Introduction to Robotics

Manipulation and Programming

Unit 2: Kinematics

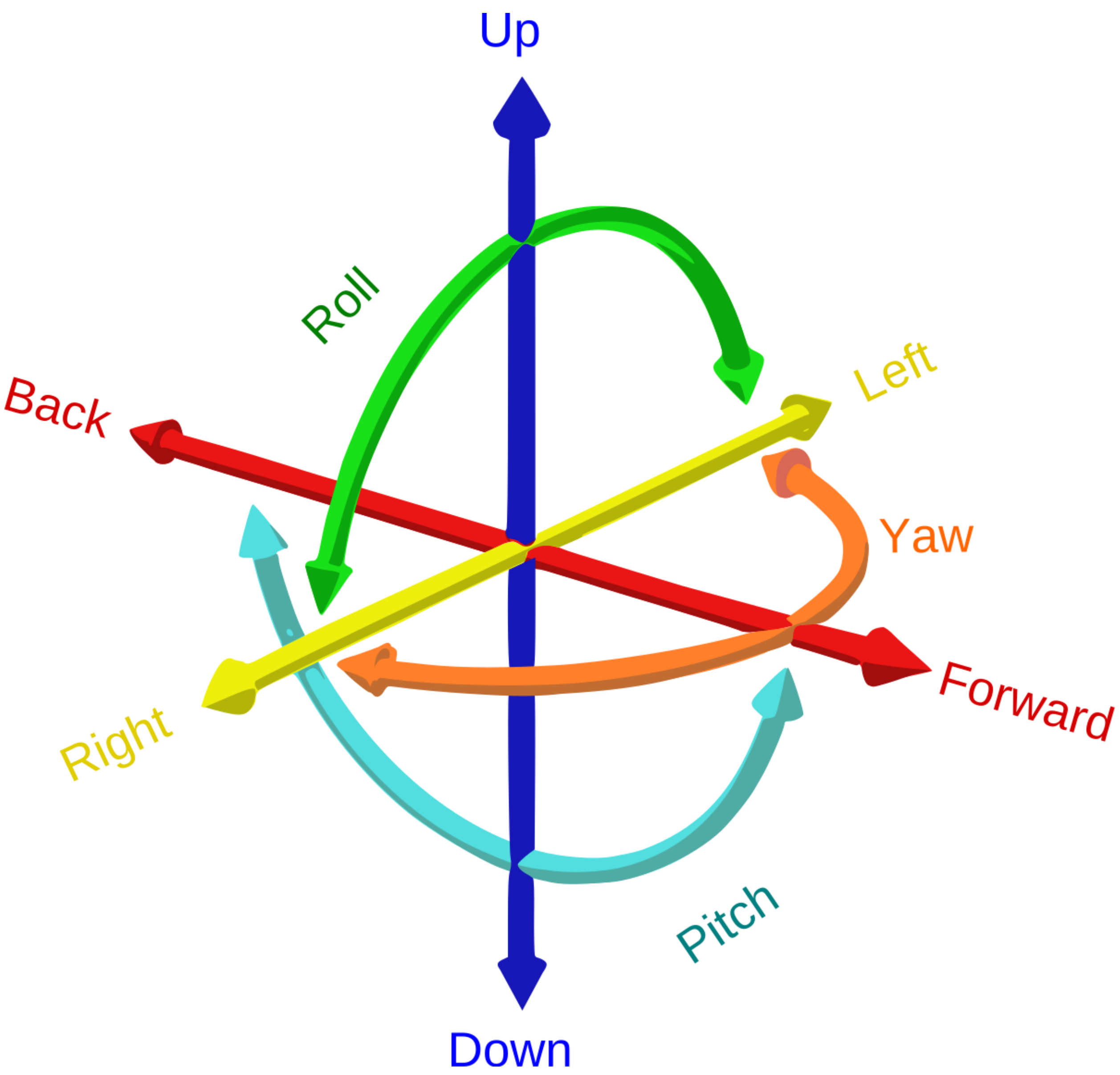
JACOBIAN MATRIX

DR. ERIC CHOU

IEEE SENIOR MEMBER

Objectives

- Jacobian Matrix
- Velocity Analysis for Robots.


Up
Roll
Back
Left
Yaw
Right
Forward
Pitch
Down

Jacobian Matrix

SECTION 1

Kinematic Relations

$$\theta = \begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \\ \theta_4 \\ \theta_5 \\ \theta_6 \end{bmatrix} \qquad X = \begin{bmatrix} x \\ y \\ z \\ \phi \\ \theta \\ \varphi \end{bmatrix}$$

X=FK(θ)

θ =IK(X)

Joint Space

Task Space

- Location of the tool can be specified using a joint space or a cartesian space description

Velocity Relations

- Relation between joint velocity and cartesian velocity.
- JACOBIAN ***matrix J(θ)***

$$\begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \\ \dot{\theta}_3 \\ \dot{\theta}_4 \\ \dot{\theta}_5 \\ \dot{\theta}_6 \end{bmatrix} \quad \begin{matrix} \dot{X} = J(\theta)\dot{\theta} \\ \Longrightarrow \\ \\ \Longleftarrow \\ \dot{\theta} = J^{-1}(\theta)\dot{X} \end{matrix} \quad \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \omega_x \\ \omega_y \\ \omega_z \end{bmatrix}$$

Joint Space

Task Space

Jacobian

- Suppose a **position** and **orientation** vector of a manipulator is a function of 6 joint variables: (from forward kinematics)

$$X = h(q)$$

$$X = \begin{bmatrix} x \\ y \\ z \\ \phi \\ \theta \\ \varphi \end{bmatrix} = h\left(\begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \\ q_5 \\ q_6 \end{bmatrix}\right)_{6\times 1} = \begin{bmatrix} h_1(q_1, q_2, \cdots, q_6) \\ h_2(q_1, q_2, \cdots, q_6) \\ h_3(q_1, q_2, \cdots, q_6) \\ h_4(q_1, q_2, \cdots, q_6) \\ h_5(q_1, q_2, \cdots, q_6) \\ h_6(q_1, q_2, \cdots, q_6) \end{bmatrix}_{6\times 1}$$

Jacobian Matrix

Forward kinematics

$$X_{6\times1} = h(q_{n\times1}) \Rightarrow \dot{X}_{6\times1} = \frac{d}{dt}h(q_{n\times1}) = \frac{dh(q)}{dq}\frac{dq}{dt} = \frac{dh(q)}{dq}\dot{q}$$

$$\dot{X}_{6\times1} = J_{6\times n}\dot{q}_{n\times1}$$

$$J = \frac{dh(q)}{dq}$$

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} = \boxed{\left[\frac{dh(q)}{dq}\right]_{6\times n}} \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \vdots \\ \dot{q}_n \end{bmatrix}_{n\times1}$$

Jacobian Matrix

- Jacobian is a function of q, it is not a constant!

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} = \left[\frac{dh(q)}{dq}\right]_{6\times n} \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \vdots \\ \dot{q}_n \end{bmatrix}_{n\times 1} \longrightarrow J = \left(\frac{dh(q)}{dq}\right)_{6\times n} = \begin{bmatrix} \frac{\partial h_1}{\partial q_1} & \frac{\partial h_1}{\partial q_2} & \cdots & \frac{\partial h_1}{\partial q_n} \\ \frac{\partial h_2}{\partial q_1} & \frac{\partial h_2}{\partial q_2} & \cdots & \frac{\partial h_2}{\partial q_n} \\ \vdots & \vdots & \vdots & \vdots \\ \frac{\partial h_6}{\partial q_1} & \frac{\partial h_6}{\partial q_2} & \cdots & \frac{\partial h_6}{\partial q_n} \end{bmatrix}_{6\times n}$$

Jacobian Matrix

$$\dot{X} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} = \begin{bmatrix} V \\ \Omega \end{bmatrix}$$

Linear velocity

$$V = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix}$$

Angular velocity

$$\Omega = \begin{bmatrix} \omega_x = \dot{\phi} \\ \omega_y = \dot{\theta} \\ \omega_z = \dot{\psi} \end{bmatrix} \quad \dot{q}_{n\times 1} = \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \vdots \\ \dot{q}_n \end{bmatrix}_{n\times 1}$$

The Jacbian Equation

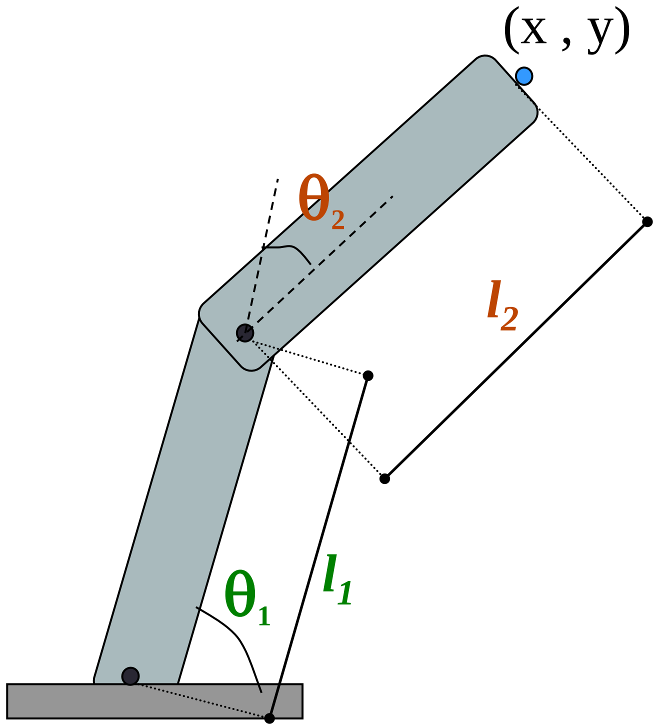
$$\dot{X} = J_{6\times n}\dot{q}_{n\times 1}$$

Example: 2-DOF planar robot arm
Given *l1*, *l2* , Find: Jacobian

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} l_1 \cos\theta_1 + l_2 \cos(\theta_1 + \theta_2) \\ l_1 \sin\theta_1 + l_2 \sin(\theta_1 + \theta_2) \end{bmatrix} = \begin{bmatrix} h_1(\theta_1, \theta_2) \\ h_2(\theta_1, \theta_2) \end{bmatrix}$$

$$\dot{Y} = \begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} = J \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix}$$

$$J = \begin{bmatrix} \frac{\partial h_1}{\partial \theta_1} & \frac{\partial h_1}{\partial \theta_2} \\ \frac{\partial h_2}{\partial \theta_1} & \frac{\partial h_2}{\partial \theta_2} \end{bmatrix} = \begin{bmatrix} -l_1 \sin\theta_1 - l_2 \sin(\theta_1 + \theta_2) & -l_2 \sin(\theta_1 + \theta_2) \\ l_1 \cos\theta_1 + l_2 \cos(\theta_1 + \theta_2) & l_2 \cos(\theta_1 + \theta_2) \end{bmatrix}$$



Singularity

SECTION 2

Singularities

- The inverse of the jacobian matrix cannot be calculated when

$$det\ [J(\theta)] = 0$$

- Singular points are such values of θ that cause the determinant of the Jacobian to be zero

Singularities

Find the singularity configuration of the 2-DOF planar robot arm

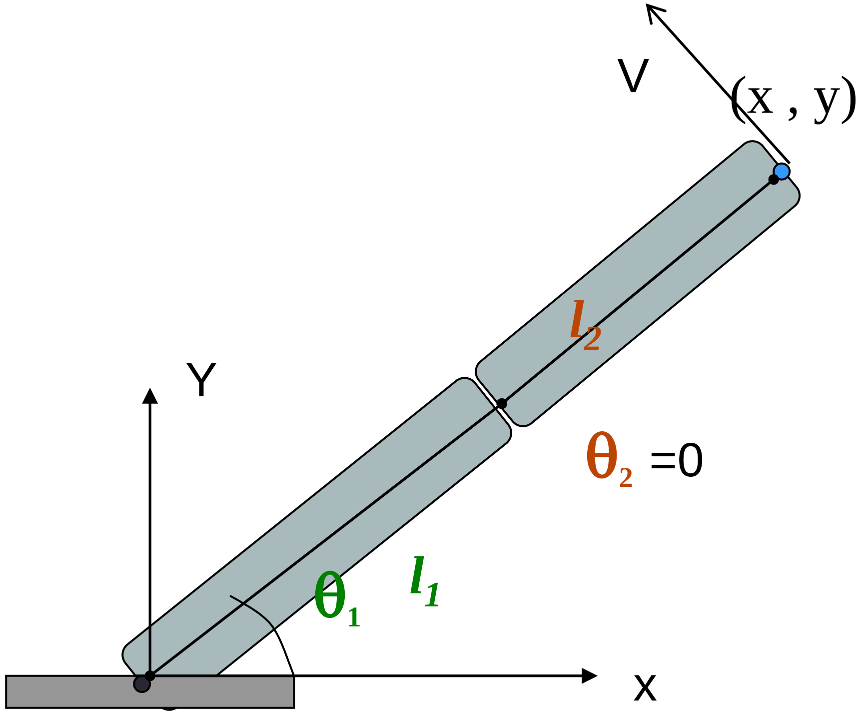
determinant(J)=0 ➡ Not full rank

$$\dot{X} = \begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} = J \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix}$$

$$J = \begin{bmatrix} -l_1 \sin\theta_1 - l_2 \sin(\theta_1 + \theta_2) & -l_2 \sin(\theta_1 + \theta_2) \\ l_1 \cos\theta_1 + l_2 \cos(\theta_1 + \theta_2) & l_2 \cos(\theta_1 + \theta_2) \end{bmatrix}$$

Det(J)=0

$$\theta_2 = 0$$



Jacobian Matrix

Pseudoinverse

- Let A be an $m \times n$ matrix, and let A^+ be the pseudoinverse of A. If A is of full rank, then A^+ can be computed as:

$$A^+ = \begin{cases} A^T[AA^T]^{-1} & m \le n \\ A^{-1} & m = n \\ [A^T A]^{-1} A^T & m \ge n \end{cases}$$

Jacobian Matrix

Example: Find X s.t.

$$\begin{bmatrix} 1 & 0 & 2 \\ 1 & -1 & 0 \end{bmatrix} x = \begin{bmatrix} 3 \\ -2 \end{bmatrix}$$

$$A^{+} = A^{T}[AA^{T}]^{-1} = \begin{bmatrix} 1 & 1 \\ 0 & -1 \\ 2 & 0 \end{bmatrix} \begin{bmatrix} 5 & 1 \\ 1 & 2 \end{bmatrix}^{-1} = \frac{1}{9} \begin{bmatrix} 1 & 4 \\ 1 & -5 \\ 4 & -2 \end{bmatrix}$$

Matlab Command: pinv(A) to calculate

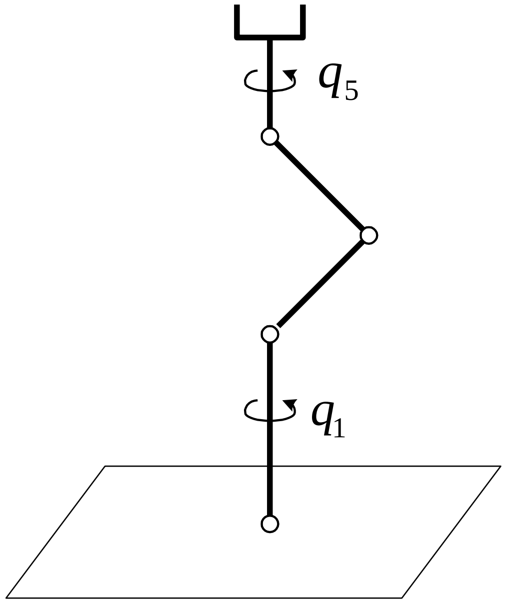
$$x = A^{+}b = \frac{1}{9} \begin{bmatrix} -5 \\ 13 \\ 16 \end{bmatrix}$$

Jacobian Matrix

Inverse Jacobian

$$\dot{X} = J\dot{q} = \begin{bmatrix} J_{11} & J_{12} & \cdots & J_{16} \\ J_{21} & J_{22} & \cdots & J_{26} \\ \vdots & \vdots & \vdots & \vdots \\ J_{61} & J_{62} & \cdots & J_{66} \end{bmatrix} \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \\ \dot{q}_4 \\ \dot{q}_5 \\ \dot{q}_6 \end{bmatrix}$$

$$\dot{q} = J^{-1}\dot{X}$$



Singularity

- rank(J)<n : Jacobian Matrix is less than full rank
- Jacobian is non-invertable
- **Boundary Singularities**: occur when the tool tip is on the surface of the work envelop.
- **Interior Singularities**: occur inside the work envelope when two or more of the axes of the robot form a straight line, i.e., collinear

Singularity

At Singularities:

- the manipulator end effector can’t move in certain directions.
- Bounded End-Effector velocities may correspond to unbounded joint velocities.
- Bounded joint torques may correspond to unbounded End-Effector forces and torques.

Jacobian Matrix Operations

SECTION 3

Jacobian Matrix

- If $A = \begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix}, B = \begin{bmatrix} b_x \\ b_y \\ b_z \end{bmatrix}$
- Then the cross product

$$A \times B = \begin{vmatrix} i & j & k \\ a_x & a_y & a_z \\ b_x & b_y & b_z \end{vmatrix} = \begin{bmatrix} a_y b_z - a_z b_y \\ -(a_x b_z - a_z b_x) \\ a_x b_y - a_y b_x \end{bmatrix}$$

Jacobian Matrix

- The Denavit-Hartenberg matrix T

$$A_i = \begin{bmatrix} \mathrm{c}\theta_{\mathrm{i}} & -\mathrm{c}\alpha_{\mathrm{i}}\mathrm{s}\theta_{\mathrm{i}} & \mathrm{s}\alpha_{\mathrm{i}}\mathrm{s}\theta_{\mathrm{i}} & r_{\mathrm{i}}\mathrm{c}\theta_{\mathrm{i}} \\ \mathrm{s}\theta_{\mathrm{i}} & \mathrm{c}\theta_{\mathrm{i}}\mathrm{c}\alpha_{\mathrm{i}} & -\mathrm{s}\alpha_{\mathrm{i}}\mathrm{c}\theta_{\mathrm{i}} & r_{\mathrm{i}}\mathrm{s}\theta_{\mathrm{i}} \\ 0 & \mathrm{s}\alpha_{\mathrm{i}} & \mathrm{c}\alpha_{\mathrm{i}} & \mathrm{d}_{\mathrm{i}} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

- The transformation matrix T

$$T_i^0 = A_1 A_2 \ldots\ldots A_i$$

Jacobian Matrix (n= n DOF)

$$J = [J_1 \quad J_2 \quad \cdots \quad J_n]$$

where if join (i) is revolute

$$J_i = \begin{bmatrix} Z_{i-1} \times (O_n - O_{i-1}) \\ Z_{i-1} \end{bmatrix}$$

- And if joint (i) is prismatic

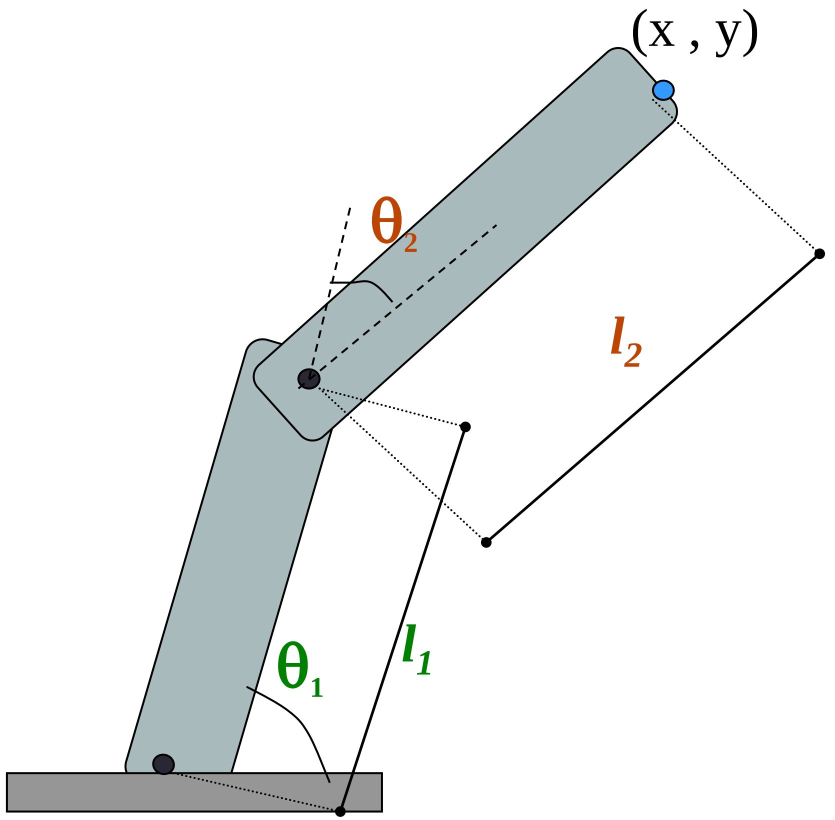
$$J_i = \begin{bmatrix} Z_{i-1} \\ 0 \end{bmatrix}$$

- Where Z_i is the first three elements in the 3rd column of the T_i^0 matrix, and O_i is the first three elements in the 4th column of the T_i^0 matrix

Jacobian Matrix

2-DOF planar robot arm, Given *l1*, *l2* , *Find: Jacobian*

- *Here, n=2,*



Link	a_i	α_i	d_i	θ_i
1	a_1	0	0	θ_1^*
2	a_2	0	0	θ_2^*

* variable

$$A_i = \begin{bmatrix} c\theta_i & -c\alpha_i s\theta_i & s\alpha_i s\theta_i & a_i c\theta_i \\ s\theta_i & c\theta_i c\alpha_i & -s\alpha_i c\theta_i & a_i s\theta_i \\ 0 & s\alpha_i & c\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_1 = \begin{bmatrix} c_1 & -s_1 & 0 & a_1c_1 \\ s_1 & c_1 & 0 & a_1s_1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

$$A_2 = \begin{bmatrix} c_2 & -s_2 & 0 & a_2c_2 \\ s_2 & c_2 & 0 & a_2s_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$Z_0 = Z_1 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

$$T_1^0 = A_1.$$

$$T_2^0 = A_1A_2 = \begin{bmatrix} c_{12} & -s_{12} & 0 & a_1c_1 + a_2c_{12} \\ s_{12} & c_{12} & 0 & a_1s_1 + a_2s_{12} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Where $(\theta_1 + \theta_2)$ denoted by θ_{12} , r_i by a_i and $\cos(\theta_1 + \theta_2)$ by c_{12}

$$O_0 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, O_1 = \begin{bmatrix} a_1\cos\theta_1 \\ a_1\sin\theta_1 \\ 0 \end{bmatrix}, O_2 = \begin{bmatrix} a_1\cos\theta_1 + a_2\cos(\theta_1+\theta_2) \\ a_1\sin\theta_1 + a_2\sin(\theta_1+\theta_2) \\ 0 \end{bmatrix}$$

Jacobian Matrix

2-DOF planar robot arm

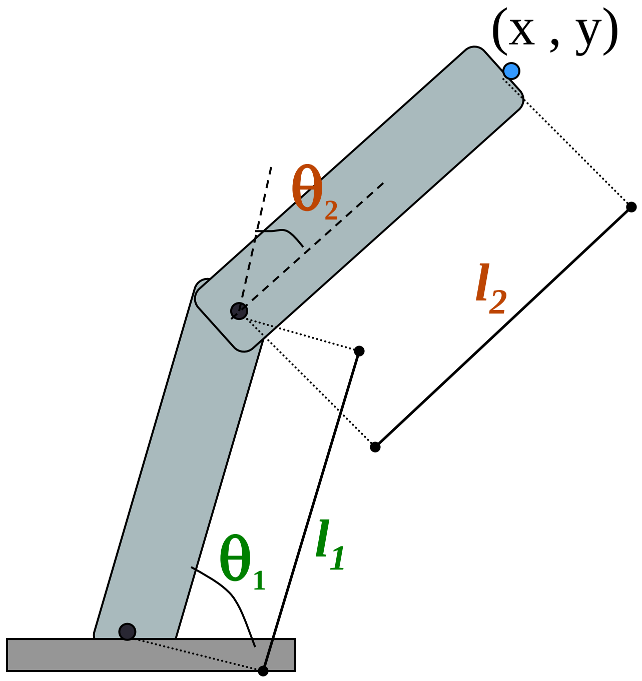
Given ***l1***, ***l2*** **, *Find: Jacobian***

- *Here, n=2*

Link	a_i	α_i	d_i	θ_i
1	a_1	0	0	θ_1^*
2	a_2	0	0	θ_2^*

* variable

$$J_1 = \begin{bmatrix} z_0 \times (o_2 - o_0) \\ z_0 \end{bmatrix}, J_2 = \begin{bmatrix} z_1 \times (o_2 - o_1) \\ z_1 \end{bmatrix}$$



Jacobian Matrix

$$J_1 = \begin{bmatrix} z_0 \times (o_2 - o_0) \\ z_0 \end{bmatrix}$$

$$Z_0 \times (o_2 - o_0) = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \times \begin{bmatrix} a_1 \cos\theta_1 + a_2 \cos(\theta_1 + \theta_2) \\ a_1 \sin\theta_1 + a_2 \sin(\theta_1 + \theta_2) \\ 0 \end{bmatrix}$$

$$= \begin{bmatrix} i & j & k \\ 0 & 0 & 1 \\ a_1 \cos\theta_1 + a_2 \cos(\theta_1 + \theta_2) & a_1 \sin\theta_1 + a_2 \sin(\theta_1 + \theta_2) & 0 \end{bmatrix}$$

$$= \begin{bmatrix} -a_1 \sin\theta_1 - a_2 \sin(\theta_1 + \theta_2) \\ a_1 \cos\theta_1 + a_2 \cos(\theta_1 + \theta_2) \\ 0 \end{bmatrix}$$

Jacobian Matrix

$$J_2 = \begin{bmatrix} z_1 \times (o_2 - o_1) \\ z_1 \end{bmatrix}$$

$$Z_1 \times (o_2 - o_1) = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \times \begin{bmatrix} a_2 \cos(\theta_1 + \theta_2) \\ a_2 \sin(\theta_1 + \theta_2) \\ 0 \end{bmatrix}$$

$$= \begin{bmatrix} i & j & k \\ 0 & 0 & 1 \\ a_2 \cos(\theta_1 + \theta_2) & a_2 \sin(\theta_1 + \theta_2) & 0 \end{bmatrix}$$

$$= \begin{bmatrix} -a_2 \sin(\theta_1 + \theta_2) \\ a_2 \cos(\theta_1 + \theta_2) \\ 0 \end{bmatrix}$$

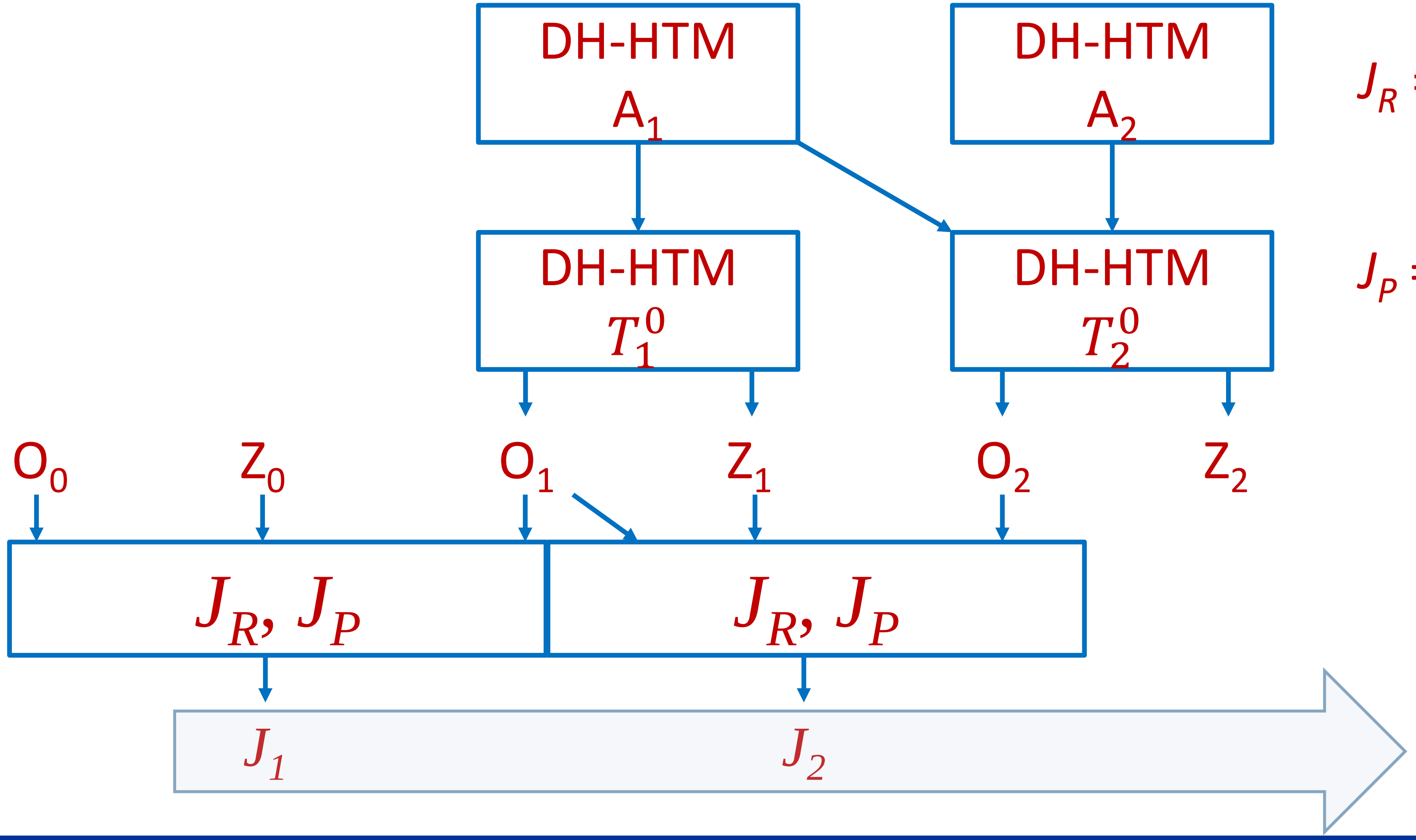
Jacobian Matrix

$$J_1 = \begin{bmatrix} -a_1 \sin\theta_1 - a_2 \sin(\theta_1 + \theta_2) \\ a_1 \cos\theta_1 + a_2 \cos(\theta_1 + \theta_2) \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \qquad J_2 = \begin{bmatrix} -a_2 \sin(\theta_1 + \theta_2) \\ a_2 \cos(\theta_1 + \theta_2) \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

The required Jacobian matrix **J**

$$J = [J_1 \quad J_2]$$

Generation of Jacobian Matrix



$$J_R = \begin{bmatrix} Z_{i-1} \times (O_n - O_{i-1}) \\ Z_{i-1} \end{bmatrix}$$

$$J_P = \begin{bmatrix} Z_{i-1} \\ 0 \end{bmatrix}$$

$$J = [J_1 \quad J_2]$$

Python Example

SECTION 4

Jacobian Matrix Calculation

- Denavit-Hartenberg frame Matrix
- Denavit-Hartenberg series Matrix
- O-Z vectors (Linear/Angular velocities)
- Jacobian Matrix