Nov 11, 2017 Elec 442 HW 3 Jason Chan 3205455
Ryan Wang 37414132

1) $R = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}$, $\underline{R} = e^{\theta S x} = \underline{C_f}/\underline{C_i}$

$R_x(\theta) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta_x & -\sin\theta_x \\ 0 & \sin\theta_x & \cos\theta_x \end{bmatrix}$, $R_y(\theta) = \begin{bmatrix} \cos\theta_y & 0 & \sin\theta_y \\ 0 & 1 & 0 \\ -\sin\theta_y & 0 & \cos\theta_y \end{bmatrix}$, $R_z(\theta) = \begin{bmatrix} \cos\theta_z & -\sin\theta_z & 0 \\ \sin\theta_z & \cos\theta_z & 0 \\ 0 & 0 & 1 \end{bmatrix}$

→ $R = R_x(\theta) \cdot R_y(\theta) \cdot R_z(\theta)$
= some combination of rotation in the x, y, z directions
→ solve for Euler angles:
→ $\theta_x = \text{atan2}(r_{32}, r_{33}) = \theta_i$ = rotation in the i-direction
$\theta_y = \text{atan2}(-r_{31}, \sqrt{r_{32}^2 + r_{33}^2}) = \theta_j$ = rotation in j-direction
$\theta_z = \text{atan2}(r_{21}, r_{11}) = \theta_k$ = rotation in k-direction

useful for Q2 matlab code

→ to solve for θ:
trace(R) = diagonal elements = $r_{11} + r_{22} + r_{33}$
trace(R) = $2\cos\theta + 1$ → $\cos\theta = \frac{r_{11} + r_{22} + r_{33} - 1}{2}$

$\sin\theta = r_{21} + r_{32} - r_{31}$
→ $\tan\theta = \frac{\sin\theta}{\cos\theta} = \frac{r_{21} + r_{32} - r_{31}}{\frac{r_{11} + r_{22} + r_{33} - 1}{2}}$ → $\boxed{\theta = \tan^{-1}\left(2 \cdot \left(\frac{r_{21} + r_{32} - r_{31}}{r_{11} + r_{22} + r_{33} - 1}\right)\right)}$

→ to solve for S:

$$\boxed{S = 0.5 \cdot (R - R^T)}$$

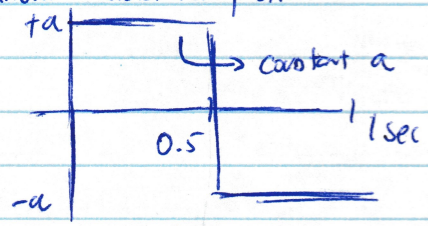
374143L
Jason Chow
32079155

ELEC 442 Hw3

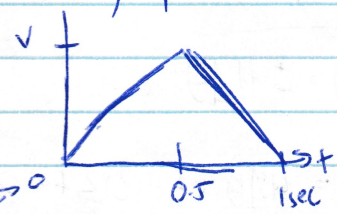
Finding $\bar{v}_n(t)$ and $\dot{\bar{v}}_n(t)$:

$\underline{v}_n(t) = [\dot{\underline{\phi}}_n(t)^T \quad \underline{\omega}_n(t)^T]^T \longrightarrow$ 6×1 matrix 3×1 for $\dot{\underline{\phi}}$, 3×1 for ω_n

Known acceleration profile



velocity profile



basic Kinematic eq. $d = \cancel{v_0 t}^{\,0} + \frac{1}{2} a t^2$

$$\boxed{a = \begin{cases} 4d & \text{for } 0 \le t < 0.5 \\ -4d & \text{for } 0.5 \le t < 1 \end{cases}}$$

$$\Rightarrow \left.\begin{aligned} a_x &= \pm 4 * [\underline{\phi}_{fx} - \underline{\phi}_{ix}] \\ a_y &= \pm 4 * [\underline{\phi}_{fy} - \underline{\phi}_{iy}] \\ a_z &= \pm 4 * [\underline{\phi}_{fz} - \underline{\phi}_{iz}] \end{aligned}\right] \longrightarrow \ddot{\underline{\phi}}_n(t) \text{ for } \ddot{\bar{v}}_n(t)$$

find $\dot{\underline{\phi}}_n(t)$ by integrating

$$\dot{\underline{\phi}}_n(t) = \int \ddot{\underline{\phi}}_n(t) = \begin{cases} 4[\underline{\phi}_{fx,y,z} - \underline{\phi}_{ix,y,z}]t & \text{for } 0 \le t < 0.5 \\ -4[\underline{\phi}_{fx,y,z} - \underline{\phi}_{ix,y,z}]t + \dot{\underline{\phi}}_n(1) & \text{after} \end{cases}$$

$\hookrightarrow C = 4[\underline{\phi}_{fx,y,z} - \underline{\phi}_{ix,y,z}]$

$\hookrightarrow$ first 3×1 for $v_n(t)$

finally find $\underline{\phi}_n(t) \rightarrow$ useful for checking displacement value

$$\underline{\phi}_n(t) = \int \dot{\underline{\phi}}_n(t) = \begin{cases} 2[\underline{\phi}_{fx,y,z} - \underline{\phi}_{ix,y,z}]t^2 & 0 \le t \le 0.5 \\ -2[\underline{\phi}_{fx,y,z} - \underline{\phi}_{ix,y,z}]t^2 + \dot{\underline{\phi}}_n(t) - \ddot{\underline{\phi}}_n(t)/4 & \text{after} \end{cases}$$

$\ddot{\underline{\phi}}_n(t)/4 \rightarrow C$

Note, this basic Kinematic function is used to find $q(t)$

$$\boxed{q_{i+1} = q_i + \dot{q}_i \Delta t + \frac{1}{2} \ddot{q}_i \Delta t^2}$$

find $\omega_n(t)$ via similar method

need to know Euler Angles, find using method described in question 1.

once $\Delta\theta_{x,y,z}$ is known

$$\dot{\omega}_n(t) = \pm 4\Delta\theta$$

$$\omega_n(t) = 4\Delta\theta * t \quad 0 \le t < 0.5$$
$$-4\Delta\theta * t + 4\Delta\theta \rightarrow c \quad 0.5 \le t \le 1$$

$$\theta(t) = 2\Delta\theta t^2 \quad 0 \le t < 0.5$$
$$= -2\Delta\theta t^2 + 4\Delta\theta t - \Delta\theta$$

$$V_n = [\dot{\phi}_n(t)^T \ \omega_n(t)^T]^T$$

$$\dot{V}_n = [\ddot{\phi}_n(t)^T \ \dot{\omega}_n(t)^T]^T$$

Computing the Jacobian:

$$J(q) = \begin{bmatrix} \underset{\underline{k}_0}{\underline{k}_0 \times (\underline{O}_6 - \underline{O}_0)} & \underset{\underline{k}_1}{\underline{k}_1 \times (\underline{O}_6 - \underline{O}_1)} & \underset{\underline{k}_2}{\underline{k}_2 \times (\underline{O}_6 - \underline{O}_2)} & \underset{\underline{k}_3}{\underline{k}_3 \times (\underline{O}_6 - \underline{O}_3)} & \underset{\underline{k}_4}{\underline{k}_4 \times (\underline{O}_6 - \underline{O}_4)} & \underset{\underline{k}_5}{\underline{k}_5 \times (\underline{O}_6 - \underline{O}_5)} \end{bmatrix}$$

computing $\dot{q}(t) = [J(q(t))]^{-1} V_n(t)$ – joint velocities

$$V_n(t) = \begin{bmatrix} \dot{\underline{O}}_n \\ \underline{\omega}_n \end{bmatrix}, \quad J(q)^{-1} = \text{inverse}(J(q))$$

- since we solved for $V_n(t)$ for the end effector, this gives us the entire system joint variables wrt time: $\underset{(6\times1)}{\dot{q}(t)} = \underset{(6\times6)}{J(q(t))^{-1}} \cdot \underset{(6\times1)}{V_n(t)}$

Computing $\ddot{q}(t)$ – joint accelerations

$$\ddot{q}(t) = [J(q(t))]^{-1} \left[\dot{V}_n(t) - \dot{J}(q(t))\, \dot{q}(t) \right]$$

inverse Jacobian

$\begin{bmatrix} \ddot{\underline{O}}_n \\ \dot{\underline{\omega}}_n \end{bmatrix}$

derivative of Jacobian

joint velocities.

Computing q_{i+1} – the next set of joint variables

$$q_{i+1} \simeq q_i + \dot{q}_i \Delta t + \frac{1}{2} \ddot{q}_i (\Delta t)^2$$

$$\dot{\underline{J}}(q(t)) = [\dot{J}_1, \dot{J}_2, \dot{J}_3, \dot{J}_4, \dot{J}_5, \dot{J}_6]$$

Computing $\dot{\underline{J}}(q(t))$

(derivative of Jacobian)

$$\dot{J}_1 = \begin{bmatrix} (\underline{w}_0 \times \underline{C}_0 k) \times (\underline{Q}_6 - \underline{Q}_0) + \underline{k}_0 \times (\dot{\underline{Q}}_6 - \dot{\underline{Q}}_0) \\ w_0 \times \underline{C}_0 k \end{bmatrix}$$

$$\dot{J}_2 = \begin{bmatrix} (\underline{w}_1 \times C_1 k) \times (\underline{Q}_6 - \underline{Q}_1) + \underline{k}_1 \times (\dot{\underline{Q}}_6 - \dot{\underline{Q}}_1) \\ w_1 \times \underline{C}_1 k \end{bmatrix}$$

$$\dot{J}_3 = \begin{bmatrix} (\underline{w}_2 \times \underline{C}_2 k) \times (\underline{Q}_6 - \underline{Q}_2) + \underline{k}_2 \times (\dot{\underline{Q}}_6 - \dot{\underline{Q}}_2) \\ w_2 \times \underline{C}_2 k \end{bmatrix}$$

$$\vdots$$

→ $\underline{C}_0 k = \underline{k}_0$, $\underline{C}_1 k = \underline{k}_1$, ...

→ $\dot{\underline{Q}}_n$ & $\underline{w}_n$ can be computed because we know $\dot{q}_n(t)$ for all joint variables. Therefore, with knowing all of the time derivatives for all joint variables, we can use the formula $[\dot{q}_n(t)]^{-1} \cdot J(q(t)) = V_n(t) = [\dot{\underline{Q}}_n^T \ \underline{w}_n^T]^T$ to find each joint angular & translational velocities