Análisis Numérico
Curso 2015–2016
Prácticas
Hoja 2. Métodos monopaso

1 Estructura general de las prácticas para resolver ecuaciones de la forma $\mathbf{x}'(\mathbf{t}) = \mathbf{f}(\mathbf{t}, \mathbf{x}(\mathbf{t}))$.

Se considera el problema de valor inicial

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

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

a) Escribir un fichero de tipo función, con el nombre `funccorazon.m`, para la función $f(t, x)$ de esta ecuación diferencial con el siguiente contenido:

```
function f=funccorazon(t,x)
f1=x(2);
f2=-16*x(1)+4*sin(2*t);
f=[f1;f2];
```

b) Resolver el problema de valor inicial anterior utilizando la función `ode45` de `MATLAB`, que resuelve numéricamente problemas de valor inicial mediante un método adaptativo de tipo Runge–Kutta. Concretamente, escribir

```
[t,x]=ode45(@funccorazon,[0 2*pi],[0 2])
[t,x]=ode45(@funccorazon,[0:0.01:2*pi],[0 2])
figure(1)
subplot(2,1,1)
plot(t,x(:,1))
subplot(2,1,2)
plot(t,x(:,2))
figure(2)
plot(x(:,1),x(:,2))
```

c) Repetir los apartados a) y b) para los siguientes problemas de valor inicial:

I) **Oscilador armónico**: Supongamos que un cuerpo de masa m está sujeto en el extremo de un muelle.

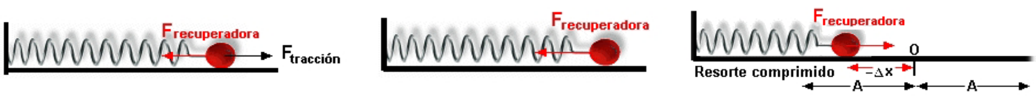


Figura 1: *Movimiento armónico simple*

Si desplazamos la masa respecto de su posición de equilibrio y después la soltamos, a partir de la Segunda Ley de Newton y de la Ley de Hooke (que afirma que el muelle ejerce una fuerza de restitución proporcional al alargamiento), en ausencia de rozamiento y de fuerzas externas, se tiene que el desplazamiento $x(t)$ respecto a la posición de equilibrio (*elongación*) en el instante de tiempo t es solución del problema de valor inicial

$$\begin{cases} mx''(t) = -kx(t),\ t \in [0, T] \\ x(0) = x_0 \\ x'(0) = v_0. \end{cases}$$

Considerar $k = m = 1$, $T = 10$, $x_0 = 1$, $v_0 = 0$ y llamar `funcosciladorarmo.m` al fichero de tipo función que contiene la función $f(t, x)$ del problema anterior. Explorar el comportamiento de las soluciones para otros datos iniciales.

II) **Ecuación de Van der Pol**: Describe el comportamiento de circuitos electrónicos no lineales

$$\begin{cases} x''(t) + \alpha\left(x^2(t) - \beta\right)x'(t) + x(t) = 0,\ t \in [0, T] \\ x(0) = x_0 \\ x'(0) = v_0. \end{cases}$$

Considerar $\alpha = \beta = 1$, $T = 10$, $x_0 = 0'1$, $v_0 = 0'2$ y llamar `funcvanderpol.m` al fichero de tipo función que contiene la función $f(t, x)$ del problema anterior. Explorar, también, el comportamiento de las soluciones para otros datos iniciales.

d) Crear un fichero de tipo script llamado `datos.m` que contenga, en líneas distintas, los datos de entrada del comando `ode45` para una malla de `N+1` puntos equiespaciados, para cada una de las ecuaciones que se van a resolver, es decir,

```
f=@funccorazon; intervalo=[0 2*pi]; x0=[0 2]; N=1000;
% f=@funcosciladorarmo; intervalo=[0 10]; x0=[1 0]; N=1000;
% f=@funcvanderpol; intervalo=[0 10]; x0=[0.1 0.2]; N=1000;
```

Comentar en `datos.m` todas las líneas (usando para ello el símbolo `%`) excepto la correspondiente a la práctica que se desee ejecutar.

e) Crear un fichero tipo script con nombre `graficas.m` que, al ejecutarlo tras obtener la solución aplicando la función `ode45` de `MATLAB` como en el apartado b), dibuje:

I) Si la ecuación diferencial es escalar: la gráfica de la solución.

II) Si la ecuación diferencial es en $\mathbb{R}^2$ o en $\mathbb{R}^3$: la gráfica de cada una de las componentes, en una misma ventana, usando los comandos `subplot`, `plot` y que, tras una pausa, dibuje en otra ventana la trayectoria de la solución.
Opcional: Utilizar el comando `title` y/o `legend` para indicar la curva de cada subventana.

Código de colores: Para unificar la notación, se utilizará el siguiente código de colores:

- Las gráficas para problemas escalares y las trayectorias para problemas 2D y 3D, serán de color rojo.
- Para problemas 2D, las componentes se dibujarán, respectivamente, en rojo y verde.
- Para problemas 3D los colores de las componentes serán, respectivamente, en rojo, verde y azul.

f) Crear un fichero tipo script con nombre `testode45.m` que lea los datos del fichero `datos.m`, ejecute `ode45.m` y dibuje utilizando el fichero `graficas.m`

Como regla general, en cada uno de los ficheros que ejecutan los programas principales (la mayor parte de los cuales se llaman `test***.m`), introducir en la primera línea la sentencia `datos` para que, de esta forma, se lea en primer lugar el contenido de `datos.m`

Para evitar conflictos con las variables utilizadas, emplear, adecuadamente, la sentencia `clear all`

2 Método de Euler (explícito).

a) Crear un fichero de tipo función, de nombre `euler.m`, que implemente el *Método de Euler*, evaluando la función de la ecuación diferencial de un fichero externo. El fichero `euler.m` empezará de la siguiente forma:

```
function [t,x]=euler(f,intervalo,x0,N)

% La función euler resuelve un problema de valor inicial de la forma
%  x'=f(t,x) en [t0,T]
%  x(t0)=x0,
% con x0 en R^n, mediante el método de Euler (explícito).
%
% ENTRADA:
%  f=@func: función f definida en el fichero func.m
%  intervalo: [t0,T], donde está planteado el sistema de ecuaciones diferenciales
%  x0: vector inicial de R^n
%  N: número de subintervalos
%
% SALIDA:
%  t: vector columna de abscisas donde se va a aproximar la solución de tipo (N+1,1)
%  x: matriz de ordenadas de la solución aproximada de tipo (N+1,n)
```

b) Crear un fichero de tipo script, de nombre `testeuler.m` que lea los datos del fichero `datos.m`, ejecute `euler.m` y dibuje la salida del algoritmo `euler.m` haciendo una llamada al fichero `graficas.m`

3 Método de Euler modificado. Repetir la Práctica 2 para el *Método de Euler modificado* (ficheros `eulermod.m` y `testeulermod.m`).

$$
\begin{array}{ll}
\underline{\text{Paso 1}} & x_0 \simeq \xi_0 \\
\underline{\text{Paso 2}} & \text{Para } i = 0, 1, \ldots, N-1 \\
& \left\{ \begin{array}{l} F_1 = f\left(t_i, x_i\right) \\ F_2 = f\left(t_i + \dfrac{h}{2}, x_i + \dfrac{h}{2} F_1\right) \end{array} \right. \\
& x_{i+1} = x_i + hF_2
\end{array}
$$

4 Método de Euler mejorado. Repetir la Práctica 2 para el *Método de Euler mejorado* (ficheros `eulermej.m` y `testeulermej.m`).

$$
\begin{array}{ll}
\underline{\text{Paso 1}} & x_0 \simeq \xi_0 \\
\underline{\text{Paso 2}} & \text{Para } i = 0, 1, \ldots, N-1 \\
& \left\{ \begin{array}{l} F_1 = f\left(t_i, x_i\right) \\ F_2 = f\left(t_{i+1}, x_i + hF_1\right) \end{array} \right. \\
& x_{i+1} = x_i + \dfrac{h}{2}\left(F_1 + F_2\right)
\end{array}
$$

5 Un Método de Runge–Kutta de orden 3. Repetir la Práctica 2 para el *Método de Runge–Kutta* de orden 3 (ficheros `rk3.m` y `testrk3.m`) dado por

$$
\begin{array}{ll}
\underline{\text{Paso 1}}: & x_0 \simeq \xi_0 \\
\underline{\text{Paso 2}}: & \text{Para } i = 0, 1, \ldots, N-1 \\
& \left\{ \begin{array}{l} F_1 = f(t_i, x_i) \\ F_2 = f\left(t_i + \dfrac{h}{2}, x_i + \dfrac{h}{2} F_1\right) \\ F_3 = f\left(t_i + \dfrac{3h}{4}, x_i + \dfrac{3h}{4} F_2\right) \end{array} \right. \\
& x_{i+1} = x_i + \dfrac{h}{9}\left(2F_1 + 3F_2 + 4F_3\right)
\end{array}
$$

6 Método de Runge–Kutta de orden 4 (clásico). Repetir la Práctica 2 para el siguiente *Método de Runge–Kutta* de orden 4 (ficheros `rk4.m` y `testrk4.m`).

$$
\begin{array}{ll}
\underline{\text{Paso 1}} & x_0 \simeq \xi_0 \\
\underline{\text{Paso 2}} & \text{Para } i = 0, 1, \ldots, N-1 \\
& \left\{ \begin{array}{l} F_1 = f(t_i, x_i) \\ F_2 = f\left(t_i + \dfrac{h}{2}, x_i + \dfrac{h}{2} F_1\right) \\ F_3 = f\left(t_i + \dfrac{h}{2}, x_i + \dfrac{h}{2} F_2\right) \\ F_4 = f\left(t_i + h, x_i + hF_3\right) \end{array} \right. \\
& x_{i+1} = x_i + \dfrac{h}{6}\left(F_1 + 2F_2 + 2F_3 + F_4\right)
\end{array}
$$

7 Crear un fichero de tipo función, de nombre `testmet.m`, que añada como variable de entrada un método ya implementado, lo ejecute y lo dibuje.

8 Crear un fichero de tipo función, de nombre `comp2met.m`, que tomando como dato dos métodos ya implementados, los ejecute y dibuje, siguiendo el código establecido de colores, de la siguiente forma:

a) Si la ecuación diferencial es escalar: la gráfica de la solución por el primero de los métodos; tras una pausa y en otra ventana, la diferencia entre las soluciones obtenidas por ambos métodos y, tras una pausa y en otra ventana, la norma (infinito) de la diferencia entre las soluciones obtenidas por ambos métodos, utilizando el comando `legend` para mostrar su valor.

b) Si la ecuación diferencial es en $\mathbb{R}^2$ o en $\mathbb{R}^3$: la gráfica de todas las componentes en la misma ventana por el primero de los métodos; tras una pausa, dibujar en otra ventana la diferencia entre las componentes respectivas de las soluciones obtenidas por ambos métodos; tras una pausa, dibujar la trayectoria de la solución por el primer método y, tras una pausa y en otra ventana, la norma (infinito) de la diferencia entre las soluciones obtenidas por ambos métodos, utilizando el comando `legend` para mostrar su valor.

Indicación: Utilizar dos datos adicionales `met1` y `met2`, respectivamente, para indicar cuáles son los métodos que se están empleando y considerar, también, las sentencias utilizadas en el fichero `graficas.m`

9 Crear un fichero de tipo función, de nombre `comp2ode45.m`, que tomando como dato uno de los métodos, lo compare con la función `ode45` de `MATLAB`, siguiendo las indicaciones de la Práctica 8.

10 En el caso de que se conozca la solución exacta del problema de valor inicial, crear un fichero de tipo función, de nombre `comp2solexac.m` que, utilizando como entradas uno de los métodos implementados y la solución exacta, lo ejecute, evalúe la solución exacta de la ecuación en los mismos nodos y dibuje, siguiendo el código establecido de colores, de la siguiente forma:

a) Si la ecuación diferencial es escalar: la gráfica de la solución exacta; tras una pausa y en otra ventana, la diferencia entre las soluciones exacta y aproximada y, tras una pausa y en otra ventana, la norma (infinito) de la diferencia entre la solución exacta y la aproximada, utilizando el comando `legend` para mostrar su valor.

b) Si la ecuación diferencial es en $\mathbb{R}^2$ o en $\mathbb{R}^3$: la gráfica de todas las componentes de la solución exacta en la misma ventana; tras una pausa, dibujar en otra ventana la diferencia entre las componentes respectivas de las soluciones exacta y aproximada; tras una pausa, dibujar la trayectoria de la solución exacta y, tras una pausa y en otra ventana, la norma (infinito) de la diferencia entre las soluciones exacta y aproximada, utilizando el comando `legend` para determinar su valor.

11 Utilizar las Prácticas 8 y 10 para comparar entre diversos métodos numéricos, así como con la solución exacta, cuando se les aplica a la resolución de los siguientes problemas de valor inicial:

a) $$\begin{cases} x'(t) = -0'1x(t) + 2y(t),\ t \in [0,10] \\ y'(t) = -2x(t) - 0'1y(t),\ t \in [0,10] \\ x(0) = 0 \\ y(0) = 1. \end{cases}$$ <u>Solución exacta</u>: $x(t) = e^{-0'1t}\operatorname{sen}(2t),\ y(t) = e^{-0'1t}\cos(2t).$

b) $$\begin{cases} x''(t) + 2x(t) = \cos(3t),\ t \in [0,10] \\ x(0) = 1 \\ x'(0) = 0. \end{cases}$$ <u>Solución exacta</u>: $x(t) = \frac{8}{7}\cos(\sqrt{2}t) - \frac{1}{7}\cos(3t).$

Notación: Se llamarán, respectivamente, `func1.m` y `func2.m` los ficheros que contienen las funciones de cada ecuación diferencial y `solexac1.m` y `solexac2.m` los ficheros que contienen las soluciones respectivas.

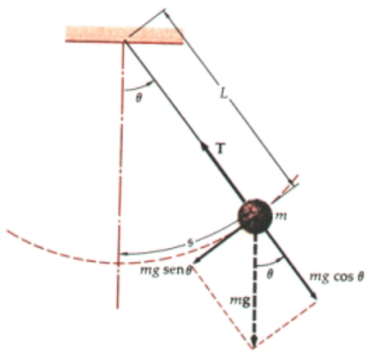
12 **Ecuación del péndulo**.

El desplazamiento angular $\theta(t)$ de un péndulo de longitud L, con respecto a la vertical, es solución del problema de valor inicial

$$\begin{cases} mL\theta''(t) + 2L\beta\theta'(t) + mg\operatorname{sen}(\theta(t)) = F,\ t \in [0,T] \\ \theta(0) = \theta_0 \\ \theta'(0) = w_0. \end{cases}$$

Se consideran los valores $m = 1$, $g = 9'8\,\mathrm{m/s^2}$ y $T = 10$.

- Suponiendo que $F = 0$ y $L = 1\,\mathrm{m}$ explorar, utilizando el *Método de Runge–Kutta* clásico de orden 4, el comportamiento de las soluciones para diversas elecciones de los datos iniciales. Considerar, en primer lugar, el valor $\beta = 0$ y, luego, los valores $\beta = 0'25$ y $\beta = 1'5$.



Movimiento del péndulo

- Para cada uno de los datos iniciales escogidos, comparar los valores encontrados con los obtenidos al resolver el *problema linealizado*
$$\begin{cases} mL\theta''(t) + 2L\beta\theta'(t) + mg\theta(t) = 0, \; t \in [0,T] \\ \theta(0) = \theta_0 \\ \theta'(0) = w_0. \end{cases}$$
Nótese que la similitud de las soluciones y trayectorias únicamente se da cuando los datos iniciales son pequeños. ¿Qué ocurre si se parte con el péndulo en posición vertical, es decir, si $\theta_0 = \pi$ y $w_0 = 0$?

- Considérese ahora el valor $\beta = 0'5$. Comprobar que si $F = 1$ entonces
$$\left(\widehat{\theta}_0, \widehat{w}_0\right) = \left(\operatorname{arcsen}\left(\frac{1}{g}\right), 0\right)$$
es un punto de equilibrio del péndulo. Tomar datos iniciales próximos a este punto de equilibrio, variar F según los valores $F = 0'9$, $F = 1$ y $F = 1'1$ y observar el cambio de comportamiento.

Notación: Los ficheros que contienen las funciones de la ecuación del péndulo no lineal y lineal se llamarán, respectivamente, `funcpendulo.m` y `funcpendulolin.m`.

13 Para cada uno de los siguientes *problemas autónomos* explorar, mediante el *Método de Runge–Kutta* clásico de orden 4, el comportamiento de las soluciones en el intervalo $[0, 100]$, para diversas elecciones de los datos iniciales.

a) **Sistemas depredador–presa 1 (Lotka–Volterra)**: Si $x(t)$ e $y(t)$ denotan la población de presas y depredadores, respectivamente, en el instante t, el modelo matemático más simple que gobierna su evolución, es
$$\begin{cases} x'(t) = ax(t) - bx(t)y(t) \\ y'(t) = -cy(t) + dx(t)y(t) \end{cases}$$
donde los coeficientes a, b, c, d son no negativos. Tomar los casos $a = b = c = d = 1$ y $a = 3$, $b = 0'2$, $c = 0'6$, $d = 5$ y datos iniciales $x(0), y(0) > 0$.

b) **Sistema depredador–presa 2**: Un modelo de depredador–presa más completo, en el que se tiene en cuenta la saturación de presas en ausencia de depredadores y viceversa, es
$$\begin{cases} x'(t) = ax(t) - bx(t)y(t) - ex^2(t) \\ y'(t) = -cy(t) + dx(t)y(t) - fy^2(t). \end{cases}$$
donde los coeficientes a, b, c, d, e, f son no negativos. Tomar los valores $a = b = c = d = 1$, $e = 0'4$ y $f = 0'02$ y datos iniciales $x(0), y(0) > 0$.

c) **Ecuación de Van der Pol**: Esta ecuación describe el comportamiento de circuitos electrónicos no lineales
$$x''(t) + \alpha(x^2(t) - \beta)x'(t) + x(t) = 0.$$
Considerar, en primer lugar, $\alpha = 1$ y tomar datos iniciales próximos al origen para $\beta = -0'2$, $\beta = 0$ y $\beta = 0'2$. Fijar, a continuación, $\beta = 1$ y aumentar el valor de α desde $\alpha = 1$ hasta $\alpha = 8$.

d) **Ecuación de Duffing**: Esta ecuación describe el movimiento de una varilla bajo efectos magnéticos
$$x''(t) + \alpha x'(t) + x^3(t) - x(t) = 0.$$
Tomar, en primer lugar, $\alpha = 0$ y explorar las soluciones en torno a los 3 equilibrios del problema. Tomar, a continuación, $\alpha = 1$ y estudiar el cambio en el comportamiento de las soluciones.

e) **Sistema de Lorenz**: En 1963, tratando de analizar el comportamiento impredecible del tiempo meteorológico, el meteorólogo E. N. Lorenz obtuvo el siguiente sistema de ecuaciones no lineales
$$\begin{cases} x'(t) = \sigma(y(t) - x(t)) \\ y'(t) = \rho x(t) - y(t) - x(t)z(t) \\ z'(t) = x(t)y(t) - \beta z(t). \end{cases}$$
Tomar $\sigma = 10$ y $\beta = \frac{8}{3}$ e ir 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 $\rho = 100'5$, $N = 10000$ y el dato inicial $(0, 5, 75)$ para observar una solución periódica. Manteniendo el dato inicial anterior, mover ρ entre $99'524$ y $100'795$ (por ejemplo $\rho = 99'65$) y observar el cambio de dinámica.

Notación: Se llamarán `funcdeppresa1.m`, `funcdeppresa2.m`, `funcvanderpol.m`, `funcduffing.m` y `funclorenz.m` los ficheros que contienen las funciones de las respectivas ecuaciones diferenciales.

14 Oscilador armónico forzado. Se considera el siguiente problema de valor inicial gobernado por la ecuación del *oscilador armónico forzado*

$$\begin{cases} x''(t) + 2\beta x'(t) + a^2 x(t) = A\cos(wt) \\ x(0) = x'(0) = 0. \end{cases}$$

Explorar, utilizando el *Método de Runge–Kutta* clásico de orden 4, el comportamiento de las soluciones en el intervalo $[0, 10]$ para las siguientes elecciones de los parámetros:

a) <u>Caso sin rozamiento</u>: $\beta = 0$. Tomar los valores $A = 1$, $a = 10$ y $w = 12$. Ir disminuyendo el valor de w hasta que $w = a$. Continuar disminuyendo a un poco más. ¿Qué ocurre cuando $w = a$?

b) <u>Caso con rozamiento</u>: $\beta > 0$. Tomar los valores $\beta = 1$, $A = 1$, $a = 10$. Comenzar con $w = 8$ e ir aumentando los valores hasta $w = 12$. ¿Qué se observa en las soluciones a medida que transcurre el tiempo? ¿Qué se observa cuando $w \simeq 9'8995$? Repetir lo anterior tomando $\beta = 15$. ¿Qué diferencia hay en las soluciones?

Notación: El fichero que contiene la función de esta ecuación diferencial se llamará `funcoscilador.m`