

Tema 7

Problemas de Valor Inicial II

Dra. Paula Triguero Navarro

Máster en Ingeniería Matemática y Computación
Escuela Superior en Ingeniería y Tecnología



- 1 Introducción
- 2 Métodos explícitos: Adams-Bashforth
- 3 Métodos implícitos: Adams-Moulton
- 4 Métodos predictor-corrector
- 5 Resolución de problemas rígidos

1

Introducción

Métodos de un paso y métodos multipaso

Tema 6

Métodos de un paso: la aproximación en el punto t_{k+1} solo implica conocer la información del punto anterior t_k

- ➔ La solución en un subintervalo está basada exclusivamente en la información disponible en dicho subintervalo

Métodos de un paso y métodos multipaso

Tema 6

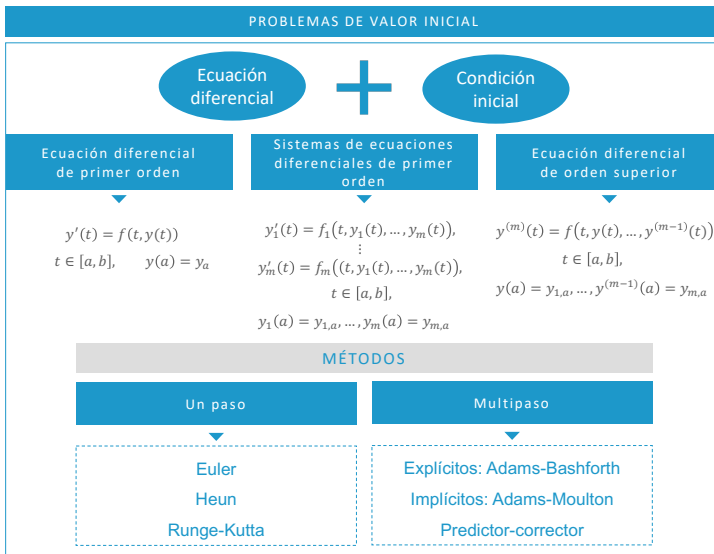
Métodos de un paso: la aproximación en el punto t_{k+1} solo implica conocer la información del punto anterior t_k

- La solución en un subintervalo está basada exclusivamente en la información disponible en dicho subintervalo

Tema 7

Métodos multipaso: utilizan la aproximación en más de un nodo previo para determinar la solución aproximada en el siguiente nodo

- En lugar de utilizar solo la información del subintervalo, utilizan de forma más precisa toda la información que se dispone hasta este nodo



Objetivos

- Comprender la diferencia entre método numérico explícito o implícito.
- Estudiar los métodos explícitos e implícitos fundamentales.
- Estudiar el diseño de métodos predictor-corrector.
- Conocer las ecuaciones rígidas y los problemas de resolución numérica relacionados con estas ecuaciones.
- Analizar y comparar la estabilidad en problemas rígidos de los métodos implícitos frente a los explícitos.

- Consideremos la expresión general de un PVI definido por una EDO de primer orden:

$$y'(t) = f(t, y(t)), \quad t \in [a, b], \quad y(a) = y_a.$$

- Integrando directamente la ecuación diferencial en cada subintervalo $[t_k, t_{k+1}]$ obtenemos

$$y(t_{k+1}) = y(t_k) + \int_{t_k}^{t_{k+1}} f(\tau, y(\tau)) d\tau.$$

- Para aproximar la integral se sustituye el integrando por un polinomio interpolador que pasa por los puntos:

$$\{(t_{k-n}, f(t_{k-n}, y_{k-n})), \dots, (t_{k-1}, f(t_{k-1}, y_{k-1})), (t_k, f(t_k, y_k))\}$$



Métodos explícitos

$$\{(t_{k-n}, f(t_{k-n}, y_{k-n})), \dots, (t_{k-1}, f(t_k, y_k)), (t_{k+1}, f(t_{k+1}, y_{k+1}))\}$$



Métodos implícitos

2

Métodos explícitos: Adams-Bashforth

- Partimos de dos puntos: $(t_{k-1}, f(t_{k-1}, y_{k-1}))$ y $(t_k, f(t_k, y_k))$
- Calculamos el único polinomio de interpolación que pasa por ellos:

$$\begin{aligned} p(\tau) &= f(t_k, y_k) + \frac{f(t_k, y_k) - f(t_{k-1}, y_{k-1})}{t_k - t_{k-1}}(\tau - t_k) \\ &= \frac{\tau - t_{k-1}}{h} f(t_k, y_k) + \frac{t_k - \tau}{h} f(t_{k-1}, y_{k-1}). \end{aligned}$$

- Aproximando la función por el polinomio de interpolación, obtenemos:

$$\begin{aligned} y_{k+1} &= y_k + \int_{t_k}^{t_{k+1}} p(\tau) d\tau \\ &= y_k + \int_{t_k}^{t_{k+1}} \left(\frac{\tau - t_{k-1}}{h} f(t_k, y_k) + \frac{t_k - \tau}{h} f(t_{k-1}, y_{k-1}) \right) d\tau. \end{aligned}$$

- Calculando la integral polinómica y desarrollando la expresión:

$$y_{k+1} = y_k + \frac{3h}{2} f(t_k, y_k) - \frac{h}{2} f(t_{k-1}, y_{k-1}).$$

Método de Adams-Bashforth de dos pasos (AB2)

$$y_{k+1} = y_k + \frac{3h}{2}f(t_k, y_k) - \frac{h}{2}f(t_{k-1}, y_{k-1})$$

- Método explícito de segundo orden con error global $\mathcal{O}(h^2)$

Método de Adams-Bashforth de dos pasos (AB2)

$$y_{k+1} = y_k + \frac{3h}{2}f(t_k, y_k) - \frac{h}{2}f(t_{k-1}, y_{k-1})$$

- Método explícito de segundo orden con error global $\mathcal{O}(h^2)$

Desarrollo AB2

- y_0 conocido
- Cálculo de y_1 :

$$k = 0 \quad \Rightarrow \quad y_1 = y_0 + \frac{h}{2} (3f(t_0, y_0) - f(t_{-1}, y_{-1})).$$

- ➔ Como (t_{-1}, y_{-1}) no existe, calculamos y_1 utilizando otro método numérico (normalmente del mismo orden):

CI:	y_0
Heun:	y_1
AB2:	y_2, y_3, \dots, y_N

Método de Adams-Bashforth de tres pasos (AB3)

- Puntos:

$$\{(t_{k-2}, f(t_{k-2}, y_{k-2})), (t_{k-1}, f(t_{k-1}, y_{k-1})), (t_k, f(t_k, y_k))\}$$

- Algoritmo:

$$y_{k+1} = y_k + \frac{h}{12} (23f(t_k, y_k) - 16f(t_{k-1}, y_{k-1}) + 5f(t_{k-2}, y_{k-2}))$$

- Método explícito con error global $\mathcal{O}(h^3)$

Desarrollo AB3

CI:	y_0
Runge-Kutta:	$y_1, y_2,$
AB3:	y_3, y_4, \dots, y_N

Método de Adams-Bashforth de cuatro pasos (AB4)

■ Puntos:

$$\{(t_{k-3}, f(t_{k-3}, y_{k-3})), (t_{k-2}, f(t_{k-2}, y_{k-2})), (t_{k-1}, f(t_{k-1}, y_{k-1})), (t_k, f(t_k, y_k))\},$$

■ Algoritmo:

$$y_{k+1} = y_k + \frac{h}{24} (55f(t_k, y_k) - 59f(t_{k-1}, y_{k-1}) + 37f(t_{k-2}, y_{k-2}) - 9f(t_{k-3}, y_{k-3}))$$

■ Método explícito con error global $\mathcal{O}(h^4)$

Desarrollo AB4

CI:	y_0
Runge-Kutta:	y_1, y_2, y_3
AB3:	y_4, y_5, \dots, y_N

- Entrada: f, a, b, N, y_a
- Proceso:
 - Obtención de la variable independiente discretizada t
 - Inicialización del vector solución y en a
 - Para k desde 0 hasta 2 aplicar el método de Runge-Kutta:
 - Cálculo de k_1, k_2, k_3 y k_4 :

$$k_1 = f(t_k, y_k), \quad k_2 = f\left(t_k + \frac{h}{2}, y_k + \frac{h}{2}k_1\right),$$
$$k_3 = f\left(t_k + \frac{h}{2}, y_k + \frac{h}{2}k_2\right), \quad k_4 = f(t_{k+1}, y_k + hk_3).$$

- $y_{k+1} = y_k + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4)$

- Fin para k
- Para k desde 4 hasta N :

$$y_{k+1} = y_k + \frac{h}{24} (55f(t_k, y_k) - 59f(t_{k-1}, y_{k-1}) + 37f(t_{k-2}, y_{k-2}) - 9f(t_{k-3}, y_{k-3}))$$

- Fin para k
- Salida: t, y

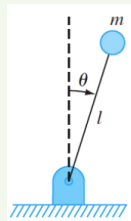
Ejemplo 1.

Consideremos una barra de longitud l que se mueve en el plano XY , como se ilustra en la figura. La barra se fija en uno de sus extremos con un alfiler y con una masa en el otro. Consideremos $g = 9.81m/s^2$ y $l = 0.5$.

Este sistema se modeliza mediante

$$\theta''(t) - \frac{g}{l}\theta(t) = 0,$$

con condiciones iniciales $\theta(0) = 0$ y $\theta'(0) = 0.25rad/s$



Ejemplo 1.

Consideremos una barra de longitud l que se mueve en el plano XY , como se ilustra en la figura. La barra se fija en uno de sus extremos con un alfiler y con una masa en el otro. Consideremos $g = 9.81m/s^2$ y $l = 0.5$.

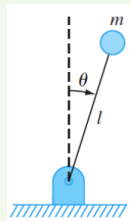
Este sistema se modeliza mediante

$$\theta''(t) - \frac{g}{l}\theta(t) = 0,$$

con condiciones iniciales $\theta(0) = 0$ y $\theta'(0) = 0.25rad/s$

■ Transformación del PVI:

$$\left. \begin{array}{l} \theta_1(t) = \theta(t) \\ \theta_2(t) = \theta'(t) \end{array} \right\} \Rightarrow \left. \begin{array}{l} \theta'_1(t) = \theta'(t) = \theta_2(t) \\ \theta'_2(t) = \theta''(t) = \frac{g}{l}\theta_1(t) \end{array} \right\} \quad \theta_1(0) = 0, \theta_2(0) = 0.25$$



Ejemplo 1.

Consideremos una barra de longitud l que se mueve en el plano XY , como se ilustra en la figura. La barra se fija en uno de sus extremos con un alfiler y con una masa en el otro. Consideremos $g = 9.81m/s^2$ y $l = 0.5$.

Este sistema se modeliza mediante

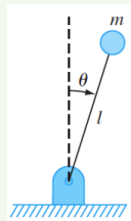
$$\theta''(t) - \frac{g}{l}\theta(t) = 0,$$

con condiciones iniciales $\theta(0) = 0$ y $\theta'(0) = 0.25rad/s$

■ Transformación del PVI:

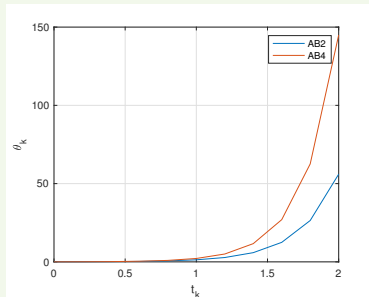
$$\left. \begin{array}{l} \theta_1(t) = \theta(t) \\ \theta_2(t) = \theta'(t) \end{array} \right\} \Rightarrow \left. \begin{array}{l} \theta'_1(t) = \theta'(t) = \theta_2(t) \\ \theta'_2(t) = \theta''(t) = \frac{g}{l}\theta_1(t) \end{array} \right\} \quad \theta_1(0) = 0, \theta_2(0) = 0.25$$

```
function dy = PVI(t,theta)
    t1=theta(1); t2=theta(2);
    dy=[t2; 9.81/0.5*t1];
end
```



Ejemplo. 1

t_k	θ_k , AB2	θ_k , AB4
0	0	0
0.2	0.0500	0.0565
0.4	0.1294	0.1604
0.6	0.2873	0.3978
0.8	0.6149	0.9338
1	1.3074	2.1677
1.2	2.7728	5.0179
1.4	5.8792	11.6454
1.6	12.4216	26.9700
1.8	26.4216	62.5868
2	56.0108	144.9417



Ejemplo. 1

- Estimación numérica del orden:

$$\epsilon_N = \left\| \theta_k^{(N)} - \theta_{1+2k}^{(2N)} \right\|, \quad k = 0, 1, \dots, N$$

N	AB2		AB4	
	ϵ_N	$\log_2 \left(\frac{\epsilon_{N/2}}{\epsilon_N} \right)$	ϵ_N	$\log_2 \left(\frac{\epsilon_{N/2}}{\epsilon_N} \right)$
10	–	–	–	–
20	71.999063	–	46.905822	–
40	61.943449	0.217026	11.105819	2.078451
80	31.681017	0.967333	1.425948	2.961322
160	12.711503	1.317484	0.148408	3.264278
320	4.681163	1.441195	0.014194	3.386138
640	1.677780	1.480313	0.001304	3.443546

3

Métodos implícitos: Adams-Moulton

Métodos implícitos

- Puntos:

$$\{(t_{k-n}, f(t_{k-n}, y_{k-n})), \dots, (t_k, f(t_k, y_k)), (t_{k+1}, f(t_{k+1}, y_{k+1}))\}.$$

- En los métodos implícitos se utilizan puntos desde y_{k-n} hasta y_k , pero también se utiliza y_{k+1}
- ➔ Para definir el polinomio de interpolación utilizaremos puntos que no conocemos todavía
- ➔ Son más complejos que los métodos explícitos, pero en general son más estables que los métodos explícitos

- Partimos de dos puntos: $(t_k, f(t_k, y_k))$ y $(t_{k+1}, f(t_{k+1}, y_{k+1}))$,
- Calculamos el único polinomio de interpolación que pasa por ellos:

$$\begin{aligned} p(\tau) &= f(t_{k+1}, y_{k+1}) + \frac{f(t_{k+1}, y_{k+1}) - f(t_k, y_k)}{t_{k+1} - t_k}(\tau - t_{k+1}) \\ &= \frac{\tau - t_k}{h} f(t_{k+1}, y_{k+1}) + \frac{t_{k+1} - \tau}{h} f(t_k, y_k). \end{aligned}$$

- Aproximando la función por el polinomio de interpolación:

$$\begin{aligned} y(t_{k+1}) &= y(t_k) + \int_{t_k}^{t_{k+1}} f(\tau, y(\tau)) d\tau \\ &\approx y_k + \int_{t_k}^{t_{k+1}} p(\tau) d\tau = y_k + \frac{h}{2} f(t_{k+1}, y_{k+1}) + \frac{h}{2} f(t_k, y_k). \end{aligned}$$

Método de Adams-Moulton de un paso (AM2)

$$y_{k+1} = y_k + \frac{h}{2} (f(t_{k+1}, y_{k+1}) + f(t_k, y_k))$$

- Método implícito con error global $\mathcal{O}(h^2)$.

Método de Adams-Moulton de dos pasos (AM3)

$$y_{k+1} = y_k + \frac{h}{12} (-f(t_{k-1}, y_{k-1}) + 8f(t_k, y_k) + 5f(t_{k+1}, y_{k+1})),$$

- Método implícito con error global $\mathcal{O}(h^3)$.

Método de Adams-Moulton de tres pasos (AM4)

$$y_{k+1} = y_k + \frac{h}{24} (f(t_{k-2}, y_{k-2}) - 5f(t_{k-1}, y_{k-1}) + 19f(t_k, y_k) + 9f(t_{k+1}, y_{k+1}))$$

- Método implícito con error global $\mathcal{O}(h^4)$.

Métodos de Adams-Moulton

En general:

- Un método implícito de Adams-Moulton de n pasos, tendrá orden $n + 1$
- Se necesita un método distinto para inicializar y_1, y_2, \dots
- Hay que resolver una ecuación no lineal para aplicar el método y calcular y_{k+1}

- Entrada: f, a, b, N, y_a
- Proceso:
 - Discretización de de la variable independiente: t_0, t_1, \dots, t_N
 - Inicialización del vector solución y en a : y_0
 - Aplicar el método de Runge-Kutta para calcular y_1 e y_2
 - Para k desde 2 hasta $N - 1$, encontrar cada nuevo valor y_{k+1} con AM4 resolviendo la ecuación no lineal:

$$g(y_{k+1}) = y_{k+1} - y_k - \frac{h}{24}(f(t_{k-2}, y_{k-2}) - 5f(t_{k-1}, y_{k-1}) \\ + 19f(t_k, y_k) + 9f(t_{k+1}, y_{k+1})) = 0,$$

utilizando el [método de Newton](#) con $x_0 = y_k$ como estimación inicial.

- Salida: t, y

Método de Newton

- Expresión iterativa:

$$x_{n+1} = x_n - \frac{g(x_n)}{g'(x_n)}, \quad n = 0, 1, 2, \dots,$$

- Utilizado para aproximar soluciones de ecuaciones no lineales $g(x) = 0$
- Requiere de una primera aproximación x_0 de la solución
- Genera una secuencia $\{x_n\}_{n \in \mathbb{N}}$ de estimaciones cada vez más próximas a la solución bajo ciertas condiciones de convergencia

```
function [t,y] = AM4(f,a,b,N,ya)
h=(b-a)/N;
t=a:h:b;
t=t(:);
y=zeros(N+1,1);
y(1)=ya;
maxiter=10;
tol=1e-6;
% Primeros pasos con el metodo de Runge-Kutta
for k=1:2
    k1=feval(f,t(k),y(k))';
    k2=feval(f,t(k)+h/2,y(k)+h*k1/2)';
    k3=feval(f,t(k)+h/2,y(k)+h*k2/2)';
    k4=feval(f,t(k+1),y(k)+h*k3)';
    y(k+1)=y(k)+h*(k1+2*k2+2*k3+k4)/6;
end
% Siguientes pasos con el metodo AM4
for k=3:N
    fk = feval(f,t(k),y(k));
    fkm1 = feval(f,t(k-1),y(k-1));
    fkm2 = feval(f,t(k-2),y(k-2));
    % Metodo de Newton
    iter=1; dif=1;
    x0=y(k);
    while and(iter<maxiter,dif>tol)
        [fx0,dfx0]=feval(f,t(k+1),x0);
        g=x0-y(k)-h/24*(fkm2-5*fkm1+19*fk+9*fx0);
        dg=1-h/24*9*dfx0;
        x1=x0-g/dg;
        dif=abs(x1-x0);
        iter=iter+1;
        x0=x1;
    end
    y(k+1)=y(k)+h/24*( fkm2-5*fkm1+19*fk+9*feval(f,t(k+1),x0));
end
end
```

Ejemplo 2. $y'(t) = (1 - 2t)y(t)$, $y(0) = 1$, $t \in [0, 3]$ con AM4 y $N = 16$

1. Definimos $y_0 = 1$

2. Calculamos y_1 e y_2 con RK4:

$$y_1 = 1.164558, \quad y_2 = 1.264115$$

3. Partiendo de y_1 e y_2 , aplicamos el **método de Newton** ($tol = 10^{-6}$, $maxiter = 10$) para encontrar y_3 con AM4 resolviendo $g(y_3) = 0$ donde:

$$g(y_3) = y_3 - y_2 - \frac{h}{24}(f(t_0, y_0) - 5f(t_1, y_1) + 19f(t_2, y_2) + 9f(t_3, y_3))$$

tomando como **estimación inicial** $x_0 = y_2$:

$$x_i = x_{i-1} - \frac{g(x_{i-1})}{g'(x_{i-1})}, \quad i = 1, 2, 3, \dots \quad \Rightarrow y_3$$

4. Calculamos y_4 con AM4 resolviendo con el método de Newton:

$$g(y_4) = y_4 - y_3 - \frac{h}{24}(f(t_1, y_1) - 5f(t_2, y_2) + 19f(t_3, y_3) + 9f(t_4, y_4)) = 0$$

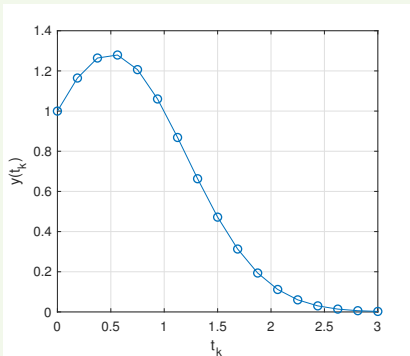
tomando como **estimación inicial** $x_0 = y_3$.

5. ...

Ejemplo 2. $y'(t) = (1 - 2t)y(t)$, $y(0) = 1$, $t \in [0, 3]$ con AM4 y $N = 16$

t_k	y_k
0	1
0.1875	1.164558
0.3750	1.264115
0.5625	1.279164
0.7500	1.206360
0.9375	1.060307
1.1250	0.868565
1.3125	0.663162
1.500	0.471989
1.6875	0.313181
1.8750	0.193762
2.0625	0.111785
2.2500	0.060136
2.4375	0.030159
2.6250	0.014092
2.8125	0.006129
3	0.002477

```
function [fun,dfun] = PVI(t,y)
    fun=(1-2*t).*y;
    dfun=1-2*t;
end
```



4

Métodos predictor-corrector

- Los métodos implícitos se suelen utilizar para mejorar los resultados obtenidos con un método explícito
- La combinación de los métodos explícito e implícito dan lugar a los métodos predictor-corrector

Métodos predictor-corrector

- En los métodos predictor-corrector, la aproximación y_{k+1} se calcula como una predicción $y_{k+1}^{(p)}$ utilizando un método explícito y posteriormente se utiliza un método implícito del mismo orden para mejorar o corregir esta predicción

ABM2

$$\text{Predictor AB2: } y_{k+1}^{(p)} = y_k + \frac{h}{2}(3f(t_k, y_k) - f(t_{k-1}, y_{k-1}))$$

$$\text{Corrector AM2: } y_{k+1} = y_k + \frac{h}{2}(f(t_{k+1}, y_{k+1}^{(p)}) + f(t_k, y_k))$$

ABM4

$$y_{k+1}^{(p)} = y_k + \frac{h}{24}(55f(t_k, y_k) - 59f(t_{k-1}, y_{k-1}) + 37f(t_{k-2}, y_{k-2}) - 9f(t_{k-3}, y_{k-3})),$$

$$y_{k+1} = y_k + \frac{h}{24}(f(t_{k-2}, y_{k-2}) - 5f(t_{k-1}, y_{k-1}) + 19f(t_k, y_k) + 9f(t_{k+1}, y_{k+1}^{(p)})),$$

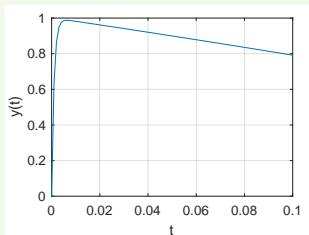
5

Resolución de problemas rígidos

Ecuaciones rígidas o stiff

- La ecuación presenta cambios muy bruscos en un espacio reducido de tiempo
- ➔ Determinados métodos numéricos son numéricamente inestables
- ➔ En general, los métodos numéricos funcionan mejor cuanto más bajo sea su orden y suelen tener mejor funcionamiento los métodos implícitos frente a los explícitos

Ejemplo 3. $y'(t) = -1000y(t) + 3000 - 2000e^t$, $t \in [0, 0.1]$, $y(0) = 0$



- Solución exacta: $y(t) = 3 - 1.9980e^t - 1.002e^{-1000t}$
- En unas pocas milésimas experimenta un gran crecimiento
- Tras este crecimiento, la función tiene una tasa de cambio pequeña.
- Todas las regiones no requieren ser estudiadas con el mismo detalle.
- Analizando la parte homogénea de la EDO, se puede determinar el tamaño de paso necesario para que la solución sea estable.

Métodos de paso adaptativo

- Utilizan un mayor o menor número de puntos en cada región del intervalo dependiendo del problema
- No se definen nodos equiespaciados y se concentra un mayor número de puntos en las regiones en las cuales el comportamiento de la función cambia más rápidamente

Resolución de problemas rígidos con Matlab




Matlab ODE solvers

ode23	non-stiff, low order
ode113	non-stiff, variable order
ode15s	stiff, variable order, includes DAE
ode23s	stiff, low order
ode23t	trapezoid rule
ode23tb	stiff, low order
ode45	non-stiff, medium order (Runge-Kutta)

- Todos los algoritmos usan paso adaptativo
- Pueden aproximar la solución en los nodos solicitados
- Mejor opción para problemas rígidos: ode15s

Ejemplo 3. $y'(t) = -1000y(t) + 3000 - 2000e^t$, $t \in [0, 0.1]$, $y(0) = 0$

Método	N	E_N
Euler	10	3.4938e+09
	100	0.3686
Euler implícito	10	0.0911
	100	0.1324
AB4	10	8.0190e+16
	100	1.3998e+37
AM4	10	5.9765e+06
	100	0.0071
ABM4	10	3.0556e+20
	100	0.0240
ode23	10	0.0011
	20	0.0016
ode45	10	1.8939e-04
	20	6.4718e-04
ode15s	10	2.4195e-05
	20	2.7014e-04

-  Lecciones magistrales
-  Material complementario: A fondo
-  Bibliografía recomendada

...Y por supuesto:

TEST DE APRENDIZAJE!!

