– Práctica 1 –

Métodos unipaso para problemas de valor inicial

1. **El método de Euler**. En este apartado vamos a realizar una implementación del método de Euler en Python para resolver el problema de valor inicial

$$(\mathrm{P}) \qquad \begin{cases} y' = \dfrac{1}{2}(t^2 - y), & t \in [0,10], \\ y(0) = 1, \end{cases}$$

cuya solución exacta es la función $y(t) = t^2 - 4t + 8 - 7e^{-t/2}$.

```python
# Resolucion del problema de valor inicial
# y'=f(t,y), y(t0)=y0,
# mediante el metodo de Euler.

from pylab import *
from time import perf_counter

def f(t, y):
    """Funcion que define la ecuacion diferencial"""
    return 0.5*(t**2 - y)

def exacta(t):
    """Solucion exacta del problema de valor inicial"""
    return t**2 - 4*t + 8 - 7.*exp(-0.5*t)

def euler(a, b, fun, N, y0):
    """Implementacion del metodo de Euler en el intervalo [a, b]
    usando N particiones y condicion inicial y0"""

    h = (b-a)/N # paso de malla
    t = zeros(N+1) # inicializacion del vector de nodos
    y = zeros(N+1) # inicializacion del vector de resultados
    t[0] = a # nodo inicial
    y[0] = y0 # valor inicial

    # Metodo de Euler
    for k in range(N):
        y[k+1] = y[k]+h*fun(t[k], y[k])
        t[k+1] = t[k]+h

    return (t, y)

# Datos del problema
a = 0. # extremo inferior del intervalo
b = 10. # extremo superior del intervalo
N = 20 # numero de particiones
y0 = 1. # condicion inicial

tini = perf_counter()

(t, y) = euler(a, b, f, N, y0) # llamada al metodo de Euler
```

```
tfin=perf_counter()

ye = exacta(t) # calculo de la solucion exacta

# Dibujamos las soluciones
plot(t, y, '-*') # dibuja la solucion aproximada
plot(t, ye, 'k') # dibuja la solucion exacta
xlabel('t')
ylabel('y')
legend(['Euler', 'exacta'])
grid(True)

# Calculo del error cometido
error = max(abs(y-ye))

# Resultados
print('-----')
print('Tiempo CPU: ' + str(tfin-tini))
print('Error: ' + str(error))
print('Paso de malla: ' + str((b-a)/N))
print('-----')
```

Usando el programa anterior, se pide:

a) Compruebe que, para el problema considerado, el orden del método de Euler es uno. Para ello, realice una serie de ejecuciones del programa con $N = 10, 20, 40, 80, 160$. Obtenga en cada caso el error e_N que se comete y calcule los cocientes e_N/e_{2N}, N =10, 20, 40, 80. Dibuje en una gráfica las aproximaciones obtenidas con los distintos valores de N junto con la solución exacta.

b) En un depósito de 20 l, que inicialmente está lleno de agua pura, empieza a entrar, desde el instante $t = 0$, agua con una concentración de 3 grs/l de sal a razón de 2 l por segundo. Al mismo tiempo, empieza a salir agua con la misma velocidad. En clase se vio que la cantidad de sal $S(t)$ que hay en el depósito en el instante t sigue aproximadamente la siguiente ecuación diferencial:

$$\text{(Q)} \qquad \begin{cases} S' = 6 - \dfrac{S}{10}, \\ S(0) = 0. \end{cases}$$

Se vio también que la solución exacta es $S(t) = 60\left(1 - e^{-t/10.}\right)$. Use el programa del apartado anterior para resolver el problema con el método de Euler en el intervalo $[0, 20]$. Repita el apartado anterior para este problema.

c) Suponga que, en el problema anterior, a partir del instante $t = 0$ en vez de salir agua a razón de 2 l/seg lo hace con una velocidad de 3 l/seg, permaneciendo iguales todos los demás datos del problema. Modifique la ecuación para tener en cuenta esta variación y aplique el método de Euler con $N = 2000$ a la ecuación resultante en el máximo intervalo de tiempo en el que tenga sentido la ecuación. Cuál es, aproximadamente, el máximo de la cantidad de sal en el depósito? En qué instante de tiempo se alcanza?

2. Tomando como modelo el programa anterior, escriba la implementación en Python de los métodos de Taylor de orden 2 y 3 para el problema de Cauchy (P). Compruebe el orden repitiendo el apartado (a) del ejercicio anterior.

3. Tomando como modelo el programa anterior, escriba la implementación en Python de los métodos de Heun, punto medio y RK4. Repita los tres apartados del Ejercicio 1 usando estos métodos.

4. **Modelo depredador/presa de Lotka-Volterra.**

   Escriba implementaciones en Python de los métodos de Euler, Heun, punto medio y RK4 para sistemas. Como problema a resolver se considera el modelo depredador/presa de Lotka-Volterra:

$$\begin{cases} \dfrac{dx}{dt} = 0.25x - 0.01xy, \\ \dfrac{dy}{dt} = -y + 0.01xy, \\ x(0) = 80, \quad y(0) = 30 \end{cases}$$

   Resuelva el problema en el intervalo $[0, 20]$ con $N = 20, 40, 80, 160, 320, 640$ usando los diferentes métodos. Compare en una misma gráfica las trayectorias obtenidas con un mismo método usando distintos pasos de tiempo.

   Indicación: Escriba el sistema en forma vectorial y usar la clase `array` para trabajar con vectores y matrices.

5. Se considera el problema de Cauchy:

$$\begin{cases} x'' + 20x' + 101x = 0, \\ x(0) = 1, \quad x'(0) = -10. \end{cases}$$

   cuya solución es $x(t) = e^{-10t}\cos(t)$. Reescriba la ecuación como un sistema de dos ecuaciones de primer orden y aplique los métodos del ejercicio anterior en el intervalo $[0, 7]$ con $N = 20, 40, 80, 160, 320, 640$. Compare en una misma gráfica la solución exacta $x(t)$ con las aproximaciones obtenidas con un mismo método usando distintos pasos de tiempo. Se observan resultados anómalos?

6. **Modelo SIR** Para simular la evolución de una epidemia, se consideran las siguientes tres clases disjuntas de individuos:

   - S: individuos susceptibles de ser contagiados.
   - I: individuos infectados.
   - R: individuos que no están en ninguna de las dos clases anteriores, ya sea por aislamiento, recuperación (y, consecuentemente, inmunidad), o por muerte.

   Sean $x(t)$, $y(t)$ y $z(t)$ el número de individuos (medido en millares) que están en el instante t (medido en días) en las clases S, I y R, respectivamente. Se considera el siguiente modelo para simular la evolución de la epidemia:

$$\begin{aligned} \frac{dx}{dt} &= -k_1xy - k_3x, \\ \frac{dy}{dt} &= k_1xy - k_2y, \\ \frac{dz}{dt} &= k_3x + k_2y. \end{aligned}$$

   siendo k_1 la tasa de contagios, k_2 la tasa de eliminación de infectados por las diversas causas posibles y k_3 la tasa de mortalidad de los individuos sanos.

   a) Supongamos que el número total de individuos en el instante $t = 0$ es $N = 100$ y que en el instante inicial hay 1 millar contagiados y 99 millares de susceptibles. Supongamos además que $k_1 = 2 \times 10^{-3}$, $k_2 = 5 \cdot 10^{-3}$ y $k_3 = 10^{-5}$. Aproxime la solución del sistema de ecuaciones diferenciales en el intervalo de tiempo $[0, 60]$ usando el método RK4 con paso $h = 0.05$. Represente en una sola gráfica la evolución de $x(t)$, $y(t)$, $z(t)$. Comente brevemente los resultados que parecen deducirse sobre la evolución de la epidemia.

b) Repita el apartado anterior con todos los datos iguales, pero con $k_2 = 5 \cdot 10^{-2}$: compare las diferencias.

c) Modifique el modelo para que contemple la posibilidad de que individuos infectados puedan pasar a ser de nuevo susceptibles por pérdida de inmunidad, siendo la velocidad de este proceso proporcional al número de infectados: $k_4\, y$. Simule la evolución con los datos del apartado anterior tomando $k_4 = 0.5\, k_2$: compare los resultados.

7. **Modelo de cohete propulsado.** El sigiuiente problema de Cauchy modela el movimiento de un cohete que se lanza verticalmente:

$$\begin{cases} \dfrac{dz}{dt} = v, \\ \dfrac{dv}{dt} = -g + \dfrac{T}{M + m_f} - \dfrac{C}{M + m_f} v|v| + \alpha \dfrac{T}{M + m_f} v, \\ \dfrac{dm_f}{dt} = -\alpha T, \\ z(0) = 0, \\ v(0) = v_0, \\ m_f(0) = m_{f,0}, \end{cases}$$

siendo $z(t)$ la altura del centro de gravedad del cohete en el instante t (se toma como altura 0 la posición inicial); $v(t)$ la velocidad del cohete; $T(t)$ la fuerza ejercida por el motor; $m_f(t)$ la masa de combustible; M la masa del cohete con el depósito de combustible vacío; C el coeficiente de fricción con el aire; v_0 la velocidad de lanzamient y $m_{f,0}$ la masa inicial de combustible. La velocidad a la que se reduce la masa de combustible se supone proporcional a la fuerza que ejerce el motor, siendo α la constante de proporcionalidad. Supondremos los siguientes valores:

$$g = 9.81, \quad M = 7.5, \quad \alpha = 0.02.$$

Supondremos además que la fuerza que ejerce el cohete es una función de la forma:

$$T(t) = \begin{cases} T_0 & \text{si } m_f > 0, \\ 0 & \text{si } m_f = 0, \end{cases}$$

siendo T_0 un valor constante.

a) Escriba un programa Python que permita resolver el problema planteado usando el método RK4 con paso h **mientras se cumpla** $z \geqslant 0$.

b) Aplique el programa para resolver el problema con condición inicial $v_0 = 50$, $m_{f,0} = 7.5$, paso $h = 0.05$ y las siguientes elecciones de T_0 y C: (i) $T_0 = C = 0$; (ii) $T = 0$, $C = 0.02$; (iii) $T_0 = 50$, $C = 0.02$.

c) Calcule la altura máxima alcanzada por el cohete en cada caso y estime el tiempo de caída. Compare las gráficas de la altura del cohete frente al tiempo obtenida en los tres casos. En el caso (c) dibuje también la gráfica de la masa de combustible frente al tiempo y calcule en qué momento se acaba el combustible.