

Visualización y Simulaciones en MAPLE

Luis A. Núñez

Escuela de Física, Universidad Industrial de Santander, Bucaramanga Colombia

Departamento de Física, Facultad de Ciencias, Universidad de Los Andes, Mérida 5101, Venezuela

e-mail: lnunez@uis.edu.co

<http://webdelprofesor.ula.ve/ciencias/nunez>

actualizado: Octubre 2015

Movimiento Vertical: Trayectorias de Gotas, Balas, Corchos y Burbujas en Fluidos

Esta hoja de trabajo muestra la evolución de un gota, una bala y una burbuja lanzadas (o dejadas caer) verticalmente en presencia del resistencia del aire o un fluido cualquiera.. El movimiento se describe mediante la ecuación diferencial y se modela el efecto de la resistencia del aire.

Para el modelaje tomamos:

- ρ_0 la densidad del fluido y ρ_1 la densidad del cuerpo (gotas/balas/conchos/burbujas) en Kilogramos/ m^3
- K es el coeficiente de fricción que depende de la forma de cuerpo. En el caso de una esfera en un fluido $K=6\pi R$ y
- η es el coeficiente de fricción que depende de la viscosidad del fluido. En el sistema MKS se expresa en $N\ s/g/m^2$ y se mide en Poise que es igual a un décimo de la unidad en MKS.
- g es la aceleración de gravedad, la cual consideramos constante igual a 9.8 metros por segundo cuadrado
- $y(t)$ es la altura del movil (gotas, balas, corchos o burbujas) en un tiempo t .

La evolución en el tiempo viene gobernada por una ecuación diferencial de segundo orden.

```
> restart;  
> m0:=(4*Pi/3)*rho0*R^3;  
m:=(4*Pi/3)*rho1*R^3;  
rho0:=xi*rho;  
rho1:=phi*rho;
```

$$m_0 := \frac{4}{3} \pi \rho_0 R^3$$

$$m := \frac{4}{3} \pi \rho_l R^3$$

$$\rho_0 := \xi \rho$$

$$\rho_l := \phi \rho \quad (1.1)$$

Donde ξ y ϕ representan las densidades relativas del fluido y del cuerpo respecto al agua (de densidad ρ), respectivamente.

Para el caso de una gota de lluvia cayendo, una pelota lanzada hacia arriba o una burbuja (o un corcho) que se despegue desde el fondo

de un vaso de refresco, la ecuación diferencial que describe el movimiento puede ser escrita como

> **emov := m*diff(y(t),t\$2) = m*g - K*eta*diff(y(t),t)-m0*g ;**
K:=6*Pi*R;

$$emov := \frac{4}{3} \pi \phi \rho R^3 \left(\frac{d^2}{dt^2} y(t) \right) = \frac{4}{3} \pi \phi \rho R^3 g - K \eta \left(\frac{d}{dt} y(t) \right) - \frac{4}{3} \pi \xi \rho R^3 g$$

$$K := 6 \pi R \quad (1.2)$$

Tenemos tres fuerzas presentes: el peso, la resistencia del aire y la fuerza de Arquímedes (empuje o flotación).

La solución general será:

> **soluciongeneral := dsolve(emov, y(t));**

$$soluciongeneral := y(t) = -\frac{2}{9} \frac{\phi \rho R^2 e^{-\frac{9}{2} \frac{\eta t}{R^2 \phi \rho}} _C1}{\eta} + \frac{2}{9} \frac{R^2 \rho g (\phi - \xi) t}{\eta} + _C2 \quad (1.3)$$

Donde $_C1$ y $_C2$ son constantes a ser determinadas por las condiciones iniciales. Es de hacer notar que, primeramente, una misma ecuación diferencial describe una variedad de fenómenos físicos.

Igualmente, es importante señalar que la ecuación queda parametrizada $K \eta$, ξ, ρ y R . Estas cantidades nos permiten modelar el fenómeno.

Para el caso de la gota de agua supondremos que parte del reposo, por lo tanto la solución general será:

> **solucionparticular := dsolve({emov,y(0)=0,D(y)(0)=v0}, y(t));**

$$solucionparticular := y(t) = \frac{2}{81} \frac{\phi \rho R^2 e^{-\frac{9}{2} \frac{\eta t}{R^2 \phi \rho}} (-9 v_0 \eta + 2 R^2 \phi \rho g - 2 R^2 \xi \rho g)}{\eta^2} \quad (1.4)$$

$$+ \frac{2}{9} \frac{R^2 \rho g (\phi - \xi) t}{\eta} - \frac{2}{81} \frac{(-9 v_0 \eta + 2 R^2 \phi \rho g - 2 R^2 \xi \rho g) \phi \rho R^2}{\eta^2}$$

La posición y la velocidad vendrán dadas por

> **posicion:=rhs(solucionparticular);**

$$\begin{aligned}
 \text{posicion} := & \frac{2}{81} \frac{\phi \rho R^2 e^{-\frac{9}{2} \frac{\eta t}{R^2 \phi \rho}} (-9 v_0 \eta + 2 R^2 \phi \rho g - 2 R^2 \xi \rho g)}{\eta^2} \\
 & + \frac{2}{9} \frac{R^2 \rho g (\phi - \xi) t}{\eta} - \frac{2}{81} \frac{(-9 v_0 \eta + 2 R^2 \phi \rho g - 2 R^2 \xi \rho g) \phi \rho R^2}{\eta^2}
 \end{aligned} \tag{1.5}$$

> **velocidad:=diff(posicion,t);**

$$\begin{aligned}
 \text{velocidad} := & -\frac{1}{9} \frac{e^{-\frac{9}{2} \frac{\eta t}{R^2 \phi \rho}} (-9 v_0 \eta + 2 R^2 \phi \rho g - 2 R^2 \xi \rho g)}{\eta} \\
 & + \frac{2}{9} \frac{R^2 \rho g (\phi - \xi)}{\eta}
 \end{aligned} \tag{1.6}$$

El cambio de signo en las fuerzas indica que en algún momento la fuerza de fricción y el empuje anularán al peso. En ese caso estaremos hablando una velocidad límite. Es decir de una velocidad a partir de la cual el móvil cae sin aceleración.

> **ecuac1:=0=rhs(emov);**

vellimite:= simplify((4/3*Pi*(rho1-rho0)*R^3*g)/(6*Pi*R*eta));

$$\text{vellimite} := \frac{2}{9} \frac{R^2 \rho g (\phi - \xi)}{\eta} \tag{1.7}$$

La velocidad límite será constante. En el caso de una gota de lluvia la densidad del aire respecto al agua es $1,30 \cdot 10^{-3}$, y consideremos el diámetro de la gota de agua $1\text{mm} = 10^{-3}\text{m}$.

> **gota:= [g = 9.8, R=0.001, rho=10^3, xi=1.3*10^(-3), phi=1, v0=0, eta=0.01894*10^(-3)];**

vellim:=subs(gota,vellimite);

$$\text{vellim} := 114.8335093 \tag{1.8}$$

La gota tiene una velocidad límite de 120 mts/sg o

> **vellimiteKmh:=vellim*(0.001 / ((1/60)*(1/60)));**

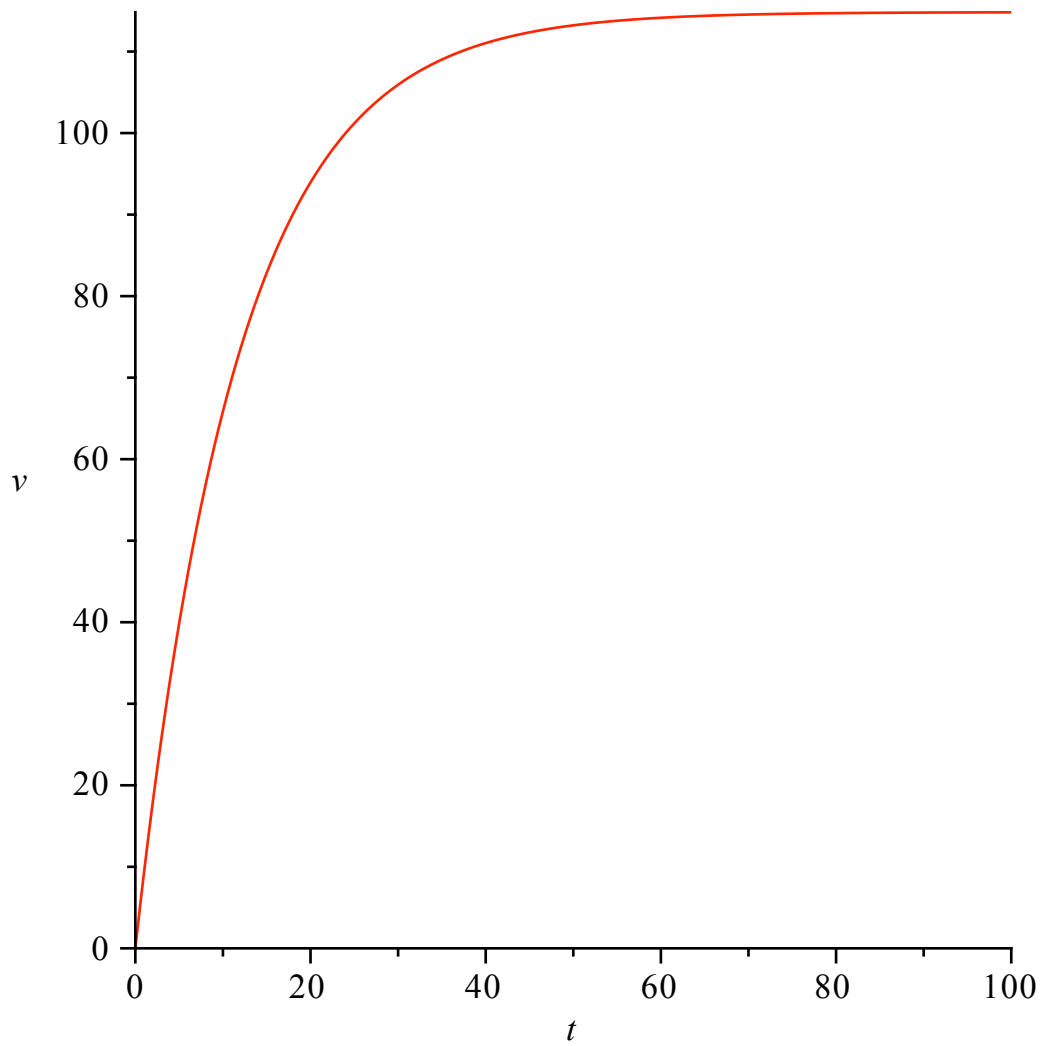
$$\text{vellimiteKmh} := 413.4006335 \tag{1.9}$$

° casi 500 Kmh !

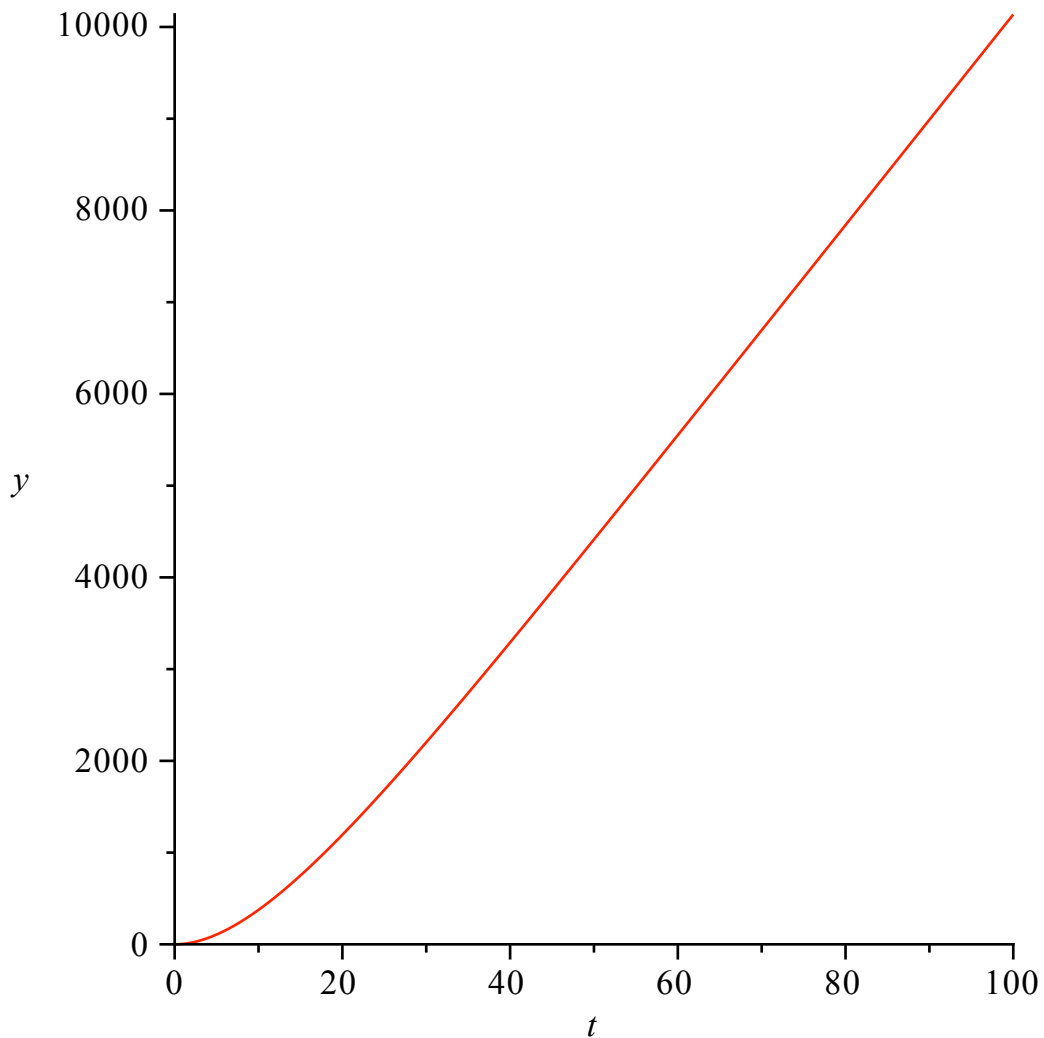
> **vell:=subs(gota,velocidad);**

plot(vell,t=0..100,labels=[t,v]);

$$\text{vell} := -114.8335093 e^{-0.08523000000 t} + 114.8335093$$

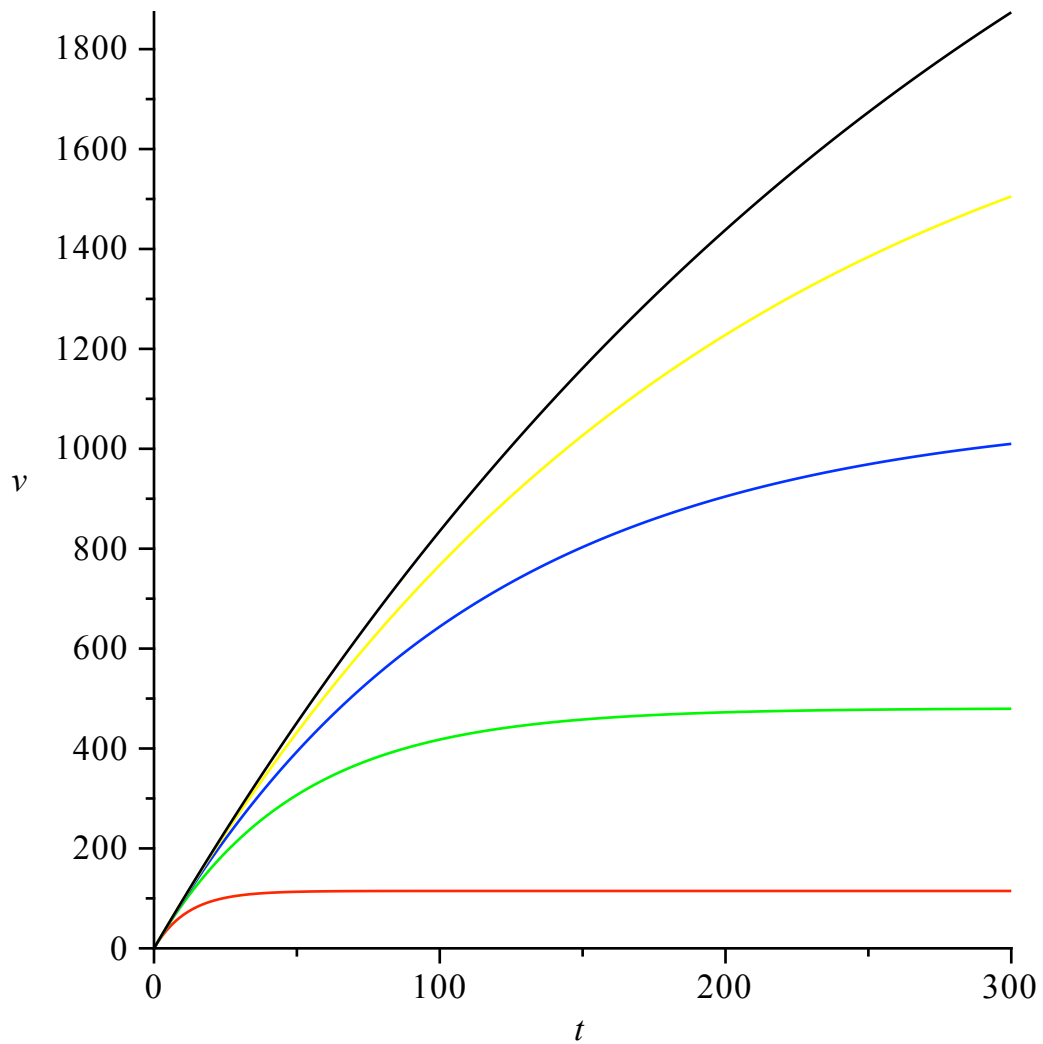


```
> pos1:=subs(gota, posicion);  
plot(pos1,t=0..100,labels=[t,y]);  
pos1 := 1347.336728 e-0.08523000000 t + 114.8335093 t - 1347.336728
```

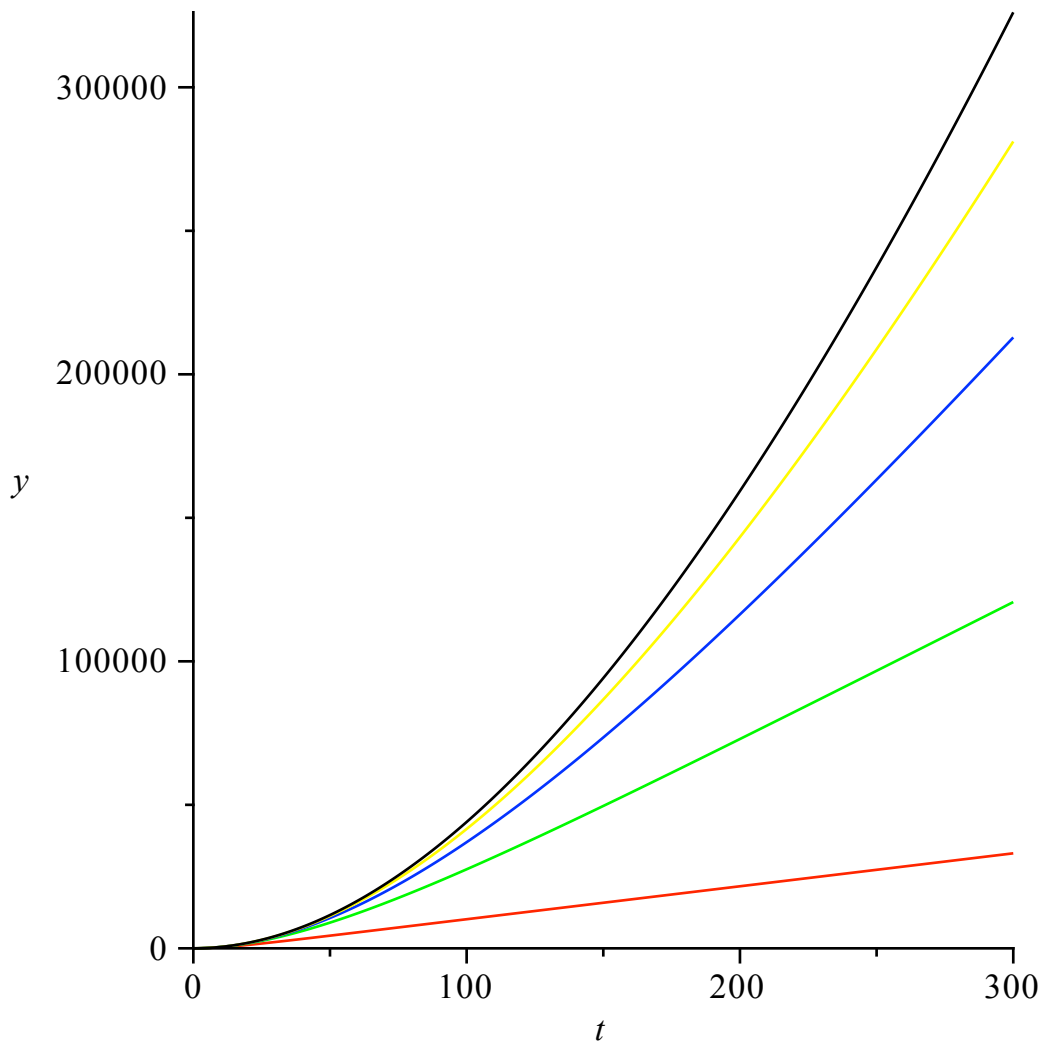


Si la gota crece a 0,2, 0,3, 0,4, 0,5 mm tendremos

```
> gota2:= [g = 9.8, R=0.002,rho=10^3, xi=1.3*10^(-3), phi=1,v0=0,
eta=1.81*10^(-5)]:
gota3:= [g = 9.8, R=0.003,rho=10^3, xi=1.3*10^(-3),phi=1, v0=0,
eta=1.81*10^(-5)]:
gota4:= [g = 9.8, R=0.004,rho=10^3,phi=1, xi=1.3*10^(-3), v0=0,
eta=1.81*10^(-5)]:
gota5:= [g = 9.8, R=0.005,rho=10^3, xi=1.3*10^(-3), phi=1,v0=0,
eta=1.81*10^(-5)]:
> vel2:=subs(gota2,velocidad):
vel3:=subs(gota3,velocidad):
vel4:=subs(gota4,velocidad):
vel5:=subs(gota5,velocidad):
plot([vel1,vel2,vel3,vel4,vel5],t=0..300,labels=[t,v],color=
[red,green,blue,yellow,black]);
```



```
> pos2:=subs(gota2, posicion):  
pos3:=subs(gota3, posicion):  
pos4:=subs(gota4, posicion):  
pos5:=subs(gota5, posicion):  
plot([pos1, pos2, pos3, pos4, pos5], t=0..300, labels=[t, y], color=  
[red, green, blue, yellow, black]);
```



Consideremos el caso de una bala disparada verticalmente hacia arriba. Una bala la podemos aproximar por una esfera de 10 mm con 15 gr de masa y se dispara con una velocidad de 600 m/s. La densidad del hierro relativa al agua es 7,86

```
> velocidadesr:=v0-g*t;
posicionsr:=v0*t-g*t^2/2;
```

$$velocidadsr := v0 - g t$$

$$posicionsr := v0 t - \frac{1}{2} g t^2 \quad (1.10)$$

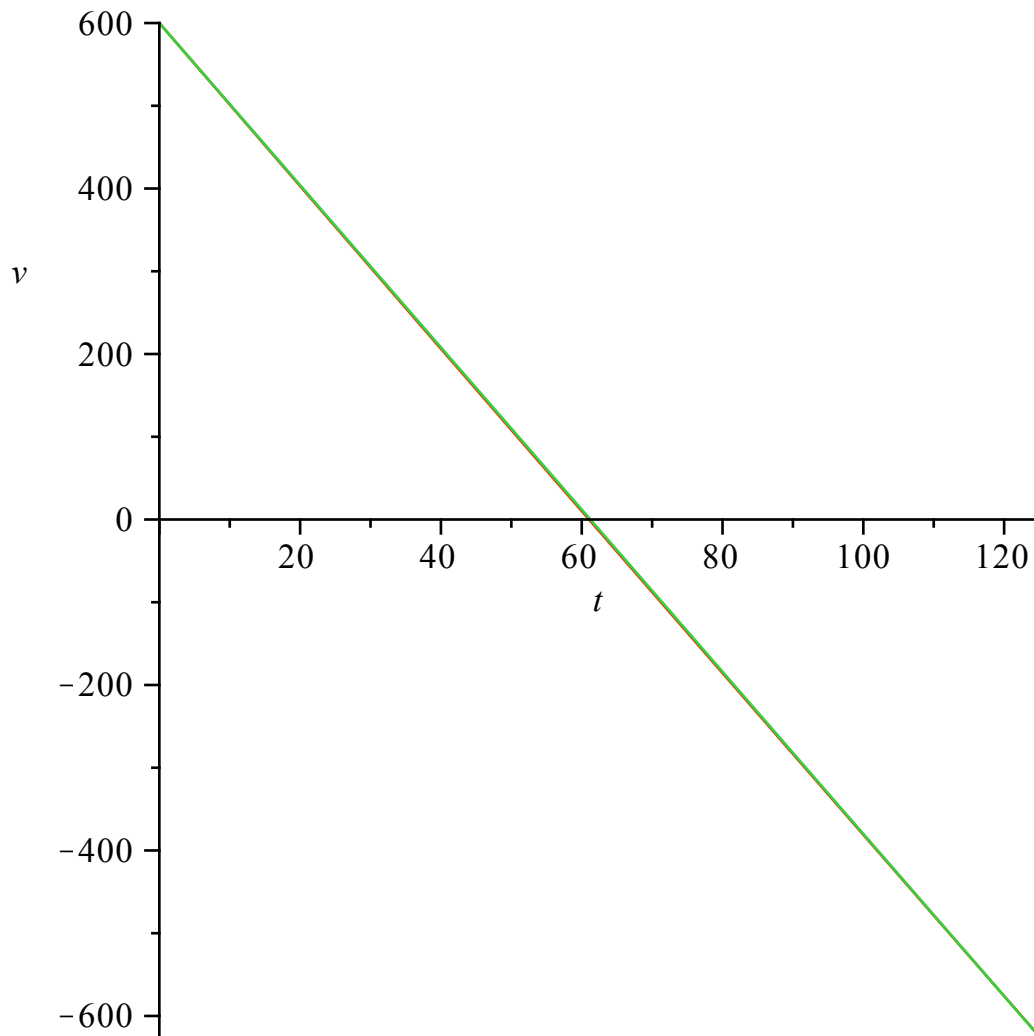
```
> balasr:=[g=9.8,v0=600]:
velsr:=subs(balasr,velocidadsr);
possr:=subs(balasr,posicionsr);
```

$$velsr := 600 - 9.8 t$$

$$possr := 600 t - 4.900000000 t^2 \quad (1.11)$$

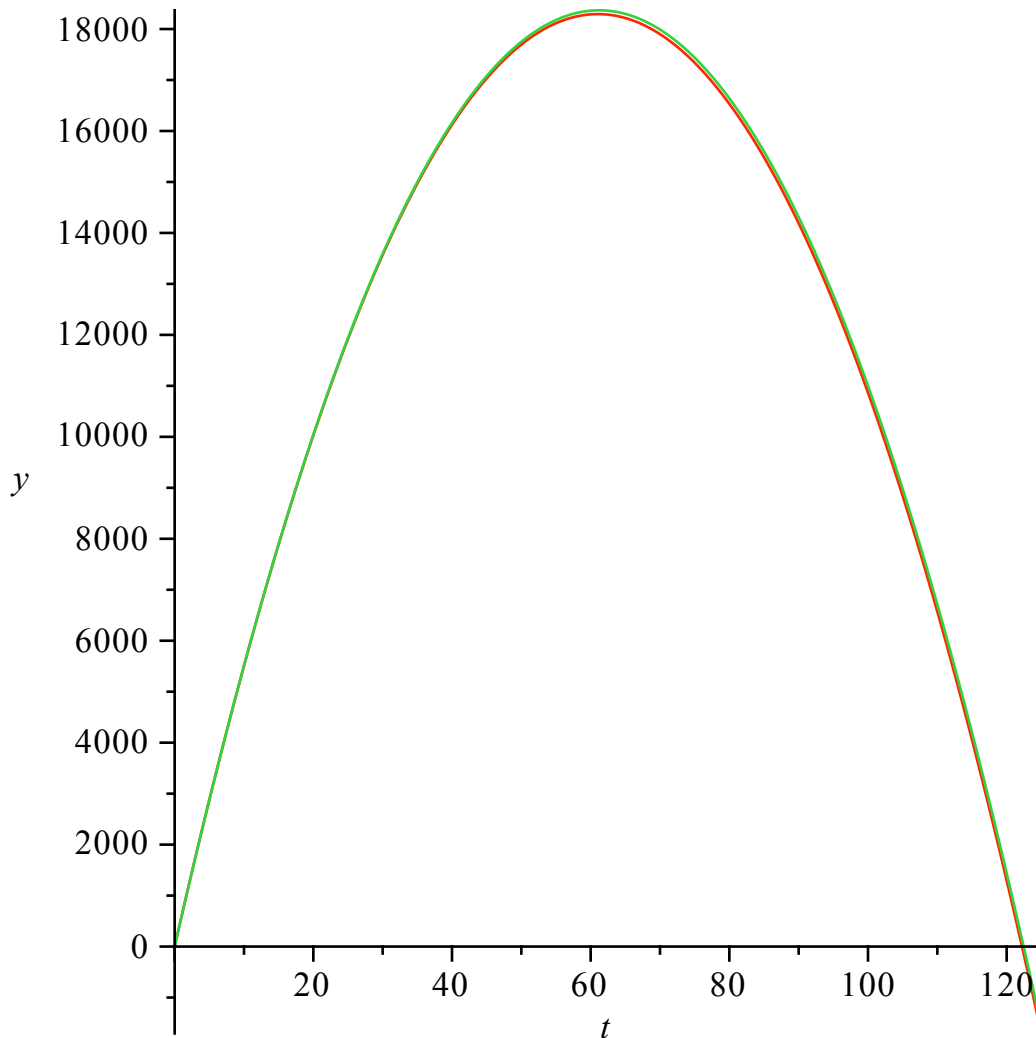
```
> bala:= [g = -9.8, R=0.01,rho=10^3, xi=1.3*10^(-3), phi=7.86,
v0=600,eta=1.81*10^(-5)]:
vell:=subs(bala,velocidad); plot([vell,velsr],t=0..125,labels=
[t,v]);
```

$$vell := 95155.26089 e^{-0.0001036259542 t} - 94555.26089$$



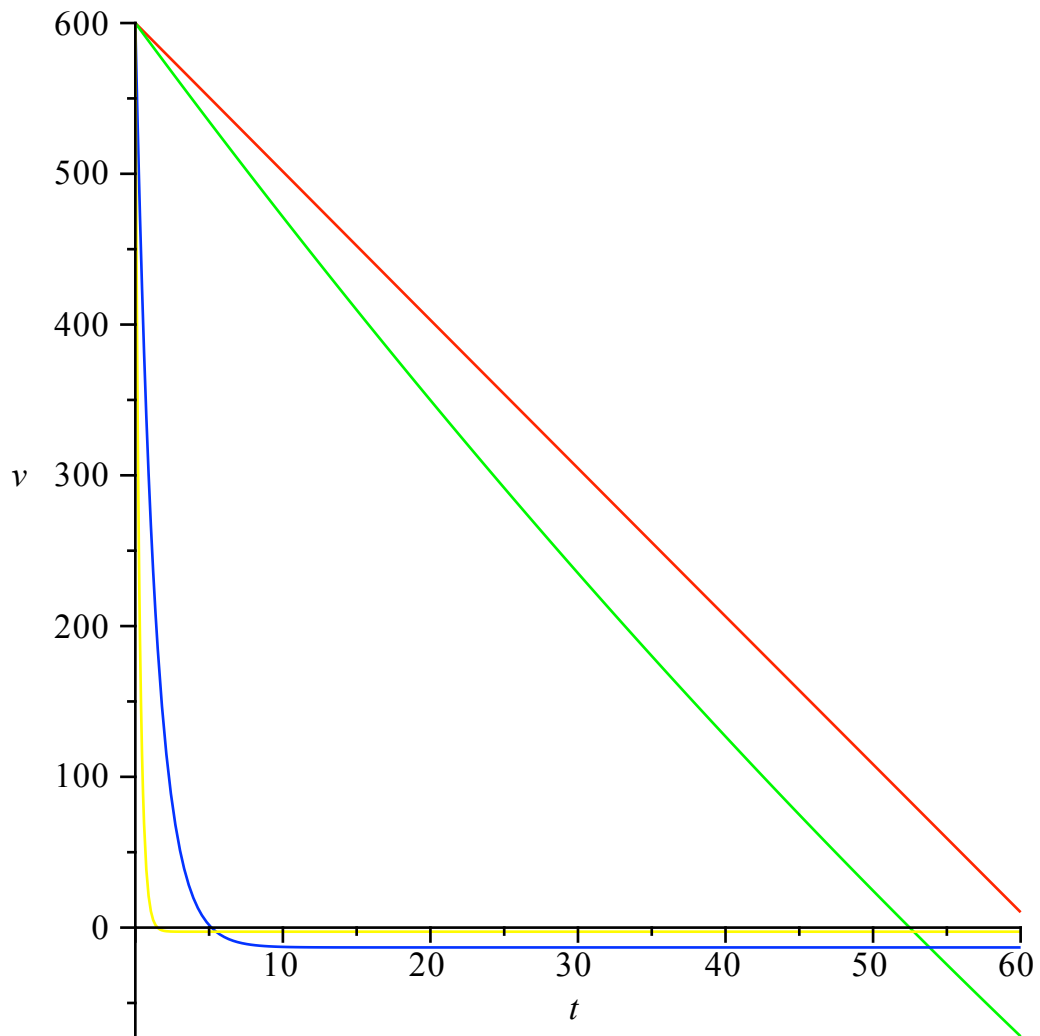
```
> pos1:=subs(bala,posicion); plot([pos1,possr],t=0..125,labels=[t,y]);
```

$$pos1 := -9.182570296 \cdot 10^8 e^{-0.0001036259542 t} - 94555.26089 t + 9.182570296 \cdot 10^8$$

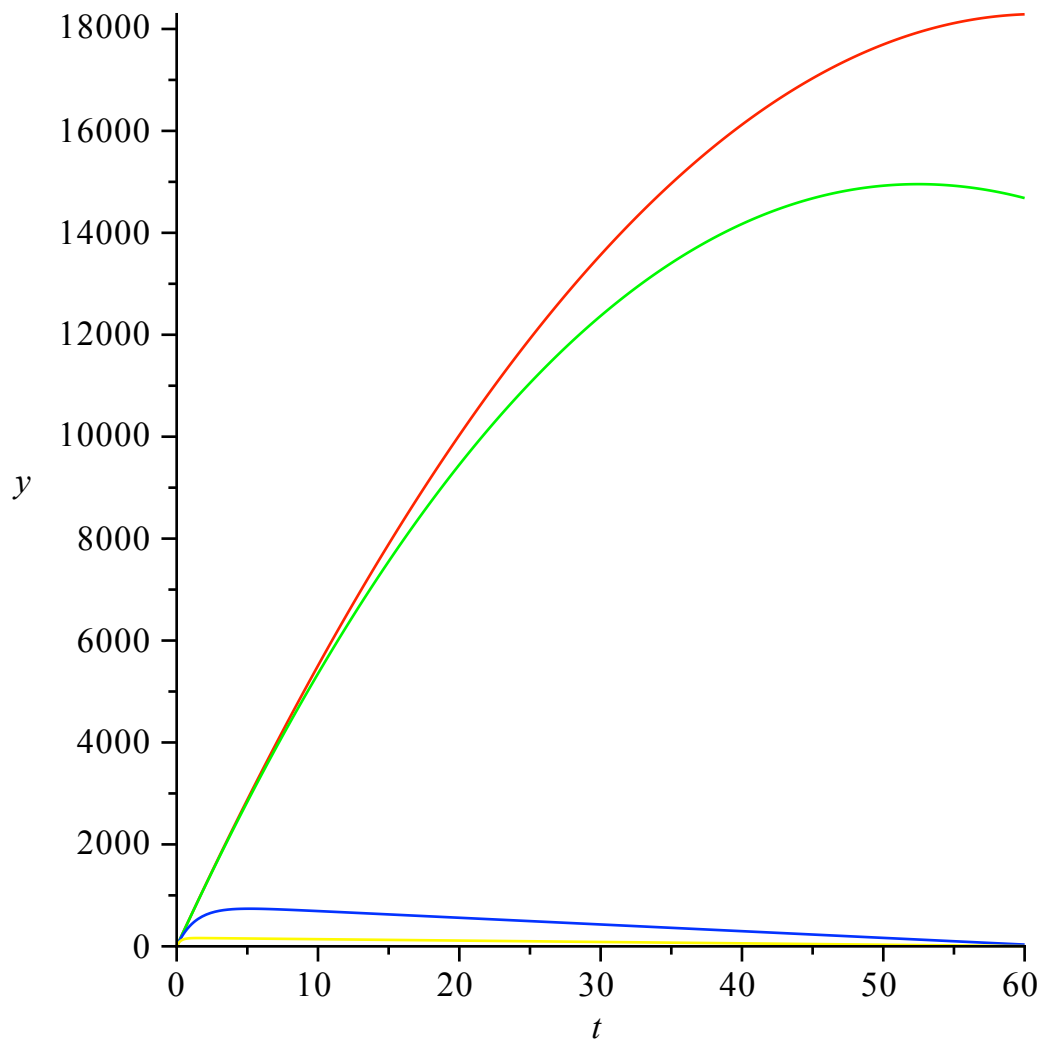


Si aumentamos la viscosidad del liquido a agua, aceite y glicerina

```
> bala2:= [g = -9.8, R=0.01,rho=10^3, xi=1.3*10^(-3), phi=7.86,
v0=600,eta=1.002*10^(-3)]:
bala3:= [g = -9.8, R=0.01,rho=10^3, xi=1.3*10^(-3), phi=7.86,
v0=600,eta=130*10^(-3)]:
bala4:= [g = -9.8, R=0.01,rho=10^3, xi=1.3*10^(-3), phi=7.86,
v0=600,eta=629*10^(-3)]:
> vel2:=subs(bala2,velocidad):
vel3:=subs(bala3,velocidad):
vel4:=subs(bala4,velocidad):
plot([vel1,vel2,vel3,vel4],t=0..60,labels=[t,v],color=[red,
green,blue,yellow]);
```

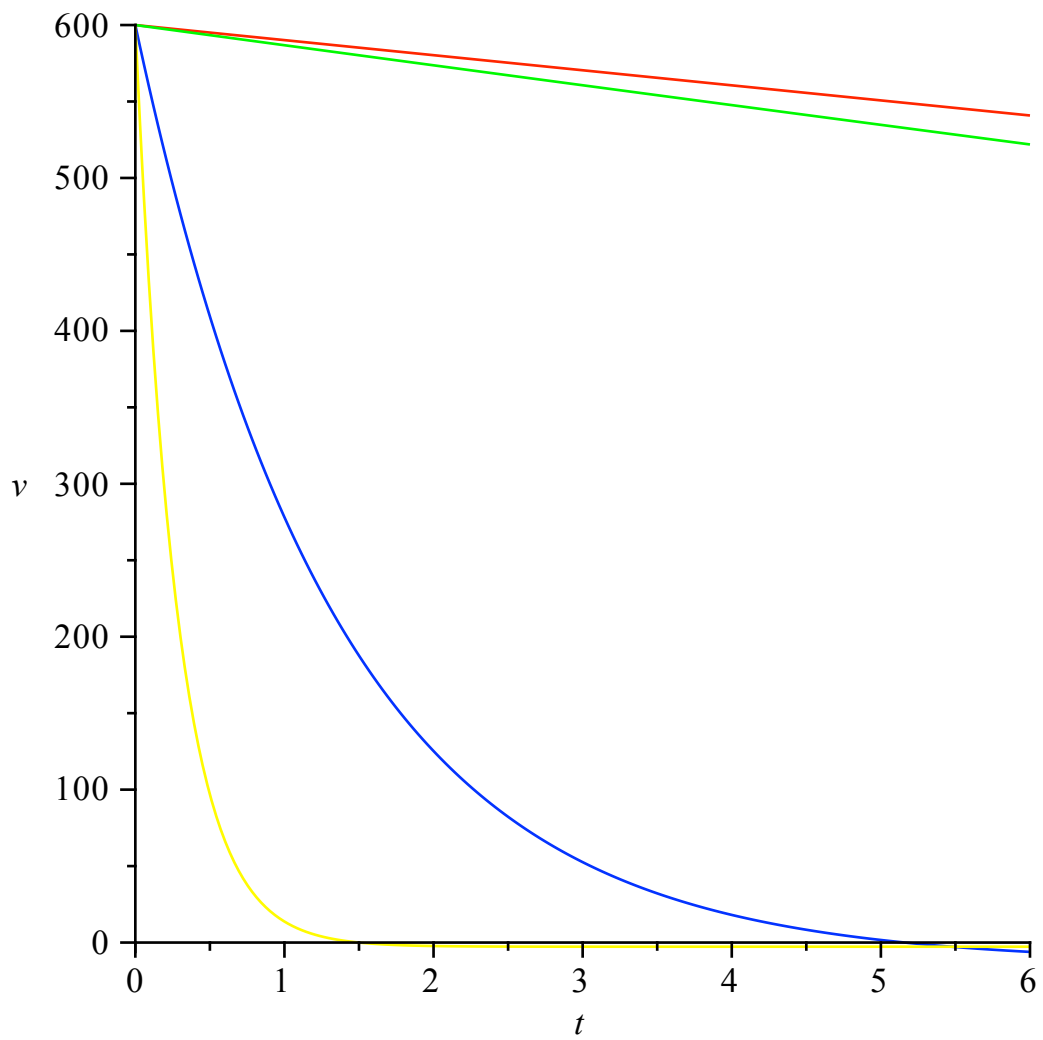


```
> pos2:=subs(bala2, posicion):  
pos3:=subs(bala3, posicion):  
pos4:=subs(bala4, posicion):  
plot([pos1, pos2, pos3, pos4], t=0..60, labels=[t, y], color=[red,  
green, blue, yellow]);
```

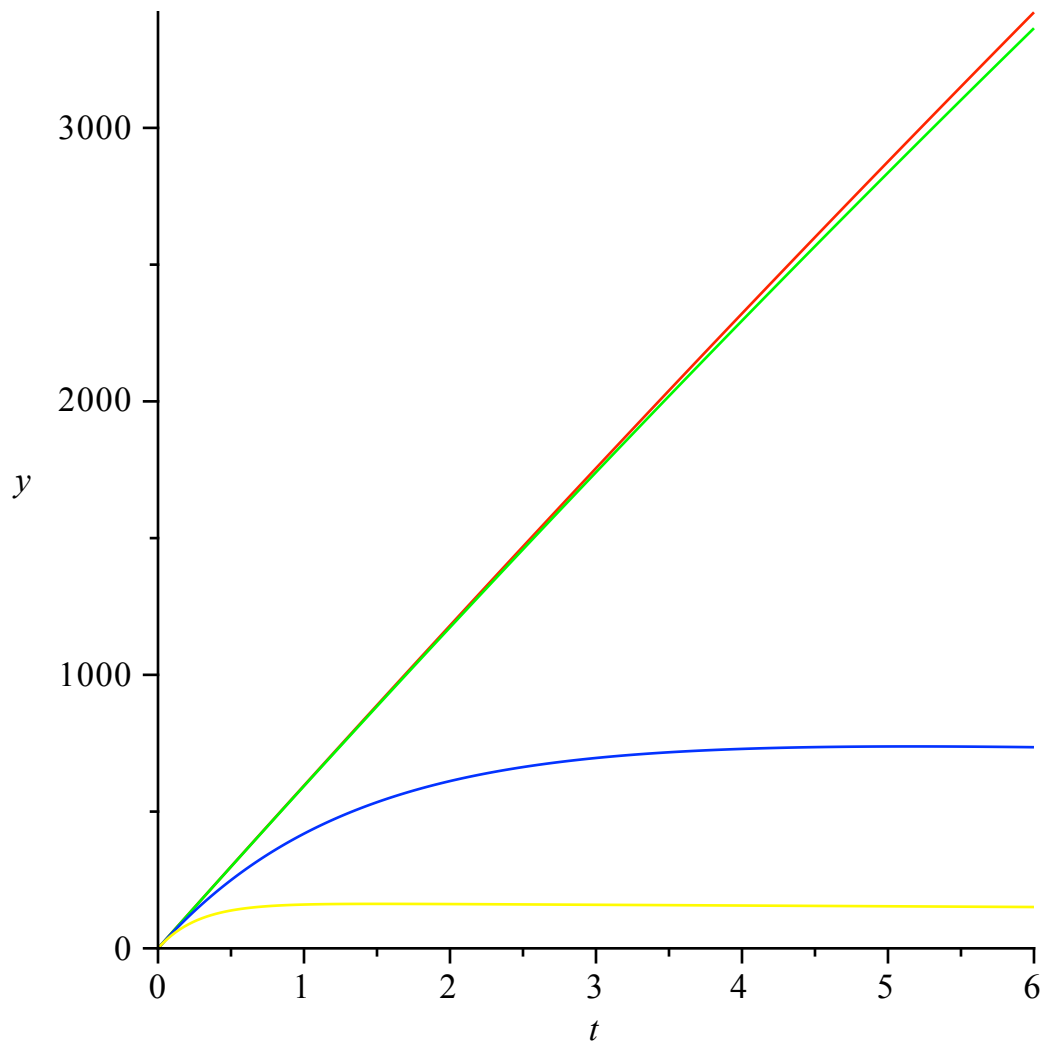


recortando la escala de tiempo a t= 6sg

```
> plot([vel1,vel2,vel3,vel4],t=0..6,labels=[t,v],color=[red,  
green,blue,yellow]);
```

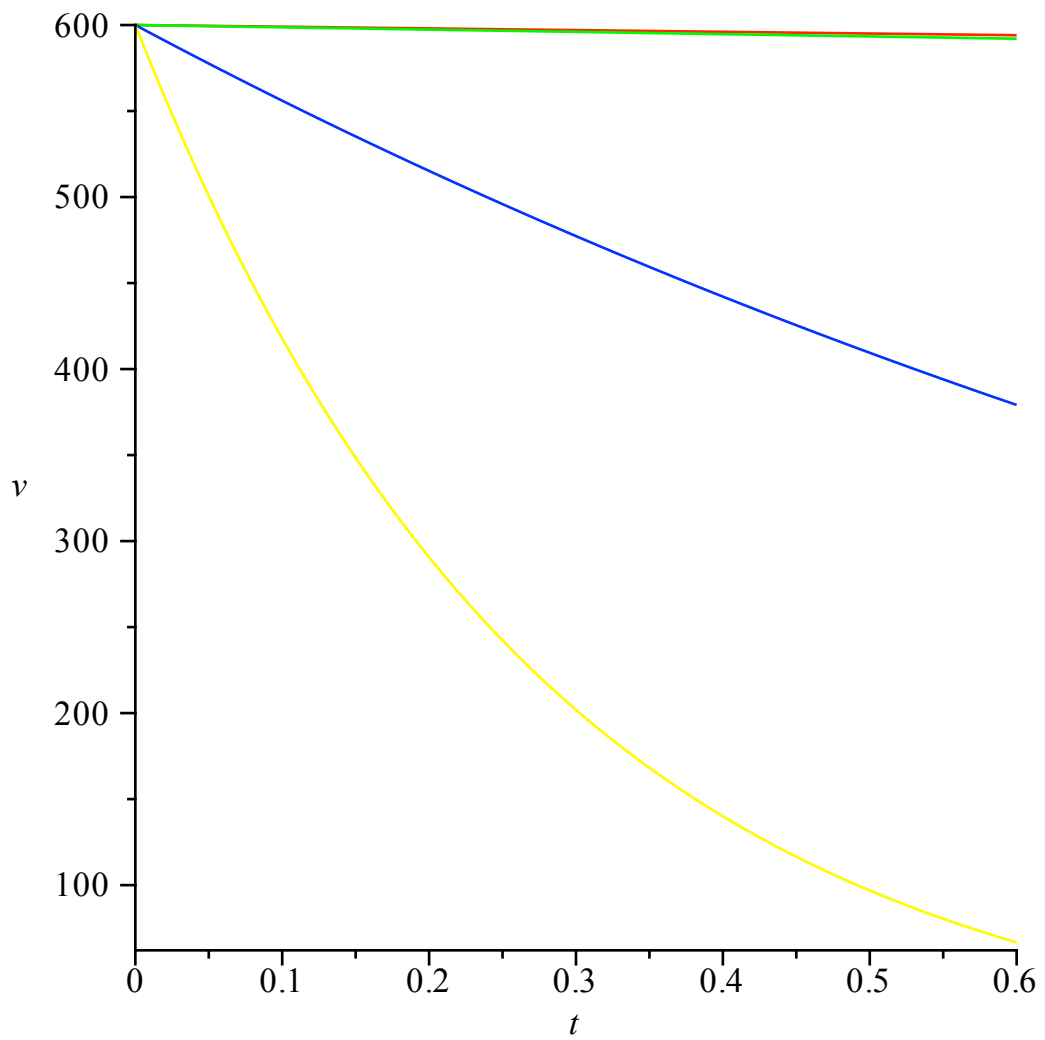


```
> plot([pos1, pos2, pos3, pos4], t=0..6, labels=[t, y], color=[red,  
green, blue, yellow]);
```

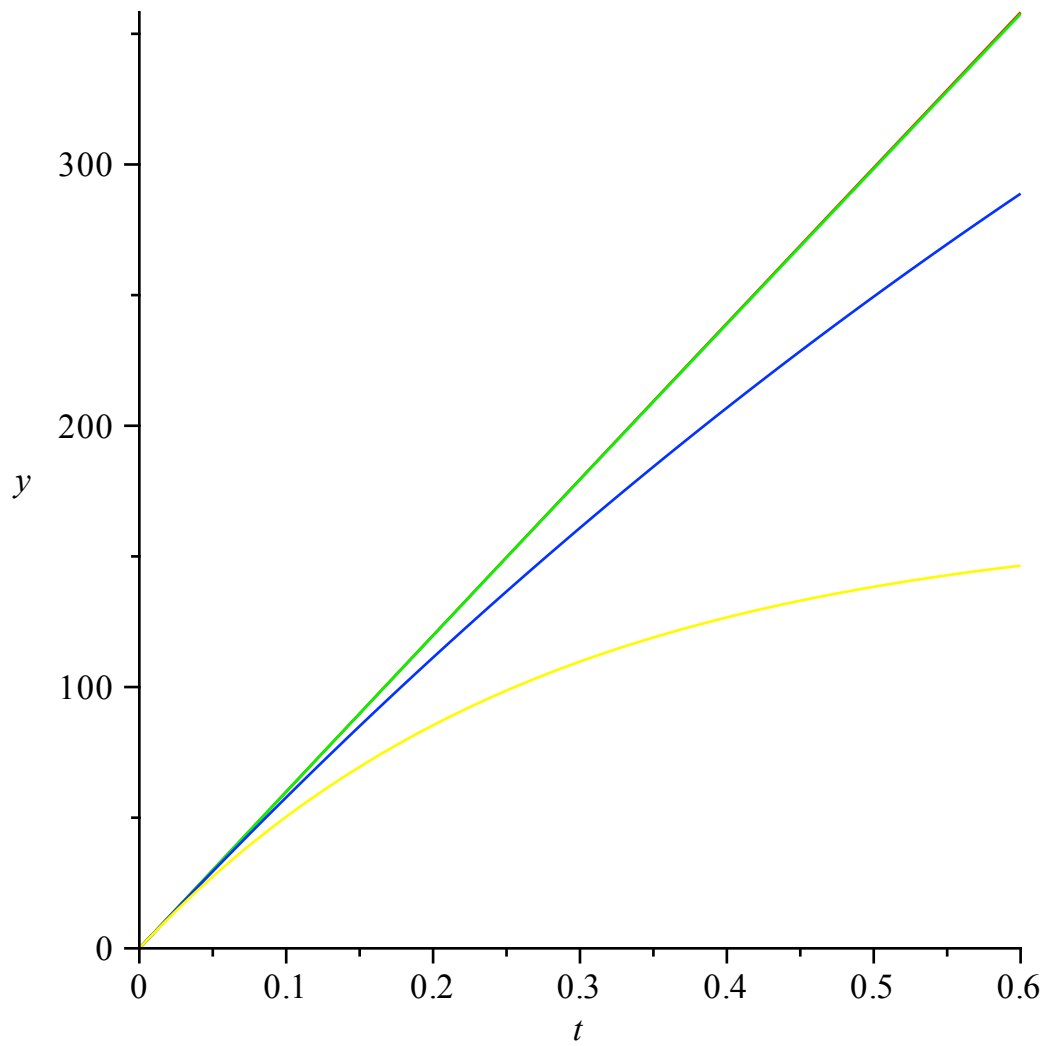


y a t=0.6sg

```
> plot([vel1,vel2,vel3,vel4],t=0..0.6,labels=[t,v],color=[red,  
green,blue,yellow]);
```

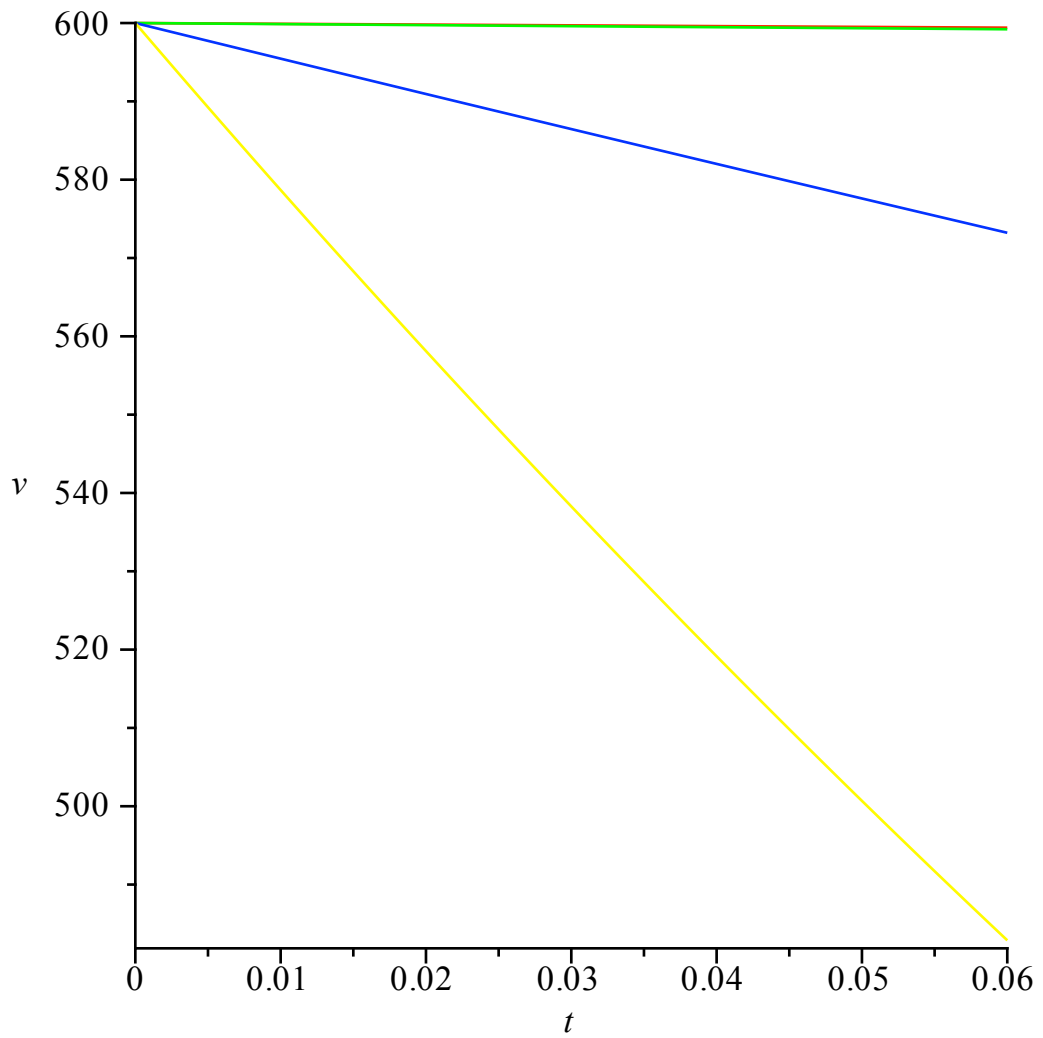


```
> plot([pos1, pos2, pos3, pos4], t=0..0.6, labels=[t, y], color=[red,  
green, blue, yellow]);
```

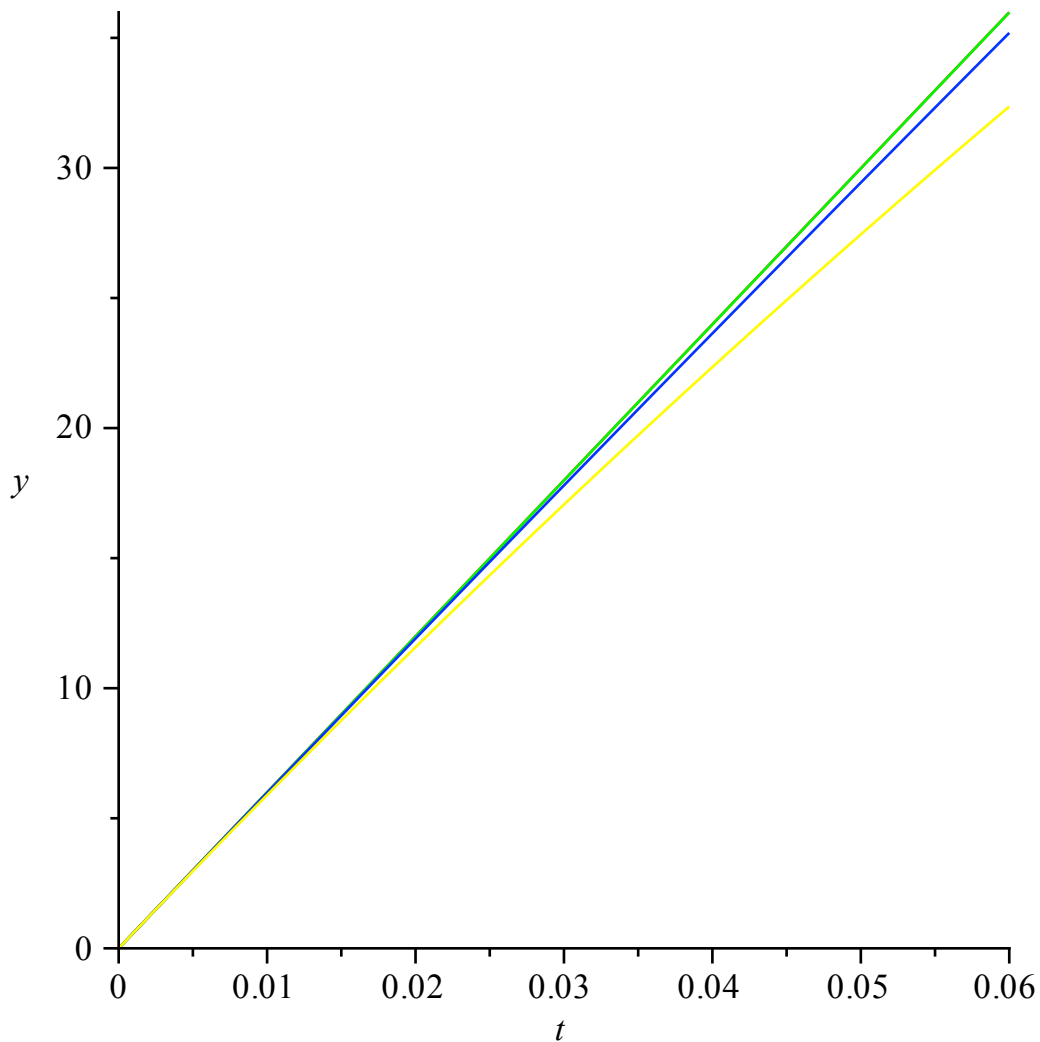


y a t=0.06sg

```
> plot([vel1,vel2,vel3,vel4],t=0..0.06,labels=[t,v],color=[red,  
green,blue,yellow]);
```



```
> plot([pos1, pos2, pos3, pos4], t=0..0.06, labels=[t, y], color=[red,  
green, blue, yellow]);
```

Consideremos ahora un corcho esférico que parte del reposo en el fondo de recipiente con agua, también tendrá una velocidad límite dada por

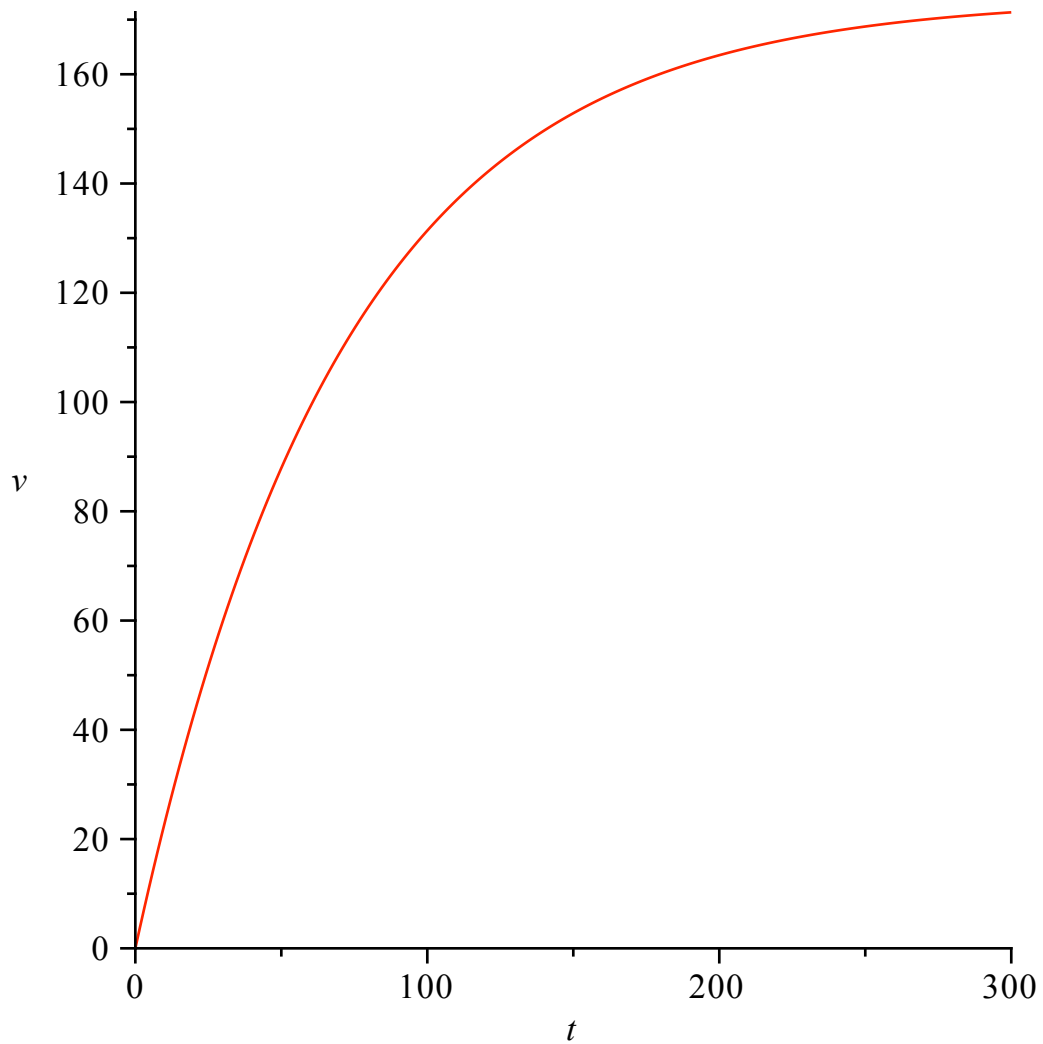
```
> corcho:= [g = -9.8, R=0.02, rho=10^3, xi=1, phi=0.8, v0=0, eta=
1.002*10^(-3)]:
vellim:=subs(corcho,vellimite);
```

vellim := 173.8744733

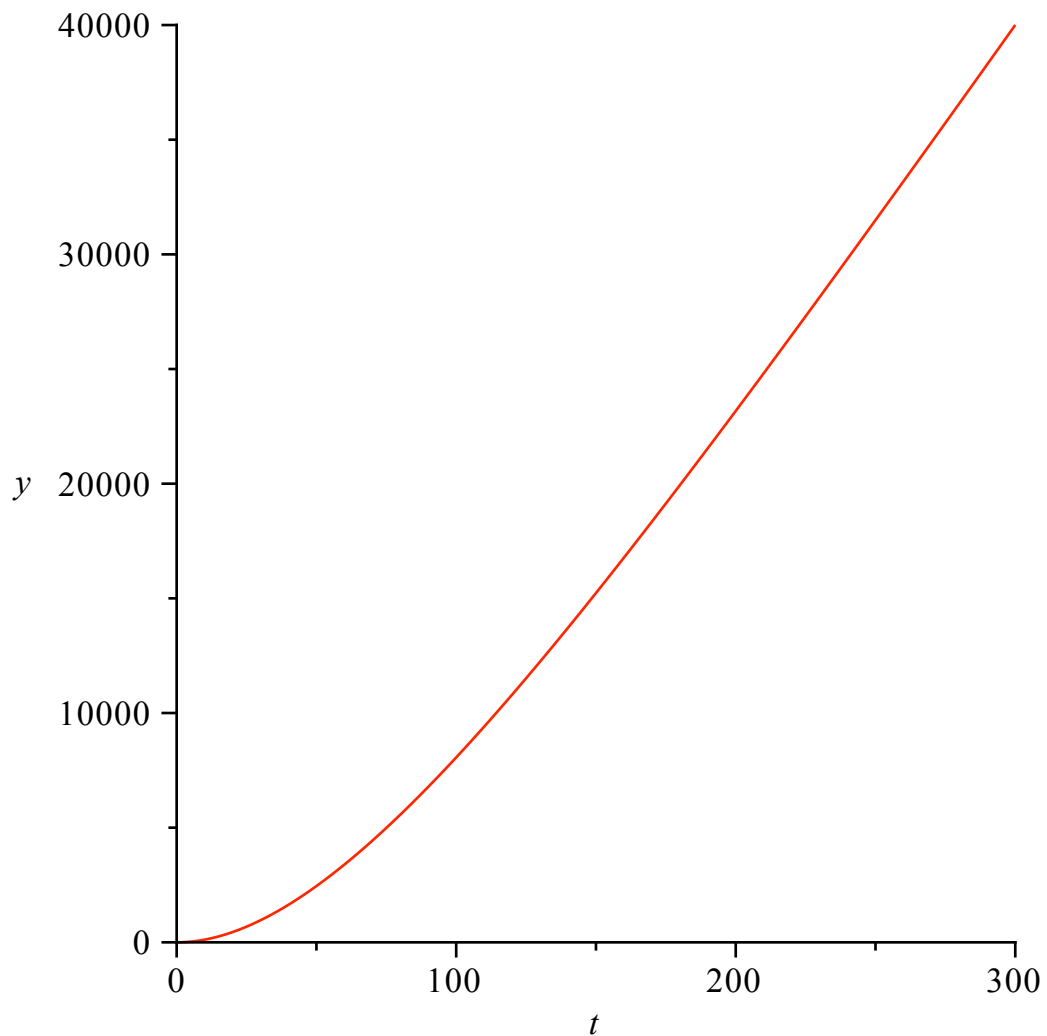
(1.12)

y podremos graficar su velocidad y su aceleración

```
> vell:=subs(corcho,velocidad);
plot(vell,t=0..300,labels=[t,v]);
vell := -173.8744732 e-0.01409062500 t + 173.8744733
```

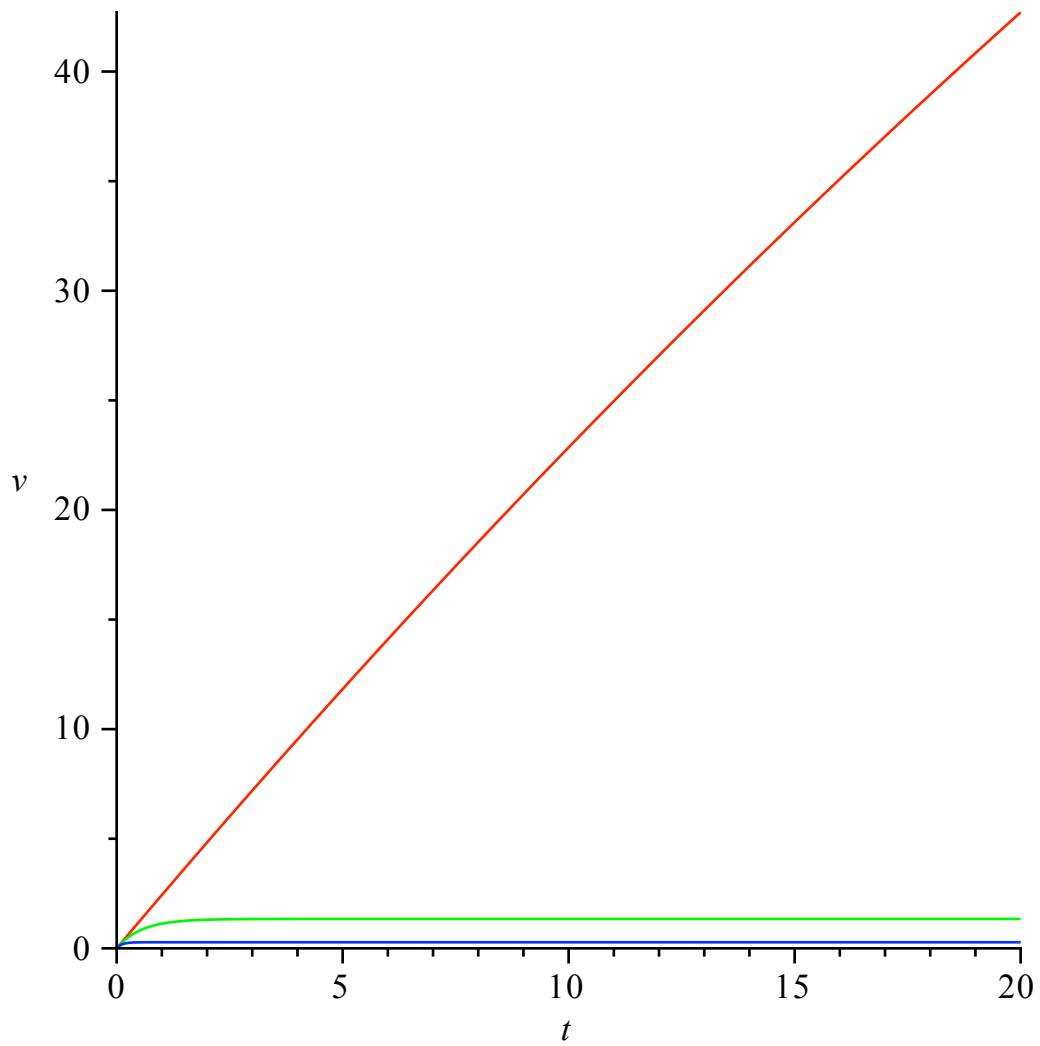


```
> pos1:=subs(corcho, posicion);  
plot(pos1,t=0..300,labels=[t,y]);  
pos1 := 12339.72753 e-0.01409062500 t + 173.8744733 t - 12339.72754
```

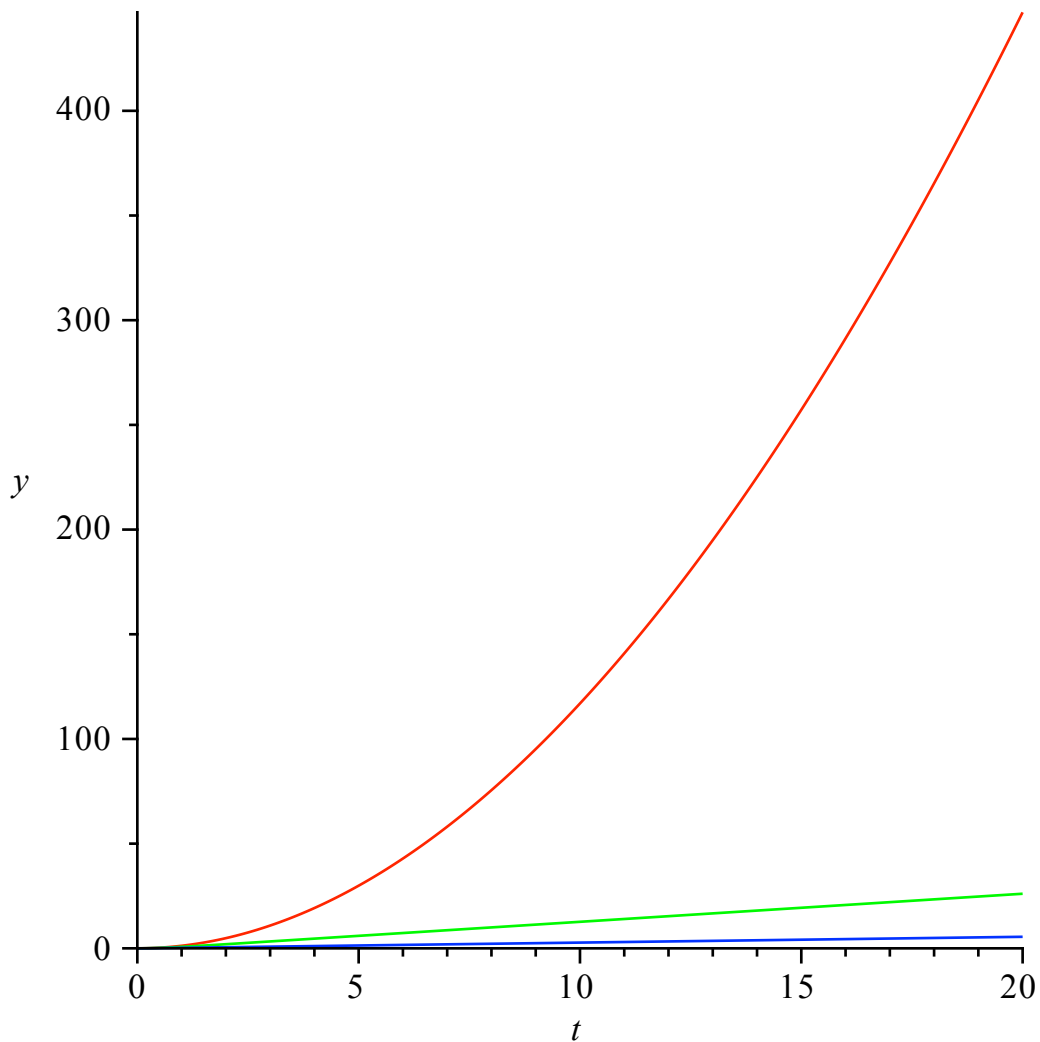


Al cambiar la viscosidad a aceite y reducir el tiempo

```
> corcho2:= [g = -9.8, R=0.02, rho=10^3, xi=1, phi=0.8, v0=0, eta=
130*10^(-3)]:
corcho3:= [g = -9.8, R=0.02, rho=10^3, xi=1, phi=0.8, v0=0, eta=
629*10^(-3)]:
> vel2:=subs(corcho2, velocidad):
vel3:=subs(corcho3, velocidad):
plot([vel1, vel2, vel3], t=0..20, labels=[t, v], color=[red, green,
blue]);
```



```
> pos2:=subs(corcho2, posicion):  
pos3:=subs(corcho3, posicion):  
plot([pos1, pos2, pos3], t=0..20, labels=[t, y], color=[red, green,  
blue]);
```



> $P \cdot V = N \cdot R \cdot T; P[0] \cdot V[0]; P[0]; V[0]; P = \rho \cdot g \cdot y(t);$
 $\rho \cdot g \cdot y(t) \cdot \left(\frac{4}{3}\right) \cdot \pi \cdot R(t)^3 = \rho \cdot g \cdot y(0) \cdot \left(\frac{4}{3}\right) \cdot \pi \cdot R(0)^3;$

$$P V = N R T$$

$$P_0 V_0$$

$$P_0$$

$$V_0$$

$$P = \rho g y(t)$$

$$\frac{4}{3} \rho g y(t) \pi R(t)^3 = \frac{4}{3} \rho g y(0) \pi R(0)^3$$

(1.13)

Finalmente supongamos una burbuja. Por lo tanto, el radio de la burbuja varía con la presión. Si suponemos que la burbuja está compuesta por un gas ideal tendremos que $P V = N R T = P_0 V_0 =$ constante, donde P_0 y V_0 son la presión y el volumen inicial respectivamente.

Ahora bien, si la presión en un líquido viene dada por $P = \rho g y(t)$, con $y(t)$ la profundidad, tendremos que

$$\frac{4 \rho g y(t) \pi R(t)^3}{3} = \frac{4 \rho g y(0) \pi R(0)^3}{3} = P_0 V_0 = \text{constante}$$

con lo cual

> **restart:R:=gamma0*y(t)^(-1/3);**

$$R := \frac{\gamma_0}{y(t)^{1/3}} \quad (1.14)$$

donde $\gamma_0 = R(0) * (y(0))^{1/3}$

nótese que la masa de la burbuja es constante, no así la masa desplazada por ella. Adicionalmente que hemos

cambiado el origen de coordenadas y ahora la posición inicial, $y(0) = y_0$ es distinta de cero.

> **gamma0 := R0*y0^(1/3);**
m0 := (4*Pi/3)*rho0*R^3;
mB := (4*Pi/3)*rho1*R0^3;
rho0 := xi*rho;
rho1 := phi*rho;
K := 6*Pi*R;

$$\gamma_0 := R_0 y_0^{1/3}$$

$$m_0 := \frac{4}{3} \frac{\pi \rho_0 R_0^3 y_0}{y(t)}$$

$$m_B := \frac{4}{3} \pi \rho_1 R_0^3$$

$$\rho_0 := \xi \rho$$

$$\rho_1 := \phi \rho$$

$$K := \frac{6 \pi R_0 y_0^{1/3}}{y(t)^{1/3}} \quad (1.15)$$

Donde ξ y ϕ representan las densidad relativas del fluido y del cuerpo respecto al agua (de densidad ρ), respectivamente. Por su parte

R_0 representa el radio inicial de la burbuja.

Para el caso una burbuja que se despega desde el fondo de un vaso de refresco la ecuación diferencial que describe el movimiento

puede ser escrita como

> **emov2 := mB*diff(y(t),t\$2) = -mB*g - K*eta*diff(y(t),t)+m0*g ;**

$$\text{emov2} := \frac{4}{3} \pi \phi \rho R_0^3 \left(\frac{d^2}{dt^2} y(t) \right) = -\frac{4}{3} \pi \phi \rho R_0^3 g - \frac{6 \pi R_0 y_0^{1/3} \eta \left(\frac{d}{dt} y(t) \right)}{y(t)^{1/3}} + \frac{4}{3} \frac{\pi \xi \rho R_0^3 y_0 g}{y(t)} \quad (1.16)$$

> **dsolve({emov2,y(0)=-y0,D(y)(0)=0},y(t));**

Por lo tanto debemos resolverla numéricamente. Para ello es imperioso adimensionalizarla. La

integración numérica se hace

sobre números, por lo que las variables deben ser adimensionales. Para ello cambiamos la variable independiente (el tiempo, t)

y la variable dependiente (la profundidad, y(t)) por

$$tt = t/tfinal \quad y \quad yy(tt) = y(t)/y(0)$$

```
> emov2AD := mB*diff(yy(tt),tt$2)(y0/(tfinal^2)) = -mB*g -6*Pi*
R0/yy(tt)^(1/3)*eta*diff(yy(tt),tt)*(y0/tfinal)+4/3*Pi*xi*rho*
R0^3/yy(tt)*g;
```

$$emov2AD := \frac{4}{3} \pi \phi \rho R0^3 \frac{d^2}{dt^2} yy(tt) \left(\frac{y0}{tfinal^2} \right) = -\frac{4}{3} \pi \phi \rho R0^3 g \quad (1.17)$$

$$- \frac{6 \pi R0 \eta \left(\frac{d}{dt} yy(tt) \right) y0}{yy(tt)^{1/3} tfinal} + \frac{4}{3} \frac{\pi \xi \rho R0^3 g}{yy(tt)}$$

acomodando un poco queda como

```
> emov2AD1:=diff(yy(tt),tt$2)=-(g/(y0/tfinal^2))
-(6*Pi*R0/yy(tt)^(1/3)*eta*diff(yy(tt),tt)*y0/tfinal)/(4/3*Pi*
phi*rho*R0^3*(y0/tfinal^2))
+(4/3*Pi*xi*rho*R0^3/yy(tt)*g)/(4/3*Pi*phi*rho*R0^3*
(y0/tfinal^2));
```

$$emov2AD1 := \frac{d^2}{dt^2} yy(tt) = -\frac{g tfinal^2}{y0} - \frac{9}{2} \frac{\eta \left(\frac{d}{dt} yy(tt) \right) tfinal}{R0^2 yy(tt)^{1/3} \phi \rho} + \frac{\xi g tfinal^2}{yy(tt) \phi y0} \quad (1.18)$$

y ahora convertimos esta ecuación de segundo orden en, dos ecuaciones de primer orden mediante un cambio de variable

$$\frac{d}{dt} yy(tt) = pp(tt) \quad \text{con lo cual} \quad \frac{d^2}{dt^2} yy(tt) = \frac{d}{dt} pp(tt)$$

y entonces el sistema queda como

```
> ecuac1a:=diff(yy(tt),tt)=pp(tt);
ecuac1b:=diff(pp(tt),tt) = -g/y0*tfinal^2 -9/2*1/R0^2/yy(tt)^(
1/3)*eta*pp(tt)*tfinal/phi/rho +xi/yy(tt)*g/phi/y0*tfinal^2;
```

$$ecuac1a := \frac{d}{dt} yy(tt) = pp(tt)$$

$$ecuac1b := \frac{d}{dt} pp(tt) = -\frac{g tfinal^2}{y0} - \frac{9}{2} \frac{\eta pp(tt) tfinal}{R0^2 yy(tt)^{1/3} \phi \rho} + \frac{\xi g tfinal^2}{yy(tt) \phi y0} \quad (1.19)$$

para una burbuja e agua esto es: $\eta = 0.001002000000$

```
> burbuja110:= [g = -9.8, R0=0.002,rho=10^3, xi=1, phi=0.08, eta=
1.002*10^(-3),y0 ==-10, tfinal=50]:
ecuac1a1:=subs(burbuja110,ecuac1a);
ecuac1b1:=subs(burbuja110,ecuac1b);
sist1:=ecuac1a1,ecuac1b1:
```

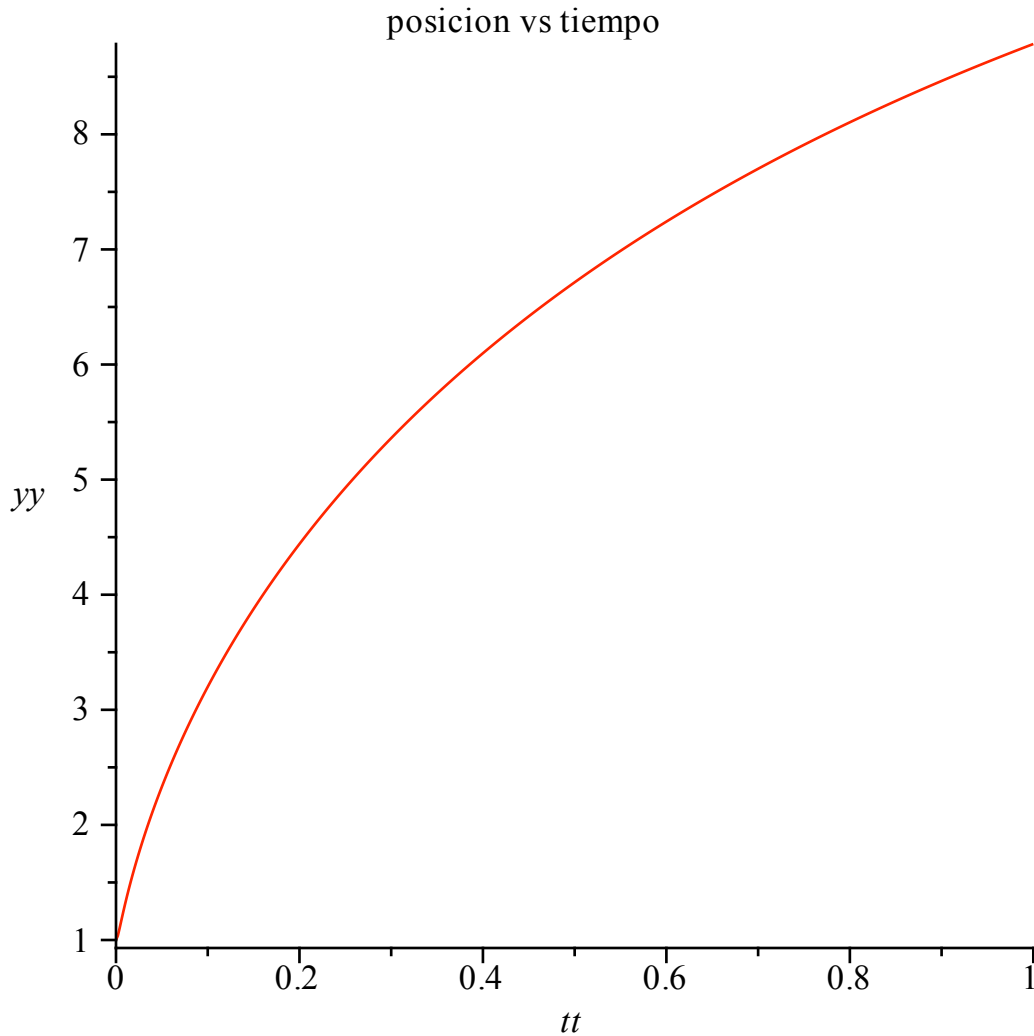
$$ecuac1a1 := \frac{d}{dt} yy(tt) = pp(tt)$$

(1.20)

$$ecuac1b1 := \frac{d}{dt} pp(tt) = -2450.000000 - \frac{704.5312500 pp(tt)}{yy(tt)^{1/3}} + \frac{30625.00000}{yy(tt)} \quad (1.20)$$

```
> with(plots):Digits:=20:
sol:=dsolve({sist1,pp(0)=0,yy(0)=1},{pp(tt),yy(tt)},numeric,
stiff=true,range=0..1);
odeplot(sol,[tt,yy(tt)],0..1,numpoints=500,title="posicion vs
tiempo");

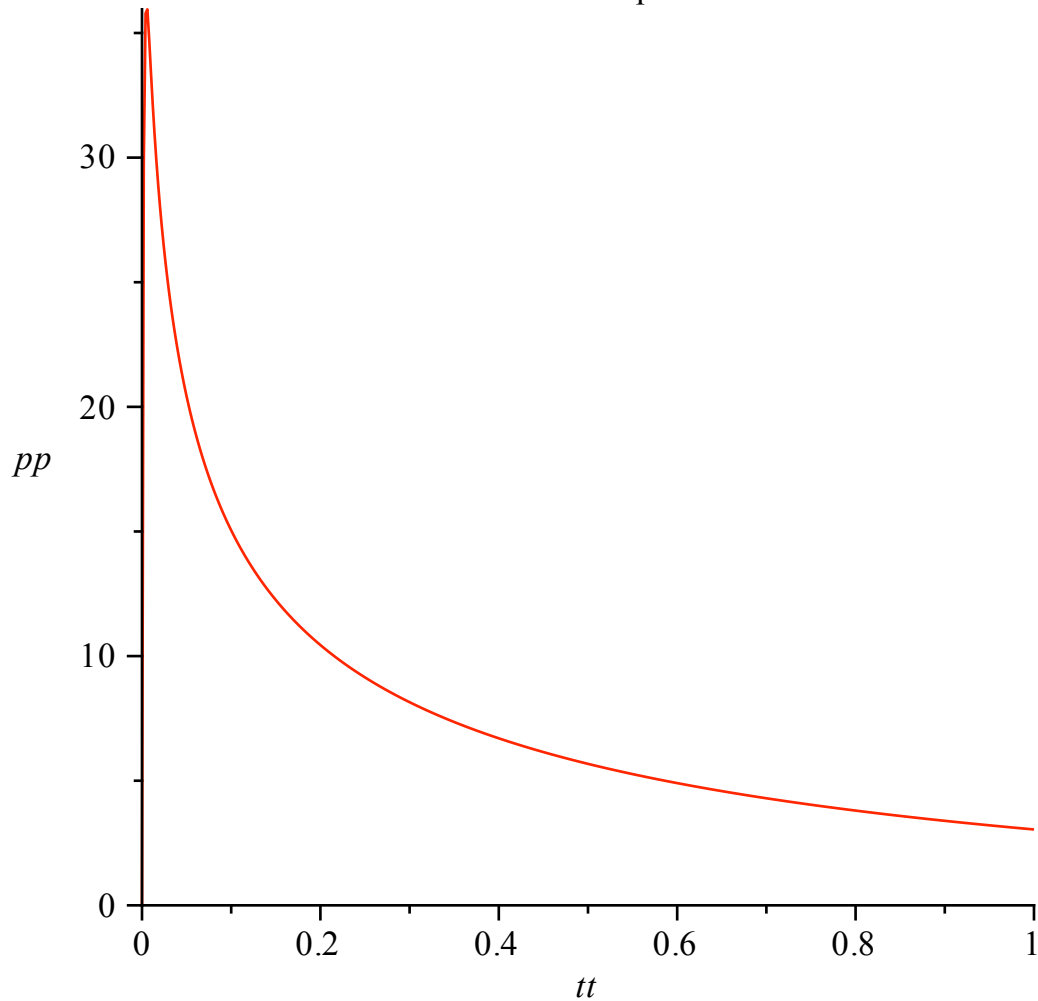
sol := proc(x_rosenbrock) ... end proc
```



y la gráfica de la velocidad será

```
> odeplot(sol,[tt,pp(tt)],0..1,numpoints=500,title="velocidad vs
tiempo");
```


velocidad vs tiempo



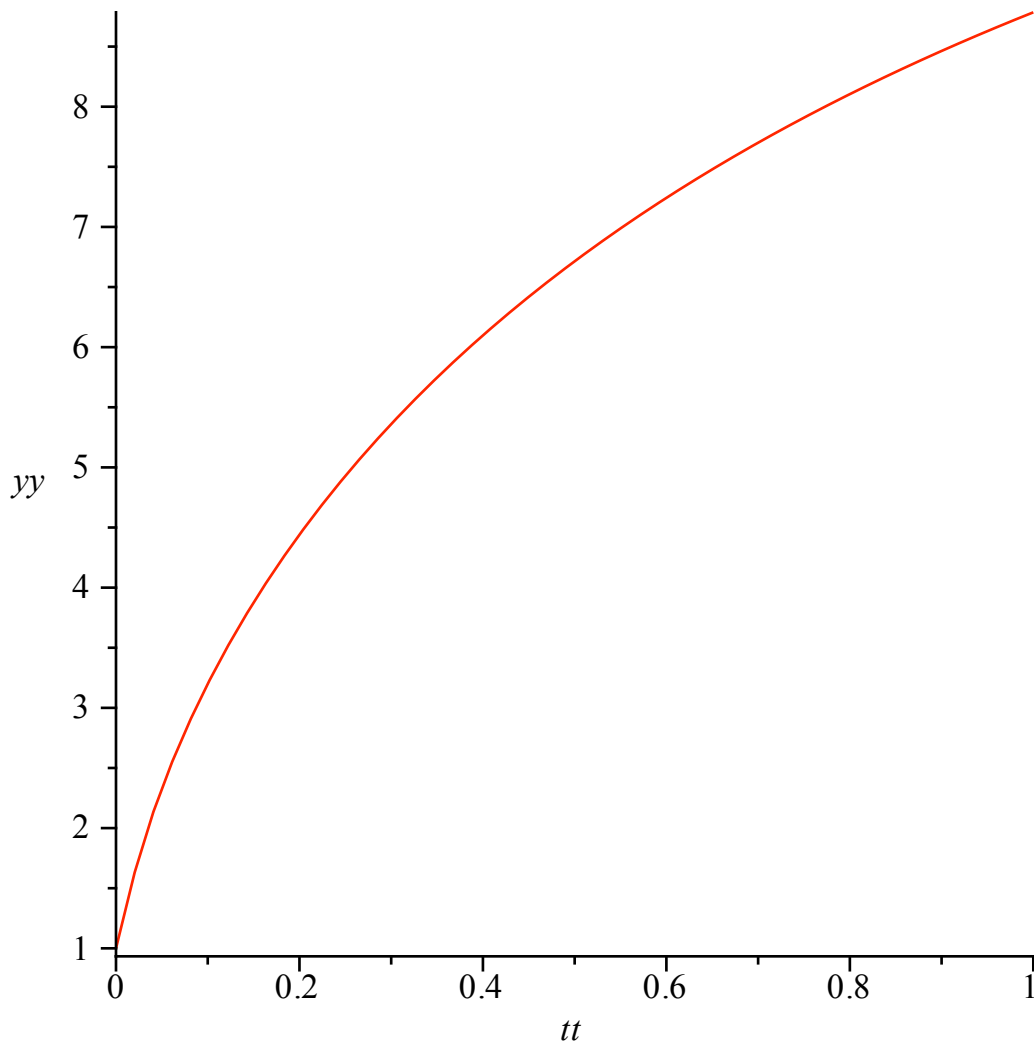
Equivalentemente, MAPLE puede resolver la ecuación de segundo orden directamente.

```
> emov2AD10:= subs(burbuja110,emov2AD1);
sol2:=dsolve({emov2AD10,yy(0)=1,D(yy)(0)=0},numeric,stiff=true)
;
odeplot(sol2,[tt,yy(tt)],0..1,numpoints=50);
```

$$emov2AD10 := \frac{d^2}{dt^2} yy(tt) = -2450.0000000000000000$$

$$- \frac{704.53125000000000000000 \left(\frac{d}{dt} yy(tt) \right)}{yy(tt)^{1/3}} + \frac{30625.0000000000000000}{yy(tt)}$$

```
sol2 := proc(x_rosenbrock) ... end proc
```



▼ Movimiento Parabólico: Trayectoria de Balas en Fluidos

`> restart;with(plots):`

Toca ahora simular el lanzamiento de proyectiles en Fluidos. En física general, aún en secundaria, es proverbial resolver el movimiento en dos dimensiones (2D) con atracción gravitatoria. En esas tempranas épocas de nuestro conocimiento de física a este "problema" se le conoce con el nombre de movimiento parabólico, por la trayectoria que describe el objeto.

Si suponemos un cuerpo de masa constante, m , que se mueve en un fluido con resistencia de despreciable las ecuaciones diferenciales que

describen el movimiento no pueden ser otras que aquellas que provengan de las ecuaciones de Newton y en este caso corresponden a

$$\sum_{i=1}^n F_{ext} = m \left(\frac{d^2}{dt^2} r(t) \right)$$

donde $r(t)$ es el radio vector posición.

`> ecuacxlib:=m*diff(xlib(t),t,t)=0;`
`ecuaclylib:=m*diff(ylib(t),t,t)=-m*g;`

$$ecuac1xlib := m \left(\frac{d^2}{dt^2} xlib(t) \right) = 0$$

$$ecuac1ylib := m \left(\frac{d^2}{dt^2} ylib(t) \right) = -m g \quad (2.1)$$

las cuales se resuelven de forma inmediata

```
> solxlib:=dsolve({ecuac1xlib,xlib(0)=x0,D(xlib)(0)=V0x});
solylib:=dsolve({ecuac1ylib,ylib(0)=y0,D(ylib)(0)=V0y});
assign(solxlib):assign(solylib):
```

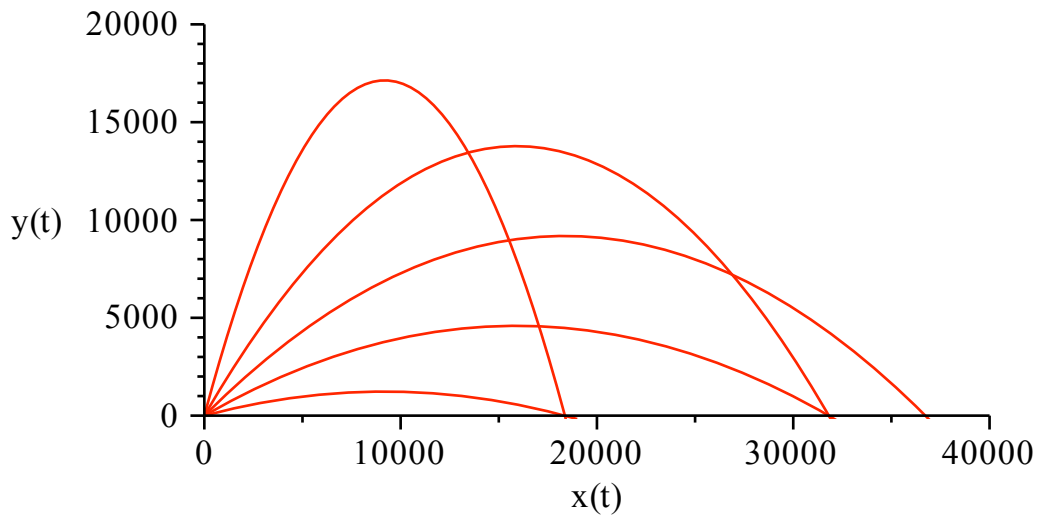
$$solxlib := xlib(t) = V0x t + x0$$

$$solylib := ylib(t) = -\frac{1}{2} g t^2 + V0y t + y0 \quad (2.2)$$

que no son otra cosa que las formulitas que siempre nos aprendimos

```
> otrosparamlib:=[g=9.8]:
pos0:=[x0=0,y0=0]:
vel0pi12:=[V0x=600*evalf(cos(Pi/12)),V0y=600*evalf(sin(Pi/12))]
:
vel0pi6:=[V0x=600*evalf(cos(Pi/6)),V0y=600*evalf(sin(Pi/6))]:
vel0pi4:=[V0x=600*evalf(cos(Pi/4)),V0y=600*evalf(sin(Pi/4))]:
vel0pi3:=[V0x=600*evalf(cos(Pi/3)),V0y=600*evalf(sin(Pi/3))]:
vel05pi12:=[V0x=600*evalf(cos(5*Pi/12)),V0y=600*evalf(sin(5*
Pi/12))]:
> xpi12(t):=subs(pos0,vel0pi12,otrosparamlib,xlib(t)):
ypi12(t):=subs(pos0,vel0pi12,otrosparamlib,ylib(t)):
grafpi12:=plot([xpi12(t),ypi12(t),t=0..500],scaling=
constrained):
xpi6(t):=subs(pos0,vel0pi6,otrosparamlib,xlib(t)):
ypi6(t):=subs(pos0,vel0pi6,otrosparamlib,ylib(t)):
grafpi6:=plot([xpi6(t),ypi6(t),t=0..500],scaling=constrained):
xpi4(t):=subs(pos0,vel0pi4,otrosparamlib,xlib(t)):
ypi4(t):=subs(pos0,vel0pi4,otrosparamlib,ylib(t)):
grafpi4:=plot([xpi4(t),ypi4(t),t=0..500],scaling=constrained):
xpi3(t):=subs(pos0,vel0pi3,otrosparamlib,xlib(t)):
ypi3(t):=subs(pos0,vel0pi3,otrosparamlib,ylib(t)):
grafpi3:=plot([xpi3(t),ypi3(t),t=0..500],scaling=constrained):
x5pi12(t):=subs(pos0,vel05pi12,otrosparamlib,xlib(t)):
y5pi12(t):=subs(pos0,vel05pi12,otrosparamlib,ylib(t)):
graf5pi12:=plot([x5pi12(t),y5pi12(t),t=0..500],scaling=
constrained):
display({grafpi12,grafpi6,grafpi4,grafpi3,graf5pi12},view=[0.
.40000,0..20000],
title="Movimiento
Parabólico, sin roce",labels=["x(t)","y(t)"]);
```

Movimiento Parabólico, sin roce



arriba se ilustra las trayectorias correspondientes a distintos movimientos parabólicos con una velocidad inicial $V_0 = 25$ y distintos ángulos de

$$\text{disparo } \theta = \frac{1 \pi / 12 \quad \pi / 6 \quad \pi / 4 \quad \pi / 3 \quad \pi / 2}{12 \cdot 6 \cdot 4 \cdot 3 \cdot 2}$$

Una cosa distinta se presenta para el movimiento en 2D con resistencia del aire. En este caso, como en todos los casos en los cuales una

fuerza de origen microscópico es descrita macroscópicamente, su representación matemática es fenomenológica. Vale decir, mediante una

fórmula matemática, utilizamos una descripción promedio de su efecto. En el caso de la fuerza de roce en fluidos se presentan dos casos:

uno en el cual la fuerza de resistencia con el aire es proporcional a la velocidad,

$f_{roce} = -K \eta \sqrt{V_x^2 + V_y^2} U_v$ y otro en el cual supondremos la fuerza de roce proporcional al cuadrado de su velocidad, $f_{roce} = -K \eta (V_x^2 + V_y^2) U_v$. En estos casos U_v es el vector unitario en la dirección

tangente $U_v = \frac{V_x i}{(V_x^2 + V_y^2)^{\frac{1}{2}}} + \frac{V_y j}{(V_x^2 + V_y^2)^{\frac{1}{2}}}$ y η es el coeficiente de fricción que depende de la

viscosidad del fluido. En el sistema MKS se expresa en N s/g/m^2 y se mide en Poise que es igual a un décimo de la unidad en MKS. Esto es $10\text{P}=1\text{N s/g/m}^2$. En ambos casos K es el coeficiente de fricción que depende de la forma de cuerpo. Para una esfera de Radio, R , tendremos $K=6\pi R$. La utilización de una u otra expresión para la fuerza de roce se basa en la magnitud de la velocidad del cuerpo respecto al fluido. Para los casos en los cuales la velocidad del cuerpo es mucho menor a la velocidad del sonido, la fuerza de roce será proporcional a la velocidad. En aquellos casos en los cuales el orden de magnitud del módulo de la velocidad del cuerpo sea el de la velocidad del sonido en ese medio, utilizaremos que la fuerza de roce será proporcional al cuadrado de la velocidad.

Con lo cual para el caso de desplazamientos lentos a través de fluidos tendremos

```
> ecuacxrocelent:=m*diff(xrocelent(t),t,t)=
      -K*eta*sqrt(diff(xrocelent(t),t)^2 +
diff(yrocelent(t),t)^2 )*(diff(xrocelent(t),t)/
      (sqrt
      (diff(xrocelent(t),t)^2 +diff(yrocelent(t),t)^2 )));
ecuacyrocelent:=m*diff(yrocelent(t),t,t)=-m*g
      -K*eta*sqrt(diff(xrocelent(t),t)^2 +
diff(yrocelent(t),t)^2 )*(diff(yrocelent(t),t)/
      (sqrt
      (diff(xrocelent(t),t)^2 +diff(yrocelent(t),t)^2 )));
```

$$ecuacxrocelent := m \left(\frac{d^2}{dt^2} xrocelent(t) \right) = -K \eta \left(\frac{d}{dt} xrocelent(t) \right)$$

$$ecuacyrocelent := m \left(\frac{d^2}{dt^2} yrocelent(t) \right) = -m g - K \eta \left(\frac{d}{dt} yrocelent(t) \right) \quad (2.3)$$

Las cuales pueden ser integradas analíticamente como

```
> K:=6*Pi*R;
solxrocelent:=dsolve({ecuacxrocelent,xrocelent(0)=x0,D
(xrocelent)(0)=V0x});
solyrocelent:=dsolve({ecuacyrocelent,yrocelent(0)=y0,D
(yrocelent)(0)=V0y});
assign(solxrocelent);
assign(solyrocelent);
```

$$K := 6 \pi R$$

$$solxrocelent := xrocelent(t) = \frac{1}{6} \frac{6 x0 \pi R \eta + V0x m}{\pi R \eta} - \frac{1}{6} \frac{V0x m e^{-\frac{6 \pi R \eta t}{m}}}{\pi R \eta}$$

$$solyrocelent := yrocelent(t) = -\frac{1}{36} \frac{m e^{-\frac{6 \pi R \eta t}{m}} (6 V0y \pi R \eta + m g)}{\pi^2 R^2 \eta^2} - \frac{1}{6} \frac{m g t}{\pi R \eta} \quad (2.4)$$

$$+ \frac{1}{36} \frac{36 y0 \pi^2 R^2 \eta^2 + 6 m V0y \pi R \eta + m^2 g}{\pi^2 R^2 \eta^2}$$

Ahora sustituimos los valores para distintas fuerzas de roce: aire, agua y aceite

```
> otrosparamroceaire:=[g=9.8,m=0.015,eta=0.01894*10^(-3),R=0.015]
:
xpi12lent(t):=subs(pos0,vel0pi12,otrosparamroceaire,xrocelent
```

```

(t)):
ypil2lent(t):=subs(pos0,vel0pil2,otrosparamroceaire,yrocelent
(t)):
grafpil2lentaire:=plot([xpil2lent(t),ypil2lent(t),t=0..500],
scaling=constrained):
xpi6lent(t):=subs(pos0,vel0pi6,otrosparamroceaire,xrocelent(t))
:
ypi6lent(t):=subs(pos0,vel0pi6,otrosparamroceaire,yrocelent(t))
:
grafpi6lentaire:=plot([xpi6lent(t),ypi6lent(t),t=0..500],
scaling=constrained):
xpi4lent(t):=subs(pos0,vel0pi4,otrosparamroceaire,xrocelent(t))
:
ypi4lent(t):=subs(pos0,vel0pi4,otrosparamroceaire,yrocelent(t))
:
grafpi4lentaire:=plot([xpi4lent(t),ypi4lent(t),t=0..500],
scaling=constrained):
xpi3lent(t):=subs(pos0,vel0pi3,otrosparamroceaire,xrocelent(t))
:
ypi3lent(t):=subs(pos0,vel0pi3,otrosparamroceaire,yrocelent(t))
:
grafpi3lentaire:=plot([xpi3lent(t),ypi3lent(t),t=0..500],
scaling=constrained):
x5pil2lent(t):=subs(pos0,vel05pi12,otrosparamroceaire,xrocelent
(t)):
y5pil2lent(t):=subs(pos0,vel05pi12,otrosparamroceaire,yrocelent
(t)):
graf5pil2lentaire:=plot([x5pil2lent(t),y5pil2lent(t),t=0..500],
scaling=constrained):
> otrosparamroceagua:=[g=9.8,m=0.015,eta=1.002*10^(-3),R=0.015]:
xpil2lent(t):=subs(pos0,vel0pil2,otrosparamroceagua,xrocelent
(t)):
ypil2lent(t):=subs(pos0,vel0pil2,otrosparamroceagua,yrocelent
(t)):
grafpil2lentagua:=plot([xpil2lent(t),ypil2lent(t),t=0..500],
scaling=constrained):
xpi6lent(t):=subs(pos0,vel0pi6,otrosparamroceagua,xrocelent(t))
:
ypi6lent(t):=subs(pos0,vel0pi6,otrosparamroceagua,yrocelent(t))
:
grafpi6lentagua:=plot([xpi6lent(t),ypi6lent(t),t=0..500],
scaling=constrained):
xpi4lent(t):=subs(pos0,vel0pi4,otrosparamroceagua,xrocelent(t))
:
ypi4lent(t):=subs(pos0,vel0pi4,otrosparamroceagua,yrocelent(t))
:
grafpi4lentagua:=plot([xpi4lent(t),ypi4lent(t),t=0..500],
scaling=constrained):
xpi3lent(t):=subs(pos0,vel0pi3,otrosparamroceagua,xrocelent(t))
:
ypi3lent(t):=subs(pos0,vel0pi3,otrosparamroceagua,yrocelent(t))
:
grafpi3lentagua:=plot([xpi3lent(t),ypi3lent(t),t=0..500],
scaling=constrained):
x5pil2lent(t):=subs(pos0,vel05pi12,otrosparamroceagua,xrocelent
(t)):
y5pil2lent(t):=subs(pos0,vel05pi12,otrosparamroceagua,yrocelent
(t)):
graf5pil2lentagua:=plot([x5pil2lent(t),y5pil2lent(t),t=0..500],
scaling=constrained):
> otrosparamroceaceite:=[g=9.8,m=0.015,eta=130*10^(-3),R=0.015]:

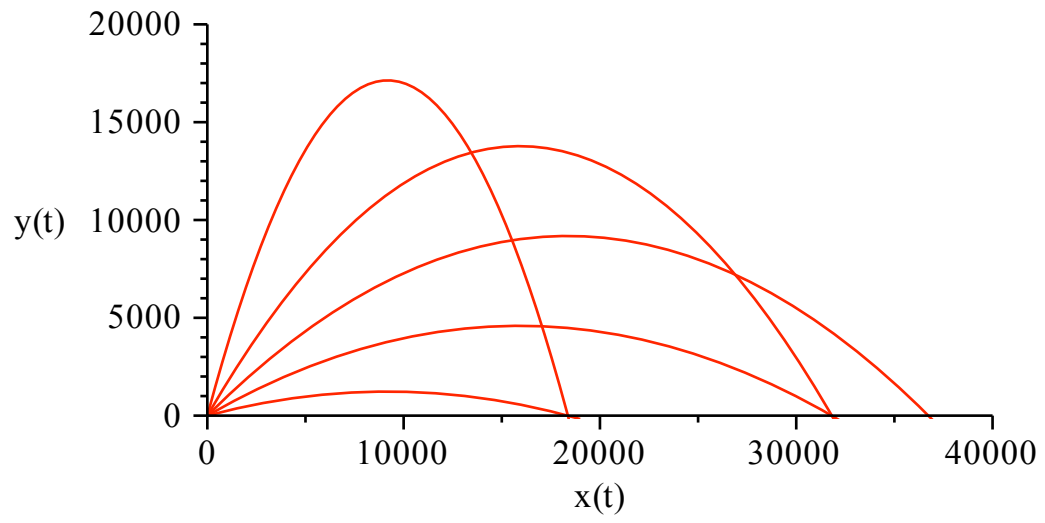
```

```

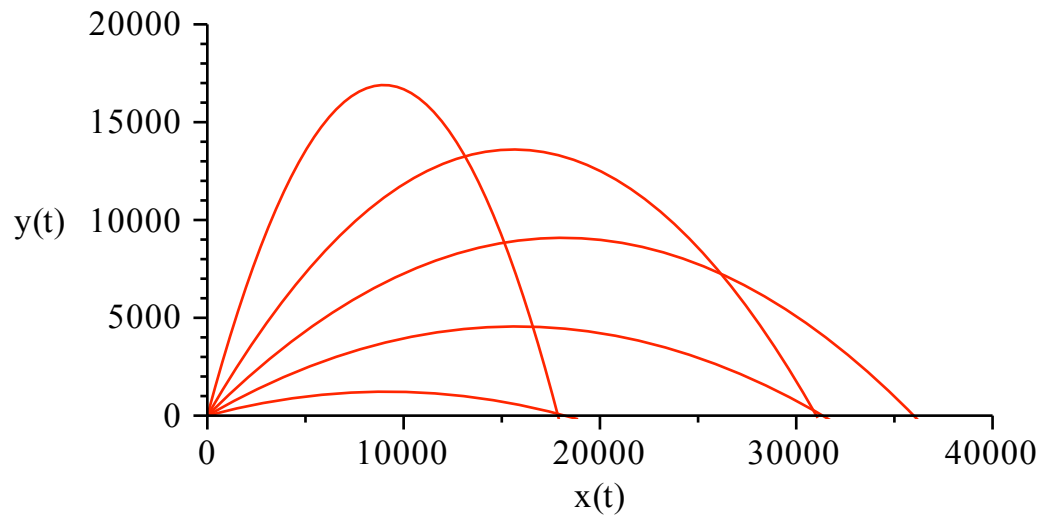
xpi12lent(t):=subs(pos0,vel0pi12,otrosparamroceaceite,xrocelent
(t)):
ypi12lent(t):=subs(pos0,vel0pi12,otrosparamroceaceite,yrocelent
(t)):
grafpi12lentaceite:=plot([xpi12lent(t),ypi12lent(t),t=0..500],
scaling=constrained):
xpi6lent(t):=subs(pos0,vel0pi6,otrosparamroceaceite,xrocelent
(t)):
ypi6lent(t):=subs(pos0,vel0pi6,otrosparamroceaceite,yrocelent
(t)):
grafpi6lentaceite:=plot([xpi6lent(t),ypi6lent(t),t=0..500],
scaling=constrained):
xpi4lent(t):=subs(pos0,vel0pi4,otrosparamroceaceite,xrocelent
(t)):
ypi4lent(t):=subs(pos0,vel0pi4,otrosparamroceaceite,yrocelent
(t)):
grafpi4lentaceite:=plot([xpi4lent(t),ypi4lent(t),t=0..500],
scaling=constrained):
xpi3lent(t):=subs(pos0,vel0pi3,otrosparamroceaceite,xrocelent
(t)):
ypi3lent(t):=subs(pos0,vel0pi3,otrosparamroceaceite,yrocelent
(t)):
grafpi3lentaceite:=plot([xpi3lent(t),ypi3lent(t),t=0..500],
scaling=constrained):
x5pi12lent(t):=subs(pos0,vel05pi12,otrosparamroceaceite,
xrocelent(t)):
y5pi12lent(t):=subs(pos0,vel05pi12,otrosparamroceaceite,
yrocelent(t)):
graf5pi12lentaceite:=plot([x5pi12lent(t),y5pi12lent(t),t=0.
.500],scaling=constrained):
> display({grafpi12,grafpi6,grafpi4,grafpi3,graf5pi12},view=[0.
.40000,0..20000],
title="Movimiento
Parabólico, sin roce",labels=["x(t)","y(t)"]);
display({grafpi12lentaire,grafpi6lentaire,grafpi4lentaire,
grafpi3lentaire,graf5pi12lentaire },view=[0..40000,0..20000],
title="Movimiento Parabólico
lento, con roce en aire",labels=["x(t)","y(t)"]);
display({grafpi12lentagua,grafpi6lentagua,grafpi4lentagua,
grafpi3lentagua,graf5pi12lentagua},view=[0..20000,0..12000],
title="Movimiento Parabólico
lento, con roce en agua",labels=["x(t)","y(t)"]);
display({grafpi12lentaceite,grafpi6lentaceite,
grafpi4lentaceite,grafpi3lentaceite,graf5pi12lentaceite},view=
[0..500,0..250],
title="Movimiento Parabólico
lento, con roce en aceite",labels=["x(t)","y(t)"]);

```

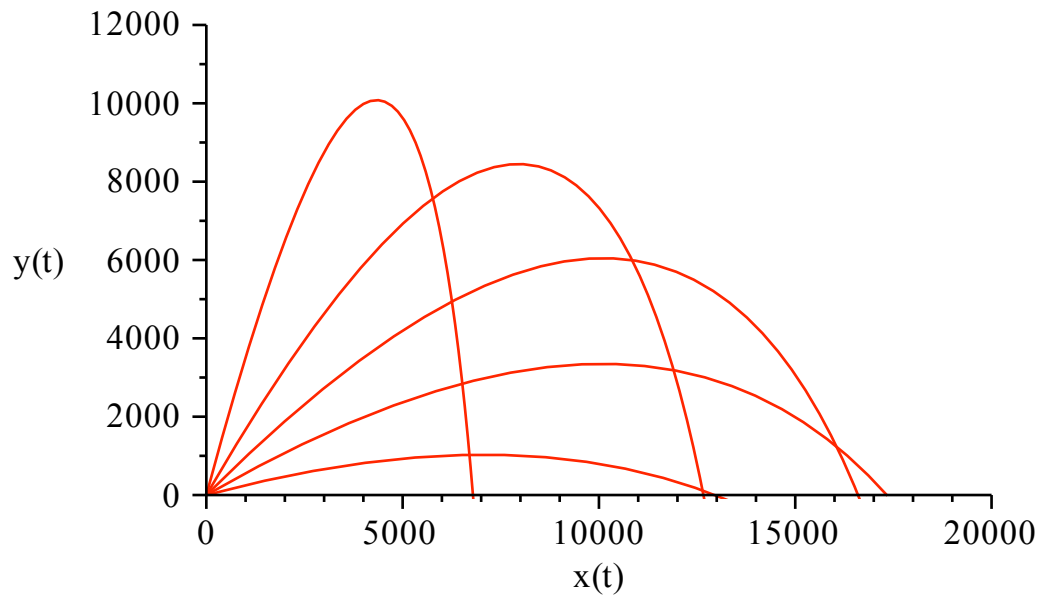
Movimiento Parabólico, sin roce



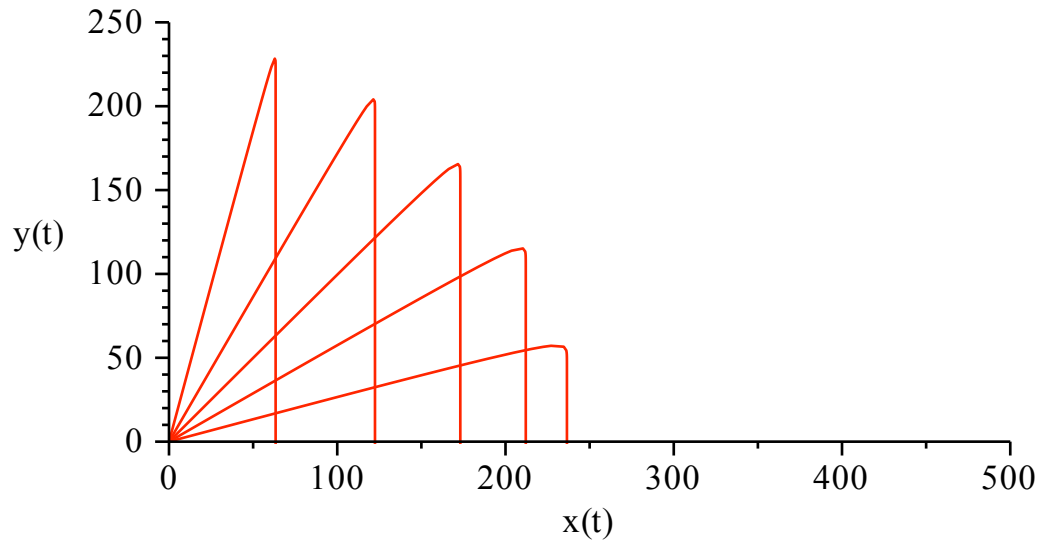
Movimiento Parabólico lento, con roce en aire



Movimiento Parabólico lento, con roce en agua



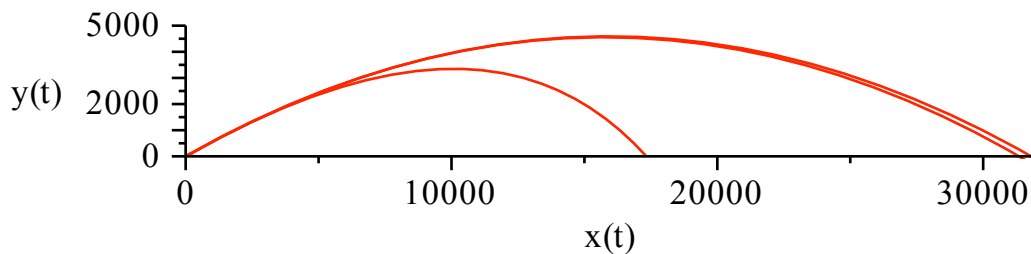
Movimiento Parabólico lento, con roce en aceite



También podemos graficar disparos con un mismo ángulo y distintos medios dispersivos

```
> display({grafpi6,grafpi6lentaire,grafpi6lentagua,  
grafpi6lentaceite},view=[0..32000,0..5000],  
title="Movimiento  
Parabólico, para un ángulo  $\text{Pi}/6$ ",labels=["x(t)", "y(t)"]);
```

Movimiento Parabólico, para un ángulo $\pi/6$



Supongamos que ahora la fuerza de roce es proporcional al cuadrado de la velocidad. Esto implica que las ecuaciones quedan como

$$\begin{aligned}
 > \text{ecuacxrocerap} := m \cdot \text{diff}(xrocerap(t), t, t) = \\
 & \quad -K \cdot \eta \cdot (\text{diff}(xrocerap(t), t)^2 + \text{diff}(yrocerap(t), t)^2) \cdot (\text{diff}(xrocerap(t), t) / \\
 & \quad (\text{diff}(xrocerap(t), t)^2 + \text{diff}(yrocerap(t), t)^2)); \quad (\text{sqrt} \\
 \text{ecuacyrocerap} := m \cdot \text{diff}(yrocerap(t), t, t) &= -m \cdot g \\
 & \quad -K \cdot \eta \cdot (\text{diff}(xrocerap(t), t)^2 + \text{diff}(yrocerap(t), t)^2) \cdot (\text{diff}(yrocerap(t), t) / \\
 & \quad (\text{diff}(xrocerap(t), t)^2 + \text{diff}(yrocerap(t), t)^2)); \quad (\text{sqrt}
 \end{aligned}$$

$$\text{ecuacxrocerap} := m \left(\frac{d^2}{dt^2} xrocerap(t) \right) =$$

$$-6 \pi R \eta \sqrt{\left(\frac{d}{dt} xrocerap(t) \right)^2 + \left(\frac{d}{dt} yrocerap(t) \right)^2} \left(\frac{d}{dt} xrocerap(t) \right)$$

$$\text{ecuacyrocerap} := m \left(\frac{d^2}{dt^2} yrocerap(t) \right) = -m g$$

(2.5)

$$-6\pi R\eta \sqrt{\left(\frac{d}{dt} x_{rocerap}(t)\right)^2 + \left(\frac{d}{dt} y_{rocerap}(t)\right)^2} \left(\frac{d}{dt} y_{rocerap}(t)\right)$$

que ahora están mezcladas y el sistema de ecuaciones diferenciales NO se separa.

El Péndulo Físico

Oscilador Armónico Simple

Alguien, de los famosos de Física, dijo una vez: quien entienda el oscilador armónico simple en todas sus facetas, entiende la mitad de toda la Física. Por ello vamos a tratar de estudiarlo.

Llamaremos Oscilador armónico simple o pendulo fisico linealizado aquel sistema fisico que tiene oscilaciones pequeñas y por lo tanto se puede linealizar la ecuación diferencial

`> restart;`

Oscilador Armónico Simple Libre y no amortiguado)

El oscilador armónico simple, libre tiene como ecuación diferencial

`> ecuacosarmsimple:=diff(xsimp(t),t$2)=-(\omega0^2)*xsimp(t);`

$$ecuacosarmsimple := \frac{d^2}{dt^2} xsimp(t) = -\omega_0^2 xsimp(t) \quad (3.1.1.1)$$

que tendrá como solución

`> solxsimp:=dsolve({ecuacosarmsimple,xsimp(0)=x0,D(xsimp)(0)=v0},xsimp(t));`
`assign(solxsimp);vsimp(t):=diff(xsimp(t),t);`

$$solxsimp := xsimp(t) = \frac{v_0 \sin(\omega_0 t)}{\omega_0} + x_0 \cos(\omega_0 t)$$

$$vsimp(t) := v_0 \cos(\omega_0 t) - x_0 \sin(\omega_0 t) \omega_0 \quad (3.1.1.2)$$

Supongamos que $\omega_0 = 2$ y que el móvil parte de $x_0 = 0$ con distintas velocidades iniciales. Vale decir

$$v(0) = 3, 5, 2\sqrt{10}, 7, 8.$$

`> omega0:='omega0';x0:='x0'; v0:='v0';`
`Xsimp:=unapply(xsimp(t),omega0,x0,v0,t);`
`Vsimp:=unapply(vsimp(t),omega0,x0,v0,t);`

$$\omega_0 := \omega_0$$

$$x_0 := x_0$$

$$v_0 := v_0$$

$$Xsimp := (\omega_0, x_0, v_0, t) \rightarrow \frac{v_0 \sin(\omega_0 t)}{\omega_0} + x_0 \cos(\omega_0 t)$$

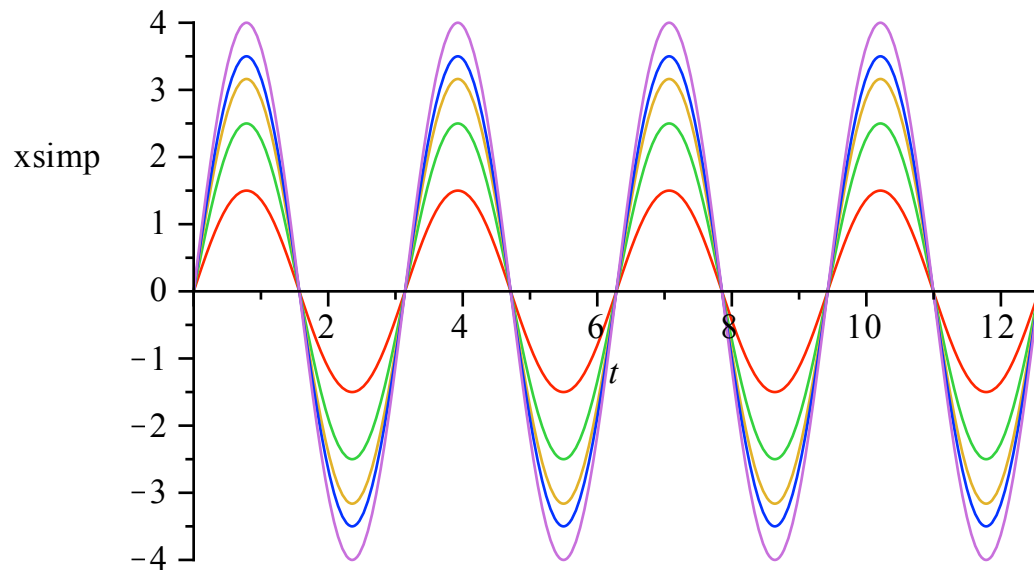
$$Vsimp := (\omega_0, x_0, v_0, t) \rightarrow v_0 \cos(\omega_0 t) - x_0 \sin(\omega_0 t) \omega_0 \quad (3.1.1.3)$$

con lo cual obtendremos la siguiente gráfica posición tiempo

`> plot([Xsimp(2,0,3,t),Xsimp(2,0,5,t),Xsimp(2,0,2*sqrt(10),t),Xsimp(2,0,7,t),Xsimp(2,0,8,t)],t=0..4*Pi,labels=[t,"xsimp"], title="Oscilador Armonico Simple, Evolucion`

```
temporal", scaling=constrained);
```

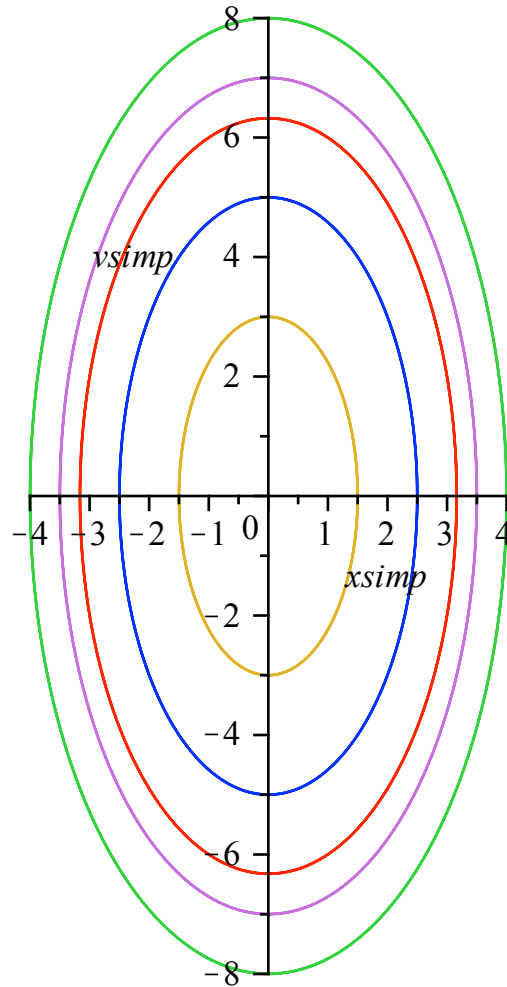
Oscilador Armonico Simple, Evolucion temporal



y la representación en el espacio de fases será

```
> plot([Xsimp(2,0,3,t),Vsimp(2,0,3,t),t=0..2*Pi],  
      [Xsimp(2,0,5,t),Vsimp(2,0,5,t),t=0..2*Pi],  
      [Xsimp(2,0,2*sqrt(10),t),Vsimp(2,0,2*sqrt(10),t),  
      t=0..2*Pi],  
      [Xsimp(2,0,7,t),Vsimp(2,0,7,t),t=0..2*Pi],  
      [Xsimp(2,0,8,t),Vsimp(2,0,8,t),t=0..2*Pi] },  
      scaling=constrained,labels=[xsimp,vsimp],  
      title="Oscilador Armonico Simple, Diagrama de Fase ");
```

Oscilador Armonico Simple, Diagrama de Fase



En muchos problemas esta primera integral será fácil de obtener, no así la integral de la posición respecto al tiempo. Por ello son muy útiles las gráficas $V(t)$ vs $X(t)$. Esas gráficas se denominan gráficas del espacio de fases, y cada una de ellas representa un valor distinto para la energía. En ella los puntos de equilibrio se identifican gráficamente. En este caso el punto de equilibrio, como se esperaba, es

Esto se obtiene fácilmente si multiplicamos por $V(t) = \frac{d}{dt} x(t)$ e integrando.

> **ecuacE:= E=(1/2)*omega^2 + (1/2)*omega0^2*theta^2;**

$$ecuacE := E = \frac{1}{2} \omega^2 + \frac{1}{2} \omega_0^2 \theta^2 \quad (3.1.1.4)$$

El primer término representa la energía cinética y el segundo la energía potencial gravitatoria. Cada una de las elipses le corresponde un valor de la Energía total

▼ Oscilador Armónico Amortiguado

Si consideramos el caso del oscilador armónico amortiguado, tendremos que la ecuación diferencial será

> **ecuacosarmamort:=diff(xamort(t),t\$2)=-2*mu*diff(xamort(t),t)-(omega0^2)*xamort(t);**

$$\text{ecuacosarmamort} := \frac{d^2}{dt^2} x_{\text{amort}}(t) = -2\mu \left(\frac{d}{dt} x_{\text{amort}}(t) \right) - \omega_0^2 x_{\text{amort}}(t) \quad (3.1.2.1)$$

y su solución será

```
> solxamortt:=dsolve({ecuacosarmamort,xamort(0)=x0,D(xamort)
(0)=v0},xamort(t));
assign(solxamortt);vamort(t):=(diff(xamort(t),t));
```

$$\text{solxamortt} := x_{\text{amort}}(t) = \frac{1}{2} \frac{(\mu x_0 + \sqrt{\mu^2 - \omega_0^2} x_0 + v_0) e^{(-\mu + \sqrt{\mu^2 - \omega_0^2})t}}{\sqrt{\mu^2 - \omega_0^2}} - \frac{1}{2} \frac{(v_0 + \mu x_0 - \sqrt{\mu^2 - \omega_0^2} x_0) e^{(-\mu - \sqrt{\mu^2 - \omega_0^2})t}}{\sqrt{\mu^2 - \omega_0^2}}$$

$v_{\text{amort}}(t) :=$

(3.1.2.2)

$$\frac{1}{2} \frac{1}{\sqrt{\mu^2 - \omega_0^2}} \left((\mu x_0 + \sqrt{\mu^2 - \omega_0^2} x_0 + v_0) (-\mu + \sqrt{\mu^2 - \omega_0^2}) e^{(-\mu + \sqrt{\mu^2 - \omega_0^2})t} - (\mu x_0 + \sqrt{\mu^2 - \omega_0^2} x_0 + v_0) (-\mu - \sqrt{\mu^2 - \omega_0^2}) e^{(-\mu - \sqrt{\mu^2 - \omega_0^2})t} \right)$$

con lo cual tendremos tres soluciones dependiendo del valor (mayor, igual o menor que cero) de la cantidad subradical $\mu^2 - \omega_0^2$ con lo cual tendremos los casos:

- subamortiguado si $\mu^2 - \omega_0^2 < 0$;
- sobreamortiguado si $\mu^2 - \omega_0^2 > 0$ y
- críticamente amortiguado si $\mu^2 - \omega_0^2 = 0$.

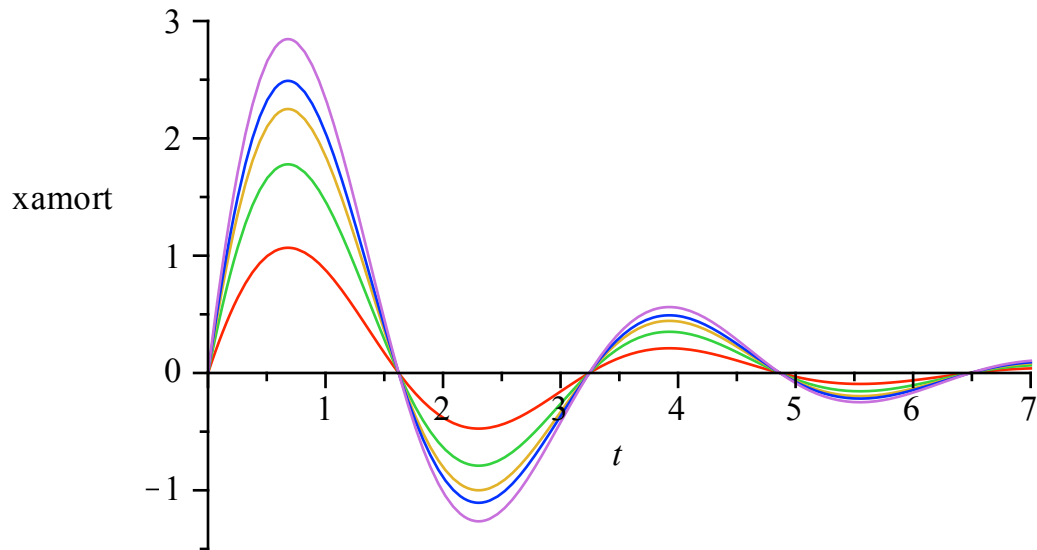
Estudiaremos los tres casos haciendo $\mu = 0,5, 3,5,$ y 2 respectivamente.

```
> mu:='mu':
Xamort:=unapply(xamort(t),mu,omega0,x0,v0,t):
Vamort:=unapply(vamort(t),mu,omega0,x0,v0,t):
```

Para el caso subamortiguado ($\mu^2 - \omega_0^2 < 0$;) tendremos la evolución del sistema, dada por

```
> plot([Xamort(0.5,2,0,3,t),Xamort(0.5,2,0,5,t),Xamort(0.5,2,
0,2*sqrt(10),t),Xamort(0.5,2,0,7,t),Xamort(0.5,2,0,8,t)],
t=0..4*Pi,labels=[t,"xamort"],title="Oscilador Armonico
Amortiguado, Evolucion temporal",scaling=constrained,
view=[0..7,-1.5..3]);
```

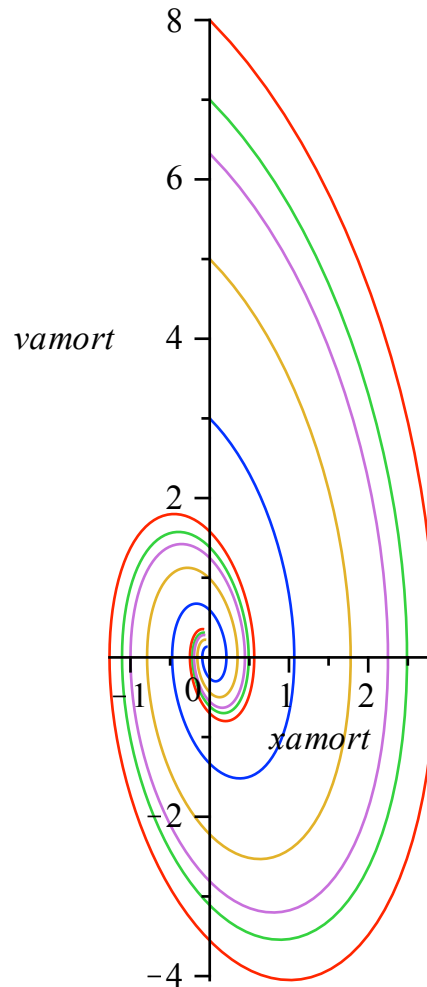

Oscilador Armonico Amortiguado, Evolucion temporal



y la representación en el espacio de fases será

```
> plot([Xamort(0.5,2,0,3,t),Vamort(0.5,2,0,3,t),t=0..2*Pi],  
[Xamort(0.5,2,0,5,t),Vamort(0.5,2,0,5,t),t=0..2*Pi],[Xamort  
(0.5,2,0,2*sqrt(10),t),Vamort(0.5,2,0,2*sqrt(10),t),t=0..2*  
Pi],[Xamort(0.5,2,0,7,t),Vamort(0.5,2,0,7,t),t=0..2*Pi],  
[Xamort(0.5,2,0,8,t),Vamort(0.5,2,0,8,t),t=0..2*Pi] },  
scaling=constrained,labels=[xamort,vamort],title="Oscilador  
Armonico Subamortiguado, Diagrama de Fase ");
```

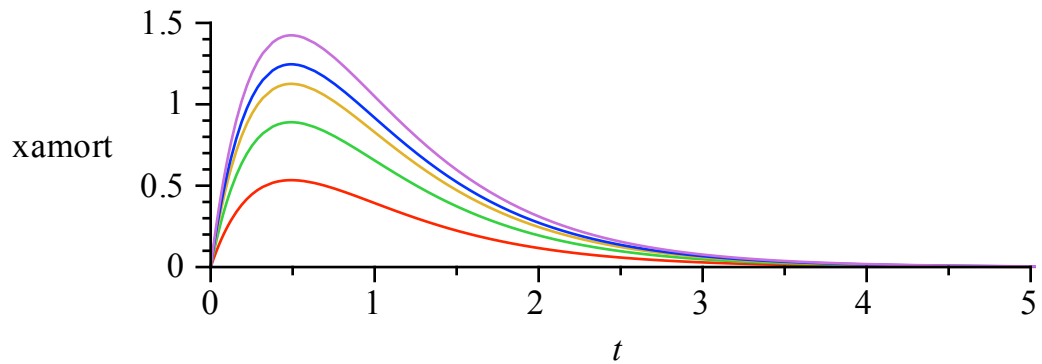
Oscilador Armonico Subamortiguado, Diagrama de Fase



Para el caso sobreamortiguado (si $\mu^2 - \omega_0^2 > 0$) la evolución temporal del oscilador será

```
> plot([Xamort(2.1,2,0,3,t),Xamort(2.1,2,0,5,t),Xamort(2.1,2,0,2*sqrt(10),t),Xamort(2.1,2,0,7,t),Xamort(2.1,2,0,8,t)],  
t=0..4*Pi,labels=[t,"xamort"],title="Oscilador Armonico Sobreamortiguado, Evolucion temporal",scaling=constrained,  
view=[0..5,0..1.5] );
```

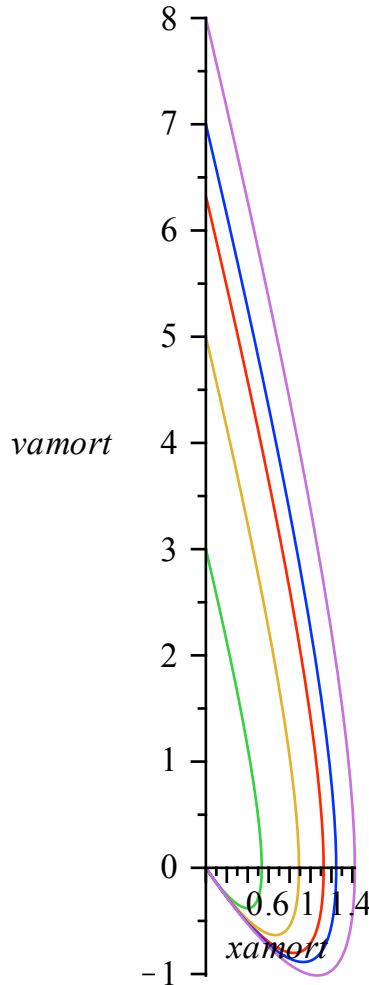
Oscilador Armonico Sobreamortiguado, Evolucion temporal



mientras que el diagrama de fase será

```
> plot([Xamort(2.1,2,0,3,t),Vamort(2.1,2,0,3,t),t=0..2*Pi],  
[Xamort(2.1,2,0,5,t),Vamort(2.1,2,0,5,t),t=0..2*Pi],[Xamort  
(2.1,2,0,2*sqrt(10),t),Vamort(2.1,2,0,2*sqrt(10),t),t=0..2*  
Pi],[Xamort(2.1,2,0,7,t),Vamort(2.1,2,0,7,t),t=0..2*Pi],  
[Xamort(2.1,2,0,8,t),Vamort(2.1,2,0,8,t),t=0..2*Pi] },  
scaling=constrained,labels=[xamort,vamort],title="Oscilador  
Armonico Sobreamortiguado, Diagrama de Fase ");
```

Oscilador Armonico Sobreamortiguado, Diagrama de Fase



El Péndulo Físico

```
[> restart;with(plots):
```

▼ Análisis del Movimiento

Consideremos ahora el caso sin aproximación, vale decir que la ecuación diferencial que describe la oscilación queda como

```
> ecuacosarmtot:=diff(theta(t),t$2)=- (omega0^2) *sin(theta(t))
;
```

$$\text{ecuacosarmtot} := \frac{d^2}{dt^2} \theta(t) = -\omega_0^2 \sin(\theta(t)) \quad (3.2.1.1)$$

Que constituye el sistema físico: Péndulo Físico, libre y sin amortiguamiento. Es decir, el sistema equivalente, sin aproximación de ángulo pequeño, al del oscilador armónico simple.

La ecuación anterior tendrá como soluciones:

```
> dsolve(ecuacosarmtot);assign(solthetat);
```

$$\int_{\theta(t)}^{\theta(t)} \frac{1}{\sqrt{2 \omega_0^2 \cos(_a) + _C1}} d_a - t - _C2 = 0, \int_{\theta(t)}^{\theta(t)} \left(\quad \right) \quad (3.2.1.2)$$

$$-\frac{1}{\sqrt{2\omega_0^2 \cos(a) + C_1}} \Big|_{a-t-C_2} = 0$$

las cuales corresponden a integrales elípticas de primera especie, las cuales son forzozamente resueltas numéricamente.

Más aún, si procedemos a incorporar las condiciones iniciales, esto es, dejamos caer el cuerpo ($\omega_0 = 0$) el péndulo desde un ángulo $\theta(0) = \theta_0$ tendremos las siguientes expresiones para la posición y la velocidad

```
> solthetat:=dsolve({ecuacosarmtot, theta(0)=theta0, D(theta)
(0)=dtheta0}, theta(t));
assign(solthetat); omega(t):=(diff(theta(t), t));
```

$$\text{solthetat} := \theta(t) = \text{RootOf} \left(- \left(\int_0^{-Z} \frac{1}{\sqrt{2\omega_0^2 \cos(a) + \text{RootOf}(d\theta_0 - \sqrt{2\omega_0^2 \cos(\theta) + Z})}} d_a \right) + t + \int_0^{\theta_0} \frac{1}{\sqrt{2\omega_0^2 \cos(a) + \text{RootOf}(d\theta_0 - \sqrt{2\omega_0^2 \cos(\theta) + Z})}} d_a \right), \theta(t) = \text{RootOf} \left(- \left(\int_0^{-Z} \left(- \frac{1}{\sqrt{2\omega_0^2 \cos(a) + \text{RootOf}(d\theta_0 + \sqrt{2\omega_0^2 \cos(\theta) + Z})}} \right) d_a \right) + t + \int_0^{\theta_0} \left(- \frac{1}{\sqrt{2\omega_0^2 \cos(a) + \text{RootOf}(d\theta_0 + \sqrt{2\omega_0^2 \cos(\theta) + Z})}} \right) d_a \right)$$

$\omega(t) :=$

(3.2.1.3)

$$\left. \begin{aligned} & - \left(2 \omega_0^2 \cos \left(\text{RootOf} \left(- \left(\int_0^{-Z} \right. \right. \right. \right. \right. \\ & \left. \left. \left. \left. \left. \frac{1}{\sqrt{2 \omega_0^2 \cos(_a) + \text{RootOf}(d\theta_0 + \sqrt{2 \omega_0^2 \cos(\theta_0) + _Z})}} \right) d_a \right) \right. \right. \right. \\ & \left. \left. \left. \left. \left. + t + \int_0^{\theta_0} \left(\right. \right. \right. \right. \right. \right. \\ & \left. \left. \left. \left. \left. - \frac{1}{\sqrt{2 \omega_0^2 \cos(_a) + \text{RootOf}(d\theta_0 + \sqrt{2 \omega_0^2 \cos(\theta_0) + _Z})}} \right) d_a \right) \right) \right) \right) \\ & \left. \left. \left. \left. \left. + \text{RootOf}(d\theta_0 + \sqrt{2 \omega_0^2 \cos(\theta_0) + _Z}) \right) \right) \right) \right)^{1/2} \end{aligned} \right)$$

Necesariamente debemos proceder por otros métodos, en este caso analizar soluciones mediante gráficos. La energía total del sistema vendrá dada por

> **ecuacpendulofisico:=C=Thetapunt^2/2 -omega0^2*cos(Theta);**

$$\text{ecuacpendulofisico} := C = \frac{1}{2} \text{Thetapunt}^2 - \omega_0^2 \cos(\Theta) \quad (3.2.1.4)$$

Procedemos a graficar el diagrama de fase para el péndulo físico libre para distintos valores de C

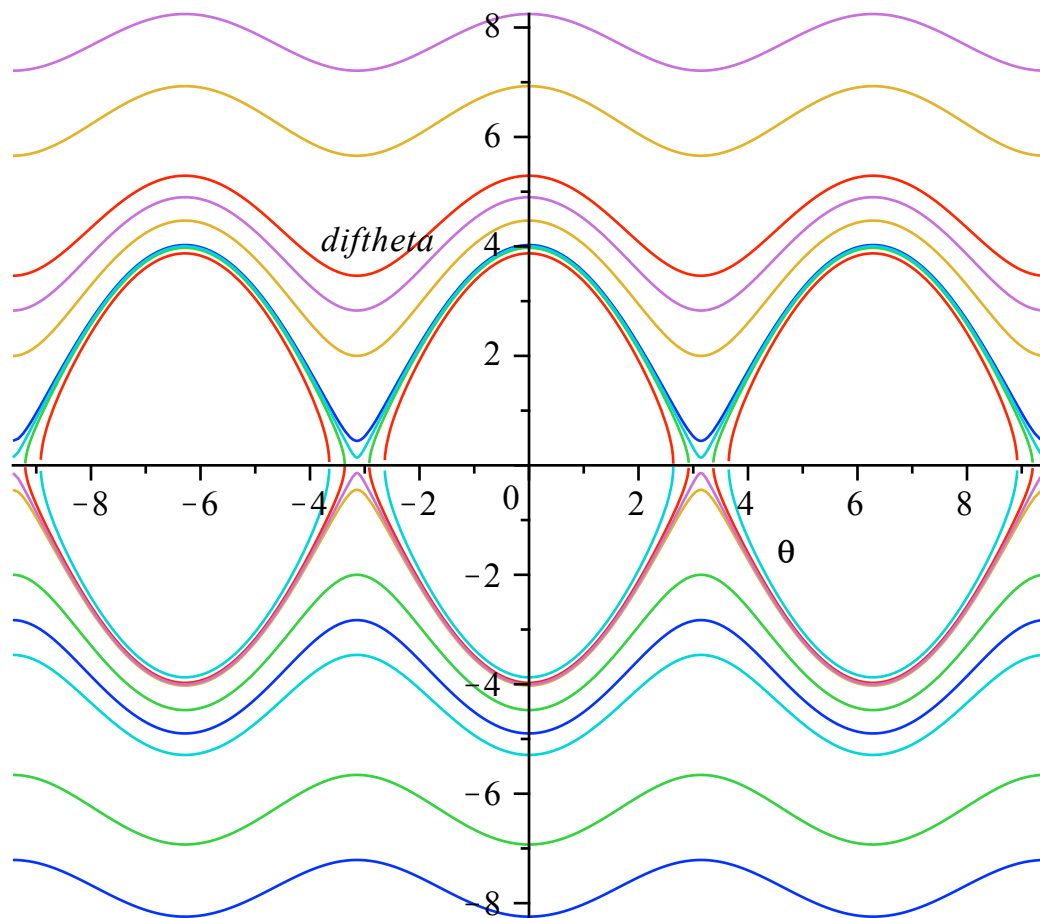
```
> ecuacpendfisico9:=solve(subs([C=3.5,omega0=2],
ecuacpendulofisico),Thetapunt):
ecuacpendfisico8:=solve(subs([C=3.9,omega0=2],
ecuacpendulofisico),Thetapunt):
ecuacpendfisico7:=solve(subs([C=30,omega0=2],
ecuacpendulofisico),Thetapunt):
ecuacpendfisico6:=solve(subs([C=20,omega0=2],
ecuacpendulofisico),Thetapunt):
ecuacpendfisico5:=solve(subs([C=10,omega0=2],
ecuacpendulofisico),Thetapunt):
ecuacpendfisico4:=solve(subs([C=8,omega0=2],
ecuacpendulofisico),Thetapunt):
ecuacpendfisico3:=solve(subs([C=6,omega0=2],
ecuacpendulofisico),Thetapunt):
ecuacpendfisico2:=solve(subs([C=4.1,omega0=2],
ecuacpendulofisico),Thetapunt):
ecuacpendfisico1:=solve(subs([C=4.01,omega0=2],
ecuacpendulofisico),Thetapunt):
plot({ecuacpendfisico1[1],ecuacpendfisico1[2],
ecuacpendfisico2[1],ecuacpendfisico2[2],
ecuacpendfisico3[1],ecuacpendfisico3[2],ecuacpendfisico4
[1],ecuacpendfisico4[2],
```

```

ecuacpendfisico5[1],ecuacpendfisico5[2],ecuacpendfisico6
[1],ecuacpendfisico6[2],
ecuacpendfisico7[1],ecuacpendfisico7[2],ecuacpendfisico8
[1],ecuacpendfisico8[2],
ecuacpendfisico9[1],ecuacpendfisico9[2]
},Theta=-3*Pi..3*Pi,scaling=constrained,labels=[theta,
diftheta],title="Péndulo Físico, Diagrama de Fase ");

```

Péndulo Físico, Diagrama de Fase



Las curvas cerradas respresentan trayectorias oscilatorias mientras que para las abiertas el ángulo de giro crece continuamente

El Período del Péndulo Físico

Analicemos como varía el período de las soluciones oscilatorias. Para ello recordamos que el período del oscilador armónico simple viene dado por

> $T_0 := 2 \cdot \pi \cdot \sqrt{L/g};$

$$T_0 := 2 \pi \sqrt{\frac{L}{g}} \quad (3.2.2.1)$$

Mientras que el período para el Péndulo Físico viene de la Integración de la ecuación de la energía y es

```

> 4*sqrt(L/g)*Int(1/sqrt(1-sin(Phi/2)^2*sin(chi)^2 ), chi=0..
Pi/2)=4*sqrt(L/g)*int(1/sqrt(1-sin(Phi/2)^2*sin(chi)^2 ),
chi=0..Pi/2);
T:=4*sqrt(L/g)*int(1/sqrt(1-sin(Phi/2)^2*sin(chi)^2 ), chi=

```

0..Pi/2):

$$4 \sqrt{\frac{L}{g}} \left(\int_0^{\frac{1}{2} \pi} \frac{1}{\sqrt{1 - \sin\left(\frac{1}{2} \Phi\right)^2 \sin^2(\chi)}} d\chi \right) \quad (3.2.2.2)$$

$$= 4 \sqrt{\frac{L}{g}} \text{EllipticK}\left(\sin\left(\frac{1}{2} \Phi\right)\right)$$

Es interesante que esa integral elíptica completa de primera especie puede ser aproximada por una serie, cuando el denominador del lado izquierdo se expande en términos del ángulo inicial de lanzamiento y se integra término a término.. Esto es:

```
> T_T0=simplify(series(T/T0,Phi,5));
```

$$T_{T0} = 1 + \frac{1}{16} \Phi^2 + \frac{11}{3072} \Phi^4 + O(\Phi^6) \quad (3.2.2.3)$$

Con esta expansión en series o directamente comparando la evaluación de la integral elíptica con el valor del período para el caso del oscilador armónico simple tendremos que podremos comprobar cuan buena es la aproximación que linealiza la función seno en la ecuación diferencial

Así, tendremos el siguiente cuadro de aproximación

```
> T00:=evalf(subs([L=2,g=9.8],T0));
TPi12:=evalf(subs([Phi=Pi/12,L=2,g=9.8],T)):Delta12:=100*
(TPi12-T00)/TPi12:
TPi6:=evalf(subs([Phi=Pi/6,L=2,g=9.8],T)):Delta6:=100*
(TPi6-T00)/TPi6:
TPi4:=evalf(subs([Phi=Pi/4,L=2,g=9.8],T)):Delta4:=100*
(TPi4-T00)/TPi4:
TPi3:=evalf(subs([Phi=Pi/3,L=2,g=9.8],T)):Delta3:=100*
(TPi3-T00)/TPi3:
TPi2:=evalf(subs([Phi=Pi/2,L=2,g=9.8],T)):Delta2:=100*
(TPi2-T00)/TPi2:
T2Pi3:=evalf(subs([Phi=2*Pi/3,L=2,g=9.8],T)):Delta23:=100*
(T2Pi3-T00)/T2Pi3:
array(1..3,1..7,[
[T00, pi/12, pi/6, pi/4, pi/3,pi/2,
2*pi/3],
["Periodo",TPi12,TPi6,TPi4,TPi3,
TPi2,T2Pi3],
["Error",Delta12,Delta6,Delta4,
Delta3,Delta2,Delta23] ]);
```

$$\left[\left[2.838453791, \frac{1}{12} \pi, \frac{1}{6} \pi, \frac{1}{4} \pi, \frac{1}{3} \pi, \frac{1}{2} \pi, \frac{2}{3} \pi \right], \quad (3.2.2.4)$$

$$\left[\text{"Periodo"}, 2.850660786, 2.887867858, 2.951916279, 3.046177536, \right.$$

$$\left. 3.350342247, 3.896857860 \right],$$


```
["Error", 0.4282163300, 1.711091692, 3.843689227, 6.819160819,
15.27869150, 27.16044842]]
```

Es claro que la aproximación es buena, ya que hasta ángulos cercanos a $\frac{1}{4}\pi$ el error que se obtiene es del orden del 3%

Integración Numérica de la Ecuación de Movimiento

Proderemos a integrar numéricamente la ecuación diferencial del péndulo físico, en este caso para un ángulo inicial de lanzamiento $\theta_0 = 0$ y para distintos valores de la velocidad inicial

A partir de las ecuacion

```
> restart;with(plots);diff(theta(t),t$2)=- (omega0^2)*sin(theta(t));
```

```
[animate, animate3d, animatecurve, arrow, changecoords, complexplot,
complexplot3d, conformal, conformal3d, contourplot, contourplot3d, coordplot,
coordplot3d, densityplot, display, dualaxisplot, fieldplot, fieldplot3d, gradplot,
gradplot3d, implicitplot, implicitplot3d, inequal, interactive, interactiveparams,
intersectplot, listcontplot, listcontplot3d, listdensityplot, listplot, listplot3d,
loglogplot, logplot, matrixplot, multiple, odeplot, pareto, plotcompare, pointplot,
pointplot3d, polarplot, polygonplot, polygonplot3d, polyhedra_supported,
polyhedraplot, rootlocus, semilogplot, setcolors, setoptions, setoptions3d,
spacecurve, sparsematrixplot, surfdata, textplot, textplot3d, tubeplot]
```

$$\frac{d^2}{dt^2} \theta(t) = -\omega^2 \sin(\theta(t)) \quad (3.2.3.1)$$

Que es una ecuación diferencial, no lineal de segundo orden, construimos varios sistemas ADIMENSIONALES de ecuaciones diferenciales ordinarias. Es importante recalcar la condición de adimensionalidad que obliga a incorporar las velocidades iniciales de lanzamiento en la constante de adimensionalización. Así los sistemas serán

```
> sist1 := subs([omega0=2, phi0=3.5, tfinal=10], diff(theta(t),
t)=phi0*tfinal*phi(t)),
subs([omega0=2, phi0=3.5, tfinal=10], diff(phi
(t), t)=-((omega0^2*tfinal)/phi0)*sin(theta(t)));
sist2 := subs([omega0=2, phi0=3.9, tfinal=10], diff(theta(t),
t)=phi0*tfinal*phi(t)),
subs([omega0=2, phi0=3.9, tfinal=10], diff(phi
(t), t)=-((omega0^2*tfinal)/phi0)*sin(theta(t)));
sist3 := subs([omega0=2, phi0=4, tfinal=10], diff(theta(t), t)=
phi0*tfinal*phi(t)),
subs([omega0=2, phi0=4, tfinal=10], diff(phi(t),
t)=-((omega0^2*tfinal)/phi0)*sin(theta(t)));
sist4 := subs([omega0=2, phi0=4.1, tfinal=10], diff(theta(t),
t)=phi0*tfinal*phi(t)),
subs([omega0=2, phi0=4.1, tfinal=10], diff(phi
(t), t)=-((omega0^2*tfinal)/phi0)*sin(theta(t)));
```

```

sist5 := subs([omega0=2,phi0=4.5,tfinal=10],diff(theta(t),
t)=phi0*tfinal*phi(t)),
      subs([omega0=2,phi0=4.5,tfinal=10],diff(phi
(t),t)=-((omega0^2*tfinal)/phi0)*sin(theta(t)));
funs := {theta(t), phi(t)};
soluc1:= dsolve({sist1,theta(0)=0,phi(0)=1},funs,type=
numeric,abserr=0.00001):
soluc2:= dsolve({sist2,theta(0)=0,phi(0)=1},funs,type=
numeric,abserr=0.00001):
soluc3:= dsolve({sist3,theta(0)=0,phi(0)=1},funs,type=
numeric,abserr=0.00001):
soluc4:= dsolve({sist4,theta(0)=0,phi(0)=1},funs,type=
numeric,abserr=0.00001):
soluc5:= dsolve({sist5,theta(0)=0,phi(0)=1},funs,type=
numeric,abserr=0.00001):
graf1:=odeplot(soluc1, [t,theta(t)], 0..1, numpoints=50):
graf2:=odeplot(soluc2, [t,theta(t)], 0..1, numpoints=50):
graf3:=odeplot(soluc3, [t,theta(t)], 0..1, numpoints=50):
graf4:=odeplot(soluc4, [t,theta(t)], 0..1, numpoints=50):
graf5:=odeplot(soluc5, [t,theta(t)], 0..1, numpoints=50):
graf10:=odeplot(soluc1, [theta(t),phi(t)], 0..1, numpoints=
50):
graf20:=odeplot(soluc2, [theta(t),phi(t)], 0..1, numpoints=
50):
graf30:=odeplot(soluc3, [theta(t),phi(t)], 0..1, numpoints=
50):
graf40:=odeplot(soluc4, [theta(t),phi(t)], 0..1, numpoints=
50):
graf50:=odeplot(soluc5, [theta(t),phi(t)], 0..1, numpoints=
50):

```

$$sist1 := \frac{d}{dt} \theta(t) = 35.0 \phi(t), \quad \frac{d}{dt} \phi(t) = -11.42857143 \sin(\theta(t))$$

$$sist2 := \frac{d}{dt} \theta(t) = 39.0 \phi(t), \quad \frac{d}{dt} \phi(t) = -10.25641026 \sin(\theta(t))$$

$$sist3 := \frac{d}{dt} \theta(t) = 40 \phi(t), \quad \frac{d}{dt} \phi(t) = -10 \sin(\theta(t))$$

$$sist4 := \frac{d}{dt} \theta(t) = 41.0 \phi(t), \quad \frac{d}{dt} \phi(t) = -9.756097561 \sin(\theta(t))$$

$$sist5 := \frac{d}{dt} \theta(t) = 45.0 \phi(t), \quad \frac{d}{dt} \phi(t) = -8.888888889 \sin(\theta(t))$$

$$funs := \{\phi(t), \theta(t)\}$$

(3.2.3.2)

Las cuales se integran numericamete y proveen las siguientes gráficas

```

> display([graf1,graf2,graf3,graf4,graf5],view=[0..1,-2*Pi.
.2*Pi]);
display([graf10,graf20,graf30,graf40,graf50],scaling=
constrained,view=[-5..5,-2..2]);

```

