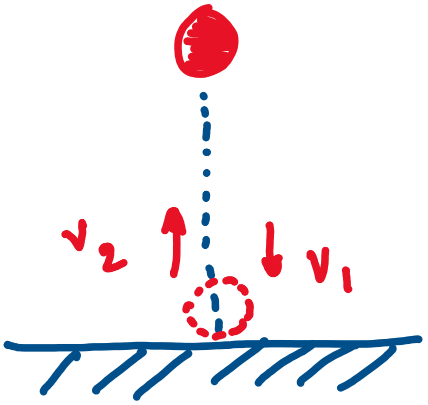
# Hybrid Systems

Dynamics change with time

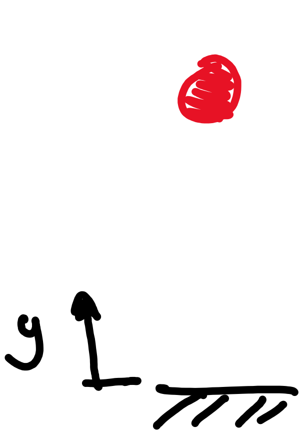
e.g. pong game, ball bouncing, walking, running, hopping.

## Example : Bouncing ball

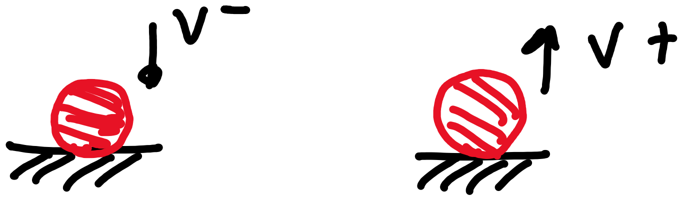


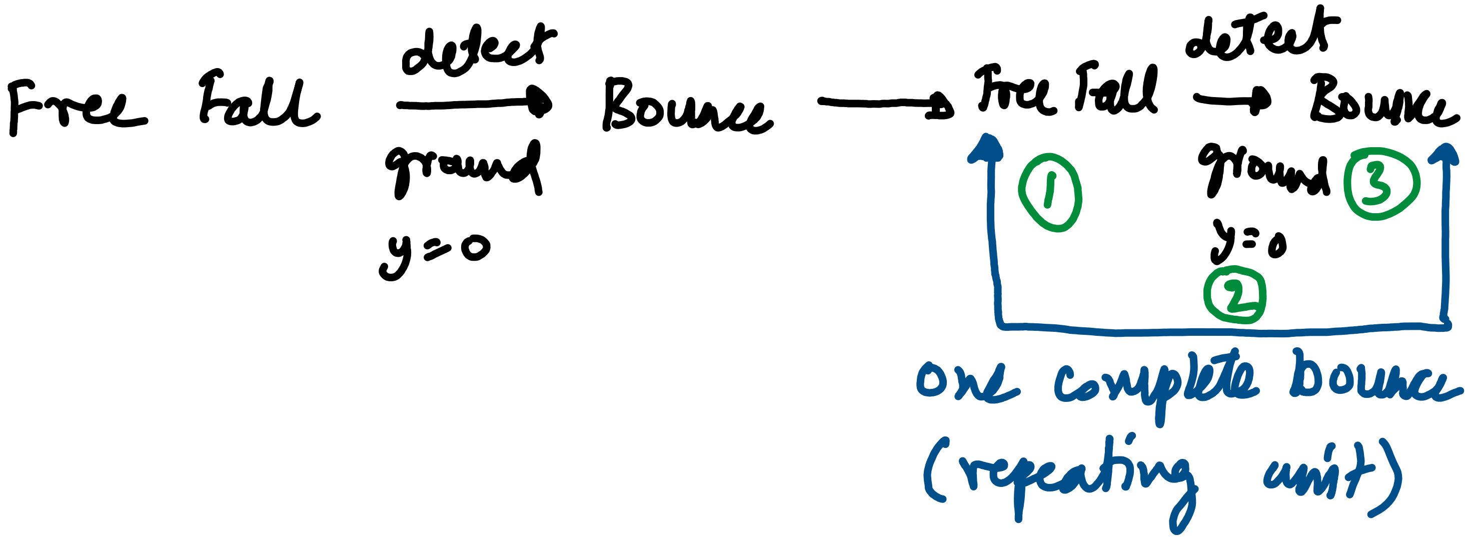
How to simulate / animate / analyze such systems.

## Free Fall



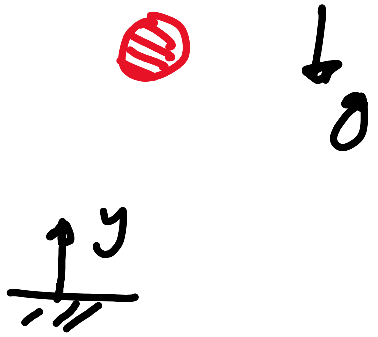
## Bounce





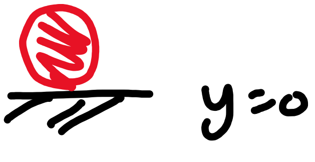
① Free Fall

$$\boxed{\ddot{y} = -g} \quad -①$$

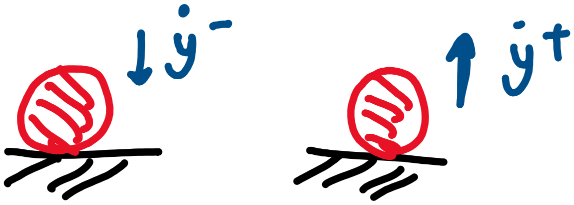


② Contact

t = ? when $y = 0$



③ Bounce



$\dot{y}^-$ - velocity before bounce

$\dot{y}^+$ - velocity after bounce

Law of restitution

$$-e = \frac{\dot{y}^+}{\dot{y}^-}$$

e - coefficient of restitution

$e = 0$ plastic collision ⇒ $\dot{y}^+ = 0$

$e = 1$ elastic collision ⇒ $\dot{y}^+ = -\dot{y}^-$

$0 \leq e \leq 1$ in real world.

# Simulation in Python

① function called one-bounce

$$\begin{tikzcd}\text{Free Fall} \arrow[r, "\text{detect}", "\text{ground}"'] & \text{Bounce} \arrow[l, bend left, "\text{Repeat}"]\end{tikzcd}$$

② Free Fall $\ddot{y} = -g$. [Integrate odeint]

Contact / Detect ground : stop integration
when $y=0$

integration: solve_ivp( EOM, $[t_0 \; t_f]$, $z_0$, \

$\dot{v}_y = -g$

$\dot{y} = v_y$

init condition

method='RK45',

event = contact ~ $y=0$, parm)

Bounce: $\dot{y}^+ = -e\,\dot{y}^-$