- Root finding
- Polynomial interpolation / splines
- Least squares & minimax approximations
- Quadrature
- Numerical linear algebra (factorizations, eigenvalues)
- Numerical solution of ODEs:

$$\vec{x}(t) = \begin{bmatrix} x_1(t) \\ x_2(t) \\ \vdots \\ x_n(t) \end{bmatrix}$$

$$\left.\begin{aligned} \frac{d\vec{x}}{dt} &= \vec{F}(\vec{x}) \\ \vec{x}(0) &= \vec{x}_0 \end{aligned}\right\} \text{ Initial value problem (IVP)}$$

Simple example: $\dfrac{dx}{dt} = \lambda x(t) \qquad \lambda \in \mathbb{C}$

$$\Rightarrow \quad x(t) = \alpha e^{\lambda t}$$

$$\Rightarrow \quad x(0) = \alpha \Rightarrow \alpha = x(0)$$

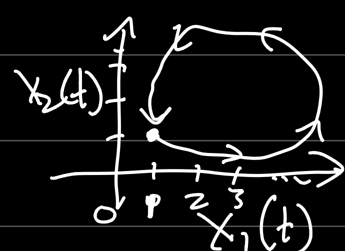
Prototype for solvers, systems of ODEs, etc.

Ex: Lotka-Volterra (predator-prey model)

$$\frac{d\vec{x}}{dt} = \frac{d}{dt}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \vec{F}(\vec{x}) = \begin{bmatrix} x_1 - x_1 x_2 \\ -x_2 + x_1 x_2 \end{bmatrix}$$

x_1 = prey population

x_2 = predator population

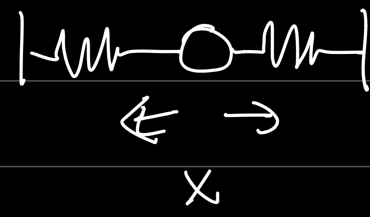


Ex: SIR model → susceptible, infectious, recovered

UC Boulder

Ex:

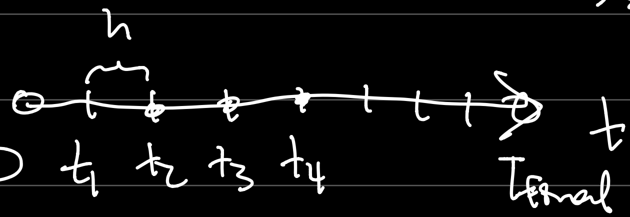
$$\frac{d^2 x}{dt^2} = f(x)$$



$$\frac{d}{dt}\begin{bmatrix} x \\ v \end{bmatrix} = \begin{bmatrix} v \\ f(x) \end{bmatrix}$$

Time-stepping methods: given $\vec{x}_k \approx x(t_k)$

$\Rightarrow$ solve for $\vec{x}_{k+1} \approx x(t_{k+1})$

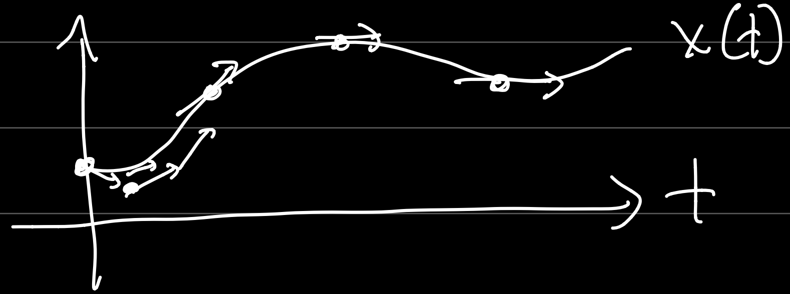


$t_{k+1} = t_k + h$ $\quad\quad$ $\vec{x}_0 \approx \vec{x}(t_0)$ = initial condition.

Forward Euler's method: $\quad \dfrac{dx}{dt} = \lim_{h\to 0} \dfrac{x(t+h) - x(t)}{h}$

$$\frac{x(\underbrace{t_0+h}_{t_1}) - x(t_0)}{h} \approx \frac{dx}{dt} = f(x(t_0))$$

$$\left.\begin{aligned} x_1 &= x_0 + h f(x_0) \\ x_2 &= x_1 + h f(x_1) \end{aligned}\right\} \quad x_{k+1} = x_k + h f(x_k)$$



Quantify $\underline{\text{accuracy}}$. – local accuracy
– global accuracy

$$\frac{x(t_{k+1}) - x(t_k)}{h} \approx \left.\frac{dx}{dt}\right|_{t_k}$$

$$\Rightarrow \left| \frac{x(t_{k+1}) - x(t_k)}{h} - \left.\frac{dx}{dt}\right|_{t_k} \right| = O(h) \quad \text{first order approx.}$$

$$x(t_{k+1}) = x(t_k+h) = x(t_k) + h\left.\frac{dx}{dt}\right|_{t_k} + \frac{h^2}{2}\left.\frac{d^2x}{dt^2}\right|_{t_k} + \dots$$

$$\frac{x(t_k+h) - x(t_k)}{h} - \left.\frac{dx}{dt}\right|_{t_k} = \frac{h}{2}\frac{d^2x}{dt^2} + \text{higher order terms}$$

$\underline{\text{One-step methods}}$: $x'(t) = f(x,t)$

$O(h)$ ① Fwd Euler: $x_{k+1} = x_k + hf(x_k, t_k)$

$O(h)$ ② Backward Euler: $x_{k+1} = x_k + hf(x_{k+1}, t_{k+1})$

$O(h^2)$ ③ Trapezoid rule: $x_{k+1} = x_k + \frac{h}{2}\left(f(x_k,t_k) + f(x_{k+1},t_{k+1})\right)$

$O(h^2)$ ④ Improved Euler / Heun's method: $x_{k+1} = x_k + \frac{h}{2}\left(f(x_k,t_k) + f(\underbrace{x_k + hf(x_k,t_k)}, t_{k+1})\right)$

①, ④ = "explicit" methods

②, ③ = "implicit" methods

Approximation to $x_{k+1} \Rightarrow \tilde{x}_{k+1}$

$O(h^4)$ ⑤ RK-4 (4th order Runge-Kutta method)

4 "stages" but one-step.
$$\begin{cases} k_1 = f(x_k, t_k) \\ k_2 = f(x_k + \frac{h}{2} k_1, t_k + \frac{h}{2}) \\ k_3 = f(x_k + \frac{h}{2} k_2, t_k + \frac{h}{2}) \\ k_4 = f(x_k + h k_3, t_k + h) \end{cases}$$

$$x_{k+1} = x_k + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4)$$

General framework:

one step methods of the form

$$x_{k+1} = x_k + h \underbrace{\varphi(x_k, t_k, h)}_{\text{timestepper details}}$$

Assume φ = continuous in each variable

error:

Local → ① Truncation error: if $x_k = x(t_k)$ what is the error b/w $x(t_{k+1}) - x_{k+1}$?

Globals ② How truncation error accumulates & depends on h.

Def: Truncation error

$$T_k = \frac{x(t_{k+1}) - x(t_k)}{h} - \varphi(\overbrace{x(t_k)}^{\text{exact sol. at } t_k}, t_k, h)$$

① Fwd Euler : using Taylor expansions

$$x(t_{k+1}) = x(t_k + h) = x(t_k) + h x'(t_k) + \frac{h^2}{2} x''(\xi) \quad \xi \in [t_k, t_{k+1}]$$

$$T_k = \frac{1}{2} h x''(\xi) \longrightarrow O(h) \to 0 \text{ as } h \to 0$$

② Trapezoidal rule

$$x_{k+1} = x_k + \underbrace{\frac{h}{2}\left(f(x_k, t_k) + f(x_{k+1}, t_{k+1})\right)}$$

$$\varphi(x_k, x_{k+1}, t_k, h) = \frac{1}{2}\left(f(x_k, t_k) + f(x_{k+1}, t_{k+1})\right)$$

$$T_k = \frac{x(t_{k+1}) - x(t_k)}{h} - \varphi(x(t_k), x(t_{k+1}), t_k, h)$$

$$x(t_{k+1}) - x(t_k) = \int_{t_k}^{t_{k+1}} x'(t) = \int_{t_k}^{t_{k+1}} f(x, t)$$

$$\approx \underbrace{\frac{(t_{k+1} - t_k)}{2}}\left(f(x(t_k), t_k) + f(x(t_{k+1}), t_{k+1})\right)$$

$$\frac{h}{2}\left(f(x(t_k), t_k) + f(x(t_{k+1}), t_{k+1})\right)$$

$$T_k = \frac{x(t_{k+1}) - x(t_k)}{h} - \varphi(x(t_k), x(t_{k+1}), t_k, h)$$

$$= \frac{1}{h}\left(\int_{t_k}^{t_{k+1}} f(x, t) - \frac{h}{2}\left(f(x(t_k), t_k) + f(x(t_{k+1}), t_{k+1})\right)\right)$$

$$\leq \frac{h^2}{12} \max_{t \in [t_k, t_{k+1}]} |x'''(t)|$$

$$T_k \leq \frac{h^2}{12} \max_{t \in [t_k, t_{k+1}]} |x'''(t)|$$

Local errors = done.

Global error = need theory on exact solutions to ODEs.