

3.4 Movimento ao longo de uma curva no espaço (parte segunda)

3.4.4 Mais exemplos sobre curvas no espaço.

No parágrafo anterior discutimos os elementos que entram na descrição de uma trajetória de uma partícula no espaço tridimensional (no 3-espaço).

Vamos, primeiramente, considerar um problema sobre os conceitos introduzidos até aqui (curvatura, raio de curvatura, tangente etc.).

Depois retomamos os programas usados na seção 3.4 (primeira parte) para conhecer alguns aperfeiçoamentos na sua formulação. Não precisamos, na realidade, refazer os programas, é suficiente familiarizar-nos com técnicas mais curtas e mais elegantes que MuPAD oferece para representar graficamente pontos, grupos de curvas, setas etc.

1. Velocidade, tangente, curvatura etc.

Uma partícula move-se ao longo de uma trajetória C cujas equações paramétricas são

$$x = 3 \cdot \cos(2t), y = 3 \cdot \sin(2t), z = 8t - 4; t \text{ é o tempo.}$$

- a. Determinar um vetor-tangente (velocidade) à C .
- b. Qual o módulo deste vetor?
- c. Mostrar que $\mathbf{t} = -3/5 \cdot \sin(2t) \cdot \mathbf{i} + 3/5 \cdot \cos(2t) \cdot \mathbf{j} + 4/5 \cdot \mathbf{k}$ é um vetor-tangente-unitário à C .
- d. Mostrar que a curvatura é $3/25$
- e. Faça uma representação gráfica da trajetória C .

Solução:

```

x:=t->3*cos(2*t)://Problema 1
y:=t->3*sin(2*t):
z:=t->(8*t-4):
pos:=matrix([[x(t),y(t),z(t)]])://vetor-posição
v:=matrix([[x'(t),y'(t),z'(t)]])://vetor-velocidade
ac:=matrix([[x''(t),y''(t),z''(t)]])://vetor-aceleração
absolv:=sqrt(v[1]^2+v[2]^2+v[3]^2):
vt:=v/absolv:
vt_abs:=sqrt(vt[1]^2+vt[2]^2+vt[3]^2):
k:=diff(vt,t)/absolv://eq. (3.4.1-6) ou (14)
k_abs:=sqrt(k[1]^2+k[2]^2+k[3]^2):
simplify(v);
simplify(absolv);
simplify(vt);//vetor-tangente-unitário
simplify(vt_abs);
simplify(k);
simplify(k_abs);

```

$(-6 \cdot \sin(2 \cdot t) \ 6 \cdot \cos(2 \cdot t) \ 8)$

10

$\left(-\frac{3 \cdot \sin(2 \cdot t)}{5} \ \frac{3 \cdot \cos(2 \cdot t)}{5} \ \frac{4}{5} \right)$

1

$$\left(-\frac{3 \cdot \cos(2 \cdot t)}{25} -\frac{3 \cdot \sin(2 \cdot t)}{25} 0 \right)$$

$$\frac{3}{25}$$

- `curve := plot::Curve3d([3*cos(2*t), 3*sin(2*t), 8*t-4],
t=0..2*PI):`
`//t = 0..a, a = 0..2*PI -para uma animação`
`plot(curve)`

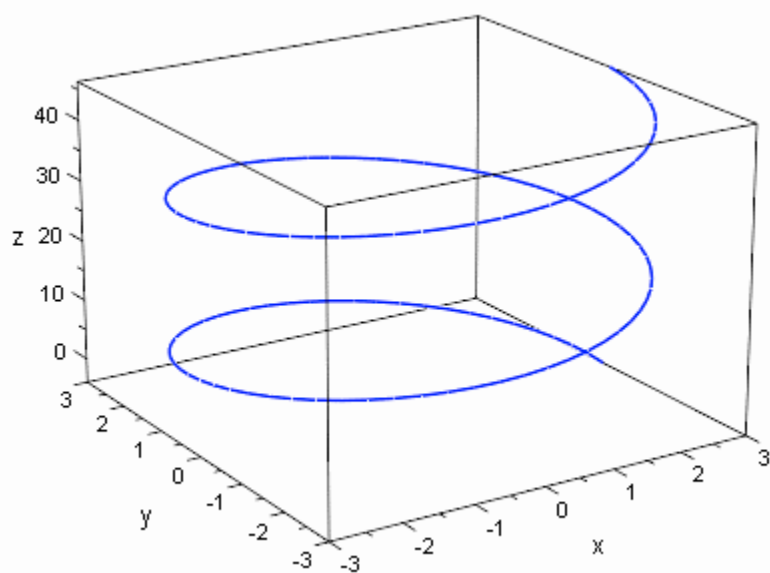


Fig. 3.4-6

2. Faça um gráfico de três elipses com a mesma origem.
(Sugestão: empregar $x = i \cdot \cos(t)$, $y = i/2 \cdot \sin(t)$ para $i = 1,2,3$)

Solução:

```

• reset() // 3 elipses
  for i from 1 to 3 do
    x:=t->i*cos(t):
    y:=t->i/2*sin(t):
    curve(i):=plot::Curve2d([x(t),y(t)],t=0..2*PI):
  end_for;

```

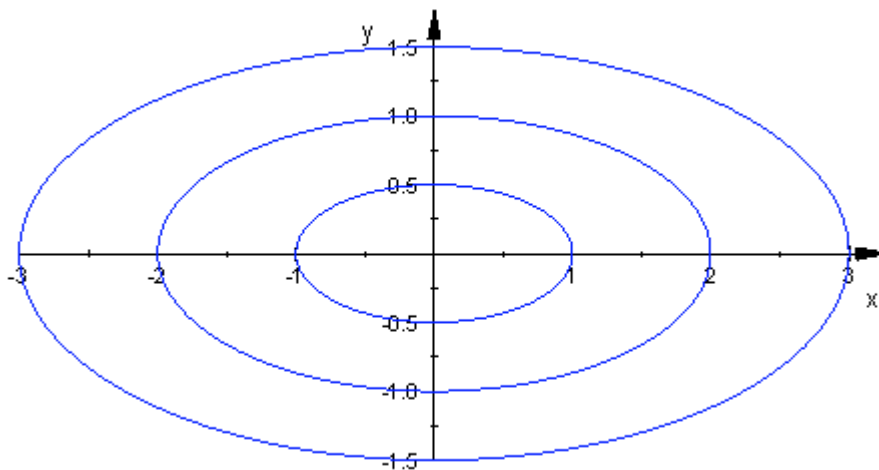


Fig. 3.4-7

```

plot(curve(i)$ i=1..3,
      Scaling=Constrained)

```

3. Coloque 6 pontos $(x(i), y(i))$, $i = 1 \dots 6$ sobre a elipse $\{9 \cdot \cos(t), 5 \cdot \sin(t)\}$ (a elipse serve de pano de fundo).

Solução:

```
reset() :  
x:=t->9*cos(t) :  
y:=t->5*sin(t) :  
curve:=plot::Curve2d([x(t),y(t)],Color=RGB::Red,t=0..2*PI) :  
for i from 1 to 6 do  
p(i):=plot::Point2d([x(i),y(i)]) :  
end_for ;  
plot(curve,p(i) $ i = 1 ..6,  
Scaling=Constrained)
```

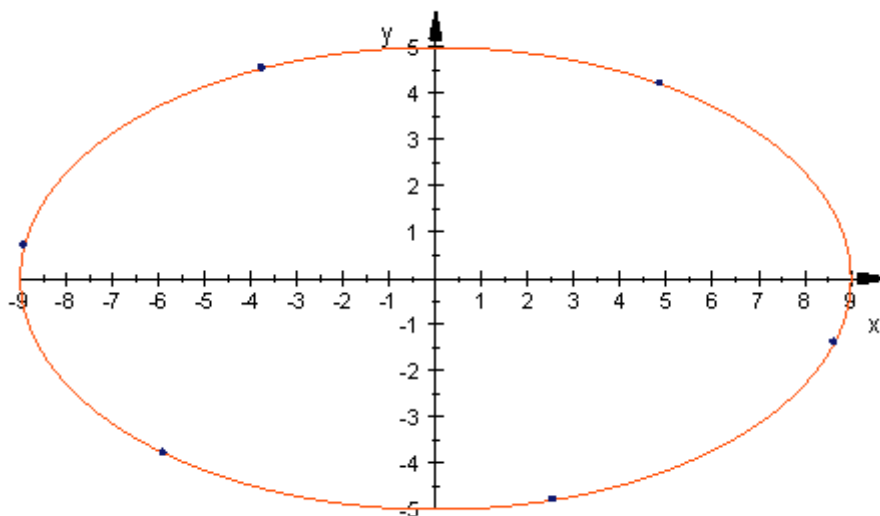


Fig. 3.4-8

4. Refazer o problema 3 para valores de i da lista "valores = [PI/3,PI/4,PI/5]"

Solução:

```

• reset() :
valores := [PI/3, PI/4, PI/5] :
x := t -> 9 * cos(t) :
y := t -> 5 * sin(t) :
pos := matrix([[x(t), y(t)]]) :
curve := plot::Curve2d(pos, Color=RGB::Red, t=0..2*PI) :
p := plot::Point2d(subs([pos[1], pos[2]], t=valores[i]),
Color=RGB::Green, PointSize=2*unit::mm) $ i=1..3:
plot(curve, p, Scaling=Constrained)

```

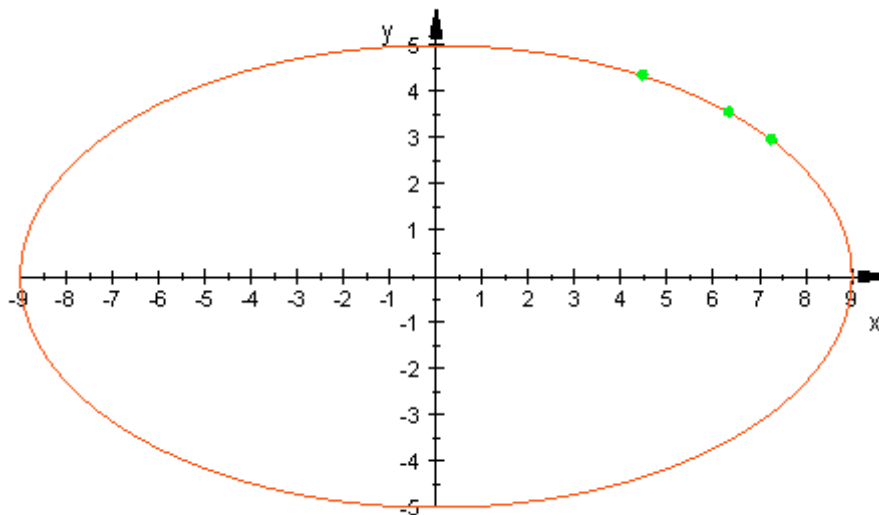


Fig. 3.4-9

5. Coloque nos 3 pontos do gráfico anterior os vetores-tangenciais unitários.

Solução:

```

reset() :
valores:=[PI/3,PI/4,PI/5] :
x:=t->9*cos(t) :
y:=t->5*sin(t) :
pos:=matrix([[x(t),y(t)]]) :
v:=matrix([[x'(t),y'(t)]]) :
vabs:=sqrt(v[1]^2+v[2]^2) :
vt:=v/vabs :
curve:=plot::Curve2d(pos,Color=RGB::Red,t=0..2*PI) :
x1:=pos[1]: y1:=pos[2]:
x2:=x1+2*vt[1]: y2:=y1+2*vt[2]:
p:=plot::Point2d(subs([x1,y1],t=valores[i]),
Color=RGB::Green,PointSize=2*unit::mm) $ i=1..3:
vtg:=plot::Arrow2d(subs(([x1,y1],[x2,y2]),t=valores[i]),
Color=RGB::Blue) $ i=1..3:
plot(curve,p,vtg,Scaling=Constrained)

```

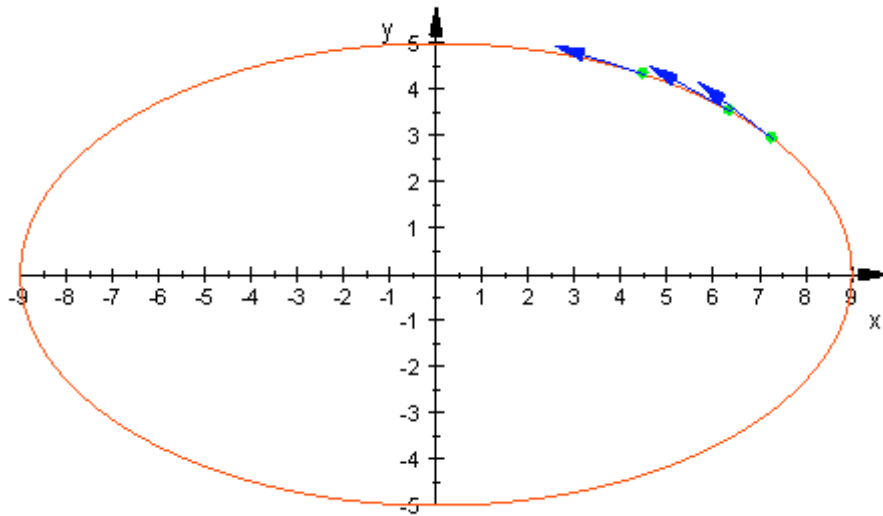


Fig. 3.4-10

6. Ampliar o gráfico anterior com os vetores da curvatura

Solução:

```

reset() //t-k-plot
valores:=[PI/2,PI/4,PI/8]:
scale:=20:
x:=t->9*cos(t):
y:=t->5*sin(t):
pos:=matrix([[x(t),y(t)]]):
v:=matrix([[x'(t),y'(t)]]):
vabs:=sqrt(v[1]^2+v[2]^2):
vt:=v/vabs:
k:=diff(vt,t)/vabs:
curve:=plot::Curve2d(pos,Color=RGB::Red,t=0..2*PI):

```



```

x1:=pos[1]: y1:=pos[2]:
x2:=x1+0.5*v[1]: y2:=y1+0.5*v[2]:
x3:=x1+scale*k[1]:y3:=y1+scale*k[2]:
p:=plot::Point2d(subs([x1,y1],t=valores[i]),
Color=RGB::Green,PointSize=2*unit::mm) $ i=1..3:
vtg:=plot::Arrow2d(subs(([x1,y1],[x2,y2]),t=valores[i]),
Color=RGB::Blue) $ i=1..3:
vk:=plot::Arrow2d(subs(([x1,y1],[x3,y3]),t=valores[i]),
Color=RGB::Black) $ i=1..3:
plot(curve,p,vtg,vk,Scaling=Constrained)

```

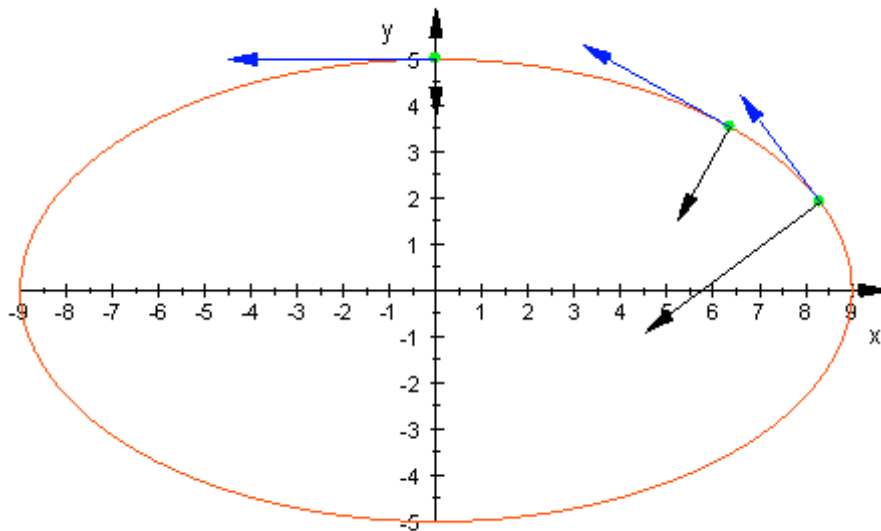


Fig. 3.4-11

7. Para fazer um gráfico de uma função $f_a(x)$, para k diferentes valores do parâmetro a , utilizamos a função `plotfunc2d(subs(f_a, a=valores[i])`
`$ i = 1..k, xrange = -1..5, yrange = -1..1)`
 Estabelecer o gráfico de $f_a(x) = \ln x / x^a$, x positivo e valores de $a = [0, 0.5, 1, 2]$.

Solução:

- `assume(x, Type::Positive):`
`f_a:=ln(x)/x^a:`
`valores_de_a:=[0,0.5,1,2]:`
`plotfunc2d(subs(f_a,a=valores_de_a[i]) $ i=1..4,`
`xrange=-1..5, yrange=-1..1);`

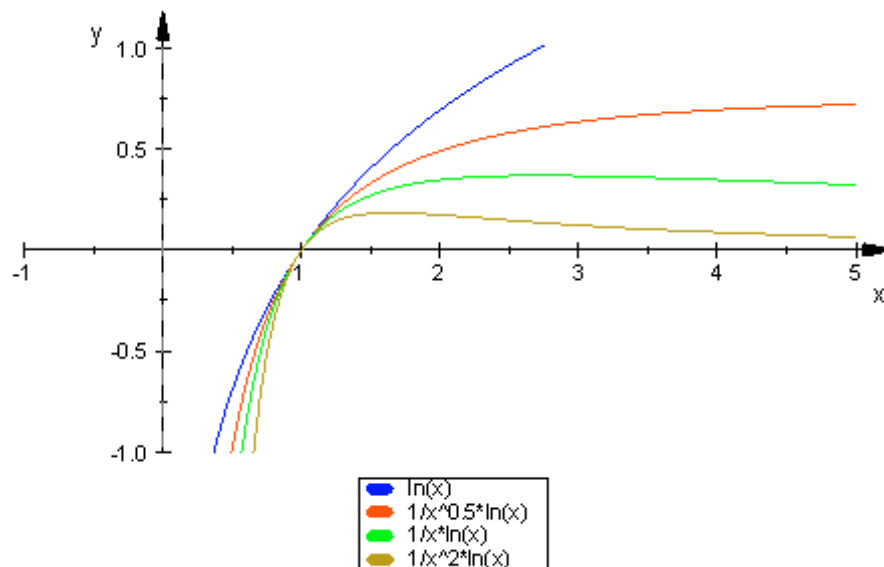


Fig. 3.4-12

3.4.5 Superfície e círculo osculador no espaço tridimensional

Uma função de duas variáveis pode ser representada, graficamente, por uma superfície no espaço de três dimensões. Para funções de três ou mais variáveis, essa representação não é possível.

Existe, porém, a possibilidade de representar funções de três variáveis por meio de superfícies de nível. É o método usado, por exemplo, para estabelecer o potencial gravitacional constante ao redor da Terra. Na meteorologia conhecemos as superfícies de temperatura constante ou pressão constante.

Você se lembra, certamente, que o tema das superfícies já foi matéria de estudo em algumas seções passadas, veja p.ex. as seções 1.1 ou 2.3. Na seção 2.3 usamos a função `plot::Surface` e na seção 1.1.3 foi introduzida a função `plotfunc3d`. É aconselhável reler essas seções de tempos passados, certo?

Vamos agora considerar mais alguns **exemplos**.

Exemplo 1:

- Faça um gráfico da **superfície** $z = 4 - x^2 - 2y^2$ para $x = -4 \dots 4$ e $y = -3 \dots 3$
- Determine a equação do **plano tangente** à superfície $z = 4 - x^2 - 2y^2$ no ponto $(2, -1)$. (Veja "Com Lapis e papel". O plano tangente será $z_t = 10 - 4(x-y)$).

Solução:

Parte a:

- `reset()` :
- `superficie:=4-x^2-2*y^2:`
- `plotfunc3d(superficie,x=-4..4,y=-3..3,`
- `CameraDirection=[10,-5,35])// experimentar!`

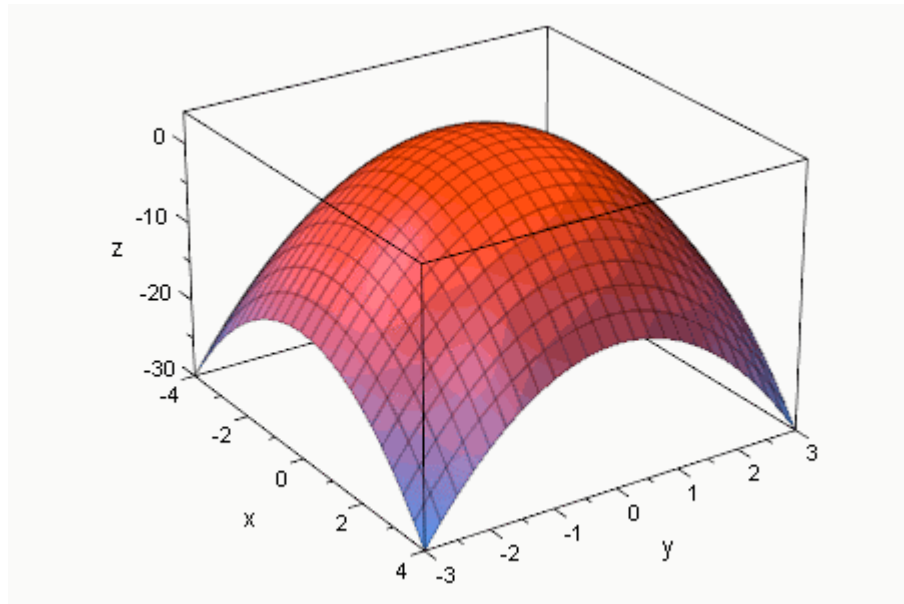


Fig. 3.4-13

Parte b:

- `superficie:=4-x^2-2*y^2:`
`plano:=10-4*(x-y):`
`plotfunc3d(superficie,plano,x=-4..4,y=-3..3,`
`CameraDirection=[10,-5,35])`

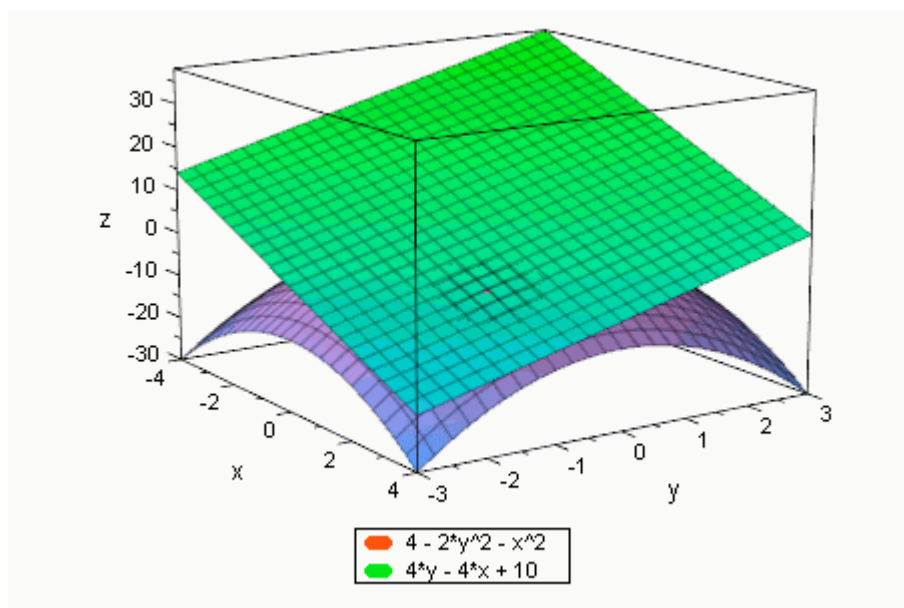


Fig.3.4-14

No parágrafo 2.3.3, mencionei as diferentes ferramentas (Tools) que MuPAD oferece para inspecionar os gráficos em três dimensões. Com Tools/Query pode-se ler, aproximadamente, as coordenadas do ponto de contato entre plano e superfície; encontra-se, por exemplo: $a \sim 1,96$, $b \sim -1,02$, $f(2,-1) \sim -1,93$.

Exemplo 2

- Representar o Círculo Osculador no espaço tridimensional utilizando o programa do círculo osculador no plano-xy (Programa 4).
Você encontra as informações sobre as pequenas mudanças necessárias em "Com lapis e papel".
- Usar como curva $x = \cos(t)$, $y = 2 \sin(t)$, $z = t$, e como "instante" de contato $t_1 = \pi/4$. Também incluir no gráfico o vetor de curvatura para $t = t_1$.

- `reset()`://círculo-osculador no espaço

```

t1:=PI/4:
x:=t->cos(t):
y:=t->2*sin(t):
z:=t->t:
scale:=5:
curva:=plot::Curve3d([x(t),y(t),z(t)],t=0..PI):
pos:=matrix([[x(t),y(t),z(t)]])://vetor-posição
v:=matrix([[x'(t),y'(t),z'(t)]])://vetor-velocidade
absolv:=sqrt(v[1]^2+v[2]^2+v[3]^2):
vt:=v/absolv://vetor-tangente-unitário
k:=diff(vt,t)/absolv://vetor de curvatura
raio:=1/sqrt(k[1]^2+k[2]^2+k[3]^2):
centro:=pos+k*raio^2:
vn:=k*raio://vetor-normal-unitário
t:=t1:
x1:=pos[1]:
y1:=pos[2]:
z1:=pos[3]:
x2:=x1+scale*k[1]:
y2:=y1+scale*k[2]:
z2:=z1+scale*k[3]:
circ:=centro+raio*(cos(t2)*vt+sin(t2)*vn):
x3:=circ[1]:
y3:=circ[2]:

```

```

z3:=circ[3]:
x4:=centro[1]:
y4:=centro[2]:
z4:=centro[3]:
circulo:=plot::Curve3d([x3,y3,z3],t2=0..2*PI,Color=RGB::Black):
pc:=plot::Point3d(x4,y4,z4,Color=RGB::Black,PointSize=2*unit:
:mm):
p:=plot::Point3d(x1,y1,z1,Color=RGB::Green,PointSize=2*unit:
:mm):
kc:=plot::Arrow3d([x1,y1,z1],[x2,y2,z2],Color=RGB::Red):
plot(curva,pc,p,kc,circulo,Scaling=Constrained)

```

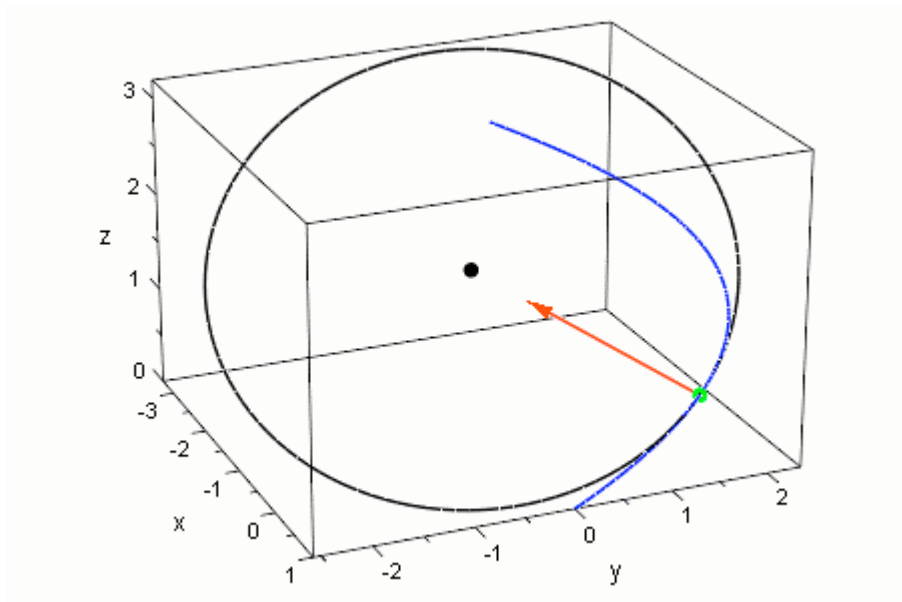


Fig. 3.4-15

3.4.6 Com lápis e papel

Derivadas Parciais e plano tangente.

A equação da *reta tangente* à curva $y = f(x)$ no ponto $(a, f(a))$ é -ver parágrafo 1.2.1-

$$y = f(a) + f'(a)(x-a) \quad (1)$$

No caso de uma função $z = f(x,y)$ temos um *plano tangente* no ponto (a,b) do domínio de $f(x,y)$ no plano-xy.

$(a, b, f(a,b))$ é o ponto de contato entre plano tangente e superfície (sabemos que a função $z = f(x,y)$ pode ser representada por uma superfície).

A equação $y = b$ representa um plano que corta a superfície, dando origem a uma curva C.

A derivada parcial f_x em (a, b) pode ser interpretada como a inclinação da reta tangente à curva C, ou seja, como sendo $\text{tg } \alpha$.

Por exemplo, seja $z = 4 - x^2 - 2y^2$ uma função definida num domínio D do plano-xy e seja $(2, -1)$ um ponto de D.

Para $y = -1$ (ou $+1$), temos $z = 2 - x^2$, de sorte que a derivada ao longo da curva C (fazer um gráfico!) é $f_x = -2x$; fazendo $x = 2$, vemos que $f_x(2, -1)$ é igual a -4 .

Define-se de modo análogo a derivada parcial $f_y(a,b)$. Em nosso caso, temos, para $x = 2$, $z = -2y^2$ e $f_y = -4y$; fazendo $y = -1$, resulta $f_y(2, -1) = 4$.

No caso de funções explícitas $z = f(x,y)$, se calcula essas derivadas de modo usual, pois trata-se sempre de derivar uma função de uma variável, sendo a outra variável tratada como uma constante.

Para a equação do **plano tangente** à superfície $z = f(x,y)$ no ponto (a, b) esperamos uma equação do tipo da Eq. (1), mais ampliada por um termo que toma em conta a inclinação na direção-y. Efetivamente, temos

$$z_t = f(a,b) + f_x(a,b)(x-a) + f_y(a,b)(y-b) \quad (2)$$

No caso da função $z = 4 - x^2 - 2y^2$ no ponto $(2, -1)$, confira a figura mais acima,

$$z_t = -2 + (-4)(x-2) + (+4)(y+1) = 10 - 4(x-y)$$

Círculo osculador no espaço.

Para representar o círculo osculador no espaço tridimensional à uma curva C , precisamos de dois vetores unitários que definem o plano no qual se encontra o círculo osculador. Podemos escolher os vetores unitários ortogonais \mathbf{t} e \mathbf{n} para tal fim.

Seja A o ponto de contato entre o círculo osculador e a curva C para $t = a$ (ou $t = t_1$). Qual é a equação deste círculo?

Os vetores \mathbf{t} e \mathbf{n} (no "instante" $t = a$ e trasladados no centro \mathbf{m} do círculo), são dois vetores unitários, assim como \mathbf{i} e \mathbf{j} .

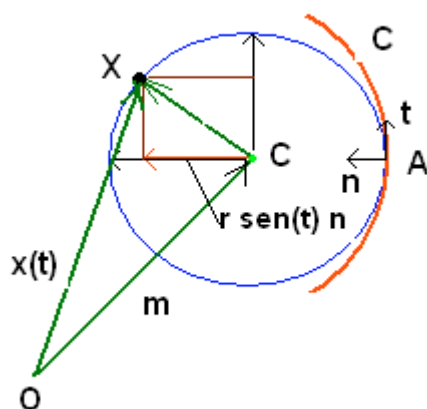


Fig. 3.4-16

Um ponto \mathbf{X} qualquer do círculo tem, respeito do centro C , as coordenadas $(r \cos(t), r \sin(t))$, ou seja, respeito da origem O ,

$$\mathbf{x}(t) = \mathbf{m} + \mathbf{C}\mathbf{X} = \mathbf{m} + r (\cos(t) \cdot \mathbf{t}(a) + \sin(t) \cdot \mathbf{n}(a)) \quad (3)$$

É esta a equação do círculo osculador que foi usado no último programa, onde

```
circ:=centro+raio*(cos(t2)*vt+sin(t2)*vn):
```

é o vetor $\mathbf{x}(t)$. O parâmetro t_2 foi introduzido para fazer o gráfico do círculo por meio da instrução

```
circulo:=plot::Curve3d([x3,y3,z3],t2=0..2*PI,Color=RGB::Black):
```

x3:=circ[1,1] é a primeira coordenada do vetor $\mathbf{x}(t)$.

Parametrização de planos e retas no 3-espaço

No parágrafo anterior utilizamos os vetores unitários \mathbf{t} e \mathbf{n} para definir um plano no 3-espaço. Equação (3) é uma equação vetorial paramétrica de um círculo de raio r nesse plano. O vetor \mathbf{m} foi o vetor posição do centro do círculo.

Considere agora um **plano** E contendo dois vetores não paralelos \mathbf{u} e \mathbf{v} e um ponto P_0 com o vetor posição \mathbf{r}_0 . Podemos chegar a qualquer ponto do plano partindo de P_0 e caminhando paralelamente a \mathbf{u} e \mathbf{v} , somando múltiplos deles a \mathbf{r}_0 , ver esboço 3.4-17.

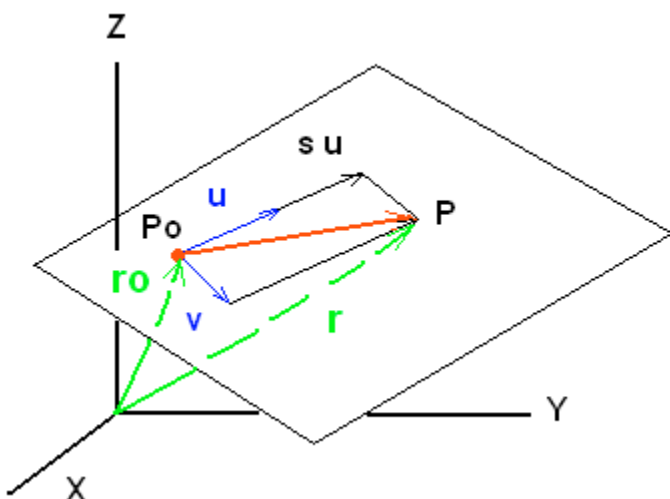


Fig. 3.4-17

Assim temos a seguinte equação vetorial paramétrica do plano E

$$E: \mathbf{r}(s,t) = \mathbf{r}_0 + s \cdot \mathbf{u} + t \cdot \mathbf{v} \quad (4)$$

Cada escolha dos parâmetros (s, t) marca um ponto P no plano E . Se $\mathbf{r}_0 = x_0\mathbf{i} + y_0\mathbf{j} + z_0\mathbf{k}$ e $\mathbf{u} = a_1\mathbf{i} + a_2\mathbf{j} + a_3\mathbf{k}$ e $\mathbf{v} = b_1\mathbf{i} + b_2\mathbf{j} + b_3\mathbf{k}$, então as equações paramétricas do plano E podem ser escritas na forma:

$$x = x_0 + s a_1 + t b_1, \quad y = y_0 + s a_2 + t b_2, \quad z = z_0 + s a_3 + t b_3 \quad (5)$$

Observe que a parametrização do plano expressa as coordenadas x , y e z como funções lineares dos parâmetros s e t .

Exemplo 1 (plano):

Escreva uma equação paramétrica para o plano pelo ponto $(2, -1, 3)$ e contendo os vetores $\mathbf{u} = 2\mathbf{i} + 3\mathbf{j} - \mathbf{k}$ e $\mathbf{v} = \mathbf{i} - 4\mathbf{j} + 5\mathbf{k}$.

Solução:

A equação paramétrica é $\mathbf{r}(s,t) = 2\mathbf{i} - \mathbf{j} + 3\mathbf{k} + s(2\mathbf{i} + 3\mathbf{j} - \mathbf{k}) + t(\mathbf{i} - 4\mathbf{j} + 5\mathbf{k})$, ou seja: $\mathbf{r}(s,t) = (2+2s+t)\mathbf{i} + (-1+3s-4t)\mathbf{j} + (3-s+5t)\mathbf{k}$. Em forma de coordenadas:

$$x = 2 + 2s + t, \quad y = -1 + 3s - 4t, \quad z = 3 - s + 5t$$

Um plano paralelo ao plano- yz tem uma equação da forma $x = k$, onde k é uma constante. Na seguinte figura estão representadas os planos paralelos aos planos de coordenadas juntos com o gráfico da função $z = 4 - 0.5(x^2 + y^2)$. O ponto P_0 está sempre sobre um eixo numa distância 2; os vetores \mathbf{u} e \mathbf{v} são os vetores unitários \mathbf{i} , \mathbf{j} , \mathbf{k} . Por exemplo: O plano paralelo ao plano- xz tem $P_0 = (0, 2, 0)$, $\mathbf{u} = (1, 0, 0)$, $\mathbf{v} = (0, 0, 1)$. As equações paramétricas são $x = s$, $y = 2$, $z = t$. Para os plots dos 3 planos utilizamos a função `plot::Surface`, e para a função $z = 4 - 0.5(x^2 + y^2)$ utilizamos `plot::Function3d`.

- `reset() // 3 planos`

```
superficie:=-0.5*x^2-0.5*y^2+4:
```

```
f:=plot::Function3d(superficie,x=-4..4,y=-4..4):
```

```
E1:=plot::Surface([2,s,t], s=-4..4,
t=-12..4,Color=RGB::Green):
```

```
E2:=plot::Surface([s,2,t], s=-4..4,
t=-12..4,Color=RGB::Blue):
```

```
E3:=plot::Surface([s,t,2], s=-4..4,
t=-4..4,Color=RGB::Yellow):
```

```
plot(f,E1,E2,E3)
```

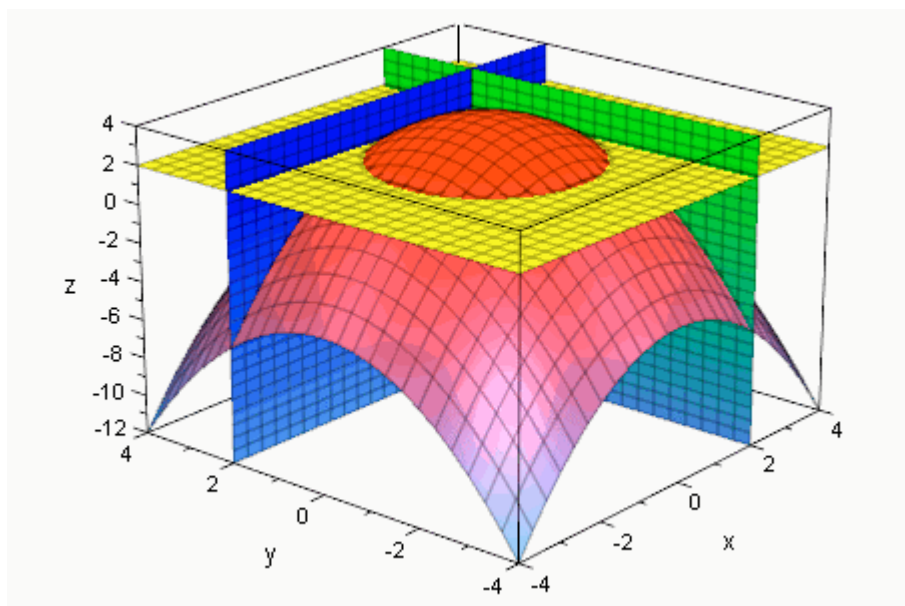


Fig. 3.4-18

Exemplo 2 (reta):

Agora considere uma **reta** g na direção do vetor \mathbf{v} passando pelo ponto (x_0, y_0, z_0) com vetor posição \mathbf{r}_0 . Começamos em \mathbf{r}_0 e nos movemos pela reta somando diferentes múltiplos de \mathbf{v} . Desta forma todo ponto da reta pode ser escrito como

$$\mathbf{g}: \mathbf{r}(t) = \mathbf{r}_0 + t \cdot \mathbf{v} \quad (6)$$

Se $\mathbf{v} = a\mathbf{i} + b\mathbf{j} + c\mathbf{k}$, obtemos as seguintes equações paramétricas da reta:

$$x(t) = x_0 + t \cdot a, \quad y(t) = y_0 + t \cdot b, \quad z(t) = z_0 + t \cdot c \quad (7)$$

t pode ser um número real qualquer (isto é, t pode assumir todos os valores no intervalo $(-\infty, +\infty)$).

Fazemos agora um esboço da reta que é paralela a um dado vetor (2,5,3) e que contém o ponto (7,8,9) .

As equações paramétricas são

$$x = 7 + 2 \cdot t, \quad y = 8 + 5 \cdot t, \quad z = 9 + 3 \cdot t$$

- `reset() //reta no 3-espaço`

```
dir:=plot::Arrow3d([0,0,0],[2,5,3],Color=RGB::Red)//direção
```

```
reta:=plot::Curve3d([7+2*t,8+5*t,9+3*t],  
t=-4..1,Color=RGB::Green):
```

```
ponto:=plot::Point3d(7,8,9,Color=RGB::Black,  
PointSize=3*unit::mm):
```

```
plot(dir,reta,ponto)
```

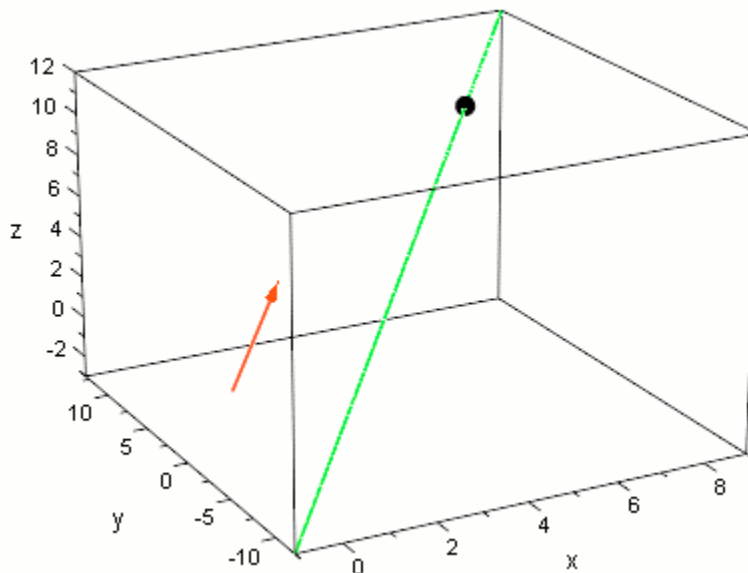


Fig. 3.4-19

3.4.7 Planos em forma de coordenadas cartesianas

MuPAD oferece, a partir da versão 3, a nova função `plot::Implicit3d` que é ideal para fazer gráficos de planos dados em forma implícita.

A equação de um plano que passa pelo ponto (x_0, y_0) é dada pela equação

$$z = z_0 + m(x - x_0) + n(y - y_0) \quad (6)$$

m é a inclinação do plano na direção- x e n é sua inclinação na direção- y . A Eq. (6) é bem como a equação de uma reta, exceto que há duas inclinações em vez de uma. Com a Eq. (6) calculamos a coordenada- z do ponto do plano diretamente sobre o ponto (x, y) no plano- xy .

A seguir alguns **exemplos** com planos verticais:

- `plot(plot::Implicit3d(x + y + 5,`
`x = -5..5,`
`y = -5..5,`
`z = -5..5))`

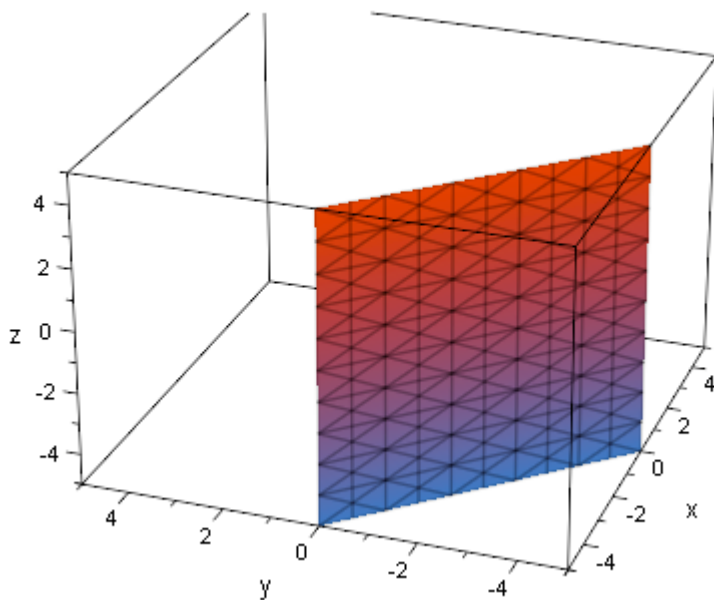


Fig. 3.4-20

- `plot(plot::Implicit3d(x + 2,`
`x = -5..5,`
`y = -5..5,`
`z = -5..5))`

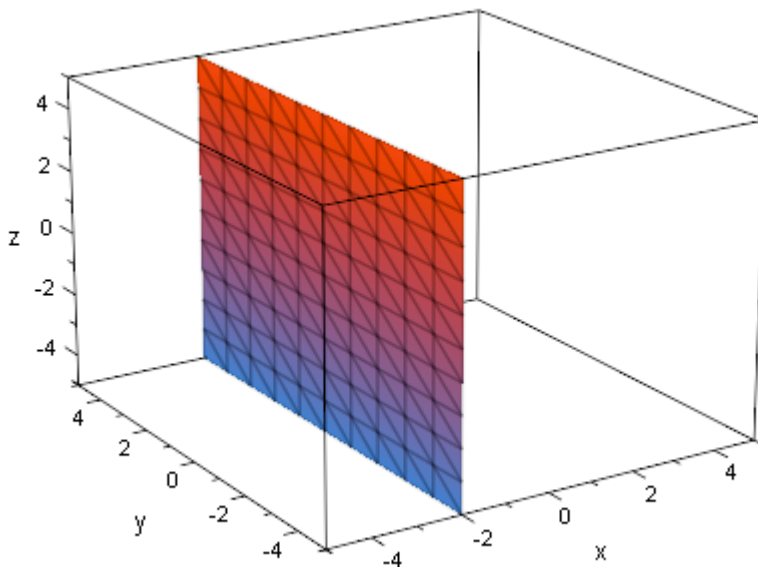


Fig. 3.4-21

Na seguinte figura vemos uma cabana limitada pelos planos verticais $x = 0$, $x = 8$ (verde), $y = 0$ e $y = 16$ (azul). O teto é formado pelo plano $z = 12 - x/4 - y/8$ (vermelho). Cortando os planos, obtemos para as alturas dos quatro cantos 10, 12, 10 e 8 metros, respectivamente. Na próxima seção vamos calcular o volume desta cabana forma.

- `p1:=plot::Implicit3d(z +x/4+y/8-12,`
`x = 0..10,`
`y = 0..18,`
`z = 0..15):`

```
p2:=plot::Implicit3d(y - 16,  
                    x = 0..10,  
                    y = 0..18,  
                    z = 0..10):  
p3:=plot::Implicit3d(x - 8,  
                    x = 0..10,  
                    y = 0..18,  
                    z = 0..10):  
p1:=Color:=RGB::Red:  
p2:=Color:=RGB::Blue:  
p3:=Color:=RGB::Green:  
plot(p1,p2,p3)
```

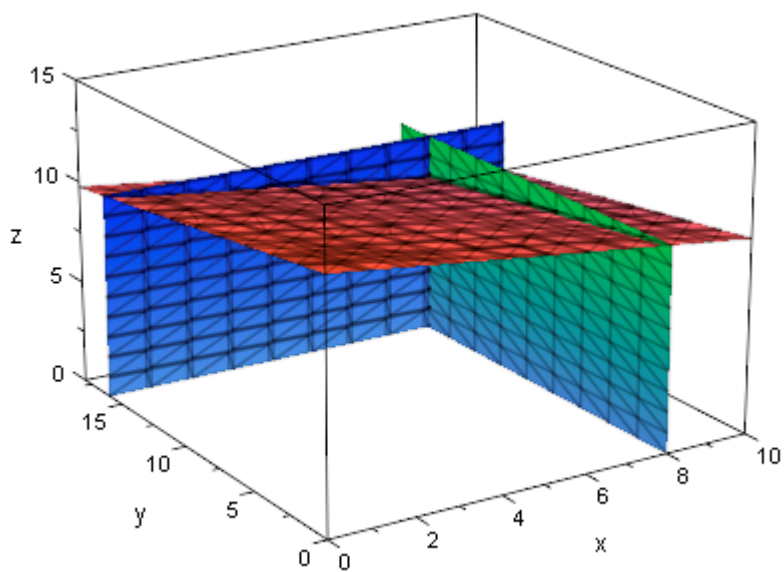


Fig. 3.4-22