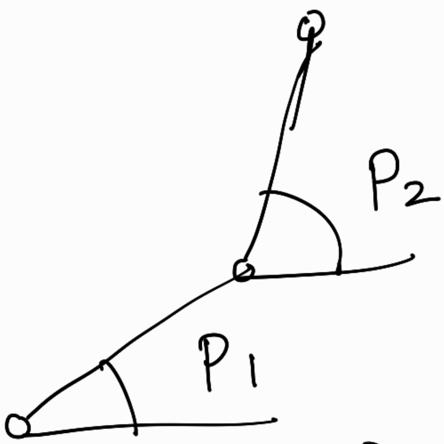
Task 0 :-

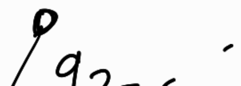
measured values –

$l_1 = 0.1\ m$ link lengths.

$l_2 = 0.1\ m$

The given OSACE setup is a remotely driven manipulator.

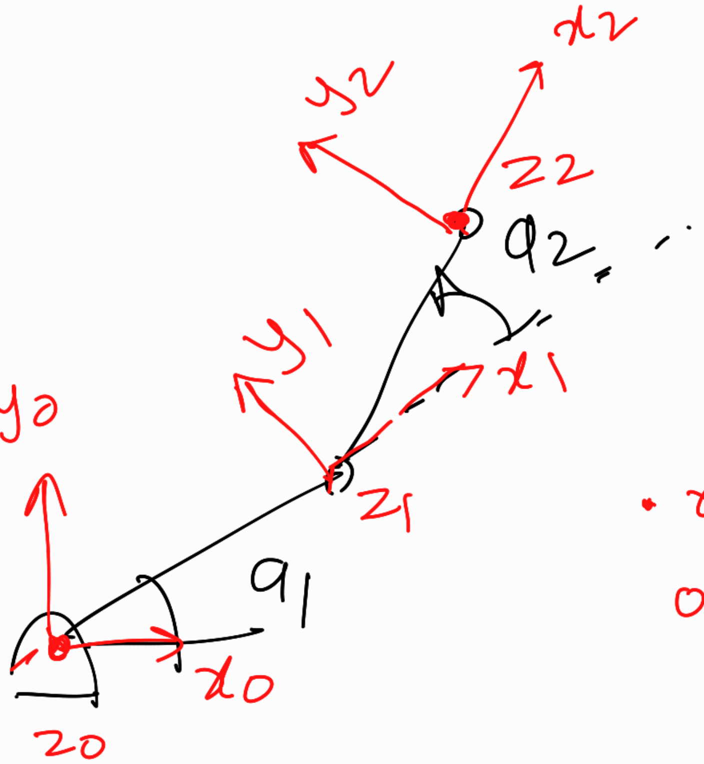




But angles provided to us were q_1 and q_2.

DH parameters are calculated using the axis defined according to DH convention.

DH convention defines angles and distances relative to previous frame and not with respect to fixed frame



• represents out of plane.

Link	a_i	d_i	α_i	θ_i
1	l_1	0	0	q_1^*
2	l_2	0	0	q_2^*

$$A_1 = \begin{bmatrix} \cos q_1 & -\sin q_1 & 0 & l_1 \cos q_1 \\ \sin q_1 & \cos q_1 & 0 & l_1 \sin q_1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_2 = \begin{bmatrix} \cos q_2 & -\sin q_2 & 0 & l_2 \cos q_2 \\ \sin q_2 & \cos q_2 & 0 & l_2 \sin q_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T_0^2 = A_1 A_2$$

$$= \begin{bmatrix} c_{12} & -s_{12} & 0 & l_1 c_1 + l_2 s_{12} \\ s_{12} & c_{12} & 0 & l_1 s_1 + l_2 s_{12} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$x = l_1 c_1 + l_2 c_{12} \quad \text{—(1)}$$

$$y = l_1 s_1 + l_2 s_{12}$$

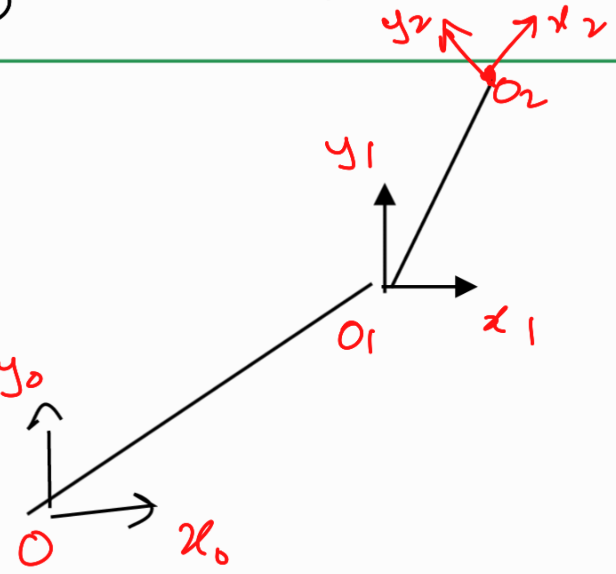
Since we were given q_1 and q_2 in stm32 ide. we use eq(1) for forward kinematics.

But for Jacobian derivation cannot use the DH parameters and Homogenous matrix derived above.

$$\boxed{R_0^2 = R_0^1 R_1^2}$$

If the second rotation is performed relative to fixed frame.

$$\boxed{d_0^2 = d_0^1 + d_1^2}$$



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

$$R_1^2 = \begin{bmatrix} c(p_2) & -s(p_2) & 0 \\ s(p_2) & c(p_2) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$R_0^2 = R_1^2 R_0^1$$

$$= \begin{bmatrix} cp_2 & -sp_2 & 0 \\ sp_2 & cp_2 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$d_0^1 = \begin{bmatrix} l_1 cp_1 \\ l_1 sp_1 \\ 0 \\ 1 \end{bmatrix}$$

$$d_0^2 = \begin{bmatrix} l_2 cp_2 + l_1 cp_1 \\ l_2 sp_2 + l_1 sp_1 \\ 0 \\ 1 \end{bmatrix}$$

$$H = \begin{bmatrix} cp_2 & -sp_2 & 0 & l_2 cp_2 + l_1 cp_1 \\ sp_2 & cp_2 & 0 & l_2 sp_2 + l_1 sp_1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$H_0^1 = \begin{bmatrix} 1 & 0 & 0 & l_1 c p_1 \\ 0 & 1 & 0 & l_1 s p \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$H_1^2 = \begin{bmatrix} cp_2 & -sp_2 & 0 & l_2 cp_2 \\ sp_2 & cp_2 & 0 & l_2 sp_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\dot{d}_0^n = \sum_{i=1}^{n} \frac{\partial d_0^n}{\partial p_i} (\dot{p}_i) \quad \text{——} (2)$$

$\dfrac{\partial d_0^n}{\partial p_i}$ is the i^{th} column of J_v

$$d_0^n = d_0^{i-1} + d_{i-1}^n$$

$$\dot{d}_0^n = \dot{d}_{i-1}^n \quad \text{——} (3)$$

assuming d_0^{i-1} as constant

if i^{th} joint is

only rotated.

$$\dot{d}_{i-1}^{n} = \boxed{(\dot{p}_i - \dot{p}_{i-1})} k \times d_{i-1}^{n} \quad -(4)$$

Because i^{th} joint is only rotated. Analogous to $\dot{q}$ (relative angle).

From (3) and (4)

$$\dot{d}_0^n = (\dot{p}_i - \dot{p}_{i-1}) z_{i-1} \times (O_n - O_{i-1}) \quad -(5)$$

From (2) and (5)

$$\dot{d}_0^n = \sum_{i=1}^{n} (\dot{p}_i - \dot{p}_{i-1}) z_{i-1} \times (O_n - O_{i-1})$$

Expanding for 2R planar remotely driven manipulator.

$\to 0$

$$\dot{d}_0^n = \dot{p}_1 z_0 \times (O_n - O_0) - \cancel{\dot{p}_0} (z_0) \times (O_n - O_0)$$

$$+ \dot{p}_2 z_1 \times (O_n - O_1) - \dot{p}_1 z_1 \times (O_n - O_1)$$

$$\begin{bmatrix} \dot{d}_0^n \\ \\ \omega_0^n \end{bmatrix} = \begin{bmatrix} z_0 \times (O_n - O_0) - z_1 \times (O_n - O_1) & z_1 \times (O_n - O_1) \\ \\ z_0 & z_1 \end{bmatrix} \times \begin{bmatrix} \dot{p}_1 \\ \dot{p}_2 \end{bmatrix}$$

$$= \begin{bmatrix} -l_1 s p_1 & -l_2 s p_2 \\ l_1 c p_1 & l_2 c p_2 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} \dot{p}_1 \\ \dot{p}_2 \end{bmatrix}$$

But since were given
q_1 and q_2

$P_2 = q_1 + q_2$

Task 1

$x = x_0 + a\cos(t - \alpha)$

$y = y_0 + a\sin(t - \alpha)$

represents equation of
circle. with

center x_0, y_0.

by adjusting α we
can choose initial position
of manipulator.

from x, y q_1 and q_2
are found using Inverse

kinematics.

$$\tau = K_p(\theta_d - \theta) + K_d(\dot{\theta}_d - \dot{\theta})$$

controller was used to track the circle.

$$i = \frac{\tau}{K_t}$$

2/ $x = l_1 cq_1 + l_2 cq_2$

$y = l_1 sq_1 + l_2 sq_2$

$\tau = J^T F$

$$\begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix} = \begin{bmatrix} -l_1 sq_1 & -l_2 sq_2^* \\ l_1 cq_1 & l_2 cq_2^* \end{bmatrix}^T \begin{bmatrix} F_x \\ F_y \end{bmatrix}$$

q_2^* is ~~to~~ absolute angles and q_1 & q_2 are relative frame angles.

$$= \begin{bmatrix} -l_1 sq_1 & -l_2 s(q_1+q_2) \\ l_1 cq_1 & l_2 c(q_1+q_2) \end{bmatrix}^T \begin{bmatrix} F_x \\ F_y \end{bmatrix}$$

In our case, $q_1 = 0$, $q_2 = \pi/2$

$$= \begin{bmatrix} -l_1 sq_1 & l_1 cq_1 \\ -l_2 s(q_1+q_2) & l_2 c(q_1+q_2) \end{bmatrix} \begin{bmatrix} F_x \\ F_y \end{bmatrix}$$

$$= \begin{bmatrix} 0 & l_1 \\ -l_2 & 0 \end{bmatrix} \begin{bmatrix} 0 \\ -mg \end{bmatrix}$$

$$= \begin{bmatrix} 0 & 0.1 \\ -0.1 & 0 \end{bmatrix} \begin{bmatrix} 0 \\ \frac{-287}{1000} \times 9.81 \end{bmatrix}$$

$$\begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix} = \begin{bmatrix} -0.28 \\ 0 \end{bmatrix} \text{ in Nm}$$

Date / /
Page

Task-3 :- We want to behave the end-effector as a virtual spring about a specific point say (x_0, y_0)

Therefore, if someone (by applying Force/Torque) changes the position of end-effector it should come at its mean position by its own means we have to calculate the required torque that will be needed to do so.

=) We know that

$$\boxed{\tau = J^T F}$$

Where, to behave as a spring,

$$\boxed{\begin{aligned} F_x &= K_x(x - x_0) \\ F_y &= K_y(y - y_0) \end{aligned}}$$

We are considering K_x & K_y are the two stiffness constant which helps to behave robot as a spring in z-plane.

Finally, K will be diagonal matrix of 2x2 $= \begin{bmatrix} K_x & 0 \\ 0 & K_y \end{bmatrix}$

Where Jacobian for robot configuration will be

$$J = \begin{bmatrix} -a_1 \sin\theta_1 & -a_2 \sin(\theta_2) \\ a_1 \cos\theta_1 & a_2 \cos(\theta_2) \end{bmatrix}$$

Where, $\boxed{a_1 = a_2 = 0.1\,m}$

and $\boxed{\theta_1 = q_1 \text{ \& } \theta_2 = q_2 + q_1}$

$\downarrow$

relative angles.

Hence, $$J = \begin{bmatrix} 0.1 \sin q_1 & -0.1 \sin(q_1+q_2) \\ 0.1 \cos q_1 & 0.1 \cos(q_1+q_2) \end{bmatrix}$$

$$\boxed{\tau = J^T F}$$

Hence, $$\boxed{\tau = \begin{bmatrix} 0.1 \sin q_1 & 0.1 \cos q_1 \\ -0.1 \sin(q_1+q_2) & 0.1 \cos(q_1+q_2) \end{bmatrix} \times \begin{bmatrix} k_x & 0 \\ 0 & k_y \end{bmatrix} \times \begin{bmatrix} x - x_0 \\ y - y_0 \end{bmatrix}}$$

Date / /
Page

Key result & insight :-

1) Since we don't know the damping ~~term~~ value and Inertial term of gearbox therefore, we have neglected both of these terms.

2) As we didn't consider the damping terms, therefore there are some error in going to mean value but as you can see from video it almost trying to go it's mean position and behaving as a virtual spring.