Laboratorio de Cálculo Numérico
Ecuaciones Diferenciales
Euler Explícito e Implícito

1. Introducción

Definición. Una ecuación diferencial de primer orden con valor inicial es una etpresión de la forma

$$\begin{aligned} y'(t) &= f(t,\ y(t)) \\ y(a) &= y_0 \end{aligned} \tag{1}$$

donde $f : \mathbb{R}^2 \to \mathbb{R}$ tiene derivadas continuas de varios órdenes.

La existencia y unicidad de la función solución, $y^*(t)$, de (1) se garantiza con hipótesis adecuadas. Sin embargo, en la mayoría de los casos la función $y^*(t)$ no puede obtenerse en forma explícita, por lo cual se requiere aproximaciones numéricas en un número finito de puntos en un intervalo cerrado $[a,\ b]$.

2. Euler Explícito

Sea $\mathcal{P} = \{a = t_0 < t_1 < t_2 < \ldots < t_n = b\}$ una partición igualmente espaciada de, $n+1$ puntos del intervalo $[a,\ b]$, donde $h = t_{k+1} - t_k$ es el tamaño de paso de la partición.

Por el Teorema Fundamental del Cálculo Diferencial se tiene que

$$\int_{t_k}^{t_{k+1}} y'(t)\ dt = y(t_{k+1}) - y(t_k). \tag{2}$$

Por otra parte, si $h = t_{k+1} - t_k$ es pequeña podemos interpolar la función $y'(t)$ para $t \in [t_k,\ t_{k+1}]$ por la constante $f(t_k,\ y(t_k))$, es decir

$$\int_{t_k}^{t_{k+1}} y'(t)\ dt \int_{t_k}^{t_{k+1}} f(t_k,\ y(t_k))\ dt = hf(t_k,\ y(t_k)), \tag{3}$$

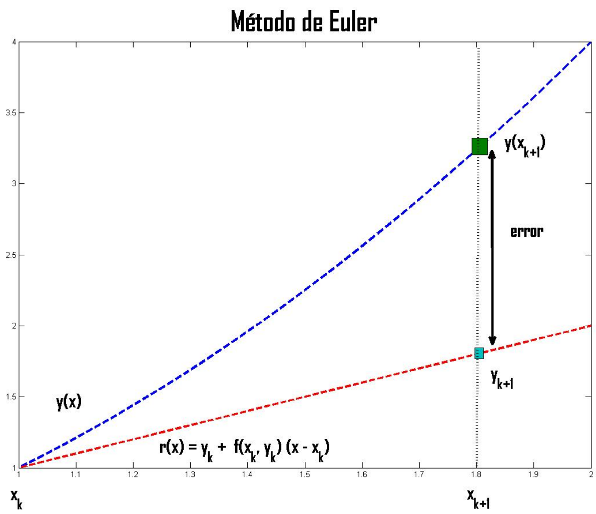
concluimos al usar 2 y 3 que

$$y(t_{k+1}) - y(t_k) \approx hf(t_k,\ y(t_k)).$$

o bien $y(t_{k+1}) \approx y(t_k) + hf(t_k,\ y(t_k))$.

Método de Euler

$y_0 \leftarrow y(t_0)$
Para $k = 0,\ 1,\ \ldots,\ n-1$ hacer
$\quad y_{k+1} \leftarrow y_k + h * f(t_k, y_k)$
Fin



Programar en Matlab :

function $[y] = euler(fname,\ a,\ b,\ y_0,\ n)$
% La cadena de caracteres $fname$ es la función en Matlab
% de $f(t,\ y(t))$ de la ecuación diferencial $y' = f(t,\ y(t)$.
% $[a,\ b]$ es el intervalo donde se aproxima la solución.
% $y(a) = y_0$ y n es tal que $h = (b-a)/(n+1)$.
% La salida y es un vector de dimensión $n+1$ obtenido por Euler tal que

% $y_k \approx y(t_k),\ k = 1,\ 2,\ \ldots,\ n+1$

3. Método de Euler Implícito

En la ecuación 3 cuando h es pequeño se interpola

$$\int_{t_k}^{t_{k+1}} y'(t)\ dt \int_{t_k}^{t_{k+1}} f(t_{k+1},\ y(t_{k+1}))\ dt = hf(t_{k+1},\ y(t_{k+1})), \qquad (4)$$

y se obtiene el esquema

$$y(t_{k+1}) = y_k + h * f(t_{k+1},\ y(t_{k+1}))$$

que es una expresión implícita.

Para tener ventaja de esta fórmula suponemos que $\hat{y}_{k+1}$ es una aproximación a $y(t_{k+1})$ entonces se corrige y_{k+1} a

$$y_{k+1} = y_k + h * f(t_{k+1},\ \hat{y}_{k+1}).$$

El término $\hat{y}_{k+1}$ se llama el término de predicción y y_{k+1} se llama el término de correción.

Euler Implícito

$y_0 \leftarrow y(t_0)$
Para $k = 0,\ 1,\ \ldots,\ n-1$ hacer
  $\hat{y}_{k+1} \leftarrow y_k + h * f(t_k, y_k)$
  $y_{k+1} \leftarrow y_k + h * f(t_{k+1},\ \hat{y}_{k+1})$
Fin

Programar en Matlab :

function $[y] = eulerimp(fname,\ a,\ b,\ y_0,\ n)$
% La cadena de caracteres $fname$ es la función en Matlab
% de $f(t,\ y(t))$ de la ecuación diferencial $y' = f(t,\ y(t)$.
% $[a,\ b]$ es el intervalo donde se aproxima la solución.

% $y(a) = y_0$ y n es tal que $h = (b-a)/(n+1)$.
% La salida y es un vector de dimensión $n+1$ obtenido por Euler con
% predicción y correción tal que
% $y_k \approx y(t_k),\ k = 1,\ 2,\ \ldots,\ n+1$

4. Ejemplos

$y'(t) = ty(t) + t^3,\ \ y(0) = 1,\ t \in [0,\ 1].$
La solución etplícita es : $y(t) = 3e^{t^2/2} - t^2 - 2$.

Use las funciones *euler.m* y *eulerimp.m* con $n = 10$ y grafica las soluciones verdaderas y las aproximaciones, en el script **pruebaeuler.m**, para obtener la siguiente gráfica: