Basic Physics Course with MATLAB's Symbolic Toolbox and Live Editor

Kurt Bräuer

7.4 Billiard Chaos (Computational example)

Here we consider the 'free' motion of one (or two) balls in a stadium-shaped area.

The reflection on the boundary is simulated by a potential.

The distance between two adjacent paths, initially very close to each other, increases exponentially with time. There is a positive Liyapunov exponent.

The reflections on the boundary prevent the separation of the degrees of freedom. The system is chaotic. After about 10 reflections, any information about the initial conditions of the trajectories has disappeared.

```
clear all
syms x y t Xn Yn X(t) Y(t) Px(t) Py(t) R K m T
Par=[m==1 R==1 K==2 T==100]
```

Par = $\begin{pmatrix} m=1 & R=1 & K=2 & T=100 \end{pmatrix}$

1 Stadium boundary as potential

$$V(x,y) = \begin{cases} K(|y|-R)^2 & \text{if } x^2 \le R^2 \;\&\; y^2 > R^2 \\ K\left(\sqrt{(|x|-R)^2+y^2}-R\right)^2 & \text{if } x^2 > R^2 \;\&\; (|x|-R)^2+y^2 > R^2 \\ 0 & \text{otherwise} \end{cases}$$

```
V(x,y)=piecewise(x^2<=R^2 & y^2>R^2,K*(R-abs(y))^2,...
    R^2<(R-abs(x))^2+y^2 & R^2<x^2,K*(R-sqrt((R-abs(x))^2+y^2))^2,...
    0)
```

V(x, y) =

$$\begin{cases} K\;(R-|y|)^2 & \text{if }\; x^2 \le R^2 \land R^2 < y^2 \\ K\;\left(R-\sqrt{(R-|x|)^2+y^2}\right)^2 & \text{if }\; R^2 < (R-|x|)^2+y^2 \land R^2 < x^2 \\ 0 & \text{otherwise} \end{cases}$$

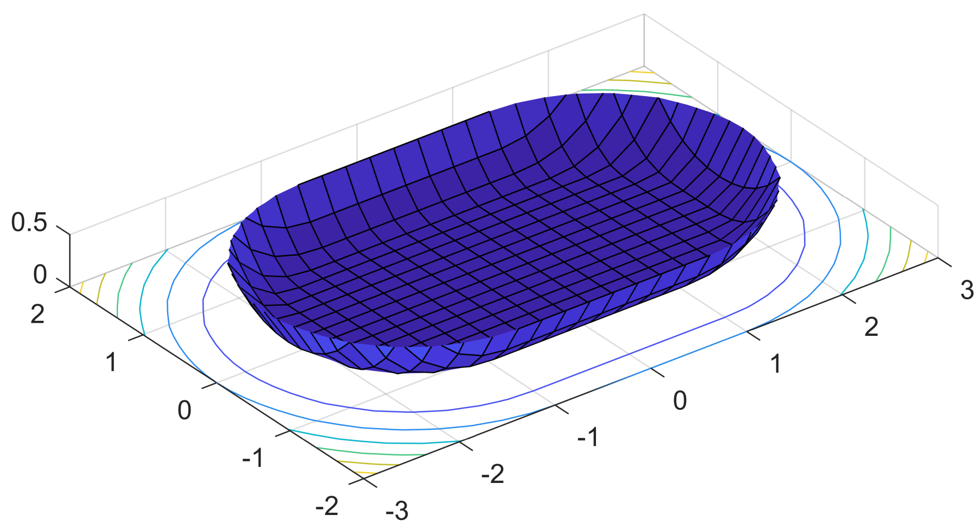
Graphics

```
[Xn,Yn]=meshgrid(linspace(-3,3,25),linspace(-2,2,25));
V=sube(V,Par)
```

```
V(x, y) =
```

$$\begin{cases} 2\,(|y|-1)^2 & \text{if } x^2 \leq 1 \wedge 1 < y^2 \\ 2\,\left(\sqrt{(|x|-1)^2+y^2}-1\right)^2 & \text{if } 1 < (|x|-1)^2+y^2 \wedge 1 < x^2 \\ 0 & \text{otherwise} \end{cases}$$

```
surfc(Xn,Yn,double(V(Xn,Yn)))
axis equal
axis([-3 3 -2 2 0 0.5])
hold off
```



2 Hamilton's equations of motion and numerical integration

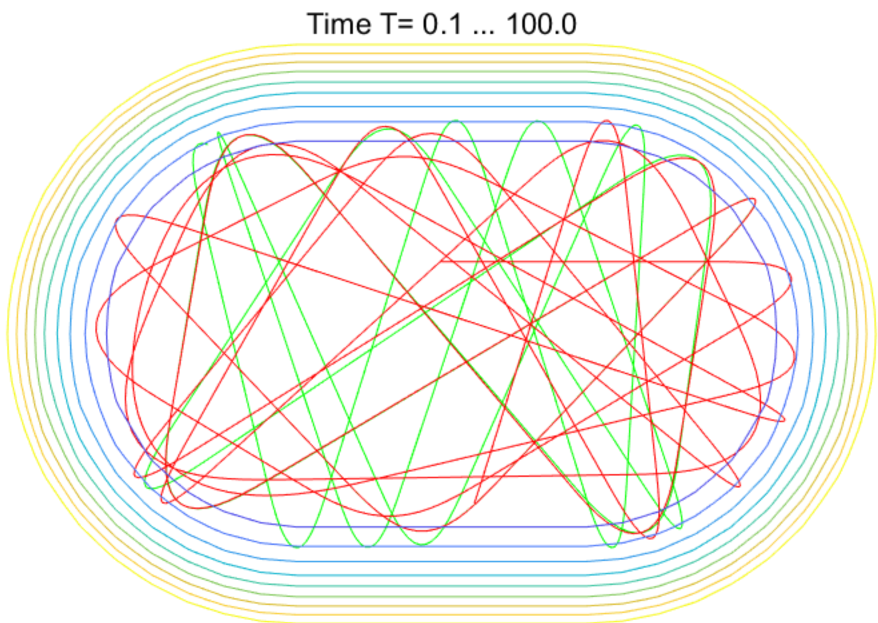
Hamiltonian: $H = \frac{1}{2m}\left(p_x^2 + p_y^2\right) + V(x, y)$

Equations of motion: $\begin{cases} \dot{x}_i = \frac{p_i}{m} \\ \dot{p}_i = -\frac{\partial H}{\partial x_i} \end{cases}$

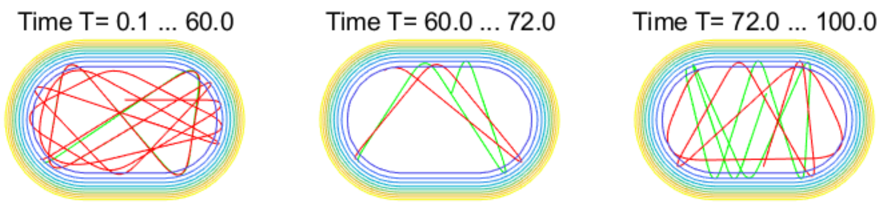
```
N=1000;
Vn=children(V);
Fx=-diff(Vn(1,1),x)*Vn(1,2)-diff(Vn(2,1),x)*Vn(2,2);
Fy=-diff(Vn(1,1),y)*Vn(1,2)-diff(Vn(2,1),y)*Vn(2,2);
var=[x==X(t),y==Y(t)];
dgl=[Px(t);Py(t);sube(Fx,var);sube(Fy,var)];
f=odeFunction(dgl,[X(t);Y(t);Px(t);Py(t)]);
%'RelTol' is chosen so small that a clear
% exponential divergences emerges (see Lyapunov exponent below)!
opt=odeset('RelTol',1e-10);
[tn,r1]=ode23(f,linspace(0,100,N),[0 0.5 1 0],opt);
[tn,r2]=ode23(f,linspace(0,100,N),[0 0.5+1e-7 1 0],opt);
```

3 Graphics

```
PloBil(r1,r2,Xn,Yn,V,1,N,N)
```




```
N1=0.6*N;
N2=0.72*N;
subplot(1,3,1)
PloBil(r1,r2,Xn,Yn,V,1,N1,N)
subplot(1,3,2)
PloBil(r1,r2,Xn,Yn,V,N1,N2,N)
subplot(1,3,3)
PloBil(r1,r2,Xn,Yn,V,N2,N,N)
```



4 Lyapunov-Exponent λ

$$\lambda : \quad |\Delta \vec{r}\,(t)| = |\Delta \vec{r}\,(0)| e^{\lambda t}, \; |\Delta \vec{r}\,| = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

There is an exponential divergence

```
subplot(1,1,1)
plot([0 100],[0 0],':k')
hold on
Delx=r2(:,1)-r1(:,1);
Dely=r2(:,2)-r1(:,2);
Del=log10(Delx.^2+Dely.^2)/2;
```

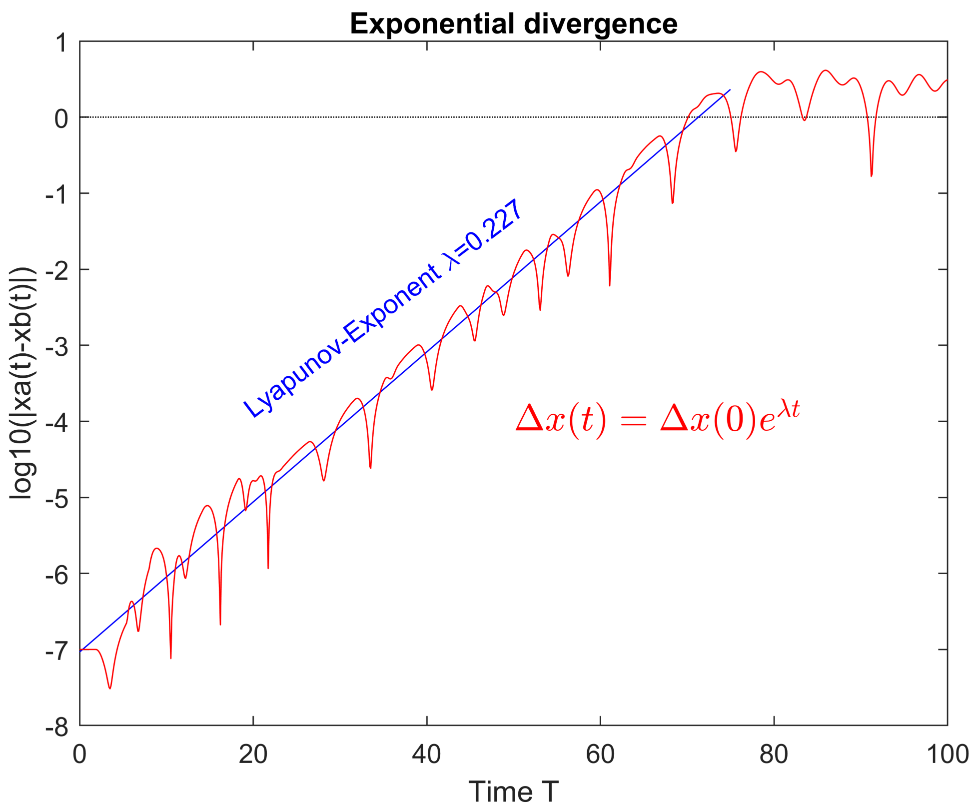
Lyapunov-Exponent

```
k1=1;
k2=0.75*N;
tn_=tn(k1:k2);
Del_=Del(k1:k2);
p1 = polyfit(tn_,Del_,1)
```

```
p1 = 1×2
     0.0987   -7.0334
```

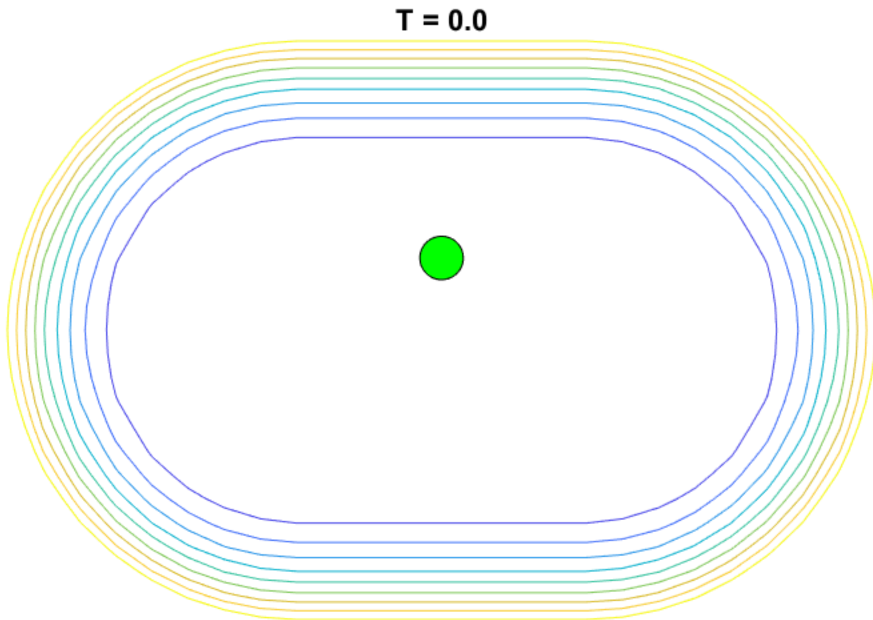
```
Liapo=p1(1)*log(10);
plot(tn_,polyval(p1,tn_),'b')
plot(tn,Del,'r')
```

```
title('Exponential divergence')
xlabel('Time T')
ylabel('log10(|xa(t)-xb(t)|)')
text(35,-2.5,sprintf('Lyapunov-Exponent \\lambda=%.3f',Liapo),...
    'HorizontalAlignment','center','Rotation',37,'Color','b')
text(50,-4,'$\Delta x(t)=\Delta x(0)e^{\lambda t}$','Color','r',...
    'Interpreter','latex','FontSize',14)
hold off
```

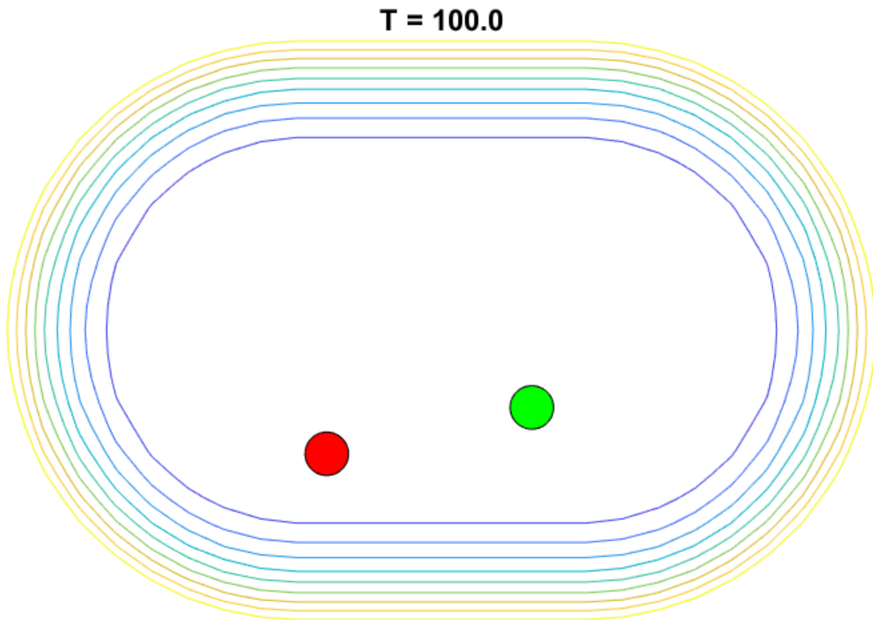


5 Animation --> 'Billiard_Animation.gif'

```
Na=500;
[tn,ya]=ode23(f,linspace(0,100,Na),[0 0.5 1 0],opt);
[tn,yb]=ode23(f,linspace(0,100,Na),[0 0.5+1e-7 1 0],opt);
figure
contour(Xn,Yn,V(Xn,Yn),linspace(0,2,10))
hold on
n=1;
r=0.15;
h1=rectangle('Position',r*[-1 -1 2 2]+[ya(n,1) ya(n,2) 0 0],'Curvature',[1 1],'FaceColor','r');
h2=rectangle('Position',r*[-1 -1 2 2]+[yb(n,1) yb(n,2) 0 0],'Curvature',[1 1],'FaceColor','g');
ht=title(sprintf('T = %2.1f',tn(n)));
axis equal
axis off
clear im
im{n}=frame2im(getframe(gcf));
```




```
for n=2:Na
    h1.Position=r*[-1 -1 2 2]+[ya(n,1) ya(n,2) 0 0];
    h2.Position=r*[-1 -1 2 2]+[yb(n,1) yb(n,2) 0 0];
    ht.String=sprintf('T = %2.1f',tn(n));
    im{n}=frame2im(getframe(gcf));
end
```




```
FiNa='Billiard_Animation.gif';
n=1;
[A,map] = rgb2ind(im{n},32);
imwrite(A,map,FiNa,'gif','LoopCount',Inf,'DelayTime',0);
for n=2:Na
    [A,map]=rgb2ind(im{n},32);
    imwrite(A,map,FiNa,'gif','WriteMode','append','DelayTime',0);
end
```

```
function PloBil(r1,r2,Xn,Yn,V,n1,n2,N)
% Plots trajectory of the stadium problem
plot(r1(n1:n2,1),r1(n1:n2,2),'g',r2(n1:n2,1),r2(n1:n2,2),'r')
hold on
contour(Xn,Yn,double(V(Xn,Yn)),linspace(0,2,10))
axis equal
axis off
hold off
title(sprintf('Time T= %2.1f ... %2.1f',100*n1/N,100*n2/N),'FontWeight','normal')
end
```