Métodos Computacionales

Ec. Diferenciales Ordinarias

Agosto 23, 2023

S. Alexis Paz

Departamento de QUÍMICA TEÓRICA Y COMPUTACIONAL
Facultad de Ciencias Químicas
Universidad Nacional de Córdoba

Ecuaciones Diferenciales Ordinarias (vs Parciales)

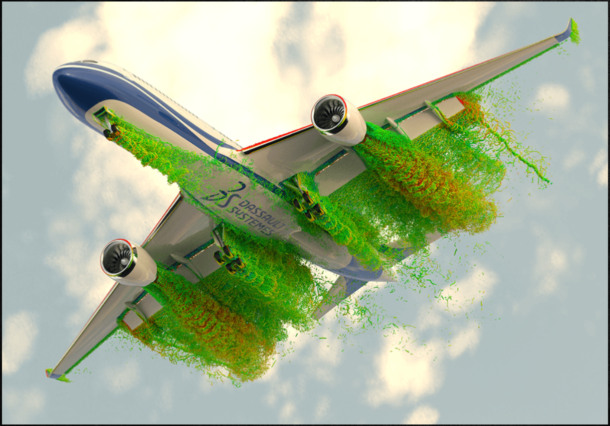
¿$y(x)$? ¿$u(x, y, \ldots, z)$?

Lineales. . . $$\sum_{i=0}^{k} a_k(x) \frac{\mathrm{d}^k y}{\mathrm{d}x^k} = b(x)$$

y . . . ¡ No Lineales !

Navier-Stokes:

$$\rho\left(\frac{\partial \mathbf{v}}{\partial t} + (\mathbf{v}\cdot\nabla)\mathbf{v}\right) = -\nabla p + \mu\nabla^2\mathbf{v} + \mathbf{f}$$



Fojas *et~al.* . (2013).

APCBEE Procedia. 7. 86–92.

SOLAER Ingeniería

Lotka-Volterra:

$$\frac{\mathrm{d}x}{\mathrm{d}t} = \alpha x - \beta x y$$

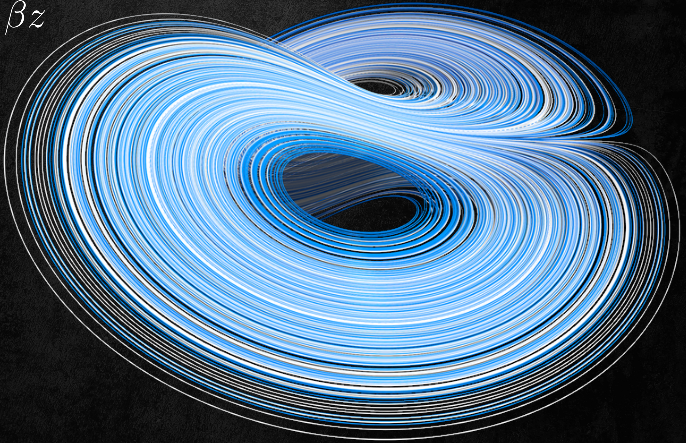
$$\frac{\mathrm{d}y}{\mathrm{d}t} = \delta x y - \gamma y$$

Atractor de Lorenz:

$$\frac{\mathrm{d}x}{\mathrm{d}t} = \sigma(y - x)$$

$$\frac{\mathrm{d}y}{\mathrm{d}t} = x(\rho - z) - y$$

$$\frac{\mathrm{d}z}{\mathrm{d}t} = xy - \beta z$$



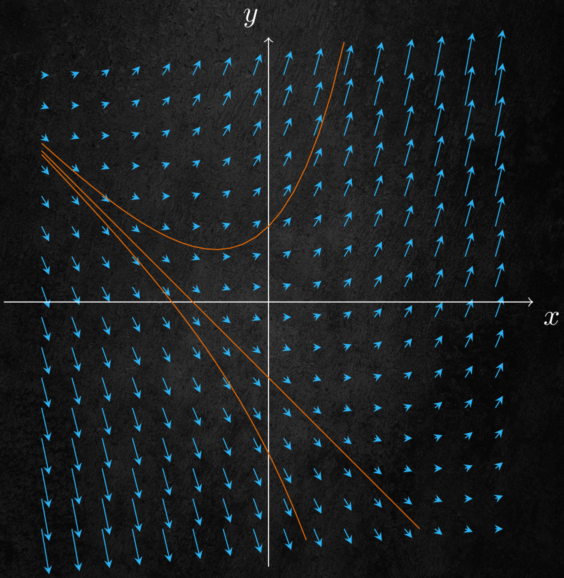
$$F\left(x, y, \frac{\mathrm{d}y}{\mathrm{d}x}, \frac{\mathrm{d}^2y}{\mathrm{d}x^2}, \ldots, \frac{\mathrm{d}^ky}{\mathrm{d}x^k}\right) = 0 \longleftarrow \text{EDO de orden } k$$

$$\left.\begin{aligned} z_i \equiv \frac{\mathrm{d}^iy}{\mathrm{d}x^i} \qquad i = 1, \ldots, k-1 \\ F\left(x, y, z_1, z_2, \ldots, z_{k-1}, \frac{\mathrm{d}^ky}{\mathrm{d}x^k}\right) = 0 \end{aligned}\right\} k \text{ EDO de orden } 1$$

EXPLÍCITAS

$$\mathbf{F}\left(x, \mathbf{y}, \frac{\mathrm{d}\mathbf{y}}{\mathrm{d}x}\right) = 0 \longrightarrow \boxed{\frac{\mathrm{d}\mathbf{y}}{\mathrm{d}x} = \mathbf{F}(x, \mathbf{y})}$$

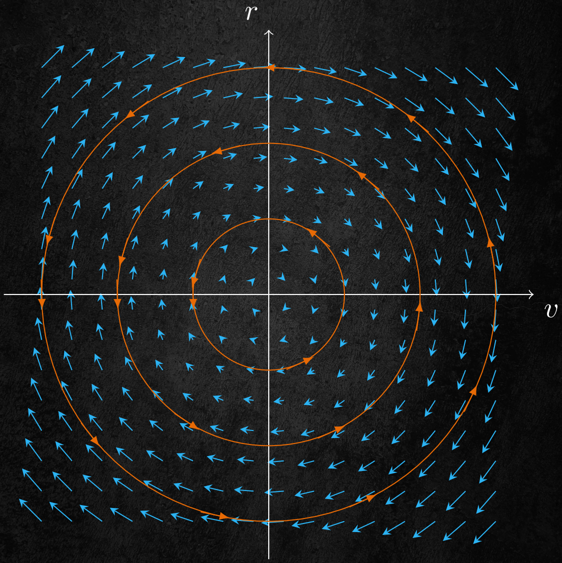
$$\frac{\mathrm{d}y}{\mathrm{d}x} = y + x$$



Osc. Armónico:

$$\frac{\mathrm{d}r}{\mathrm{d}t} = v$$

$$\frac{\mathrm{d}v}{\mathrm{d}t} = -kr$$



Ejemplo:

$$\frac{\mathrm{d}y}{\mathrm{d}x} = yx$$

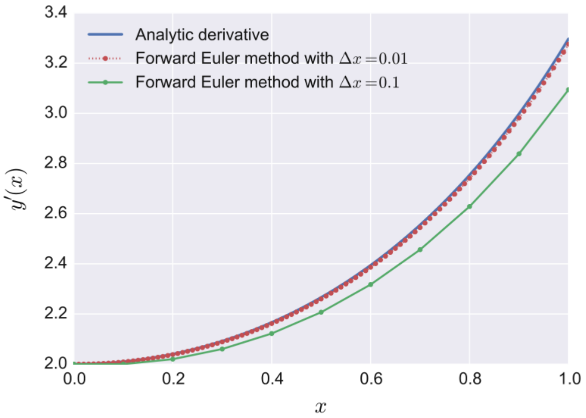
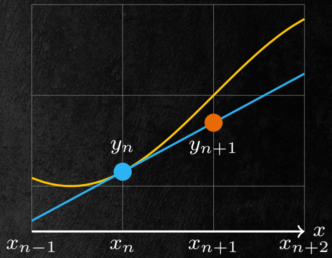


Figure 5.1 Numerical results of the Forward Euler method compared with exact analytic solution

Euler hacia adelante:

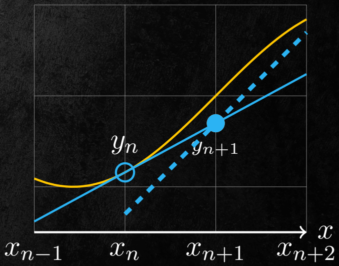
$$y_{n+1} = y_n + hF(x_n, y_n) + O\left(h^2\right)$$



Euler hacia atrás:

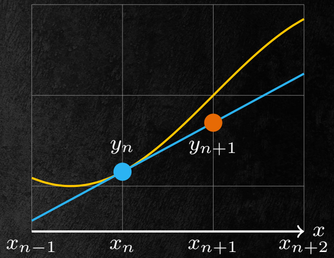
$$y_{n+1} = y_n + hF(x_{n+1}, y_{n+1}) + O\left(h^2\right)$$

Resuelvo buscando la raiz y_{n+1}.



Euler hacia adelante:

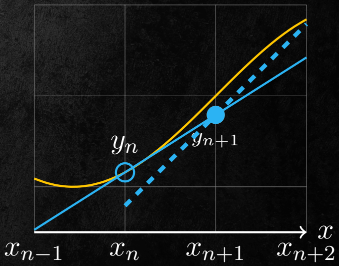
$$y_{n+1} = y_n + hF(x_n, y_n) + O\left(h^2\right)$$



Euler hacia atrás:

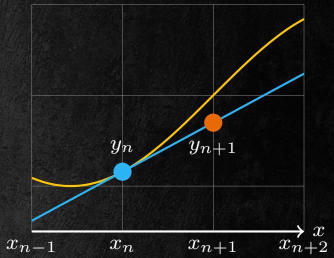
$$y_{n+1} = y_n + hF(x_{n+1}, y_{n+1}) + O\left(h^2\right)$$

Resuelvo buscando la raiz y_{n+1}.



Euler hacia adelante:

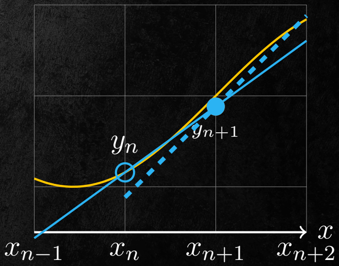
$$y_{n+1} = y_n + hF(x_n, y_n) + O\left(h^2\right)$$



Euler hacia atrás:

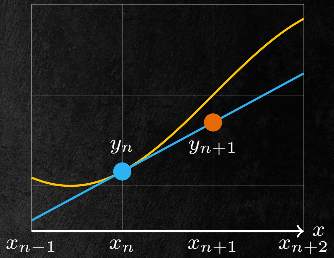
$$y_{n+1} = y_n + hF(x_{n+1}, y_{n+1}) + O\left(h^2\right)$$

Resuelvo buscando la raiz y_{n+1}.



Euler hacia adelante:

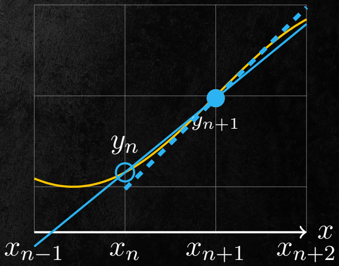
$$y_{n+1} = y_n + hF(x_n, y_n) + O\left(h^2\right)$$



Euler hacia atrás:

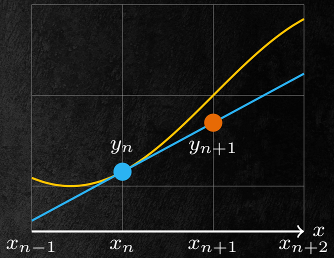
$$y_{n+1} = y_n + hF(x_{n+1}, y_{n+1}) + O\left(h^2\right)$$

Resuelvo buscando la raiz y_{n+1}.



Euler hacia adelante:

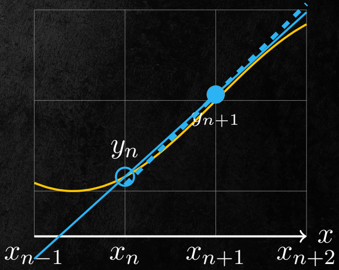
$$y_{n+1} = y_n + hF(x_n, y_n) + O\left(h^2\right)$$



Euler hacia atrás:

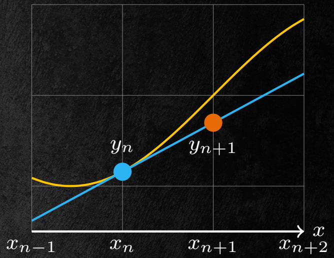
$$y_{n+1} = y_n + hF(x_{n+1}, y_{n+1}) + O\left(h^2\right)$$

Resuelvo buscando la raiz y_{n+1}.



Euler hacia adelante:

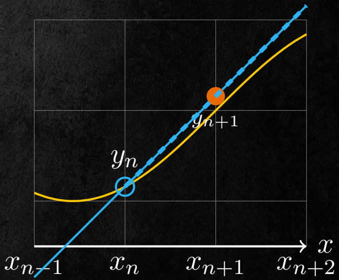
$$y_{n+1} = y_n + hF(x_n, y_n) + O\left(h^2\right)$$



Euler hacia atrás:

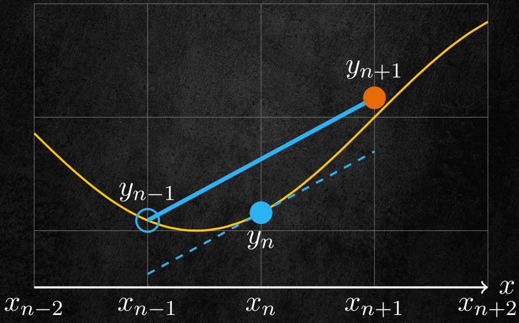
$$y_{n+1} = y_n + hF(x_{n+1}, y_{n+1}) + O\left(h^2\right)$$

Resuelvo buscando la raiz y_{n+1}.



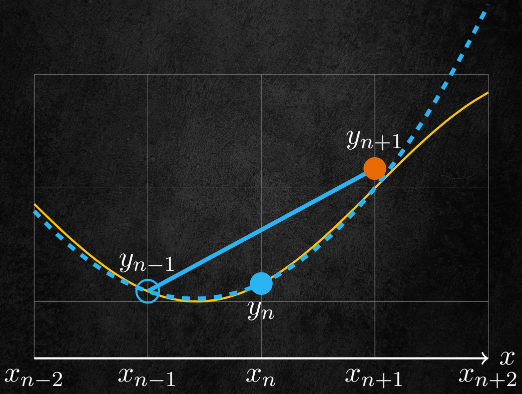
Leap-Frog:

$$y_{n+1} = y_{n-1} + 2hF(x_n, y_n) + O\left(h^3\right)$$



Verlet:

$$\mathbf{y}_{n+1} = 2\mathbf{y}_n - \mathbf{y}_{n-1} + h^2 \left[\frac{\partial \mathbf{F}(x,\mathbf{y})}{\partial \mathbf{y}}\right]_{(x_n,\mathbf{y}_n)} \mathbf{F}(x_n,\mathbf{y}_n) + O\left(h^4\right)$$



ERROR

. . . total de la iteracion n

ABSOLUTO: $\epsilon_n = |x_n - L|$

RELATIVO: $\tilde{\epsilon}_n = |x_n - L|/x_n$

. . . por usar una compu

PRECISIÓN / NUMÉRICO: $O(\epsilon_M)$

. . . por la aproximación

TRUNCAMIENTO: - *LOCAL*

$$\frac{\mathrm{d}f(x_n)}{\mathrm{d}x} = \frac{f(x_{n+1}) - f(x_n)}{2} + O\left(h^3\right)$$

$$\int_{x_n}^{x_{n+1}} f(x)\,\mathrm{d}x = f\left(x + h/2\right)h + O\left(h^2\right) \xrightarrow{\times(N-1)}$$

$$y_{n+1} = y_n + hF(x_n, y_n) + O\left(h^3\right) \xrightarrow{\times(N-1)}$$

. . . pero es recurrente!

GLOBAL:

$$(N-1)O(h^x)$$

$$= \frac{b-a}{h}O(h^x)$$

$$= O(h^{x-1})$$

CONVERGENCIA

Cuando buscamos aproximarnos a un valor (e.g. busqueda de raíces), si el error disminuye en cada iteración...

CONVERGE: $\epsilon_{n+1}/\epsilon_n \leq 1$

ESTABILIDAD

Si hacemos $(N-1)$ aproximaciones de forma recurrente, además del error global, hay que ver si una perturbación no se propaga por la recurrencia...

ESTABLE: $\delta y_{n+1}/\,\delta y_n \leq 1$

Condiciones de estabilidad

$$\left|1 + h\frac{\partial F(x,y)}{\partial y}\right| \leq 1$$ Euler hacia adelante

$$\left|1 - h\frac{\partial F(x,y)}{\partial y}\right| \geq 1$$ Euler hacia atrás

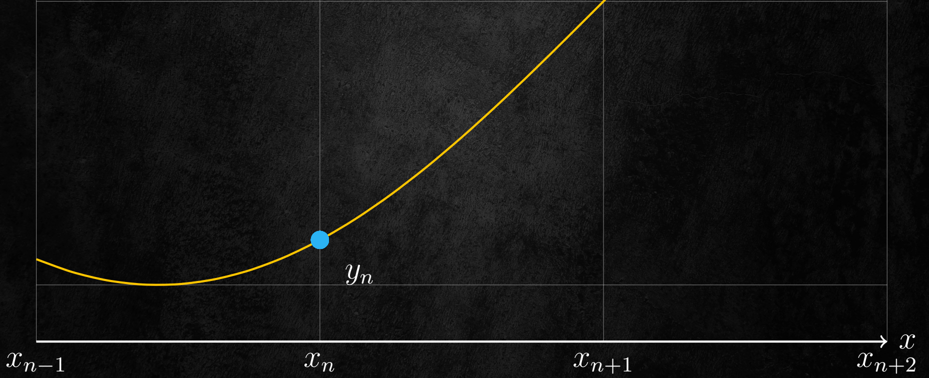
$$\mathrm{Re}\left(\frac{\partial F(x,y)}{\partial y}\right) = 0 \wedge h\left|\mathrm{Im}\left(\frac{\partial F(x,y)}{\partial y}\right)\right| \leq 1$$ Leap-Frog

Euler de punto medio:

$$y_{n+1} = y_n + k_2 + O\left(h^3\right) \longrightarrow \text{Leap-Frog}$$

$$k_2 = hF\left(x_n + h/2, \underbrace{y_n + k_1/2}_{\text{Euler}}\right)$$

$$k_1 = hF(x_n, y_n)$$

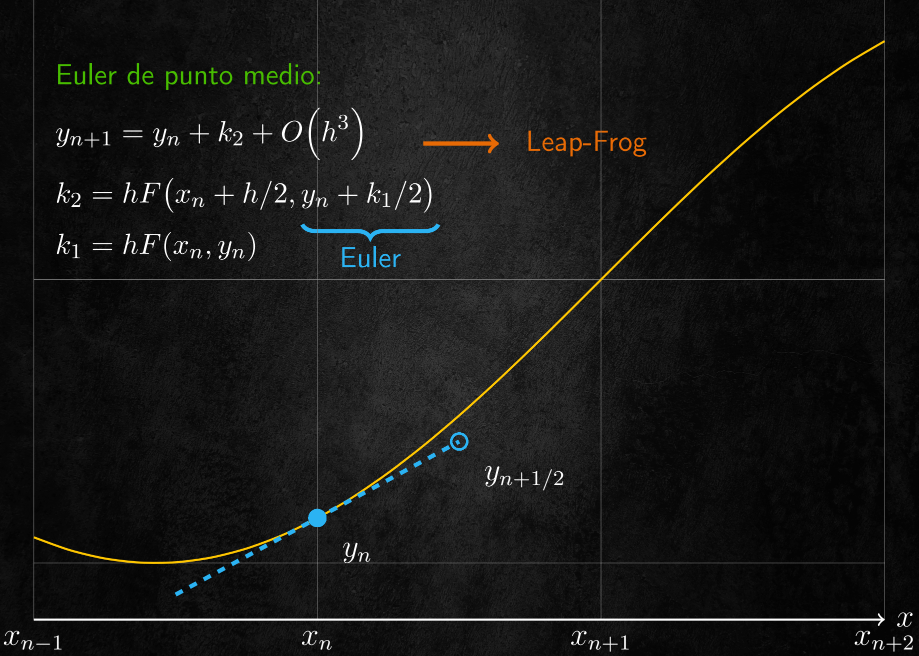


Euler de punto medio:

$$y_{n+1} = y_n + k_2 + O\left(h^3\right) \longrightarrow \text{Leap-Frog}$$

$$k_2 = hF\left(x_n + h/2, \underbrace{y_n + k_1/2}_{\text{Euler}}\right)$$

$$k_1 = hF(x_n, y_n)$$

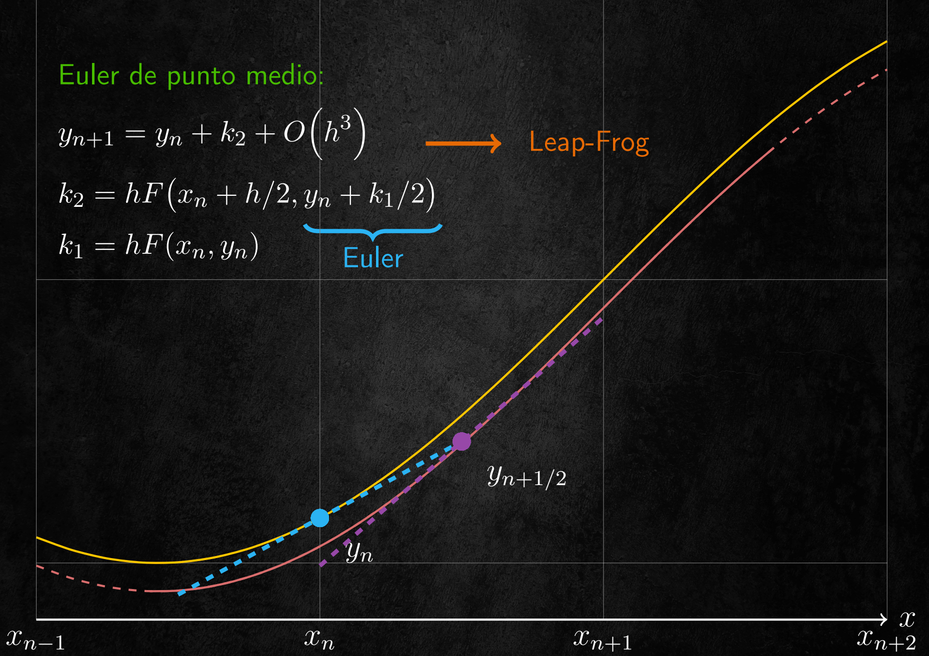


Euler de punto medio:

$$y_{n+1} = y_n + k_2 + O\left(h^3\right) \longrightarrow \text{Leap-Frog}$$

$$k_2 = hF\left(x_n + h/2, \underbrace{y_n + k_1/2}_{\text{Euler}}\right)$$

$$k_1 = hF(x_n, y_n)$$

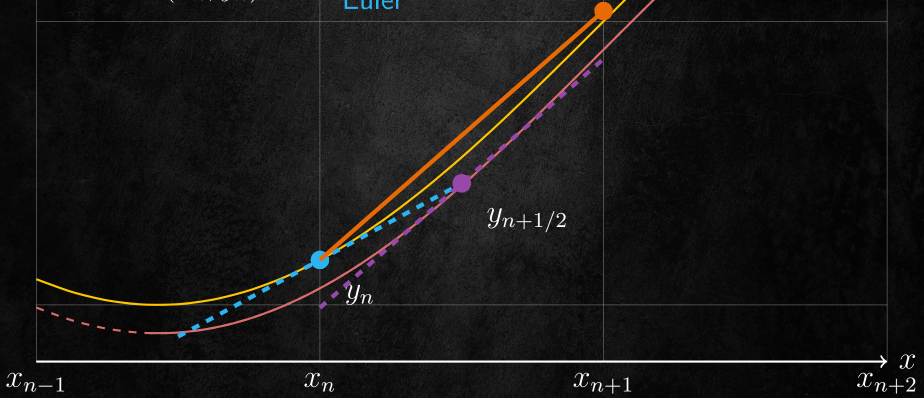


Euler de punto medio:

$$y_{n+1} = y_n + k_2 + O\left(h^3\right) \longrightarrow \text{Leap-Frog}$$

$$k_2 = hF\left(x_n + h/2, \underbrace{y_n + k_1/2}_{\text{Euler}}\right)$$

$$k_1 = hF(x_n, y_n)$$

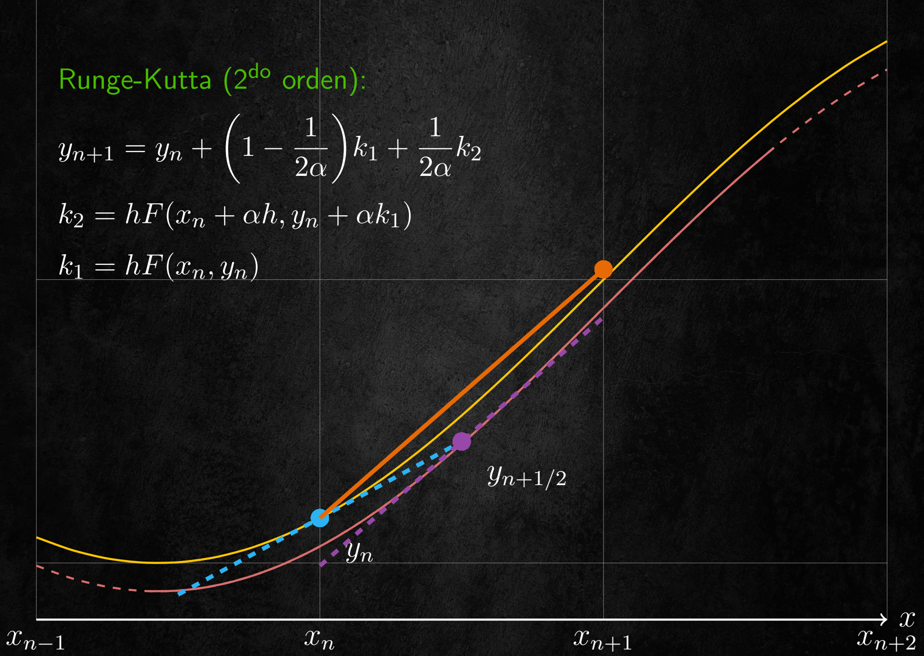


Runge-Kutta (2^{do} orden):

$$y_{n+1} = y_n + \left(1 - \frac{1}{2\alpha}\right)k_1 + \frac{1}{2\alpha}k_2$$

$$k_2 = hF(x_n + \alpha h, y_n + \alpha k_1)$$

$$k_1 = hF(x_n, y_n)$$



Runge-Kutta (4^{to} orden):

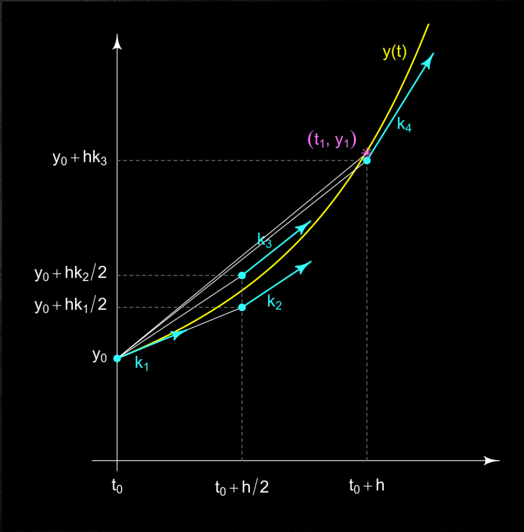
$$y_{n+1} = y_n + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4) + O\left(h^5\right)$$

$$k_4 = hF(x_n + h, y_n + k_3)$$

$$k_3 = hF\left(x_n + \frac{h}{2}, y_n + \frac{k_2}{2}\right)$$

$$k_2 = hF\left(x_n + \frac{h}{2}, y_n + \frac{k_1}{2}\right)$$

$$k_1 = hF(x_n, y_n)$$



Algoritmos EDO (orden 1)

$$y_{n+1} = y_n + hF(x_n, y_n) + O\left(h^2\right)$$ Euler hacia adelante

$$y_{n+1} = y_n + hF(x_{n+1}, y_{n+1}) + O\left(h^2\right)$$ Euler hacia atrás

$$y_{n+1} = y_n + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4) + O\left(h^5\right)$$ RK4

$$y_{n+1} = y_{n-1} + 2hF(x_n, y_n) + O\left(h^3\right)$$ Leap-Frog

$$y_{n+1} = 2y_n - y_{n-1} + h^2\left[\frac{\partial F(x_n, y_n)}{\partial y} F(x_n, y_n)\right] + O\left(h^4\right)$$ Verlet

Algoritmos EDO (orden k)

$$\mathbf{y}_{n+1} = \mathbf{y}_n + h\mathbf{F}(x_n, \mathbf{y}_n) + O\left(h^2\right) \quad \text{Euler hacia adelante}$$

$$\mathbf{y}_{n+1} = \mathbf{y}_n + h\mathbf{F}(x_{n+1}, \mathbf{y}_{n+1}) + O\left(h^2\right) \quad \text{Euler hacia atrás}$$

$$\mathbf{y}_{n+1} = \mathbf{y}_n + \frac{1}{6}(\mathbf{k}_1 + 2\mathbf{k}_2 + 2k_3 + \mathbf{k}_4) + O\left(h^5\right) \quad \text{RK4}$$

$$\mathbf{y}_{n+1} = \mathbf{y}_{n-1} + 2h\mathbf{F}(x_n, \mathbf{y}_n) + O\left(h^3\right) \quad \text{Leap-Frog}$$

$$\mathbf{y}_{n+1} = 2\mathbf{y}_n - \mathbf{y}_{n-1} + \underbrace{h^2\left[\frac{\partial \mathbf{F}(x, \mathbf{y})}{\partial \mathbf{y}}\right]_{(x_n, \mathbf{y}_n)} \mathbf{F}(x_n, \mathbf{y}_n)}_{¿?} + O\left(h^4\right) \quad \text{Verlet}$$