



ELSEVIER

Physica D 84 (1995) 329–356

PHYSICA D

Coding chaotic billiards II. Compact billiards defined on the pseudosphere

Denis Ullmo, Marie-Joya Giannoni

Division de Physique Théorique, Institut de Physique Nucléaire, F-91406 Orsay, Cedex, France¹

Received 15 November 1993; revised 25 October 1994; accepted 13 February 1995

Communicated by U. Frisch

Abstract

We study “à la Morse coding” of compact billiards defined on the pseudosphere. As for most bounded systems, the coding is non-exact (except for the non-generic case of tiling billiards). However, two sets of approximate grammar rules can be obtained, one specifying forbidden codes, and the other allowed ones. In-between some sequences remain in the “unknown” zone, but their relative amount can be reduced to zero as one lets the length of the approximate grammar rules go to infinity. The relationship between these approximate grammar rules and the “pruning front” introduced by Cvitanović et al. is discussed.

1. Introduction

For many decades, symbolic dynamics have become a large branch of research in the study of Dynamical Systems. Works in this field encompass a great variety of approaches and methods. For Euclidean billiards, the pioneering works of Sinai and Bunimovitch [1] leading to criteria for ergodicity, K - and B -properties, implicitly use symbolic dynamics via the analysis of Markov partitions. For systems defined by the geodesic motion on a manifold (or some kinds of surfaces) of negative curvature, the B -property was established by Ornstein and Weiss [2], again with simple Markovian rules related to a “symbolic” description. For the latter systems, the first idea of coding (with no underlying intention of proving any chaotic property) is due to Morse [3]. From that time, a lot of highly mathematical involved “variations on this Morse’s theme” were achieved.

More recently, the rapid development of the so called “periodic orbit theory” [4–6], in which classical or semiclassical quantities are evaluated as sums running over all the periodic orbits of a system, have emphasized the necessity of knowledge of an accurate description of the periodic orbits. For chaotic systems, having at one’s disposal symbolic dynamics constitutes at the present day the best way to reach that goal. Such a major advantage of knowing symbolic dynamics was first exploited by Gutzwiller, in a long series of paper on the

¹ Unité de Recherche des Universités Paris XI et Paris VI associée au C.N.R.S.

Anisotropic Kepler Problem (AKP) [7]. Taking benefit of an “à la Morse-like” coding of AKP, he was able to make a very clever application of the semiclassical trace formula to recover quantum energy levels from classical periodic orbits. Since then, a lot of works have been dedicated to investigations of the link between the classical and quantum mechanics of chaotic systems, some of them in particular concerning the free motion on a constant negative curvature manifold, or on billiards defined on the pseudosphere (see e.g. [8–12]). For a large number of these studies, an explicit use of a coding of the classical trajectories was made (e.g. [10,12–14]). In the analysis of chaotic classical systems also, practical calculations of quantities such as Kolmogorov-Sinai entropy, Lyapunov exponents, or rate of decay of correlations, are often supported by a symbolic description of the dynamics [15–17]. Therefore, in addition to its intrinsic interest, to gain a deeper understanding of what is a coding and of ways to characterize it should be useful for all developments implicitly using symbolic dynamics.

In a previous article [18] (referenced as I in the following), we have studied “à la Morse-like” coding of trajectories in non-compact billiards of finite area defined on the Poincaré disk. The billiards considered were polygons whose all vertices lie on the boundary of the Poincaré disk; this property allowed us to observe the coding behavior in a extremely simple dynamical configuration which, however, shares with other chaotic dynamical systems many crucial characteristics of the mechanisms governing the coding of Hamiltonian systems, in particular the feasibility of foliating a Poincaré section with stable and unstable manifolds.

The kind of “à la Morse” coding considered in Ref. I and here can be sketched as follows: having defined a finite alphabet (the labels 1, 2, ..., N of the sides of the billiard), one associates to any trajectory the doubly infinite “word”, or “code”

$$k_{-\infty} \dots k_{-1} k_0 \dots k_{\infty} \quad (k_j = 1, 2, \dots, N)$$

obtained by enumerating the sides successively hit by the trajectory during its whole history. Deriving *exact* symbolic dynamics requires to find a *finite and complete* set of admissibility rules (“grammar rules”) in such a way that there is a one-to-one correspondence between the ensemble of doubly infinite codes, and that of actual trajectories in the billiard. In the case considered in Ref. I the simplicity of the system allowed to find exact symbolic dynamics with only one grammar rule (the geometrically trivial one, i.e. the interdiction of repeating twice successively the same letter).

This property is however making the billiards studied in Ref. I rather non-representative of the chaotic billiards, and presumably of chaotic Hamiltonian systems in general.² In contrast to open systems, for which examples of exact coding are not exceptional [15,20], it is rather uncommon for bounded systems to find an exact coding. In fact the only such example we know of, other than the billiards studied in Ref. I and than systems defined on the pseudosphere [21,22] is the anisotropic Kepler problem (AKP) [7], or mappings constructed more or less for this purpose. Actually, the coding of AKP suffers from a lack of proof of *uniqueness*, (i.e. *at most* one trajectory associated with a given code), while for billiards, the difficulty stands from *existence* (*at least* one trajectory associated with a given code). The problem of uniqueness for AKP may be only technical, whereas the problem of existence for bounded billiards is actually connected with the underlying physics of the system. Another example of dynamical system with exact coding, which is not strictly speaking a bounded system, consists in collinear electronic motion in a two-electron atom [13]. In both cases - AKP, two-electron atoms - an important role seems to be played by the existence of trajectories corresponding to a collision with the nucleus, which is specific of anisotropic Kepler potential. On the contrary, a large variety

² Notice that until recently, the only systems shown to be ergodic were billiards (more generally systems generated by geodesic motion on some manifold). See however now [19].

of examples show that even for systems for which a “good” coding can be found [14,23], it is usually not possible to specify the codes actually associated with a trajectory by a finite number of grammar rules.

However, the aim of a coding is to provide a description of the trajectories, especially the periodic ones, with as ultimate purpose to study some classical or quantum properties of a system. Even if this description is not exact, it can be of great practical help if sufficiently accurate. The difficulties arising for such non-exact coding were already addressed by several authors for a Hénon-type map [24,25]; in these works, a very nice conjecture (the “pruning front conjecture”) has been put forward, stating that the codes which are not associated to any trajectory can be described via the introduction of a “pruning front”, defined in a plane of symbols.³ This conjecture is supported by various numerical studies (see however [26]). It has moreover been generalized to other kinds of systems, such as billiard systems for instance, in particular in the Ph. D. thesis of K.T. Hansen [23] which contains one of the most outstanding attempts to tackle the difficult problem of non-exact coding.

However, most of the present “status of the art” concerning non-exact coding is to a great extent supported by approaches where numerical simulations, intuitions and conjectures, are playing an important role. Therefore, it is presumably worthwhile to study one example of system, with dynamics sufficiently simple for a “rigorous” treatment of the coding properties to be hopeful, and yet sufficiently generic to be regarded as a paradigm of bounded hyperbolic systems, at least in connection with the coding properties. It is that question that we want to address in that paper by looking at compact⁴ polygonal billiards on a constant negative curvature surface. As we shall see, their main interest comes both from their generic character and their reasonable simplicity. Concerning the genericity, the coding is, in most cases, non-exact: some codes has to be pruned from the tree of all possible coding sequences, a procedure which can be visualized, in the spirit of the work of Cvitanović et al. [25], by working in a plane of symbols, in which a pruning front can be drawn. However, in the exceptional case of billiards tiling the Poincaré disk by reflections, and which therefore are fundamental domains of a discrete group of isometries, we shall generalize the conclusion of Ref. 1 that the coding is exact. As regards to simplicity, one keeps the advantages coming from the polygonal character of the billiards. A substantial number of results can be derived with admissible rigor. Among them is the construction of approximated grammar rules, which, except may be for a few billiards, can be refined up to an arbitrarily good description of codes actually associated to a trajectory provided that longer and longer grammar rules are included. Translated in the symbol plane, these approximate grammar rules define a coarse-grained version of the pruning front, which is more and more closely approximated as the grammar is made convergent. This may give some insight on the way the “pruning front conjecture” behaves on a more general footing.

Most of the derivations concerning the approximate grammar rules can however be achieved using the original Morse alphabet (i.e. the labels of the sides of the billiard) and working in the Poincaré section. We shall therefore proceed in the first sections without making any reference to the symbol plane. After some brief preliminary comments in Section 2, we shall in Section 3 state the basic grammar rules, applying to any polygonal billiard without requiring any calculations. For tiling billiards, we shall moreover see that these simple grammar rules are, except for a simple transformation, yielding an exact coding. In Section 4 we introduce the notion of canonical sequences that can be chained each other with the assurance that an associated trajectory does exist. In Section 5 we illustrate the preceding considerations on a simple example. Section 6 is then mostly devoted to the introduction of the symbol plane of the considered billiards, and to the expression, in this latter representation, of the approximate grammar rules; we shall also describe there an algorithm allowing to obtain automatically the grammar of any length. This will allow us to consider some long grammar rules, for which

³ The plane of symbols is merely a plane in which, up to a simple transformation, the abscissae are the codes to the past of the trajectory and the ordinates their codes to the future. (More details are given in Section 6.)

⁴ There is no loss of generality here, the extension to any polygonal billiard being straightforward.

differences with the finite grammar obtained using the prescription given in [23] can be exhibited. In Section 7, the question of convergence of the grammar is considered. Some concluding remarks are gathered in the last section.

Let us finally mention that the derivation of some of the results presented here were obtained through rather simple, though tedious, geometrical considerations, which may be too long to be developed in the text. Therefore, under the fatherly pressure of our referee, we were led in such cases to avoid full demonstrations, rather stating precisely the results, and mentioning the main underlying physical considerations. The reader (if any) interested in complete proofs can find them in [27]. One proof however, which we think interesting in itself, has been included in the Appendix.

2. Preliminaries

In this section, we introduce the basic definitions and notations required for the coding. We shall therefore here specify the parameters defining the billiards, and introduce (i) the Poincaré section coordinates (φ_i, M) , (ii) the expression of the bounce mapping T , (iii) the curve C_k limiting, in the plane (φ_i, M) , the set of points actually associated to a trajectory inside the billiard, (iv) the *exit* and *entry* domains of a side of the billiard. Because all these notions are very similar to their analogs presented in Ref. I for billiards whose all vertices lie on the boundary of the Poincaré disk, we shall remain here extremely concise and recommend Ref. I to the reader for a more extensive discussion. We also refer to Ref. I for an account of the basic properties of the Poincaré disk, or to [28] for a more complete introduction.

The billiards we shall consider in the following are compact polygons defined on the Poincaré disk \mathbf{D} for which all inner angles at vertices are smaller than $\pi/2$. The geometry of the billiard is completely defined by the position in the Poincaré disk of the N vertices V_k of the polygon ($k \in \{1, \dots, N\}$). A vertex V_k will be characterized by the polar angle $\tilde{\beta}_k$ and the number $\tilde{\tau}_k$ such that a boost of rapidity $\tilde{\tau}_k$ acting in the direction $\tilde{\beta}_k$ maps the center of the Poincaré disk on V_k . The vertices V_k are supposed to occur in increasing order with k when running along the boundary of the billiard anticlockwise, which means that $0 \leq \tilde{\beta}_1 < \tilde{\beta}_2 < \dots < \tilde{\beta}_N \leq 2\pi$. We then define the side number \mathbf{k} of the polygon as the one joining V_{k-1} to V_k . The initial and final angles on the boundary of the disk of the corresponding geodesic are denoted by α_i^k and α_f^k respectively. An example of such a billiard is represented in Fig. 1.

The dynamics of the system are, as usually, defined by the free geodesic motion inside the billiard and by specular reflections on its sides. For any oriented geodesic defining the trajectory between two successive bounces, which is characterized by its initial and final angles φ_i and φ_f on the boundary of \mathbf{D} , we shall use as Poincaré section coordinates the pair $(\varphi_i, M = \cot(\frac{\varphi_f - \varphi_i}{2}))$, where M can be interpreted as an angular momentum. The bounce mapping

$$T : (\varphi_i, M) \longrightarrow (\varphi'_i, M')$$

is area-preserving. As in Ref. I, we introduce the N functions f_k ($k = 1, \dots, N$) defined by the relation

$$\tan\left(\frac{f_k(\varphi_i) - \beta_k}{2}\right) = e^{-2\tau_k} \cot\left(\frac{\varphi_i - \beta_k}{2}\right). \quad (2.1)$$

Then, T can be expressed for a trajectory hitting side \mathbf{k} by

$$\begin{aligned} \varphi'_i &= f_k(\varphi_i) \\ \varphi'_f &= f_k(\varphi_f), \end{aligned} \quad (2.2)$$

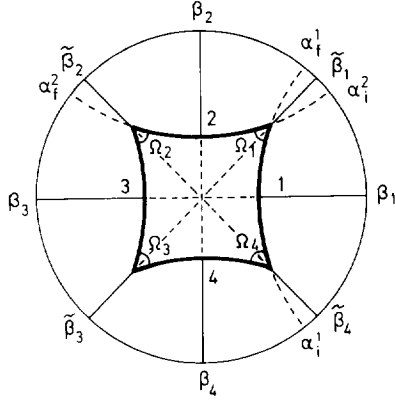


Fig. 1.

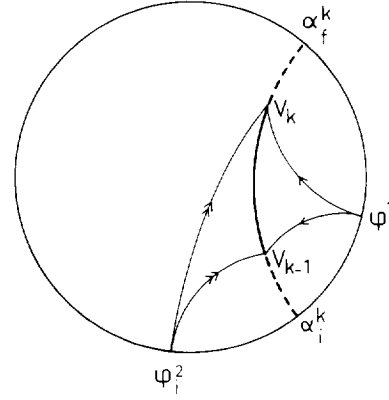


Fig. 2.

Fig. 1. Example of a compact polygonal billiard in the Poincaré disk: the square billiard with vertex angles $\Omega_k = \pi/3.6$ ($k = 1, 2, 3, 4$). The angles $\alpha_{i,f}^k$, β_k , and $\tilde{\beta}_k$ defined in the text are represented.

Fig. 2. Limit geodesics intersecting some side k , for a fixed initial angle φ_i . They correspond, in the Poincaré surface of section, to points on the curves C_{k-1} or C_k . The geodesics leave or enter side k depending on whether $\varphi_i \in [\alpha_f^k, \alpha_i^k]$ (as φ_i^1) or $\varphi_i \in [\alpha_i^k, \alpha_f^k]$ (as φ_i^2).

which leads for the angular momentum to

$$M' = -Me^{2\tau_k} \sin^2\left(\frac{\varphi_i - \beta_k}{2}\right) [1 + e^{-4\tau_k} \cot^2\left(\frac{\varphi_i - \beta_k}{2}\right)] - \sinh 2\tau_k \sin(\varphi_i - \beta_k). \quad (2.3)$$

In Eq. (2.3), β_k and τ_k are respectively the *angle* and *rapidity* of the boost mapping a diameter on side k :

$$\beta_k = \frac{\alpha_i^k + \alpha_f^k}{2} \quad (2.4)$$

and

$$e^{\tau_k} = \cot\left(\frac{\alpha_f^k - \alpha_i^k}{4}\right). \quad (2.5)$$

The definition of a symbolic dynamics for the bounce mapping T requires to partition the surface of section into a certain number K ($\leq \infty$) of regions, labelled by some alphabet $X = \{a_1, a_2, \dots, a_K\}$. One then associates to any trajectory the doubly infinite word, or *code*

$$k_{-\infty} \dots k_{-n} \dots k_0 k_1 \dots k_n \dots k_{\infty} \quad (k_j \in X, j = -\infty \dots +\infty) \quad (2.6)$$

obtained by listing all the labels of the regions the trajectory has successively entered during its whole history. The Morse coding, which we are interested in, consists in coding a trajectory by listing the label of the sides of the billiard the trajectory successively hit. This amounts to take as partition of the surface of section the set of all *entry* domains, where the entry domain of some side k is defined as the set of points associated with trajectories entering the side k at the time defined as the present. Equivalently, one can use as partition the set of *exit* domains, defined in an analogous way but for trajectories exiting side k . It is therefore important first to define the available domain in the (φ_i, M) -plane and to understand how it is divided into *exit* and *entry* domains.

The points actually associated with a trajectory inside the billiard are such that the corresponding geodesic intersects a side of the polygon defining the billiard; consequently the limiting cases are obtained by looking at

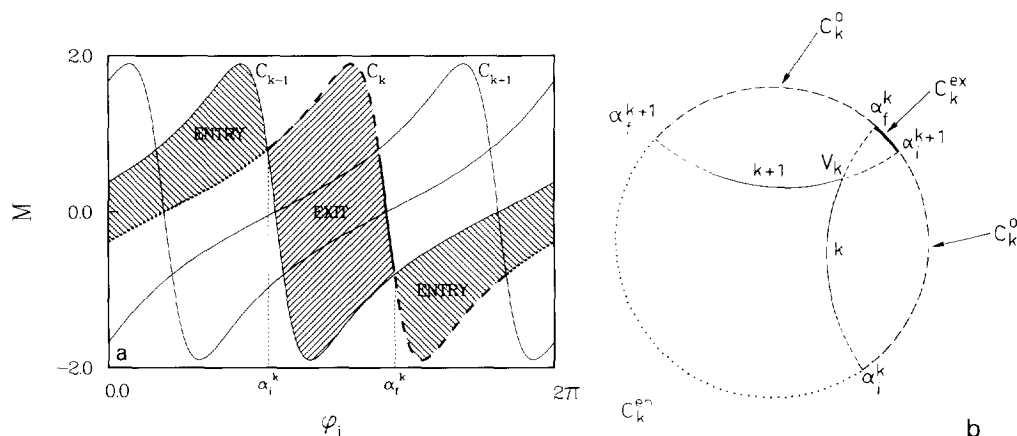


Fig. 3. Loci of the initial angles φ_i corresponding to the different parts of a curve C_k . (a) in the Poincaré surface of section; (b) on the boundary of the Poincaré disk. Dotted lines are associated to $(C^{en})_k$, thick continuous lines to $(C^{ex})_k$, and thick interrupted lines to $(C^o)_k$ (see text for further explanations). Also shown in Fig. 3a are the entry and exit domains of side k .

geodesics passing through a vertex V_k . For each vertex V_k , let us introduce the curve C_k in the (φ_i, M) -plane associated with the geodesics passing through V_k . The equation of C_k is given by (see Ref. [28]):

$$M(\varphi_i) = \frac{\sin(\varphi_i - \tilde{\beta}_k)}{\cos(\varphi_i - \tilde{\beta}_k) - \coth(\tilde{\tau}_k)}. \quad (2.7)$$

According to this definition of curves C_k , one can see in Fig. 2 that a point (φ_i, M) of the Poincaré section corresponds to a geodesic intersecting the side k if and only if it lies between the curve C_{k-1} and the curve C_k . Then, the geodesic exits the side k if $\varphi_i \in [\alpha_i^k, \alpha_f^k]$ and enters the side k if $\varphi_i \in [\alpha_f^k, \alpha_i^{k+1}]$. The limiting cases $\varphi_i = \alpha_i^k$ or $\varphi_i = \alpha_f^k$ correspond to portions of trajectories following the side k , i.e. passing through the two vertices V_{k-1} and V_k ; they are thus represented in the Poincaré section by intersections of C_k and C_{k-1} . Such trajectories can follow the boundary in any of the two possible opposite directions, so that the curves C_{k-1} and C_k intersect exactly in two points. Since the billiard is convex (all vertex angles were supposed smaller than $\pi/2$), a geodesic entering the billiard has exactly two intersections with its boundary. Therefore a point (φ_i, M) of the Poincaré section belongs to exactly one entry domain and one exit domain. Since moreover two points in the Poincaré disk, lying on different sides of the polygon, univocally define a geodesic, each entry domain is partitioned into $(N-1)$ portions of exit domains, and conversely. These remarks are illustrated in Fig. 3a.

As will be seen, the curves C_k just introduced are going to play a important role in the study of the coding. Consequently, they deserve some closer description before entering into the heart of the topic. Let us therefore consider two adjacent sides of the billiard intersecting at the vertex V_k . As shown in Fig. 3b, the geodesics passing through V_k can be separated into three classes whose bouncing properties are quite different. They are associated with three parts of the curve C_k : $(C^{en})_k$, $(C^{ex})_k$ and $(C^o)_k$ defined as follows:

- $(C^{en})_k$ is the set of points of C_k whose abscissae φ_i lie in $[\alpha_f^{k+1}, \alpha_i^k]$. Such points lie in the entry domains of both sides k and $k+1$. They correspond to geodesics which, coming from any side of the billiard other than k and $k+1$ are going towards V_k . The $(C^{en})_k$'s are the loci of discontinuity of the bounce mapping T since one may arbitrarily decide that the trajectory is hitting the k or the side $k+1$ at V_k .
- $(C^{ex})_k$ is the set of points of C_k whose abscissae belong to $[\alpha_i^{k+1}, \alpha_f^k]$. Such points thus lie in the exit domain of both k and $k+1$. They correspond to geodesics emerging from the vertex V_k , and the set of $(C^{ex})_k$ ($k \in \{1, \dots, N\}$) is the locus of discontinuity of T^{-1} . $(C^{ex})_k$ is the time-reversal of $(C^{en})_k$.

- Finally, $(C^0)_k$ is the part of C_k for which φ_i lies in $[\alpha_i^k, \alpha_i^{k+1}] \cup [\alpha_f^k, \alpha_f^{k+1}]$. As seen in Fig. 3b, a geodesic corresponding to a point of $(C^0)_k$ (which by definition passes through V_k) intersects the billiard only in V_k . Since any increasing of the modulus of M (i.e. any reduction of the radius of the geodesic) with fixed φ_i results in removing the intersection with the billiard, the $(C^0)_k$'s actually define the boundary, in the plane (φ_i, M) , of the allowed part of the Poincaré section. The points of $(C^0)_k$ are in the exit domain of \mathbf{k} (respectively $\mathbf{k} + \mathbf{1}$) and in the entry domain of $\mathbf{k} + \mathbf{1}$ (respectively \mathbf{k}) if φ_i belongs to $[\alpha_i^k, \alpha_i^{k+1}]$ (respectively $[\alpha_f^k, \alpha_f^{k+1}]$).

The different curves $(C^{\text{en}})_k$, $(C^{\text{ex}})_k$ and $(C^0)_k$ are represented in Fig. 3a for the particular case of a square billiard with angles at vertex $\Omega_k = \pi/3.6$.

3. First set of grammar rules

The Morse coding consists in characterizing any trajectory inside the billiard by the enumeration

$$k_{-\infty} \dots k_{-n} \dots k_0 k_1 \dots k_n \dots k_{\infty}, \quad (3.1)$$

of the sides it successively hits during its whole history. Each letter k_i is taken in the set $X = \{1, \dots, N\}$ of the labels of the sides of the billiard. In order to get a representation in the Poincaré section, one has to define the position of the present; we therefore introduce the notation

$$k_{-\infty} \dots k_{-n} \dots k_0 \bullet k_1 \dots k_n \dots k_{\infty}, \quad (3.2)$$

where the circle separating k_0 and k_1 indicates that at the time taken as the present the trajectory just left the side \mathbf{k}_0 and is going to hit the side \mathbf{k}_1 . In the following, we shall also consider finite codes w

$$k_{-(n-1)} \dots k_{-1} k_0 \bullet k_1 \dots k_p \quad (k_i \in X). \quad (3.3)$$

The word w will be said to be a *code to the past* if $p = 0$, a *code to the future* if $n = 0$, and generally a *mixed code* if $n \geq 0$ and $p \geq 0$. Finally, for any finite or infinite code w , we shall note $\mathcal{R}(w)$ the set of points in the Poincaré section with code w . It is clear that the bounce mapping T defined in Section 2 acts as

$$T\mathcal{R}(w) = \mathcal{R}(w'),$$

where w is of the form Eq. (3.3) and

$$w' = k_{-(n-1)} \dots k_{-1} k_0 k_1 \bullet k_2 \dots k_p \quad (k_i \in X).$$

The purpose of this paper is to search for the proper *grammar rules* associated with the considered coding, that is for the whole set of constraints on the succession of letters $k_i \in X$ such that one and only one trajectory is associated to a given doubly infinite word. The question of uniqueness (no more than one trajectory associated with a given code) can be easily overcome; indeed, transposing to the present case a proof given in Ref. I, it can be shown that, due to the hyperbolic character of the mapping T , the region $\mathcal{R}(w)$, in the Poincaré section, associated to any doubly infinite word w such as Eq. (3.2), is either reduced to a point or empty. Therefore, the search for the grammar rules we are going to establish essentially requires to distinguish between codes actually associated to a trajectory and those which are not (i.e. to solve the question of existence). As can be readily seen, the hyperbolic geometry forbids one side to be hit twice consecutively. This was found in Ref. I as the only grammar rule - referred to as **G0** in the following - for coding non-compact billiards with all vertices at infinity. The relative simplicity of that case was, as we shall see, due to the absence of vertices

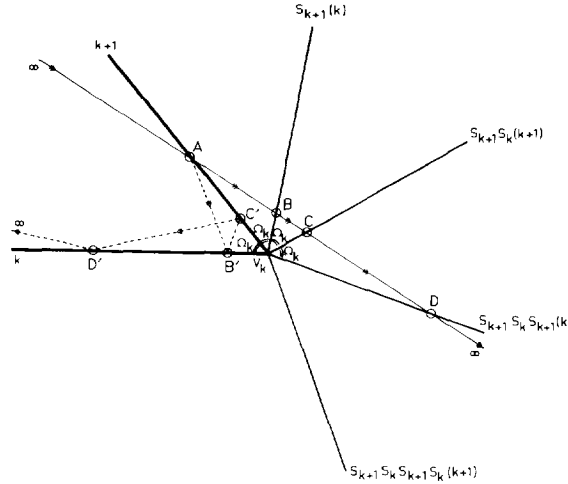


Fig. 4. Blow-up of a trajectory bouncing at a vertex. The emerging geodesic is constructed using the method of images: the two thick lines are the actual boundaries of the billiard, and the thin ones passing through V_k are images of the former by symmetry, as indicated on the figure. The true trajectory $\infty AB'C'D'\infty$ is obtained from the geodesic $\infty ABCD\infty$ by bringing back the different segments of this latter inside the billiard using the proper symmetry. See text for further explanation.

with angles different from zero. In this section, we shall state the basic additional grammar rules existing for compact billiards.

A first simple grammar rule arises from the geometrical fact that, in hyperbolic as well as Euclidean geometry, it is not possible to bounce more than $(\text{Int}(\pi/\Omega) + 1)$ times inside a wedge of angle Ω . For the billiards studied in Ref. I, for which all the angles at vertex were equal to zero, there were therefore no restrictions concerning the number of successive bounces a trajectory could make inside a vertex. This peculiarity happened to be the root of the exactness of the coding. To see why there is such a restriction for vertex angles different from zero, let us examine the behavior of a trajectory hitting a vertex V_k by observing it “with a microscope”. In the following illustrations $\Omega_k = \pi/3.6$, thus,

$$\nu_k \equiv \text{Int}(\pi/\Omega_k) = 3.$$

Locally, curved geometry behaves like Euclidean geometry, and all geodesics can be represented as straight lines. When an orbit passes through a vertex V_k , there are two possibilities: it can first hit side k or side $k+1$. In Fig. 4 is represented a trajectory bouncing first on side $k+1$. To make clear what happens for the successive bounces within V_k , we have drawn “virtual sides” $S_{k+1}(k)$, $S_{k+1}S_k(k+1)$, etc., around V_k , where S_k and S_{k+1} are respectively the symmetries with respect to sides k and $k+1$. Also drawn is a virtual straight line $\infty ABCD\infty$ representing the trajectory inside the images of the interior of the vertex, from which one can recover the actual trajectory $\infty AB'C'D'\infty$ by:

$$\begin{aligned} AB' &= S_{k+1}(AB) \\ B'C' &= S_k S_{k+1}(BC) \\ C'D' &= S_{k+1} S_k S_{k+1}(CD) \\ D'\infty &= S_k S_{k+1} S_k S_{k+1}(D\infty). \end{aligned}$$

One can see that the virtual orbit hits the vertex at the points A, B, C, D and exits from the vertex between sides $S_{k+1}S_k S_{k+1}(k)$ and $S_{k+1}S_k S_{k+1}S_{k+1}(k+1)$. For the configuration of this figure, there are thus $4 = (\nu_k + 1)$ bounces within the vertex before the orbit exits from V_k . It is easily seen, extrapolating from Fig. 4 that,

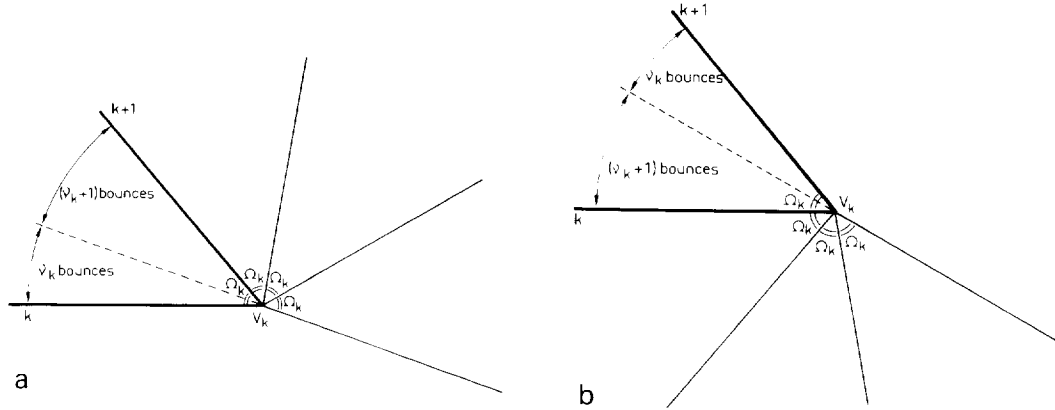


Fig. 5. Number of bounces occurring at vertex k depending of the incoming direction. (a) For a first bounce taking place on side $k + 1$; (b) For a first bounce taking place on side k .

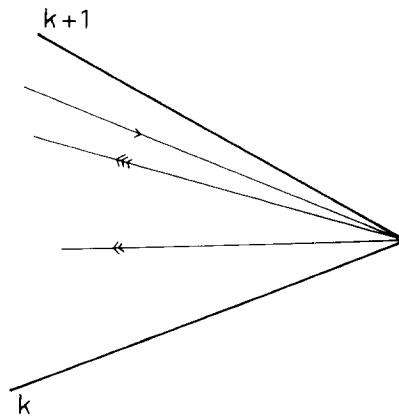


Fig. 6. Illustration of the discontinuity of the motion for a trajectory bouncing at a vertex. Depending on whether the incoming trajectory is consider to first hit side k or side $k + 1$, two possible directions exist for the outgoing trajectory.

depending on the incoming direction of the orbit, there can occur $(\nu_k + 1)$ or ν_k bounces within the vertex. The corresponding code therefore contains a succession of ν_k or $(\nu_k + 1)$ letters k and $k + 1$, which we shall in the following call a *vertex sequence*. Fig. 5 illustrates this considerations. Fig. 6 shows an incoming trajectory, together with the two possible outgoing ones depending on the side hit at the first bounce.

Let us introduce, for any vertex V_k , the following definition:

Definition. For any interior angle Ω_k at the vertex V_k , let

$$\nu_k = \text{Int} \left(\frac{\pi}{\Omega_k} \right), \quad (3.4)$$

where $\text{Int}(\xi)$ is the integer part of ξ . We call a *vertex sequence* (at vertex k) any succession of ν_k or $(\nu_k + 1)$ letters k and $(k + 1)$.

A first consequence of the above discussion is that, since a trajectory hitting V_k cannot bounce more than $(\nu_k + 1)$ successive times on sides k and $k + 1$, and since clearly one cannot increase this number by moving apart from the vertex, the following grammar rule **G1** holds:

G1: all words containing a sequence of alternating \mathbf{k} and $(\mathbf{k} + 1)$ of length strictly larger than $(\nu_k + 1)$ are forbidden.

Another consequence is that any trajectory hitting a vertex has a doubly infinite code which contains a vertex sequence (note however that the converse is obviously not true). Reciprocally, trajectories whose codes contain no vertex sequence will never come too close from a vertex. Assuming, for the coding purpose, that the only troubles originate from what is happening at the vertices, it seems natural that words containing no vertex sequences will in general be associated to a trajectory. And indeed, it can be shown [27] that the following grammar rule **G'1** also holds:

G'1: all words containing no vertex sequence are allowed.

For any word w , if **G1** is transgressed $\mathcal{R}(w)$ is empty. On the other hand, if **G'1** is fulfilled $\mathcal{R}(w)$ is non-empty, being a single point for doubly infinite codes such as Eq. (3.2) and a connected region of (φ_1, M) for finite mixed code such as Eq. (3.3).

3.1. Heuristic justification of the grammar rule **G'1**

The justification of the grammar rule **G'1** is obtained, as was done in Ref. I to show the existence of a trajectory associated to any code fulfilling **G0** for billiard with all their vertices at infinity, by constructing recursively the regions in the Poincaré section associated to code of any length. In practice, it is sufficient to work with finite words to the past since any finite mixed code can be obtained by applying T^{-n} to some code to the past, and the limit of infinite words is essentially straightforward (see Ref. I).

Following closely the approach of Ref. I, all regions associated to finite codes to the past can be constructed as follows. Regions $\mathcal{R}(w_n)$ associated to codes to the past of length n are constructed from those associated to codes of length $(n - 1)$ by intersecting them with the different entry domains $\mathcal{R}(\bullet k_0)$ ($k_0 = 1, \dots, N$), and applying the bounce mapping. Clearly, a necessary and sufficient condition for the finiteness of the grammar would be that at each iteration of this process, the regions $\mathcal{R}(w_n)$ obtained in that way intersect all the entry domains, therefore insuring that, at the next step, all possible codes to the past of length $(n + 1)$ satisfying **G0** are associated to non-empty regions. This property was straightforward for the billiards studied in Ref. I.

In the present case, the situation is more complicated due to the finiteness of the vertex angles. This led us to define vertex sequences, and to consider here only words containing no vertex sequence. What can be seen is that, in general, under this condition, everything behaves as in the particular example displayed in Fig. 7a for a square billiard with $\Omega_k = \pi/3.6$ ($k = 1, 2, 3, 4$). In this picture, it is shown, starting from a region $\mathcal{R}(k_{-1}k_0\bullet) = \mathcal{R}(42\bullet)$, how the regions associated to the “son” words $42k_1\bullet$ ($k_1 = 3, 4, 1$) obtained from $42\bullet$ are constructed. They are the images by the bounce mapping T of the regions $\mathcal{R}(42\bullet k_1) = \mathcal{R}(42\bullet) \cap \mathcal{R}(\bullet k_1)$ which are non-empty [connected] domains, basically because $\mathcal{R}(42\bullet)$ is a “nearly vertical” strip extending from the top to the bottom of the allowed part of the Poincaré section. In exactly the same way as in Ref. I, it therefore intersects all the entry domains $\mathcal{R}(\bullet k_1)$ ($k_1 = 3, 4, 1$) (see Fig. 3). Then, the region $\mathcal{R}(42k_1\bullet) = T(\mathcal{R}(42\bullet k_1))$ extends, as seen in the figures, from top to bottom of the Poincaré section. This is the case either if, as in Fig. 7a, k_1 and k_0 are not consecutive (i.e. $k_1 = 4$ since here $k_0 = 2$), or if as in Fig. 7b, k_1 and k_0 are consecutive, (region $\mathcal{R}(423\bullet)$).

It is seen on this example that the crucial point allowing for the recursive construction of codes to the past of any length is that the associated regions extend from the bottom to the top of the Poincaré section, i.e. that, for any code to the past w_n (say of length n), one boundary of $\mathcal{R}(w_n)$ lies on curve $(C^0)_k$ limiting the upper part of the Poincaré section and another one on a curve $(C^0)_{k'}$ limiting the lower part of the Poincaré section. It is this property which insures that all the entry domains are intersected, so that at the next iteration, all the possibilities allowed by **G0** remain open. Thus, the recursive construction of the regions associated to finite

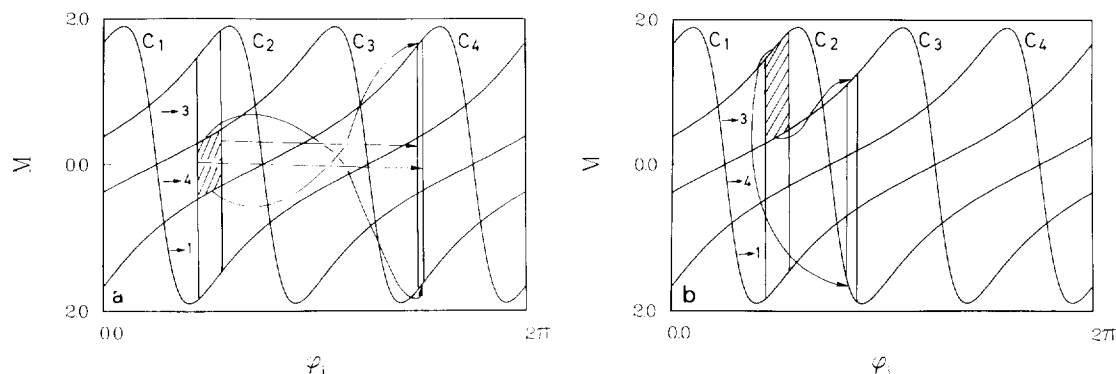


Fig. 7. Construction of $\mathcal{R}(42k_1\bullet)$ as the image by the bounce mapping T of the (shaded) intersection of $\mathcal{R}(42\bullet)$ with the entry domain $\mathcal{R}(\bullet k_1)$. The figures displayed correspond to the square billiard with vertex angles $\pi/3.6$. (a) for k_1 not consecutive to 2 ($k_1 = 4$); (b) for k_1 consecutive to 2 ($k_1 = 3$). The arrows indicate how the boundaries of $\mathcal{R}(42\bullet k_1)$ are mapped on $\mathcal{R}(42k_1\bullet)$. The entry domains of sides 3, 4, and 1, are respectively labeled by $\rightarrow 3$, $\rightarrow 4$, and $\rightarrow 1$. For a better visualisation of the boundaries of these domains, see Fig. 3a specifying the entry domain of side k .

codes to the past of any length proceeds in exactly the same way as in Ref. I. The only difference lies in the fact that for compact billiards, showing that $\mathcal{R}(w_n)$ extends from top to bottom of the Poincaré section is not any more trivial.

What we have to understand here is more precisely how this latter property of $\mathcal{R}(w_n)$ is related to the absence of any vertex sequence inside the word w_n . For this purpose, let us seek more closely what can be the boundaries of $\mathcal{R}(w_n)$. Clearly, since crossing the boundary of $\mathcal{R}(w_n)$ means that, at a given time, the associated trajectory starts bouncing on another side of the billiard, the trajectories associated to points on the boundary of $\mathcal{R}(w_n)$ are characterized by the fact that, at some place, they pass through a vertex. In other words, the boundaries of $\mathcal{R}(w)$ have to be images or preimages of the curve C_k , or a piece of a curve C_k itself.

We shall not here try to prove that, among all the possible images and preimages of the C_k 's, $\mathcal{R}(w_n)$ necessarily “manages” to have as part of its boundary a piece of some $(C^0)_k$ and a piece of some $(C^0)_{k'}$ respectively limiting the top and bottom of the Poincaré section (for a complete derivation, see [27]). Instead, we shall rather point out what specific characteristic of the $(C^0)_k$'s, distinguishing them for instance from the $(C^{\text{en}})_k$'s and the $(C^{\text{ex}})_k$'s, is underlying this property. Let us therefore come back to Fig. 4, and consider a point P of the Poincaré section lying on some C_k , with its associated doubly infinite code

$$w(P) = k_{-\infty} \dots k_0 \bullet k_1 \dots k_{\infty}.$$

It follows from the definitions of the three portions $(C^{\text{ex}})_k$, $(C^{\text{en}})_k$, $(C^0)_k$ of C_k given in Section 2, that in the Poincaré surface of section:

- the part ∞A of Fig. 4 is associated with a point of $(C^{\text{en}})_k$
- the parts AB, BC, CD are associated with a point of $(C^0)_k$
- the part $D\infty$ is associated with a point of $(C^{\text{ex}})_k$.

Therefore, on $(C^{\text{en}})_k$ the bounces are to happen, and the code to the future starts with a vertex sequence. On $(C^{\text{ex}})_k$ the bounces have just happened thus the code to the past starts with a vertex sequence. Finally, for $(C^0)_k$ the symbol of the present time is in the middle of the vertex sequence. If moreover P belongs to $\mathcal{R}(w_n)$, one has, with $w_n = k_{-n} \dots k_0 \bullet$

$$w(P) = \dots \underbrace{k_{-n} \dots k_0}_{w_n} \bullet \overbrace{\dots \dots \dots}^s \dots \quad \text{for } P \in (C^{\text{en}})_k$$

$$\begin{aligned}
 w(P) &= \cdots \underbrace{k_{-n} \cdots k_0}_{w_n} \bullet \cdots & \text{for } P \in (C^{\text{ex}})_k \\
 w(P) &= \cdots k_{-n} \cdots \underbrace{k_0}_{s} \bullet \cdots & \text{for } P \in (C^0)_k,
 \end{aligned}$$

where s denotes the vertex sequence. If $P \in (C^{\text{en}})_k$, an infinitesimal displacement of P making the associated trajectory pass on the other side of the vertex will only modify the future of $w(P)$, but not its past. Therefore in that case P lies inside $\mathcal{R}(w_n)$, but not on its boundary. More generally, for any finite mixed code w , a vertex sequence associated to a boundary of $\mathcal{R}(w)$ must have some letters in common with w . This prevents $(C^{\text{en}})_k$ to border $\mathcal{R}(w_n)$. If $P \in (C^{\text{ex}})_k$, as soon as the word w_n is longer than the considered vertex sequence, this latter is contained in w_n , which is excluded by hypothesis. Therefore, the points of a curve C_k which border a region $\mathcal{R}(w_n)$ such that w_n contains no vertex sequences should belong to some $(C^0)_k$. And indeed, following carefully the recursive construction of all words to the past such as w_n , it can be shown [27] that $\mathcal{R}(w_n)$ extends from $(C^0)_{k_0}$ on the top of the Poincaré section to $(C^0)_{k_0-1}$ on its bottom.

Starting with a region extending from top to bottom of the Poincaré section, and which therefore intersects all available entry domains, one thus ends with “sons” words whose associated regions also extend from top to bottom of the Poincaré section. Therefore, for words containing no vertex sequence, the recurrence can go forward in exactly the same way as in Ref. I, yielding the grammar rule **G’1**.

3.2. A non-generic case: tiling billiards

We consider in this subsection the very peculiar case of tiling billiards, for which we shall see that a simple transformation of the grammar rules **G1** and **G’1** actually yields an exact coding. Among the set of polygonal billiards in the Poincaré disk, tiling ones represent only a zero measure subset. Nevertheless, they have been (and continue to be) overwhelmingly studied by both mathematicians and physicists because of their connections with discrete groups of isometries, and the fact that semiclassical quantization is exact in that case (Selberg trace formula). The notion of tiling the Poincaré disk is actually close to the idea one has of the tile-layer’s work, i.e. covering the space by congruent figures, with no overlapping and no gaps. In the present case, the congruent figures are polygons (defined by the billiard under consideration). The previous conditions allow to pick up one of these polygons, called “the original cell”, and define it as the *fundamental domain* of a discrete group. Each cell of the tessellation is “similar” to the original one, in the sense that it is an image by a given element (isometry) of the group. Requiring to tessellate “by reflections” implies moreover that two adjacent cells are images one of each other by a reflection across their common geodesic boundary. It can be shown (see Ref. [29], or by using the same demonstration as in Ref. [30]), that a necessary and sufficient condition for a billiard to tessellate *by reflections*, is that all the inner angles at vertex Ω_k are such that π/Ω_k is an integer, i.e.:

$$\nu_k = \frac{\pi}{\Omega_k} \in \mathbb{N}. \quad (3.5)$$

For such tiling billiards it is shown in [27] that one can replace **G1** and **G’1** by:

T1: More than ν_k repetitions of two successive letters k and $k+1$ are forbidden

T2: All ν_k repetitions of two successive letters k and $k+1$ are considered to be equivalent whatever may be the letter which started them.

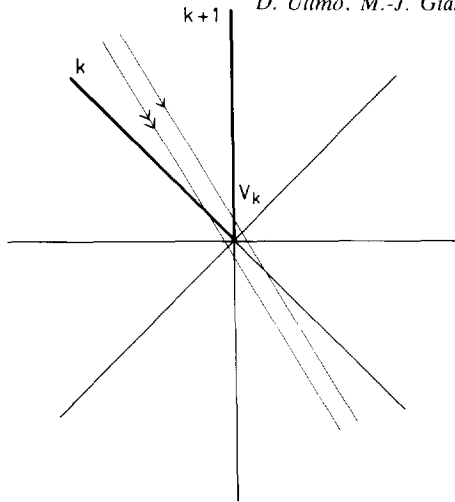


Fig. 8.

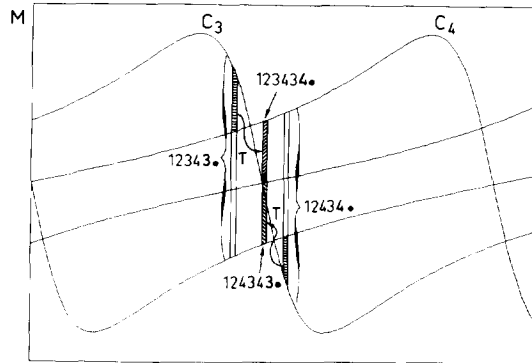


Fig. 9.

Fig. 8. Same construction as for Fig. 4, but for a vertex fulfilling Eq. (3.5) (here $\Omega_k = \pi/4$ so that $\nu_k = 4$). Contrary to the non-tiling case, the number of bounces at V_k is always equal to ν_k , and the outgoing trajectory does not depend on whether side k or side $k+1$ is supposed to be hit first.

Fig. 9. Construction of the region $\mathcal{R}(12|3|\bullet)$ for the (tiling) square billiard with angles at vertex $\pi/4$. Although $\mathcal{R}(12343 \bullet 4)$ and $\mathcal{R}(12434 \bullet 3)$ are disconnected regions, their images by the bounce mapping match exactly, so that the union of the two images is a connected region, of shape A.

The condition **T2** is not *stricto sensu* a grammar rule. It rather amounts to introduce a set of N new letters $\{|1|, \dots, |N|\}$ in the coding alphabet, where $|k|$ stands for the two equivalent successions of ν_k letters k and $(k+1)$. With that slight modification of the alphabet and the grammar rules **G0** and **T1**, the coding is exact. That is, for doubly infinite words, any code w transgressing **G0** or **T1** corresponds to an empty domain $\mathcal{R}(w) = \emptyset$, and in any other case $\mathcal{R}(w)$ is associated to one and only one point in the Poincaré section.

The idea of the proof can be summarized as follows: as far as the coding is concerned, what essentially distinguishes tiling billiards from non-tiling ones is the fundamentally different behavior of trajectories bouncing in a vertex. More precisely, let us consider, in the Poincaré section, the points (φ_i, M) lying on a curve $(C^{\text{en}})_k$. As already stated, the bounce mapping T is discontinuous for such points, since T depends on whether the trajectory is considered as bouncing on side k or on side $k+1$. As was shown in Fig. 4, the way the trajectory emerges from the vertex can be obtained, in both cases, by constructing copies of the vertex, together with the “virtual” (see the previous comments of Fig. 4, drawn for a non-tiling billiard) geodesic $\Delta(\varphi_i, M)$, and returning from each copy to the original vertex, where the actual motion takes place, by applying the proper isometry.

Fig. 8, which is the analogue of Fig. 4 for the tiling square defined by $\Omega_k = \pi/4$, makes clear that the condition Eq. (3.5) has two important consequences. First, whenever $\Delta(\varphi_i, M)$ is considered as passing on one side or on the other one of the vertex, it moves off from V_k in the ν_k 'th cell constructed. Therefore, no more than ν_k bounces can occur, yielding **T1**. Moreover, this geodesic, considered as hitting side k or $k+1$, can be viewed as two different paths joining the same endpoints. However, for tiling billiard, the isometry mapping the cell C_{ν_k} of the tiling, in which $\Delta(\varphi_i, M)$ exits from the vertex, onto the actual billiard, only depends on the cell and not on the path to this cell. Therefore the actual trajectory inside the billiard, being obtained by applying this isometry to the portion of $\Delta(\varphi_i, M)$ lying inside C_{ν_k} , does not depend on which side $\Delta(\varphi_i, M)$ is supposed to hit.

For the mapping T , this can be summarized by the following: if a vertex V_k of the billiard fulfills Eq. (3.5), then

- (i) Each point of $(C^{\text{en}})_k$ bounces exactly ν_k times on V_k before exiting V_k .
- (ii) Although $T, T^2, \dots, T^{(\nu_k-1)}$ are discontinuous on $(C^{\text{en}})_k$, T^{ν_k} is indeed continuous. Moreover, since the points of $(C^0)_k$ correspond to geodesics intersecting the billiard only on V_k (i.e.: the part of the geodesic corresponding to an actual trajectory is of a zero length), there is no discontinuity at all for the motion in phase space.

For billiards tessellating by reflections, the properties (i) and (ii) stated above can be used to formulate the grammar rules **T1** and **T2** insuring that any finite code to the past fulfilling these conditions corresponds to a non-empty connected region. Fig. 9 illustrates the preceding considerations. It shows for the square $\Omega_k = \pi/4$ ($k = 1, 2, 3, 4$) two regions associated with codes $\tilde{k}232\bullet$ and $\tilde{k}323\bullet$ (with $\tilde{k}=1$), thus describing a trajectory having bounced in the vertex no. 2, $(\nu_k - 1) = 3$ times. Also shown in the same figure are the two regions associated with the words $\tilde{k}2323\bullet$ and $\tilde{k}3232\bullet$, which match to define a unique region $\mathcal{R}(\tilde{k}|k|)$ ($k = 2$) extending from top to bottom of the Poincaré section. This matching now appears as the very origin of the exactness of the coding, and as a direct consequence of the tiling property. In the same time as we get from it a positive answer to the question of constructing exact symbolic dynamics for tiling billiards, we understand that a finite number of grammar rules will never lead to a similar result for non-tiling ones, for which we need to imagine the construction of approximate grammar rules together with a converging procedure.

Finally, it is worth stressing that, for coding purpose, the tiling property of the billiards considered in this subsection has been used only through the condition Eq. (3.5). If some, but not all, of the vertices of a billiard fulfil this condition, the treatment described in this subsection can be used for these vertices, the remaining ones keeping all the complications we discuss now.

4. Dressed vertex sequences

From now on we shall only consider the generic case of non-tiling billiards. Let \mathcal{W} be the set of doubly infinite words which, say, fulfil **G0**, and $\overset{\circ}{\mathcal{W}} \subset \mathcal{W}$ the subset of words actually associated to a trajectory. The grammar rules **G1** and **G'1** stated above can be considered as approximate grammar rules since, **G1** specifies a subset of \mathcal{W} encompassing words which *are not* associated to trajectories (those containing a succession of more than $(\nu_k + 1)$ letters \mathbf{k} and $\mathbf{k} + 1$), and **G'1** a subset of \mathcal{W} of words which *are* associated to trajectories (those containing no vertex sequence). In between, words in which vertex sequences occur but such that **G1** is not transgressed remain in the “unknown zone”. To reduce this zone, we may try, by adding letters on the right or left side of the vertex sequences, to create “dressed vertex sequences” for which it can be decided whether they exclude from $\overset{\circ}{\mathcal{W}}$ all words containing them (*excluding sequence*), or if on the contrary they cure the problem due to the presence of the vertex sequence (*including sequence*).⁵ We shall see in this section that it is indeed possible to characterize such dressed vertex sequences. Moreover this can be done in such a way that, as will be seen in Section 7, it is possible to describe arbitrarily well $\overset{\circ}{\mathcal{W}}$ provided one lets the length of the grammar rules (i.e. of the including and excluding sequences) increase up to a sufficient value.

Excluding dressed sequences are easy to characterize since it suffices that the regions associated to, say, the corresponding codes to the past, are empty. On the other hand, for *including dressed sequences* it is asked that if all the vertex sequences of a word w can be dressed into an including sequence, then $\mathcal{R}(w)$ is non-empty whatever may be the position of the dressed sequences in w and whatever is going on elsewhere. “Including

⁵ What we call a sequence is basically a finite word, with the only difference that there is no specification of the present time.

dressed sequences” s_i are thus not at all characterized by the fact that $\mathcal{R}(s_i \bullet)$ is non-empty. We shall see however that they can be characterized by investigating the shape of regions associated to finite codes.

For this purpose, let us consider more closely how can look in general the boundaries of a region $\mathcal{R}(w_{np})$ associated to a finite mixed code of length $n + p$, $w_{np} = k_{-(n-1)} \dots k_0 \bullet k_1 \dots k_p$ ($n, p \geq 0$). We shall use as illustration the particular case of a polygon with, say, four sides ($N = 4$), and such for instance that all (ν_k) 's ($k = 1, 2, 3, 4$) are equal to three (e.g. $\Omega_k = \pi/3.6$). Moreover we shall consider a particular w_{np} , chosen essentially in such a way that it has no particular property, namely the word of length thirteen

$$w_{67} = 3 \underline{12324} \bullet \underline{1434312}.$$

Underlined characters are just there to make the vertex sequences of w_{67} more apparent. As already mentioned, the boundaries of a $\mathcal{R}(w_{np})$ are necessarily images or preimages by the bounce mapping of pieces of curves C_k , that is points of the Poincaré section whose doubly infinite codes contain a vertex sequence. Since, as discussed in Section 3.1, images of pieces of curve C_k corresponding to vertex sequences exterior to w_{np} has to be strictly inside $\mathcal{R}(w_{np})$ (not on its boundary), one is left with only two kinds of boundaries for $\mathcal{R}(w_{np})$:

- (i) *Images of pieces of curve C_k , associated to vertex sequences interior to w_{np} .* For w_{67} above, there are two inner vertex sequences, one being at vertex 3 (4343), the other at vertex 2 (232). Using the fact (see Section 3.1) that for the (doubly infinite) code of a point of $(C^{\text{ex}})_k$ (resp. $(C^{\text{en}})_k$), the vertex sequence is on the left (resp. on the right) of the present time, one can be even more specific. For instance for the vertex sequence at vertex 3, if a piece of the boundary of $\mathcal{R}(w_{67})$ comes from C_3 , it has to lie at the intersection of $T^{-1}((C^{\text{en}})_3)$ and $T^{-5}((C^{\text{ex}})_3)$.
- (ii) *Images of pieces of curves C_k , associated to vertex sequences on one border of w_{np} (i.e. such that one has to add a few letters on the future or past of w_{np} to complete the vertex sequence).* Going on with w_{67} this would correspond to points whose codes belong to one of the four following types (letters in parenthesis are optional):

$$\begin{aligned} w &= k_{-\infty} \dots w_{67} 1(2) \dots k_{\infty} & (\in T^{-5}(C^{\text{en}})_1) \\ w' &= k'_{-\infty} \dots w_{67} 32(3) \dots k'_{\infty} & (\in T^{-6}(C^{\text{en}})_2) \\ w'' &= k''_{-\infty} \dots (4) 34 w_{67} \dots k''_{\infty} & (\in T^{+5}(C^{\text{ex}})_3) \\ w''' &= k'''_{-\infty} \dots (2) 32 w_{67} \dots k'''_{\infty} & (\in T^{+5}(C^{\text{ex}})_1). \end{aligned}$$

For words containing no vertex sequence, for which existence is proven, only the second kinds of boundaries are present. One may wonder - and we shall see that this is indeed the case - whether the problems due to the vertex sequences occurring inside w_{np} are cured if, in the same way, $\mathcal{R}(w_{np})$ has no boundaries of type (i).

This leads us to introduce the following definition

Definition. A region $\mathcal{R}(w_{np})$, associated to a finite mixed code w_{np} will be said to have a “canonical shape” if $\mathcal{R}(w_{np})$ is non-empty and its boundaries are only of the type (ii). (To avoid any ambiguity, we emphasize that this condition systematically excludes regions associated to words w_{np} which start or end with a vertex sequence.)

If $\mathcal{R}(w_{np})$ has a canonical shape, this is obviously also the case for $T^\ell \mathcal{R}(w_{np})$, if $-p \leq \ell \leq (n-1)$. We shall then say that a sequence

$$s = k_{-(n-1)} \dots k_0 k_1 \dots k_p$$

is *canonical* if one (and thus all) word w obtained by specifying the place of the present time inside s , or just on its right or left side, is such that $\mathcal{R}(w)$ is of canonical shape.

Note that all sequences containing no vertex sequence are canonical, and that, if a sequence s is canonical, the time reversed sequence

$$s^{-1} = k_p \dots k_1 k_0 \dots k_{-(n-1)}$$

obtained by writing the letters of s in reversed order is also canonical.

The following proposition allows to specify completely the inclusion rules of length M just by investigating what are the boundaries of the regions associated to mixed codes of length M .

Proposition. Let w be any finite or infinite word. A sufficient condition for $\mathcal{R}(w)$ to be a non-empty region is that all its vertex sequences can be “dressed” into a canonical sequence, i.e. that each of its vertex sequences is a part of a canonical sequence.

In particular, if w is a doubly infinite word, it belongs to \mathcal{W}^o .

The above proposition relies on the following intermediate result:

Let

$$\begin{aligned} s &= k_{-(n-1)} \dots k_0, \\ s' &= k'_{-(\ell-1)} \dots k'_0 \dots k'_p \quad (0 \leq \ell < n) \end{aligned}$$

be two canonical sequences such that the ℓ last letters of s are the same as the ℓ first letters of s' (i.e.: $k_m = k'_m$ for $-(\ell-1) \leq m \leq 0$). The new sequence s'' constructed by “concatenation” of s and s'

$$s'' = k_{-(n-1)} \dots k_0 k'_1 \dots k'_p = k_{-(n-1)} \dots k_{-\ell} k'_{-(\ell-1)} \dots k'_n$$

is also canonical, provided that no vertex sequence is created by the procedure of concatenation (i.e. that all the vertex sequences contained in s'' belong either to s or to s').

Note that the particular case $\ell = 0$ just consists in making s and s' juxtaposed. If moreover s or s' is a sequence of length one, which is always canonical, this means that adding a letter on the left or right side of a canonical sequence maintains its canonical character unless it completes a vertex sequence.

To give some idea of how it happens to be so, let us consider the words w , w' and w'' obtained respectively from s , s' and s'' by specifying the position of the present time at some place (for instance on the right of $k_0 = k'_0$). One needs to demonstrate i) that $\mathcal{R}(w)$ intersects $\mathcal{R}(w')$, and ii) that $\mathcal{R}(w'') = \mathcal{R}(w) \cap \mathcal{R}(w')$ contains no boundary of type (i).

Showing the first point is not quite straightforward, and we shall here admit that statement i) is always satisfied. In fact this can be proven using essentially the same kind of reasoning as we sketched for words containing no vertex sequence (see [27] for more details). Let us then assume that $\mathcal{R}(w'')$ has a boundary of type (i), that is corresponding to a vertex sequence s_v interior to w'' . Since a boundary of $\mathcal{R}(w'')$ is either boundary of $\mathcal{R}(w)$ or boundary of $\mathcal{R}(w')$, both being of canonical shape, s_v cannot be interior neither to w nor to w' . These remarks lead to the result stated above that s'' can happen to be non-canonical only if a vertex sequence has been created when concatenating s and s' .

Notice that if dressed vertex sequences are chained from nearest neighbour to nearest neighbour in a doubly infinite word w , then the condition that no vertex sequence is created at their junction is trivially fulfilled. Thus the proposition naturally follows from the above consideration since we have established that canonical sequences can be “chained” to form new longer canonical sequences. This, in particular, insures that the related words are associated with non-empty regions.

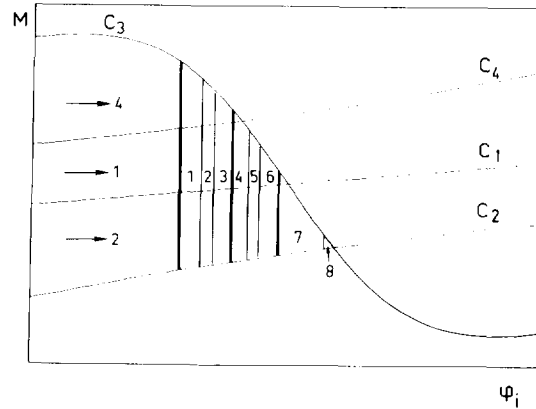


Fig. 10. Blow-up, for the square billiard with angles $\pi/3.6$, of the region $\mathcal{R}(343\bullet)$. This latter has been divided in eight subregions \mathcal{R}_i ($i = 1, 2, \dots, 8$), obtained by addition of two letters on the left side of $343\bullet$ (see text). The thin nearly vertical lines are pieces of some $T^4((C^{\text{ex}})_k)$, when the thick ones are parts of some $T^2((C^{\text{ex}})_k)$ or $T^3((C^{\text{ex}})_k)$.

To summarize, we have seen in this section that it is possible to characterize dressed vertex sequences (the canonical ones) which, although they contain a vertex sequence, can be treated for the coding purpose as if they contained none. These “canonical” sequences define inclusion rules which can be used in addition to the trivial exclusion rules. Basically, what characterizes canonical sequences is that, although they may contain vertex sequences, these latter are not actually associated to any trajectory bouncing at the corresponding vertex. This makes it clear that as far as the coding is concerned, the difficulties are coming from the trajectories hitting a vertex - whose doubly infinite codes therefore contain a vertex sequence - but not from the vertex sequences themselves if they are not associated to a trajectory going through a vertex. We shall now see how the characterization of a canonical sequence obtained in this section can be used practically to refine the grammar rules **G1** and **G'1**.

5. Simple example of application

In this section, we shall consider a simple concrete example, the square billiard with vertex angles $\Omega_k = \pi/3.6$ ($k = 1, 2, 3, 4$), and work out in a pedestrian way the first approximate grammar rules.

To construct the sets of exclusion and inclusion rules at a given order M ,⁶ one has in principle to investigate, for all possible sequences of length M , whether they are “excluding” sequences, “including” (i.e.: canonical) ones, or none of the two (“unknown” sequences). However, when increasing the length of the sequences from one unit, their classification is most of the time straightforward. For instance, it is clear that the daughter sequences $s_{M+1} = s_M k$ and $s'_{M+1} = k' s_M$ obtained by addition of a letter on the right or left side of a sequence s_M of length M are excluding sequences if s_M is so, whatever may be k and k' . In the same way, s_{M+1} and s'_{M+1} are canonical if s_M is so, unless k or k' completes a vertex sequence. Thus the essential part of the job will consist in decomposing the unknown sequences, and to investigate to which class (i.e.: excluding, including, or unknown) belong the constructed daughter sequences.

The first step in the refinement of the grammar therefore consists in decomposing the vertex sequences. In Fig. 10 the region $\mathcal{R}(343\bullet)$, associated to the vertex sequence $s = 343$, has been divided into eight sub-regions \mathcal{R}_i ($i = 1, \dots, 8$):

⁶ In the following, we shall refer to this set of inclusion and exclusion rules as the *grammar of length M* .

$$\begin{aligned}
\mathcal{R}_1 &= \mathcal{R}(32343\bullet) \leftrightarrow s_1 \\
\mathcal{R}_2 &= \mathcal{R}(42343\bullet) \leftrightarrow s_2 \\
\mathcal{R}_3 &= \mathcal{R}(12343\bullet) \leftrightarrow s_3 \\
\mathcal{R}_4 &= \mathcal{R}(21343\bullet) \leftrightarrow s_4 \\
\mathcal{R}_5 &= \mathcal{R}(31343\bullet) \leftrightarrow s_5 \\
\mathcal{R}_6 &= \mathcal{R}(41343\bullet) \leftrightarrow s_6 \\
\mathcal{R}_7 &= \mathcal{R}(14343\bullet) \leftrightarrow s_7 \\
\mathcal{R}_8 &= \mathcal{R}(24343\bullet) \leftrightarrow s_8
\end{aligned}$$

associated with the finite words obtained from $343\bullet$ by adding twice on the left side each of the three possible letters allowed by the grammar rules **G0** and **G1** ($\mathcal{R}_9 = \mathcal{R}(34343\bullet) = \emptyset$ from **G1**). The corresponding sequences have been denoted by s_i . Clearly, none of the s_i is canonical since they all end with a vertex sequence. However, in Fig. 10 the “nearly vertical” boundaries of the \mathcal{R}_i ’s have been constructed as, from left to right:

- pieces of $T^2((C^{\text{ex}})_2)$, $T^3((C^{\text{ex}})_1)$, and $T^3((C^{\text{ex}})_4)$, for the thick lines ;
- pieces of $T^4((C^{\text{ex}})_3)$, $T^4((C^{\text{ex}})_4)$, $T^4((C^{\text{ex}})_2)$, $T^4((C^{\text{ex}})_3)$, and $T^4((C^{\text{ex}})_1)$, for the thin ones.

Moreover, the parts of C_4 , C_1 , and C_2 , intersecting the \mathcal{R}_i are entry curves. Since here, from the definition of a region of canonical shape, a sequence $s_i k$ obtained by addition of one letter on the right of a s_i is canonical if the corresponding region $\mathcal{R}(s_i \bullet k)$ does not touch C_3 , one sees that:

- $s_1 1$, $s_1 2$, $s_2 1$, $s_2 2$, $s_3 1$, $s_3 2$, $s_4 2$, $s_5 2$, and $s_6 2$ are canonical sequences.
- $s_5 4$, $s_6 4$, and $s_8 3$, are excluding sequences. $s_7 4$ and $s_8 4$ are also so, as a direct consequence of **G1**.

The vertex sequence s has been decomposed into 24 (–2) daughter sequences among which less than one half remain in the “unknown” category. Using the different symmetries of the square billiard: time reversal, $\pi/4$ rotation (which amounts to replace each letter k by $k+1$), and symmetry with respect to the main diagonal (which amounts to interchange sides 1 and 2, and sides 3 and 4), one easily checks that in fact all possible sequences of length six have been investigated.

We have thus, without much effort, significantly refined the grammar of the square billiard. To define one possible quantitative measure of this refinement, let us introduce the subsets $(\mathcal{W}^i)_M$ (included) and $(\mathcal{W}^e)_M$ (excluded) of \mathcal{W} , with $(\mathcal{W}^i)_M \cap (\mathcal{W}^e)_M = \emptyset$, defined as the sets of codes respectively excluded by the exclusion rules of length M and included by the inclusion rules of length M . Clearly, $(\mathcal{W}^i)_M$ and $(\mathcal{W}^e)_M$ are increasing sequences of \mathcal{W} ’s subset as M increases, so that the larger M , the better $\overset{\circ}{\mathcal{W}}$ is specified:

$$\dots \subset (\mathcal{W}^i)_M \subset (\mathcal{W}^i)_{M-1} \subset \dots \subset \overset{\circ}{\mathcal{W}} \subset \dots \subset \overline{(\mathcal{W}^e)_{M'+1}} \subset \overline{(\mathcal{W}^e)_{M'}} \subset \dots \subset \mathcal{W},$$

where $\overline{(\mathcal{W}^e)_M}$ denotes the complementary set of $(\mathcal{W}^e)_M$ in \mathcal{W} . The sets $(\mathcal{W}^i)_M$ and $\overline{(\mathcal{W}^e)_M}$ have a Cantor-like structure and are therefore zero-measure subsets of \mathcal{W} (as therefore is $\overset{\circ}{\mathcal{W}}$). However their relative Hausdorff dimension $d((\mathcal{W}^i)_M)$ and $d(\overline{(\mathcal{W}^e)_M})$ can be easily calculated. For exclusion rules for instance, let us denote by $\lambda_M^e (< N-1)$ the rate of increase of the number of finite codes containing no exclusion rule of length smaller than M , i.e.

$$\begin{aligned}
\lambda_M^e &= \lim_{n \text{ or } p \rightarrow \infty} \frac{\# \text{ of } w_{n(p+1)} \text{ with no exclusion rule } \geq M}{\# \text{ of } w_{np} \text{ with no exclusion rule } \geq M} \\
&= \lim_{n \text{ or } p \rightarrow \infty} \frac{\# \text{ of } w_{(n+1)p} \text{ with no exclusion rule } \geq M}{\# \text{ of } w_{np} \text{ with no exclusion rule } \geq M}.
\end{aligned}$$

The relative dimension $d(\overline{(\mathcal{W}^e)_M})$ is defined as

$$d((\overline{\mathcal{W}^e})_M) = \frac{\log \lambda_M^e}{\log(N-1)}$$

(here $(N-1) = 3$). The rate of increase λ_M^i (resp. $d((\mathcal{W}^i)_M)$) associated to inclusion rules, defined similarly, and λ_M^e (resp. $d((\mathcal{W}^e)_M)$) are respectively lower and upper bounds of the analogue quantity $\overset{o}{\lambda}$ (resp. $d(\overset{o}{\mathcal{W}})$) defined for $\overset{o}{\mathcal{W}}$. Moreover

$$0 \leq \dots \leq d((\mathcal{W}^i)_M) \leq d((\mathcal{W}^i)_{M+1}) \leq \dots \leq d(\overset{o}{\mathcal{W}}) \leq \dots \\ \leq d((\overline{\mathcal{W}^e})_{M'+1}) \leq d((\overline{\mathcal{W}^e})_{M'}) \leq \dots \leq d(\mathcal{W}) = 1.$$

Therefore, $d((\overline{\mathcal{W}^e})_M) - d((\mathcal{W}^i)_M)$ may serve as a good measure of how well $\overset{o}{\mathcal{W}}$ is specified. The progress made in the simple case treated above is summarized in Table 1.

6. The symbol plane. Automatic search for the grammar rules

When dealing with non-exact coding, a great amount of insight into the structure of grammar rules has been obtained by Cvitanović et al. [25] by introducing the notion of “pruning front”. The goal of this section is to make a connection between this latter and the approximate grammar rules introduced here in Section 4.

The pruning front is defined in a plane of symbols (p, f) where the “past” p and “future” f are two real numbers whose decimal developments (here in basis $N-1$) are codes of a trajectory in the Poincaré surface of section in a way similar to the past and future parts of a doubly infinite word $w = k_{-\infty} \dots k_0 \bullet k_1 \dots k_{\infty}$. They are moreover required to be “topologically well ordered” [23], which here means that wherever it is defined, $\varphi_i(p, f)$ (resp. $\varphi_f(p, f)$) is an increasing function of p (resp. f). In the same way as in Section 6 of Ref. I, it can be seen that this requirement is fulfilled here if p and f are defined from $w = k_{-\infty} \dots k_0 \bullet k_1 \dots k_{\infty}$ by fixing the integer part of p and f respectively to $k_0 - 1$ and 0, and by writing their decimal part in base $N-1$ (remind that N is the number of letters of the original alphabet) as

$$p - (k_0 - 1) = .p_1 p_2 p_3 \dots \\ f = .f_1 f_2 f_3 \dots$$

with

$$p_n = \begin{cases} \delta k - 1 & \text{if } n \text{ even} \\ (N - \delta k) - 1 & \text{if } n \text{ odd} \end{cases} \quad (\delta k = (k_{-n} - k_{-(n-1)}) \bmod (N-1)) \\ f_n = \begin{cases} (N - \delta k) - 1 & \text{if } n \text{ even} \\ \delta k - 1 & \text{if } n \text{ odd} \end{cases} \quad (\delta k = (k_n - k_{(n-1)}) \bmod (N-1)).$$

In the symbol plane, the bounce mapping is no more a shift, as it was for the words w , but takes the relatively simple form

$$(p, f) \longrightarrow (p', f') \\ f' = 1 - (N-1)f \bmod 1, \\ p' = \text{Int}(p') + \frac{\text{Int}((N-1)f) + (1-p)}{N-1}, \\ \text{with } \text{Int}(p') = \text{Int}(p) + \text{Int}((N-1)f) \bmod (N-1). \quad (6.1)$$

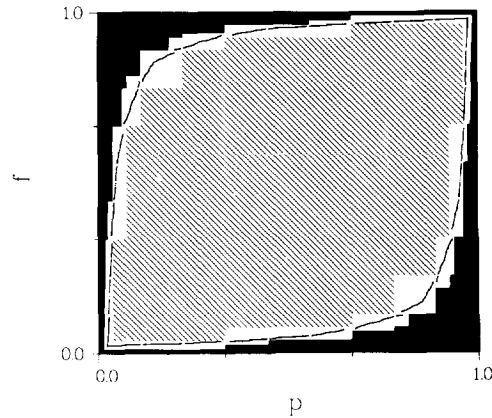


Fig. 11. Representation in the symbol plane of the pruning front and of its fattened version corresponding to grammar rule of length 6, for the square billiard with angles at vertex $\pi/3.6$. The dark region corresponds to the forbidden zone, the shaded one to authorized zone, and the white part to the unknown region. Because of the symmetries of the billiard, the symbol plane is unchanged when p (i.e. the abscissa) is translated from one unity. Therefore only the range $p \in [0, 1]$ is displayed, which correspond to the entry domain of side 1.

To avoid introducing new notations, and since there is no risk of ambiguity, we shall keep the notation T for the bounce mapping in the symbol plane ($T(p, f) = (p', f')$), and call a trajectory in the symbol plane the set of iterates to the future and to the past of some point (p, f) .

Noting \mathcal{S} the mapping which, to any point (φ_i, M) (or (φ_i, φ_f)), associates its symbol (p, f) , the “pruning front” is the image by \mathcal{S} of the boundaries of the exit regions $\mathcal{R}(k_0 \bullet)$ ($k_0 = 1, \dots, N$), i.e. of the basic partition used to define the coding. These boundaries are the parts $(C^0)_{k_0}$ and $(C^{\text{ex}})_{k_0}$ of the curves C_{k_0} . Since \mathcal{S} maps the authorized part of the Poincaré section onto a *fractal* subset of the symbol plane, *the pruning front has also a fractal structure*.

The very appealing nature of the concept of pruning front appears immediately when looking at Fig. 11 in which it is represented for the square billiard with angles $\pi/3.6$ studied in the previous section. It is intended to separates an “outside” disallowed region from an “inside” one such that symbols (p, f) whose all images by T^n ($n = -\infty \dots +\infty$) remain in it are “most probably” associated to an actual trajectory in the Poincaré surface of section. This is particularly interesting when investigating closed trajectories, since it suffices to look at the position, relative to the pruning front, of the (finitely many for a given period) periodic points of the Baker-like transformation Eq. (6.1) to decide which ones have a “good chance” to correspond to a true closed orbit, and which ones clearly do not.

As we understand it however, the main drawback of the pruning front (in the way it is defined for a billiard system) comes from its fractal nature, which prevents to define unambiguously an internal and an external part.⁷ With the finite resolution of Fig. 11, one can see some “macroscopic” holes, but holes evidently are present at any scale. Each of these holes occurs from the intersection of the pruning front with one of its iterates; the highest is the degree of the iteration, the smallest is the size of the hole. If a symbol (p, f) has an iterate by Eq. (6.1) “rather” inside the pruning front, but “near” one of its holes, it may be impossible to give a clear-cut answer to the question of existence of an associated trajectory. In full rigor, such considerations should also apply for symbols whose iterates remain “well” inside the pruning front, although this may be of less importance from a practical point of view.

Actually, for the pruning front to allow an unambiguous separation between allowed and forbidden trajectories would require to know it perfectly well, or equivalently to know the future and the past of the curves C_k for

⁷ This remark makes clearer the meaning of the expressions put in quotation marks in the previous paragraph.

infinite time. The procedure we suggest here consists in constructing instead what we may call a “fat” pruning front, such as the one displayed in Fig. 11 using the grammar rule of length 6 derived in the previous section. Given the list of the canonical, forbidden, and unknown sequences for a given length of the grammar, it is constructed by the following successive steps (using, as in Fig. 11, the convention that what is forbidden is shaded, what is unknown is in dark and what is authorized in white):

- 0th step (*once for all*): turn to dark the rectangles corresponding to a vertex sequence surrounding the present time. Then, at each increase of the length of the grammar:
- 1st step: shade the rectangles corresponding to a forbidden sequence.
- 2nd step: turn to white the dark parts of the rectangles corresponding to a canonical sequence, but only if the vertex sequence surrounding the present time does not touch the extremities of the sequence.

With this procedure, symbols whose iterates always remain in the white zone are such that any vertex sequence is dressed by a canonical sequence. Thus, following the proposition stated in Section 4, each of them is actually associated to a trajectory in the Poincaré surface of section.

One obtains in this way a kind of *coarse-grained pruning front*, much rougher than the actual one for short grammar rules. In counterpart one gains that an allowed region is *unambiguously* defined, although only a finite number of iterations of the curves C_k are needed. When looking for closed orbits for instance, placing the n iterates of some periodic point of period n of the mapping Eq. (6.1) in the figures gives a definite answer about the existence of an actual associated closed orbit, either if at least one of these iterates enters the forbidden region, or if all of them remain in the (white) authorized region. Of course, the longer the period, the larger the probability to enter the unknown zone. It will therefore be necessary to be able to refine at will the grammar, which supposes an algorithmic (practicable on a computer) procedure.

Without entering into details, one can check that the following procedure allows to determine whether a sequence of length n , say, $s = k_{-(n-1)} \dots k_0$ is canonical, forbidden, or unknown. The first step consists in constructing the part $\mathcal{S}((C^{\text{ex}})_k)$ ($k = 1, \dots, N$) of the pruning front corresponding to the $(C^{\text{ex}})_k$ with a precision of n digits for both p and f . This amounts to map n times the curves $(C^{\text{ex}})_k$ to the past and to the future. Second, consider the rectangle $[p, p + (N-1)^{n-1}] \times [0, 1]$ in the symbol plane associated to the code to the past $k_{-(n-1)} \dots k_0 \bullet$, and the n first iterates of the $\mathcal{S}((C^{\text{ex}})_k)$. Then:

- if none of the iterates intersects the rectangle, the sequence s is (obviously) forbidden.
- if either s starts or ends with a vertex sequence, or if there is a vertex k such than an iterate $T^j \mathcal{S}((C^{\text{ex}})_k)$ intersects the rectangle, with $0 < j < n - \nu_k$ (ν_k is defined by Eq. (3.4)), then s is an unknown sequence.
- in all other cases, s is canonical.

Going on with the square billiard with angles $\pi/3.6$, we have applied this algorithm up to grammar of length 9 yielding the fat pruning fronts displayed in Fig. 12. It is interesting to compare the results we obtain in this way with what would give the application, on the billiards under study, of the prescription given in [23] to construct a set of approximate grammar rules of a given length from the knowledge of the pruning front. Translated in our language, this prescription consists in considering as “unknown” only the sequences such that the corresponding rectangles, in the symbol plane, contain a point of the pruning front. As far as only one vertex sequence can fit inside the considered sequence (here for grammar rules of length smaller than seven), this would give precisely the same result as shown in Fig. 12. However, for grammar rules of length eight and nine, one observes in Fig. 12 small tendrils, originating from the neighborhood of the gaps of the pruning front, which cannot be reproduced if only the pruning front is considered. Indeed, these tendrils correspond to sequences containing two vertex sequences, and result from the fact the pruning front intersects one of its iterates. Therefore, the slightly more complicated prescription arising from our approach appears as necessary to avoid putting some sequences in the wrong classes. In other words, from the approximate grammar rules derived in Section 4 a clear-cut implementation of the notion of pruning front can be obtained for the billiards

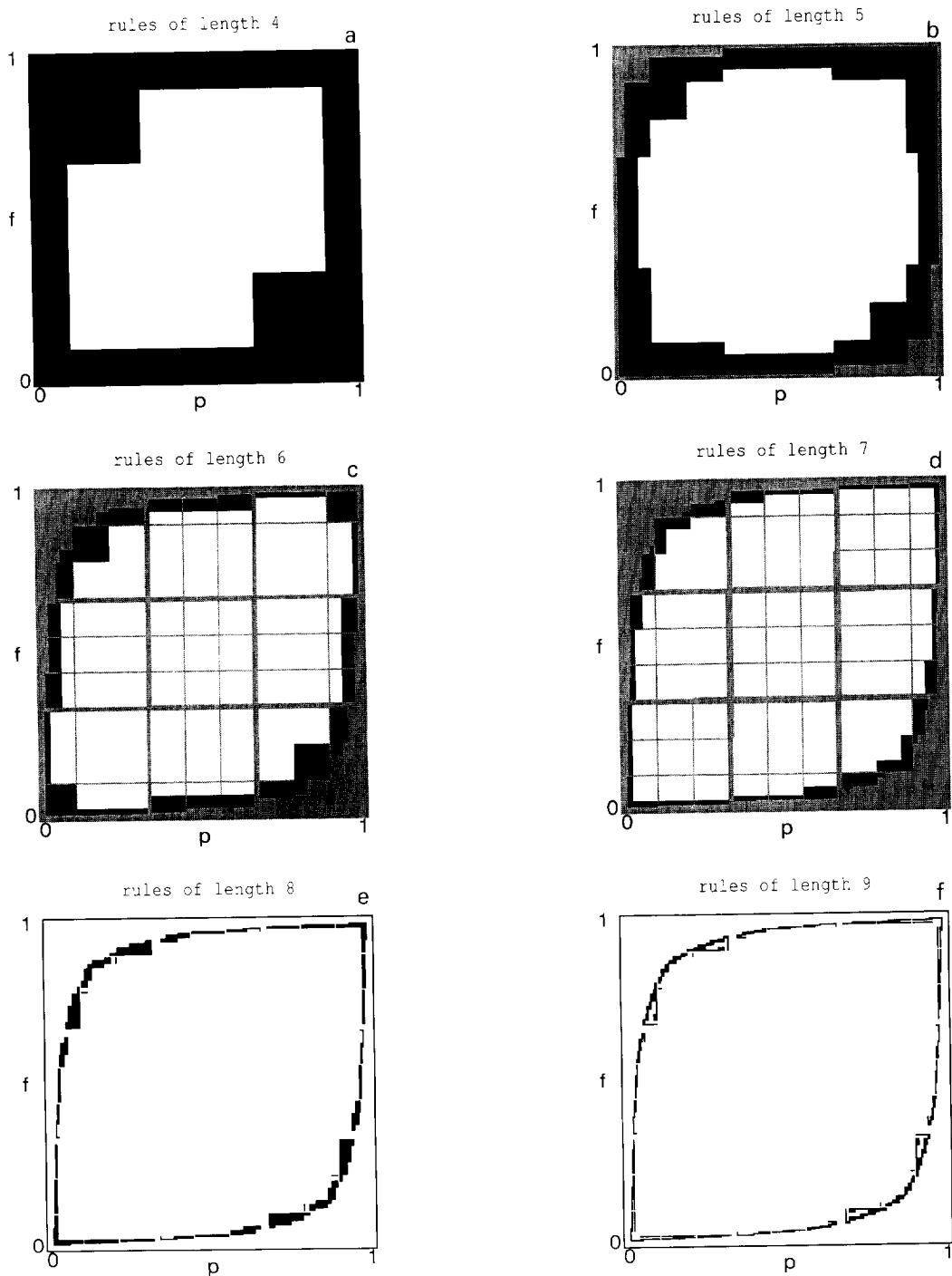


Fig. 12. Evolution of the fat pruning front when the length of the grammar goes from four to nine, for the square billiard with angles at vertex $\pi/3.6$. The authorized regions are in white, the forbidden one in grey, and the unknown in dark. For sake of legibility, only the unknown region has been represented for grammar rules of length eight and nine. As in Fig. 11, only the range $p \in [0, 1]$, corresponding to the entry domain of side 1, is represented.

studied here. It is seen however to require a more involved analysis than usually assumed.

7. Limit of infinite length for the grammar rules

In Fig. 12, it is seen that the area occupied by the fat pruning front (i.e. the area of the “unknown” region) diminishes very rapidly as the length of the grammar rules increases. However, in the same time, the relative area of the forbidden zone is clearly increasing to one, which does not leave much room for the fat pruning front. To claim that one can describe the set of actual trajectories “as well as one wishes provided that sufficient effort is made”, the relevant quantity is the ratio

$$r_n = \frac{\text{Area(fat pruning front)}}{\text{Area(authorized region)}} ,$$

for grammar rule of length n , and its limit r_∞ when n goes to ∞ . The aim of this section is to derive a simple criterion (sufficient condition), fulfilled by most of the compact billiards, under which r_∞ is indeed equal to zero.

What can be actually shown is that, as soon as the number of non-forbidden finite codes grows sufficiently fast with their length, the proportion (among the non-forbidden ones) of unknown sequences of length n can be bounded from above for large n by some number F , strictly smaller than one. This implies $r_\infty = 0$ since, starting from the set of symbols such that the associated code contains a vertex sequence s contiguous with or containing the present time, we have seen that the fat pruning front is constructed by removing all symbols for which s can be dressed by a canonical sequence. Therefore, assuming some rather uniform distribution of the location, in the symbol plane, of the rectangles associated to canonical sequences, - an hypothesis which is natural because of the hyperbolicity of the dynamical system - r_n will roughly be diminished by a factor F at each increase of n .

Let $N_u(n)$ and $N_{\text{tot}}(n)$ be respectively the number of unknown and non-forbidden (i.e. unknown + canonical) sequences of length n . As shown in the Appendix⁸, an upper bound for the ratio $N_u(n)/N_{\text{tot}}(n)$ can actually be obtained in terms of ν and of λ_{inf} , where ν is the smallest among the ν_k 's (see Eq. 3.4) and λ_{inf} is any lower bound of the rate of increase λ introduced at the end of Section 5. One has indeed, for sufficiently large n

$$\frac{N_u(n)}{N_{\text{tot}}(n)} \leq F(\lambda_{\text{inf}}, \nu) = \frac{2}{G(\lambda_{\text{inf}}, \nu)^{-1} - 1} ,$$

$$G(\lambda_{\text{inf}}, \nu) = \nu \lambda_{\text{inf}}^{-(\nu-1)} - (\nu - 1) \lambda_{\text{inf}}^{-\nu} . \quad (7.1)$$

When $G(\lambda_{\text{inf}}, \nu)$ is less than $1/3$, $F(\lambda_{\text{inf}}, \nu)$ is smaller than 1, which is a sufficient condition for the “fat pruning front” to reduce faster in size than the authorized region. For a given billiard, a lower bound λ_{inf} of λ can be obtained from any set of approximate grammar rules in the same way as we did to construct the second column of Table 1 for the square billiard with angle $\pi/3.6$. For this particular example for instance, $\nu = 3$, and the lower bound λ_{inf} given by the grammar of length 3 (see Table 1) is $\lambda_{\text{inf}} = 3^{.802} \simeq 2.41$. The corresponding value of $F(\lambda_{\text{inf}}, \nu)$ is slightly greater than one ($F(2.41, 3) \simeq 1.19$). It is therefore necessary to use the bound given by the grammar of length 6 ($\lambda_{\text{inf}} = 3^{.915} \simeq 2.73$) which gives $F(\lambda_{\text{inf}}, \nu) \simeq 0.87$, thus ensuring the convergence of the grammar.

⁸ To avoid the inclusion of a rather technical derivation in the main text, we have released this demonstration in the Appendix. We stress though that the derivation by itself may have its own interest, and may therefore worth to be looked at.

Table 1

Evolution of the relative Hausdorff dimension d of the sets $(\overline{(\mathcal{W}^e)_M})$ and $(\mathcal{W}^i)_M$ when increasing the length M of the grammar rule from zero to six, for the square billiard with vertex angles equal to $\pi/3.6$. The difference $d(\overline{(\mathcal{W}^e)_M}) - d(\mathcal{W}^i)_M$ may be used as a measure of how accurately the set of authorized codes is defined

	$d(\overline{(\mathcal{W}^e)_M})$	$d(\mathcal{W}^i)_M$	$d(\overline{(\mathcal{W}^e)_M}) - d(\mathcal{W}^i)_M$
G0 alone	1	0	1
grammar of length 3 (i.e.: only G0, G1, and G'1)	.984	.802	.182
grammar of length 6	.969	.915	.054

As a final remark, we stress that although in the particular example treated above F is rather close to one, the convergence of the grammar should hold quite generally. This is the case first because Eq. (7.1) it is a rather rough upper bound for $N_u(n)/N_{\text{tot}}(n)$, which can be easily refined if ever needed. In particular the constraint (ii) of the appendix is given in its simplest form. Also, $G(\lambda_{\text{inf}}, \nu)$ (and thus $F(\lambda_{\text{inf}}, \nu)$) is a decreasing function of both λ_{inf} (in the sense that $\partial G(\lambda_{\text{inf}}, \nu)/\partial \lambda_{\text{inf}}$ is negative for λ_{inf} greater than one, which is consistent with the fact that one can express F in terms of lower bounds of λ), and ν (in the sense that $G(\lambda_{\text{inf}}, \nu + 1) - G(\lambda_{\text{inf}}, \nu)$ is negative). Since F is already smaller than one for the billiard we consider, it will generally be so if one increases either the number of sides of the billiard, or diminishes the vertex angles (which increases ν). Most presumably, the simple grammar rules **G1** and **G'1**, obtained without doing any numerical computations, will usually suffice to insure the convergence of the grammar.

8. Conclusion

In this paper, we have investigated some properties of the “à la Morse” coding of polygonal compact billiards on constant negative curvature manifold. In the non-generic case of tiling billiards we show, as in Ref. I, that the coding is exact. On the contrary, and as for most bounded Hamiltonian systems, the coding is not exact for generic (non-tiling) polygons. It is possible however to construct two sets of approximate grammar rules. Each set can be obtained for arbitrary lengths of the grammar (except for computing time limitation) with a computer program. The first one consists in exclusion rules, which specify forbidden words. It is constructed in a simple way, analogue to the customary method used for other systems possessing a “good” code. The second set specifies authorized words. Its construction is less obvious, and the simplicity of the systems worked on (polygons) certainly greatly facilitates their derivation. (A generalization, even at the price of some complications, for other systems for which non-exact coding exists may be worthwhile.) In between remain “unknown” words for which it is not possible to decide whether they are allowed or forbidden. A few billiards may be excepted, their relative amount can be reduced arbitrarily by increasing the length of the grammar rules. In a plane of symbols, one can define a pruning front, analogous to the ones introduced by Cvitanović et al. [25] and Hansen [23]. However, the Cantor-like structure of this “exact” pruning front prohibits a rigorous criterion for sorting allowed and forbidden words. This weakness is overcome here by the construction, from the set of approximate grammar rules of a given length, of a “fattened” pruning front, which covers the original one. Such a fat pruning front corresponds to a region of unknown codes, but on the other hand unambiguously separates the forbidden region from the allowed one. As for the proportion of unknown codes, the relative area of the fat pruning front can be reduced arbitrarily as the grammar is made converging.

It is interesting to compare the properties of the symbolic dynamics derived from the Morse-like coding studied here to those obtained using Markov partitions. In the former case, the partition of the Poincaré section used to define the symbolic dynamics is a premise of the problem. It is chosen with a finite number of regions:

the entry (or equivalently exit) domains of the N sides of the billiard. Moreover the partition is intimately related to the dynamics since the entry (respectively exit) domains are the connected components of the point where the bounce mapping T (respectively T^{-1}) is continuous. In the latter case, the problem is tackled from the opposite side since working with Markov partition is essentially a way to impose the exactness of the coding, but the partition remains to be constructed. There exist highly non-generic systems, such as the tiling billiards (compact [cf. Section 3.2] or non-compact [I]) where in fact the partition used in the Morse coding happens to be essentially a finite Markov partition. In such a case, the problem is simple whatever may be the side from which it is tackled, and one ends up with a exact coding, i.e. a finite number of grammar rules, and a generating partition which has also a finite number of regions. In more generic cases, and in particular for the non-tiling polygonal billiards considered here one either imposes the exactness of the coding (Markov partitions) and ends up with an *infinite* partition, or impose that the partition is simple (Morse coding), and ends up with a non-Markovian symbolic dynamics. Clearly, when considering problems for which the exactness of the coding is fundamental, but the explicit construction of the partition is not necessary (for instance to prove the B -character of the dynamics), Markov partitions have obviously to be preferred (see e.g. the contributions of Pesin and Bunimovitch in [31] for a recent tutorial review). However, since for the Morse coding we have obtained a constructive way to obtain the grammar, this latter should be useful when explicit construction of the partition and the grammar rules are needed. This may be the case in particular when calculating a rate of decay of correlations within a thermodynamical formalism (see e.g. [17] for an introduction). Indeed such an approach involves classical trace formulae, in which an explicit enumeration of the periodic orbits is needed. The kind of coding studied here should provide precise upper and lower bounds of such rates of decay.

Concerning the quantum mechanics of classically chaotic systems, coding is probably a “passage obligé” in the application of the Gutzwiller trace formula or other methods derived from it as far as one pretends to compute a great number of energy levels with a high resolution. From this point of view, non-tiling compact polygonal billiards of constant negative curvature certainly constitute a good compromise between genericity and simplicity. On one hand, they share with most Hamiltonian bounded systems the property that no exact coding exists, and that moreover the Gutzwiller trace formula is a semiclassical approximation. (We recall that, on the contrary, tiling billiards reveal their non-generic character via the existence of exact coding and exact quantization procedure, the Gutzwiller trace formula happening in this case to coincide with the Selberg trace formula.) One can “experiment” on them the effects due to the finite size of \hbar as well as those due to the non-exactness of the coding. On the other hand, they still remain extremely tractable from a computational point a view. This is the case, to start with, because of the above mentioned properties of the coding, i.e. that a set of approximate grammar rule, which converges to an exact description of the trajectories, can be obtained automatically thanks to a computer program. In addition, one keeps the property of tiling billiard associated to Fuchsian groups that, once the code associated to a closed trajectory is known, all ingredients entering into the Gutzwiller trace formula (length of the orbit and linear stability) are also available. Therefore, knowing the code of an orbit, no further calculation is needed (except for two by two matrix multiplications) to obtain its contribution to the semiclassical density of states. Such billiards are certainly among the best models to look at when studying the quantization of chaotic systems.

Acknowledgments

It is a pleasure to thank O. Bohigas, P. Cvitanović, M.C. Gutzwiller, K.T. Hansen, C. Schmit, for stimulating discussions. We are also indebted to F. Leyvraz for the interest he showed in the latest stage of this work.

Appendix. Calculation of an upper bound for the ratio $N_u(n)/N_{\text{tot}}(n)$

Denoting respectively by $N_u(n)$ and $N_{\text{tot}}(n)$ the number of unknown and non-forbidden (i.e. unknown + canonical) sequences of length n , we calculate in this appendix an upper bound for the $n \rightarrow \infty$ limit of the ratio $N_u(n)/N_{\text{tot}}(n)$. The principle of the derivation relies on the fact that such an upper bound can be obtained by considering the intersections of the iterates by the bounce mapping of curves $(C^{\text{ex}})_k$ and $(C^{\text{en}})_{k'}$. Let us therefore introduce \mathcal{E}_j the set of all intersection points of a curve $(C^{\text{en}})_{k'}$ with some $T^j(C^{\text{ex}})_k$ ($k, k' = 1, 2, \dots, N$). Consider moreover a point M_j of \mathcal{E}_j , with its (doubly infinite) code,⁹ and

$$w_n = k_1 \cdots k_p \bullet k_{p+1} \cdots k_n$$

a finite subcode of w of total length n and length to the past p . The code w therefore can be written schematically as:

$$w = \cdots \overbrace{\left(\begin{array}{c} \text{vertex} \\ \text{sequence} \\ \text{at } k \end{array} \right) \cdots \cdots \cdots \bullet \cdots \cdots \left(\begin{array}{c} \text{vertex} \\ \text{sequence} \\ \text{at } k' \end{array} \right)}^{\text{word } w_n} \cdots \quad (\text{H.1})$$

If $j < p < n$, then w_n contains at least one letter of both vertex sequences at vertices k and k' . In that case, M_j lies on the boundary of $\mathcal{R}(w_n)$, and the pieces of $(C^{\text{en}})_{k'}$ and $T^j(C^{\text{ex}})_k$ intersecting at M_j are parts of the boundary of $\mathcal{R}(w_n)$. We shall then say that M_j is an *apex* (of order j) of the region $\mathcal{R}(w_n)$. Roughly speaking, we shall also refer to M_j as a j -apex of the sequence $s_n = k_1 \cdots k_n$. The condition $j < p < n$ implies that, for a fixed value of n , M_j is a j -apex of exactly

$$\theta(j) = [n - (j + 1)] \quad (\text{H.2})$$

sequences of length n .

Let us then consider the set of all sequences of length n . Simple constraints on the number of apices of a given order each sequence should have will allow us to obtain an upper bound for $N_u(n)$ and a lower bound for $N_{\text{tot}}(n)$, and consequently an upper bound for their ratio. Indeed, one may easily be convinced that, noting ν the smallest among the ν_k 's:

(i) For a given sequence s_n of length n , the total number of j -apices such that

$$n - \nu \leq j \leq n - 1 \quad (\text{H.3})$$

is equal or smaller than four.

This is due to the fact that a region \mathcal{R}_n associated to a code of length n has at most four (two for each - past and future - extremities of the word) boundaries associated to vertex sequences which are not interior to the word (the so called boundary of type (ii) in Section 4). If all four of them are present, their four intersections may fulfil the above condition. If one is missing, or if a boundary of type (i) (associated to a vertex sequence interior to s_n) takes place between two of them, the number of j -apices fulfilling Eq. (H.3) is less than four.

⁹ M_j being associated to a trajectory which hits both vertex k and vertex k' , at which the motion is discontinuous, it can in fact be associated to four codes. Choosing one of them amounts to specify from which side the vertices k and k' are approached.

(ii) An unknown sequence s_n of length n contains at least two j -apices fulfilling

$$j < n - \nu. \quad (\text{H.4})$$

Indeed, if a region \mathcal{R}_n associated to a non-forbidden code of length n is not of canonical shape (i.e. if the corresponding sequence s_n is “unknown”), its boundary has to include an iterate of a piece of curve $(C^{\text{en}})_k$ corresponding to a vertex sequence at vertex k interior to s_n . The two extremities of this piece are j -apices. A necessary (not sufficient) condition for the vertex sequence at vertex k to fit into s_n is that they both fulfil Eq. (H.4).

The end of the argument comes from the fact that for large j , $c(j)$, the number of points in \mathcal{E}_j , clearly behaves as the number of non-forbidden codes of length j , i.e.:

$$\lim_{j \rightarrow \infty} \frac{c(j+1)}{c(j)} = \overset{o}{\lambda},$$

where the rate of increase $\overset{o}{\lambda}$ is the same as the one introduced at the end of Section 5 for \mathcal{W} .¹⁰ In particular, lower bounds λ_{inf} such as those obtained from the second column of Table 1 for the square billiard with angle $\pi/3.6$ are available. For any such lower bound λ_{inf} of $\overset{o}{\lambda}$ one has after some rank K (that we shall suppose equal to zero) $c(j+l) \geq \lambda_{\text{inf}}^l c(j)$ if $l \geq 0$, and $c(j+l) \leq \lambda_{\text{inf}}^l c(j)$ if $l \leq 0$. Thus, together with Eq. (H.2), item (i) provides a lower bound for $N_{\text{tot}}(n)$

$$N_{\text{tot}}(n) \geq \frac{1}{4} \left[\sum_{j=n-\nu}^{n-1} \theta(j) c(j) \right] \geq \frac{c(n-(\nu+1))}{4} \sum_{\theta=0}^{\nu-1} \theta \lambda_{\text{inf}}^{\nu-\theta} \quad (\text{H.5})$$

and item (ii) an upper bound for $N_u(n)$

$$N_u(n) \leq \frac{1}{2} \left[\sum_{j=0}^{n-(\nu+1)} \theta(j) c(j) \right] \leq \frac{c(n-(\nu+1))}{2} \sum_{\theta=\nu}^{\infty} \theta \lambda_{\text{inf}}^{\nu-\theta}. \quad (\text{H.6})$$

After some short algebra, this leads to Eq. (7.1), i.e.:

$$\frac{N_u(n)}{N_{\text{tot}}(n)} \leq F(\lambda_{\text{inf}}, \nu) = \frac{2}{G(\lambda_{\text{inf}}, \nu)^{-1} - 1}, \quad (\text{H.7})$$

where

$$G(\lambda_{\text{inf}}, \nu) = \nu \lambda_{\text{inf}}^{-(\nu-1)} - (\nu-1) \lambda_{\text{inf}}^{-\nu}. \quad (\text{H.8})$$

When $G(\lambda_{\text{inf}}, \nu)$ is less than $1/3$, $F(\lambda_{\text{inf}}, \nu)$ is smaller than 1, which is a sufficient condition for the “fat pruning front” to reduce faster in size than the authorized region. This happens to be so because the proliferation of j -apices is large enough to ensure that, for a given n , a majority of them satisfy $n - \nu \leq j \leq n - 1$. This guarantees that a non-zero proportion of sequences s_n of length n admit only this kind of apices, and thus are canonical.

¹⁰ In case a justification is worth, notice that if, in the first inequality of Eq. (H.6), one pushes the upper bound of the summation to $j = n - 1$, one obtains a lower bound of $N_{\text{tot}}(n)$ instead of $N_u(n)$. Admitting that

$$\lim_{j \rightarrow \infty} \frac{c(j+1)}{c(j)} = f,$$

this lower bound together with the upper bound Eq. (H.5) readily yields $f = \overset{o}{\lambda}$.

References

- [1] See e.g. L.A. Bunimovitch and Ya.G. Sinai, *Commun. Math. Phys.* 78 (1980) 247, and references therein.
- [2] D.S. Ornstein and B. Weiss, *Israel J. Math.* 14 (1973) 184.
- [3] M. Morse, *Trans. Amer. Math. Soc.* 22 (1921) 33; and in: *Symbolic Dynamics*, Institute for Advanced Study Notes (Notes by Rufus Oldenberger from unpublished notes of Morse, 1938) (Princeton, 1966).
- [4] M.C. Gutzwiller, *Chaos in Classical and Quantum Mechanics* (Springer Verlag, New York, 1990).
- [5] R. Balian and C. Bloch, *Ann. Phys.* 69 (1972) 76.
- [6] CHAOS 2 (1992) 1, Chaos Focus Issue on Periodic Orbit Theory.
- [7] See M.C. Gutzwiller, *Physica D* 5 (1982) 183, and references therein.
- [8] M.C. Gutzwiller, *Phys. Rev. Lett.* 45 (1980) 150–153.
- [9] R. Aurich, M. Sieber and F. Steiner, *Phys. Rev. Lett.* 61 (1988) 483–486.
- [10] C. Schmit, Quantum and classical properties of some billiards on the hyperbolic plane, in: M.-J. Giannoni, A. Voros and J. Zinn-Justin (eds.), *Chaos and Quantum Physics*, Proceedings, Les Houches Summer School 1989 (Amsterdam, North-Holland, 1991) pp. 333–370; E. Bogomolny, C. Schmit, *Nonlinearity* 6 (1993) 523–547.
- [11] E. Bogomolny, B. Georgeot, M.-J. Giannoni and C. Schmit, *Phys. Rev. Lett.* 69 (1992) 1477–1478; *Phys. Rev. E* 47 (1993) R2217–R2220.
- [12] B. Georgeot, thèse de doctorat de l'Université Paris 11 (October 1993); E. Bogomolny, B. Georgeot, M.-J. Giannoni and C. Schmit, preprint IPNO/TH 93-51.
- [13] K. Richter, Ph.D. Thesis, University of Freiburg (1991); G.S. Ezra, K. Richter, G. Tanner and D. Wintgen, *J. Phys. B* 24 (1991) L413–L420; D. Wintgen, K. Richter and G. Tanner, *CHAOS* 2 (1992) 19–32.
- [14] M. Sieber, Ph.D. Thesis, University of Hamburg (1991); M. Sieber and F. Steiner, *Physica D* 44 (1990) 248–266; *Phys. Lett. A* 148 (1990) 415–419; *CHAOS* 2 (1992) 35–42.
- [15] B. Eckhardt, *J. Phys. A* 20 (1987) 5971–5979.
- [16] R. Artuso, E. Aurell and P. Cvitanović, *Nonlinearity* 3 (1990) 325–359, 361–386.
- [17] B. Eckhardt and S. Grossmann, Correlation functions in chaotic systems from periodic orbits, preprint.
- [18] M.-J. Giannoni and D. Ullmo, *Physica D* 41 (1990) 371–390.
- [19] V. Donnay and C. Liverani, *Commun. Math. Phys.* 135 (1991) 267–302.
- [20] G. Troll, *Physica D* 50 (1991) 276–296; U. Smilansky, The classical and quantum theory of scattering in: M.-J. Giannoni, A. Voros and J. Zinn-Justin (eds.), *Chaos and Quantum Physics*, Proceedings, Les Houches Summer School 1989 (Amsterdam, North-Holland, 1991) pp. 333–370.
- [21] R.L. Adler and L. Flatto, Cross section maps for geodesic flow, in: *Ergodic Theory and Dynamical Systems*, Vol. 2, Proceedings, Special Year Maryland 1979–1980, *Progress in Math.* 21 (1982) p. 103.
- [22] C. Series, *Proc. R. Soc. Lond. A* 413 (1987) 171–182; *Ergodic Theory Dynam. Sys.* 6 (1986) 601–625.
- [23] K.T. Hansen, Ph.D. Thesis, University of Oslo, September 1993; see also *CHAOS* 2 (1992) 71; *Nonlinearity* 6 (1993) 753, 771;
- [24] P. Grassberger, H. Kantz and U. Moening, *J. Phys. A* 22 (1989) 5217–5230.
- [25] P. Cvitanović, G.H. Gunaratne and I. Procaccia, *Phys. Rev. A* 38 (1988) 1503–1520.
- [26] K.T. Hansen, *Phys. Lett. A* 165 100 (1992).
- [27] M.-J. Giannoni and D. Ullmo, IPNO/TH 93-62 and 93-63 (unpublished).
- [28] N. Balazs and A. Voros, *Phys. Rep.* 143 (1986) 109.
- [29] H.S.M. Coxeter and W.O.J. Moser, *Generators and Relations for Discrete Groups* (Springer Verlag, 1980) p. 55.
- [30] H. Poincaré, *Théorie des groupes Fuchsien*, *Acta Mathematica* 1 (1882) 1–62.
- [31] *Encyclopaedia of Mathematical Sciences*, Vol. 2, *Dynamical systems II*, Ya.G. Sinai ed. (Springer Verlag, 1989).