

Hoja 5. Problemas rígidos y métodos adaptativos

1 Dado $\mu > 0$, consideremos el problema escalar de valor inicial

$$\begin{cases} x'(t) = -\mu(x(t) - \cos(t)), & t \in [0, 1] \\ x(0) = 0, \end{cases} \quad (1)$$

cuya solución exacta es

$$x(t) = \frac{\mu^2}{1 + \mu^2} \left(\cos(t) - e^{-\mu t} + \frac{1}{\mu} \sin(t) \right).$$

a) Crear un fichero de tipo script llamado `estabs1.m` en el que se:

- I) Aplique el Método de Euler (explícito) a la resolución del problema (1) cuando $\mu = 50$ y se divide el intervalo $[0, 1]$ en $N = 24$ y $N = 26$ subintervalos, respectivamente.
- II) Representen gráficamente ambos resultados, junto con la solución exacta, en una misma ventana. ¿Qué se aprecia?

b) Repetir el apartado a) para otros valores de N (mayores y menores que 25) y analizar los comportamientos que se obtienen.

c) Crear un fichero de tipo script llamado `estabs2.m` en el que se:

- I) Aplique el Método de Euler implícito a la resolución del problema (1) cuando $\mu = 50$ para el valor $N = 24$.
- II) Representen gráficamente el resultado anterior, junto con la solución exacta, en una misma ventana. ¿Qué ocurre ahora?

d) Repetir el apartado c) para otros valores de N (mayores y menores que 25) y analizar los comportamientos que se obtienen.

e) ¿Qué puede decirse acerca de los Métodos de Euler explícito e implícito cuando se aplican a la resolución del problema de valor inicial (1)?

2 Se considera el siguiente problema de valor inicial gobernado por la ecuación de Van der Pol:

$$\begin{cases} x''(t) + \alpha(x^2(t) - \beta)x'(t) + x(t) = 0, & t \in [0, T] \\ x(0) = \xi_0, \\ x'(0) = \eta_0 \end{cases} \quad (2)$$

con $\beta = 1$ y $\xi_0 = \eta_0 = 0.1$.

a) Crear un fichero de tipo script llamado `estabsvanderpol.m` en el que se:

- I) Aplique la función `ode45` de MATLAB para resolver el problema (2) cuando $\alpha = 0.1$ y $T = 150$ con una tolerancia $TOL = 10^{-5}$. **Indicación:** la sintaxis es

```
opciones=odeset('Stats','off','AbsTol',1.e-5,'RelTol',1.e-5);  
[t1,x1]=ode45(f,intervalo,x0,opciones)
```

siendo, como en las prácticas anteriores, `f`, `intervalo` y `x0`, variables que representan el nombre de la función, el intervalo y los datos iniciales del problema de valor inicial que se pretende resolver.

- II) Represente gráficamente en una ventana la solución x_1 aproximada obtenida.
- III) Aplique la función `ode15s` de MATLAB (específica para problemas rígidos) para resolver el problema (2) cuando $\alpha = 0.1$ y $T = 150$ con una tolerancia $TOL = 10^{-5}$. **Indicación:** la sintaxis es

```
opciones=odeset('Stats','off','AbsTol',1.e-5,'RelTol',1.e-5);  
[t2,x2]=ode15s(f,intervalo,x0,opciones)
```

- IV) Represente gráficamente en una nueva ventana la solución x_2 aproximada obtenida.

- v) Represente gráficamente en otra ventana los pasos considerados en cada uno de los métodos `ode45` y `ode15s` y determinar, mediante el comando `legend`, el número de pasos considerado en cada método. Para ello, se dibujarán los conjuntos de puntos

$$\{(i, t1(i+1) - t1(i)) : i \in \{1, 2, \dots, \text{length}(t1) - 1\}\}$$

y

$$\{(i, t2(i+1) - t2(i)) : i \in \{1, 2, \dots, \text{length}(t2) - 1\}\}$$

(se puede usar el comando `diff` de MATLAB, por comodidad).

- b) Repetir el apartado a) cuando se consideran $\alpha = 30$ y $T = 250$.
- c) A la vista de los resultados obtenidos, ¿cuál de las dos rutinas (`ode45` y `ode15s`) resulta más eficiente para resolver el problema de valor inicial (2)?

3 (Método de Runge–Kutta–Fehlberg) Consideremos el problema de valor inicial

$$\begin{cases} x'(t) = f(t, x(t)), & t \in [t_0, T] \\ x(t_0) = \xi_0, \end{cases}$$

donde $\xi_0 \in \mathbb{R}^n$ y $t_0, T \in \mathbb{R}$ con $t_0 < T$ donde, a diferencia de los métodos implementados hasta ahora, el paso h en cada subintervalo de la partición puede ir cambiando. Es decir, introducimos la posibilidad de modificar (reducir o aumentar) h en cada paso con el objetivo de minimizar el error que se vaya cometiendo. De este modo se obtiene una secuencia de pasos $\{h_1, h_2, \dots, h_N\}$ donde

$$h_{i+1} = t_{i+1} - t_i, \quad i = 0, 1, \dots, N - 1.$$

El *Método de Runge–Kutta–Fehlberg* es uno de estos métodos de paso adaptativo en el que se usan dos métodos de Runge–Kutta de órdenes 4 y 5, de forma que, para ir de (t_i, x_i) a (t_{i+1}, x_{i+1}) :

- 1) x_{i+1} se calcula con un método de Runge–Kutta de orden 4 con 5 evaluaciones $\{F_1, F_2, F_3, F_4, F_5\}$ dado por

$$x_{i+1} = x_i + h \left(\frac{25}{216} F_1 + \frac{1408}{2565} F_3 + \frac{2197}{4104} F_4 - \frac{1}{5} F_5 \right)$$

- 2) Para estimar el error y decidir si se amplía o se reduce el paso h , se utiliza un método de Runge–Kutta de orden 5 con 6 evaluaciones $\{F_1, F_2, F_3, F_4, F_5, F_6\}$, siendo comunes en ambos las 5 primeras evaluaciones. Viene dado por

$$y_{i+1} = x_i + h \left(\frac{16}{135} F_1 + \frac{6656}{12825} F_3 + \frac{28561}{56430} F_4 - \frac{9}{50} F_5 + \frac{2}{55} F_6 \right),$$

siendo

$$\begin{cases} F_1 = f(t_i, x_i) \\ F_2 = f\left(t_i + \frac{h}{4}, x_i + \frac{h}{4} F_1\right) \\ F_3 = f\left(t_i + \frac{3h}{8}, x_i + \frac{3h}{32} F_1 + \frac{9h}{32} F_2\right) \\ F_4 = f\left(t_i + \frac{12h}{13}, x_i + \frac{1932h}{2197} F_1 - \frac{7200h}{2197} F_2 + \frac{7296h}{2197} F_3\right) \\ F_5 = f\left(t_i + h, x_i + \frac{439h}{216} F_1 - 8h F_2 + \frac{3680h}{513} F_3 - \frac{845h}{4104} F_4\right) \\ F_6 = f\left(t_i + \frac{h}{2}, x_i - \frac{8h}{27} F_1 + 2h F_2 - \frac{3544h}{2565} F_3 + \frac{1859h}{4104} F_4 - \frac{11h}{40} F_5\right). \end{cases}$$

La estimación del error que se obtiene, en cada iteración, viene dada por

$$\text{ERR}(h) = h \left(\frac{1}{360} F_1 - \frac{128}{4275} F_3 - \frac{2197}{75240} F_4 + \frac{1}{50} F_5 + \frac{2}{55} F_6 \right).$$

Nótese que no es necesario calcular las iteraciones y_{i+1} del método, sino que se utiliza la fórmula de la estimación del error directamente. En cada iteración sólo se necesitan 6 evaluaciones de la función f .

Para más detalles sobre el estudio de este método véase, por ejemplo, el libro [Mathews–Fink].

Utilizar el programa que se adjunta `rkf.m`, donde viene implementado este método, para resolver los problemas de valor inicial de la Hoja 2 y comparar estos métodos con los métodos monopaso y multipaso ya implementados.