

Resolución de Ecuaciones Diferenciales Ordinarias Lineales con la symbolic toolbox de Matlab: respuesta forzada de un muelle

© 2022, Antonio Sala Piqueras, Universitat Politècnica de València. Todos los derechos reservados.

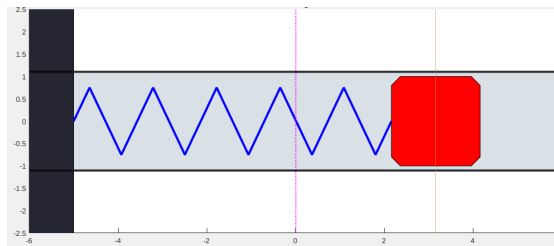
Este código funcionó sin errores en Matlab R2022a

Presentación en vídeo: <http://personales.upv.es/asala/YT/V/masmusymForz.html>

Objetivo: analizar la solución de EDOs lineales provista por Matlab (symbolic toolbox), comprender su significado y representarla gráficamente; en este caso, haremos el ejemplo con un muelle "forzado".

EDO en forma no normalizada

Sistema a modelar:



Resolvamos la ecuación de un sistema masa-muelle-amortiguador (linealizado, equilibrio cero), ante condiciones iniciales fuera del equilibrio y entrada de fuerza, escrita como:

$$M \cdot \frac{d^2 y(t)}{dt^2} = -k \cdot y(t) - b \cdot \frac{dy(t)}{dt} + F(t)$$

```
M=0.5;k=2; b=.25; %parámetros constantes, damos valores numéricos
syms y(t) F(t) %función del tiempo simbólica
d1ydt=diff(y); % notación de la velocidad
d2ydt=diff(y,2); % notación de la aceleración
```

```
EDOmuelle= M*d2ydt== -k*y -b*d1ydt +F(t) %si fuera no lineal, no se podría en general en
```

```
EDOmuelle(t) =
```

$$\frac{\frac{\partial^2}{\partial t^2} y(t)}{2} = -\frac{\frac{\partial}{\partial t} y(t)}{4} + F(t) - 2 y(t)$$

Solución general

Sin condiciones iniciales fijadas, hay infinitas soluciones formalmente relacionadas con dos constantes de integración (y cond. iniciales), y términos "forzados":

```
sol=dsolve(EDOmuelle)
```

sol =

$$C_1 e^{-\frac{t}{4}} \sigma_2 - C_2 e^{-\frac{t}{4}} \sigma_1 - e^{-\frac{t}{4}} \sigma_2 \int \frac{8 \sqrt{7} e^{t/4} \sigma_1 F(t)}{21} dt + e^{-\frac{t}{4}} \sigma_1 \int \frac{8 \sqrt{7} e^{t/4} \sigma_2 F(t)}{21} dt$$

where

$$\sigma_1 = \sin\left(\frac{3 \sqrt{7} t}{4}\right)$$

$$\sigma_2 = \cos\left(\frac{3 \sqrt{7} t}{4}\right)$$

Solución particular para unas CI y fuerza dadas

Sustituimos una fuerza $F(t)$ (la función del tiempo arbitraria que queramos)

```
EDOmuelle=subs(EDOmuelle, F(t), -2.5+3*sin(6*t));
```

Y una posición y velocidad iniciales (las que queramos):

```
posmuelle=simplify(dsolve(EDOmuelle, y(0)==3, dlydt(0)==0)) %parado en otro sitio inicial
```

posmuelle =

$$\frac{17633 e^{-\frac{t}{4}} \cos\left(\frac{3 \sqrt{7} t}{4}\right)}{4132} - \frac{192 \sin(6 t)}{1033} - \frac{18 \cos(6 t)}{1033} + \frac{36065 \sqrt{7} e^{-\frac{t}{4}} \sin\left(\frac{3 \sqrt{7} t}{4}\right)}{86772} - \frac{5}{4}$$

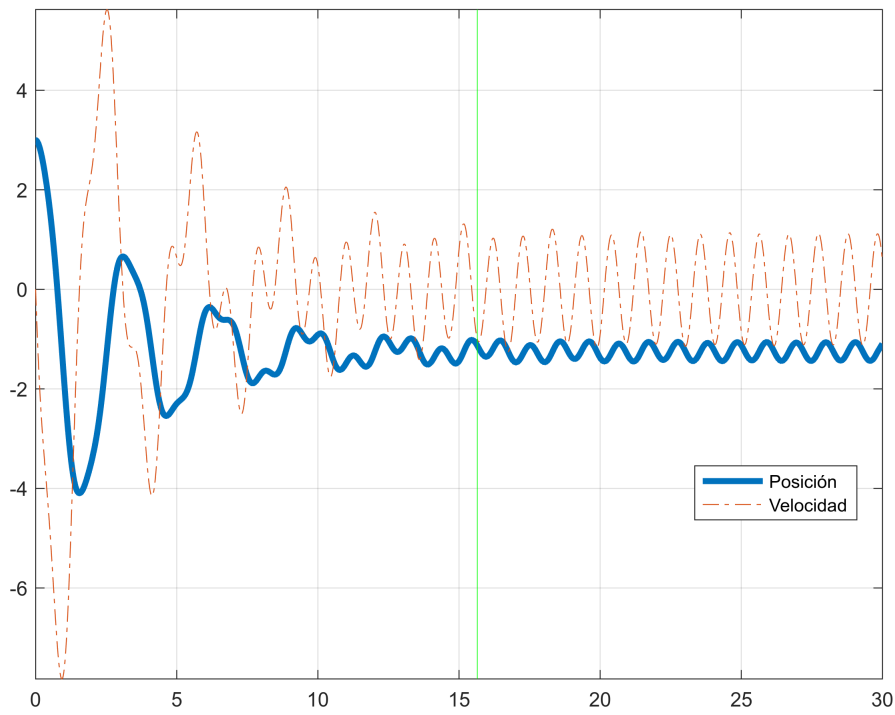
El "transitorio" dura más o menos $e^{-\frac{t_{est}}{4}} \approx 0.02$, $t_{est} \approx \log 0.02 * (-4)$

```
t_est=log(.02)*(-4)
```

```
t_est = 15.6481
```

Dibujemos la solución (posición y velocidad):

```
Tf=30;  
fplot(posmuelle,[0 Tf],LineWidth=3), hold on  
fplot(diff(posmuelle),[0 Tf],LineStyle='-.'), hold off, grid on,  
xline(t_est,'g')  
legend("Posición","Velocidad", location='best');
```



```
fplot(posmuelle,[0 Tf],LineWidth=3), hold on
fplot(diff(posmuelle),[0 Tf],LineStyle='-.'), hold off, grid on,
xline(t_est,'g')
legend("Posición","Velocidad", location='best')
```

