INTRODUCTION TO LAGRANGIAN MODELING
(for mech. systems)

(INFORMAL)

▷ PHYSICS we All know (and love)

Newton's law : $m\vec{a} = \vec{F} \rightarrow \begin{cases} m\ddot{x} = F_x \\ m\ddot{y} = F_y \\ m\ddot{z} = F_z \end{cases}$

(Particle in 3D) (Component wise)

<u>DRAWBACKS</u> : ▷ Form of equations is NOT invariant wrt. coordinate changes

→ Not so easy (systematic) to apply to many-body systems

→ Not easy to insert constraints in configuration space

How to build a more "practical" approach to modeling (mech.) systems?

1^{st} STEP) INCLUDE CONSTRAINTS FROM THE START.

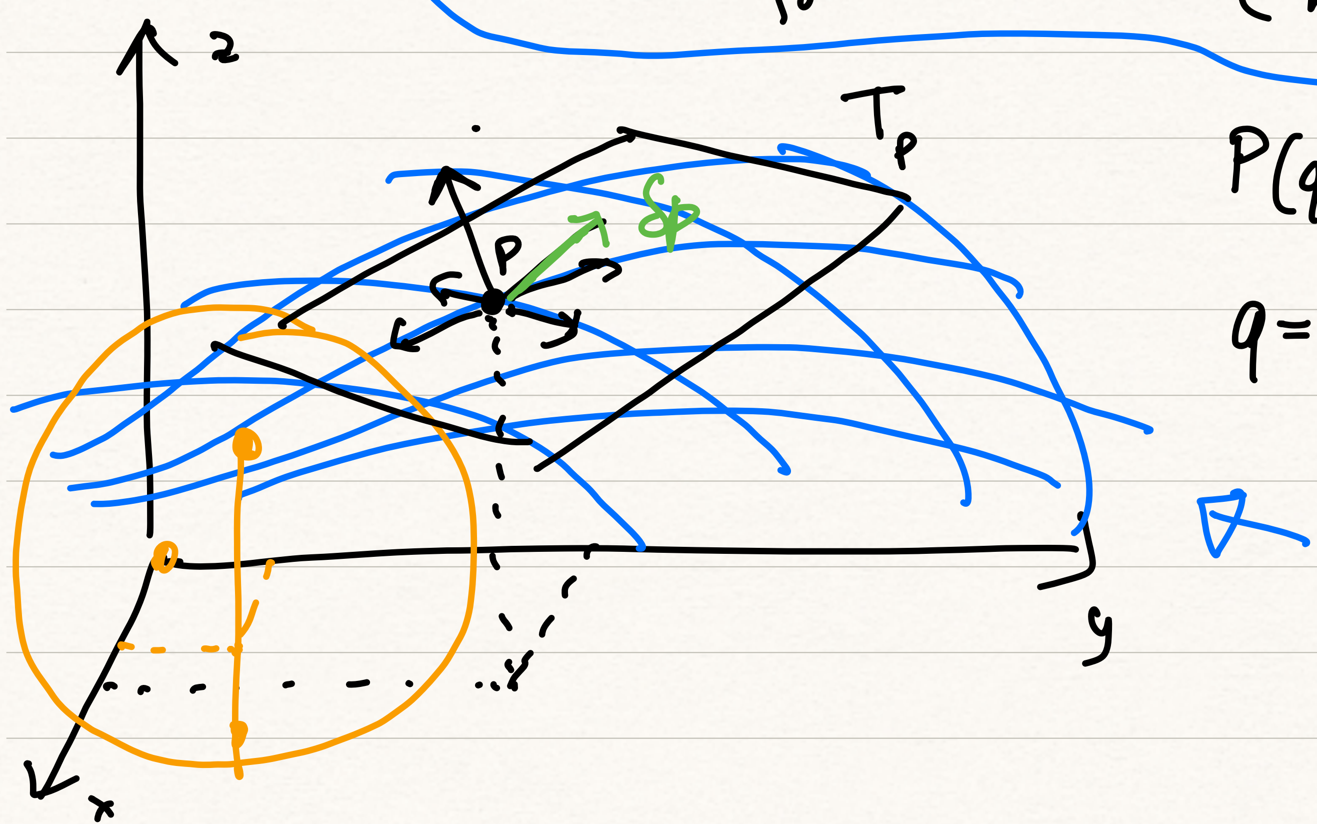
$\vec{x}$: config. vector, but not all possible config. are allowed.
$\in \mathbb{R}^n$

$\vec{q}$: generalized coordinates
$\in \mathbb{R}^k$

$k \leq n$

$\rightarrow$ we need $P(\vec{q}) = \vec{x}$

$\left[\frac{\partial P_i}{\partial q_j}\right](q)$ Full rank (k)



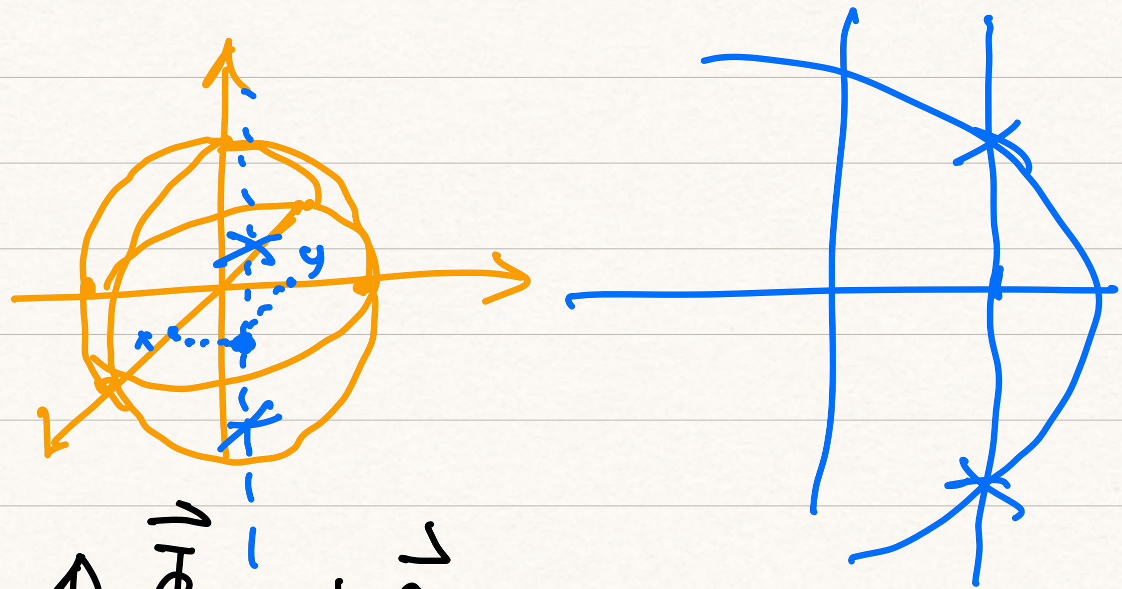
$P(q) = \begin{pmatrix} x \\ y \\ z(x,y) \end{pmatrix}$

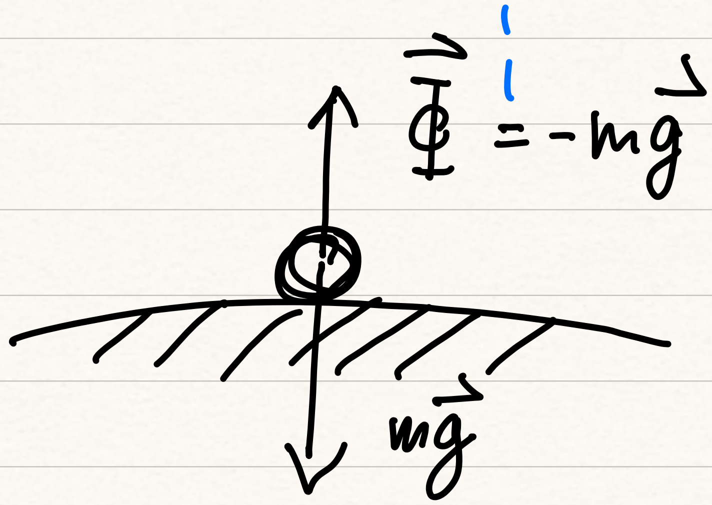
$q = \begin{pmatrix} x \\ y \end{pmatrix}$

Typically : CONSTRAINTS are given as implicit functions

$$\longrightarrow \Phi(x) = 0$$

Ex $\phi(x,y,z) = \underbrace{x^2 + y^2 + z^2}\ - \underline{1} = 0$





REACTION FORCES (~ constraints)

they induce <u>NO MOVEMENT</u>

$$\langle \vec{\Phi} , \delta p \rangle = 0$$

[VIRTUAL WORK PRINCIPLE]

δp : allowed "local" movement

~ Tangent vector