

## 3次元動画

Copyright ©2006 by Shigeto R. Nishitani

### 課題

animationを参照して、ルーレットのように回る玉のシミュレーションを作れ。

>

theta = 0.



ヒント：球の3次元表示はplottoolsのsphereを使い、

### 解答例

## 惑星のシミュレーション

これを改良した、太陽の周りを回る惑星のシミュレーションは以下のとおり。

ケプラーの第一法則にあるとおり、楕円軌道とする。

```
> restart;
a:=1:
b:=0.8:
with(plots):with(plottools):
e:=sqrt(1-b^2/a^2):
```

ケプラーの第二法則を満たすように一定時間に焦点に対して掃く面積が等しくなるように、角速度をあらかじめ計算しておいた。(この計算は見せたくないほどどろどろ。だれかスマートな解法をしたら教えて)

```
> theta40:=[0, .1702850391, .3621441466, .5843340015,
.8499000261, 1.173249127, 1.548794064, 1.927856352,
2.258146178, 2.529619975, 2.755986277, 2.950772893,
3.123171483, 3.279183922, 3.422833759, 3.556924188,
3.683483607, 3.804031515, 3.919742362, 4.031550481,
4.140219423, 4.246389688, 4.350612453, 4.453374408,
4.555116757, 4.656250390, 4.757168589, 4.858258539,
4.959912319, 5.062538193, 5.166573260, 5.272498249,
```

```
5.380855976, 5.492275381, 5.607504105, 5.727454284,
5.853268956, 5.986421978, 6.128873603, 6.283321858]:
```

描画関数 xを受け取って、それをtheta40にある角速度に置き換えている。あとは上で求めた玉の表示とおなじ。

```
> planet2:=proc(x)
local t;
t:=theta40[round(x)];
plots[display](
sphere([a*sin(t),b*cos(t),-1/sqrt((a*sin(t)-e*a)^2+(b*cos(t))^2)],0.1),
style=patchngrid );
end:
```

表示するポテンシャル曲面を作っておく。

```
> potential:=plot3d(-1/sqrt((x-e*a)^2+y^2),x=-1.2..1.2,y=-1.2..1.2,view=
-3..0,style=wireframe);
potential:=PLOT3D(...)
```

(2.1)

これらをすべて表示、つなぎ目(xx=1,39)でスムーズに表示するようにひとこま(xx=40)飛ばしている。

```
> animate(planet2,[xx],xx=1..39,frames=39,
background=potential, orientation=[-90,0],
scaling=constrained,axes=none);
```

