30|04|2023

${}^a\underline{\mu}_{b/a}$

${}^b\underline{\mu}_{b/a}$

Conosco inoltre che:

$${}^a\underline{\mu}_{b/a} = {}^a_bR\,{}^b\underline{\mu}_{b/a}$$

$\underline{\mu} \to \mu(t)$ Time Variant Vector Function

$$\boxed{\frac{d_a}{dt}{}^a\underline{\mu}(t) = \frac{d_b}{dt}{}^b\mu(t) + \underline{\omega}_{b/a} \times \mu(t)}$$

Mother of All Formulas

$$\frac{d}{dt}{}^a\underline{\mu} = {}^a_bR\,\frac{d}{dt}{}^b\underline{\mu} + \left[{}^b\underline{\omega}_{b/a}\times\right]\underbrace{{}^a\underline{\mu}(t)}_{{}^a_bR\,{}^b\underline{\mu}(t)}$$

La derivata di un vettore proiettato nel frame a è uguale alla derivata di quel frame in b più il prodotto vettoriale tra vettore e velocità angolare dei due frame di riferimento

The derivate of all therm of ${}^a\mu$ respect to frame a

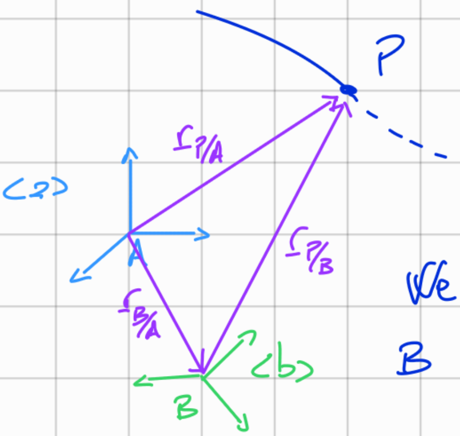
Strap down Equation

$${}^a_b\dot{R} = \left[{}^a\underline{\omega}_{b/a}\times\right]{}^a_bR$$

$${}^b_a\dot{R} = \left[{}^b\underline{\omega}_{a/b}\times\right]{}^b_aR$$

$$\underline{\omega}_{b/a} = -\underline{\omega}_{a/b}$$

$$\boxed{\underline{\omega}_{c/a} = \omega_{c/b} + \omega_{b/a}}$$



P is moving in space, we know the future trajectory

We want to know how P is seen by A and B

$$\underline{\rho}_{P/A} \triangleq (P-A); \quad \underline{\rho}_{P/B} \triangleq (P-B); \quad \underline{\rho}_{B/A} \triangleq (B-A)$$

$$\underline{\rho}_{P/A} = \underline{\rho}_{P/B} + \underline{\rho}_{B/A}; \quad {}^{a}_{b}R \triangleq \left[e_i^a \cdot e_j^b \right]$$

NOTA: Ricordo Inoltre $V = \omega \times r$

I want to know the velocity

$$\underbrace{\frac{d}{dt}\underline{\rho}_{P/A}}_{\underline{v}_{P/A}} = \frac{d_a}{dt}\left[\underline{\rho}_{P/B} + \underline{\rho}_{B/A}\right] = \underbrace{\frac{d_a}{dt}\underline{\rho}_{P/B}}_{\frac{d_b}{dt}\underline{\rho}_{P/B} + \underline{\omega}_{b/a} \times \underline{\rho}_{P/B}} + \underbrace{\frac{d_a}{dt}\underline{\rho}_{B/A}}_{\underline{v}_{B/A}} =$$

$$\underbrace{\frac{d_b}{dt}\underline{\rho}_{P/B}}_{\underline{v}_{P/B}} + \underline{\omega}_{b/a} \times \underline{\rho}_{P/B}$$

Galileo Theorem of Velocity Distribution

$$\rightarrow \boxed{\underline{v}_{P/A} = \underbrace{\underline{v}_{P/B}}_{\text{Velocità lin del I frame}} + \underbrace{\underline{\omega}_{b/a} \times \underline{\rho}_{P/B}}_{\text{Vel dovuta ad una rotazione del I frame}} + \underbrace{\underline{v}_{B/A}}_{\text{Linear motion of P respect to A}}}$$

↳ Velocità rispettiva dei 2 frames

$$\underline{v}_{P/A} = \underline{v}_{P/B} + \underline{\omega}_{b/a} + \underline{\rho}_{P/B} + \underline{\omega}_{B/a}$$

$$\underbrace{\frac{d_a}{dt}\underline{v}_{P/A}}_{\underline{a}_{P/A}} = \frac{d_a}{dt}\left[\underline{v}_{P/B} + \underline{\omega}_{b/a} \times \underline{\rho}_{P/B} + \underline{v}_{B/a}\right] =$$

$$= \underbrace{\frac{d_a}{dt}\underline{v}_{P/B}}_{\text{Applico rather of All Formulas}} + \underbrace{\frac{d_a}{dt}\left[\underline{\omega}_{b/a} \times \underline{\rho}_{P/B}\right]}_{\left(\frac{d_a}{dt}\underline{\omega}_{b/a}\right) \times \underline{\rho}_{P/B} + \underline{\omega}_{b/a} \times \overbrace{\left[\underline{v}_{P/B} + \underline{\omega}_{b/a} \times \underline{\rho}_{P/B}\right]}^{\frac{d_a}{dt}\underline{\rho}_{P/B}}} + \underbrace{\frac{d_a}{dt}\underline{v}_{B/A}}_{\underline{a}_{B/A}}$$

$$\underbrace{\frac{d_b}{dt}\underline{v}_{P/B}}_{\underline{a}_{P/b}} + \underline{\omega}_{b/a} \times \underline{v}_{P/B}$$

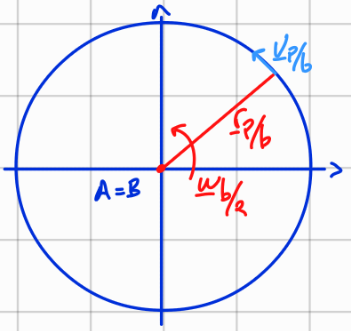
$$\underline{a}_{P/B} = \underline{a}_{P/b} + \underbrace{2\,\underline{\omega}_{b/a} + \underline{v}_{P/b}}_{\text{Coriolis. Acceleration}} + \left(\frac{d}{dt}\underline{\omega}_{b/a}\right) + \underline{r}_{P/B} + \underbrace{\underline{\omega}_{b/a} \times \left[\underline{\omega}_{b/a} \times \underline{r}_{P/B}\right]}_{\text{Centripetal Acceleration}} + \underline{a}_{B/A}$$

NOTE

$$\frac{d^a}{dt}\underset{\mu(t)}{\underline{\omega}_{b/a}} = \frac{d^b}{dt}\omega_{b/a} + \cancel{\omega_{b/a} \times \omega_{b/a}}_{\,0} \Rightarrow \frac{d^a}{dt}\underline{\omega}_{b/a} = \underline{\dot{\omega}}_{b/a}$$

$$\underline{a}_{P/a} = \underline{a}_{P/b} + \underbrace{2\underline{\omega}_{b/a} \times \underline{v}_{P/b}}_{\text{Coreolis Acc}} + \underline{\dot{\omega}}_{b/a} \times \underline{r}_{P/B} + \underbrace{\underline{\omega}_{b/a} \times (\underline{\omega}_{b/a} \times \underline{r}_{P/B})}_{\text{Centripetal Acceleration}} + \underline{a}_{B/a}$$

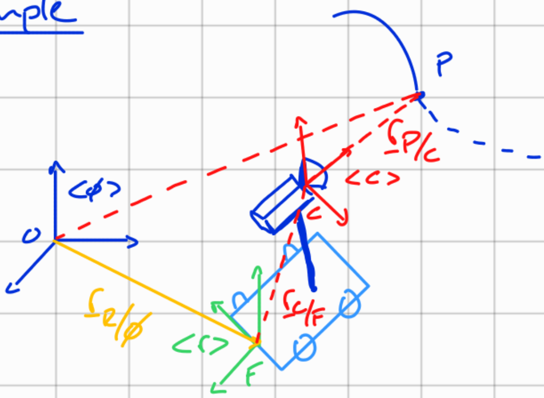
NOTE



$2\underline{\omega}_{b/a} \times v_{P/b} = \odot$

$\Rightarrow$ Sento una "perdita di peso"
Se vado dall'altra parte, $v_{P/b}$
è al contrario $\Rightarrow$ + peso

Example



RGBD Camera (Give me the distance)

We want to know:

$\underline{r}_{P/\phi}$
$\underline{v}_{P/\phi}$
$\underline{a}_{P/\phi}$ } "Smooth" time functions

Assume These quantity
Are known at $t = t_m$ (measuring Time)

Goal: "predict" $r_{P/\phi}$ for $t > t_m$

$$r_{P/\phi}(t) \cong \underline{r}_{P/\phi}(t_m) + \underline{v}_{P/\phi}(t_m)(t - t_m) + \frac{1}{2}a_{P/\phi}(t_m)(t - t_m)^2$$

Approximate by using II° order Taylor Formula

I CAN compute the line $r_{P/\phi}$ written as before

$${}^{\phi}r_{P/\phi} \cong {}^{o}r_{P/\phi} + {}^{o}v_{P/\phi}(t-t_m) + \frac{1}{2}\,{}^{\phi}a_{P/\phi}(t-t_m)^2$$

NOTE

We Assume that $(C, \langle c \rangle)$, Are Calibrated w.r.t. $(F, \langle r \rangle)$: ${}^{r}\underline{r}_{C/F}$: Known

${}^{r}_{c}R$: Known

1) $r_{P/\phi} = \underline{r}_{P/C} + \underline{r}_{C/F} + \underline{r}_{F/\phi}$

$${}^{o}_{c}R = {}^{\phi}_{F}R\;{}^{\phi}_{c}R$$

Known Known From

Calibration

2) $$\underline{v}_{P/\phi} = \frac{d_\phi}{dt}\,r_{P/\phi} = \frac{d_o}{dt}\left[\underline{r}_{P/C} + \underline{r}_{C/F} + \underline{r}_{F/\phi}\right] =$$

$$= \Big[\underbrace{\frac{d_c}{dt}\,\underline{r}_{F/\phi}}_{\underline{v}_{P/C}} + \underline{w}_{C/\phi} \times \underline{r}_{P/C}\Big] + \Big[\frac{d_r}{dt}\,\underline{r}_{C/F} + \underline{w}_{R/\phi} \times \underline{r}_{C/F}\Big] + \underline{v}_{F/\phi} =$$

$= \underline{w}_{C/r} + \underline{w}_{r/\phi}$

$$\boxed{= \underline{v}_{P/\phi} = \underline{v}_{P/C} + w_{r/\phi} \times (\underline{r}_{P/C} + \underline{r}_{C/F}) + \underline{v}_{F/\phi}}$$

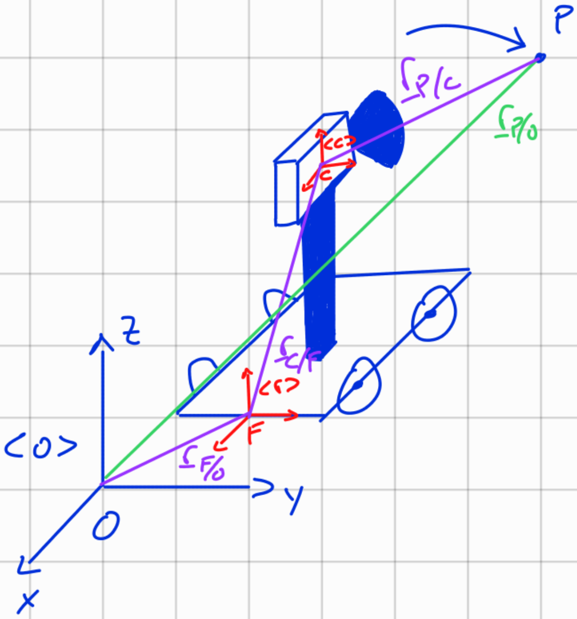
Velocity of P respect to ϕ

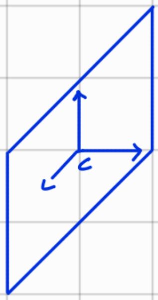
estimated by camera

How Write in Computation Way

$${}^{o}v_{P/\phi} = {}^{o}_{c}R\,{}^{c}\underline{v}_{P/C} + \left[{}^{\phi}\underline{w}_{C/\phi}\times\right]\left[{}^{o}_{c}R\,{}^{\phi}\underline{r}_{P/\phi} + {}^{o}_{c}R\,{}^{r}\underline{r}_{C/F}\right] + {}^{\phi}\underline{v}_{F/\phi}$$

3/Nov/2023

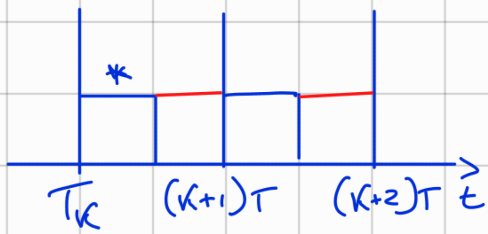




I need this numbers:

$$\begin{cases} {}^{F}r_{C/F} \\ {}^{F}_{C}R \end{cases}$$


```
for ( ; ; ) {
    read sensors
    compute errors
    compute controll
    send control to numbers
    < suspended >
}
```



→ Real Time Software for Control.

→ Foundamental for garantee a correct working of Controller

$r_{P/O}(t) = r_{P/O}(t_m) + v_{P/O}(t_m)(t-t_m) + \frac{1}{2} a_{P/O}(t_m)(t-t_m)^2$

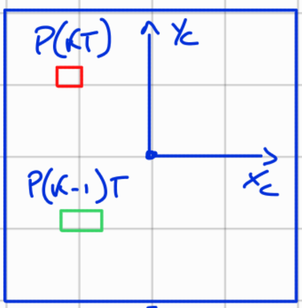
$r_{P/O}(t_m) = r_{C/F} + r_{F/O}$

Linear Velocity of point F respect to Frame (O) Is given

$\frac{{}^c d}{dt} r_{P/O} = v_{P/O}(t_m) = \underbrace{\frac{{}^c d}{dt} r_{P/C}}_{v_{P/C}} + \underbrace{\omega_{C/O}}_{\cancel{\omega_{C/F}} + \omega_{C/O}} \times r_{P/C} + \cancel{\frac{{}^c d}{dt} r_{C/F}} + \omega_{C/O} \times \underbrace{r_{C/F}}_{\text{Calibration parameters}} + v_{F/O}$

Measurable

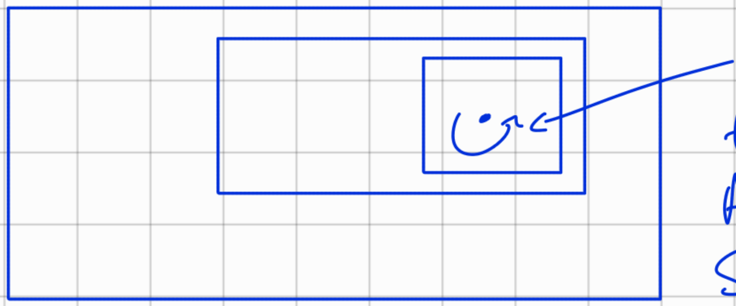
$\frac{{}^c d}{dt} r_{P/O} \triangleq v_{P/O} = v_{P/C} + \omega_{C/O} \times \left[r_{C/F} + r_{P/C} \right] + v_{F/O}$



$a_{P/O}(t_m)$

$v_{P/C} = \frac{1}{T}\left[P(KT) - P[(K-1)T]\right]$

${}^c\omega_{C/O}$



This Chip Give The the rotation around the Axis coming out of Screen

$a_{P/O}(t_m) = \frac{{}^c d}{dt} v_{P/O} = \underbrace{\frac{{}^c d}{dt} v_{P/C}}_{a_{P/C}} + \omega_{C/O} \times v_{P/C} + \omega_{C/O} \times r_{P/F} + \omega_{C/O} \times \left[\frac{{}^c d}{dt} r_{P/F} + \omega_{C/O} \times r_{P/F}\right] + a_{F/O}$

Known At camera level ${}^c a_{P/C}$

$\omega_{C/O}$ probably known in frame C: ${}^c\omega_{C/O}$

$v_{P/C}$ known in frame C ${}^c v_{P/C}$

${}^c r_{P/C}$

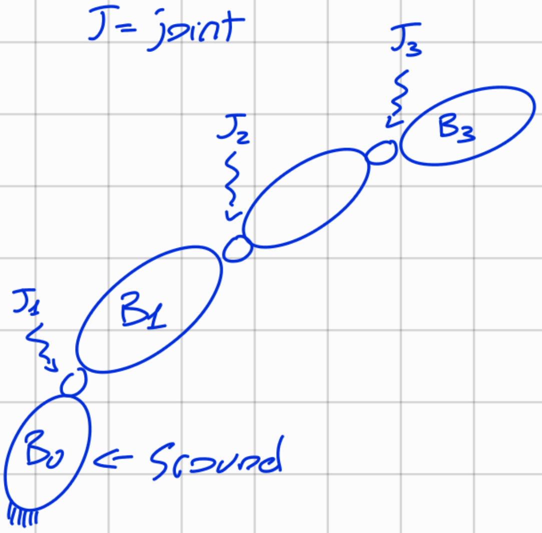
${}^c r_{C/F}$

After having Calculation, need to Convert All to frame F

Kinematic Chains

Holonomic Robot
-> Simpler Connection of rigid bodies

J = joint



B_i is rigid body.

Types of Joint

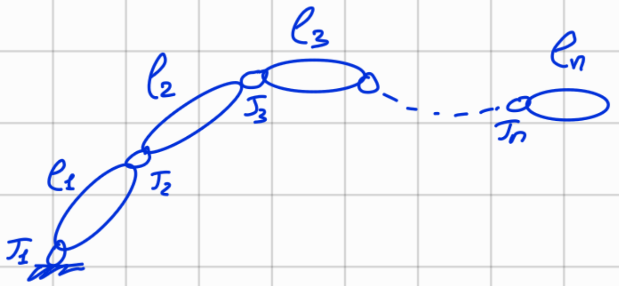
-> Rotational Joint

1D q_i = Joint variable,
(Movement Around one Single Axes) correspond to joint Angle (is An Angle)

-> Prismatic Joint

1D q_i = Joint Variable
Joint displacement

6/nov/2023



Joint reduce the mobility of each joint, becase they link to the before or next link.

Rotation Angle   Axis of Rotation

- Rotational Joint (1 D)
- Prismatic Joint (1 D)

$\left\{ (q_i, \underline{k}_i) \right.$

Joint Variable

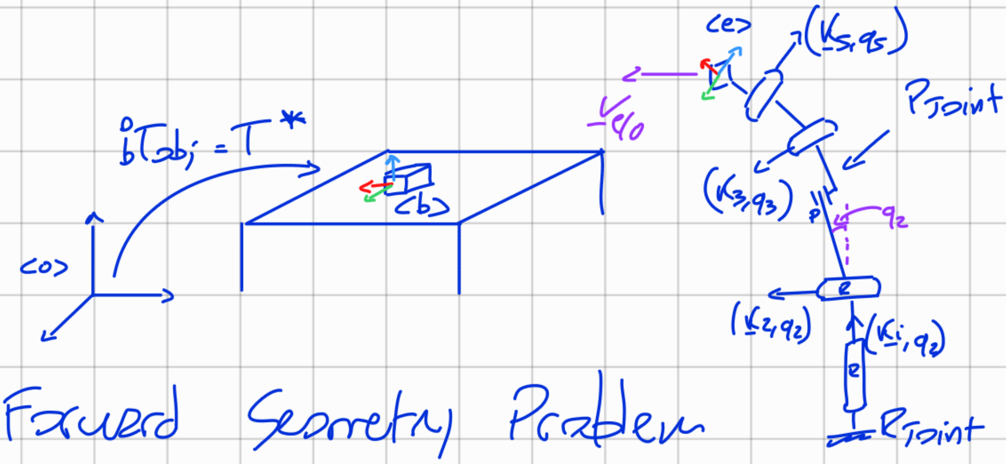
1) **Forward Geometry** -> Only The Geometry

-> Set of procedure that given $(q_i, \underline{k}_i)$ make possible how this body is displaced in Space

2) **Forward Kinematic** -> The velocity of the Bodies

-> Specify what is the motions of All the bodyes given $(q_i, \underline{k}_i)$ And the time derivative $\frac{d}{dt}(q_i, \underline{k}_i)$

es:



1) Forward Geometry Problem

$$ {}^0_eT = \left[\begin{array}{c|c} {}^0_eR & {}^0r_{e/0} \\ \hline \underline{e}^T & 1 \end{array} \right] = {}^0_eT(q_1, q_2, q_3, q_4, q_5) $$

$$\underline{q} \triangleq \begin{bmatrix} q_1 \\ q_2 \\ \vdots \\ q_n \end{bmatrix}$$

Assume $q_1, \dots, q_n$ Independent, I CAN chose the value As I want

Joint Variabile vector

$${}^0_eT = {}^0_eT(\underline{q})$$ ← I need An Algoritmic Method for Compute This Calculation

-> Inverse Geometric problem

$${}^0_bT_{obj} = T^*$$

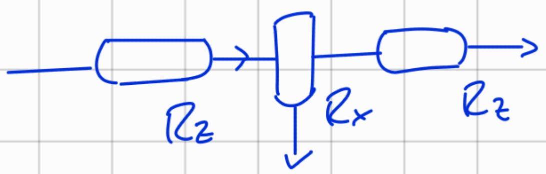
$$T^* = {}^0_eT(\underline{q}) \quad (\bullet)$$

Need to Compute q Knowing •
-> Could exist Solution
-> " " many Solutions
-> " " NO Solution

<u>NOTE</u>

YAW, pitch, roll $(\theta, \varphi, \psi) \longrightarrow R = R_z(\theta)R_x(\varphi)R_z(\psi)$

es: Industrial wrist



2) Forward Kinematic Problem

$$\begin{bmatrix} {}^{0}\underline{\omega}_{e/0} \\ {}^{0}\underline{v}_{e/0} \end{bmatrix} = f(\underline{q}, \dot{\underline{q}}) = J\dot{\underline{q}}$$

Configuration of Robot

Velocity of Joints

$J_e = J_e(q)$ ← Inverse Kinematic Problem

Jacobian Matrix

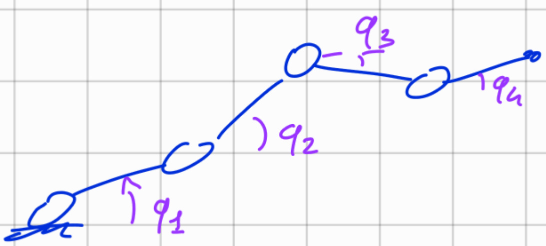
Arrive to a point setting the Angle of rotation of All Joint

→ $\underline{q}$ Speak About Geometry

→ $q, \dot{q}$ Speak About Velocity

→ $q, \dot{q}, \ddot{q}$ Speak About Acceleration

Type of Kinematic Chains

- 

Serial open KC

n dof

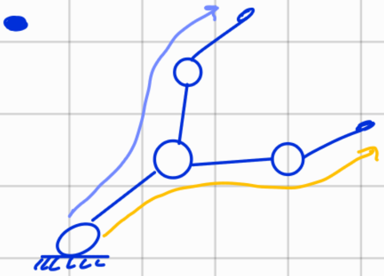
- 

Serial Closed Loop KC

Joint Angle Are not more Indipendent, $\underline{q}$ Are NOT indipendent

$m < n$ ← numb. of Joints

dof

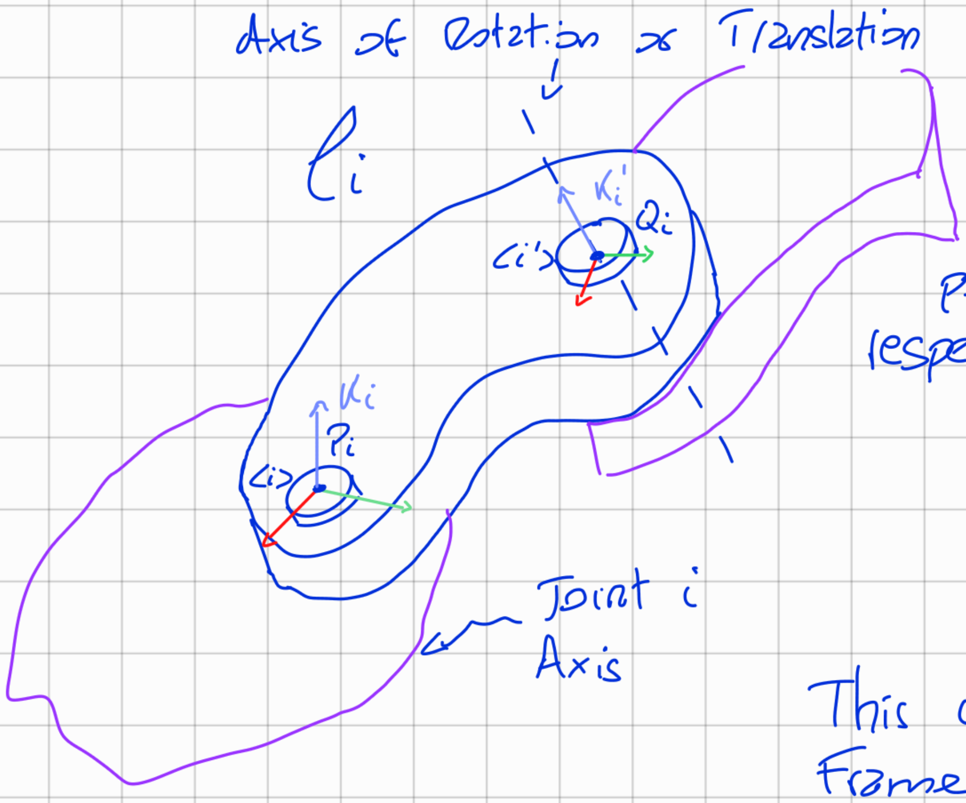


Tree-Structures KC

or Branch KC

ℓ_0 = link$_0$ = ground, has NO Father
every other Link have a Father
ℓ_i has A father ℓ_{i-1}

Every link ℓ_i could have a motion, rot or translation respect his father



k_i' have a specific position And rotation respect to k_i

This is expressed in Frame $\langle i \rangle$

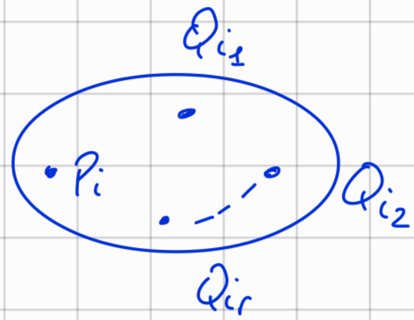
$$ {}^{i}\underline{r}_{Q_i/P_i} \triangleq \underline{r}_i \quad \text{Constant} $$

$$ {}^{i}_{i'}R \triangleq R_i \quad \text{Constant} $$

$$ \Rightarrow T_i = \left[\begin{array}{c|c} R_i & \underline{r}_i \\ \hline \underline{0}^T & 1 \end{array}\right] $$

NOTE

Father With Multiple Child



$P_{i+1} \equiv Q_i$ if The Joint $i+1$ is Rotational

is Prismatic And $q_{i-1} = 0$

This Frame Are Coincident

$K_{i+1} \equiv K_i'$

So $\langle i+1 \rangle \equiv \langle i' \rangle$ if the Joint $\langle i+1 \rangle$ is prismatic

We will Consider only rotation Around z Axis

$K_{i+1} = i_i'$ $\quad {}^{2}_{b}\dot{R} = \left[{}^{2}\underline{\omega}_{b/2} \times \right] {}^{2}_{b}R$

↙ Simbolic Vector, store the Type of the Joint I have

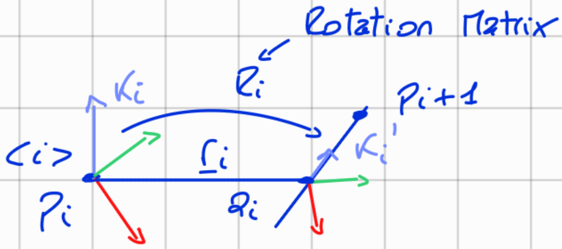
$$\Gamma = \left[\quad * \quad \right]$$

n-elements

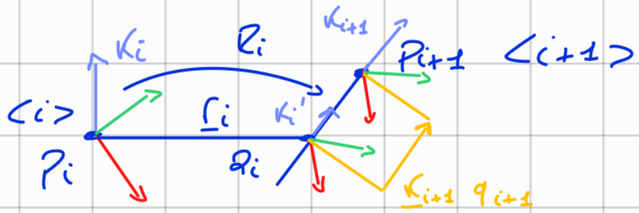
$$* = \begin{cases} R \\ P \end{cases} \Rightarrow \begin{aligned} &\Gamma_i = R = \text{rotational Joint} \\ &\Gamma_i = P = \text{Prismatic Joint} \end{aligned}$$

The Robot we saw before is

$$\Gamma = [R, R, P, R, R]$$



If $\Gamma_{i+1} = P$



We tell where P_{i+1} is placed with respect to P_i

$$\boxed{r_{i+1/i} \equiv r_{P_{i+1}/P_i} = r_i + k_{i+1}\, q_{i+1}}$$

Semplification of Notation

$${}^i r_{i+1/i} = {}^i r_i + {}^i k_{i+1}\, q_{i+1}$$

$= {}^i k_i'$ → Known Value

MOTE

$${}^{i+1}k_{i+1} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

vector k_{i+1} respect to frame <i+1>

$${}^{i}k_i' = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

$${}^{i}k_i' = R_i \, {}^{i'}k_i'$$

This Value is Knowing Because il is pre-compute

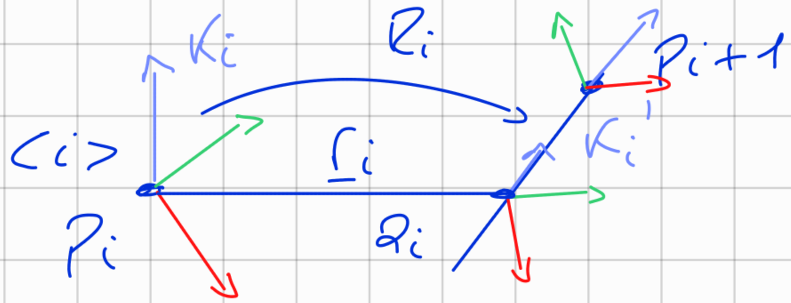
$$R_i \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} * \\ * \\ * \end{bmatrix}$$

$$3\times3 = \begin{bmatrix} * & * & * \\ * & * & * \\ * & * & * \end{bmatrix}$$

$* =$ number

$${}^{i}_{i+1}R = R_i$$

If $r_{i+1} = R$



Rotation Around z-Axes Around the k_i' Axes

$$r_{i+1/i} = r_i$$

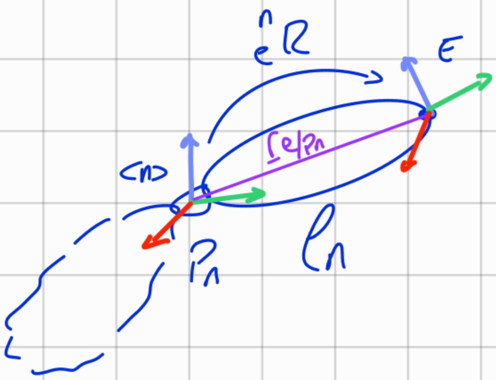
$${}^{i}_{i+1}R = R_i \, R_z(q_{i+1})$$

Angle

$$R_z(\alpha) = \begin{bmatrix} \cos\alpha & -\text{sen}\,\alpha & 0 \\ \text{sen}\,\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\underline{r}_{i/i-1} = \begin{cases} \underline{r}'_{i/i-1} & \text{if } \Gamma_i = R \\ \underline{r}'_{i/i-1} + \underline{k}_i q_i & \text{if } \Gamma_i = P \end{cases}$$

$$^{i+1}_{\ i}R = \begin{cases} R_i\, R_z(q_i) & \text{if } \Gamma_i = R \\ r_{i-1} & \text{if } \Gamma_i = P \end{cases}$$



P_n is the last point describe the geometry of the robot without considering the task of the Robot

Robot Forward Geometry (Complete)

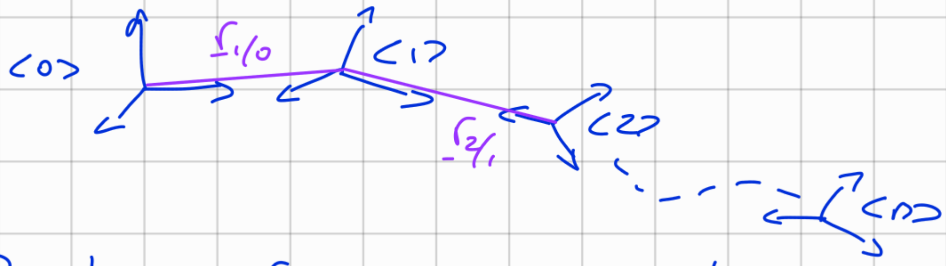
P_0, Q_0 $\quad$ Q_0 = where my robot is Attached

P_1, Q_1 $\quad$ P_0 = Point where there is ground reference Frame

⋮

P_{n-1}, Q_{n-1}

$P_n, (E)$



Position of <n> respect to <0>

$$\underline{r}_{n/0} = \underline{r}_{1/0} + \underline{r}_{2/1} + \underline{r}_{3/2} + \dots + \underline{r}_{n/n-1}$$

$${}^{0}_{n}R = {}^{0}_{1}R\,{}^{1}_{2}R\,{}^{2}_{3}R + \dots + {}^{n-1}_{n}R$$

$${}^{0}_{1}T = \left[\begin{array}{c|c} {}^{0}_{1}R(p_1) & {}^{0}\underline{r}_{1/0} \\ \hline 0\;0\;0 & 1 \end{array}\right]$$

$$\boxed{{}^{i-1}_{i}T = \left[\begin{array}{c|c} {}^{i-1}_{i}R(q_i) & {}^{i-1}\underline{r}_{i/i-1} \\ \hline 0\;0\;0 & 1 \end{array}\right] \Rightarrow {}^{0}_{n}T = \prod_{i=1}^{n} {}^{i-1}_{i}T(q_i)}$$

```
for(i=0; i<=n; i++){
    T_i_i-1 = ---
    T *= T_i_i-1;
}
```

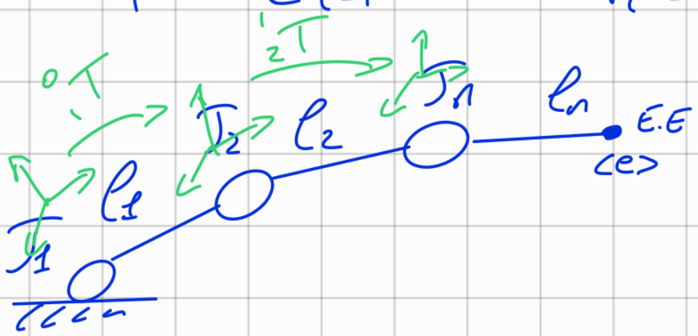
10/nov/2023

Convention

$$\underline{r}_{i/i-1} = \begin{cases} \underline{r}_{i/i-1} & \Gamma_i = R \\ \underline{r}_{i/i-1} + \underline{k}_i q_i & \Gamma_i = P \end{cases}$$

$k_i = k'_{i-1}$

$$^{i-1}_{i}R = \begin{cases} R_{i-1}\, R_z(q_i) & \Gamma_i = R \\ R_{i-1} & \Gamma_i = P \end{cases}$$



$$^{i-1}_{i}T = \left[\begin{array}{c|c} ^{i-1}_{i}R & ^{i-1}\underline{r}_{i/i-1} \\ \hline \underline{0}^T & 1 \end{array}\right] = {}^{i-1}_{i}T(q_i)$$

The only quantity can change is q_i

$$R_z(a) = \begin{bmatrix} ca & -sa & 0 \\ sa & ca & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$^{i}\underline{k}_i = {}^{i}k_{i-1} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \Rightarrow {}^{i-1}k'_{i-1} = R_{i-1}\, {}^{i}k'_{i-1}$$

$$ {}^{0}_{n}T = {}^{0}_{1}T \cdot {}^{1}_{2}T \cdots {}^{n-2}_{n-1}T \; {}^{n-1}_{n}T = {}^{0}_{n}T(q) $$

Is compute numerically

$$ \left({}^{n}r_{e/n} \quad {}^{n}_{e}R \right) \to {}^{n}_{e}T = \left[\begin{array}{c|c} {}^{n}_{e}R & {}^{n}r_{e/n} \\ \hline 0^T & 1 \end{array} \right] $$

Specify How EE is Attached At last frame

$$ \Rightarrow {}^{0}_{E}T = {}^{0}_{n}T \; {}^{n}_{E}T $$

q

Structure of the robot

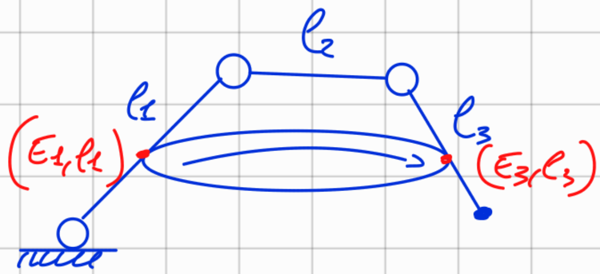
Transform Matrix of EE respect to the last link

<u>REMARK</u> EE Attached to link i

$$ {}^{0}_{E}T = {}^{0}_{i}T(q_1 \ldots q_i) \; {}^{i}_{E}T $$

Cut the chain At interest point

<u>Example</u>



Need to know:

$$ \left({}^{1}r_{E_1/1} , {}^{1}_{\ell_1}R \right) $$

$$ \left({}^{3}r_{E_3/3} , {}^{3}_{\ell_3}R \right) $$

Want to Compute: This bar stands for Homogeneous Cordinate

$$ {}^{\phi}\bar{\underline{r}}_{\epsilon_3/\epsilon_1} = {}^{\phi}\bar{\underline{r}}_{\epsilon_3/\phi} - {}^{\phi}\bar{r}_{\epsilon_1/\phi} = {}^{0}_{3}T \; {}^{3}\bar{\underline{r}}_{\epsilon_3/3} - {}^{0}_{1}T \; {}^{1}\bar{\underline{r}}_{\epsilon_1/1} $$

($\bar{\underline{r}}$ is homogeneous vector, I can't put directly to T

$\hookrightarrow \bar{\underline{r}} = \begin{bmatrix} x \\ x \\ x \\ 1 \end{bmatrix}$ x = component)

$$ {}^{0}_{\epsilon_3}R = {}^{0}_{\epsilon_1}R \; {}^{\epsilon_1}_{\epsilon_3}R $$

← Unknown Matrix

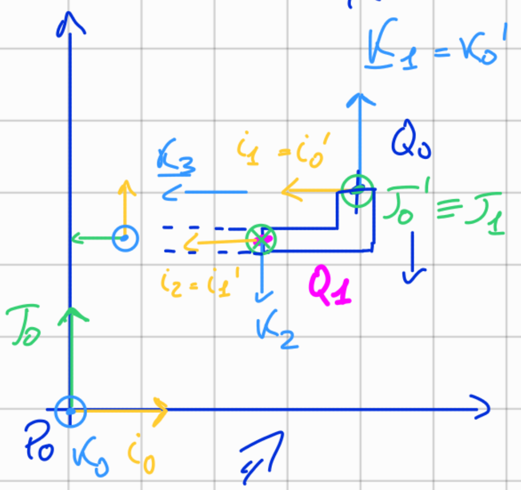
$$ \Rightarrow {}^{\epsilon_1}_{\epsilon_3}R = {}^{0}_{\epsilon_1}R^T \; {}^{0}_{\epsilon_3}R $$

$$ {}^{0}_{\epsilon_1}R^T = \left({}^{0}_{1}R \cdot {}^{1}_{\epsilon_1}R \right) \qquad {}^{0}_{3}R \; {}^{3}_{\epsilon_3}R $$

REMARK

$$ \underline{(P_i, Q_i)} $$

$$ {}^{i}\underline{r}_{i/i-1} $$



$$ {}^{0}r_{0'/0} = \begin{pmatrix} 20 \\ 10 \\ 6 \end{pmatrix} $$

$$ R_0 = \begin{pmatrix} -1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix} $$

$$ {}^{0}r_{1/0} = {}^{0}r_{0'/0} $$

$$ {}^{0}_{1}R = R_{\phi} R_z(q_1) $$

Model for Joint 1

Model For Link ϕ

Model For Link 1

(P_1, Q_1)

$${}^{1}r_{1'/1} = \begin{pmatrix} 2 \\ 0 \\ 1 \end{pmatrix} \qquad R_1 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{pmatrix}$$

$$\begin{cases} {}^{1}r_{2/1} = r_{1'/1} \\ {}^{1}_{2}R = R_1 \, R_z(q_2) \end{cases}$$

Model For Joint 2

Model For Link 2

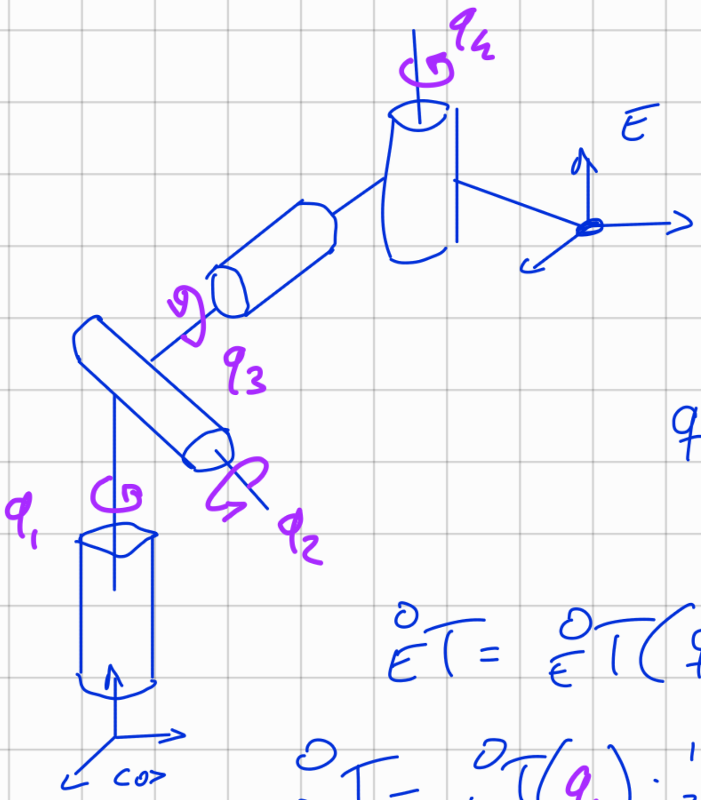
(P_2, Q_2)

I decide that Axes Are Coincident Are Apply in same point

$$\begin{cases} {}^{2}r_{2'/2} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \\ {}^{1}_{2}R = R_2 \, R_z(q_2) \end{cases}$$

$$R_2 = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ -1 & 0 & 0 \end{pmatrix}$$

20/nov/2023



$$q = \begin{bmatrix} q_1 \\ q_2 \\ \vdots \\ q_n \end{bmatrix} \in \mathbb{R}^n \qquad n \geq 6$$

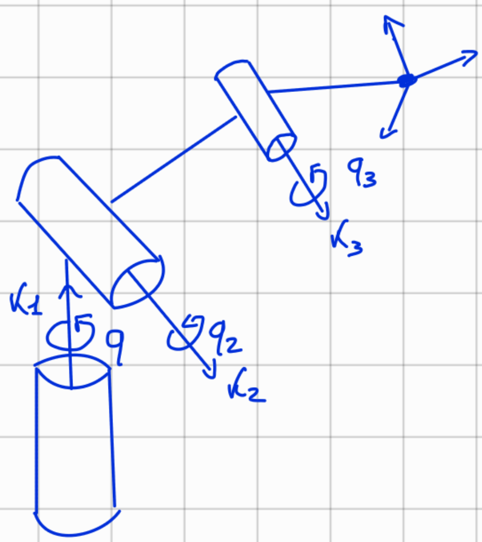
$${}^0_E T = {}^0_E T(q)$$

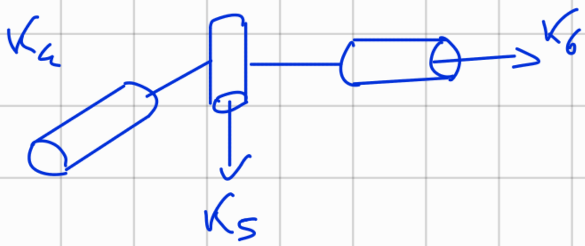
$${}^0_e T = {}^0_1 T(q_1) \cdot {}^1_2 T(q_2) \cdots {}^{n-1}_n T(q_n)\, {}^n_E T$$

Forward Geometry Problem

I Assign a value to q and find the position of E.E.

Simple rappresentation of Jawl, pich, roll





$${}^0_n T = {}^0_1 T\, {}^1_2 T \cdots {}^{n-1}_n T$$

Inverse Geometry Is difficult. Has solution only in some special Case

Redondant robot: have more Joint than the DOF needed for EE. We consider robot with DOF ≥ 6

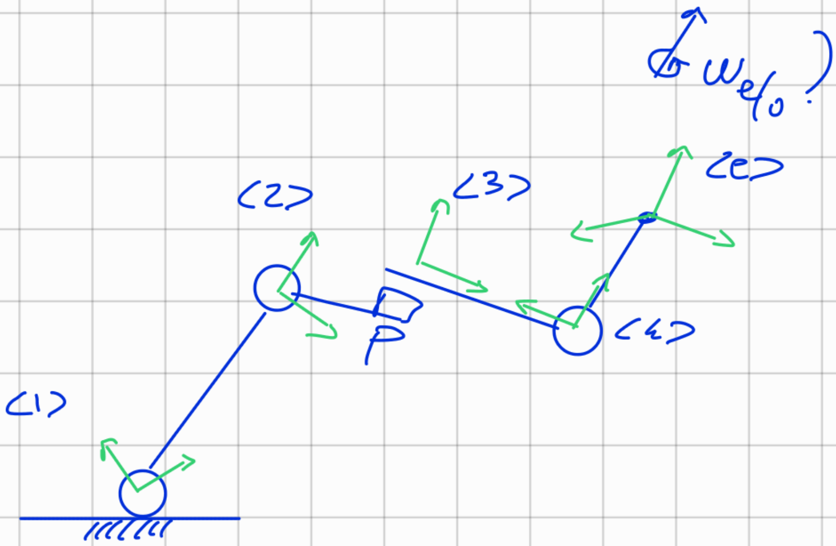
If robot is redondand there are much more then one solutions for Inverse Geometry

Robot Kinematics

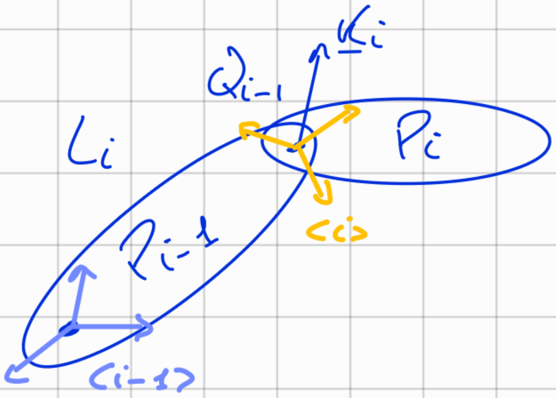
Joint variable

If I change q_i how EE move?

INPUT: Joint position, Joint velocity



At time t I have this configuration
-> Need to find All Joint velocity

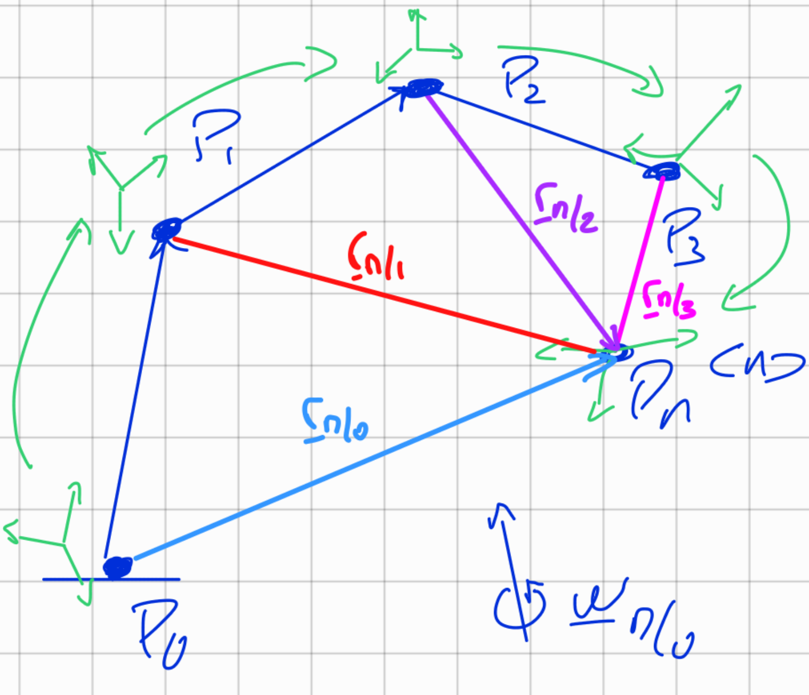


* Need To Compute: $P_i = Q_{i-1}$

$$\underline{v}_{i/i-1} \triangleq \frac{d_{i-1}}{dt}(P_i - P_{i-1}) = \begin{cases} \underline{\emptyset} & \text{if } \Gamma_i = R \\ \underline{k}_i \dot{q}_i & \text{if } \Gamma_i = P \end{cases}$$

$$\underline{\omega}_{i/i-1} = \begin{cases} \underline{k}_i \dot{q}_i & \text{if } \Gamma_i = R \\ \underline{\emptyset} & \text{if } \Gamma_i = P \end{cases}$$

Want to compute $\omega_{n/0}$



I refer To This example for the next Computation

$$\underline{\omega}_{n/0} = \underline{\omega}_{n/n-1} + \underline{\omega}_{n/n-2} + \cdots + \underline{\omega}_{n/2} + \underline{\omega}_{n/1}$$

- $$\underline{\omega}_{n/0} = \underset{\underline{J}_1^A \dot{q}_1}{\underline{\omega}_{1/0}} + \underset{J_2^A \dot{q}_2}{\underline{\omega}_{2/1}} + \cdots + \underset{J_n^A \dot{q}_n}{\underline{\omega}_{n/n-1}} = \sum_{i=1}^{n} J_i^A \dot{q}_i$$

Linear Combinations of Vector

J_1^A = is A function with inside the if, referred to previous functions

$$J_i^A = \begin{cases} \underline{k}_i & \text{if } \Gamma_i = R \\ \phi & \text{if } \Gamma_i = P \end{cases}$$

Now I compute the $\omega_{n/0}$ referred to one reference frame, for example the first:

$$^\phi\omega_{n/\phi} = \sum_{i=1}^{n} {}^\phi J_i^A \dot{q}_i = \underbrace{\left[{}^0\underline{J}_1^A \,\vdots\, {}^0\underline{J}_2^A \,\vdots\, \cdots \,\vdots\, {}^0\underline{J}_n^A \right]}_{{}^\phi J_n^A \in \mathbb{R}^{3\times n}} \underbrace{\begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \vdots \\ \dot{q}_n \end{bmatrix}}_{\dot{\underline{q}}}$$

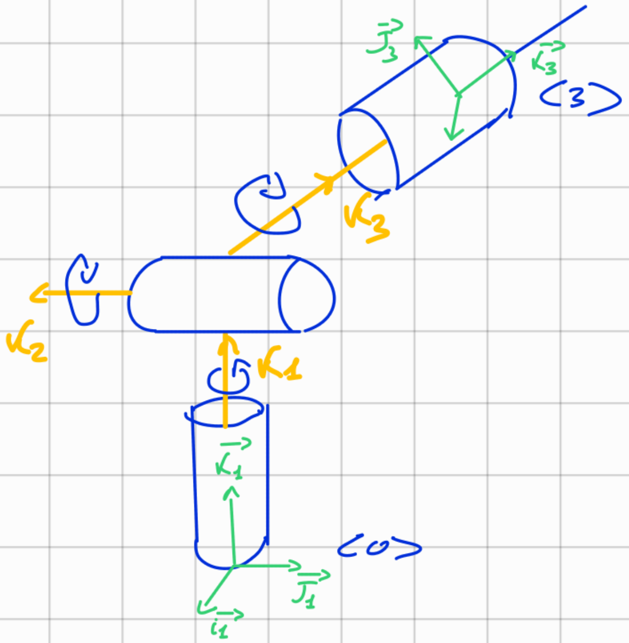
$^\phi J_n^A$ = Angular velocity Jacobian Matrix

If I want frame <n> want to touch All the position on plane, need 3 R joint NOT coplanar or parallel

$$^\phi\omega_{n/0} = {}^0J^A(q)\,\dot{\underline{q}}$$

Example

$q_1 = \phi$
$q_2 = \pi/4$
$q_3 = \phi$



$$J_i^A = \begin{cases} \underline{k}_i & \text{if } \Gamma = R \\ \phi & \text{if } \Gamma = P \end{cases}$$

$$\underline{w}_{3/0} = \underline{w}_{1/0} + \underline{w}_{2/1} + \underline{w}_{3/2} = \underline{J}_1^A \dot{q}_1 + J_2^A \dot{q}_2 + J_3^A \dot{q}_3 =$$

$$\underline{k}_1 \dot{q}_1 + \underline{k}_2 \dot{q}_2 + \underline{k}_3 \dot{q}_3$$

In This Case k_1, k_2, k_3 Are linear independent.

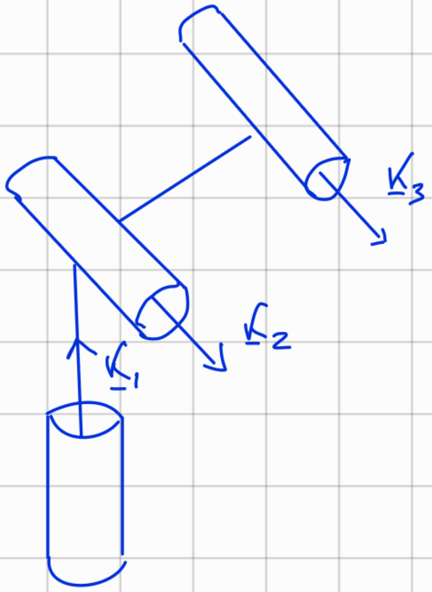
$$\Rightarrow w_{3/0} = \left[{}^0\underline{k}_1 , {}^\phi\underline{k}_2 , {}^\phi\underline{k}_3 \right] \dot{q} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & * \\ 1 & 0 & * \end{bmatrix} \dot{q}$$

I compute with calculate ${}^2_1T$

(This Matrix is invertible, * Are different to zero

$$\underline{v}_{n/0} \triangleq \frac{d_0}{dt}(P_n - P_0) = \frac{d_0}{dt} \underline{r}_{n/0}$$

$\underline{k}_2 \parallel \underline{k}_3$



$${}^0 J^A_{3/0} = \left[\underline{k}_1^{\,0} \;\; \underline{k}_2^{\,0} \;\; \underline{k}_3^{\,0} \right]$$

$$\mathrm{rank}\left({}^0 J^A_{3/0} \right) = 2$$

$$\underline{v}_{n/0} = \frac{d_0}{dt} \underline{r}_{n/0} = \frac{d_0}{dt} \left[\underline{r}_{1/0} + \underline{r}_{n/1} \right] =$$

$$\underbrace{\frac{d_0}{dt} \underline{r}_{1/0}}_{\underline{v}_{1/0}} + \frac{d_0}{dt} \underline{r}_{n/1} =$$

Applico Patter of All formulas

$$\frac{d_1}{dt} \underline{r}_{n/1} + \underline{\omega}_{1/0} \times \underline{r}_{n/1}$$

$$= \underbrace{\left(\underline{v}_{1/0} + \underline{\omega}_{1/0} \times \underline{r}_{n/1} \right)}_{*} + \frac{d_1}{dt} \underline{r}_{n/1} =$$

I'll do Another step

expand this term

$$(*) + \frac{d_1}{dt} \left[\underline{r}_{2/1} + \underline{r}_{n/2} \right] = \underline{v}_{2/1} + \frac{d_1}{dt} \underline{r}_{n/2} =$$

$$\frac{d_1}{dt} \underline{r}_{n/2} + \underline{w}_{2/1} \times \underline{r}_{n/2}$$

$$= \left(\underline{v}_{1/0} + \underline{\omega}_{1/0} \times \underline{r}_{n/1}\right) + \left(\underline{v}_{2/1} + \underline{\omega}_{2/1} \times \underline{r}_{n/2}\right) + \frac{d_2}{dt} r_{n/2} =$$

I expand All the term:

$$\underline{v}_{n/0} = \left(\underline{v}_{1/0} + \underline{\omega}_{1/0} \times \underline{r}_{n/1}\right) + \left(\underline{v}_{2/1} + \underline{\omega}_{2/1} \times \underline{r}_{n/2}\right) + \dots + \left(\underline{v}_{n/n-1} + \underline{\omega}_{n/n-1} \times \underline{r}_{n/n}\right) =$$

$$= \sum_{i}^{n} \underline{v}_{i/i-1} + \underline{\omega}_{i/i-1} \times \underline{r}_{n/i} \triangleq \sum_{i=1}^{n} J^{L}_{n/i} \dot{q}_i$$

$$J^{L}_{i} = \begin{cases} \underline{k}_i \text{ → only Translation} & \text{if } \Gamma = P \\ \underline{k}_i \times r_{n/i} & \text{if } \underline{\Gamma = R} \end{cases}$$

The result speed is the vectorial product of the rotation Axis and distance vector beteuen the two Joint

$${}^{0}\underline{v}_{n/0} = \underbrace{\left[{}^{0}J^{L}_{n/1} \;\vdots\; {}^{0}J^{L}_{n/2} \;\vdots\; \cdots \;\vdots\; {}^{0}J^{L}_{n/n} \right]}_{} \dot{q}$$

${}^{0}J^{L}_{n/0} \in \mathbb{R}^{3 \times n}$ Linear Jacobian Matrix

$$\boxed{\begin{bmatrix} {}^{0}\underline{\omega}_{n/0} \\ {}^{0}\underline{v}_{n/0} \end{bmatrix} = \begin{bmatrix} {}^{0}J^{A}_{n/0} \\ {}^{0}J^{L}_{n/0} \end{bmatrix} \dot{q} \triangleq {}^{0}J_{n/0}\, \dot{q}}$$

Twist

${}^{0}J_{n/0} \in \mathbb{R}^{6 \times n}$

Basic Robot Jacobian

Riassunto

$$\underline{J}_i^A = \begin{cases} \underline{k}_i & \text{if } \Gamma_i = R \\ \phi & \text{if } \Gamma_i = P \end{cases}$$

$${}^{\phi}w_{n/\phi} = \sum_{i=1}^{n} {}^{\phi}\underline{J}_i^A \dot{q}_i = \underbrace{\left[{}^0\underline{J}_1^A \,\vdots\, {}^0\underline{J}_2^A \,\vdots\, \cdots \,\vdots\, \underline{J}_n^A \right]}_{{}^{\phi}J_n^A \in \mathbb{R}^{3\times n}} \underbrace{\begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \vdots \\ \dot{q}_n \end{bmatrix}}_{\dot{\underline{q}}}$$

${}^{\phi}J_n^A$ = Angular velocity Jacobian Matrix

$$J_i^L = \begin{cases} \underline{k}_i & \text{if } \Gamma = P \\ \underline{k}_i \times \underline{r}_{n/i} & \text{if } \Gamma = R \end{cases}$$

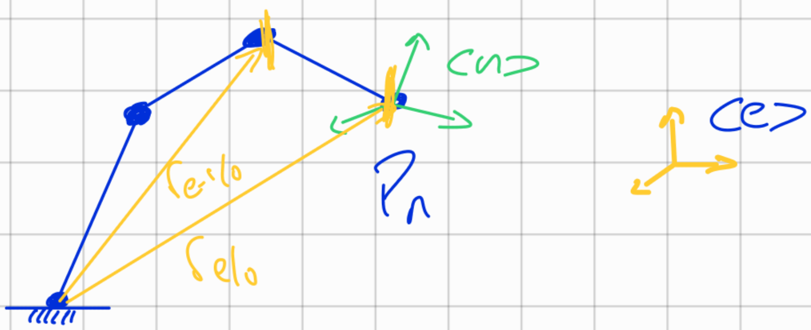
Distance Between the Last Joint/Tool and the i-esimo Joint Considered

$${}^{\phi}v_{n/0} = \underbrace{\left[{}^{\phi}J_{n/1}^L \,\vdots\, {}^{\phi}J_{n/2}^L \,\vdots\, \cdots \,\vdots\, {}^{\phi}J_{n/n}^L \right]}_{{}^{\phi}J_{n/0}^L \in \mathbb{R}^{3\times n}} \dot{\underline{q}}$$

Linear Jacobian Matrix

$$\underbrace{\begin{bmatrix} {}^{\phi}\underline{w}_{n/0} \\ {}^{\phi}\underline{v}_{n/0} \end{bmatrix}}_{\text{Twist}} = \begin{bmatrix} {}^0J_{n/0}^A \\ \hline {}^0J_{n/0}^L \end{bmatrix} \dot{\underline{q}} \triangleq \underbrace{{}^0J_{n/0}}_{\in \mathbb{R}^{6\times n}} \dot{\underline{q}}$$

Basic Robot Jacobian



$$\underline{\omega}_{e|0} = \underline{\omega}_{n|0}$$

$$\underline{v}_{e|0} = \frac{d_0}{dt}\left[\underline{r}_{n|0} + \underline{r}_{e|n}\right] = \frac{d_0}{dt}\underline{r}_{n|0} + \frac{d_0}{dt}\underline{r}_{e|n} =$$

The reciprocal is Antisymmetric

$$-\underline{r}_{n|0} \times \underline{\omega}_{n|0}$$

$$= \underline{v}_{n|0} + \overset{\phi}{\cancel{\frac{d_n}{dt}\underline{r}_{e|n}}} + \underline{\omega}_{n|0} \times \underline{r}_{n|0}$$

$$^0\underline{\omega}_{e|0} = \begin{bmatrix} I_{3\times3} & \phi \end{bmatrix} \begin{bmatrix} ^0\underline{\omega}_{n|0} \\ ^0\underline{v}_{n|0} \end{bmatrix}$$

$$^0\underline{v}_{e|0} = \begin{bmatrix} \left[^0\underline{r}_{e|n}\times\right]^T & I_{3\times3} \end{bmatrix} \begin{bmatrix} ^0\underline{\omega}_{n|0} \\ ^0\underline{v}_{n|0} \end{bmatrix}$$

$$\begin{bmatrix} ^0\underline{\omega}_{e|0} \\ ^0\underline{v}_{e|0} \end{bmatrix} = {}^0S_{e|n}\; {}^0J_{n|0}\; \underline{\dot{q}}$$

Case when I have An end effector or Tool

Rigid Body Jacobian of point e in body n

$$\begin{bmatrix} I_{3\times3} & \phi_{3\times3} \\ \left[^0\underline{r}_{e|n}\times\right]^T & I_{3\times3} \end{bmatrix} \in \mathbb{R}^{6\times6}$$

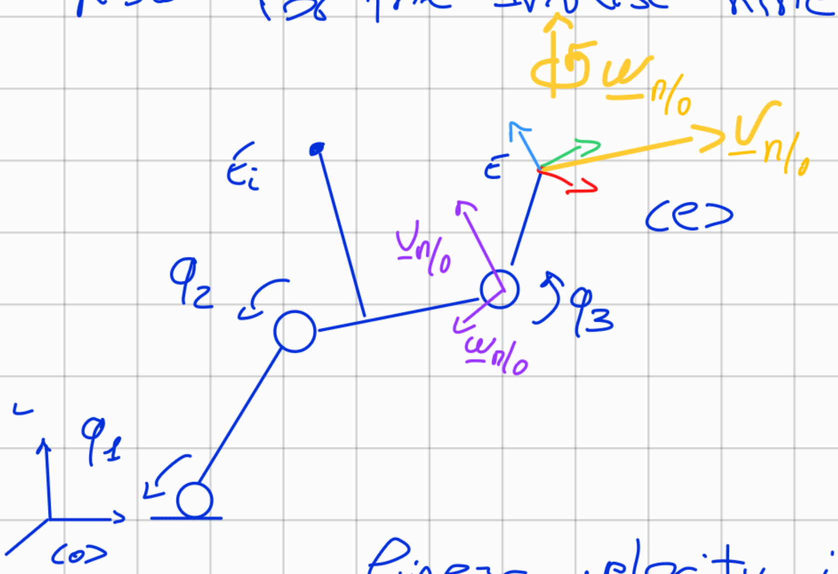
$^0r_{e|n}$ = Distance Between the last link and the e-e, Projected on <0>

24/nov/2023

Now we have the $\underline{\omega}_{n/0}$ and $\underline{v}_{n/0}$ (linear and Angular Velocities) of the e-e respet frame <0>

Now For the Inverse kinematics we want to find

To find $\dot{q}$ in function of $\underline{\omega}_{n/0}$ and $\underline{v}_{n/0}$



$\dot{q}$ = linear velocity if $\Gamma = P$ / speed of motor if $\Gamma = R$

Joint Velocity

$$\dot{q} \longrightarrow \begin{pmatrix} \underline{\omega} \\ \underline{v} \end{pmatrix}$$

Transformation
Linear Transformation

$$\mathbb{R}^n \longrightarrow \mathbb{R}^6$$

$n \geq 6$

n = number of Joint

$$\begin{pmatrix} \underline{\omega} \\ \underline{v} \end{pmatrix} = J \dot{q}$$

Jacobian

$J = J(q)$

Is A function and depend of the robot Configuration

If Configuration change
=> $J(q)$ change

${}^0J_{n/0}$ basic Jacobian

$${}^0J_i^A = \begin{cases} \underline{0} & \Gamma_i = P \\ {}^0\underline{k}_i & \Gamma_i = R \end{cases}$$

$${}^0J_{n/0} = \left[\begin{array}{c} {}^0J^A_{n/0} \\ \hline {}^0J^L_{n/0} \end{array}\right] = \left[\begin{array}{cccccc} * & * & * & * & * & * \\ \hline * & * & * & * & * & * \end{array}\right]$$

$$\begin{cases} {}^0\underline{k}_i & \Gamma_i = P \\ (\underline{k}_i \times \underline{r}_{n/i}) & \Gamma_i = R \end{cases}$$

$S_{E/n}$ operator is constant matrix

$\longrightarrow$ Rigid Body Jacobian

$$S_{E/n} = \left[\begin{array}{c|c} I_{3\times3} & \emptyset \\ \hline [r_{n/E}\times] & I_{3\times3} \end{array}\right]$$

$$J = J(q)$$

$$[\underline{a}\times] = \begin{bmatrix} 0 & -z & y \\ z & 0 & -x \\ -y & x & 0 \end{bmatrix}$$

$$a = \begin{bmatrix} x \\ y \\ z \end{bmatrix}$$

$${}^0\underline{w}_{e/0} = \cancel{\underline{w}_{e/n}} + \underline{w}_{n/0}$$

$$\underline{v}_{e/0} = \underline{v}_{n/0} + {}^0(\underline{w}_{n/0} \times \underline{r}_{E/n}) =$$

$${}^0(-\underline{r}_{E/n} \times w_{n/0})$$

$$-[\underline{r}_{E/n}\times]\,{}^0\underline{w}_{n/0}$$

$$+[\underline{r}_{E/n}\times]^T\,{}^0w_{n/0}$$

$$\boxed{\begin{bmatrix} {}^0\underline{w}_{E/0} \\ {}^0\underline{v}_{E/0} \end{bmatrix} = S_{E/n}\, J_{n/0}\, \dot{q}}$$

$$ {}^{0}S_{E/n} = \left[\begin{array}{c|c} \mathbb{1} & \emptyset \\ \hline A & \mathbb{1} \end{array}\right] \qquad \left[\begin{array}{c|c} \mathbb{1} & \emptyset \\ \hline A^T & \mathbb{1} \end{array}\right] = \left[\begin{array}{c|c} \mathbb{1} & \emptyset \\ \hline \emptyset & \mathbb{1} \end{array}\right] $$

Inverse of S

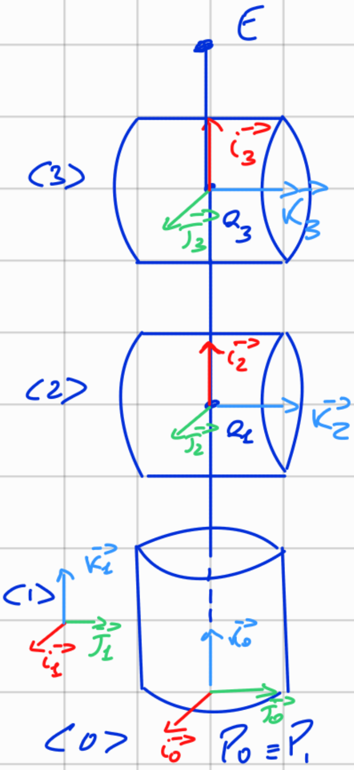
$J_{i/0}$ -> Jacobian express to Joint i

$$ {}^{0}J_{i/0} = \begin{bmatrix} \underline{k_1} & \underline{k_2} & \underline{k_i} & \emptyset & \emptyset \\ (k_1 \times r_{i/1}) & (k_2 \times r_{i/2}) & \emptyset & \emptyset & \emptyset \end{bmatrix} $$

example

$S_{E_i/i}$

Example



Link 0 ${}^0\underline{r}_{Q_0/0} \equiv \underline{\emptyset}$

Link 1 : ${}^1\underline{r}_{Q_1/1} = \ell_1\, {}^1\underline{k}_1 = \begin{bmatrix} 0 \\ 0 \\ \ell_1 \end{bmatrix}$

Link 2: ${}^2\underline{r}_{Q_2/2} = \ell_2\, {}^2\underline{i}_2 = \begin{bmatrix} \ell_2 \\ 0 \\ 0 \end{bmatrix}$

Link 3: ${}^3\underline{r}_{E/3} = \ell_3\, {}^3\underline{i}_3 = \begin{bmatrix} \ell_3 \\ 0 \\ 0 \end{bmatrix}$

Joint 1:

$${}^0\underline{r}_{P_1/0} = {}^0\underline{r}_{Q_0/0}$$

$$R_0 = I_{3\times3}$$

$${}^0_1R = R_0 R_z(q_1)$$

Joint 2:

$${}^1\underline{r}_{P_2/1} = {}^2\underline{r}_{Q_1/1}$$

$$R_1 = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}$$

$${}^1_2R = R_1 R_z(q_2)$$

Joint 3:

$${}^2\underline{r}_{P_3/2} = {}^2\underline{r}_{Q_2/2}$$

$$R_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$${}^2_3R = R_2 R_z(q_2)$$

Need to compute Transformation Matrix

$${}^{0}_{3}T = {}^{0}_{1}T \; {}^{1}_{2}T \; {}^{2}_{3}T$$ Forward Geometry

$${}^{\emptyset}J_{3/\emptyset} = \left[\begin{array}{c|c|c} {}^{0}\underline{k}_1 & {}^{0}\underline{k}_2 & {}^{0}\underline{k}_3 \\ \hline {}^{0}(k_1 \times r_{2/1}) & {}^{0}(k_2 \times r_{3/2}) & \emptyset \end{array}\right]$$

$${}^{0}_{1}T = \left[\begin{array}{c|c} {}^{0}_{1}R & {}^{0}\underline{r}_{1/0} \\ \hline \underline{\emptyset}^T & 1 \end{array}\right] \qquad {}^{0}_{2}T = \left[\begin{array}{c|c} {}^{0}_{2}R & {}^{0}r_{2/0} \\ \hline \underline{\emptyset}^T & 1 \end{array}\right]$$

k_1 = third column of ${}^{0}_{1}R$

k_2 = third column of ${}^{0}_{2}R$

$${}^{0}_{3}T = \left[\begin{array}{c|c} {}^{0}_{3}R & {}^{0}r_{3/0} \\ \hline \underline{\emptyset}^T & 1 \end{array}\right]$$

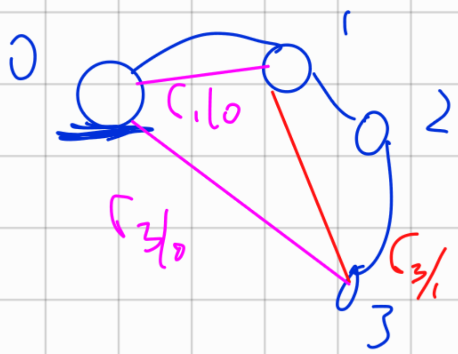
<u>NOTE</u> Mathlab Pseudo Code

${}^{0}\underline{k}_i = {}^{0}_{i}T(1:3, 3)$; ← Extract the 3° column of R who is inside T

$$R = \left[\,\underline{1} \;\middle|\; \underline{0}\,\right] T \left[\begin{array}{c} \underline{1} \\ \hline \underline{0} \end{array}\right]$$ Algebric way (Not Matlab)

$${}^{0}\underline{r}_{3/1} = {}^{0}\underline{r}_{3/0} - {}^{0}r_{1/0}$$

$${}^{0}\underline{r}_{3/2} = \underline{r}_{3/0} - \underline{r}_{2/0}$$



$${}^{0}J_{3/0}\Big|_{\varphi=\phi} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & \ell_2 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

Now want to know the velocity of EE

$${}^{0}S_{E/0} = \left[\begin{array}{c|c} \mathbb{1} & \phi \\ \hline \left[{}^{0}r_{E/3}\times\right] & \mathbb{1} \end{array}\right]$$

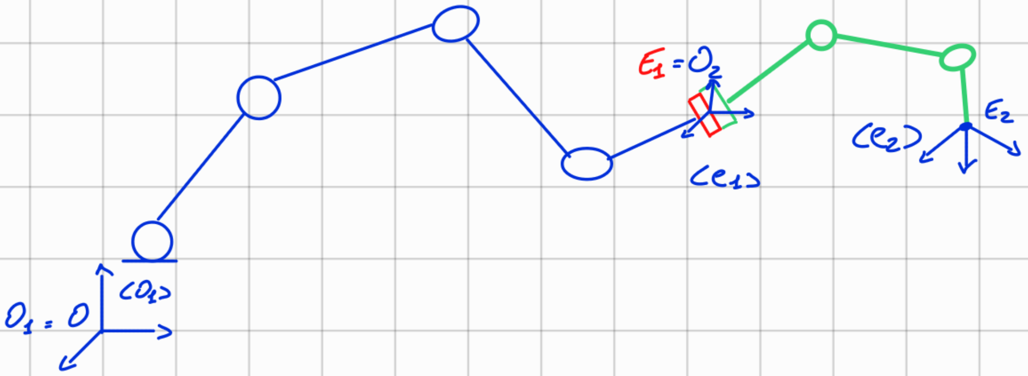
$${}^{3}\underline{r}_{E/3} = \ell_3 \, {}^{3}\underline{i}_3 \longrightarrow {}^{3}\vec{r}_{E/3} = \begin{matrix} \ell_3 \\ 0 \\ 0 \end{matrix}$$

$${}^{0}\bar{\underline{r}}_{E/3} = {}^{0}_{3}T \, {}^{3}\vec{r}_{E/3}$$

1 / Dic / 2023 Important Exercise

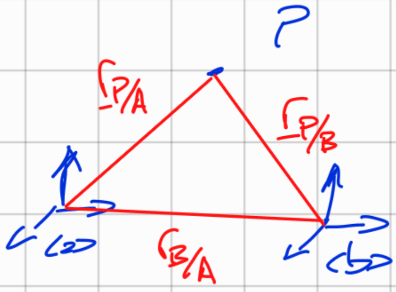
More on Jacobian

$$\text{Robot } 1 \left\{ \begin{array}{l} {}^{O_1}_{E_1}T(q_1) \\ {}^{O_1}J_{E_1/O_1}(q_1) \end{array} \right. \qquad \text{Robot } 2 \left\{ \begin{array}{l} {}^{O_2}_{E_2}T(q_2) \\ {}^{O_2}J_{E_2/O_2}(q_2) \end{array} \right.$$



I Assume: $E_1 \equiv O_2 \implies {}^{e_1}_{O_2}T = I(3\times 3)$

$\langle e_1 \rangle \equiv \langle O_2 \rangle$



$$\left\{ \begin{array}{l} \omega_{e_2/0} = \omega_{e_1/0} + \omega_{e_2/e_1} \\ v_{E_2/0} = v_{E_1/0} + \omega_{e_1/0} \times r_{e_2/e_1} + v_{E_2/E_1} \end{array} \right. \implies$$

So Che:

$$v_{P/A} = v_{P/B} + \omega_{b/a} \times r_{P/B} + v_{B/A}$$

Project everyThings to Frame $\emptyset$

Convert in Algebric Equations

Porto tutto riferito ad $\emptyset$

$$\left\{ \begin{array}{l} {}^{0}\omega_{e_2/0} = {}^{0}J^{A}_{e_1/0}\,\dot q_1 + {}^{\emptyset}_{\emptyset_2}R\;{}^{O_2}J^{A}_{e_2/e_1}\,\dot q_2 = \left[J^{A}_{e_1/0} \,\vdots\, {}^{0}_{O_2}R\;{}^{O_2}J^{A}_{e_2/e_1} \right] \begin{bmatrix} \dot q_1 \\ \dot q_2 \end{bmatrix} \\ {}^{0}v_{e_2/0} = {}^{0}J^{L}_{e_1/0}\,\dot q_1 + \left[r_{e_2/e_1}\times \right]^{T} {}^{0}J^{A}_{e_1/0}\,\dot q_1 + {}^{0}_{e_1}R\;{}^{O_2}J^{L}_{e_2/O_2}\,\dot q_2 = \end{array} \right.$$

$$= \left(\left[r_{e_2/e_1}\times \right]^{T} \,\vdots\, I \right) {}^{0}J^{A}_{e_1/0}\,\dot q_1 + {}^{0}_{E_1}R\;{}^{E_1}J^{L}_{E_2/E_1}\,\dot q_2$$

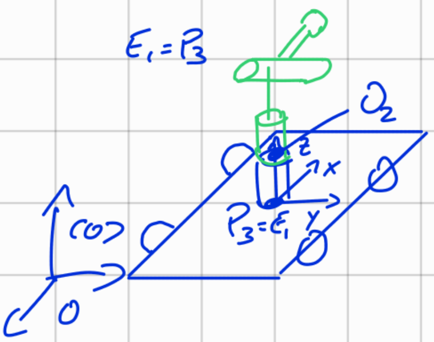
$$\begin{bmatrix} {}^0\underline{w}_{e_2/0} \\ {}^0\underline{v}_{e_2/0} \end{bmatrix} = \left[\begin{array}{c|c} {}^0S_{E_2/E_1}\ {}^0J^A_{E_1/0} & {}^0J^L_{E_2/E_1} \end{array} \right] \dot{q}$$

General Transformation in case of Serial Composition of Robot

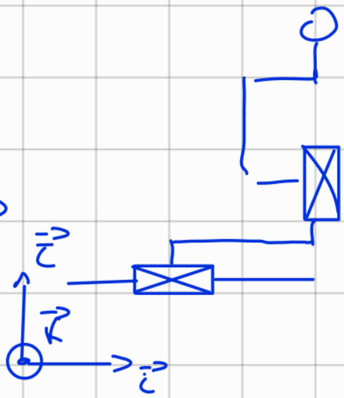
---

Consider Holonomic platform $\to$ Could Move in every directions, Avanti/Indietro, Rotazioni e trasla lateralmente

P, P, R



$\Rightarrow$



$$\begin{cases} q_1 = x \\ q_2 = y \\ q_3 = \theta \end{cases} \qquad \begin{cases} \dot{q}_1 = \dot{x} \\ \dot{q}_2 = \dot{y} \\ \dot{q}_3 = \dot{\theta} \end{cases}$$

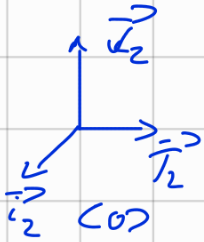
$${}^0J_{E_1/0} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \\ \hline {}^0\underline{k}_1 & {}^0\underline{k}_2 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$R_0 \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \qquad \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

$$R_0 = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ -1 & 0 & 0 \end{bmatrix}$$

Second Robot (Green Robot)

$$J = \left[\begin{array}{cc} V_{i1} & V_{i2} \\ \hline 0 & 0 \end{array}\right] =$$



$P_0 \equiv P_1 \Leftarrow$ Application point of First Revolute

$P_1 \equiv P_2$ -> Because the two Application point of

<1> <2> Frame Are Coincident

$R_1 R_2(\dot{q}_2)$

$$^{0_2}S_{\epsilon_2/0_1} = \begin{bmatrix} 0 & 1 & * \\ 0 & 1 & * \\ 1 & 1 & * \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

$$^{0}_{\epsilon_1}T = {}^{0}_{1}T$$

4/Dic/2023

Inverse Kinematic Problem

Compute the Joint variable, knowing the position of E.E

${}^{0}_{e}T(q)$ Used to find linear velocity for EE

${}^{0}J_{e/0}(q) \rightarrow J \in \mathbb{R}^{6 \times n}$ n = number Of Links

To Compute $\dot{q}$ s.t.

$$\begin{bmatrix} {}^{0}\underline{w}_{e/0} \\ {}^{0}\underline{v}_{e/0} \end{bmatrix} = \begin{bmatrix} \underline{w}^* \\ \underline{v}^* \end{bmatrix} = \underline{\dot{x}}^*$$

[Where $\underline{w}^*$, $\underline{v}^*$, Are the Velocities of Reference Frame]

$$\underline{\dot{x}}^* = J\dot{q}$$

$\dot{q}$ = desired Velocities of end-effector

$$q = \arg\min_{\dot{q}} \frac{1}{2}\|\dot{q}\|^2 = \arg\min_{\dot{q}} \frac{1}{2}\dot{q}^T\dot{q}$$

i) $n = 6$ $\Rightarrow$ J is square matrix

$\Rightarrow \dot{q} = J^{-1}\underline{\dot{x}}^*$ if $|J| \neq 0$ ($\exists!$ Solution)

ii) $n > 6$ Case of Redundant DOF Robot

$\Rightarrow$ ($\exists\ \infty^{n-6}$ Solutions)

iii) $n < 6$

$\Rightarrow \nexists$ Solution for any choice of $\underline{\dot{x}}^*$

Case $n \geq 6$ (Include case i and ii)

$$\underbrace{A x}_{\text{Linear Combination of Columns of A}} = b \qquad \Rightarrow b = \text{SPAN}(A)$$

Linear Combination of
Columns of A

<u>NOTE</u>

$f(\dot{q}) = \dot{q}^T A \dot{q}$ quadratic form, Is Always <u>NON</u> negative

$A = A^T \in \mathbb{R}^{n \times n}$ A is Simmetric $\Rightarrow$ real eigenvalues

$A > 0 \rightarrow A$ is stricty positive definite
$\Leftrightarrow$ All eigenvalue of A are stricty positive

$$f(\dot{q}) > 0 \quad \forall \dot{q} \neq 0$$

$$\dot{q} = \arg\min_{\dot{q}} \frac{1}{2} \|\dot{q}\|^2 = \arg\min_{\dot{q}} \frac{1}{2} \dot{q}^T \dot{q}$$

particular case of quadratic Form
$A = I$

It's means that All the Joint variabile has the same importance

Use Lagrange Multiplies Method

i) $$L(\dot{q}, \lambda) = \underbrace{\frac{1}{2} \dot{q}^T \dot{q}}_{\text{Scalar}} + \lambda^T \underbrace{[J\dot{q} - \dot{x}^*]}_{\text{Scalar}}$$

New Function, new
cost I want to
minimize

ii) $\frac{\partial L}{\partial \dot{q}} = 0 \iff \dot{q} + J^T \underline{\lambda} = 0$

$\frac{\partial L}{\partial \lambda} = 0 \iff J\dot{q} - \underline{\dot{x}}^* = 0$

iii) Develop The Computations

$\dot{q} = -J^T \underline{\lambda}$

$J\dot{q} = \underline{\dot{x}}^* \Rightarrow -\underbrace{J J^T}\underline{\lambda} = \underline{\dot{x}}^*$

$JJ^T \in \mathbb{R}^{6\times 6}$ Is Full Rank Matrix if J is non-Singular

$(JJ^T)^T = (JJ^T) \geq 0$

$\Rightarrow$ I can Compute $\underline{\lambda} = -(JJ^T)^{-1} \underline{\dot{x}}^*$

$\Rightarrow \quad \dot{q} = \underbrace{J^T (JJ^T)^{-1}}_{\text{Right Pseudoinvers of Matrix } J} \underline{\dot{x}}^*$

$$\dot{q} = J^{\#} \underline{\dot{x}}^*$$

With: $\dot{q}$ = desired Joints Variable, $\underline{\dot{x}}^*$ = Actual e-e velocities

Where

$$J^{\#} = J^T(q)\left[J(q)\,J^T(q)\right]^{-1}$$

NOTE Inverting Matrix is Analitic Tools, not Computational Tool.

NOTE

Singular Value Decompositions (SVD)

$A \in \mathbb{R}^{n \times m} \quad \exists \; U \in \mathbb{R}^{n \times n}, \; V \in \mathbb{R}^{m \times m}$

$$\boxed{A = U^T \Sigma V}$$

$$\begin{bmatrix} \lambda_1 & \cdots & 0 & \\ & \ddots & & 0 \\ 0 & & \lambda_n & \end{bmatrix}$$

$U^T = U^{-1}$

$V^T = V^{-1}$

Σ is the pseudo diagonal Matrix

$$\begin{bmatrix} \lambda_1 & & 0 \\ & \ddots & \\ & & \lambda_m \\ & 0 & \end{bmatrix}$$

λ_i is singular Value

$\lambda_i = \sqrt{\sigma_i (A A^T)}$ $\quad$ σ_i = eigenvalue of Matrix A

$\lambda_1 \geq \lambda_2 \geq \cdots \geq \underbrace{\lambda_n \geq 0}$

"$\lambda_n = 0$" if The robot is in Singular Configuration

$$(AB)^{-1} = B^{-1}A^{-1}$$

$$J\dot{q} = \dot{x}^* \Rightarrow \dot{q} = J^{\#}\dot{x}^{\#}$$

$$J = U^T \Sigma_i V, \quad J^T = V^T \Sigma_i^T U$$

$$J^{\#} = J^T \left[J J^T \right]^{-1}$$

$$J^{\#} = V^T \Sigma_i^T U \left[U^T \Sigma_i \cancel{V} \cancel{V^T} \Sigma_i^T U \right]^{-1}$$

$$\left[U^T \begin{bmatrix} \lambda_1^2 & & \\ & \ddots & \\ & & \lambda_n^2 \end{bmatrix} U \right] = U^T \begin{bmatrix} 1/\lambda_1^2 & & 0 \\ & \ddots & \\ 0 & & 1/\lambda_n^2 \end{bmatrix} U$$

$$J^{\#} = V^T \Sigma_i^T \underbrace{\cancel{U} \cancel{U^T}}_{1} \begin{bmatrix} 1/\lambda_1^2 & & 0 \\ & \ddots & \\ 0 & & 1/\lambda_n^2 \end{bmatrix} U =$$

$$= V^T \begin{bmatrix} 1/\lambda_1^2 & & 0 \\ & \ddots & \\ 0 & & 1/\lambda_n^2 \end{bmatrix} U$$

Never Compute A^{-1} in Algebric way inside a Computer (Computational way)

Sensitivity Analisis

$$A = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} = \begin{bmatrix} \underline{v}_1, \underline{v}_2 \end{bmatrix}$$

$$A\underline{y} = \underline{b}$$

Want to estimate : $\dfrac{\|\delta y\|}{\|y\|} \leq \underbrace{\dfrac{\lambda_{max}(A)}{\lambda_{min}(B)}}_{\text{Conditions numbers of A}} \dfrac{\|\delta \underline{b}\|}{\|\underline{b}\|}$ $\quad \chi(A)$

δy = perturbed Solutions
y = nominal Solutions

$$A(\underline{y} + \delta\underline{y}) = (\underline{b} + \delta\underline{b})$$

If A Robot is tending to Singular Configuration
$\lambda_{min}(A)$ tend to 0

=> We introduce the regularization Tecnique
necessary for Avoid "strappi" nel movimento,
Voglio un movimento Armonico, con $\dot{\underline{x}}^*$
Continuo

NOTE $\dot{\underline{x}}^*$ = input of System

$$J^{\#} = J^T(JJ^T + \gamma I_{6\times 6})^{-1}$$

For Avoid that Matrix reach the
Singularity Configurations.

IKP

$$\dot{q} = \arg\min_{\dot{q}} \; \dot{q}^T \dot{q}$$

s.t.

$$J\dot{q} = \dot{x}^*$$

This is the least squares Solution of the $\dot{q} = \ldots$ equation

We can reformulate the equation:

$$\dot{q} = \arg\min_{\dot{q}} \; \dot{q}^T W \dot{q}$$

s.t

$$J\dot{q} = \dot{x}^*$$

We made some Assumptions:

$W = W^T > 0$ (Symmetric and Definite Positive)

Example:

a) $W = \begin{bmatrix} W_1 & & 0 \\ & \ddots & \\ 0 & & W_n \end{bmatrix}$

b) $|\dot{q}_i| \leq q_i^{max}$ Construction Specific

c) $W \triangleq A(q)$

$\dot{q}^T A(q)\, \dot{q} \triangleq$ Kinematic energy

Where $A = A^T > 0$

$$\mathcal{L}(\dot{q}, \lambda) = \frac{1}{2}\dot{q}^T W \dot{q} + \lambda^T \left[J\dot{q} - \underline{x}^* \right]$$

$$\frac{\partial \mathcal{L}}{\partial \dot{q}} = 0 \qquad W\dot{q} + J^T \underline{\lambda} = 0$$

$$\Rightarrow \dot{q} = -W^{-1} J^T \underline{\lambda}$$

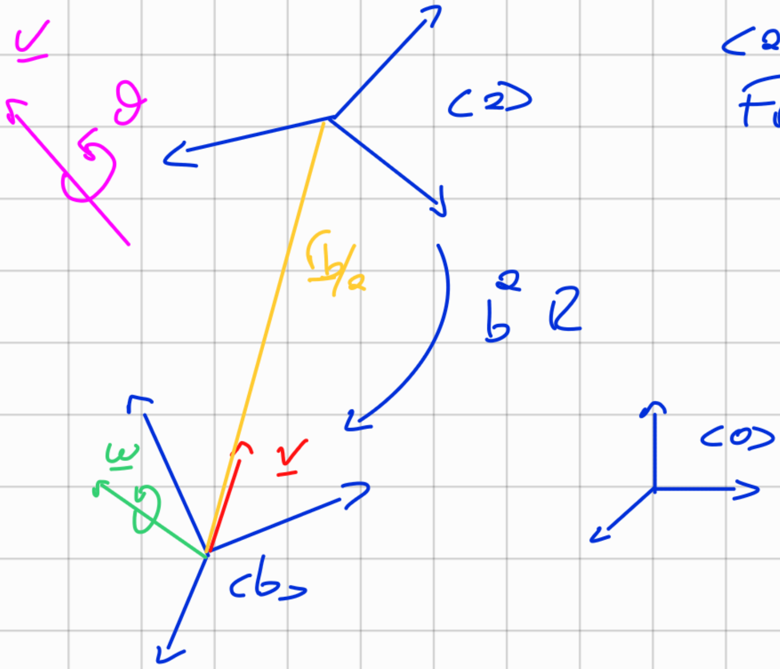
$$\frac{\partial \mathcal{L}}{\partial \underline{\lambda}} = 0 \qquad J\dot{q} - \underline{\dot{x}}^* = \underline{0}$$

$$-(J W^{-1} J^T)\underline{\lambda} = \underline{\dot{x}}^*$$

$$\underline{\lambda} = -(J W^{-1} J^T)^{-1} \underline{\dot{x}}^*$$

$$\dot{q} = W^{-1} J^T (J W^{-1} J^T)^{-1} \underline{x}^*$$

Weighted Leasted Squared



<a> is My Target ref Frame

${}^0_a T$

${}^0_b T$

The goal is ${}^0_a T \equiv {}^0_b T$

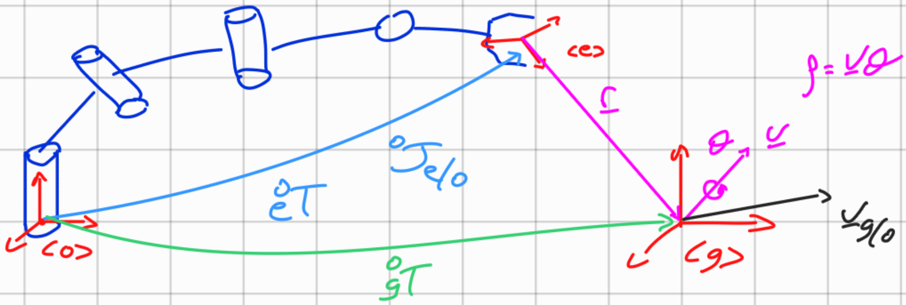
-> I Assume I have a velocity $\underline{v}$ and an angular velocity $\underline{\omega}$

If $r_{b/a} = 0$ => origin <a> ≡ origin <b>

If ${}^a_b R = I$ => The two frame Are Aligned

How To generate $\underline{v}$ and $\underline{\omega}$ for having $r \to 0$ And ${}^a_b R \to I$

Closed Loop Kinematic Control



${}^{0}_{g}T$ Could Be calculated Through Sensors, or other Method
We Assume having ${}^{0}_{g}T$

$\underline{r}$ = distance Between <g> and <e>

We need to know also the speed of the <g> frame
↳ Velocity of the goal $\underline{v}_{g/0}$

If I want <e> on Top of <g> I have this limit:

$$\lim_{t \to +\infty} {}^{0}_{e}T = {}^{0}_{g}T \iff \lim_{t \to +\infty} {}^{e}_{g}T = I$$

$$= \begin{cases} \lim_{t \to +\infty} {}^{e}_{g}R = I \iff \lim_{t \to +\infty} \underline{r} = 0 \\ \lim_{t \to +\infty} \underline{r} = 0 \end{cases}$$

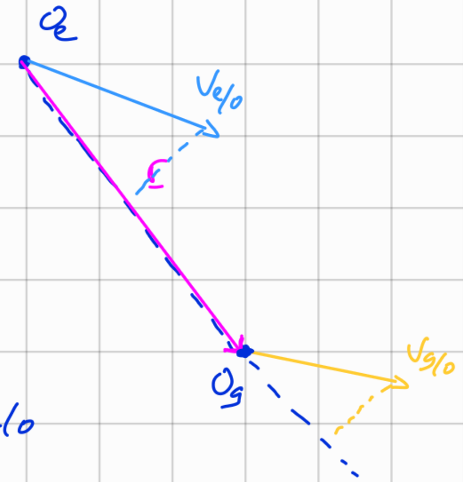
1) Problem: Distance Zeroing

$$\lim_{t \to +\infty} \underline{r} = \underline{0}$$

I want to find out the relationship to how the robot move and how the r change. So I consider the derivative of $\underline{r}$

O_g = origin of g $\quad O_e$ = origin of e

$$\underline{r} = (O_g - O_e)$$



$$\underbrace{D_0\, \underline{r}}_{=\ \dot{\underline{r}}} = D_0 (O_g - O_e) = \underline{v}_{g/0} - \underline{v}_{e/0}$$

Derivata rispetto al frame zero

Now I want to understand the velocity $\underline{v}_{e/0}$

$\dot{\underline{r}}$ = Actual Derivative of $\underline{r}$

$\dot{\bar{\underline{r}}}$ = Desire Derivative of $\underline{r}$

Actual Velocity of EE

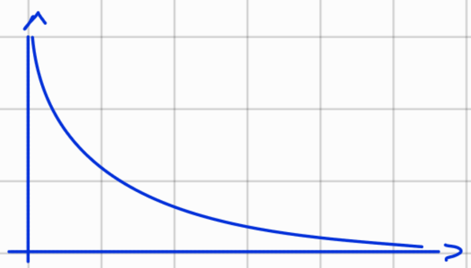
If:

$$\dot{\bar{\underline{r}}} \triangleq -\lambda \underline{r}$$

Differential first order equation Solution is an exponential

Desire Behaviour of $\dot{\underline{r}} = D_0\, \underline{r}$ (Derivate of r in frame zero) ($\Rightarrow D_0\, \underline{r} = {}^0 v_{g/e}$)

- λ = define how fast the curve decrease (λ = time constancy)



$$\dot{\bar{\underline{r}}} = -\lambda \underline{r} = \underline{v}_{g/0} - \underline{v}^*_{e/0}$$

$$\boxed{\underline{v}^*_{e/0} = \underline{v}_{g/0} + \lambda \underline{r}}$$

Desired Velocity for end-effector

Alternative Solution

$\underline{s} = \underline{\mu}\,\sigma$ $\qquad$ $\underline{\mu}$ = unit Vector

σ = scalar

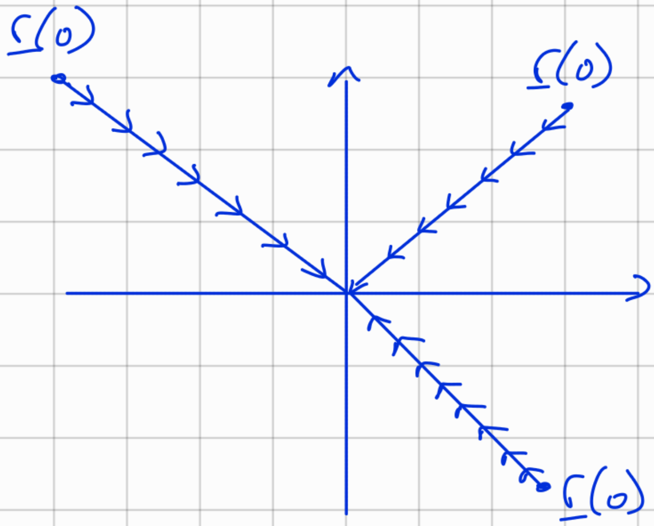
$\dot{\sigma} = \underline{\mu} \cdot (\underline{v}_{g/0} \cdot \underline{v}_{e/0})$ ⟵ Scalar relationship, Not Geometric relationship

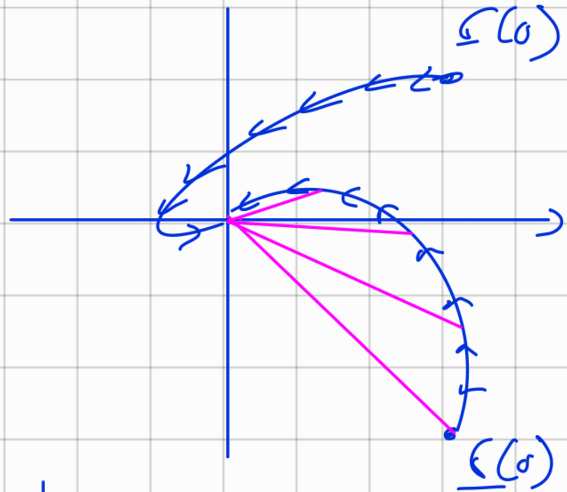
$\updownarrow$

$\dot{\sigma} = -\lambda\sigma$ ⇐ Desired σ

$-\lambda\sigma = \underline{\mu} \cdot \underline{v}_{g/0} - \underline{\mu} \cdot \underline{v}^{*}_{e/0}$

$$\boxed{\underline{\mu} \cdot \underline{v}^{*}_{e/0} = \underline{\mu} \cdot \underline{v}_{g/0} + \lambda\sigma}$$





1° Control Law

Pro: Robot Follow A Straight Line

I have More Precision

Use Only 3 DOF of Robot

2° Control Law

Pro: Scalar Equation

Robot Use his 6 DOF And Could Do More Task

Moument Are less Predictable

2) Problem: Misaligment Zeroing

Change the Orientation for Reach the Goal With the Correct Orientation

Reduced The Angle $\rho = \underline{v}\,\theta$

$$D_0\,\rho = D_e\,\rho + w_{e/0} \times \rho$$

Actual derivate of ρ

$$D_e\,\rho = D_e(\underline{v}\theta) = \underline{v}\,\dot{\theta} + \theta\,D_e(\underline{v}) =$$

$$= \underbrace{\underline{v}\cdot(\underline{v}\cdot w_{g/e})}_{\text{parallel To } \underline{v}} + \underbrace{N(\theta)\,\underline{w}_{g/e}}_{\text{orthogonal to } \underline{v}}$$

Operator Rappresenting Scalar product Is the Transpose of $\underline{v} = \underline{v}^T$

$$N(\theta) = \frac{\theta}{2}\left[\frac{I}{\tan(\theta/2)} - [\underline{v}\times]\right]\left(I - \underline{v}(\underline{v}\cdot)\right)$$

$D_e\,\dot{\rho} = -\lambda\rho$ desired Behaviour

$$\dot{\rho} = -\lambda\,\theta\,\underline{v}$$

$$w_{g/e} = -\lambda\,\theta\,\underline{v} = -\lambda\rho$$

$$\underline{w}_{g/0} - \underline{w}_{e/0} = -\lambda\,\underline{\rho}$$

$$\boxed{\underline{w}^*_{e/0} = \underline{w}_{g/0} + \lambda\rho}$$

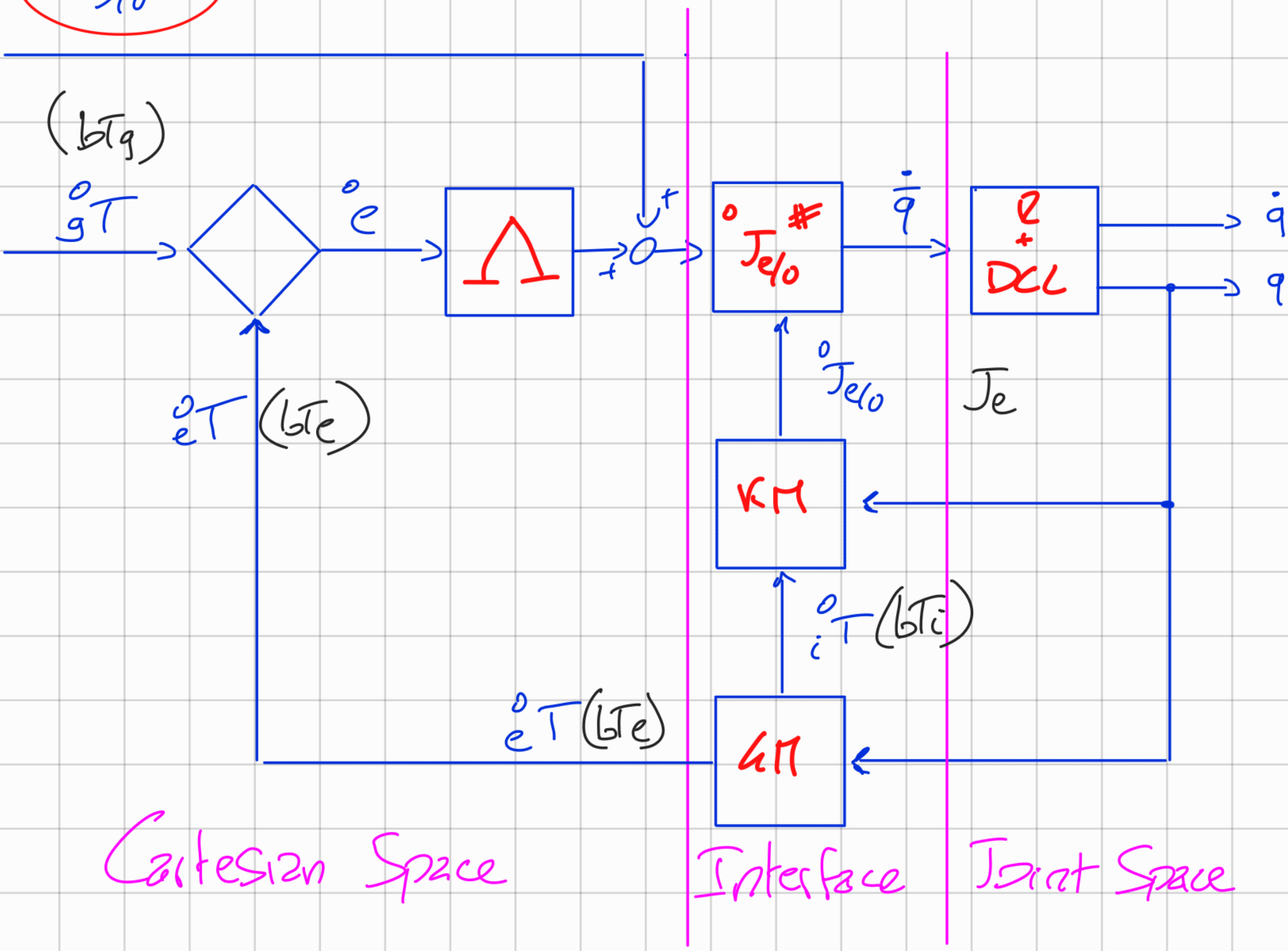
Reference Behaviour for $\underline{w}_{e/0}$

$$\overset{{}^0\dot{\underline{x}}_{e/0}}{\begin{bmatrix} {}^0\underline{\omega}_{e/0} \\ {}^0\underline{v}_{e/0} \end{bmatrix}} = \underline{\Lambda} \overset{{}^0\underline{e}}{\begin{bmatrix} {}^0\underline{\rho} \\ {}^0\underline{r} \end{bmatrix}} + \overset{\dot{\underline{x}}_{g/0}}{\begin{bmatrix} {}^0\underline{\omega}_{g/0} \\ {}^0\underline{v}_{g/0} \end{bmatrix}}$$

$$\underline{\Lambda} \triangleq \begin{bmatrix} \lambda_1 I & \phi_{3\times3} \\ \phi_{3\times3} & \lambda_2 I \end{bmatrix} = \begin{bmatrix} \lambda_1 & 0 & 0 & 0 & 0 & 0 \\ 0 & \lambda_1 & 0 & 0 & 0 & 0 \\ 0 & 0 & \lambda_1 & 0 & 0 & 0 \\ 0 & 0 & 0 & \lambda_2 & 0 & 0 \\ 0 & 0 & 0 & 0 & \lambda_2 & 0 \\ 0 & 0 & 0 & 0 & 0 & \lambda_2 \end{bmatrix}$$

$$\boxed{{}^0\dot{\underline{x}}_{e/0} = \underline{\Lambda}\,{}^0\underline{e} + {}^0\dot{\underline{x}}_{g/0}}$$

$x_{g/0}$ Angular Velocity and Velocity of Goals



This work only if the movement of Robot are inside the robot workspace

Alternative ways to Control Orientation

$$\rho_e = \begin{bmatrix} \psi \\ \vartheta \\ \phi \end{bmatrix}$$

YAW_Pitch - Roll

Actual Derivative of Y-P-R Inverse Mapping From

$$\tilde{\rho} = \rho_g - \rho_e$$

$$\dot{\tilde{\rho}} = \dot{\rho}_{g/0} - \dot{\rho}_{e/0} = T^{-1}_{g/0}(\rho_g)\,\underline{\omega}_{g/0} - T^{-1}_{e/0}(\rho_e)\,\underline{\omega}_{e/0}$$

$${}^{2}T_{b/2}(\rho)\,\dot{\rho} = \begin{bmatrix} 0 & -s\psi & c\psi c\vartheta \\ 0 & c\psi & s\psi c\vartheta \\ 1 & 0 & -s\vartheta \end{bmatrix} \begin{bmatrix} \dot{\psi} \\ \dot{\vartheta} \\ \dot{\phi} \end{bmatrix} = \underline{\omega}_{b/2}$$

$$\dot{\tilde{\rho}}^{*} = -\lambda\tilde{\rho}$$

$$\underline{\omega}^{*}_{e/0} = T_{e/0}(\rho_e)\left[T^{-1}_{g/0}(\rho_g)\,\omega_{g/0} + \lambda\tilde{\rho}\right]$$

2° Alternative way

if $\lim_{e \to g} {}^{0}_{g}R \; {}^{0}_{e}R^T = I$

$$C \triangleq {}^{0}_{g}R \; {}^{0}_{e}R^T \longrightarrow I$$

$$C = {}^{e}_{g}R \; \cancel{{}^{e}_{e}R^T}^{=I}$$

$$\dot{C} C^T = [\underline{\omega}_c \times] = {}^{e}_{g}\dot{R} \; {}^{e}_{g}R^* = [\underline{\omega}_{g/e} \times]$$

$$\Rightarrow \omega_c = \omega_{g/e} = \omega_{g/0} - \omega_{e/0}$$

$$\omega^*_{e/0} = \omega_{g/0} - \omega_c$$

If we chose

$$\underline{\omega}_c = -K \rho_c$$

$$\rho = \theta \underline{v}$$

$$\dot{\rho}_c = -K \rho_c$$

Identical to the misalignment zeroing

$$\underline{\omega}^*_{e/0} = \underline{\omega}_{g/0} + K \rho_c$$

If we chose

Inverse operator of Cross Product

$$\underline{\omega}_c = -K \, vex\left(\frac{C - C^T}{2}\right) = -K \underline{v}_c \sin\theta_c$$

$$\boxed{\dot{\rho}_c = -K \underline{v}_c \sin\theta_c}$$

The vantage is to avoid to define the ρ when $\theta = 0$, because when $\theta = 0$ ρ is undefined, so In this way I remove that problem