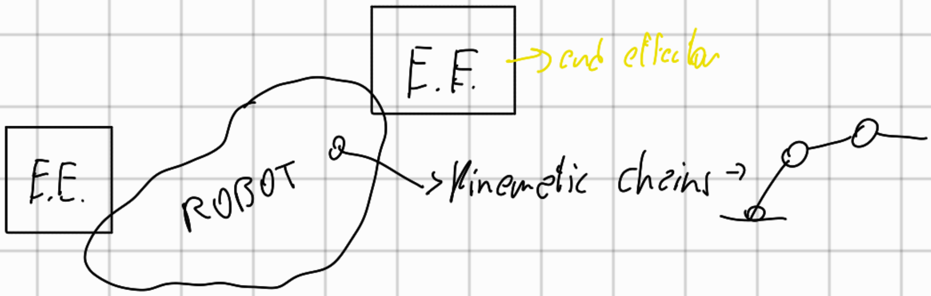
Constraint:

End effector: part of a robot that interacts with the world. It's moving.



In a mecanism there are bodies (rigid) and Joints.

Joint: connection that constrains the motion in a way of another

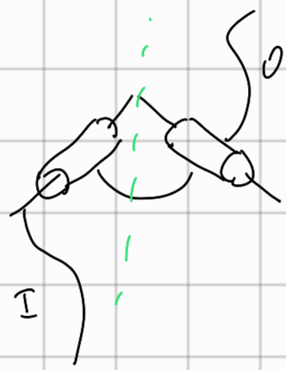
we will use only this one

Joints → Revolute (R), Prismatic (P), Helical (H) — 1 dof
Universal — 2 dof
Cilindrical
Ball — 3 dof

Revolute: hinge of a door
Prismatic: Sliding without rotating

Helical has a pitch (passo) wich is the rate between rotation and movement.

Universal Joint

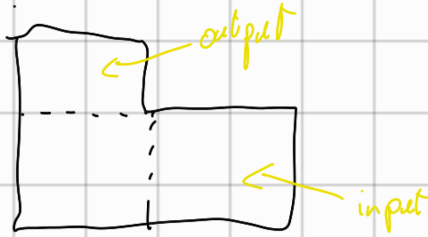


→ combining two revolut Joints you get a universal Joint. These two Joint must not have a ∅ degree between them. Only thing that does not allow is rotation.

With combinazation of revolut and prismatic Joints we can create every type of Joint.

An example of universal joint is done by folding a piece of paper like this

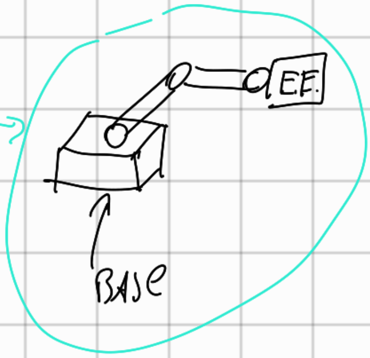
and then cutting one rectangle

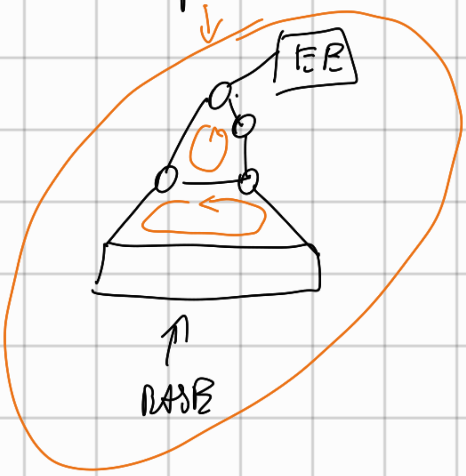


We have different types of kinematic chains

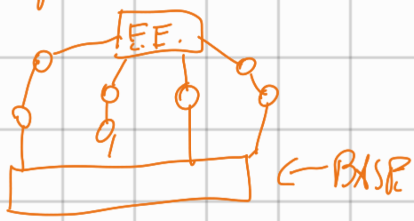
↳ open chains

↳ closed-loop chains





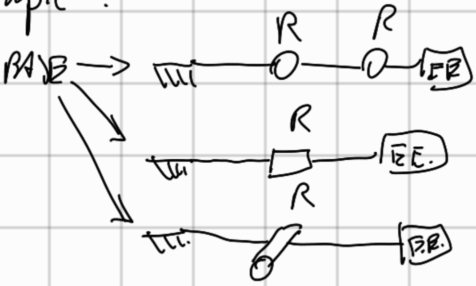
The only type of closed loop chains that we will do are purely parallel



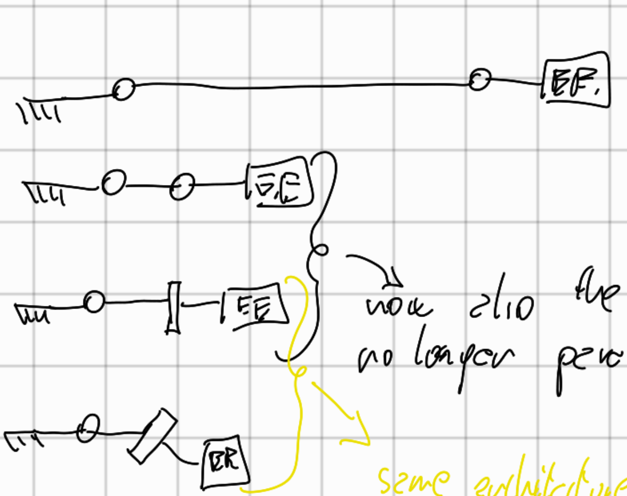
In a mechanism we have:

- architecture: the joints that you have and how they are organized.

  For example:



→ the architecture is two parallel revolut joint

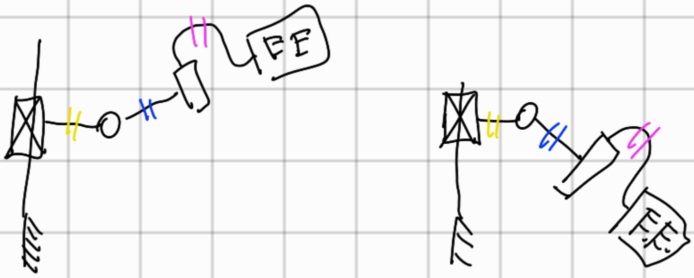


the architecture is the same but geometry changes

now also the architecture changes because they are no longer parallel joints but an angle.

same architecture but different geometry because they are two revolute joints at an angle (that changes)

the prismatic joint is represented like a crossed box 
(In drawings a cross means fact)



Same architecture ✓
Same geometry ✓
Different configuration ✗

In robots we have mechanisms
↓
BASE → kinematic chain ← End effector
↓
joints - - - - - - - → (helical joint)
↙ ↘
revolute prismatic

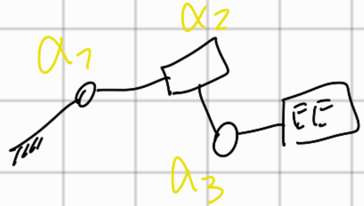
they generate every other linkage

Architecture : the way the chain is organized.

Geometry: the particular sizing of the chain
Configuration: the way the chain is organized in space and can always be modified.

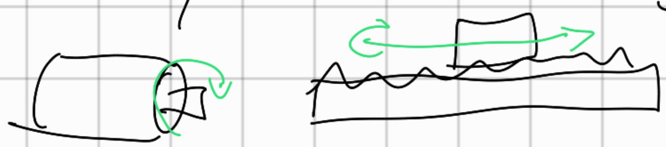
21/09/23

Simple mechanism like openchain.



Problems related to the robotic arm.
- position problem
  > forward : config param → pose of E.E.
  > inverse : pose → config parameters

motors :
rotary
linear (translating)



In the motors we have constraint that are limiting the movement of the shaft in only one direction.

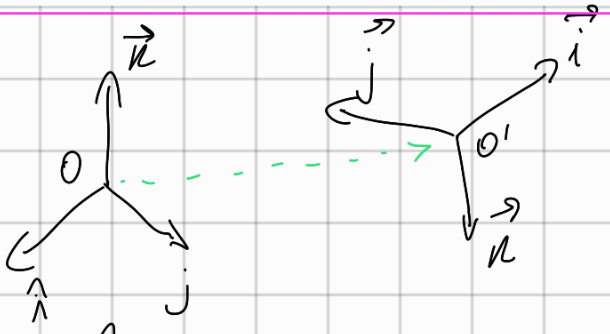
With motors we want to force reconfiguration in a mechanism.

$$\begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \end{bmatrix} = \begin{bmatrix} \frac{\pi}{6} \\ -\frac{\pi}{3} \\ -\frac{2\pi}{3} \end{bmatrix}$$

given these values we are able to calculate everything about the mechanism. this is the forward position problem.

The output of the forward is the univocal representation of the pose of the mechanism. In this case is the position of the end effector.

Euler angles

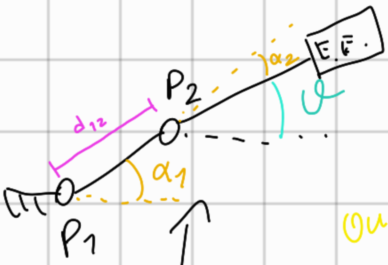


reference frame of the object.

You need the new position of O' in reference to O and then you need three angles to describe the rotation of the object

To turn the object you turn in order around each axes in order to reach the final position. So you first turn around $\vec{i}$, then turn around the new $\vec{k}$ and finally the new new $\vec{j}$

note on Euler angles



This is a planar mechanism. because is on the plane of the board

To provide the pose of the end effector you provide

output $\longrightarrow$ $\underbrace{[P_{2x}, P_{2y}, \vartheta]}_{\text{this is using Euclidean Planes.}}$

$\vartheta = \alpha_1 + \alpha_2$

$$\boxed{\begin{aligned} P_{2x} &= d_{12}\, \underset{\cos \alpha_1}{c_1} \\ P_{2y} &= d_{12}\, \underset{\sin \alpha_1}{s_1} \\ \vartheta &= \alpha_1 + \alpha_2 \end{aligned}}$$

FORWARD POSITION KINEMATIC

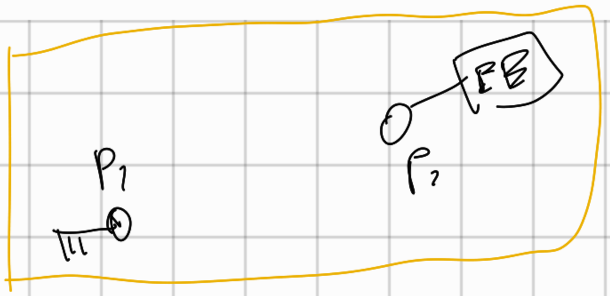
theta θ

The encoder on P_2 will measure α_2. Same goes for P_1

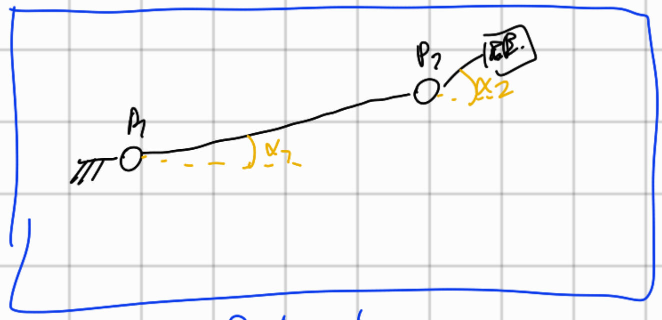
You can solve it in an open chain but not always in a closed chain in open form meaning something = to something.

Inverse position problem

We have the position of the EE and we have to figure out the position of the mechanism.

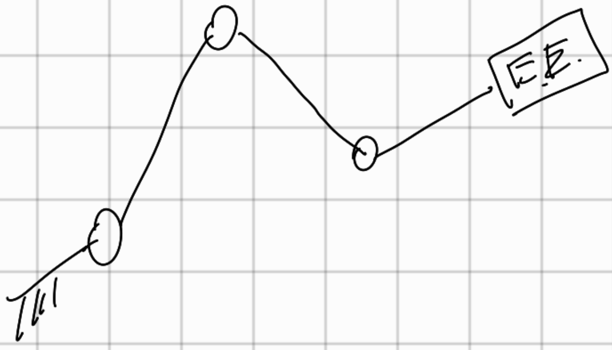


Input



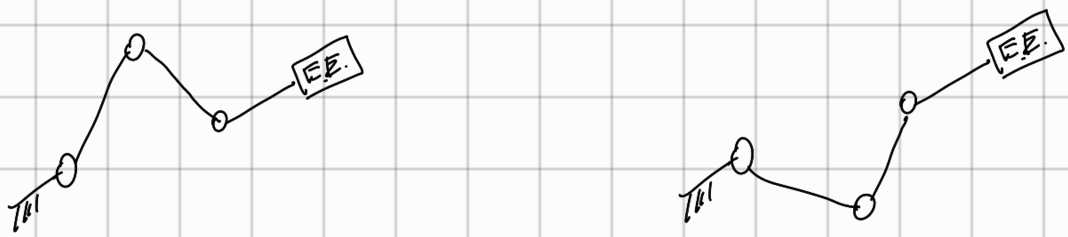
Output

3R



This is a planar mechanism and 3R means that the 3 joints are motorized.

Position of the base and of E.E. are given and we have to calculate the position of the joint. The solution is not unique:

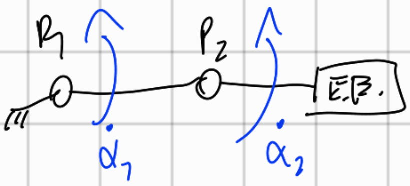


If for a 6 degree of freedom robot it's possible to get 5 values and then an uncertainty for the 6th value that can get ∞ values. So the robot becomes a 5 degree of freedom. Search robotics singularities to understand the lowering of degree of freedom.

Velocity Kinematics

Everything that works for position also goes for velocity

- forward: $\dot{\alpha}_1, \dot{\alpha}_2 \to [\dot{\vartheta}, \dot{P}_{2x}, \dot{P}_{2y}]$
- inverse: $[\dot{\vartheta}, \dot{P}_{2x}, \dot{P}_{2y}] \to \dot{\alpha}_1, \dot{\alpha}_2$



$$\begin{bmatrix}\dot\vartheta\\ \dot P_{2x}\\ \dot P_{2y}\end{bmatrix}$$ ← this is what you want for the forward.

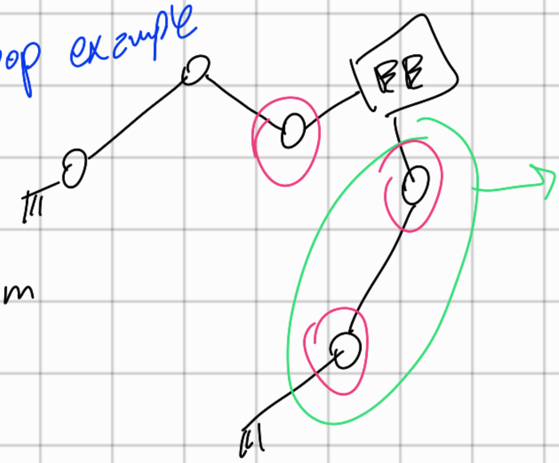
$$\left.\begin{aligned}\dot\vartheta &= \dot\alpha_1 + \dot\alpha_2\\ \dot P_{2x} &= d_{12}\, s_1\, \dot\alpha_1 = A\,\dot\alpha_1\\ \dot P_{2y} &= d_{12}\, c_1\, \dot\alpha_2 = B\,\dot\alpha_2\end{aligned}\right\}$$ they are linear in the derivatives of the position but they're non linear in the position of the joints.

We will see that $\dot{pos} = J\begin{bmatrix}\dot\alpha_1\\ \dot\alpha_2\end{bmatrix}$

J ↓ jacobian → linear

We can derive once more and we get acceleration. We derive once more and we get jerk.

Closed loop example

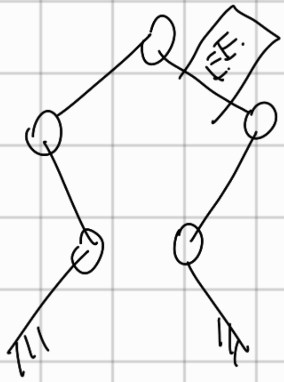


Planar mechanism

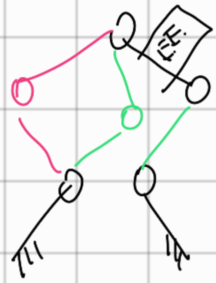
this should not be motorized otherwise they will fight the other motors if not moving in agreement. For example. But you could choose another subset to be motorized and would be the same.

For example

Different ways for the joint to position in a way to put EE. in a certain position.

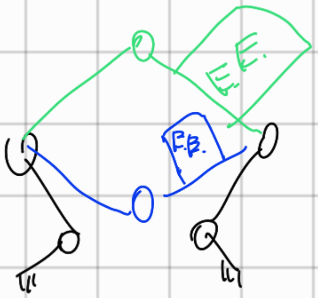


If you assign the EE you have 2 working modes:



Inverse position Kinematics because we have the position of EE.

If we assign the modes we get two $\boxed{\text{assembly modes}}$



this is forward position kinematics

Different ways of assemblying the robot given the position of some joints

TODO notes on vector spaces subsets