CURVAS

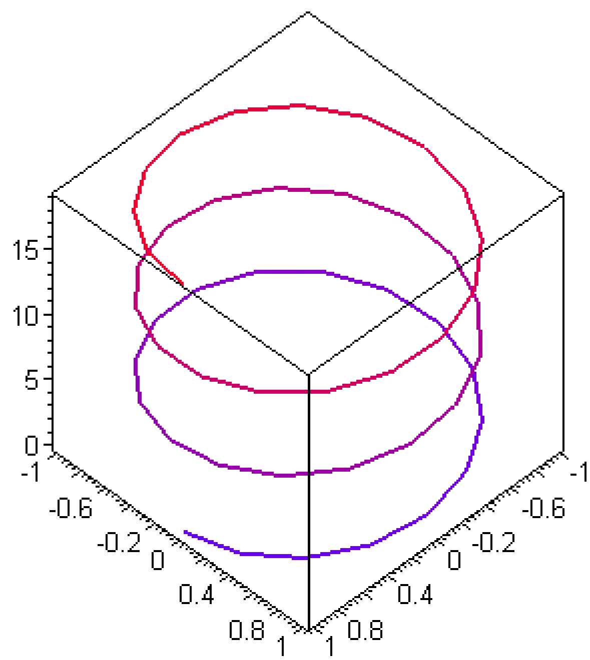
Curvas parametrizadas- comprimento de arco

```
> position:=[cos(t),sin(t),t];
```

$$position := [\cos(t), \sin(t), t]$$

```
> with(plots):
```

```
> spacecurve(position,t=0..6*Pi,axes=boxed,shading=z,thickness=2);
```




```
> t:='t':
```

```
> velocity:=diff(position,t);
```

$$velocity := [-\sin(t), \cos(t), 1]$$

```
> acceleration:=diff(velocity,t);
```

$$acceleration := [-\cos(t), -\sin(t), 0]$$

> **with(linalg):**

> **speed:=norm(velocity,2);**

$$speed := \sqrt{|\sin(t)|^2 + |\cos(t)|^2 + 1}$$

> **simplify(speed,trig);**

$$\sqrt{|\sin(t)|^2 + |\cos(t)|^2 + 1}$$

Produto interno

> **speed:=sqrt(dotprod(velocity,velocity));**

$$speed := \sqrt{\sin(t)^2 + \cos(t)^2 + 1}$$

> **speed:=simplify(speed);**

$$speed := \sqrt{2}$$

> **unittangent:=scalarmul(velocity,1/speed);**

$$unittangent := \left[-\frac{1}{2}\sqrt{2}\sin(t), \frac{1}{2}\sqrt{2}\cos(t), \frac{1}{2}\sqrt{2}\right]$$

> **arclength:=int(subs(t=u,speed),u=0..t);**

$$arclength := t\sqrt{2}$$

Usualmente o comprimento de arco é denotado pela letra *s* **. A idéia fundamental é considerar comprimento de arco como função de t, mas usualmente não podemos simplificar a integral associada.**

"Reparameterização pelo comprimento de arco" é mencionado em todo lugar. Ela assume velocidade constante unitária e é usado em muitas outras contas.

Como s é sempre crescente como função de t (pois v(t) nunca é zero),

é sempre invertível -- isto é "t como função de s" que é usado na reparametrização. Entretanto, em apenas poucos casos obtemos uma boa fórmula para s como função de t, que nós então podemos resolver t em função de s.

A reparametrização pelo comprimento de arco então usa esta fórmula de na definição da curva.

> **tt:=solve(s=arclength,t);**

$$tt := \frac{1}{2} s \sqrt{2}$$

> r:=subs(t=tt,position);

$$r := \left[\cos\left(\frac{1}{2} s \sqrt{2}\right), \sin\left(\frac{1}{2} s \sqrt{2}\right), \frac{1}{2} s \sqrt{2} \right]$$

> v:=diff(r,s);

$$v := \left[-\frac{1}{2} \sin\left(\frac{1}{2} s \sqrt{2}\right) \sqrt{2}, \frac{1}{2} \cos\left(\frac{1}{2} s \sqrt{2}\right) \sqrt{2}, \frac{1}{2} \sqrt{2} \right]$$

> speed:=sqrt(dotprod(v,v));

$$speed := \frac{1}{2} \sqrt{2 \sin\left(\frac{1}{2} s \sqrt{2}\right)^2 + 2 \cos\left(\frac{1}{2} s \sqrt{2}\right)^2 + 2}$$

> simplify(speed);

$$1$$

A reparametrização traça os mesmos pontos que a curva original, mas com velocidade unitária.

Vamos calcular um pouco mais de objetos geométricos associados com curvas.

> binormal:=crossprod(velocity,acceleration);

$$binormal := [\sin(t), -\cos(t), \sin(t)^2 + \cos(t)^2]$$

> binormal:=map(simplify,");

$$binormal := [\sin(t), -\cos(t), 1]$$

> principalnormal:=crossprod(binormal,velocity);

$$principalnormal := [-2 \cos(t), -2 \sin(t), 0]$$

Finalmente obtemos os três vetores normalizados:

> T:=evalm(unittangent);

$$T := \left[-\frac{1}{2} \sqrt{2} \sin(t), \frac{1}{2} \sqrt{2} \cos(t), \frac{1}{2} \sqrt{2} \right]$$

> N:=scalarmul(principalnormal,1/sqrt(dotprod(principalnormal,principalnormal)));

$$N := \left[-\frac{\cos(t)}{\sqrt{\sin(t)^2+\cos(t)^2}}, -\frac{\sin(t)}{\sqrt{\sin(t)^2+\cos(t)^2}}, 0\right]$$

```
> N:=map(simplify,N);
```

$$N := [-\cos(t), -\sin(t), 0]$$

```
> B:=scalarmul(binormal,1/sqrt(dotprod(binormal,binormal)));
```

$$B := \left[\frac{\sin(t)}{\sqrt{\sin(t)^2+\cos(t)^2+1}}, -\frac{\cos(t)}{\sqrt{\sin(t)^2+\cos(t)^2+1}}, \frac{1}{\sqrt{\sin(t)^2+\cos(t)^2+1}}\right]$$

```
> B:=map(simplify,B);
```

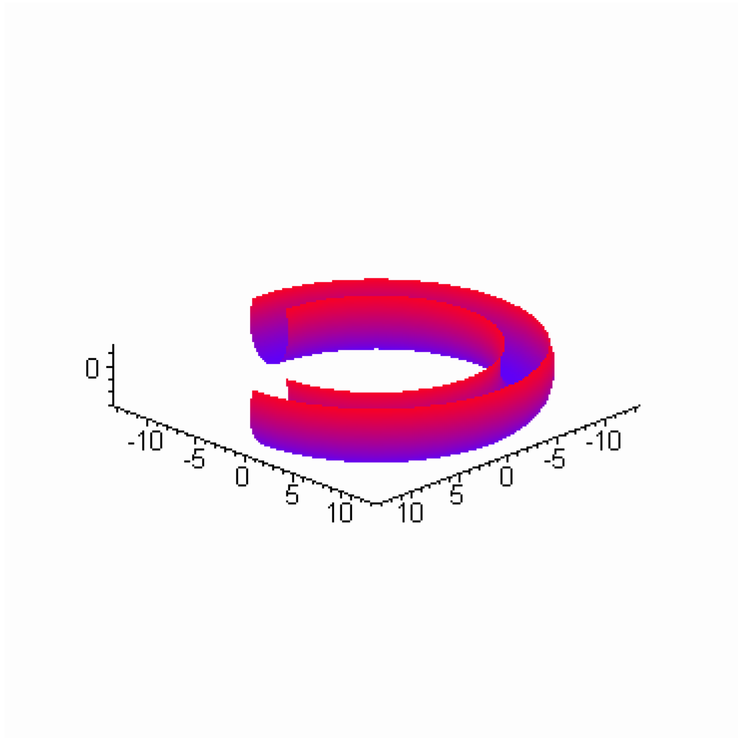
$$B := \left[\frac{1}{2}\sqrt{2}\sin(t), -\frac{1}{2}\sqrt{2}\cos(t), \frac{1}{2}\sqrt{2}\right]$$

Exercício:

Verifique que os vetores T,N,B são unitários e são mutuamente ortogonais.

Lembremos a diferença entre equações paramétricas de uma reta e de um plano. Inves de um parâmetro t, nós precisamos de dois parâmetros *s* e *t* . Usualmente usamos os símbolos θ e ϕ , ou letras *u* e *v* .

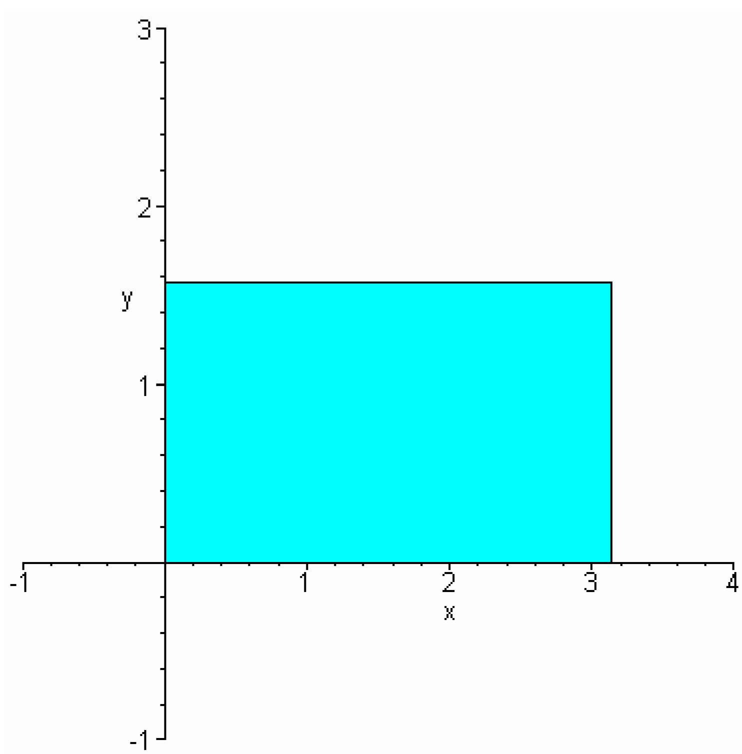
```
> plot3d([(11+2*cos(x))*cos(y),(11+2*cos(x))*sin(y),3*sin
(x)],x=0.8*Pi..2.2*Pi,y=0..3/2*Pi,orientation=
[45,68],shading=Z,scaling=constrained,style=patchnogrid,axes=frame);
```



Exemplo:

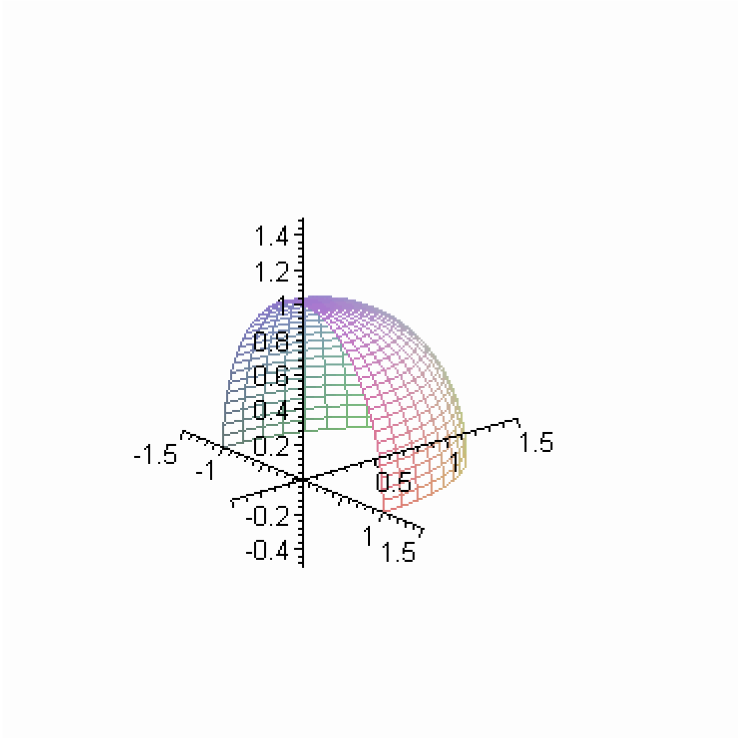
O seguinte parametriza a região retangular em parte da esfera:

> **polygonplot([[0,0],[Pi,0],[Pi,Pi/2],[0,Pi/2]],view=[-1..4,-1..3],color=cyan,labels=[x,y]);**



> **region:=":**

> plot3d([cos(x)*sin(y),sin(x)*sin(y),cos(y)],x=0..Pi,y=0..Pi/2, orientation=[-40,70],axes=normal,view=[-1.5..1.5,-.5..1.5,-.5..1.5]);

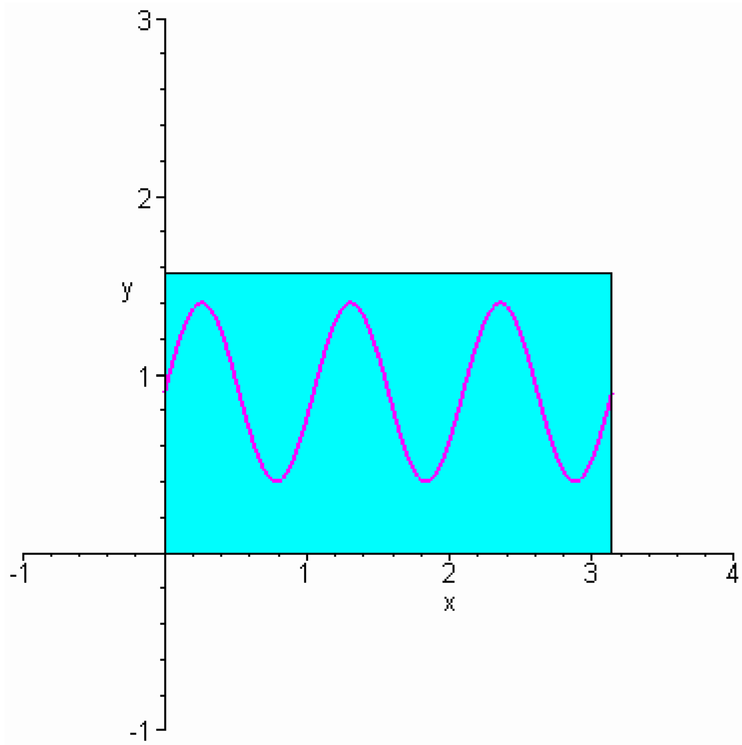


> surf:='':

Agora vamos desenhar uma curva no plano uv, dentro da região:

> curve:=plot([t,0.9+0.5*sin(6*t),t=0..Pi],color=magenta,thickness=2):

> display({curve,region});



Esta curva definida u=t, v=0.6+0.5*sin(6*t) para apropriados valores de t.

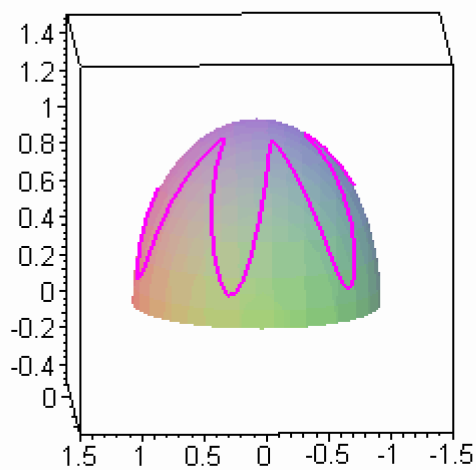
A curva será levada sobre a superficie:

```
> r:=1.01:

> scurve:=spacecurve([r*cos(t)*sin(0.9+0.5*sin(6*t)), r*sin(t)*sin(0.9+0.5*sin(6*t)),r*cos
(0.9+0.5*sin(6*t))],

> t=0..Pi,color=magenta,thickness=2):

> display({surf,scurve},axes=boxed,style=patchnogrid,orientation=[88,82]);
```


1.4
1.2
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
0
1.5
1
0.5
0
-0.5
-1
-1.5

>

>

>