

Solución de ecuaciones algebraicas y diferenciales empleando Matlab

1. Solución de ecuaciones algebraicas

Se desea resolver el siguiente sistema de ecuaciones algebraicas:

$$\begin{aligned}f_1 &= a + bx_1 - x_1^3 \\f_2 &= cx_2 + a\end{aligned}$$

Para resolver varias ecuaciones algebraicas simultáneas $f(x)=0$ empleando Matlab se emplea una función del toolbox de optimización llamada `fsolve`. Se requiere hacer un "archivo m" (*m-file*) que describa los parámetros, las funciones y las variables involucradas. La estructura general del *m-file* es:

```
% Comentarios, instrucciones, etc.
function [f]=ecalg(x)
% Parámetros
a=6;b=1.5;c=-14;
% Ecuaciones
f(1)=a+b*x(1)-x(1)^3;
f(2)=c*x(2)+a;
```

Notas:

- Cualquier comentario debe iniciarse con %
- El comando `function` debe ir al inicio
- El vector **f** contiene las funciones (en este caso $f(1)$ y $f(2)$)
- El vector **x** contiene las variables (en este caso $x(1)$ y $x(2)$)
- `ecalg` es el nombre de la función y debe ser el nombre con el que se guarda el archivo. Conviene emplear un nombre que identifique al problema, en este caso, *ecuaciones algebraicas*. Así, el archivo debe guardarse como `ecalg.m`
- Todos los parámetros y valores empleados por las funciones deben introducirse antes de definir las funciones $f(1)$ y $f(2)$.

Para resolver el sistema de ecuaciones algebraicas definido en el archivo `ecalg.m` se debe introducir en la ventana de comandos de Matlab la instrucción:

```
>>x0=[1,1]; [x,feval,flag]=fsolve('ecalg',x0)
```

Esta orden llamará la función `fsolve` y Matlab empleando el toolbox de optimización encontrará los valores de $x(1)$ y $x(2)$ que satisfacen que $f(1)=f(2)=0$ empleando los valores iniciales $x_0 = [1,1]$ para $x(1)$ y $x(2)$ respectivamente.

Los valores reportados serán los valores de $x(1)$ y $x(2)$ agrupados en el vector **x**. El resultado **feval** proporciona el valor de las funciones $f(1)$ y $f(2)$ evaluadas con los valores de **x** encontrados, y la variable **flag** nos indica si el paquete de optimización encontró una solución.

Es necesario verificar que la respuesta del sistema si sea una raíz. Esto se logra cuando se cumple lo siguiente:

- a. se lee el mensaje **optimization terminated succesfully**.
- b. el valor de la variable **feval** es prácticamente cero (al menos igual o menor a $1e-5$).

c. la variable **flag** tiene un valor de 1.

En este ejemplo los resultados proporcionados son:

```
Optimization terminated successfully:
First-order optimality less than OPTIONS.TolFun, and no negative/zero
curvature detected
x =
    2.0905    0.4286
feval =
    1.0e-007 *
   -0.4932         0
flag =
     1
```

Por tanto para este caso las raíces son 2.0905 y 0.4286.

Para mayores detalles introducir `help fsolve` en la ventana de comandos de Matlab.

2. Solución de ecuaciones diferenciales

Se desea resolver el siguiente sistema de ecuaciones algebraicas:

$$\begin{aligned}\frac{dx_1}{dt} &= a + bx_1 - x_1^3 \\ \frac{dx_2}{dt} &= cx_2 + a\end{aligned}$$

Para resolver ecuaciones diferenciales en Matlab se sigue un procedimiento equivalente al empleado en la solución de ecuaciones algebraicas. La diferencia es que se emplea una función para integrar (p.ejm. `ode45`, `ode23`, `ode15s`).

Se requiere hacer un archivo m (*m-file*) que describa los parámetros, las funciones y las variables involucradas. La estructura general del *m-file* es:

```
% Comentarios, instrucciones, etc.
function [f]=ecdif(t,x)
% Parámetros
a=6;b=1.5;c=-14;
% Ecuaciones
f(1)=a+b*x(1)-x(1)^3;
f(2)=c*x(2)+a;
f = f';
```

Notas:

- Cualquier comentario debe iniciarse con %
- El comando `function` debe ir al inicio
- El vector **f** contiene las funciones (en este caso $f(1)$ y $f(2)$)
- El vector **x** contiene las variables (en este caso $x(1)$ y $x(2)$)

- El vector \mathbf{t} contiene la variable de tiempo
- *ecdif* es el nombre de la función y debe ser el nombre con el que se guarda el archivo. Conviene emplear un nombre que identifique al problema, en este caso, *ecuaciones diferenciales*. Así, el archivo debe guardarse como *ecdif.m*
- Todos los parámetros y valores empleados por las funciones deben introducirse antes de definir las funciones $f(1)$ y $f(2)$.

Para resolver el sistema de ecuaciones diferenciales definido en el archivo *ecdif.m* se debe introducir en la ventana de comando de Matlab la instrucción:

```
>>tr=[0,1];x0=[1,1];[t,x]=ode45('ecdif',tr,x0);
```

Esta orden llamará la función *ode45* y resolverá el sistema *ecdif* empleando ese método de integración en un rango de tiempo de $\mathbf{tr}=[0,1]$ partiendo de un valor inicial de $x(1)$ y $x(2)$ igual a uno ($\mathbf{x0}=[1,1]$). Los valores reportados serán los valores de $x(1)$ y $x(2)$ agrupados en la matriz \mathbf{x} y los valores del tiempo en el vector columna \mathbf{t} .

Para mayores detalles introducir `help ode45` (`help ode23`, `helpode15s`) y `help odefile` en la ventana de comandos de Matlab.

Para graficar los resultados obtenidos de la simulación dinámica (solución de las ecuaciones diferenciales en un rango de tiempo) simplemente se introduce en la ventana de comandos de Matlab:

```
>>plot(t,x)
```

La gráfica obtenida se muestra a continuación. Para mayores detalles introducir `help plot` en la ventana de comandos de Matlab.

