Differential eqs.

- Ordinary diff. eqs. (ODEs) w/ boundary conditions
  - Finite diff. schemes → matrix problems
  - Shooting method

  ↳ Projects 1, 2

Now → ◉ ODEs w/ initial value conditions

  ↳ Project 3

- Partial diff. eqs. (PDEs) (multiple indep. variables)

  ↳ Project 5

ODEs w/ initial value conditions [Chap. 8]

- Recap classification:

- $$\frac{dy}{dt} = \underset{\substack{\uparrow \\ \text{known function}}}{f}(t, y)$$

  - First order: $\frac{dy}{dt}$
  - ODE: One indep. variable (t)
  - Init. value: $Y(t_0)$

- $$\frac{d^2y}{dt^2} = f(t, \frac{dy}{dt}, y)$$

  - Second order: $\frac{d^2y}{dt^2}$
  - ODE: (t)
  - Init values: $Y(t_0)$, $Y'(t_0)$

Example: Newton's 2nd law:

$$\frac{d^2x}{dt^2} = \frac{1}{m} F(t, \frac{dx}{dt}, x)$$

- Second-order eqs. can often be rewritten as two first-order eqs.

Example

$$\frac{d^2x}{dt^2} = \frac{1}{m} F\left(t, \frac{dx}{dt}, x\right) \qquad (*)$$

- Define a new variable: $v \equiv \frac{dx}{dt}$ (conveniently named...)
- Can write $\frac{d^2x}{dt^2}$ as $\frac{dv}{dt}$
- Two eqs:

1) $\frac{dx}{dt} = v(t)$ from def.

2) $\frac{dv}{dt} = \frac{1}{m} F(t, v, x)$ from $(*)$

- Two coupled, first-order diff. eqs. for the variables x and v, both functions of a single indep. variable t.

- You should do this in Project 3

- Linear vs non-linear diff. eqs.:

$$\frac{dy}{dt} = g^3(t)\, y(t) \qquad \underline{\text{Linear}} \quad (y(t))$$

$$\frac{dy}{dt} = g^3(t)\, y(t) - h(t)\, y^2(t) \qquad \underline{\text{Non-linear}} \quad (y^2(t))$$

- In project 3, the Coulomb interaction between particles produce non-linear eqs.

Eq. for $\frac{d^2x}{dt^2}$ contains term $\propto \frac{x - x_j}{|\vec{r} - \vec{r}_j|^3} = \frac{x - x_j}{\sqrt{(x-x_j)^2 + (y-y_j)^2 + (z-z_j)^2}^{\,3}}$

Non-lin. diff. eqs. typically requires a numerical approach! (An analytical sol. often does not exist.)

{ This type of info can be useful to mention in a report when motivating why you've studied something numerically }

Methods

- Euler's forward method ✓
  - ↳ Euler-Cromer's method ✓
  - ↳ Midpoint-method
  - ↳ Half-step method
- Varlet and Leapfrog ✓
- Predictor - Corrector ✓
- Runge-Kutta ✓

- Local (trunc.) error: $O(h^k)$
- Make this error at every step
- If we have n steps $n \propto \frac{1}{h}$

$\Rightarrow$ Global error

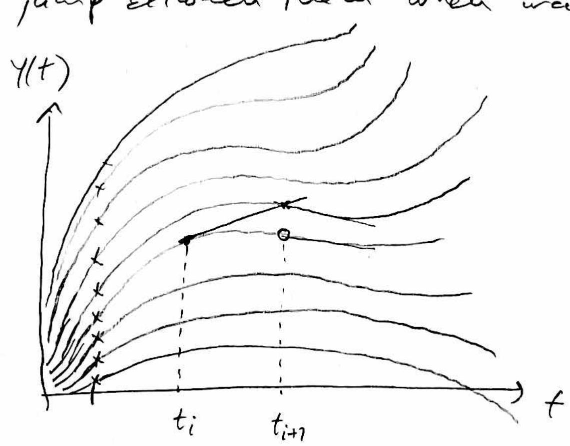
$$O(nh^k) = O(\tfrac{1}{h} h^k) = O(h^{k-1})$$

- General considerations:
  - Local / global truncation errors?
  - Number of function evaluations needed in order to find next step y_{i+1}? ($\longleftrightarrow$ FLOPs) $\quad f(t, y, y')$

$\longrightarrow$ A balancing act! Some methods particularly useful for special problems

- There's typically an infinite number of solutions, and we basically jump between them when making approx.

$$\frac{dy}{dt} = f(t, y)$$



Initial conditions pick out one of an infinite number of solutions