DEPARTAMENTO DE MATEMATICA APLICADA
UNIVERSIDAD COMPLUTENSE DE MADRID
Análisis Numérico
PRACTICAS. Hoja 2
Métodos Monopaso

Práctica 4. (Estructura general de las prácticas usando rutinas propias de Matlab para resolver EDO). La ecuación del corazón. Considerar la ecuación diferencial $x' = f(t, x)$

$$\begin{cases} x_1' = x_2 \\ x_2' = -16x_1 + 4\,\mathrm{sen}(2t) \\ x_1(0) = 0 \\ x_2(0) = 2 \end{cases}$$

en el intervalo, $[0, 2\pi]$

1. Escribir un fichero **tipo función**, con el nombre **funccorazon.m,** para la función de esta ecuación diferencial, es decir la expresión $f(t, x)$, en concreto:

```
function f=funccorazon(t,x)
f=[x(2);-16*x(1)+4*sin(2*t)];
```

2. Usar, en la lineas de comandos, las propias rutinas de Matlab para resolver la ecuación numéricamente, utilizando un método adaptativo de Runge-Kutta de orden 4, **ode45**. En concreto, escribir

```
>> [t,u]=ode45(@funccorazon,[0,2*pi],[0;2])
>> [t,u]=ode45(@funccorazon,[0:0.01:2*pi],[0;2])
>> u=u.';
>> figure(1)
>> subplot(2,1,1)
>> plot(t,u(1,:))
>> subplot(2,1,2)
>> plot(t,u(2,:))
>> figure(2)
>> plot(u(1,:),u(2,:))
```

$\begin{cases} x_1 = x \\ x_2 = x' \end{cases} \Rightarrow \begin{cases} x_1' = x_2 \\ x_2' = (1 - x_1^2)\,x_2 - x_1 \end{cases}$

3. Repetir los puntos 1 y 2 de arriba, para el siguiente problema de valor inicial: **Ecuación de Van der Pol**: $x''(t) + (x^2(t) - 1)x'(t) + x(t) = 0$, $t \in [0, 10]$, con datos iniciales $x(0) = 0{,}1$, $x'(0) = 0{,}2$. El fichero función que contiene la función de la ecuación diferencial debe llamarse **funcvanderpol.m**. Explorar también el comportamiento de las soluciones para otros valores inciales.

4. Crear un fichero tipo script llamado **mispracticas.m** que contenga los datos de entrada del comando **ode45**, para cada una de las ecuaciones que vamos a resolver, es decir:

```
fun=@funcvanderpol; x0=[0.1;0.2]; tinic=0; tfin=10; N=1000;
```

5. Crear un fichero tipo script con nombre **misgraficas.m** que pinte

   a) Si la EDO es escalar: la gráfica de la solución aproximada.

   b) Si la EDO es en $I\!R^2$ o en $I\!R^3$: la gráfica de cada una de las componentes en una misma pantalla usando los comandos `subplot`, `plot` y, tras una pausa, pinte en otra ventana la trayectoria de la solución .

   Opcional: Usar el comando `title` para poner un título a la ventana y el comando `legend` para indicar la curva en cada subventana.

   Código obligatorio de colores: Las gráficas para problemas escalares y las trayectorias para problemas 2D y 3D, han de pintarse **en rojo**. Para problemas 2D, las componentes han de pintarse **en rojo y verde**, respectivamente. Para problemas 3D las componentes han de pintarse en **rojo, verde y azul** respectivamente.

6. Crear un fichero tipo script con nombre **testmiode45.m** que haga lo siguiente:

   i) Lea los datos correspondientes a una ecuación del fichero **mispracticas.m**. Para esto será necesario comentar en **mispracticas.m** todas aquellas lineas que no corresponden a esta ecuación. Para comentar una linea ponemos el símbolo % delante de ella;

   ii) ejecute `ode45.m`;

   iii) pinte las gráficas de la solución haciendo una llamada al fichero **misgraficas.m**, es decir

```
mispracticas
[t,u]=ode45(fun,[tinic ,tfin],x0)
u=u.';
misgraficas
```

Práctica 5. 1. Crear un fichero tipo función con MATLAB, con nombre **mieuler.m**, que implemente el Método de Euler, evaluando la función de la EDO de un fichero externo.

2. Crear un fichero tipo script con nombre **testmieuler.m** que ejecute **mieuler.m** y pinte las gráficas de las soluciones haciendo una llamada al fichero **misgraficas.m**.

Práctica 6. Repetir la práctica anterior para el Método de Runge–Kutta de orden 4 (ficheros **mirk4.m, testmirk4.m**).

Práctica 7. Repetir la práctica anterior para el Método del Trapecio (ficheros **mitrap.m, testmitrap.m**). **Observación:** en cada paso tienes que resolver una ecuación implícita $z = g(h, x, z)$. Utilizar el método de Newton; tienes que incorporar a las variables de entrada `jac, itmax,` donde `jac` es el nombre del fichero con el jacobiano de `f`, e `itmax` es el número máximo de iteraciones en el método de Newton.

Práctica 8. Explorar, usando el Método de Runge–Kutta de orden 4, y el del trapecio, el comportamiento de las soluciones de **una ecuación rígida:** $x'(t) = -50(x(t) - \cos(t))$ en el intervalo $0 \leq t \leq \pi/2$, con dato inicial $x(0) = 0$, para varias elecciones de $N = 25, 40, 50, 100...$

Práctica 9. Considerar la **ecuación del péndulo** $\begin{cases} \theta''(t) + 2\beta\theta'(t) + \frac{g}{l}\operatorname{sen}(\theta(t)) = M \\ \theta(0) = \theta_0, \quad \theta'(0) = \omega_0 \end{cases}$.

1. Suponiendo que $M = 0$, $l = 1m$ y que $g = 9{,}8m/s^2$ explorar, usando el Método de Runge–Kutta de orden 4, el comportamiento de las soluciones en el intervalo $0 \leq t \leq 10$, para varias elecciones de los datos iniciales. Tomar primero el valor $\beta = 0$ y posteriormente $\beta = 0{,}25$ y $\beta = 1{,}5$.

2. Para cada uno de los datos iniciales escogidos, comparar los valores encontrados con los obtenidos al resolver el **problema linealizado** $\begin{cases} \theta''(t) + 2\beta\theta'(t) + \frac{g}{l}\theta(t) = 0 \\ \theta(0) = \theta_0, \quad \theta'(0) = \omega_0 \end{cases}$.

   Observar la similitud de las soluciones y trayectorias unicamente cuando los datos iniciales son pequeños.

3. ¿Que ocurre cuando partimos con el péndulo en posición vertical, es decir con $\theta_0 = \pi$, $\omega_0 = 0$?.

4. Tomar ahora $\beta = 1/2$. Comprobar que si $M = g/l = 9{,}8$ entonces $\theta_0 = \pi/2$, $\omega_0 = 0$ es un equilibrio del péndulo. Tomar datos iniciales próximos a este equilibrio y variar M con los valores $M = 9{,}7$, $M = 9{,}8$ y $M = 9{,}9$ y observar el cambio de comportamiento.

   Notación obligatoria: Los ficheros conteniendo las funciones de estas EDOs deben llamarse **funcpendulo.m y funcpendulolin.m**.

   Sugerencia: Se pueden definir los parámetros como una variable de entrada de nombre `par`, que se escribe al final del fichero que contiene el segundo miembro de la ecuación diferencial y que se deja vacía en caso de que la función no contenga parámetros.

```
function fun=funcpendulo(t,x,par)
M=par(1);
l=par(2);
g=par(3);
b=par(4);

fun=[x(2,:); ...
     -2*b*x(2,:)-(g/l)*sin(x(1,:))+M];
```

Práctica 10. Para cada uno de los **problemas autónomos** siguientes explorar, usando el Método de Runge–Kutta de orden 4, el comportamiento de las soluciones en el intervalo $0 \leq t \leq 10$, para varias elecciones de los datos iniciales.

1. **Sistema depredador-presa:** $\begin{cases} x'(t) = ax(t) - bx(t)y(t) \\ y'(t) = -cy(t) + dx(t)y(t) \end{cases}$, donde los coeficientes a, b, c, d son no negativos. Tomar los casos $a = 1$, $b = 1$, $c = 1$, $d = 2$ y $a = 3$, $b = 0{,}2$, $c = 0{,}6$, $d = 5$ y datos iniciales $x(0), y(0) > 0$. Toma los valores $a = 0,4$, $b = 0,01$, $c = 0,3$, $d = 0{,}005$, y datos iniciales $(x(0), y(0)) = (30, 50)$ y $(x(0), y(0)) = (100, 5000)$ en el intervalo de tiempo $[0, 100]$.

2. **Sistema de competición:** $\begin{cases} x'(t) = ax(t) - bx(t)y(t) - ex^2(t) \\ y'(t) = -cy(t) + dx(t)y(t) - fy^2(t) \end{cases}$, donde los coeficientes a, b, c, d, e, f son no negativos. Tomar los valores $a = b = c = 1$, $d = 0{,}2$, $e = 0{,}4$, y $f = 0{,}0001$ y datos iniciales $x(0), y(0) > 0$.

3. **Ecuación de Van der Pol** $x''(t) + \alpha(x^2(t) - \beta)x'(t) + x(t) = 0$. Considerar primero $\alpha = 1$ y toma datos iniciales próximos al origen para $\beta = -0{,}2$, $\beta = 0$ y $\beta = 0{,}2$. Fija despues $\beta = 1$ y aumenta el valor de α desde $\alpha = 1$ hasta $\alpha = 10$.

4. **Ecuación de Duffing** $x''(t) + \alpha x'(t) + x^3(t) - x(t) = 0$. Tomar primero $\alpha = 0$ y explora las soluciones entorno a los 3 equilibrios del problema. Tomar ahora $\alpha = 1$ y estudia el cambio en el comportamiento de las soluciones.

5. **Sistema de Lorenz** $\begin{cases} x'(t) = \sigma(y(t) - x(t)) \\ y'(t) = \rho x(t) - y(t) - x(t)z(t) \\ z'(t) = x(t)y(t) - \beta z(t) \end{cases}$, tomando $\sigma = 10$, $\beta = 8/3$. Ve aumentando desde $\rho = 0{,}1$ a $\rho = 30$. Observar la dinámica en los valores intermedios $\rho = 1$, $\rho = 13{,}962$ y $\rho = 24{,}74$.

   Tomar $\rho = 100{,}5$ y el dato inicial $(0, 5, 75)$ para ver una solución periódica. Manteniendo el dato inicial mueve ρ entre 99,524 y 100,795 (p.ej. $\rho = 99{,}65$) y observar el cambio de dinámica.

Notación obligatoria: Los ficheros conteniendo las funciones de estas EDOs deben llamarse **funcdeppresa.m, funccompet.m, funcvanderpol.m, funcduffing.m y funclorenz.m** respectivamente.

Práctica 11. Explorar, usando el Método de Runge–Kutta de orden 4, y el del trapecio, el comportamiento de las soluciones de **la ecuación de Van der Pol** con los siguientes datos: $x_0 = (2, 8)$, $t \in [0, 100]$, $N = 4000$, $\alpha = 30$, $\beta = 1$.

Práctica 12. Explorar, usando el Método del trapecio, el comportamiento de las soluciones de **la ecuación de Belousov-Zhabotinsky** dada por

$$\begin{cases} x_1' &= \alpha(x_2 - x_1x_2 + x_1 - \beta x_1^2) \\ x_2' &= \frac{1}{\alpha}(\gamma x_3 - x_2 - x_1x_2) \\ x_3' &= \delta(x_1 - x_3) \end{cases}$$

donde $\alpha, \beta, \gamma, \delta$ son constantes positivas. Considerar el dato inicial $x_0 = (0{,}25; 0{,}75; 0{,}25)$, y valores de los parámetros $\alpha = 2 \times 10^4, \beta = 8 \times 10^{-4}, \gamma = 5000, \delta = 0{,}75$. Comprobar que el sistema es extremadamente rígido. **Notación obligatoria:** El fichero conteniendo la función de esta EDO debe llamarse **funbelza.m**.