

Método de Euler y Runge-Kutta explícitos

Guión:

En estas notas implementaremos en MATLAB el método de Euler y métodos Runge-Kutta explícitos para resolver correctamente el problema de valor inicial (PVI)

$$\begin{aligned}u'(t) &= f(t, u(t)), & t \in (t_0, T), \\u(t_0) &= u_0\end{aligned}$$

1. Método de Euler

La ecuación modificada asociada al PVI es

$$u_{n+1} = u_n + hf(t_n, u_n), \quad n = 0, \dots, N-1,$$

con

$$h = \frac{T - t_0}{N}.$$

En primer lugar, vamos a implementar la función **euler.m**:

```
function [u,t]=euler(f,N,t0,T,u0)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Esta función resuelve el problema de valor inicial
%      u'=f(t,u)
%      u(t0)=u0
% utilizando el método de Euler
%
%      [u,t]=euler(N,t0,T,u0)
%
% Variables de Entrada:
%
%      f: vector columna. función que rige el sistema de EDO,
%          tiene dos argumentos f(t,u) donde t es escalar
%          y u vector columna.
%      N: número de pasos en los que dividimos el intervalo de
%          integración
%      t0: tiempo inicial
%      T: tiempo final
%      u0: vector columna. Dato inicial
%
% Variables de Salida:
%      u: matriz de length(u0) x N que contiene la solución
%      t: vector de tiempos
%
```

2. Método de Runge-Kutta explícito

Queremos ahora construir una función que nos permita resolver un método Runge-Kutta explícito general de s etapas:

$$u_{n+1} = u_n + h \sum_{i=1}^s b_i k_i$$

donde

$$k_1 = f(t_n + c_1 h, u_n)$$

y para $i > 1$

$$k_i = f(t_n + c_i h, u_n + h \sum_{j=1}^{i-1} a_{ij} k_j), \quad i = 2, \dots, s.$$

1. Implementar la función **RKexplicito.m**:

```
function [u,t]=RKexplicito(f,T,t0,N,u0,b,c,A)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Esta función resuelve el problema de valor inicial
%      u'=f(t,u)
%      u(t0)=u0
% utilizando un método Runge-Kutta explícito
%
%      [u,t]=RKexplicito(f,T,t0,N,u0,b,c,A)
%
% Variables de Entrada:
%
%      f: vector columna. función que rige el sistema de EDO,
%          tiene dos argumentos f(t,u) donde t es escalar
%          y u vector columna.
%      N: número de pasos en los que dividimos el intervalo de
%          integración
%      t0: tiempo inicial
%      T: tiempo final
%      u0: vector columna, dato inicial
%      b,c,A: coeficientes del tablero de BUTCHER.
%          A: matriz cuadrada de orden s
%          c: vector columna de orden s
%      b: vector fila de orden s
%
% Variables de Salida:
%
%      u: matriz de length(u0) x length(t) que contiene la solución
%      t: vector de tiempos
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

Algunos tableros de Butcher de Runge-Kutta explícitos:

- Método de Runge (1895) de orden 2:

0	0	0
1/2	1/2	0
<hr/>		0 1

- Método de Heun de orden 2:

0		0	0
1		1	0
		1/2	1/2

- Método de Heun de orden 3:

0		0	0	0
1/3		1/3	0	0
2/3		0	2/3	0
		1/4	0	3/4

- Método de Kutta (1905) de orden 4:

0		0	0	0	0
1/2		1/2	0	0	0
1/2		0	1/2	0	0
1		0	0	1	0
		1/6	1/3	1/3	1/6

- Método de los 3/8 de orden 4:

0		0	0	0	0
1/3		1/3	0	0	0
2/3		-1/3	1	0	0
1		1	-1	1	0
		1/8	3/8	3/8	1/8

3. Ejercicios:

1. Utilizar las funciones Euler y RKexplicito con los tableros dados anteriormente para resolver el problema de valor inicial siguiente: En el intervalo $[0, 10]$,

$$\begin{cases} u' = \begin{bmatrix} -2 & 1 \\ 1 & -2 \end{bmatrix} u + \begin{bmatrix} 2 \sin(t) \\ 2(\cos(t) - \sin(t)) \end{bmatrix} \\ u(0) = \begin{bmatrix} 2 \\ 3 \end{bmatrix} \end{cases}$$

2. Representar gráficamente la solución exacta frente a la solución aproximada para varios valores de h .
3. Representar gráficamente las poligonales formadas por los valores (h, error) y calcular la media de las pendientes de cada uno de los segmentos.
4. Aportar conclusiones relacionando los resultados numéricos obtenidos con lo visto en teoría.
5. Para $t \in [0, 10]$, consideramos el problema de valor inicial:

$$\begin{cases} u' = \begin{bmatrix} -2 & 1 \\ 998 & -999 \end{bmatrix} u + \begin{bmatrix} 2 \sin(t) \\ 999(\cos(t) - \sin(t)) \end{bmatrix} \\ u(0) = \begin{bmatrix} 2 \\ 3 \end{bmatrix} \end{cases}$$

Este sistema tiene la misma solución que el ejercicio 1, sin embargo, con un número de pasos no muy grande ($N = 10000$) nos aparecen problemas. ¿Cuál es la razón.