts de $[0, 1[$, notons u la suite sturmiennne $u(x, \alpha)$ et v la suite sturmiennne $u(y, \alpha)$. Soit ω la suite de terme général $\omega_n = \sum_0^{n-1} v_n - \sum_0^{n-1} u_n$, pour tout nombre entier n ω_n prend au plus trois valeurs 0, 1, -1 car les suites sturmiennes sont équilibrées [7]. D'après le Théorème 2, la suite ω est, en fait, une suite sur un alphabet à deux lettres $\{0, 1\}$ ou $\{-1, 0\}$.

6.2. Caractérisation

Soit ω_0 tel que la suite w soit une suite sur l'alphabet $\{0, 1\}$, alors, par construction, la suite ω est le codage du point x sous la rotation d'angle α , pour la partition $[0, x - y[$, $[x - y, 1[$.

Remarque. Le Théorème 2 généralise un théorème de Rote qui donne une caractérisation des suites qui codent les itinéraires d'une rotation pour la partition $[0, \frac{1}{2}[, [\frac{1}{2}, 1[, [18]$.

Corollaire. La méthode établie au Théorème 2 permet d'obtenir des règles de construction du codage z de l'itinéraire d'un point x_0 sous la rotation \mathcal{R}_α pour une partition en intervalles de la forme $[0, \beta_1[, \dots, [\beta_{k-1}, \beta_k[, [\beta_k, 1[$, où $\beta_1 \dots \beta_k$ vérifient: $\beta_0 = 0 < \beta_1 < \dots < \beta_k < \beta_{k+1} = 1$.

Etant donné les suites sturmiennes $u(\alpha, x_0), u(\alpha, x_0 - \beta_1) \dots u(\alpha, x_0 - \beta_k)$, on construit par l'algorithme précédent, les suites w^1, \dots, w^k telles que w^i code l'itinéraire de x_0 sous la partition $[0, \beta_i[, [\beta_i, 1[$ pour tout i compris entre 0 et k . On en déduit facilement la suite z : soit n un nombre entier, il existe i compris entre 1 et k tel que $\beta_i < n\alpha + x \bmod 1 < \beta_{i+1}$ donc $(w^1)_n = \dots = (w^i)_n = 1$ et $(w^{i+1})_n = \dots = (w^k)_n = 0$ et par conséquent $z_n = i$.

Remarque. le corollaire ci-dessus permet de construire les mots infinis obtenus comme codage de trajectoires de billard dans un demi-triangle équilatéral (Section 4).

Par ailleurs cette méthode peut s'appliquer aux échanges de 3 intervalles car l'application induite d'un échange de trois intervalles sur le plus grand intervalle admissible est une rotation codée par intervalles.

Références

- [1] P. Alessandri, Codage des rotations, Mémoire de D.E.A., Université Claude Bernard, Lyon I (1993).
- [2] P. Arnoux, C. Mauduit, I. Shiokawa and J. Tamura, Complexity of sequences defined by billiards in the cube, *Bull Soc. Math. France* **122** (1994) 1–12.
- [3] Yu. Baryshnikov, Complexity of trajectories in rectangular billiards, preprint 1994.
- [4] J. Berstel and M. Pocciola, A geometric proof of the enumeration formula for Sturmian words, *Inf. J. Algebra Computation* **3** (1993) 349–355.
- [5] C. Boldrighini, M. Keane and F. Marchetti, Billiards in polygons, *Ann. Probab.* **6** (1978) 532–540.
- [6] I.P. Cornfeld, S.V. Formin and Ya.G. Sinai, *Ergodic Theory* (Springer, Berlin, 1982).
- [7] E.M. Coven and G.A. Hedlund, Sequences with minimal block, growth, *Math. System Theory* **7** (2) (1973) 138–153.
- [8] A. de Luca and F. Mignosi, Some combinatorial properties of Sturmian words, *Theoret. Comput. Sci.* **136** (1994) 361–385.
- [9] S. Duluc and D. Gouyou-Beauchamps, Sur les facteurs des suites de Sturm, *Theoret. Comput. Sci.* **71** (1990) 381–400.
- [10] G.A. Hedlund and M. Morse, Symbolic Dynamics II. sturmian trajectories, *Amer. J. Math.* **62** (1940) 1–42.
- [11] P. Hubert and G. Rauzy, Dynamique Symbolique du billard dans le triangle équilatéral, préprint, 1994.
- [12] P. Hubert, Dynamique symbolique des billards polygonaux rationnels, Thèse, Université Aix-Marseille II, 1995.
- [13] P. Hubert, Complexité de suites définies par des billards rationnels, *Bull Soc. Math. France* **123** (1995) 257–270.

- [14] S. Ito and S. Yasutomi, On continued fractions, substitutions and characteristic sequences, *Japan J. Math* **16** (1990) 287–306.
- [15] F. Mignosi, On the number of factors of Sturmian words, *Theoret. Comput. Sci.* **82** (1991) 71–84.
- [16] N. Nishioka, I. Shiokawa, J. Tamura, Arithmetical properties of a certain power series, *J. Number Theory* **42** (1992) 61–87.
- [17] G. Rauzy, Mots infinis en arithmétique, dans: M. Nivat et D. Perrin, eds., *Automata on Infinite Words*, Lecture Notes in Computer Science, Vol. 192 (Springer, Berlin, 1985) 165–171.
- [18] G. Rote, Sequence with subword complexity $2n$, *J. Number Theory* **46** (1994) 196–213.
- [19] P. Séébold, Fibonacci morphisms and sturmian words *Theoret. Comput. Sci.* **88** (1991) 367–384.
- [20] S. Tabachnikov, Billiards, *Panoramias et Synthèses* **1** (1995) *Soc. Math. France*.