Robot Dynamics
Fundamentals

vector

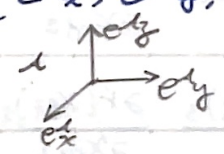
r / $\vec{r}$

$B \to P$ $\quad r_{BP}$

in $\mathcal{A}$ frame $\quad {}_{\mathcal{A}}r_{BP}$

orthonormal basis of $\mathbb{R}^3$

$$(e^{\mathcal{A}}_x, e^{\mathcal{A}}_y, e^{\mathcal{A}}_z) := \text{orthonormal basis of } \mathbb{R}^3$$



parameterization of ${}_{\mathcal{A}}r_{AB} = x\, e^{\mathcal{A}}_x + y\, e^{\mathcal{A}}_y + z\, e^{\mathcal{A}}_z$

Cartesian coordinates

$$\chi_{Pc} = \begin{pmatrix} x \\ y \\ z \end{pmatrix}$$

$$\triangle \; {}_{\mathcal{A}}r_{AB} = x\, e^{\mathcal{A}}_x + y\, e^{\mathcal{A}}_y + z\, e^{\mathcal{A}}_z$$

Cylindral coordinates

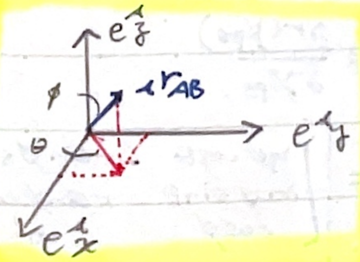
$$\chi_{Pz} = \begin{pmatrix} \rho \\ \theta \\ z \end{pmatrix}$$

$$\triangle \; {}_{\mathcal{A}}r_{AB} = \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} \rho\cos\theta \\ \rho\sin\theta \\ z \end{pmatrix}$$

Spherical Coordinates

$$\chi_{Ps} = \begin{pmatrix} r \\ \theta \\ \phi \end{pmatrix}$$

$$\triangle \; {}_{\mathcal{A}}r_{AB} = \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} r\sin\phi\cos\theta \\ r\sin\phi\sin\theta \\ r\cos\phi \end{pmatrix}$$



ex1-1.

$${}_{\mathcal{A}}r_{AP} = {}_{\mathcal{A}}r_{AB} + {}_{\mathcal{A}}r_{BP}$$

$$\begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ 1 \\ 1 \end{pmatrix}$$

get $\begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$: $\begin{cases} \chi_{Pc} = \\ \chi_{Pz} = \\ \chi_{Ps} = \end{cases}$

$\begin{pmatrix} 0 \\ 1 \\ 1 \end{pmatrix}$: $\begin{cases} \chi_{Pc} = \\ \chi_{Pz} = \\ \chi_{Ps} = \end{cases}$

$\begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}$: $\begin{cases} \chi_{Pc} = \\ \chi_{Pz} = \\ \chi_{Ps} = \end{cases}$

$\begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$ $\quad \chi_{Pc} = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} \rho\cos\theta \\ \rho\sin\theta \\ z \end{pmatrix} = \begin{pmatrix} r\sin\phi\cos\theta \\ r\sin\phi\sin\theta \\ r\cos\phi \end{pmatrix}$

$$\therefore \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 1\cos 0^\circ \\ 1\sin 0^\circ \\ z \end{pmatrix} = \begin{pmatrix} 1\sin 90^\circ \cos 0^\circ \\ 1\sin 90^\circ \sin 0^\circ \\ 1\cos 90^\circ \end{pmatrix}$$

$$\chi_{Pz} = \begin{pmatrix} 1 \\ 0^\circ \\ 0 \end{pmatrix}$$

$$\chi_{Ps} = \begin{pmatrix} 1 \\ 0^\circ \\ 90^\circ \end{pmatrix}$$

$\begin{pmatrix} 0 \\ 1 \\ 1 \end{pmatrix}$ $\quad \chi_{Pc} = \begin{pmatrix} 0 \\ 1 \\ 1 \end{pmatrix} = \begin{pmatrix} \rho\cos\theta \\ \rho\sin\theta \\ z \end{pmatrix} = \begin{pmatrix} r\sin\phi\cos\theta \\ r\sin\phi\sin\theta \\ r\cos\phi \end{pmatrix}$

$$\therefore \begin{pmatrix} 0 \\ 1 \\ 1 \end{pmatrix} = \begin{pmatrix} 1\cos 90^\circ \\ 1\sin 90^\circ \\ 1 \end{pmatrix} = \begin{pmatrix} \sqrt{2}\sin 45^\circ \cos 90^\circ \\ \sqrt{2}\sin 45^\circ \sin 90^\circ \\ \sqrt{2}\cos 45^\circ \end{pmatrix}$$

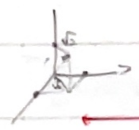
$$\chi_{Pz} = \begin{pmatrix} 1 \\ 90^\circ \\ 1 \end{pmatrix}$$

$$\chi_{Ps} = \begin{pmatrix} \sqrt{2} \\ 90^\circ \\ 45^\circ \end{pmatrix}$$

$\begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}$ $\quad \chi_{Pc} = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} = \begin{pmatrix} \sqrt{2}\cos 45^\circ \\ \sqrt{2}\sin 45^\circ \\ 1 \end{pmatrix} = \begin{pmatrix} \sqrt{3}\sin(\operatorname{atan}\sqrt{2})\cos 45^\circ \\ \sqrt{3}\sin(\operatorname{atan}\sqrt{3})\sin 45^\circ \\ \sqrt{3}\cos(\operatorname{atan}\sqrt{2}) \end{pmatrix}$

$$\chi_{Pz} = \begin{pmatrix} \sqrt{2} \\ 45^\circ \\ 1 \end{pmatrix}$$

$$\chi_{Ps} = \begin{pmatrix} \sqrt{3} \\ 45^\circ \\ \operatorname{atan}\sqrt{2} \end{pmatrix}$$



---

vector derivatives (linear velocity)

$r = r(\chi)$ $\quad$ vector being represented by specific parameterization

$$\dot{r} = \dot{r}(\chi)\,\dot{\chi}$$

$$\Rightarrow \dot{r} = \underbrace{\frac{\partial r}{\partial \chi}}_{E_P(\chi)} \dot{\chi}$$

$$\boxed{\begin{aligned} &\underset{\mathbb{R}^3}{\dot{r}} = \underset{\mathbb{R}^{3\times3}}{E_P(\chi)}\,\dot{\chi} \\ &E_P(\chi)^{-1}\dot{r} = \dot{\chi} \end{aligned}}$$

$r = \begin{pmatrix} x \\ y \\ z \end{pmatrix}$ Cartesian coordinates

$$\chi_{Pc} = \begin{pmatrix} x \\ y \\ z \end{pmatrix}$$

$$\therefore E_{Pc}(\chi_{Pc}) = E^{-1}_{Pc}(\chi_{Pc}) = I$$

Cylindral coordinates

$$r = \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} \rho\cos\theta \\ \rho\sin\theta \\ z \end{pmatrix}$$

---

matrix calculus

vector-by-vector

$$\frac{\partial y}{\partial x} \quad (x, y \in \mathbb{R}^n)$$

$$\frac{\partial y}{\partial x} = \begin{bmatrix} \frac{\partial y_1}{\partial x_1} & \frac{\partial y_1}{\partial x_2} & \cdots & \frac{\partial y_1}{\partial x_n} \\ \frac{\partial y_2}{\partial x_1} & \ddots & & \vdots \\ \vdots & & \ddots & \vdots \\ \frac{\partial y_n}{\partial x_1} & \cdots & & \frac{\partial y_n}{\partial x_n} \end{bmatrix}$$

column picture !

No.

Date

$r(x_{pz}) = [\rho\cos\theta \ \ \rho\sin\theta \ \ z]$

$x_{pz} = [\rho \ \ \theta \ \ z]^T$

calculus revision

e.g. $\nabla f = \frac{\partial f}{\partial x} = \left[\frac{\partial f}{\partial x_1} \ \ \frac{\partial f}{\partial x_2} \ \ \frac{\partial f}{\partial x_3}\right]$ (gradient)

$\llcorner \mathbb{R}^3$

vector by scalar $y = [y_1, y_2 \cdots y_m]^T$, x

$$\frac{\partial y}{\partial x} = \begin{bmatrix} \frac{\partial y_1}{\partial x} \\ \frac{\partial y_2}{\partial x} \\ \vdots \\ \frac{\partial y_m}{\partial x} \end{bmatrix}$$

e.g. position $\downarrow$ velocity

scalar by vector y, $x = [x_1 \ x_2 \cdots x_n]^T$

$$\frac{\partial y}{\partial x} = \left[\frac{\partial y}{\partial x_1}, \ \frac{\partial y}{\partial x_2} \cdots \frac{\partial y}{\partial x_n}\right]$$ e.g. gradient

$$\nabla f = \begin{bmatrix} \frac{\partial f}{\partial x_1} \\ \vdots \\ \frac{\partial f}{\partial x_n} \end{bmatrix} = \left(\frac{\partial f}{\partial x}\right)^T$$

vector by vector $y = [y_1, y_2 \cdots y_m]^T$
$x = [x_1, x_2 \cdots x_n]^T$

$$\frac{\partial y}{\partial x} = \begin{bmatrix} \frac{\partial y_1}{\partial x_1} & \frac{\partial y_1}{\partial x_2} & \cdots & \frac{\partial y_1}{\partial x_n} \\ \frac{\partial y_2}{\partial x_1} & \frac{\partial y_2}{\partial x_2} & \cdots & \frac{\partial y_2}{\partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial y_m}{\partial x_1} & \frac{\partial y_m}{\partial x_2} & \cdots & \frac{\partial y_m}{\partial x_n} \end{bmatrix}$$

Jacobian Matrix

$r = r(x)$

$$\dot{r} = \frac{\partial r}{\partial x}\dot{x} = E_p(x)\dot{x}$$

Cartesian coordinates, as mentioned.

$$x_{pc} = \begin{pmatrix} x \\ y \\ z \end{pmatrix}$$

$$E_{pc}(x_{pc}) = E^{-1}_{pc}(x_{pc}) = I$$

Cylindral coordinates, from matrix calculus

$$r = \begin{pmatrix} \rho\cos\theta \\ \rho\sin\theta \\ z \end{pmatrix}$$

$$x_{pz} = \begin{pmatrix} \rho \\ \theta \\ z \end{pmatrix}$$

$$E_{pz}(x_{pz}) = \frac{\partial r(x_{pz})}{\partial x_{pz}} = \begin{bmatrix} \frac{\partial r(x_{pz})_1}{\partial x_{pz1}} & \frac{\partial r(x_{pz})_1}{\partial x_{pz2}} & \frac{\partial r(x_{pz})_1}{\partial x_{pz3}} \\ \frac{\partial r(x_{pz})_2}{\partial x_{pz1}} & \frac{\partial r(x_{pz})_2}{\partial x_{pz2}} & \frac{\partial r(x_{pz})_2}{\partial x_{pz3}} \\ \frac{\partial r(x_{pz})_3}{\partial x_{pz1}} & \frac{\partial r(x_{pz})_3}{\partial x_{pz2}} & \frac{\partial r(x_{pz})_3}{\partial x_{pz3}} \end{bmatrix}$$

$$= \begin{bmatrix} \cos\theta & -\rho\sin\theta & 0 \\ \sin\theta & \rho\cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

spherical coordinates

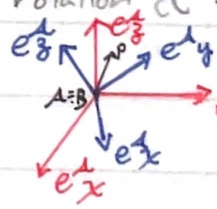
$$r = \begin{pmatrix} r\sin\phi\cos\theta \\ r\sin\phi\sin\theta \\ r\cos\phi \end{pmatrix}$$

$$x_{ps} = \begin{pmatrix} r \\ \theta \\ \phi \end{pmatrix}$$

$$E_{ps}(x_{ps}) = \frac{\partial r(x_{ps})}{\partial x_{ps}}$$

$$= \begin{bmatrix} \sin\phi\cos\theta & -r\sin\phi\sin\theta & r\cos\phi\cos\theta \\ \sin\phi\sin\theta & r\sin\phi\cos\theta & r\cos\phi\sin\theta \\ \cos\phi & 0 & -r\sin\phi \end{bmatrix}$$

rotation & transformation



$$_A r_{AP} = \begin{pmatrix} _A r_{APx} & e^A_x \\ _A r_{APy} & e^A_y \\ _A r_{APz} & e^A_z \end{pmatrix}$$

$$_B r_{AP} = \begin{pmatrix} _B r_{APx} & e^B_x \\ _B r_{APy} & e^B_y \\ _B r_{APz} & e^B_z \end{pmatrix}$$

$$_A r_{AP} = {_A e^B_x} \, {_B r_{APx}} + {_A e^B_y} \, {_B r_{APy}} + {_A e^B_z} \, {_B r_{APz}}$$

vector basis of B frame in A frame

vector in B frame

$$= [{_A e^B_x} \ \ {_A e^B_y} \ \ {_A e^B_z}] \, {_B r_{AP}}$$

$$= C_{AB} \, {_B r_{AP}}$$

$$\therefore \ {_A r_{AP}} = C_{AB} \, {_B r_{AP}}$$

what's this ? orthonormal group SO(3)

$$_A r_{AP} = C_{AB}\ {}_B r_{AP}$$

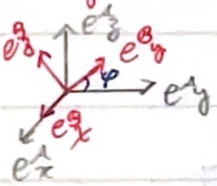
$$C_{BA} = C_{AB}^{-1} = C_{AB}^{T} \quad (SO(3))$$

matrix is orthogonal

Above shows passive rotation (different frame) $\Updownarrow$ active (same frame)

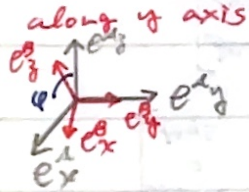
Elementary Rotation = $C_{AB} = \left[\ {}_A e_x^B \ \ {}_A e_y^B \ \ {}_A e_z^B \right]$ ↳ unit vector expressed in other frame (B in A frame)

along x axis



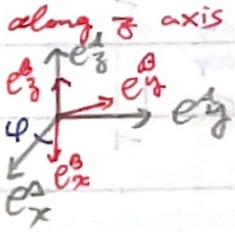
$$C_{AB} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\varphi & -\sin\varphi \\ 0 & \sin\varphi & \cos\varphi \end{bmatrix}$$

along y axis



$$C_{AB} = \begin{bmatrix} \cos\varphi & 0 & \sin\varphi \\ 0 & 1 & 0 \\ -\sin\varphi & 0 & \cos\varphi \end{bmatrix}$$

along z axis



$$C_{AB} = \begin{bmatrix} \cos\varphi & -\sin\varphi & 0 \\ \sin\varphi & \cos\varphi & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$\therefore$ Homogeneous Transformation = rotation + translation

$$r_{AP} = r_{AB} + r_{BP}$$

$$\Rightarrow {}_A r_{AP} = {}_A r_{AB} + {}_A r_{BP} = {}_A r_{AB} + C_{AB}\ {}_B r_{BP}$$

$$\Rightarrow \begin{pmatrix} {}_A r_{AP} \\ 1 \end{pmatrix} = \underbrace{\begin{bmatrix} C_{AB} & {}_A r_{AB} \\ 0_{1\times3} & 1 \end{bmatrix}}_{T} \begin{pmatrix} {}_B r_{BP} \\ 1 \end{pmatrix}$$

$$T^{-1} = \begin{bmatrix} C_{AB}^T & -C_{AB}^T\ {}_A r_{AB} \\ 0_{1\times3} & 1 \end{bmatrix}$$

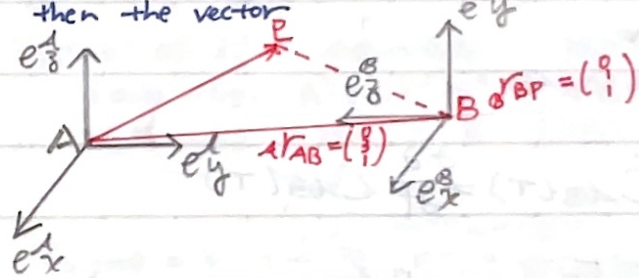
$$-C_{AB}^T\ {}_A r_{AB} = -C_{BA}\ {}_A r_{AB} = -{}_B r_{AB} = {}_B r_{BA}$$

ex. 1-3

find ${}_A r_{AP}$

find T

then the vector



rotate along x axis $90°$ ($\frac{\pi}{4}$)

$$C_{AB} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos 90° & -\sin 90° \\ 0 & \sin 90° & \cos 90° \end{bmatrix}$$

$$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos 90° & -\sin 90° & 3 \\ 0 & \sin 90° & \cos 90° & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\begin{aligned} {}_A r_{AP} &= T_{AB}\ {}_B r_{BP} \\ &= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 3 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} 0 \\ 1 \\ 1 \\ 1 \end{pmatrix} \\ &= \begin{pmatrix} 0 \\ 2 \\ 2 \\ 1 \end{pmatrix} \end{aligned}$$

$${}_A r_{AP} = [0, 2, 2]^T$$

angular velocities

${}_A \omega_{AB} :=$ relative rotational velocity of B w.r.t A in A frame

$$\therefore {}_A \omega_{AB} = -{}_A \omega_{BA}$$

given Rotation Matrix $C_{AB}(t)$

angular velocity is

$${}_A \omega_{AB} = \begin{pmatrix} \omega_x \\ \omega_y \\ \omega_z \end{pmatrix}$$

where skew-symmetric matrix

$$[{}_A \omega_{AB}]_\times = \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix} = \dot{C}_{AB} C_{AB}^T$$

$${}_B \omega_{AB} = C_{BA}\ {}_A \omega_{AB}$$

No.

Date

ex. 1-4

given $C_{AB}(t) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\alpha(t)) & \sin(\alpha(t)) \\ 0 & -\sin(\alpha(t)) & \cos(\alpha(t)) \end{bmatrix}$

determine

$_A\omega_{AB}$

$$\dot{C}_{AB}(t) = \frac{\partial}{\partial t} C_{AB}(t)$$

$$\Rightarrow \begin{bmatrix} 0 & 0 & 0 \\ 0 & -\sin(\alpha(t))\dot{\alpha}(t) & \cos(\alpha(t))\dot{\alpha}(t) \\ 0 & -\cos(\alpha(t))\dot{\alpha}(t) & -\sin(\alpha(t))\dot{\alpha}(t) \end{bmatrix}$$

$$C_{AB}^T(t) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\alpha(t)) & -\sin(\alpha(t)) \\ 0 & \sin(\alpha(t)) & \cos(\alpha(t)) \end{bmatrix}$$

$$\dot{C}_{AB}(t) \cdot C_{AB}^T(t) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & \dot{\alpha}(t) \\ 0 & -\dot{\alpha}(t) & 0 \end{bmatrix} \quad \therefore {}_A\omega_{AB} = \begin{pmatrix} -\dot{\alpha}(t) \\ 0 \\ 0 \end{pmatrix}$$

Parameterization of 3D rotation & Quaternion

now we try to parameterize rotation, similar to position ($\mathbb{R}^3$)

Rotation Matrix $C_{AB} = \left[{}_A e_x^B \;\; {}_A e_y^B \;\; {}_A e_z^B \right]$ $\mathbb{R}^{3\times3}$

Orthonormality

Parameterization

Euler Angle, Angle Axis, Rotation Vector } singularity issue

Quaternions → non-singular

Euler Angle

Recall: Elementary Rotation

$$C_{AB} = C_x(\varphi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\varphi & -\sin\varphi \\ 0 & \sin\varphi & \cos\varphi \end{bmatrix}$$

$$= C_y(\varphi) = \begin{bmatrix} \cos\varphi & 0 & \sin\varphi \\ 0 & 1 & 0 \\ -\sin\varphi & 0 & \cos\varphi \end{bmatrix}$$

$$= C_z(\varphi) = \begin{bmatrix} \cos\varphi & -\sin\varphi & 0 \\ \sin\varphi & \cos\varphi & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

△ Euler Angle Rotation Order

ZYZ & ZXZ — proper Euler Angle

ZYX — Tait-Bryan Angle — UAV Airplanes

XYZ — Cardan Angle — nautic Apps.

$\boxed{ZYZ}$

$$C_{AD} = C_{AD}(\chi_{R,\,Euler\text{-}ZYZ})$$
$$= C_{AB}(z_1)\, C_{BC}(y)\, C_{CD}(z_2)$$

$\Rightarrow {}_A r = C_{AD}\, {}_D r$

$\Rightarrow C_{AD}$

$$= \overset{C_z(\varphi)}{\begin{bmatrix} \cos z_1 & -\sin z_1 & 0 \\ \sin z_1 & \cos z_1 & 0 \\ 0 & 0 & 1 \end{bmatrix}} \overset{C_y(\varphi)}{\begin{bmatrix} \cos y & 0 & \sin y \\ 0 & 1 & 0 \\ \sin y & 0 & \cos y \end{bmatrix}} \overset{C_z(\varphi)}{\begin{bmatrix} \cos z_2 & -\sin z_2 & 0 \\ \sin z_2 & \cos z_2 & 0 \\ 0 & 0 & 1 \end{bmatrix}}$$

$$= \begin{bmatrix} C_y C_{z_1} C_{z_2} - S_{z_1} S_{z_2} & -C_{z_2} S_{z_1} - C_y C_{z_1} S_{z_2} & C_{z_1} S_y \\ C_{z_1} S_{z_2} + C_y C_{z_2} C_{z_1} & C_{z_1} C_{z_2} - C_y S_{z_1} S_{z_2} & S_y S_{z_1} \\ -C_{z_2} S_y & S_y S_{z_2} & C_y \end{bmatrix}$$

$$= \begin{bmatrix} C_{11} & C_{12} & C_{13} \\ C_{21} & C_{22} & C_{23} \\ C_{31} & C_{32} & C_{33} \end{bmatrix} \quad (\chi \to C)$$

get rotation from χ

No.

Date

$$\chi_{R,\text{Euler-ZYZ}} = \begin{pmatrix} z_1 \\ y \\ z_2 \end{pmatrix}$$ get χ from rotation

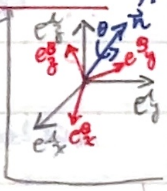
$$= \begin{pmatrix} \text{atan2}(c_{23}, c_{13}) \\ \text{atan2}(\sqrt{c_{13}^2+c_{23}^2}, c_{33}) \\ \text{atan2}(c_{32}, -c_{31}) \end{pmatrix} \quad (C \to \chi)$$

$\boxed{ZYX}$

$$C_{AD} = C_{AD}(\chi_{R,\text{Euler ZYX}})$$
$$= C_{AB}(z)\, C_{AB}(y)\, C_{AB}(x)$$
$$= \begin{bmatrix} \cos z & -\sin z & 0 \\ \sin z & \cos z & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos y & 0 & \sin y \\ 0 & 1 & 0 \\ -\sin y & 0 & \cos y \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos x & -\sin x \\ 0 & \sin x & \cos x \end{bmatrix}$$
$$= \begin{bmatrix} c_y c_z & c_z s_x s_y - c_x s_z & s_x s_z + c_x c_z s_y \\ c_y s_z & c_x c_z + s_x s_y s_z & c_x s_y s_z - c_z s_x \\ -s_y & c_y s_x & c_x c_y \end{bmatrix} \quad \chi \downarrow C$$

$$\chi_{R,\text{Euler ZYX}} = \begin{pmatrix} z \\ y \\ x \end{pmatrix} = \begin{pmatrix} \text{atan2}(c_{21}, c_{11}) \\ \text{atan2}(-c_{31}, \sqrt{c_{32}^2+c_{33}^2}) \\ \text{atan2}(c_{32}, c_{33}) \end{pmatrix} \quad C \downarrow \chi$$

Angle axis



$$\chi_{R,\text{Angle axis}} = \begin{pmatrix} \theta \\ n \end{pmatrix}$$ rotation angle, rotation axis; $n \to \mathbb{R}^3$

Rotation vector $= \varphi = \theta \cdot n \; (\mathbb{R}^3)$
a.k.a. Euler vectors

$$C_{AB}(\theta, n) = \cos\theta \cdot I_{3\times3} - \sin\theta\,[n]_\times + (1-\cos\theta)\, n n^T$$

$$\Rightarrow \begin{bmatrix} n_x^2(1-c_\theta)+c_\theta & n_x n_y(1-c_\theta) - n_z s_\theta & n_x n_z(1-c_\theta) + n_y s_\theta \\ n_x n_y(1-c_\theta) + n_z s_\theta & n_y^2(1-c_\theta)+c_\theta & n_y n_z(1-c_\theta) - n_x s_\theta \\ n_x n_z(1-c_\theta) - n_y s_\theta & n_y n_z(1-c_\theta) + n_x s_\theta & n_z^2(1-c_\theta)+c_\theta \end{bmatrix}$$

$(\chi \to C)$

$$\theta = \cos^{-1}\left(\frac{c_{11}+c_{22}+c_{33}-1}{2}\right)$$

$$n = \frac{1}{2\sin\theta} \begin{pmatrix} c_{32}-c_{23} \\ c_{13}-c_{31} \\ c_{21}-c_{12} \end{pmatrix}$$

singularity

Unit Quaternions

complex numbers in 4D $\xi = \xi_0 + \xi_1 i + \xi_2 j + \xi_3 k$

Hamiltonian convention $i^2 = j^2 = k^2 = ijk = -1$

$\chi_{R,\text{quat}} = \xi = \begin{pmatrix} \xi_0 \\ \check{\xi} \end{pmatrix} \in \mathbb{H}$

Scalar — Real part $\xi_0 = \cos\frac{\|\varphi\|}{2} = \cos\left(\frac{\theta}{2}\right)$

vector — Imaginary $\check{\xi} = \sin\frac{\|\varphi\|}{2} \cdot \frac{\varphi}{\|\varphi\|} = \sin\left(\frac{\theta}{2}\right) n = \begin{pmatrix} \xi_1 \\ \xi_2 \\ \xi_3 \end{pmatrix}$

Unitary Constraint $\xi_0^2 + \xi_1^2 + \xi_2^2 + \xi_3^2 = 1$

Inverse $\xi\begin{pmatrix} \xi_0 \\ \check{\xi} \end{pmatrix} \leftrightarrow \xi^{-1} = \begin{pmatrix} \xi_0 \\ -\check{\xi} \end{pmatrix}$

Identity $\xi = (1\ 0\ 0\ 0)^T \quad (\theta = 0)$

Rotation Matrix

$$C_{AD} = I_{3\times3} + 2\xi_0 [\check{\xi}]_\times + 2[\check{\xi}]_\times^2$$
$$= (2\xi_0^2 - 1) I_{3\times3} + 2\xi_0[\check{\xi}]_\times + 2\check{\xi}\check{\xi}^T$$

$$\begin{bmatrix} \xi_0^2+\xi_1^2-\xi_2^2-\xi_3^2 & 2\xi_1\xi_2 - 2\xi_0\xi_3 & 2\xi_0\xi_2 + 2\xi_1\xi_3 \\ 2\xi_0\xi_3 + 2\xi_1\xi_2 & \xi_0^2-\xi_1^2+\xi_2^2-\xi_3^2 & 2\xi_2\xi_3 - 2\xi_0\xi_1 \\ 2\xi_1\xi_3 - 2\xi_0\xi_2 & 2\xi_0\xi_1 + 2\xi_2\xi_3 & \xi_0^2-\xi_1^2-\xi_2^2+\xi_3^2 \end{bmatrix}$$

$\chi \to C$

$$\chi_{R,\text{quat}} = \xi_{AD} = \frac{1}{2} \begin{pmatrix} \sqrt{c_{11}+c_{22}+c_{33}+1} \\ \text{sgn}(c_{32}-c_{23})\sqrt{c_{11}-c_{22}-c_{33}+1} \\ \text{sgn}(c_{13}-c_{31})\sqrt{c_{22}-c_{33}-c_{11}+1} \\ \text{sgn}(c_{21}-c_{12})\sqrt{c_{33}-c_{11}-c_{22}+1} \end{pmatrix}$$

$C \to \chi$

Quaternions Algebra

$$q \otimes p = (q_0 + q_1 i + q_2 j + q_3 k)(p_0 + p_1 i + p_2 j + p_3 k)$$
$$= q_0p_0 + q_0p_1 i + q_0p_2 j + q_0p_3 k$$
$$+ q_1p_0 i + q_1p_1 ii + q_1p_2 ij + q_1p_3 ik$$
$$+ q_2p_0 j + q_2p_1 ji + q_2p_2 jj + q_2p_3 jk$$
$$+ q_3p_0 k + q_3p_1 ki + q_3p_2 kj + q_3p_3 kk$$

$i^2 = j^2 = k^2 = ijk = -1$
$ij = -ji = -ijk^2 = k$
$jk = -kj = i$
$ki = -ik = j$

$$= q_0p_0 - q_1p_1 - q_2p_2 - q_3p_3$$
$$+ (q_0p_1 + q_1p_0 + q_2p_3 - q_3p_2)\, i$$
$$+ (q_0p_2 - q_1p_3 + q_2p_0 + q_3p_1)\, j$$
$$+ (q_0p_3 + q_1p_2 - q_2p_1 + q_3p_0)\, k$$

$$= \begin{bmatrix} q_0 & -q_1 & -q_2 & -q_3 \\ q_1 & q_0 & -q_3 & q_2 \\ q_2 & q_3 & q_0 & -q_1 \\ q_3 & -q_2 & q_1 & q_0 \end{bmatrix} \begin{pmatrix} p_0 \\ p_1 \\ p_2 \\ p_3 \end{pmatrix}$$

$$= \begin{bmatrix} q_0 & -\check{q}^T \\ \check{q} & q_0 I + [\check{q}]_\times \end{bmatrix} p = M_l(q)\, p$$
$:= M_l(q)$

$$= \begin{bmatrix} p_0 & -p_1 & -p_2 & -p_3 \\ p_1 & p_0 & p_3 & -p_2 \\ p_2 & -p_3 & p_0 & p_1 \\ p_3 & p_2 & -p_1 & p_0 \end{bmatrix} \begin{pmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{pmatrix}$$
$:= M_r(p)$

$$= \begin{bmatrix} p_0 & -\check{p}^T \\ \check{p} & p_0 I - [\check{p}]_\times \end{bmatrix} q = M_r(p)\, q$$

$$(w_1 + \vec{v}_1)(w_2 + \vec{v}_2) = (w_1 w_2 - \vec{v}_1 \cdot \vec{v}_2,\ w_1\vec{v}_2 + w_2\vec{v}_1 + \vec{v}_1 \times \vec{v}_2)$$

No.

Date

Using Quaternion to rotate a vector

$$ {}_B r = C_{BI}\ {}_I r $$

write ${}_I r$ into Quaternion (as the imaginary part of ξ)

$$ p({}_I r) = \begin{pmatrix} 0 \\ {}_I r \end{pmatrix} $$

→ unit quaternion

$$ \therefore\ p({}_B r) = \xi_{BI} \otimes p({}_I r) \otimes \xi_{BI}^T $$

$$ \Updownarrow $$

$$ {}_B r = C_{BI}\ {}_I r $$

$$ p({}_B r) = \xi \otimes p({}_I r) \otimes \xi^T $$

$$ = M_l(\xi) M_r(\xi^T) \begin{pmatrix} 0 \\ {}_I r \end{pmatrix} $$

$$ \Rightarrow \quad \begin{pmatrix} 0 \\ {}_B r \end{pmatrix} = \begin{bmatrix} \xi_0 & -\check{\xi}^T \\ \check{\xi} & \xi_0 I + [\check{\xi}]_\times \end{bmatrix} \begin{bmatrix} \xi_0 & \check{\xi}^T \\ -\check{\xi} & \xi_0 I + [\check{\xi}]_\times \end{bmatrix} \begin{pmatrix} 0 \\ {}_I r \end{pmatrix} $$

$$ = \begin{bmatrix} \xi_0^2 + |\check{\xi}|^2 = 1 & \cancel{\xi_0 \check{\xi}^T} - \cancel{\xi_0 \check{\xi}^T} - \check{\xi}^T [\check{\xi}]_\times \\ \cancel{\xi_0 \check{\xi}} - \cancel{\xi_0 \check{\xi}} - [\check{\xi}]_\times \check{\xi} = 0 & \check{\xi}\check{\xi}^T + \xi_0^2 I + 2\xi_0 [\check{\xi}]_\times + [\check{\xi}]_\times [\check{\xi}]_\times \end{bmatrix} \cdot \begin{pmatrix} 0 \\ {}_I r \end{pmatrix} $$

$$ M_r(\xi) = \begin{bmatrix} \xi_0 & -\check{\xi}^T \\ \check{\xi} & \xi_0 I - [\check{\xi}]_\times \end{bmatrix} $$

$$ [\check{\xi}^T]_\times = -[\check{\xi}]_\times \qquad \boxed{\xi^{-1} = \xi^T = \begin{pmatrix} \xi_0 \\ -\check{\xi} \end{pmatrix}} $$

$$ -\check{\xi}^T [\check{\xi}]_\times = -\check{\xi}^T(-[\check{\xi}^T]_\times) = 0 $$

$$ [\check{\xi}]_\times [\check{\xi}]_\times = \check{\xi}\check{\xi}^T - |\check{\xi}|^2 I $$

$$ = \begin{bmatrix} 1 & 0 \\ 0 & (\xi_0^2 - |\check{\xi}|^2) I + 2\xi_0 [\check{\xi}]_\times + 2\check{\xi}\check{\xi}^T \end{bmatrix} $$

$$ \begin{pmatrix} 0 \\ {}_B r \end{pmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & (\xi_0^2 - |\check{\xi}|^2) I + \xi_0 [\check{\xi}]_\times + 2\check{\xi}\check{\xi}^T \end{bmatrix} \begin{pmatrix} 0 \\ {}_I r \end{pmatrix} $$

$$ \therefore\ {}_B r = \underbrace{\left((\xi_0^2 - |\check{\xi}|^2) I + \xi_0 [\check{\xi}]_\times + 2\check{\xi}\check{\xi}^T\right)}_{C(\xi)} {}_I r $$

Time Derivatives

$$ \omega_{AD} \Leftrightarrow \dot{\chi}_{AD} \qquad \frac{\partial r}{\partial x} $$

recall: $\dot{r} = E_P(x)\dot{\chi}$

Now: $\ {}_A\omega_{AB} = \underbrace{E_R(\chi_R)}_{\text{Find this}} \dot{\chi}_R$

$$ \boxed{{}_A\omega_{AB} = E_R(\chi_R)\dot{\chi}_R} $$

Take ZYX Euler angle for example $\begin{pmatrix} z \\ y \\ x \end{pmatrix}$

$$ {}_A\omega_{AD} = {}_A\omega_{AB} + {}_A\omega_{BC} + {}_A\omega_{CD} $$

$$ = {}_A\omega_{AB} + C_{AB}\,{}_B\omega_{BC} + C_{AB}C_{BC}\,{}_C\omega_{CD} $$

$$ = {}_A e_z^A \cdot \dot{z} + C_{AB}\,{}_B e_y^B \cdot \dot{y} + C_{AB}C_{BC}\,{}_C e_x^C \cdot \dot{x} $$

$$ = \underbrace{\begin{bmatrix} {}_A e_z^A & C_{AB}\,{}_B e_y^B & C_{AB}C_{BC}\,{}_C e_x^C \end{bmatrix}}_{E_R(\chi_R)} \begin{pmatrix} \dot{z} \\ \dot{y} \\ \dot{x} \end{pmatrix} $$

$$ C_{AB}\,{}_B e_y^B = \begin{bmatrix} c_z & -s_z & 0 \\ s_z & c_z & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} = \begin{pmatrix} -s_z \\ c_z \\ 0 \end{pmatrix} $$

$$ C_{AB}C_{BC}\,{}_C e_x^C = \begin{bmatrix} c_z & -s_z & 0 \\ s_z & c_z & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c_y & 0 & s_y \\ 0 & 1 & 0 \\ -s_y & 0 & c_y \end{bmatrix} \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} c_y c_z \\ c_y s_z \\ -s_y \end{pmatrix} $$

$$ \therefore\ E_R(\chi_R) = \begin{bmatrix} 0 & -s_z & c_y c_z \\ 0 & c_z & c_y s_z \\ 1 & 0 & -s_y \end{bmatrix} $$

$$ {}_A\omega_{AB} = \begin{bmatrix} 0 & -s_z & c_y c_z \\ 0 & c_z & c_y s_z \\ 1 & 0 & -s_y \end{bmatrix} \dot{\chi}_R $$

$\det(E_R) = -\cos(y)$ (singularity)

Angle Axis $\begin{pmatrix} \theta \\ n \end{pmatrix}$

$$ E_{R,\text{angle axis}} = \left[n \quad \sin\theta I_{3\times3} + (1-\cos\theta)[n]_\times \right] $$

$$ E_{R,\text{angle axis}}^{-1} = \begin{bmatrix} n^T \\ -\frac{1}{2}\frac{\sin\theta}{1-\cos\theta}[n]_\times^2 - \frac{1}{2}[n]_\times \end{bmatrix} $$

Rotation Vector ($\varphi = \theta n$)

$$ E_{R,\text{rotation vector}} = \left[I_{3\times3} + [\varphi]_\times \left(\frac{1-\cos\|\varphi\|}{\|\varphi\|^2}\right) + [\varphi]_\times^2 \left(\frac{\|\varphi\| - \sin\|\varphi\|}{\|\varphi\|^3}\right) \right] $$

$$ E_{R,\text{rotation vector}}^{-1} = \left[I_{3\times3} - \frac{1}{2}[\varphi]_\times + [\varphi]_\times^2 \frac{1}{\|\varphi\|^2}\left(1 - \frac{\|\varphi\|}{2}\frac{\sin\|\varphi\|}{1-\cos\|\varphi\|}\right) \right] $$

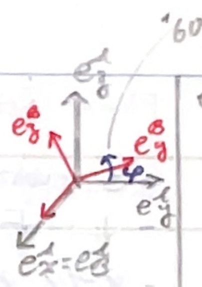
Quaternion

$$ E_{R,\text{quat}} = 2H(\xi) \qquad \text{w/ } H(\xi) = \left[-\check{\xi} \quad [\check{\xi}]_\times + \xi_0 I_{3\times3} \right] \quad R^{3\times4} $$

$$ E_{R,\text{quat}}^{-1} = \frac{1}{2}H(\xi)^T $$

$$ = \begin{bmatrix} -\xi_1 & \xi_0 & -\xi_3 & \xi_2 \\ -\xi_2 & \xi_3 & \xi_0 & -\xi_1 \\ -\xi_3 & -\xi_2 & \xi_1 & \xi_0 \end{bmatrix} $$

Campus



ex. 2-1

Find Rotation Matrix C_{AB}
Euler ZYX
Angle Axis
Quaternions

$$C_{AB} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos60^\circ & -\sin60^\circ \\ 0 & \sin60^\circ & \cos60^\circ \end{bmatrix} \quad \begin{matrix} 11 & 12 & 13 \\ 21 & 22 & 23 \\ 31 & 32 & 33 \end{matrix}$$

$$= \begin{bmatrix} 1 & 0 & 0 \\ 0 & \frac{1}{2} & -\frac{\sqrt{3}}{2} \\ 0 & \frac{\sqrt{3}}{2} & \frac{1}{2} \end{bmatrix} ※$$

Euler ZYX

$$\chi_{R,\text{EulerZYX}} = \begin{pmatrix} z \\ y \\ x \end{pmatrix} = \begin{pmatrix} \text{atan2}(C_{21}, C_{11}) \\ \text{atan2}(-C_{31}, \sqrt{C_{32}^2 + C_{33}^2}) \\ \text{atan2}(C_{32}, C_{33}) \end{pmatrix}$$

$$= \begin{pmatrix} \text{atan2}(0, 1) \\ \text{atan2}(-0, \sqrt{(\sqrt{3}/2)^2 + (1/2)^2}) \\ \text{atan2}(\sqrt{3}/2, 1/2) \end{pmatrix}$$

$$= \begin{pmatrix} 0 \\ 0 \\ 60^\circ \end{pmatrix} ※$$

Angle Axis

$$\theta = \cos^{-1}\left(\frac{C_{11} + C_{22} + C_{33} - 1}{2}\right)$$

$$= \cos^{-1}\left(\frac{1 + \cos60^\circ + \cos60^\circ - 1}{2}\right)$$

$$= \cos^{-1}(\cos60^\circ) = 60^\circ ※$$

$$n = \frac{1}{2\sin(\theta)} \begin{pmatrix} C_{32} - C_{23} \\ C_{13} - C_{31} \\ C_{21} - C_{12} \end{pmatrix} = \frac{1}{2\sin60^\circ} \begin{pmatrix} \sin60^\circ - (-\sin60^\circ) \\ 0 - 0 \\ 0 - 0 \end{pmatrix}$$

$$= \frac{1}{2\sin60^\circ} \begin{pmatrix} 2\sin60^\circ \\ 0 \\ 0 \end{pmatrix}$$

$$= \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} ※$$

Quaternions

$$\chi_{R,\text{quat}} = \frac{1}{2} \begin{pmatrix} \sqrt{C_{11} + C_{22} + C_{33} + 1} \\ \text{sgn}(C_{32} - C_{23}) \sqrt{C_{11} - C_{22} - C_{33} + 1} \\ \text{sgn}(C_{13} - C_{31}) \sqrt{C_{22} - C_{33} - C_{11} + 1} \\ \text{sgn}(C_{21} - C_{12}) \sqrt{C_{33} - C_{11} - C_{22} + 1} \end{pmatrix}$$

$$= \frac{1}{2} \begin{pmatrix} \sqrt{1 + \cos60^\circ + \cos60^\circ + 1} \\ + \sqrt{1 - \cos60^\circ - \cos60^\circ + 1} \\ 0 \cdot \sqrt{\cos60^\circ - \cos60^\circ - 1 + 1} \\ 0 \cdot \sqrt{\cos60^\circ - 1 - \cos60^\circ + 1} \end{pmatrix}$$

$$= \frac{1}{2} \begin{pmatrix} \sqrt{3} \\ 1 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} \frac{\sqrt{3}}{2} \\ \frac{1}{2} \\ 0 \\ 0 \end{pmatrix} = \xi_{AB} ※$$

ex. 2-2

Based upon ex. 2-1 $_A r = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}$ in A frame
Rotate w/ quaternions to B frame
Rotate directly w/ complex numbers to B frame

$$P(_B r) = \begin{pmatrix} 0 \\ _B r \end{pmatrix} = \xi_{BA} \otimes P(_A r) \otimes \xi_{BA}^T$$

$$= M_l(\xi_{BA}) M_R(\xi_{BA}^T) \begin{pmatrix} 0 \\ _A r \end{pmatrix}$$

from previous : $\xi_{AB} = \frac{1}{2} \begin{pmatrix} \sqrt{3} \\ 1 \\ 0 \\ 0 \end{pmatrix}$

$$\therefore \xi_{BA} = \frac{1}{2} \begin{pmatrix} \sqrt{3} \\ -1 \\ 0 \\ 0 \end{pmatrix} \begin{matrix} \cos\frac{\theta}{2} \\ n\sin\frac{\theta}{2} \end{matrix}$$

$$M_l(\xi) = \begin{bmatrix} \xi_0 & -\xi_1 & -\xi_2 & -\xi_3 \\ \xi_1 & \xi_0 & -\xi_3 & \xi_2 \\ \xi_2 & \xi_3 & \xi_0 & -\xi_1 \\ \xi_3 & -\xi_2 & \xi_1 & \xi_0 \end{bmatrix} = \frac{1}{2} \begin{bmatrix} \sqrt{3} & 1 & 0 & 0 \\ -1 & \sqrt{3} & 0 & 0 \\ 0 & 0 & \sqrt{3} & 1 \\ 0 & 0 & -1 & \sqrt{3} \end{bmatrix}$$

$$\xi_{BA}^T = \xi_{BA}^{-1} = \begin{pmatrix} \xi_0 \\ -\xi \end{pmatrix} = \frac{1}{2} \begin{pmatrix} \sqrt{3} \\ 1 \\ 0 \\ 0 \end{pmatrix}$$

$$M_r(\xi) = \begin{bmatrix} \xi_0 & -\xi_1 & -\xi_2 & -\xi_3 \\ \xi_1 & \xi_0 & \xi_3 & -\xi_2 \\ \xi_2 & -\xi_3 & \xi_0 & \xi_1 \\ \xi_3 & \xi_2 & -\xi_1 & \xi_0 \end{bmatrix} = \frac{1}{2} \begin{bmatrix} \sqrt{3} & -1 & 0 & 0 \\ 1 & \sqrt{3} & 0 & 0 \\ 0 & 0 & \sqrt{3} & 1 \\ 0 & 0 & -1 & \sqrt{3} \end{bmatrix}$$

$$P(_B r) = \begin{pmatrix} 0 \\ _B r \end{pmatrix} = \frac{1}{2} \begin{bmatrix} \sqrt{3} & 1 & 0 & 0 \\ -1 & \sqrt{3} & 0 & 0 \\ 0 & 0 & \sqrt{3} & 1 \\ 0 & 0 & -1 & \sqrt{3} \end{bmatrix} \frac{1}{2} \begin{bmatrix} \sqrt{3} & -1 & 0 & 0 \\ 1 & \sqrt{3} & 0 & 0 \\ 0 & 0 & \sqrt{3} & 1 \\ 0 & 0 & -1 & \sqrt{3} \end{bmatrix} \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix}$$

$$= \begin{pmatrix} 0 \\ 0 \\ \frac{1}{2} \\ -\frac{\sqrt{3}}{2} \end{pmatrix} \Big\} {_B r}$$

$$_B r = \begin{pmatrix} 0 \\ \frac{1}{2} \\ -\frac{\sqrt{3}}{2} \end{pmatrix} ※$$

$$P(_B r) = \begin{pmatrix} 0 \\ _B r \end{pmatrix} = \xi_{BA} \otimes P(_A r) \otimes \xi_{BA}^T \quad \left(\xi_{BA} = \frac{1}{2} \begin{pmatrix} \sqrt{3} \\ -1 \\ 0 \\ 0 \end{pmatrix}\right)$$

$$= \left(\frac{1}{2}(\sqrt{3} - i)\right)(j)\left(\frac{1}{2}(\sqrt{3} + i)\right)$$

$i^2 = j^2 = k^2 = ijk = -1$
$ij = -ji = -ijk^2 = k$
$jk = -kj = i$
$ki = -ik = j$

$$= \left(\frac{\sqrt{3}}{2} - \frac{i}{2}\right) j \left(\frac{\sqrt{3}}{2} + \frac{i}{2}\right)$$

$$= \left(\frac{\sqrt{3}}{2} j - \frac{ij}{2}\right)\left(\frac{\sqrt{3}}{2} + \frac{i}{2}\right)$$

$$= \frac{3}{4} j + \frac{\sqrt{3}}{4} ji - \frac{\sqrt{3}}{4} ij - \frac{iji}{4}$$

$$= \frac{3}{4} j - \frac{\sqrt{3}}{4} ij - \frac{\sqrt{3}}{4} ij - \frac{iji}{4}$$

$$= \frac{3}{4} j - \frac{\sqrt{3}}{2} ij - \frac{ki}{4}$$

$$= \frac{3}{4} j - \frac{\sqrt{3}}{2} ij - \frac{j}{4}$$

$$= \frac{1}{2} j - \frac{\sqrt{3}}{2} ij \quad \begin{pmatrix} 0 \\ 0 \\ \frac{1}{2} \\ -\frac{\sqrt{3}}{2} \end{pmatrix} ※$$

Kinematics

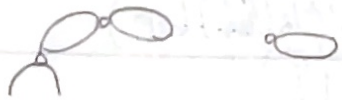
Fixed-base system

n_j joints (revolute, prismatic)

$n_l = n_j + 1$ links.

n_j moving links
1 fixed link

A Robot Arm example



n moving links: $6n$ parameters

n 1DoF joints $5n$ parameters

$6n - 5n = n$ DoFs

$$q = \begin{pmatrix} q_1 \\ \vdots \\ q_{n_j} \end{pmatrix} \in \mathbb{R}^{n_j}$$ { complete, independent, not unique } generalized coordinates

End-effectors (task space)

$$x_e = \begin{pmatrix} x_{eP} \\ x_{eR} \end{pmatrix} = \begin{pmatrix} x_1 \\ \vdots \\ x_m \end{pmatrix} \in \mathbb{R}^m$$ to I (inertial frame)

Operation space coordinates (the "real" DoF @ end-effectors)

$$x_o = \begin{pmatrix} x_{oP} \\ x_{oR} \end{pmatrix} = \begin{pmatrix} x_1 \\ \vdots \\ x_{m_o} \end{pmatrix}$$ → DoF @ end-effectors

ex. 3-1

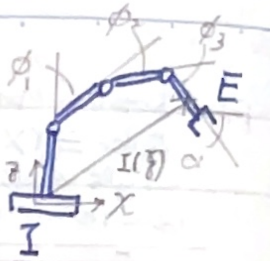
1 Most general robot arm
2 SCARA robot arm
3 ANYpulator (4 joints)
} get q, m_e, x_e, m_o, x_o

1.
$q = (q_1\ q_2\ q_3\ q_4\ q_5\ q_6)$
$m_e = 6$
$x_e = (x, y, z, \alpha_x, \beta_y, \gamma_z)$
$m_o = 6$
$x_o = (x, y, z, \alpha_x, \beta_y, \gamma_z)$

2.
$q = (\alpha, \beta, r, \zeta)$
$m_e = 6$
$x_e = (x, y, z, \alpha_x, \beta_y, \gamma_z)$
$m_o = 4$
$x_o = (x, y, z, \gamma_z)$

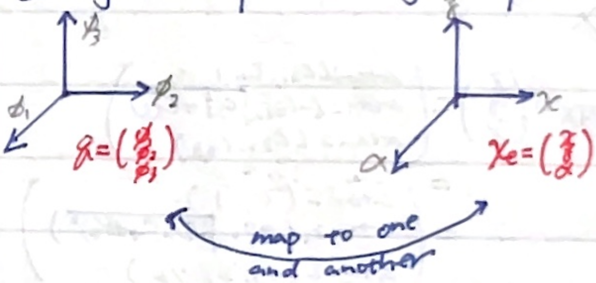
3.
$q = (q_1\ q_2\ q_3\ q_4)$
$m_e = 6$
$x_e = (x, y, z, \alpha_x, \beta_y, \gamma_y)$
$m_o = 4$
$x_o =$ hard to pick

Planar Robot Arm



- 3 revolute joints
- 1 end-effector
- $q = (q_1, q_2, q_3) \in \mathbb{R}^3$
- $m_e = 3$
- $x_e = (x, z, \alpha)$

configuration space ↔ joint space (generalize)



End-effector

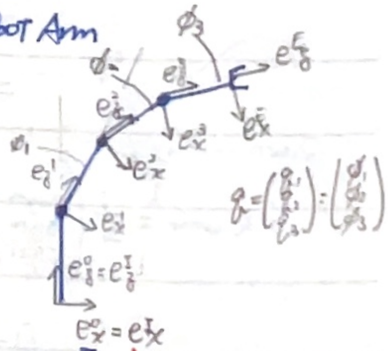
$$x_e = x_e(q) \in \mathbb{R}^{n_e}$$

Transformation Matrix from E to I

$$T_{IE}(q) = T_{I0} \left(\prod_{k=1}^{n_j} T_{k-1\,k}(q_k) \right) T_{n_j E} = \begin{bmatrix} C_{IE} & {}_I r_{IE} \\ 0_{1\times3} & 1 \end{bmatrix}$$

ex. 3-2

Get the T_{IE} of the Robot Arm RHS.



$$q = \begin{pmatrix} q_1 \\ q_2 \\ q_3 \end{pmatrix} = \begin{pmatrix} \phi_1 \\ \phi_2 \\ \phi_3 \end{pmatrix}$$

$$T_{IE} = T_{I0} \cdot T_{01} \cdot T_{12} \cdot T_{23} \cdot T_{3E}$$

$$T_{I0} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

recall: rotate around y

$$C_y(\varphi) = \begin{bmatrix} \cos\varphi & 0 & \sin\varphi \\ 0 & 1 & 0 \\ -\sin\varphi & 0 & \cos\varphi \end{bmatrix}$$

$$T_{01} = \begin{bmatrix} c\phi_1 & 0 & s\phi_1 & 0 \\ 0 & 1 & 0 & 0 \\ -s\phi_1 & 0 & c\phi_1 & l_0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad T_{12} = \begin{bmatrix} c\phi_2 & 0 & s\phi_2 & 0 \\ 0 & 1 & 0 & 0 \\ -s\phi_2 & 0 & c\phi_2 & l_1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T_{23} = \begin{bmatrix} c\phi_3 & 0 & s\phi_3 & 0 \\ 0 & 1 & 0 & 0 \\ -s\phi_3 & 0 & c\phi_3 & l_2 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad T_{3E} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & l_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

No.

Date

$l_1 s\phi_1 + l_2 s(\phi_1+\phi_2) + l_3 s(\phi_1+\phi_2+\phi_3)$

$l_0 + l_1 c\phi_1 + l_2 c(\phi_1+\phi_2) + l_3 c(\phi_1+\phi_2+\phi_3)$

(cont'd)

$$\therefore\ T_{IE} = T_{I0}\cdot T_{01}\cdot T_{12}\cdot T_{23}\cdot T_{3E}$$

$$T_{IE} = \begin{bmatrix} c(\phi_1+\phi_2+\phi_3) & 0 & s(\phi_1+\phi_2+\phi_3) & x \\ 0 & 1 & 0 & y \\ -s(\phi_1+\phi_2+\phi_3) & 0 & c(\phi_1+\phi_2+\phi_3) & z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$x = l_1 s\phi_1 + l_2 s(\phi_1+\phi_2) + l_3 s(\phi_1+\phi_2+\phi_3)$$
$$y = 0$$
$$z = l_0 + l_1 c\phi_1 + l_2 c(\phi_1+\phi_2) + l_3 c(\phi_1+\phi_2+\phi_3)$$

$$\therefore\ \chi_{eP}(q) = \begin{pmatrix} x \\ z \end{pmatrix} = \begin{pmatrix} l_1 s\phi_1 + l_2 s(\phi_1+\phi_2) + l_3 s(\phi_1+\phi_2+\phi_3) \\ l_0 + l_1 c\phi_1 + l_2 c(\phi_1+\phi_2) + l_3 c(\phi_1+\phi_2+\phi_3) \end{pmatrix}$$

$$\chi_{eR}(q) = (\phi_1+\phi_2+\phi_3)$$

Jacobians of a Robot Manipulator

recall Foward Kinematics

$$T_{IE}(q) = \begin{bmatrix} C_{IE}(q) & {}_I r_{IE}(q) \\ 0_{1\times3} & 1 \end{bmatrix} \qquad \chi_e = \begin{pmatrix} \chi_{eP} \\ \chi_{eR} \end{pmatrix} = \chi_e(q)$$

Forward Kinematics (Differential)

- Analytic

point states

$$\chi_e + \delta\chi_e = \chi_e(q+\delta q) = \chi_e(q) + \frac{\partial \chi_e(q)}{\partial q}\delta q + \cancel{O(\delta q^2)}$$

$$\therefore\ \delta\chi_e \cong \frac{\partial \chi_e(q)}{\partial q}\delta q = J_{eA}(q)\,\delta q$$

recall Matrix Calculus

$$J_{eA} = \frac{\partial \chi_e}{\partial q} = \begin{bmatrix} \frac{\partial \chi_1}{\partial q_1} & \cdots & \frac{\partial \chi_1}{\partial q_{n_j}} \\ \vdots & \ddots & \vdots \\ \frac{\partial \chi_m}{\partial q_1} & \cdots & \frac{\partial \chi_m}{\partial q_{n_j}} \end{bmatrix}$$

$$\boxed{\dot{\chi}_e = \underbrace{J_{eA}(q)}_{\mathbb{R}^{m_e\times n_j}}\,\dot{q}}$$

m_e # end-effector state

n_j # joints (generalized)

ex.3-3-1

From last example (ex.3-2) determine the analytic Jacobia

$$\chi_{eP}(q) = \begin{pmatrix} l_1 s\phi_1 + l_2 s(\phi_1+\phi_2) + l_3 s(\phi_1+\phi_2+\phi_3) \\ l_0 + l_1 c\phi_1 + l_2 c(\phi_1+\phi_2) + l_3 c(\phi_1+\phi_2+\phi_3) \end{pmatrix}$$

$$\chi_{eR}(q) = \phi_1+\phi_2+\phi_3$$

$$J_{eAP}(q) = \frac{\partial \chi_{eP}(q)}{\partial q}$$

(cont'd)

$$J_{eAP}(q) = \frac{\partial \chi_{eP}(q)}{\partial q} = \begin{bmatrix} \frac{\partial \chi_{eP}(q)_1}{\partial q_1} & \frac{\partial \chi_{eP}(q)_1}{\partial q_2} & \frac{\partial \chi_{eP}(q)_1}{\partial q_3} \\ \frac{\partial \chi_{eP}(q)_2}{\partial q_1} & \frac{\partial \chi_{eP}(q)_2}{\partial q_2} & \frac{\partial \chi_{eP}(q)_2}{\partial q_3} \end{bmatrix}$$

$$= \begin{bmatrix} l_1 c\phi_1 + l_2 c(\phi_1+\phi_2) + l_3 c(\phi_1+\phi_2+\phi_3) & l_2 c(\phi_1+\phi_2) + l_3 c(\phi_1+\phi_2+\phi_3) & l_3 c(\phi_1+\phi_2+\phi_3) \\ -l_1 s\phi_1 - l_2 s(\phi_1+\phi_2) - l_3 s(\phi_1+\phi_2+\phi_3) & -l_2 s(\phi_1+\phi_2) - l_3 s(\phi_1+\phi_2+\phi_3) & -l_3 s(\phi_1+\phi_2+\phi_3) \end{bmatrix}$$

$$J_{eAR}(q) = \frac{\partial \chi_{eR}(q)}{\partial q} = \begin{bmatrix} \frac{\partial \chi_{eR}(q)_1}{\partial q_1} & \frac{\partial \chi_{eR}(q)}{\partial q_2} & \frac{\partial \chi_{eR}(q)_1}{\partial q_3} \end{bmatrix}$$

$$= [1 \quad 1 \quad 1]$$

$$J_{eA} = \begin{bmatrix} l_1 c\phi_1 + l_2 c(\phi_1+\phi_2) + l_3 c(\phi_1+\phi_2+\phi_3) & l_2 c(\phi_1+\phi_2) + l_3 c(\phi_1+\phi_2+\phi_3) & l_3 c(\phi_1+\phi_2+\phi_3) \\ -l_1 s\phi_1 - l_2 s(\phi_1+\phi_2) - l_3 s(\phi_1+\phi_2+\phi_3) & -l_2 s(\phi_1+\phi_2) - l_3 s(\phi_1+\phi_2+\phi_3) & -l_3 s(\phi_1+\phi_2+\phi_3) \\ 1 & 1 & 1 \end{bmatrix}$$

- Analytic

$$J_{eA}(q) = \mathbb{R}^{m_e\times n_j}$$ depend on parameterization

- Geometric

$$\boxed{w_e = \begin{pmatrix} v_e \\ \omega_e \end{pmatrix} = J_{e0}(q)\dot{q}}$$ map from generalize to linear & angular velocity

$$J_{e0}(q) = \mathbb{R}^{6\times n_j}$$ always 6 !

(recall

$$w_e = E_e(\chi_e)\,\dot{\chi}_e)$$

$$J_{e0}(q) = E_e(\chi)\,J_{eA}(q)$$

$$\left\{\begin{array}{l} w_c = \begin{pmatrix} v_c \\ \omega_c \end{pmatrix} = w_B + w_{BC} \\ J_C\dot{q} = J_B\dot{q} + J_{BC}\dot{q} \\ {}_AJ_c = {}_AJ_B + {}_AJ_{BC} \end{array}\right\}$$ algebras

Derivation of Geometric Jacobian

(recall

$${}_A r_{AP} = {}_A r_{AB} + {}_A r_{BP} = {}_A r_{AB} + C_{AB}\cdot {}_B r_{BP})$$

Differentiate w/ time

$$v_P = {}_A\dot{r}_{AP} = {}_A\dot{r}_{AB} + C_{AB}\,{}_B\dot{r}_{BP} + \dot{C}_{AB}\cdot {}_B r_{BP}$$

(recall

$$[{}_A\omega_{AB}]_\times = \dot{C}_{AB}C_{AB}^T$$

$$C_{BA} = C_{AB}^{-1} = C_{AB}^T)$$

$$\therefore\ \dot{C}_{AB} = [{}_A\omega_{AB}]_\times (C_{AB}^T)^{-1}$$

$$= [{}_A\omega_{AB}]_\times \cdot C_{AB}$$

$$\therefore\ {}_A\dot{r}_{AP} = {}_A\dot{r}_{AB} + \underbrace{C_{AB}\,{}_B\dot{r}_{BP}}_{=0} + [{}_A\omega_{AB}]_\times C_{AB}\,{}_B r_{BP}$$

$$= {}_A\dot{r}_{AB} + [{}_A\omega_{AB}]_\times\,{}_A r_{BP}$$

$$= {}_A\dot{r}_{AB} + {}_A\omega_{AB}\times {}_A r_{BP}$$

$$\therefore\ v_P = v_B + \Omega\times r_{BP} \quad \text{in } A \text{ frame}$$

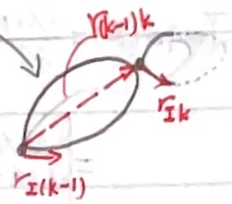
No.

Date

(cont'd)

$V_p = V_B + \Omega \times r_{BP}$ (linear), from above

$\dot{r}_{Ik} = \dot{r}_{I(k-1)} + \underbrace{\omega_{I(k-1)}}_{\text{omega of that body}} \times r_{(k-1)k}$



consecutively

$$\dot{r}_{IE} = \sum_{k=1}^{n} \omega_{I,k} \times r_{k(k+1)} \quad \text{(linear)}$$

as for angular

$$\omega_{I(k)} = \omega_{I(k-1)} + \underbrace{\omega_{(k-1)(k)}}_{= n_k \dot{q}_k}$$

n_k: normal vector of the joint; $\dot{q}_k$: velocity of the generalize coordinate.

$$\therefore \omega_{Ik} = \sum_{i=1}^{k} n_i \dot{q}_i \quad \text{(angular)}.$$

$$\therefore \dot{r}_{IE} = \sum_{k=1}^{n} \omega_{Ik} \times r_{k(k+1)} = \sum_{k=1}^{n} \left\{ \sum_{i=1}^{k} (n_i \dot{q}_i) \times r_{k(k+1)} \right\}$$

$= n_1 \dot{q}_1 \times r_{12}$

1 $+ (n_1 \dot{q}_1 + n_2 \dot{q}_2)\, r_{23}$

2 $+ (n_1 \dot{q}_1 + n_2 \dot{q}_2 + n_3 \dot{q}_3)\, r_{34}$

3 $+ \cdots$

$+ (n_1 \dot{q}_1 + n_2 \dot{q}_2 + \cdots + n_n \dot{q}_n)\, r_{n(n+1)}$

$$= n_1 \dot{q}_1 \times \underbrace{(r_{12} + r_{23} + \cdots + r_{n(n+1)})}_{r_{1(n+1)}} + n_2 \dot{q}_2 \times \underbrace{(r_{23} + \cdots + r_{n(n+1)})}_{r_{2(n+1)}} + \cdots + n_n \dot{q}_n \times \underbrace{(r_{n(n+1)})}_{r_{n(n+1)}}$$

$$= \sum_{k=1}^{n} n_k \dot{q}_k \times r_{k(n+1)}$$

$$\therefore \boxed{\dot{r}_{IE} = \sum_{k=1}^{n} n_k \dot{q}_k\, r_{k(n+1)}}$$

velocity of the end-effector → generalize coordinate → could map from $\dot{q}$ to ω!

$$\dot{r}_{IE} = \underbrace{[n_1 \times r_{1(n+1)} \quad n_2 \times r_{2(n+1)} \cdots n_n \times r_{n(n+1)}]}_{J_{eoP}} \begin{pmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \vdots \\ \dot{q}_n \end{pmatrix}$$

$$\omega_{IE} = \sum_{i=1}^{n} n_i \dot{q}_i = \underbrace{[n_1 \quad n_2 \cdots n_n]}_{J_{eoR}} \begin{pmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \vdots \\ \dot{q}_n \end{pmatrix}$$

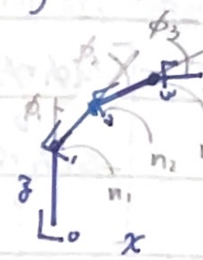
$$\boxed{{}_IJ_{eo} = \begin{bmatrix} J_{eoP} \\ J_{eoR} \end{bmatrix} = \begin{bmatrix} {}_In_1 \times {}_Ir_{1(n+1)} & {}_In_2 \times r_{2(n+1)} & \cdots & {}_In_n \times {}_Ir_{n(n+1)} \\ {}_In_1 & {}_In_2 & \cdots & {}_In_n \end{bmatrix}}$$

$${}_In_k = C_{I(k-1)} \cdot {}_{(k-1)}n_k$$

ex. 3-3-2

Determine the Jacobian (geometric)

RHS.



$${}_IJ_{eo} = \begin{bmatrix} {}_In_1 \times {}_Ir_{1(n+1)} \cdots {}_In_n \times {}_Ir_{n(n+1)} \\ {}_In_1 \cdots {}_In_n \end{bmatrix}$$

${}_IJ_{oP} \in R^{3\times3}$

${}_IJ_{oR} \in R^{3\times3}$

$$\left.\begin{aligned} {}_In_1 &= {}_0n_1 = {}_In_1 = {}_Ie_y \\ {}_In_2 &= C_{I1}\, {}_1n_2 = {}_Ie_y \\ {}_In_3 &= C_{I2}\, {}_2n_3 = {}_Ie_y \end{aligned}\right\} \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}$$

$${}_Ir_{1E} = {}_Ir_{12} + {}_Ir_{23} + {}_Ir_{3E}$$

$$= C_{I1} \cdot {}_1r_{12} + C_{I2} \cdot {}_2r_{23} + C_{I3} \cdot {}_3r_{3E}$$

$$= l_1 \begin{pmatrix} s\phi_1 \\ 0 \\ c\phi_1 \end{pmatrix} + l_2 \begin{pmatrix} s(\phi_1+\phi_2) \\ 0 \\ c(\phi_1+\phi_2) \end{pmatrix} + l_3 \begin{pmatrix} s(\phi_1+\phi_2+\phi_3) \\ 0 \\ c(\phi_1+\phi_2+\phi_3) \end{pmatrix}$$

$${}_Ir_{2E} = {}_Ir_{23} + {}_Ir_{3E}$$

$$= C_{I2} \cdot {}_2r_{23} + C_{I3} \cdot {}_3r_{3E}$$

$$= l_2 \begin{pmatrix} s(\phi_1+\phi_2) \\ 0 \\ c(\phi_1+\phi_2) \end{pmatrix} + l_3 \begin{pmatrix} s(\phi_1+\phi_2+\phi_3) \\ 0 \\ c(\phi_1+\phi_2+\phi_3) \end{pmatrix}$$

$${}_Ir_{3E} = {}_Ir_{3E} = l_3 \begin{pmatrix} s(\phi_1+\phi_2+\phi_3) \\ 0 \\ c(\phi_1+\phi_2+\phi_3) \end{pmatrix}$$

assume $l_1 = l_2 = l_3 = l$

$\therefore {}_IJ_{eo} =$

$$\begin{bmatrix} lc\phi_1 + lc(\phi_1+\phi_2) + lc(\phi_1+\phi_2+\phi_3) & lc(\phi_1+\phi_2) + lc(\phi_1+\phi_2+\phi_3) & lc(\phi_1+\phi_2+\phi_3) \\ 0 & 0 & 0 \\ -ls\phi_1 - ls(\phi_1+\phi_2) - ls(\phi_1+\phi_2+\phi_3) & -ls(\phi_1+\phi_2) - ls(\phi_1+\phi_2+\phi_3) & -ls(\phi_1+\phi_2+\phi_3) \\ 0 & 0 & 0 \\ 1 & 1 & 1 \\ 0 & 0 & 0 \end{bmatrix}$$

ex. 3-3-3

Given

$$r_p = \begin{pmatrix} l_1 C_1 \dot{q}_1 + l_1 C_{12} (\dot{q}_1 + \dot{q}_2) + l_1 C_{123} (\dot{q}_1 + \dot{q}_2 + \dot{q}_3) \\ -l_1 S_1 \dot{q}_1 - l_1 S_{12} (\dot{q}_1 + \dot{q}_2) - l_1 S_{123} (\dot{q}_1 + \dot{q}_2 + \dot{q}_3) \end{pmatrix}$$

$\omega = \dot{q}_1 + \dot{q}_2 + \dot{q}_3$

determine Jacobian (geometric)

(cont'd)

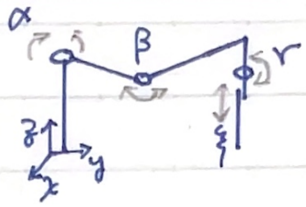
$$w_e = \begin{pmatrix} v_e \\ \omega_e \end{pmatrix} = J_{eo}(q)\,\dot{q}$$

$$\begin{pmatrix} \dot{r}_p \\ \omega \end{pmatrix} = J_{eo}(q) \cdot \begin{pmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \end{pmatrix}$$

$\therefore J_{eo}(q)$

$$= \begin{bmatrix} l_1 C_1 + l_1 C_{12} + l_1 C_{123} & l_1 C_{12} + l_1 C_{123} & l_1 C_{123} \\ -l_1 S_1 - l_1 S_{12} - l_1 S_{123} & -l_1 S_{12} - l_1 C_{123} & -l_1 S_{123} \\ 1 & 1 & 1 \end{bmatrix}$$

SCARA Robot Arm



generalized coordinate

$$\vec{p} = [\alpha\ \beta\ \gamma\ \xi]^T$$

geometric rotation Jacobian

$$\varphi = \overset{\text{offset}}{c} + \alpha + \beta + \gamma$$

$$\dot{\varphi} = \dot{\alpha} + \dot{\beta} + \dot{\gamma} \qquad J_{EoR} = [1\ 1\ 1\ 0]$$

geometric position Jacobian

$$\vec{x}_{EP} = \begin{pmatrix} \cos\alpha + \cos(\alpha+\beta) \\ \sin\alpha + \sin(\alpha+\beta) \\ c - \xi \end{pmatrix}$$

$$\dot{\vec{x}}_{EP} = \begin{pmatrix} -\dot{\alpha}\sin\alpha - (\dot{\alpha}+\dot{\beta})\sin(\alpha+\beta) \\ \dot{\alpha}\cos\alpha + (\dot{\alpha}+\dot{\beta})\cos(\alpha+\beta) \\ -\dot{\xi} \end{pmatrix}$$

$$\therefore J_{EoP} = \begin{bmatrix} -\sin(\alpha) - \sin(\alpha+\beta) & -\sin(\alpha+\beta) & 0 & 0 \\ \cos(\alpha) + \cos(\alpha+\beta) & \cos(\alpha+\beta) & 0 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix}$$

Δ Inverse Kinematics

• Recall: forward kinematics

$T_{IE}(q) = \begin{bmatrix} C_{IE}(q) & {}_I r_{IE}(q) \\ 0 & 1 \end{bmatrix} \in SE(3)$

parametrize

$x_e = \begin{pmatrix} r_e(q) \\ \phi_e(q) \end{pmatrix} = f(q)$

$x_e = \begin{pmatrix} x_{eP} \\ x_{eR} \end{pmatrix}$ position, rotation

• Q: $q = f^{-1}(x_E)$? give you x_E, give me q

Δ sol^n

• Analytic:
 - for 3 intersecting neighbouring axes
• Geometric:
 - use length, theorem ... "geometric info"
• Algebraic
 - use TFs to get joints
• numerical!

Δ numerical method: inverse differential kinematics

• recall

$w_e = J_{e0}\dot q$

$\dot q = J_{e0}^{+} w_e^{*}$ pseudo inverse — multiple sol, singular

• singularities
 - occur @ q_s ⇒ $J_{e0}(q_s)$ is column deficient
 - $\dot q = J_{e0}^{+} w_e^{*}$ — inf ← column deficient ← small
 - categories
  [boundary: when robot-arm is @ its limitation; easy to prevent
  [internal: hard to prevent
 - damped version of Moore-Penrose pseudo inverse

$\dot q = J^{+} w \Leftrightarrow \underset{q}{\text{argmin}} \|J\dot q - w\|_2$

$\Leftrightarrow \underset{q}{\text{argmin}} \|J\dot q - w\|_2^2 + \lambda^2\|w\|_2^2$

$\dot q = J^T(JJ^T + \lambda^2 I)^{-1} w$

• Redundancy

- $q \in R^n$ $w = J\dot q$

$w \in R^m$

$J \in R^{m\times n}$ $m > n$ ↳ redundancy

- $J_{e0}\dot q = w_e^{*}$

⇒ $J_{e0}(J_{e0}^{+} w_e^{*} + N\dot q_0) = w_e^{*}$

⇒ $\dot q = J_{e0}^{+} w_e^{*} + N\dot q_0$

$N = \mathcal{N}(J_{e0})$ null space

$J_{e0}N = 0$

- get N?
 • $N = I - J_{e0}^{+}J_{e0}$
 • SVD
 • QR
 } end-up w/ different basis that span Null space

Δ multi-task control

- track desired position & orientation ensure kinematics
- break down tasks, i.e. $\{J_i, w_i^{*}\}$

- $\dot q = \begin{bmatrix} J_1 \\ \vdots \\ J_{n_t} \end{bmatrix}^{+} \begin{bmatrix} w_1^{*} \\ \vdots \\ w_{n_t}^{*} \end{bmatrix}$ — $\bar J$, $\bar w$

doesn't necessarily mean the end-effector state space. It's just an objective space. J is the mapping matrix

⇒ $\min \|\bar J\dot q - \bar w\|_2$

⇒ $\min \|\dot q\|_2^2$ s.t. $\bar J\dot q = \bar w$

- weighting $[w_1, \dots, w_n]$

$J^{+W} = (\bar J^T W \bar J)^{-1}\bar J^T W$

- Prioritization

recall: $\dot q = J_1^{+} w_1^{*} + N_1\dot q_0$

⇒ $w_2 = J_2\dot q$

$= J_2(J_1^{+} w_1^{*} + N_1\dot q_0)$ modify the $\dot q_0$ to satisfy

⇒ $\dot q_0 = (J_2 N_1)^{+}(w_2^{*} - J_2 J_1^{+} w_1^{*})$

⇒ $\dot q = J_1^{+} w_1^{*} + N_1(J_2 N_1)^{+}(w_2^{*} - J_2 J_1^{+} w_1^{*})$ — $\dot q_0$ ↳ $J_2\dot q \cong w_2$

∴ $\dot q = \sum_{i=1}^{n_t} N_i\dot q_i$

w/ $\dot q_i = (J_i N_i)^{+}(w_i^{*} - J_i\sum_{k=1}^{i-1} N_k\dot q_k)$

Δ Back to inverse Kinematics

- $\dot x_e = J_{eA}(q)\dot q$ analytical

$w = J_0\dot q$ geometric

always relevant → what's the TF between

- now $\Delta x_e = J_{eA}(q)\Delta q$

- tracking a point x_e^{*}, $q_0 = q^0$

pseudo-code
```
q ← q^0
while ||x_e* - x_e(q)|| ≥ tol do
    J_eA ← ∂x_e/∂q (q)
    J_eA^+ ← (J_eA)^+
    Δx_e ← x_e* - x_e(q)
    q ← q + J_eA^+ Δx_e
```
(usually joint-space bigger than task-space → redundant system)

Δ iterative method

• problem 1

$q^{i+1} = q^i + J_{eA}^{+}\Delta x$

if Δx ↑ & J_{eA}^{+} is ill-condition

∴ $q^{i+1} = q^i + k J_{eA}^{+}\Delta x$ — slower convergence

• problem 2

Jacobian is rank deficient → bad condition

∴ $q^{i+1} = q^i + J_{eA}^T(J_{eA}J_{eA}^T + \lambda^2 I)^{-1}\Delta x$

or $q^{i+1} = q^i + \alpha J_{eA}^T\Delta x$

Δ Orientation

- depends on parameterization
- $\in SO(3)$
- $\Delta\chi_{rotate} \neq \Delta\varphi$

$C_{GS}(\Delta\varphi) = C_{GS}(\varphi^{*}) C_{GS}^T(\varphi^{t})$

⇒ $q \leftarrow q + k_{pR} J_{eOR}^{+}\Delta\varphi$ — Lie algebra; geometric one

Δ Traj. ctrl.

• given
 $r_e^{*}(t)$
 $\dot r_e^{*}(t)$

• feedback
 • $\Delta r_e^t = r_e^{*}(t) - r_e(q^t)$
 • $\dot q_P^{*} = J_{eOP}^{+}(q^t)\cdot(\dot r_e^{*}(t) + k_{pP}\Delta r_e^t)$
 $\dot q_R^{*} = J_{eOR}^{+}(q^t)\cdot(w(t)_e^{*} + k_{pR}\Delta\varphi^t)$

Δ L.A. revisit

• determinants
- $\det(A) = 0$ squeeze original state space, to lower dimension
- $\det(A) = c$ — Area = c
 if $\text{sgn}(c) = (-)$ then

• Rank
 $\text{rank}(A) = n$
 then n is the dimension of the spanned space of the output column space

• Null space (kernel)
 $Ax = 0$ $A \in R^{m\times n}$
 ↳ a set of vectors, which form a space, get "squeeze" to the origin;
 note that all A has trivial null space $\{0\}$;
 null space $\in R^{n-m}$ if $n > m$; reduced rank

- ill-condition matrix & matrix norm

Δ Dynamics

description of motion — source of motion

- kinematic v dynamic
 don't care about the cause, let me know the relation! / what's the source
- cause of motion

$\tau \longrightarrow \ddot q$

cause of motion! — description

※ $M(q)\ddot q + b(q,\dot q) + g(q) = \tau + J_c^T F_c$
 - Newton-Euler (free cut, conservation of impulse & angular momentum)
 - Projected Newton-Euler (generalized coordinates)
 - Lagrange II (energy)

Δ Principle of virtual work (D'Alembert's Principle)

• Dynamic equilibrium ⇒ 0 virtual work

$\delta W = \int_B \delta r^T\cdot(\ddot r\,dm - dF_{ext}) = 0$

• Newton's law for every particle force $F = ma$

⇒ $F = \frac{d}{dt}(mv) = \dot p$ — p momentum (linear impulse)

momentum

$r\times F = r\times ma$

$\Gamma = \frac{d}{dt}(r\times mv) = \dot N$ — N angular momentum

Δ Derivation of virtual displacements of single rigid body

- from rigid body kinematics

$r = r_S + \rho$ $[\Omega]_\times\rho = -[\rho]_\times\Omega$

$\dot r = v_S + \Omega\times\rho = [I_{3\times3}\ \ -[\rho]_\times]\begin{bmatrix} v_S \\ \Omega \end{bmatrix}$

$\ddot r = a_S + \Psi\times\rho + \Omega\times(\Omega\times\rho)$

$= [I_{3\times3}\ \ -[\rho]_\times]\begin{bmatrix} a_S \\ \Psi \end{bmatrix} + [\Omega]_\times[\Omega]_\times\rho$

$\delta r = \delta r_S + \delta\Phi\times\rho = [I_{3\times3}\ \ -[\rho]_\times]\begin{bmatrix} \delta r_S \\ \delta\Phi \end{bmatrix}$

- Apply principle of virtual work

$\delta W = \int_B \delta r^T(\ddot r\,dm - dF_{ext}) = 0$

$= \int_B \begin{pmatrix} \delta r \\ \delta\Phi \end{pmatrix}^T \begin{pmatrix} I_{3\times3} \\ [\rho]_\times \end{pmatrix}\cdot\left(\left\{[I_{3\times3}\ \ -[\rho]_\times]\begin{bmatrix} a_S \\ \Psi \end{bmatrix} + [\Omega]_\times[\Omega]_\times\rho\right\}dm - dF_{ext}\right)$

$= \begin{pmatrix} \delta r \\ \delta\Phi \end{pmatrix}^T \int_B\left(\begin{bmatrix} I\,dm & [\rho]_\times^T dm \\ [\rho]_\times dm & -[\rho]_\times^2 dm \end{bmatrix}\begin{pmatrix} a_S \\ \Psi \end{pmatrix} + \begin{pmatrix} [\Omega]_\times^2\rho\,dm \\ [\rho]_\times[\Omega]_\times^2\rho\,dm \end{pmatrix} - \begin{pmatrix} dF_{ext} \\ [\rho]_\times dF_{ext} \end{pmatrix}\right)$

w/ $\int_B dm = m$

$\int_B \rho\,dm = 0$ S = COG moment.

$\int_B -[\rho]_\times^2 dm = \int_B [\rho]_\times[\rho]_\times^T dm = \Theta_S$ moment of inertia

⇒ $0 = \delta W = \begin{pmatrix} \delta r_S \\ \delta\Phi \end{pmatrix}^T\left(\begin{bmatrix} I_{3\times3}m & 0 \\ 0 & \Theta_S \end{bmatrix}\begin{bmatrix} a_S \\ \Psi \end{bmatrix} + \begin{pmatrix} 0 \\ [\Omega]_\times\Theta_S\Omega \end{pmatrix} - \begin{pmatrix} F_{ext} \\ T_{ext} \end{pmatrix}\right)$

$\forall\begin{pmatrix} \delta r_S \\ \delta\Phi \end{pmatrix}$ → ∀ free body can move in all directions

⇒ means that

$\begin{cases} m a_S - F_{ext} = 0 \\ \Theta_S\Psi + [\Omega]_\times\Theta_S\Omega - T_{ext} = 0 \end{cases}$

note that

$P_S = m v_S$ momentum

$\dot P_S = m a_S$ (f) change in momentum

$N_S = \Theta_S\Omega_S$ angular momentum

$\dot N_S = \Theta_S\Psi + \Omega\times\Theta_S\Omega$ change in angular momentum

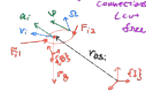
⇒ $\dot P_S = F_{ext}$ force — newton
 $\dot N_S = T_{ext}$ torque — euler
 "change in linear/angular momentum"

Δ some reminder

	linear	rotational
p	x	θ
v	v	ω
a	a	α
motion	$x = vt$	$\theta = \omega t$
	$v = v_0 + at$	$\omega = \omega_0 + \alpha t$
	$x = v_0 t + \frac12 at^2$	$\theta = \omega_0 t + \frac12\alpha t^2$
	$v^2 = v_0^2 + 2ax$	$\omega^2 = \omega_0^2 + 2\alpha\theta$
mass	m	I
	$F = ma$	$\tau = I\alpha$
	$p = mv$	$L = I\omega$
work	Fd	$\tau\theta$
	$\frac12 mv^2$	$\frac12 I\omega^2$
power	Fv	$\tau\omega$

Δ More on Newton-Euler method
(when we are not free body)

↓ remove the connections (cut all bodies free)



produce constraining forces

• based on this, w/ n rigid bodies
 - 6n equation
 - 5n constrained force

why 5n: if 1 DOF joints you are imposing 5 DOF constraints in each joint

Δ projected Newton-Euler (in generalized motion direction)

- recall

$0 = \delta W = \sum_{i=1}^{n_b}\begin{pmatrix} \delta r_{S_i} \\ \delta\Phi_{S_i} \end{pmatrix}^T\left(\begin{bmatrix} \dot P_{S_i} \\ \dot N_{S_i} \end{bmatrix} - \begin{bmatrix} F_{ext,i} \\ T_{ext,i} \end{bmatrix}\right)\quad\forall\begin{pmatrix} \delta r_{S_i} \\ \delta\Phi_{S_i} \end{pmatrix}$

now, this is limited due to linkages!

- now express in generalized coordinates to represent the joints
- express linear/angular momentum in generalized coordinates

$\begin{cases} \begin{bmatrix} v_S \\ \Omega \end{bmatrix} = \begin{bmatrix} J_P \\ J_R \end{bmatrix}\dot q \\ \begin{bmatrix} a_S \\ \Psi \end{bmatrix} = \begin{bmatrix} J_P \\ J_R \end{bmatrix}\ddot q + \begin{bmatrix} \dot J_P \\ \dot J_R \end{bmatrix}\dot q \end{cases}$

⇒ $\begin{bmatrix} \dot P_{S_i} \\ \dot N_{S_i} \end{bmatrix} = \begin{bmatrix} m a_{S_i} \\ \Theta_{S_i}\Psi_{S_i} + \Omega_{S_i}\times\Theta_{S_i}\Omega_{S_i} \end{bmatrix}$

$= \begin{pmatrix} mJ_{S_i} \\ \Theta_{S_i}J_{R_i} \end{pmatrix}\ddot q + \begin{pmatrix} m\dot J_{S_i}\dot q \\ \Theta_{S_i}\dot J_{R_i}\dot q + J_{R_i}\dot q\times\Theta_{S_i}J_{R_i}\dot q \end{pmatrix}$

• virtual displacement in generalized coordinates

$\begin{pmatrix} \delta r_S \\ \delta\Phi \end{pmatrix} = \begin{bmatrix} J_P \\ J_R \end{bmatrix}\delta q$

∴ $0 = \delta W = \sum_{i=1}^{n}\begin{pmatrix} \delta r_{S_i} \\ \delta\Phi_{S_i} \end{pmatrix}^T\left(\begin{bmatrix} \dot P_{S_i} \\ \dot N_{S_i} \end{bmatrix} - \begin{bmatrix} F_{ext,i} \\ T_{ext,i} \end{bmatrix}\right)$

$= \delta q^T\sum_{i=1}^{n}\Big\{\underbrace{\begin{pmatrix} J_{S_i} \\ J_{R_i} \end{pmatrix}^T\begin{pmatrix} mJ_{S_i} \\ \Theta_{S_i}J_{R_i} \end{pmatrix}}_{M(q)}\ddot q + \underbrace{\begin{pmatrix} J_{S_i} \\ J_{R_i} \end{pmatrix}^T\begin{pmatrix} m\dot J_{S_i}\dot q \\ \Theta_{S_i}\dot J_{R_i}\dot q + J_{R_i}\dot q\times\Theta_{S_i}J_{R_i}\dot q \end{pmatrix}}_{b(q,\dot q)} - \underbrace{\begin{pmatrix} J_{S_i} \\ J_{R_i} \end{pmatrix}^T\begin{pmatrix} F_{ext,i} \\ T_{ext,i} \end{pmatrix}}_{g(q)}\Big\}$

$\forall\delta q$

- $M(q)\ddot q + b(q,\dot q) + g(q) = 0$

$M = \sum_{i=1}^{n_b}({}_AJ_{S_i}^T m\,{}_AJ_{S_i} + {}_BJ_{R_i}^T\,{}_B\Theta_{S_i}\,{}_BJ_{R_i})$

$b = \sum_{i=1}^{n_b}\big({}_AJ_{S_i}^T m\,{}_A\dot J_{S_i}\dot q + {}_BJ_{R_i}^T({}_B\Theta_{S_i}\,{}_B\dot J_{R_i}\dot q + {}_B\Omega_{S_i}\times{}_B\Theta_{S_i}\,{}_B\Omega_{S_i})\big)$

$g = \sum_{i=1}^{n_b}(-{}_AJ_{S_i}^T\,{}_AF_{g,i})$

Δ Lagrange II

• Lagrangian $\mathcal{L} = T - U$ (kinetic, potential)

• Lagrangian Equation

$\frac{d}{dt}\left(\frac{\partial\mathcal{L}}{\partial\dot q}\right) - \frac{\partial\mathcal{L}}{\partial q} = \tau$

∵ $U = U(q)$
 $T = T(q,\dot q)$

∴ $\underbrace{\frac{d}{dt}\left(\frac{\partial T}{\partial\dot q}\right) - \frac{\partial T}{\partial q}}_{\text{inertial force}} + \underbrace{\frac{\partial U}{\partial q}}_{\text{gravity vector}} = \tau$

Δ Kinetic energy

- joint space $T = \frac12\dot q^T M\dot q$
- bodies space

$T = \sum_{i=1}^{n_b}\left(\frac12 m_i\,{}_A\dot r_{S_i}^T\,{}_A\dot r_{S_i} + \frac12\,{}_B\Omega_{S_i}^T\,{}_B\Theta_{S_i}\,{}_B\Omega_{S_i}\right)$

w/ $\dot r_{S_i} = J_{S_i}\dot q$
 $\Omega_{S_i} = J_{R_i}\dot q$

∴ $T(q,\dot q) = \frac12\dot q^T\underbrace{\left(\sum_{i=1}^{n_b}(J_{S_i}^T m J_{S_i} + J_{R_i}^T\Theta_{S_i}J_{R_i})\right)}_{M(q)}\dot q$

Δ potential energy

- gravity

$F_{g_i} = m_i g_I e_z$ ⇒ $U_g = -\sum_{i=1}^{n_b} r_{S_i}^T F_{g_i}$

- spring

$F_{S_j} = k_j(\|r - r_0\| - d_0)\frac{r - r_0}{\|r - r_0\|}$

⇒ $U_{S_j} = \frac12 k_j(d(q) - d_0)^2$

Δ External Force

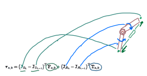
• given $n_{f,ext}$ F_j T_k

• $\underbrace{\tau_{F,ext}}_{\text{joint space}} = \sum_{j=1}^{n_f} J_{P,j}^T\underbrace{F_j}_{\text{task space}}$

• $\tau_{T,ext} = \sum_{k=1}^{n_m} J_{R,k}^T T_k$

actuator torques

$\tau_{a,k} = (J_{S_k} - J_{S_{k-1}})^T F_{a,k} + (J_{R_k} - J_{R_{k-1}})^T T_{a,k}$



Δ Dynamic Case Study
(w/ cart pendulum)

△ Dynamic Control

$M(q)\ddot{q} + b(q,\dot{q}) + g(q) = \tau + J_c^T F_c$

How do we get this

Actuator control $\xrightarrow{VA}$ Actuator $\xrightarrow{\tau}$ Robot Dynamics

torque sensor

$q, \dot{q}, \ddot{q}$

$(q^* \ \dot{q}^*)$

τ^*

△ position-based control
- don't care about dynamics
- high gain PID : good joint-level tracking
- disturbances are compensated by PID
- cannot control force directly
  ⇒ interaction force can only be controlled w/ compliant surface

△ integrate force-feedback (Dynamic control)
- active regulation of system dynamics
- model-based load compensation
- interaction force control.

△ Joint Impedance Control

- $M(q)\ddot{q} + b(q,\dot{q}) + g(q) = \tau$
- get desired τ

△ torque as functions of P/V error

$\tau^* = k_p(q^* - q) + k_d(\dot{q}^* - \dot{q})$

can think of it as spring force & damping

$\Rightarrow M(q)\ddot{q} + b(q,\dot{q}) + g(q) = k_p(q^* - q) + k_d(\dot{q}^* - \dot{q})$

static offset due to gravity (when static $M\ddot{q}, b = 0, \dot{q}^* - \dot{q} = 0$)

△ impedance control & gravity compensation

$\tau^* = k_p(q^* - q) + k_d(\dot{q}^* - \dot{q}) + \hat{g}(q)$

configuration dependent eg CoG shifts — estimated

△ independent of configuration inverse dynamics control

- $\tau = \hat{M}(q)\ddot{q}^* + \hat{b}(q,\dot{q}) + \hat{g}(q)$

get $\ddot{q}^*$, & throw into this EoD, and get the final τ.

based on inverse kinematics → inverse dynamics

result in
- assume no dynamic modelling error
$I\ddot{q} = \ddot{q}^* = k_p(q^* - q) + k_d(\dot{q}^* - \dot{q})$
&
$\omega = \sqrt{k_p} \quad D = \frac{k_d}{2\sqrt{k_p}}$

- describe from task space

$\dot{w}_e = \begin{pmatrix} \ddot{r} \\ \dot{\omega} \end{pmatrix}_e = J_e \ddot{q} + \dot{J}_e \dot{q}$

$\therefore \ddot{q} = J_e^+ (\dot{w}_e - \dot{J}_e \dot{q})$

& similarly, multi-task

- $\ddot{q} = \begin{bmatrix} J_1 \\ \vdots \\ J_{n_T} \end{bmatrix}^+ \left(\begin{bmatrix} \dot{w}_1 \\ \vdots \\ \dot{w}_{n_T} \end{bmatrix} - \begin{bmatrix} \dot{J}_1 \\ \vdots \\ \dot{J}_{n_T} \end{bmatrix} \dot{q} \right)$

parallel

- $\ddot{q} = \sum_{i=1}^{n_T} N_i \ddot{q}_i$ w/

w/ $\ddot{q}_i = (J_i N_i)^+ \left(\dot{w}_i^* - \dot{J}_i \dot{q} - J_i \sum_{k=1}^{i-1} N_k \ddot{q}_k \right)$

- get $\ddot{q}$, & insert back to E.o.M.

△ task-space dynamics

- recall joint space

$M(q)\ddot{q} + b(q,\dot{q}) + g(q) = \tau$

- for end-effector

$\Lambda \dot{w}_e + \mu + p = F_e$

linking the two — link

$\begin{cases} \tau = J_e^T F_e \\ \dot{w}_e = \begin{pmatrix} \ddot{r} \\ \dot{\omega} \end{pmatrix}_e = J_e \ddot{q} + \dot{J}_e \dot{q} \end{cases}$

$\ddot{q} = M^{-1}(\tau - b - g)$

derivation

$\Rightarrow \dot{w}_e = J_e M^{-1}(\tau - b - g) + \dot{J}_e \dot{q}$

$\Rightarrow \dot{w}_e - \dot{J}_e \dot{q} + J_e M^{-1} b + J_e M^{-1} g = J_e M^{-1} \tau$

$\Rightarrow \underbrace{\dot{w}_e - \dot{J}_e \dot{q} + J_e M^{-1} b}_{\mu} + \underbrace{J_e M^{-1} g}_{p} = \underbrace{J_e M^{-1} J_e^T}_{\Lambda^{-1}} F_e$

$\Rightarrow \Lambda \dot{w}_e + \mu + p = F_e$

↳ geometric interpretation : mental ellipsoid (depend on our configuration)

$\therefore$ get $\dot{w}_e^* = k_p E(\chi_e^* - \chi_e)$ $\Delta\chi = E_R(\chi_R)\Delta\chi_R$
$+ k_d(w_e^* - w_e)$
$+ \dot{w}_e(t)$ → feed-forward

determine desired acceleration → task-level force → Jacobi transformation → joint-space torques

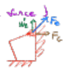
$\tau^* = \hat{J}^T(\hat{\Lambda}_e \dot{w}_e^* + \hat{\mu} + \hat{p})$

△ interaction control

operation space control

(generalized framework to control motion & force)

- $F_c + \Lambda \dot{w}_e + \mu + p = F_e$



△ selection force

$\hat{\Lambda} S_M \dot{w}_e + S_F F_c + \hat{\mu} + \hat{p} = F_e$

$\Rightarrow \tau^* = \hat{J}^T F_e$

$\begin{cases} \Sigma_p = \begin{bmatrix} \sigma_{px} & & \\ & \sigma_{py} & \\ & & \sigma_{pz} \end{bmatrix} \\ \Sigma_F = I_3 - \Sigma_p \end{cases}$ $\sigma \in \{0,1\}$

1 → can move
0 → can apply force

$\begin{cases} \Sigma_R = \begin{bmatrix} \sigma_{Rx} & & \\ & \sigma_{Ry} & \\ & & \sigma_{Rz} \end{bmatrix} \\ \Sigma_{RF} = I_3 - \Sigma_R \end{cases}$

w/ C. transformation between contact local frame ↕ world frame

$\therefore S_M = \begin{bmatrix} C^T \Sigma_p C & 0 \\ 0 & C^T \Sigma_R C \end{bmatrix}$

$S_F = \begin{bmatrix} C^T(I_3 - \Sigma_p)C & 0 \\ 0 & C^T(I_3 - \Sigma_R)C \end{bmatrix}$

$\therefore$ inject back to

$\hat{\Lambda} S_M \dot{w}_e + S_F F_c + \hat{\mu} + \hat{p} = F_e$

△ inverse dynamic as QP

1. classic inverse dynamics

recall

$w = J\dot{q}$

$\dot{w} = J\ddot{q} + \dot{J}\dot{q} \Rightarrow \ddot{q}^* = J^+(\dot{w} - \dot{J}\dot{q})$

$\therefore \tau^* = M(q)\ddot{q}^* + b(q,\dot{q}) + g$
$= M(q)[J^+(\dot{w} - \dot{J}\dot{q})] + b(q,\dot{q}) + g$

2. operation space control

$\tau^* = J_t^T F_e$
$= J_t^T(\Lambda \dot{w}_t^* + \mu + p)$

Not only sol^n

$\therefore \tau^* = J_t^T(\Lambda \dot{w}_t^* + \mu + p) + \underbrace{N(J_t^T)}_{\text{null space}} \tau_0$

τ_0: can change anything

$N(J_t^T)$
$= \underbrace{(I - J^T (J M^{-1} J^T)^{-1} J M^{-1})}_{\text{Mass, weighted pseudo inverse}} \underbrace{\left(\text{w.o. changing desired } \dot{w}\right)}_{\text{proof}}$

$\dot{w} = J\ddot{q} + \dot{J}\dot{q}$
$= JM^{-1}(\tau^* - b - g) + \dot{J}\dot{q}$
$= JM^{-1}(J_t^T(\Lambda \dot{w}_t^* + \mu + p)$
$+ (I - J^T(JM^{-1}J^T)^{-1}JM^{-1})\tau_0$ $= 0$
$- b - g)$
$+ \dot{J}\dot{q}$

- we can also see :

$\begin{cases} \tau^* = M(q)\ddot{q}^* + b(q,\dot{q}) + g \\ \tau^* = J_t^T F_e = J_t^T(\Lambda \dot{w}_t^* + \mu + p) \\ \quad = J_t^T((J_t M^{-1} J_t^T)^{-1} \dot{w} + \mu + p) \end{cases}$

$\Rightarrow \ddot{q} = M^{-1}(\tau^* - b - g)$
$= M^{-1}(J_t^T (J_t M^{-1} J_t^T)^{-1} \dot{w} + \mu + p) - b - g)$

some sort of mass-matrix weighted pseudo-inverse

3. QP

- $\begin{rcases} \tau = M\ddot{q} + b + g \\ J\ddot{q} + \dot{J}\dot{q} = \dot{w} \\ \min \|\ddot{q}\| \\ \min \|\tau\| \end{rcases}$ ↔ classic inverse dynamic

- $[M \ -I]\begin{bmatrix} \ddot{q} \\ \tau \end{bmatrix} + b + g = 0$ — x

$[J_e \ 0]\begin{bmatrix} \ddot{q} \\ \tau \end{bmatrix} + \dot{J}\dot{q} = \dot{w}_e^*$ — x

⇓

$x = \begin{bmatrix} \ddot{q} \\ \tau \end{bmatrix}$

multi-task :

$\min_x \left\| \begin{bmatrix} M & -I \\ J_e & 0 \end{bmatrix} x - \begin{bmatrix} -b-g \\ \dot{w}_e^* - \dot{J}\dot{q} \end{bmatrix} \right\|_2$

priority :

$\min_x \| [J_e \ 0] x - (\dot{w}_e^* - \dot{J}\dot{q}) \|_2$

s.t. $[M \ -I] x - (-b - g) = 0$

△ Least-square Optimization

- $Ax = b \Rightarrow x = A^+ b$

⇕

$\min_x \|Ax - b\|_2$

or

$\min_x \|x\|_2$

s.t. $Ax = b$

- $A_1 x_1 + A_2 x_2 = b$

$\Rightarrow [A_1 \ A_2]\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = b \Rightarrow \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = [A_1 \ A_2]^+ b$

⇕

$\min_{x_1, x_2} \left\| [A_1 \ A_2]\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} - b \right\|_2$

or

$\min_{x_1, x_2} \left\| \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \right\|_2$ s.t. $[A_1 \ A_2]\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = b$

- $A_1 x - b_1 = 0$
  $A_2 x - b_2 = 0$
  — Equal priority
  $x = \begin{bmatrix} A_1 \\ A_2 \end{bmatrix}^+ \begin{bmatrix} b_1 \\ b_2 \end{bmatrix}$

- Hierarchy

$x = A_1^+ b_1 + N(A_1) x_0$

$\min_x \|A_2 x - b_2\|_2$
s.t. $\|A_1 x - b_1\| = c_1$

$A_1(A_1^+ b_1 + N(A_1) x_0)$
$= A_1 A_1^+ b_1 + 0 = b_1$

$\Rightarrow A_2 x - b_2 = 0$

$\Rightarrow A_2(A_1^+ b_1 + N(A_1) x_0) - b_2 = 0$

$\Rightarrow x_0 = [A_2 N(A_1)]^+ [b_2 - A_2 A_1^+ b_1]$

try to fulfill $A_2 x = b_2$

△ case study

△ solving a set of QPs

- w/ different priorities

methods :
- exploit nullspace of tasks w/ higher priority
- everystep = QP w/ constraints
- iteratively nullspace projection

- pseudo-code

n_T = number of tasks
$x = 0$
$N_1 = I$
for $i = 1 \to n_T$ do
  $x_i = (A_i N_i)^+ (b_i - A_i x)$
  $x = x + N_i x_i$
  $N_{i+1} = N([A_1^T \cdots A_i^T]^T)$
endfor

e.g. @ $i = 1$

$x = 0$
$N_1 = I$
$x_1 = (A_1 I)^+ (b_1 - A\,0)$
$= A_1^+ b_1$
$x := 0 + I A_1^+ b_1$
$N_2 = N(A_1)$

@ $i = 2$

$x = A_1^+ b_1$
$N_2 = N(A_1)$
$x_2 = (A_2 N(A_1))^+ [b_2 - A_2 A_1^+ b_1]$
$x := A_1^+ b_1 + N(A_1)[A_2 N(A_1)]^+ [b_2 - A_2 A_1^+ b_1]$

→ use solver (most of the time)