

# EL BILLAR DE SINAI CLÁSICO

por

Santiago Andrés Triana<sup>1,2</sup> & Diógenes Campos<sup>1,3</sup>

## Resumen

S. A. Triana & D. Campos: El Billar de Sinai Clásico. Rev. Acad. Colomb. Cienc. 23(86): 61-71. 1999. ISSN 0370-3908.

Desarrollamos aquí un algoritmo para realizar experimentos numéricos con el billar de Sinai en su versión clásica, junto con una breve discusión de la precisión y limitaciones de tal algoritmo. Algunos resultados de estos experimentos se presentan revelando un comportamiento altamente caótico.

**Palabras claves:** Sistemas caóticos, billares, ergodicidad.

## Abstract

An algorithm to perform numerical experiments on the Sinai billiard (classical version) is developed and a brief discussion concerning its accuracy and limitations is included. Some results of these experiments are presented revealing a highly chaotic behaviour.

**Key words:** Chaotic systems, billiards, ergodicity

## 1. Introducción

En los últimos años la investigación de sistemas caóticos ha sido de gran importancia en diferentes áreas de la ciencia y de la técnica (Mullin, 1995). En el caso de la física, los sistemas mecánicos (clásicos y cuánticos) de pocos grados de libertad han sido objeto de estudio, observándose en ellos propiedades y comportamientos dinámicos inesperados (Blümel y Reinhardt, 1997). Algunas propiedades, que se creían exclusivas de sistemas de muchas partículas, por

ejemplo el comportamiento ergódico, han aparecido en sistemas de pocos grados de libertad.

En este artículo estudiamos a la luz de la mecánica clásica el billar de Sinai (Sinai, 1979), el cual es un sistema relativamente simple que presenta un comportamiento fuertemente caótico. El sistema consiste en una partícula que se mueve en el plano dentro de un cuadrado (arista  $L$ ), en cuyo centro se coloca un obstáculo circular (radio  $R$ ), tal como se muestra en la figura 1. Se supone que la partícula (masa  $m$ ) obedece las leyes de la mecánica clásica y sólo experimenta colisiones elásticas contra las paredes del cuadrado y contra el perímetro del obstáculo circular. Esto es, entre colisiones la partícula se mueve de manera libre siguiendo una trayectoria rectilínea.

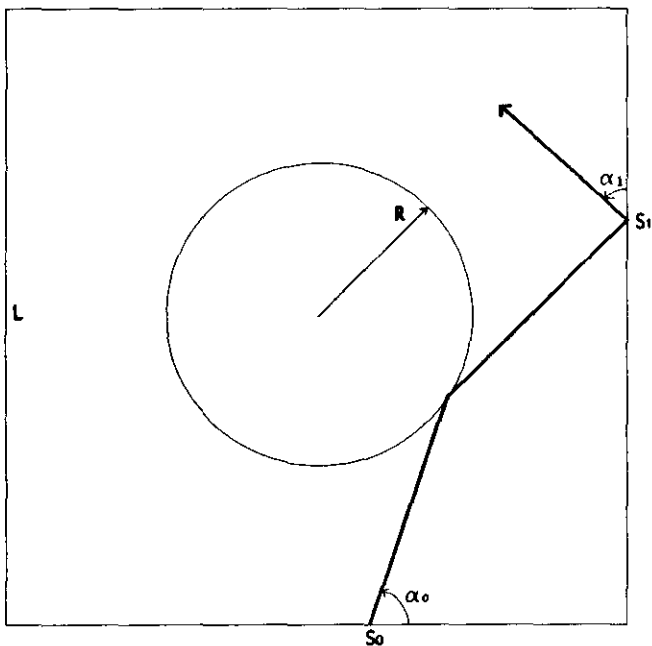
1 Departamento de Física, Universidad Nacional de Colombia, Bogotá.

2 E-mail: atriana@ciencias.ciencias.unal.edu.co.

3 E-mail: dcamposr@ciencias.ciencias.unal.edu.co.

### Elección de variables

Las colisiones de la partícula contra las paredes del cuadrado las enumeramos de manera consecutiva mediante el índice  $n$  que toma valores enteros,  $n = 0, 1, 2, 3, \dots$ . En cada colisión (digamos, la  $n$ ) se especifican dos variables, a saber: la posición del punto donde tiene lugar la colisión (variable  $s_n$ ) y el ángulo que la dirección del movimiento forma con la pared inmediatamente después de la colisión (ángulo  $\alpha_n$ ). La posición  $s_n$  la identificamos con la distancia a lo largo del perímetro medida a partir de la esquina inferior izquierda del cuadrado.



**Figura 1.** El billar de Sinai está conformado por un cuadrado y un obstáculo circular en su centro. La partícula se mueve clásicamente experimentando sólo colisiones elásticas.

### Planteamiento del problema

En el instante de tiempo inicial la partícula está en el estado  $(s_0, \alpha_0)$ , esto es, la partícula experimenta una colisión con la pared inferior en el punto  $s_0$  y emerge de la colisión formando un ángulo  $\alpha_0$ , tal como se muestra en la figura 1. Conociendo  $(s_0, \alpha_0)$  queremos predecir el estado  $(s_1, \alpha_1)$  que corresponde a la siguiente colisión de la partícula contra una pared del cuadrado, posiblemente después de haber rebotado en el objeto circular -la existencia de una colisión entre la partícula y el disco central depende de los valores  $(s_0, \alpha_0)$  y del radio  $R$  del disco-. Una vez determinado  $(s_1, \alpha_1)$  queremos en la siguiente etapa predecir el estado  $(s_2, \alpha_2)$  y así

sucesivamente. En general, queremos determinar una aplicación (en inglés *mapping*) que permita calcular el estado  $(s_n, \alpha_n)$  de la partícula en la  $n$ -ésima colisión, a partir del conocimiento del estado  $(s_{n-1}, \alpha_{n-1})$  correspondiente a la colisión  $n-1$ . En otras palabras, tenemos como propósito en este artículo encontrar la función  $F$  que permite establecer la siguiente relación:

$$(s_n, \alpha_n) = F(s_{n-1}, \alpha_{n-1}).$$

Es claro que esta función  $F$  será diferente para obstáculos centrales diferentes, y en particular dependerá explícitamente del radio del disco central.

## 2. Construcción del Algoritmo

Consideremos en primer lugar el billar en ausencia del disco central (figura 2). La partícula en este caso rebotará siempre con los mismos ángulos y los mismos lugares en cada lado. Tomando el cuadrado de lado  $L=1$  y el disco con  $R=1/4$  y dada una condición inicial es fácil entonces conocer  $s_n$  y  $\alpha_n$  de la siguiente forma: (nos desviaremos temporalmente aquí de la forma habitual de medir los ángulos en problemas de billares: por comodidad de cálculo los ángulos  $\alpha_i$  serán medidos de manera absoluta en el sentido directo desde el eje  $x$  positivo).

- Se halla la ecuación de la recta  $y = m x + b$  que corresponde a la trayectoria descrita por la partícula con condiciones iniciales  $(s_0, \alpha_0)$ , a través de  $m = \tan \alpha_0$  y de  $b = -m s_0$ , asumiendo que  $s_0$  se encuentra en el lado horizontal inferior (parte entera de  $s_0$  igual a cero:  $\text{int } s_0 = 0$ ).
- Si  $\alpha_0 < \pi/2$ , el punto de rebote  $s_1$  se encuentra calculando el valor de  $x$  para el cual  $y=1$  y luego el valor de  $y$  para el cual  $x=1$  en la ecuación de la recta (denotados  $x'$  y  $y'$  respectivamente). Sólo uno de estos dos valores pertenecerá al intervalo abierto  $(0,1)$  y corresponderá precisamente al punto de rebote de la partícula con uno de los lados del billar. Si es  $y'$  quien pertenece al intervalo  $(0,1)$  entonces las coordenadas del punto de rebote son  $(1, y')$  y podemos escribir  $s_1 = 1 + y'$  (recuérdese que  $s_i$  se mide a lo largo del perímetro del cuadrado) y también  $\alpha_1 = \pi - \alpha_0$ . Si  $x'$  es quien pertenece al intervalo  $(0,1)$  entonces las coordenadas del punto de rebote serán  $(x', 1)$  y por consiguiente  $s_1 = 3 - x'$ , en cuanto al ángulo éste será  $\alpha_1 = 2\pi - \alpha_0$ .

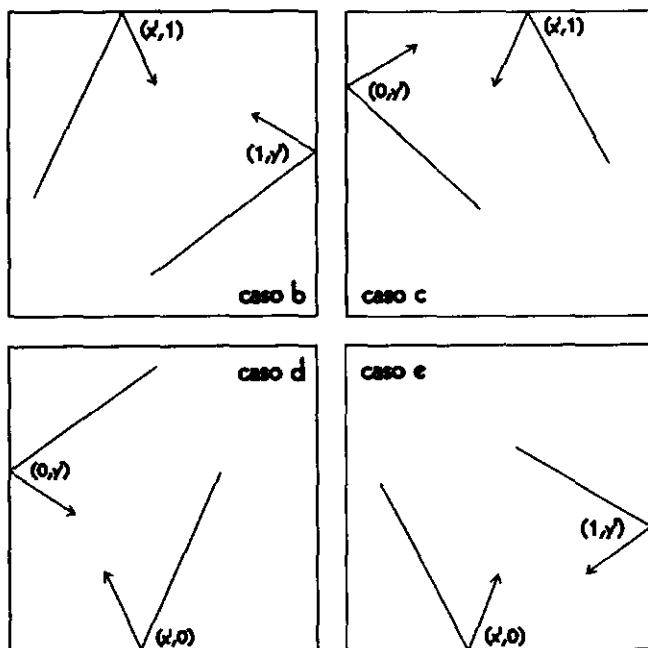


Figura 2. Determinación del punto de colisión de la partícula con las paredes del cuadrado en el caso en el que el objeto circular está ausente en el billar de Sinai.

- c. Si  $\pi/2 < \alpha_0 < \pi$  el punto de rebote se calcula de manera completamente análoga excepto que ahora se hace  $y=1$  y  $x=0$  en la ecuación de la recta. Con los dos valores obtenidos  $x'$ ,  $y'$  se realiza el mismo análisis: si  $0 < y' < 1$  entonces  $s_1=4-y'$ ,  $\alpha_1=\pi-\alpha_0$ . Si  $0 < x' < 1$  entonces  $s_1=3-x'$ ,  $\alpha_1=2\pi-\alpha_0$ .
- d. Si  $\pi < \alpha_0 < 3\pi/2$  (caso posible cuando la partícula proviene del lado vertical derecho ó del lado horizontal superior) se hace en la ecuación de la recta  $y=0$  y  $x=0$ . Se obtienen así los dos valores  $y'$  y  $x'$ . Si  $0 < y' < 1$  entonces  $s_1=4-y'$ ,  $\alpha_1=\pi-\alpha_0$ . Si  $0 < x' < 1$  entonces  $s_1=x'$ ,  $\alpha_1=2\pi-\alpha_0$ .
- e. Si  $3\pi/2 < \alpha_0 < 2\pi$  (la partícula proviene del lado vertical izquierdo ó del horizontal superior) se hace  $y=0$  y  $x=1$  obteniéndose  $y'$  y  $x'$ . Si  $0 < x' < 1$  entonces  $s_1=x'$ ,  $\alpha_1=2\pi-\alpha_0$ . Si  $0 < y' < 1$  entonces  $s_1=1+y'$ ,  $\alpha_1=\pi-\alpha_0$ .
- f. Se calcula una nueva ecuación de la recta a través de  $m=\tan \alpha_1$  y de  $b=-ms_1$  (si es un rebote sobre la banda horizontal inferior, esto es,  $\text{int } s_1=0$ ). Si se trata de un rebote sobre la banda vertical derecha ( $\text{int } s_1=1$ ) el parámetro  $b$  se calcula por medio de  $b=s_1-1-m$ . Si es un rebote sobre la banda horizontal superior ( $\text{int } s_1=2$ ) entonces  $b=1-m(3-s_1)$ . Si es un rebote sobre la banda vertical izquierda ( $\text{int } s_1=3$ ) entonces  $b=4-s_1$ .

Finalmente de regreso al paso b, reemplazando  $(s_0, \alpha_0)$  por  $(s_1, \alpha_1)$  y así sucesivamente hasta el valor de  $n$  que se desee.

Ahora sólo resta introducir el disco central en el billar. Para esto es necesario encontrar la función  $F$  de las variables  $(s_k, \alpha_k)$  que arroje como resultado el par de valores  $(s_{k+1}, \alpha_{k+1})$  asociados a un rebote con uno de los lados del billar luego de un rebote con el disco central si tuviese lugar.

Con el propósito de construir tal función primero debe calcularse el diámetro angular aparente ( $2\omega$ ) del disco y el ángulo central ( $\beta$ ) para algún valor dado de  $s$  para determinar si el ángulo de rebote  $\alpha_1$  pertenece a tal intervalo y si ocurre una colisión con el disco. A continuación se determina el sitio (el ángulo  $\theta_0$ ) sobre el círculo en el cual la colisión tiene lugar. Así el ángulo de salida ( $\alpha_2$ ) puede ser calculado y por consiguiente la ecuación de la recta que describe la partícula luego del choque con el disco.

En la figura 3 pueden apreciarse las variables involucradas. La tarea consiste en encontrar  $\theta_0$  y subsecuentemente  $\alpha_2$  como funciones de las variables conocidas  $s$ ,  $\alpha_1$  y  $R$ .

Al exigir en la colisión que el ángulo de incidencia de la partícula (respecto a la dirección normal a la superficie del disco en el punto de choque) sea igual al ángulo de rebote, puede encontrarse la relación  $\alpha_2 = 2\theta_0 - \alpha_1$ . Resta aún encontrar  $\theta_0$  y esto se logra resolviendo (para  $\theta$ ) el triángulo formado por el radio  $R$  (opuesto al ángulo  $\gamma$ ), el lado  $h$  y el primer segmento de la trayectoria de la partícula originada desde  $s$  (opuesto al ángulo  $\theta$ ). La expresión para  $\theta$  resulta ser:

$$\theta = \arcsen \left\{ \frac{\text{sen } \gamma \cos \gamma}{R} \left[ h \pm \sqrt{R^2 + (R^2 - h^2) \tan^2 \gamma} \right] \right\}$$

Una vez obtenido  $\theta$  es fácil conocer  $\theta_0$  pues el ángulo  $\delta$  es conocido. Calculadas las variables  $\theta_0$  y  $\alpha_2$  se tiene ya definida la recta correspondiente al segundo segmento de la trayectoria de la partícula luego del choque con el disco:

$$m = \tan \alpha_2, \quad b = 0.5 - R \cos \theta_0 - (0.5 + R \text{sen } \theta_0) \tan \alpha_2.$$

Con esta recta determinada queda únicamente por calcular el nuevo  $s$  correspondiente al siguiente rebote de la partícula con uno de los lados del billar de manera idéntica a la descrita en la primera parte de esta sección.

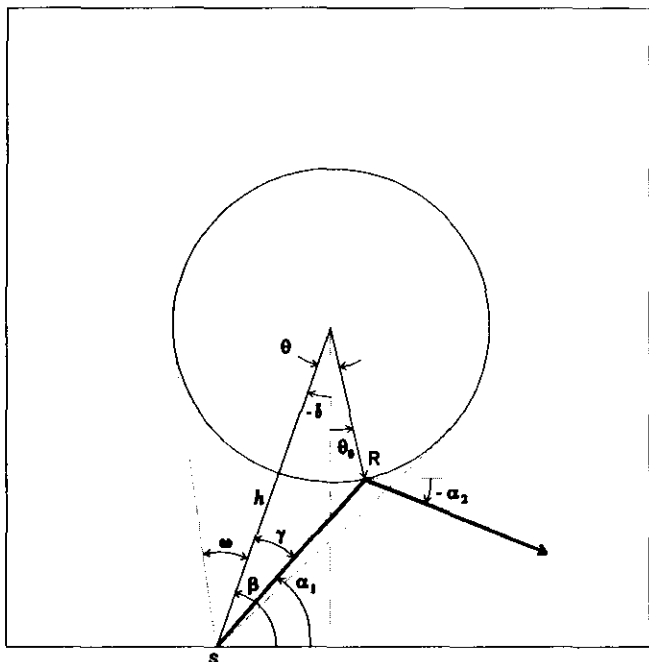


Figura 3. Variables geométricas que permiten determinar si ocurre una colisión con el disco.

La expresión para  $\theta$  y las relaciones auxiliares que definen las cantidades  $h$  y  $\gamma$  son:

$$h = \sqrt{(1/2)^2 + (1/2 - s)^2}$$

$$\gamma = |\beta - \alpha_1|$$

Se supone que la partícula proviene del lado horizontal inferior del billar. Para obtener el resultado correcto, cuando la partícula provenga de un lado cualquiera, simplemente se efectúa una rotación de una, dos ó tres veces  $\pi/2$  (sentido contrario a las manecillas del reloj) según la partícula provenga del lado derecho, superior ó izquierdo respectivamente. El resultado obtenido debe entonces ser modificado efectuando la rotación inversa.

Un listado completo de una implementación FORTRAN de este algoritmo se incluye como anexo.

### 3. Resultados

Con el algoritmo a mano es posible originar una trayectoria periódica (si de antemano se conociese su evolución) y seguirla hasta que se desviase de lo esperado y así estimar la precisión del algoritmo. Una de estas órbitas (trayectorias) periódicas más susceptibles de desviarse apreciablemente es la estrella (figura 4) pues tiene lugar una colisión con el disco central siempre entre dos rebotes sucesivos con los lados del billar. Como se ha visto en la

sección anterior, en una colisión con el disco central se ven involucradas funciones trigonométricas inversas, las cuales hacen uso directo de una representación numérica de  $\pi$  y otros números irracionales. Al representar  $\pi$  como un número de doble precisión se incurre en un pequeño error que luego de cada rebote con el disco central se va acumulando y dadas las características fuertemente caóticas del sistema este error basta para que al cabo de cierto número de colisiones la órbita periódica se pierda por completo.

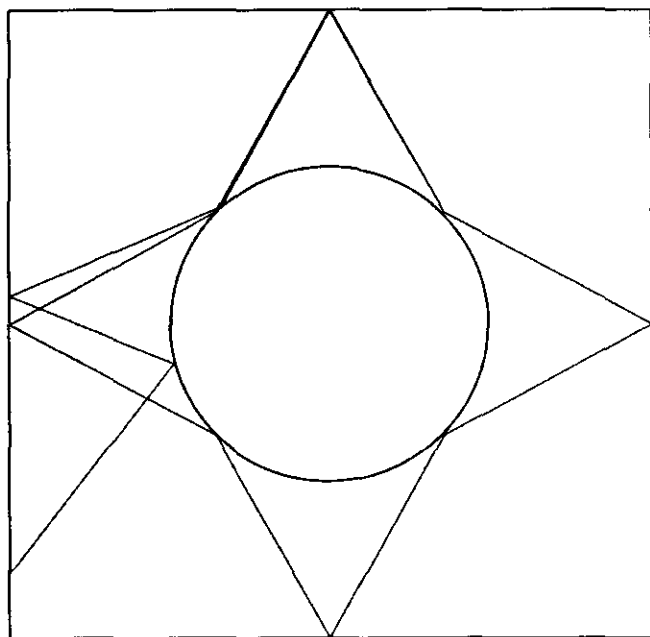


Figura 4. Análisis de la precisión del algoritmo con base en el comportamiento numérico de una órbita periódica (la estrella).

En la figura 4 (generada con el código FORTRAN anexo) puede apreciarse la órbita en forma de estrella que en principio se repite indefinidamente. Sin embargo luego de 16 rebotes con el disco central la partícula ya no rebota con el lado correcto: es ésta una órbita periódica *inestable*.

Existen muchas otras órbitas periódicas, unas de periodo largo y otras de periodo corto. El código que anexamos es útil en un análisis cualitativo de órbitas con periodos cortos, o en otras palabras, órbitas periódicas que en un solo periodo sufran menos de siete rebotes con el disco central, de modo que se repitan al menos dos veces y poder así identificar tal órbita con una órbita periódica verdadera.

Es interesante notar las huellas de las órbitas periódicas en el espacio de fase (superficie de sección o plano de Poincaré) asociado a este sistema. Como variables canónicamente conjugadas emplearemos  $s$  y el coseno del ángulo de rebote, medido esta vez sobre el lado

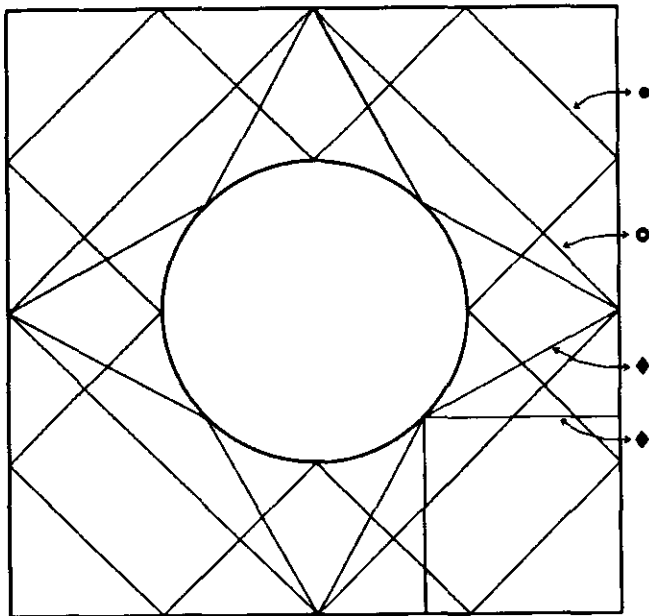


Figura 5a. Órbitas periódicas de periodo corto. Cada órbita se encuentra identificada con un símbolo.

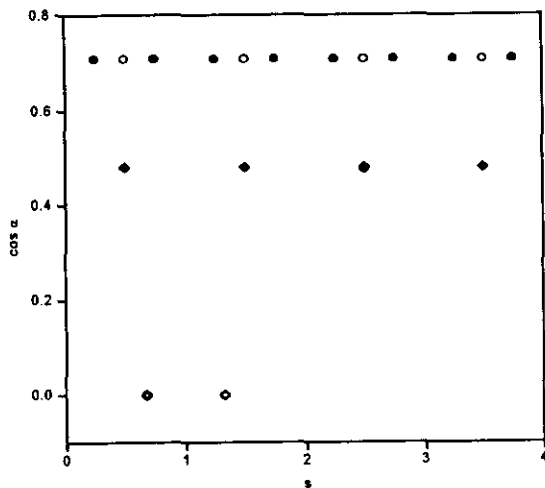


Figura 5b. Representación de las órbitas en el espacio de fase  $s$ - $\cos \alpha$ .

correspondiente, tal y como se indicó en la primera sección. No tomaremos en cuenta los rebotes con el disco central para construir la gráfica, sino únicamente los rebotes con los lados del billar de modo que cada punto en la superficie de sección corresponde a una colisión con una de las paredes. En la figura 5 podemos observar algunas órbitas periódicas de periodo corto y su representación en el espacio de fase. Adicionalmente a estas órbitas existen otras aún más triviales (que no se muestran en la gráfica) como

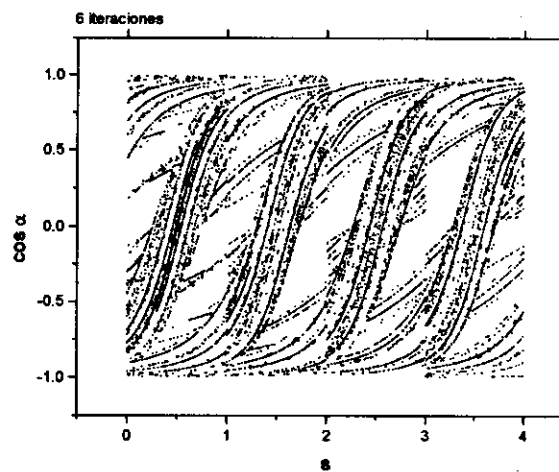
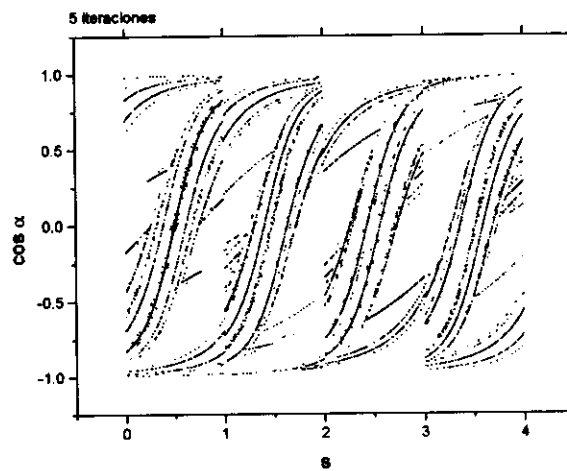
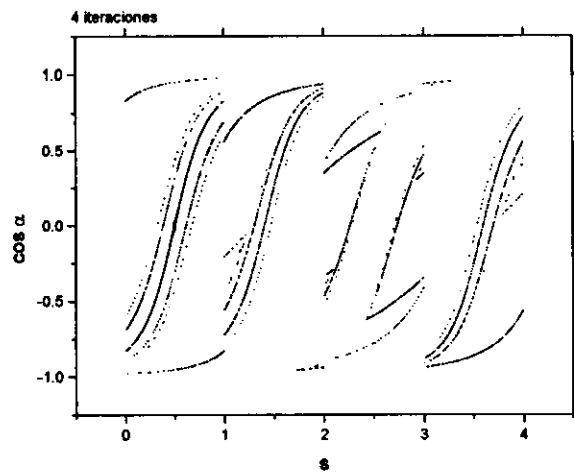
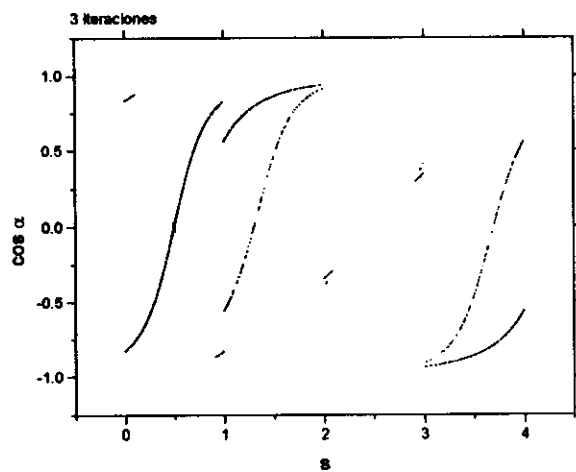
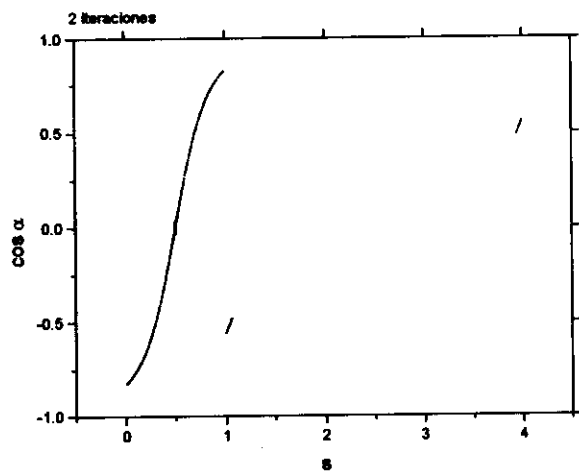
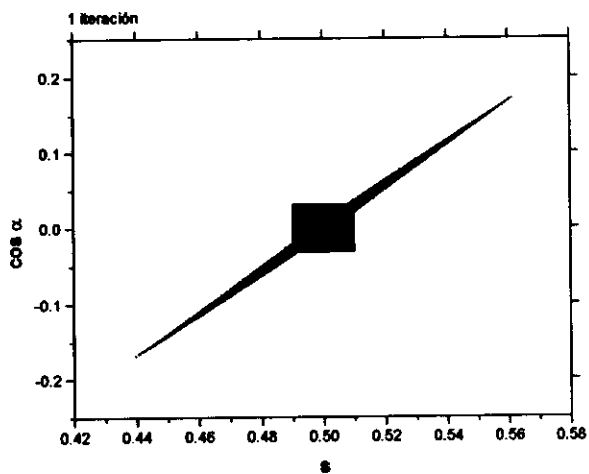
son aquellas en las que la partícula va y vuelve entre dos paredes opuestas del billar sin tocar el disco central, ó aquellas en las que la partícula rebota desde el punto medio de uno de los lados y choca perpendicularmente con el disco central.

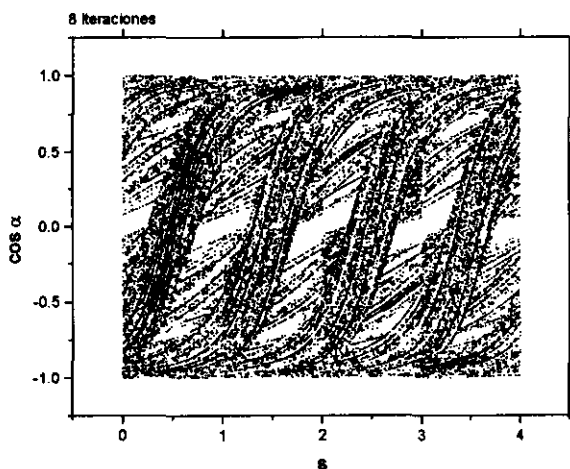
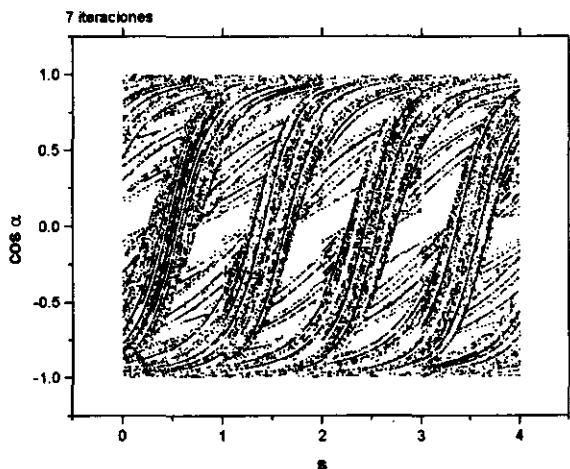
Una vez identificadas algunas órbitas regulares de periodo corto en el espacio de fase, podemos ahora realizar el siguiente experimento: preparar una colección de un gran número de billares idénticos con unas condiciones iniciales dadas y observar cómo evolucionan a medida que el número de rebotes aumenta (máximo 14 rebotes en total para obtener datos cualitativamente aceptables).

En las siguientes gráficas (figura 6) se muestra la evolución temporal de una colección de 66564 billares idénticos con condiciones iniciales  $(s_0, \alpha_0)$  distribuidas uniformemente para  $s_0$  entre 0.49 y 0.51 y para  $\alpha_0$  entre  $\pi/2 - \pi/100$  y  $\pi/2 + \pi/100$ . Estas condiciones iniciales corresponden al objeto cuadrado en la gráfica de la primera iteración. A medida que el número de iteraciones se incrementa la distribución inicial se deforma completamente y poco a poco empieza a cubrir todo el espacio accesible en la superficie de sección.

Eventualmente todo el espacio accesible es cubierto excepto por ciertas zonas que corresponden precisamente a órbitas periódicas (de periodo corto). Especialmente notorias son las zonas centrales que corresponden a órbitas que van y vienen de una pared a otra sin chocar con el disco central. Estas últimas órbitas no son estrictamente inestables puesto que es posible desplazar el punto inicial  $s_0$  infinitesimalmente y obtener de nuevo una órbita periódica del mismo tipo (conocidas en inglés como *bouncing ball modes*). Sin embargo si en el espacio de fase el desplazamiento infinitesimal de la condición inicial se realiza en ambas variables,  $s_0$  y  $\alpha_0$ , entonces las órbitas obtenidas se separarán -obedeciendo quizás una ley exponencial, caracterizada por el exponente de Lyapunov-, y llenarán todo el espacio de fase accesible. Estas órbitas constituyen un conjunto de medida nula en medio del conjunto de todas las órbitas posibles, de manera que es muy difícil que una órbita cualquiera (como una asociada con alguna de las condiciones iniciales que se han escogido en este ejemplo) coincida precisamente con una órbita periódica. A lo sumo podrá seguirla unos cuantos rebotes para luego desviarse completamente (algo parecido al caso de la estrella). Por tanto, los puntos del espacio de fase correspondientes a estas órbitas periódicas difícilmente serán cubiertos.

**Figura 6.** Evolución temporal de 66564 billares con condiciones iniciales concentradas alrededor de  $s_0=0.5$  y  $\alpha_0=\pi/2$ .





En las dos gráficas de la figura 7 se muestra el resultado luego de 14 iteraciones para el anterior experimento y también para otro en el que las condiciones iniciales se escogieron aleatoriamente de manera uniforme sobre toda la superficie de sección. En el primer caso las zonas no cubiertas (por encima y por debajo de la línea central horizontal) corresponden a todas las órbitas en forma de cuadrado ó rectángulo que nunca chocan con el disco, mientras que las situadas a lo largo de la línea central corresponden a todas las órbitas que rebotan sucesivamente entre dos lados opuestos (sin tocar el disco). En la segunda gráfica ampliada (correspondiente a condiciones iniciales aleatorias) estas últimas órbitas son también claramente discernibles.

#### 4. Conclusiones

El billar de Sinai es un sistema que presenta sensibilidad a las condiciones iniciales (requisito indispensable para poder ser calificado de caótico), como sugieren las gráficas de la evolución temporal, en donde a partir de unas condiciones iniciales muy cercanas entre sí, el sistema (la colección de sistemas) alcanza al cabo de pocos rebotes una distribución en muy buena medida uniforme sobre todo el espacio de fase accesible. Es ésta también la característica fundamental de los sistemas ergódicos, en donde un estado arbitrario en el espacio de fase accesible será alcanzado por el sistema considerado, dada una cantidad suficiente de tiempo y sin importar en absoluto la condición del sistema en el instante inicial.

La sensibilidad a las condiciones iniciales se refleja en la inestabilidad de las órbitas periódicas y esto origina una fuente de error considerable en el algoritmo que hemos desarrollado pues en él se hace uso de una representación numérica finita del número  $\pi$ . Será fruto de un nuevo trabajo (empleando una mejor representación numérica) verificar estas afirmaciones lo mismo que el cálculo de los exponentes de Lyapunov, que caracterizan el grado de inestabilidad de las órbitas.

#### Agradecimientos

Nuestros muy especiales agradecimientos a Carlos Viviescas y a Juan Diego Urbina por sus invaluable observaciones y correcciones al presente trabajo.

#### Bibliografía

- Blümel, R. y Reinhardt, W. P., 1997.** Chaos in Atomic Physics. Cambridge: Cambridge University Press.
- Mullin, Tom (Editor), 1995.** The Nature of Chaos. Oxford: Clarendon Press.
- Sinai, Y. G., 1979.** Russ. Math. Surv. **25**, pág. 137.

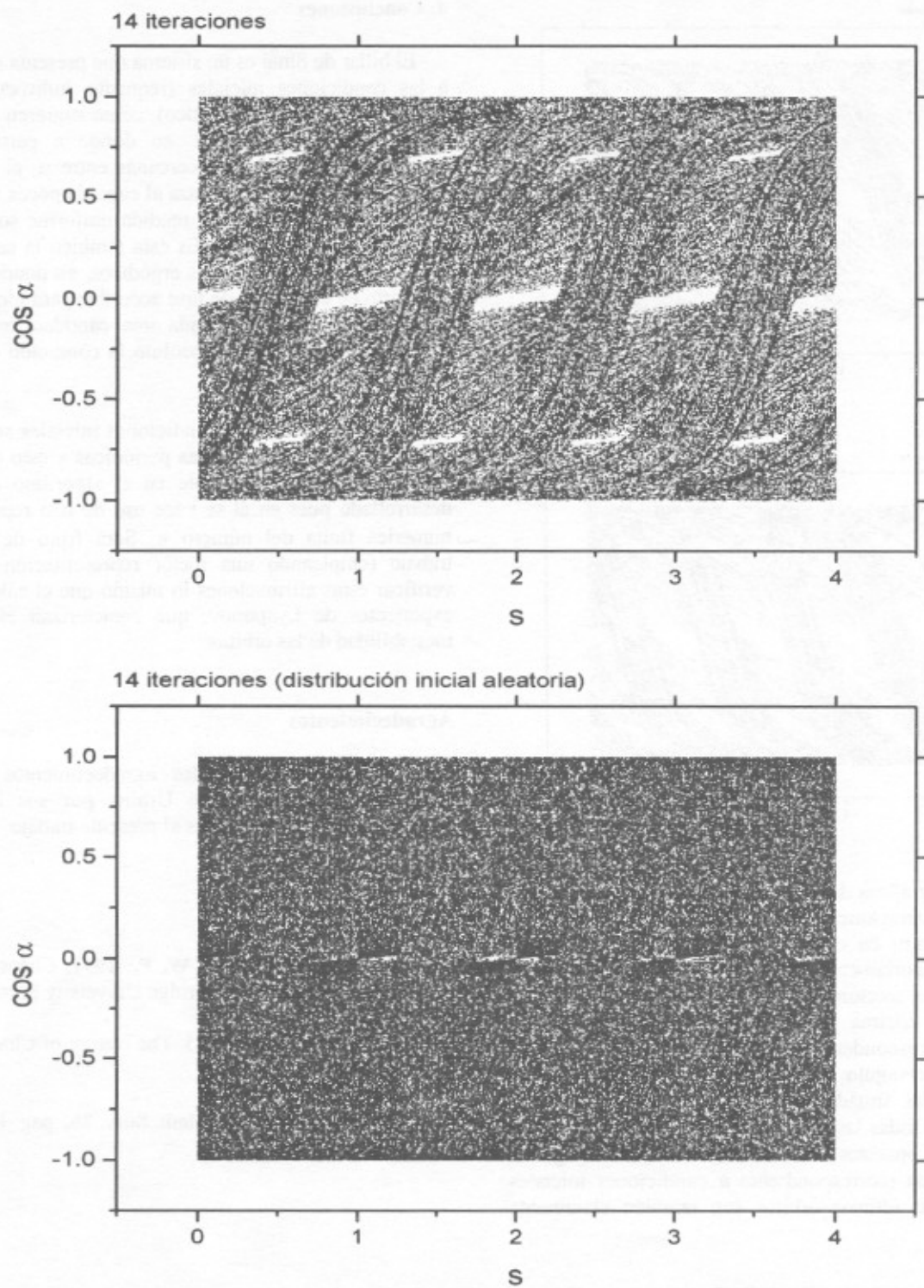


Figura 7. Evolución temporal luego de 14 iteraciones de una colección de 66564 billares con condiciones iniciales concentradas (gráfica superior) ó distribuidas aleatoriamente (gráfica inferior).



### Anexo: Una implementación FORTRAN

#### c Sinai's Billiard

```
Interface to double precision function sfunc (mf,bf,alphaf)
double precision mf, bf, alphaf
end
```

#### Program Sinai

```
double precision R, s, alpha, alpha1, alpha2, alpha3, var1, var2
double precision delta, beta, h, w, gamma, theta, theta1, theta2
double precision sfrac, b, m, theta0, theta00, pi, sfunc, x, y
character filename1*16, filename2*16
integer numic1, numic2, maxiter, sint, i, j
real*4 sr, alphas
call seed(-1)
```

```
pi=dacos(-1)
```

```
write (*,*) 'Enter Filename for Phase Space (<= 12)'  
read (*,'(a16)') filename1  
write (*,*) 'Enter Filename for Config Space (<= 12)'  
read (*,'(a16)') filename2  
write (*,*) 'Enter Disc Radius (<0.5)'  
read (*,*) R  
open (3, file=filename1)  
open (5, file=filename2)
```

```
numic1=258  
numic2=258  
maxiter=14
```

```
do 100 i=1, numic1
```

```
do 110 j=1, numic2
```

```
c s=0.49+0.02*i/numic1  
c alpha=(pi/2)-(pi/100)+(pi/50)*j/numic2
```

```
c s=0.5*i/numic1  
c alpha=(pi/2)*j/(numic2)
```

```
700 call random(sr)  
s=sr*4
```

```
call random(alphas)  
alpha=alphas*2*pi
```

```
if (int(s).eq.0) then  
if (alpha.gt.pi) then  
goto 700  
end if  
end if
```

```
if (int(s).eq.1) then  
if (alpha.lt.pi/2.or.alpha.gt.3*pi/2) then  
goto 700  
end if  
end if  
if (int(s).eq.2) then  
if (alpha.lt.pi) then  
goto 700  
end if  
end if  
if (int(s).eq.3) then  
if (alpha.gt.pi/2.and.alpha.lt.3*pi/2) then  
goto 700  
end if  
end if
```

```
c s=0.25  
c s=0.5  
c s=0.5+R/sqrt(2.0)  
c alpha=pi/4  
c alpha=pi/2-datan(R*sqrt(2.d0)/(1-R*sqrt(2.d0)))  
c alpha=pi/2
```

```
write (3,500) s, dcos(alpha)  
write (5,500) s, 0.0D0
```

```
do 300 n=1, maxiter
```

```
sint=int(s)  
sfrac=s-sint
```

```
alpha1=alpha-sint*pi/2
```

```
080 if (alpha1.gt.2*pi) then  
alpha1=alpha1-2*pi  
goto 080  
end if  
if (alpha1.lt.0) then  
alpha1=alpha1+2*pi  
goto 080  
end if
```

```
delta=datan((sfrac-0.5)/0.5)  
beta=pi/2+delta  
h=sqrt(0.25+(0.5-sfrac)**2)  
w=dasin(R/h)
```

```
if (alpha1.gt.beta-w.and.alpha1.lt.beta+w) then
```

```
gamma1=beta-alpha1  
gamma=abs(gamma1)  
var1=sqrt(R**2+(R**2-h**2)*((dtan(gamma))**2))  
var2=dsin(gamma)*dcos(gamma)/R  
theta1=dasin(var2*(h+var1))  
theta2=dasin(var2*(h-var1))
```

<pre> if (theta1.lt.theta2) then   theta=theta1 else   theta=theta2 end if  theta0=delta+ dsign(theta, gamma1) alpha2=2*theta0-alpha1  theta00=theta0+ sint*pi/2 alpha=alpha2+ sint*pi/2  x=0.5+ R*dsin(theta00) y=0.5-R*dcos(theta00) write (5,500) x, y </pre>	<pre> m=dtan(alpha) if (sint.eq.0) b=-m*sfrac if (sint.eq.1) b=sfrac-m if (sint.eq.2) b=1-m*(1-sfrac) if (sint.eq.3) b=1-sfrac </pre>
<pre> 015  if (alpha.gt.2*pi) then       alpha=alpha-2*pi       goto 015     end if     if (alpha.lt.0) then       alpha=alpha+ 2*pi       goto 015     end if     if (alpha.eq.pi/2) then       s=2.5+ R*dsin(theta0)       alpha=3*pi/2       goto 200     end if     if (alpha.eq.3*pi/2) then       s=0.5+ R*dsin(theta0)       alpha=pi/2       goto 200     end if      m=dtan(alpha)     b=0.5-R*dcos(theta00)-m*(0.5+ R*dsin(theta00))     goto 180 </pre>	<pre> 180  s=sfunc( m, b, alpha)       sint=int(s)       sfrac=s-int(s)       if (sint.eq.0.or.sint.eq.2) then         alpha=2*pi-alpha       else         alpha=pi-alpha       end if  025  if (alpha.gt.2*pi) then       alpha=alpha-2*pi       goto 025     end if     if (alpha.lt.0) then       alpha=alpha+ 2*pi       goto 025     end if  200  if (int(s).eq.0) then       x=s-int(s)       y=0     end if     if (int(s).eq.1) then       x=1       y=s-int(s)     end if     if (int(s).eq.2) then       x=1-s+ int(s)       y=1     end if     if (int(s).eq.3) then       x=0       y=1-s+ int(s) </pre>
<pre> end if  if (alpha.eq.pi/2) then   sint=2   sfrac=1-sfrac   s=sint+ sfrac   alpha=3*pi/2   goto 200 end if if (alpha.eq.3*pi/2) then   sint=0   sfrac=1-sfrac   s=sint+ sfrac   alpha=pi/2   goto 200 end if </pre>	<pre> end if  alpha3=alpha-sint*pi/2  write (3,500) s, dcos(alpha3) write (5,500) x, y  300  continue 110  continue 100  continue  500  format (2f20.16)  end </pre>

```
double precision FUNCTION sfunc( mf, bf , alphaf)
double precision mf, bf, alphaf, s1f, s2f, sfracf, pi
integer sintf
```

```
pi=dacos(-1)
```

```
if (alphaf.gt.0.and.alphaf.lt.pi/2) then
```

```
  s1f=mf+bf
  s2f=(1-bf)/mf
  if (s1f.lt.1) then
    sfracf=s1f
    sintf=1
```

```
  else
    sfracf=1-s2f
    sintf=2
```

```
  end if
end if
```

```
if (alphaf.gt.pi/2.and.alphaf.lt.pi) then
```

```
  s1f=bf
  s2f=(1-bf)/mf
  if (s1f.lt.1) then
    sfracf=1-s1f
    sintf=3
```

```
  else
    sfracf=1-s2f
    sintf=2
```

```
  end if
end if
```

```
if (alphaf.gt.pi.and.alphaf.lt.3*pi/2) then
```

```
  s1f=bf
  s2f=-bf/mf
  if (s1f.lt.1.and.s1f.gt.0) then
    sfracf=1-s1f
    sintf=3
```

```
  else
    sfracf=s2f
    sintf=0
```

```
  end if
end if
```

```
if (alphaf.gt.3*pi/2.and.alphaf.lt.2*pi) then
```

```
  s1f=-bf/mf
  s2f=mf+bf
  if (s1f.lt.1.and.s1f.gt.0) then
    sfracf=s1f
    sintf=0
```

```
  else
    sfracf=s2f
    sintf=1
```

```
  end if
end if
```

```
if (alphaf.eq.0.or.alphaf.eq.2*pi) then
```

```
  sintf=1
  sfracf=bf
```

```
end if
```

```
if (alphaf.eq.pi) then
```

```
  sintf=3
  sfracf=1-bf
```

```
end if
```

```
sfunc=sintf+sfracf
```

```
return
```

```
end
```