

Prácticas de Matlab

Diagrama de eficiencia con métodos monopaso explícitos

Hoja 3

Nombre:

Apellido:

DNI:

Table of Contents

Prácticas de Matlab.....	1
Diagrama de eficiencia con métodos monopaso explícitos.....	1
Hoja 3.....	1
1. Diagrama de eficiencia.....	1
Práctica 1 (El método de Euler explícito)	1
Práctica 2 (Euler mejorado)	2
Práctica 3 (Euler modificado).....	2
Práctica 4 (Runge-Kutta 4).....	2
Apéndice código: funciones de Euler, Euler modificado, Euler mejorado y Runge-Kutta 4, para calcular y pintar el diagrama de eficiencia y el orden.....	4

1. Diagrama de eficiencia

Práctica 1 (El método de Euler explícito)

Consideramos el siguiente problema lineal

$$y'(t) = Ay(t) + B(t) \quad \text{para } 0 \leq t \leq 10, \quad y(0) = (2, 3)^T,$$

$$A = \begin{pmatrix} -2 & 1 \\ 1 & -2 \end{pmatrix} \quad B(t) = \begin{pmatrix} 2 \sin(t) \\ 2(\cos(t) - \sin(t)) \end{pmatrix}$$

La solución exacta es:

$$y = 2e^{-t} \begin{pmatrix} 1 \\ 1 \end{pmatrix} + \begin{pmatrix} \sin(t) \\ \cos(t) \end{pmatrix}$$

Se pide lo siguiente

1. Resuelve este sistema mediante el método de *Euler explícito*, almacena el máximo en valor absoluto de la diferencia entre la solución exacta y la solución numérica calculada. **Indicación:** piensa qué norma vas a usar, dependiendo del tipo de salida (vector columna o vector fila) que haya producido tu algoritmo. Efectúa este cálculo para varias elecciones

2. del paso h_j con $j = 0, \dots, 7$ siendo $h_0 = 0.1$, $h_j = \frac{h_0}{2^j}$. Almacena los diferentes valores de h_i en un vector h_{vect} .
3. del número de puntos N siendo $N_0 = 100$, $N_i = 2^i N_0$. Almacena los diferentes valores de N_i en un vector N_{vect} .
4. número de las evaluaciones totales Ev_i que realiza cada algoritmo para cada valor de h_i . Almacena los valores en un vector Ev_{vect} .
5. Almacena los distintos errores en un vector de nombre **error_euler**

Además

- Dibuja, en una misma ventana, en escala logarítmica, el error almacenado en el apartado anterior frente al paso h , h_{vect} * Indicación:* usa el comando `loglog` en vez del comando `plot`.
- Repite en otra figura lo mismo pero dibujando el error frente al vector N_{vect}
- Calcula la pendiente de la recta.
- Repite en otra figura lo mismo pero dibujando el error frente al vector Ev_{vect} .
- Interpreta el resultado.

Práctica 2 (Euler mejorado)

Repite el apartado anterior con el método de Euler mejorado

Práctica 3 (Euler modificado)

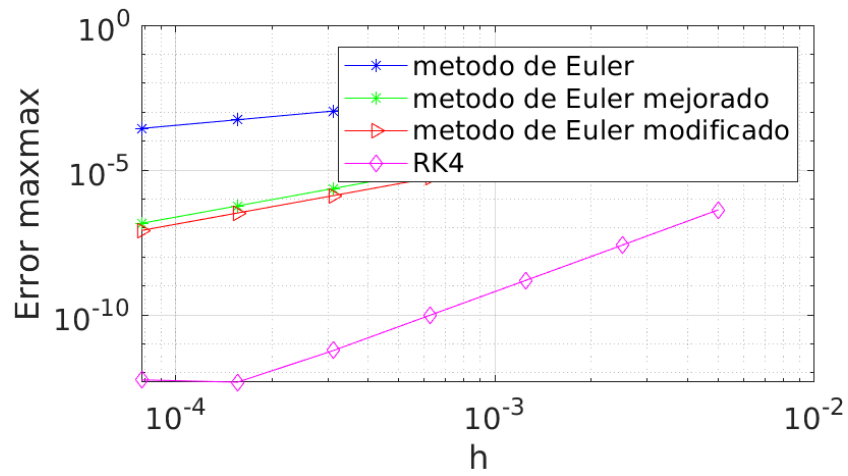
Repite el apartado anterior con el método de Euler modificado

Práctica 4 (Runge-Kutta 4)

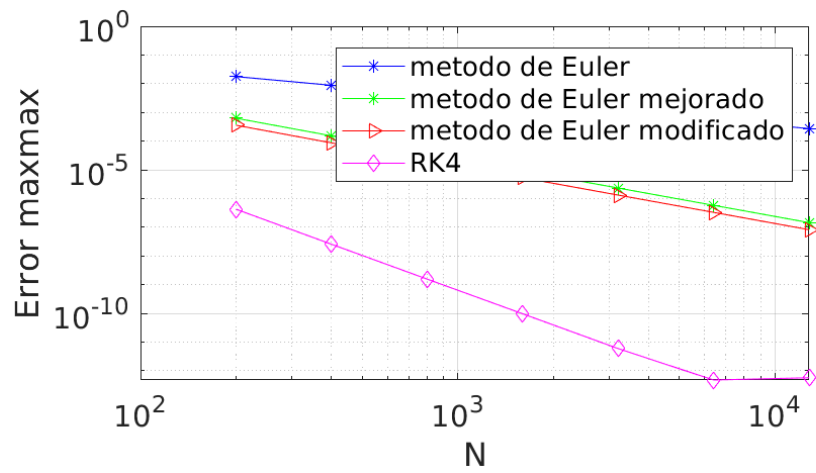
Repite el apartado anterior con el método de Runge-Kutta de orden 4.

OJO: pon siempre el diagrama de eficiencia de Euler, Euler modificado, Euler mejorado y Runge Kutta 4 en una gráfica:

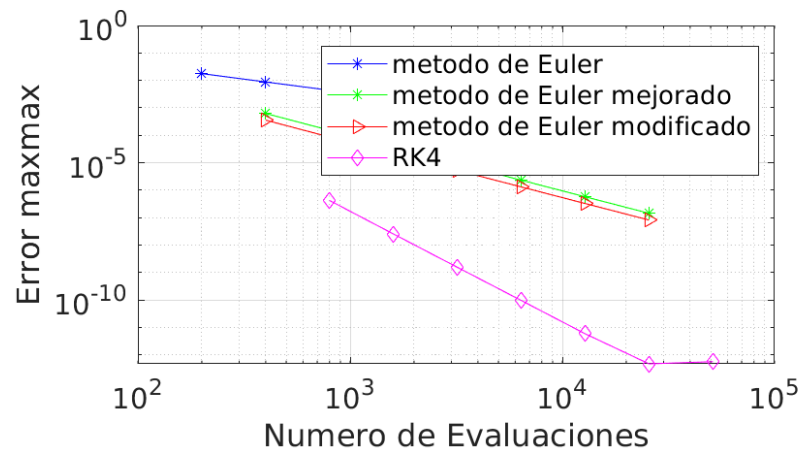
Error maximo vs h
Problema no stiff
intv=[0 10] y0=[2 3]
 $N_{\text{int}}=200$ M=7



Error maximo vs N
Problema no stiff
intv=[0 10] y0=[2 3]
 $N_{\text{int}}=200$ M=7



Error maximo vs Ev
Problema no stiff
intv=[0 10] y0=[2 3]
 $N_{int}=200$ $M=7$



Apéndice código: funciones de Euler, Euler modificado, Euler mejorado y Runge-Kutta 4, para calcular y pintar el diagrama de eficiencia y el orden