Jacobian

Introduction

Jacobian – Mapping Operator
Joint & Cartesian/Task Spaces

Kinematics Relations - Joint & Cartesian/Task Spaces

- A robot is often used to manipulate object attached to its tip (end effector).
- The location of the robot tip may be specified using one of the following descriptions:
- ***Joint Space***

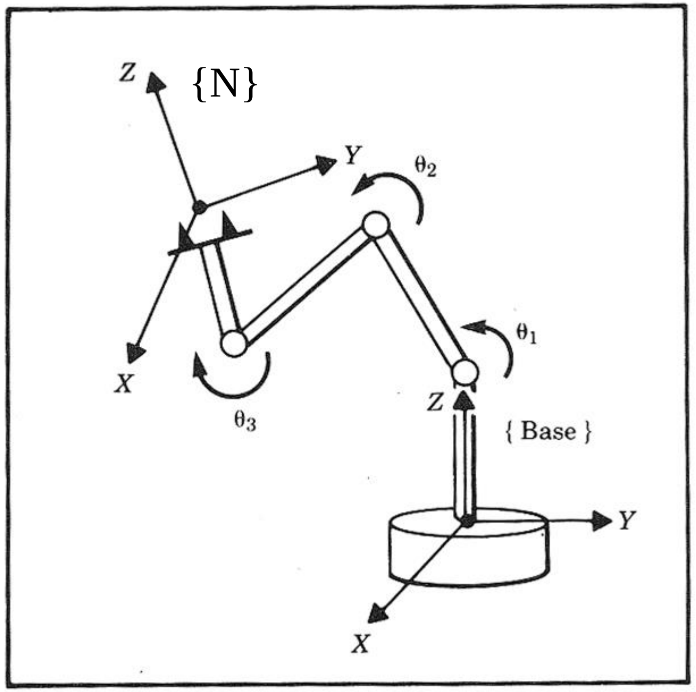
$$\theta = \begin{bmatrix} \theta_1 \\ \theta_2 \\ \vdots \\ d_i \\ \vdots \\ \theta_N \end{bmatrix}$$

- ***Task Space (Cartesian Space)***

$${}^0_N T = \begin{bmatrix} {}^0_N R & {}^0 P_N \\ 0 & 1 \end{bmatrix}$$

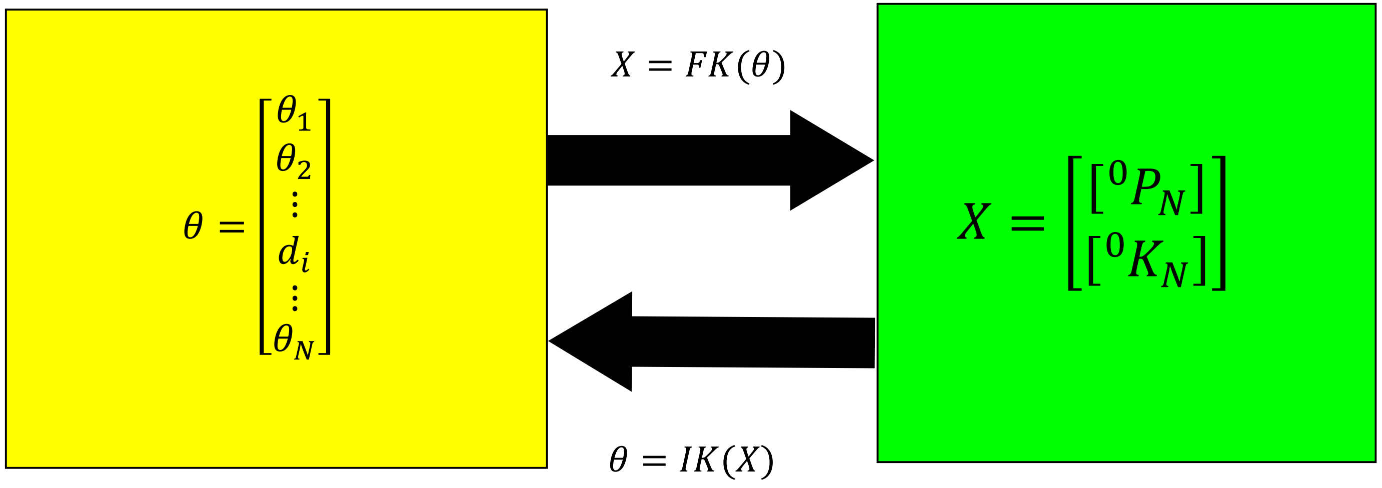
$$X = \begin{bmatrix} [{}^0 P_N] \\ [{}^0 K_N] \end{bmatrix}$$

Equivalent Axis



Kinematics Relations - Forward & Inverse

- The robot kinematic equations relate the two description of the robot tip location



Tip Location in Joint Space

Tip Location in Task/Cartesian/EE Space

Kinematics Relations - Forward & Inverse

$$\underline{\dot{x}} = J(\underline{\theta})\underline{\dot{\theta}}$$

$$\dot{\theta} = \frac{d}{dt}[\theta] = \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \\ \vdots \\ \dot{d}_i \\ \vdots \\ \dot{\theta}_N \end{bmatrix}$$

$$\dot{X} = \frac{d}{dt}[X] = \begin{bmatrix} [\, v_N] \\ [\, \omega_N] \end{bmatrix} = \begin{bmatrix} v_x \\ v_y \\ v_z \\ \omega_x \\ \omega_y \\ \omega_z \end{bmatrix}$$

$$\underline{\dot{\theta}} = J^{-1}(\underline{\theta})\underline{\dot{x}}$$

Tip Velocity in Joint Space

Tip velocity in Task/Cartesian/EE Space

Jacobian – Derivation from First Principals
Velocity Maping

Jacobian Matrix - Introduction

- **The Jacobian is a multi dimensional form of the derivative.**
- Suppose that for example we have 6 functions, each of which is a function of 6 independent variables

$$
\begin{aligned}
y_1 &= f_1(x_1, x_2, x_3, x_4, x_5, x_6) \\
y_2 &= f_2(x_1, x_2, x_3, x_4, x_5, x_6) \\
&\vdots \\
y_6 &= f_6(x_1, x_2, x_3, x_4, x_5, x_6)
\end{aligned}
$$

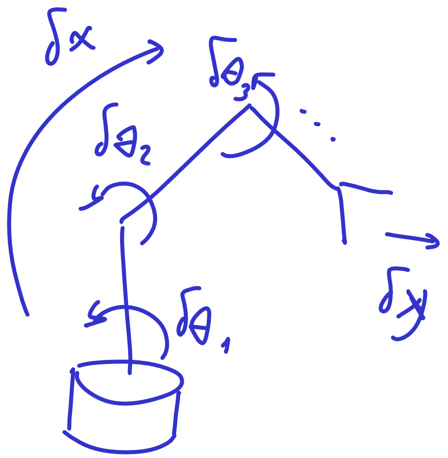
- We may also use a vector notation to write these equations as

$$Y = F(X)$$

Jacobian Matrix - Introduction

- If we wish to calculate the differential of y_i as a function of the differential x_i we use the chain rule to get

$$\delta y_1 = \frac{\partial f_1}{\partial x_1}\delta x_1 + \frac{\partial f_1}{\partial x_2}\delta x_2 + \cdots + \frac{\partial f_1}{\partial x_6}\delta x_6$$
$$\delta y_2 = \frac{\partial f_2}{\partial x_1}\delta x_1 + \frac{\partial f_2}{\partial x_2}\delta x_2 + \cdots + \frac{\partial f_2}{\partial x_6}\delta x_6$$
$$\vdots$$
$$\delta y_6 = \frac{\partial f_6}{\partial x_1}\delta x_1 + \frac{\partial f_6}{\partial x_2}\delta x_2 + \cdots + \frac{\partial f_6}{\partial x_6}\delta x_6$$



- Which again might be written more simply using a vector notation as

$$\delta Y = \frac{\partial F}{\partial X}\delta X$$

Jacobian Matrix - Introduction

- The 6x6 matrix of partial derivative is defined as the Jacobian matrix

$$\delta Y = \frac{\partial F}{\partial X} \delta X = J(X) \delta X$$

- By dividing both sides by the differential time element, we can think of the Jacobian as mapping velocities in X to those in Y

$$\dot{Y} = J(X) \dot{X}$$

- Note that the Jacobian is time varying linear transformation

Jacobian Matrix - Introduction

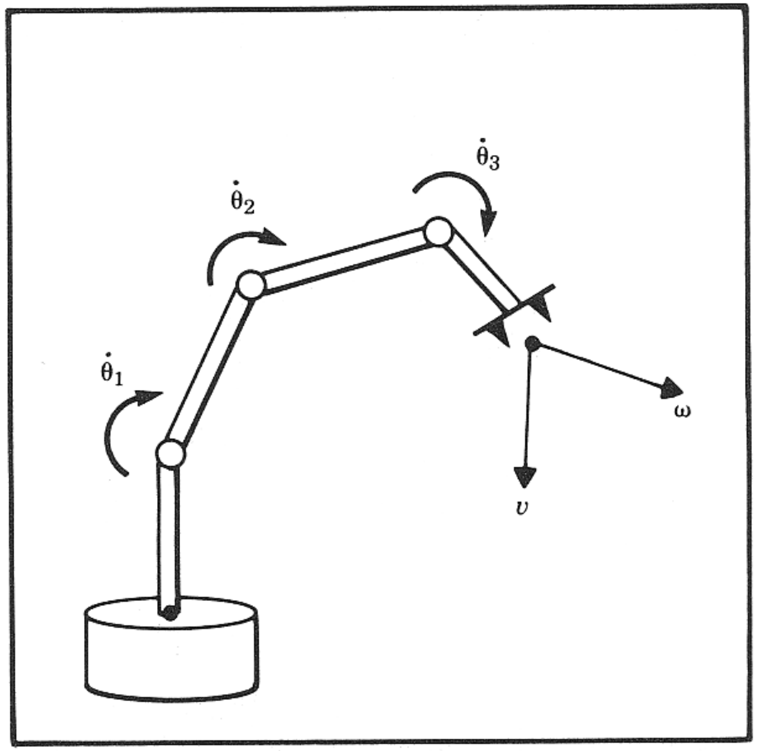
- In the field of robotics the Jacobian matrix describe the relationship between the joint angle rates ($\underline{\dot{\theta}}_N$) and the translation and rotation velocities of the end effector ($\underline{\dot{x}}$). This relationship is given by:

$$\underline{\dot{x}} = J(\underline{\theta})\underline{\dot{\theta}}$$

$$\dot{X} = \frac{d}{dt}[X] = \begin{bmatrix} [\ v_N] \\ [\ \omega_N] \end{bmatrix} = \begin{bmatrix} v_x \\ v_y \\ v_z \\ \omega_x \\ \omega_y \\ \omega_z \end{bmatrix}$$

$$\dot{\theta} = \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \\ \dot{d}_3 \\ \dot{\theta}_4 \\ \dot{\theta}_5 \\ \dot{\theta}_6 \end{bmatrix}$$

$$\underline{\dot{\theta}} = J(\underline{\theta})^{-1}\underline{\dot{x}}$$



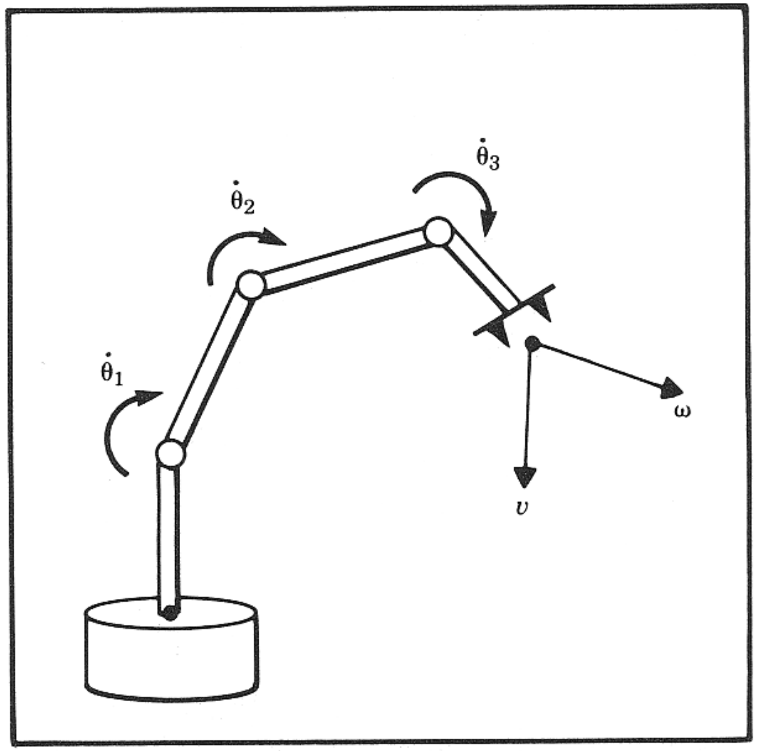
Jacobian Matrix - Introduction

- In the field of robotics the Jacobian matrix describe the relationship between the joint angle rates ($\dot{\underline{\theta}}_N$) and the translation and rotation velocities of the end effector ($\dot{\underline{x}}$). This relationship is given by:

$$\dot{\underline{x}} = J(\underline{\theta})\dot{\underline{\theta}}$$

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} = \begin{bmatrix} & & \\ & J(\underline{\theta}) & \\ & & \end{bmatrix} \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \\ \ldots \\ \dot{\theta}_N \end{bmatrix}$$

$$\dot{\underline{\theta}} = J(\underline{\theta})^{-1}\dot{\underline{x}}$$



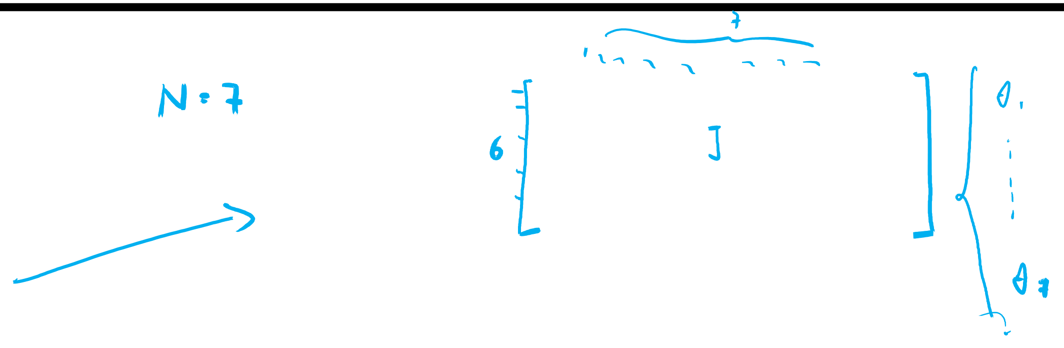
- Note: The Jacobian is a function of joint angle θ meaning that the Jacobian varies as the configuration of the arm changes

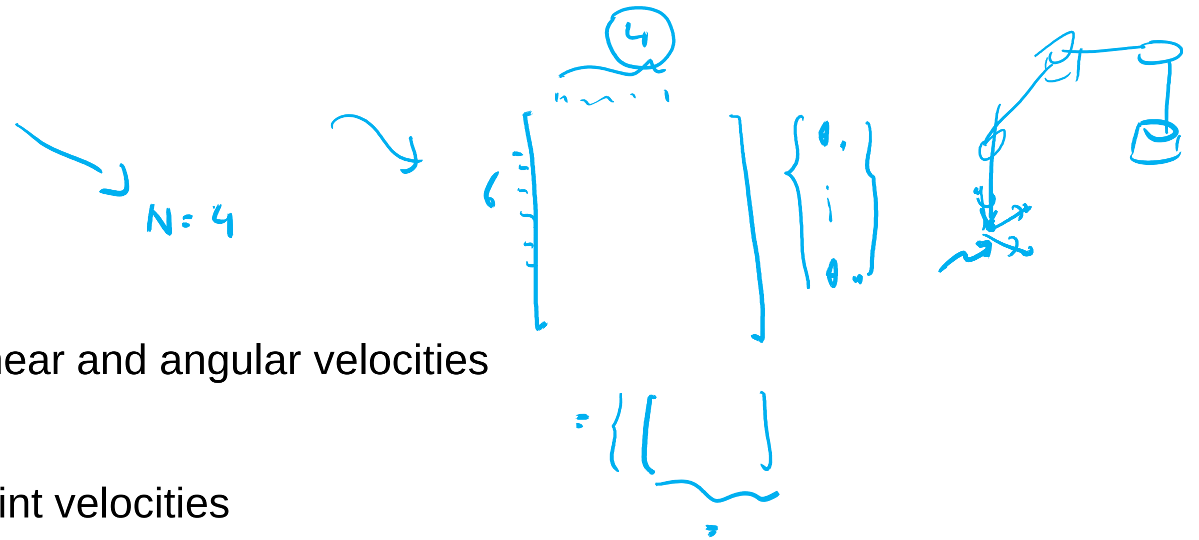
Jacobian Matrix - Introduction

- This expression can be expanded to:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} = \begin{bmatrix} J_v(\underline{\theta}) \\ J_\omega(\underline{\theta}) \end{bmatrix} \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \\ \cdots \\ \dot{\theta}_N \end{bmatrix}$$

6x1 **6xN** **Nx1**



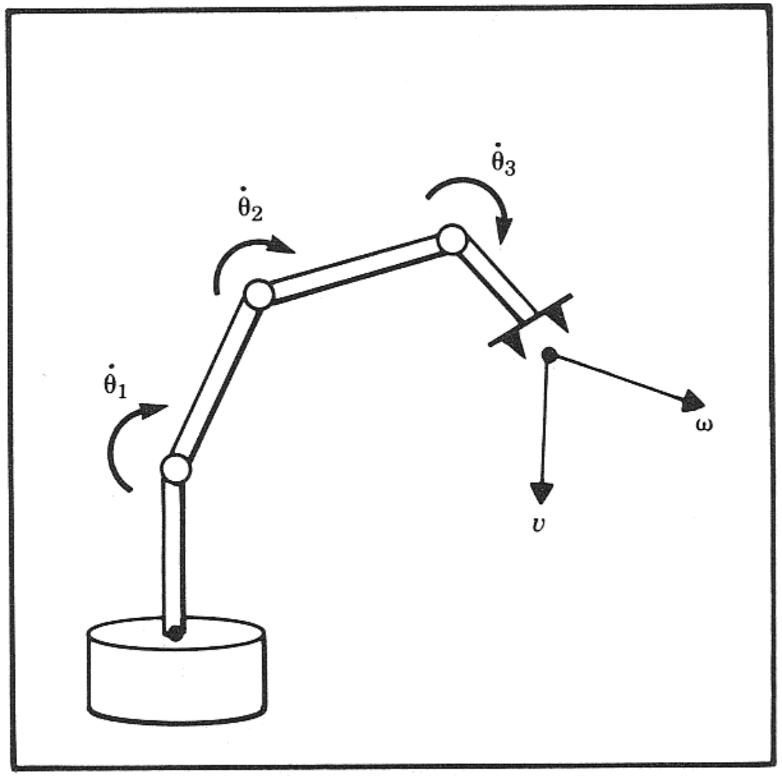


- Where:
  - $\underline{\dot{x}}$ is a 6x1 vector of the end effector linear and angular velocities
  - $J(\underline{\theta})$ is a 6xN Jacobian matrix
  - $\underline{\dot{\theta}}_N$ is a Nx1 vector of the manipulator joint velocities
  - N is the number of joints

Jacobian Matrix - Introduction

- The meaning of **<u>each line</u>** (e.g. the first line) of the Jacobian matrix:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} = \begin{bmatrix} J_{11} & J_{12} & J_{13} & J_{14} & J_{15} & J_{16} \\ & & J_v(\underline{\theta}) & & & \\ & & J_\omega(\underline{\theta}) & & & \end{bmatrix} \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \\ \dots \\ \dot{\theta}_N \end{bmatrix}$$

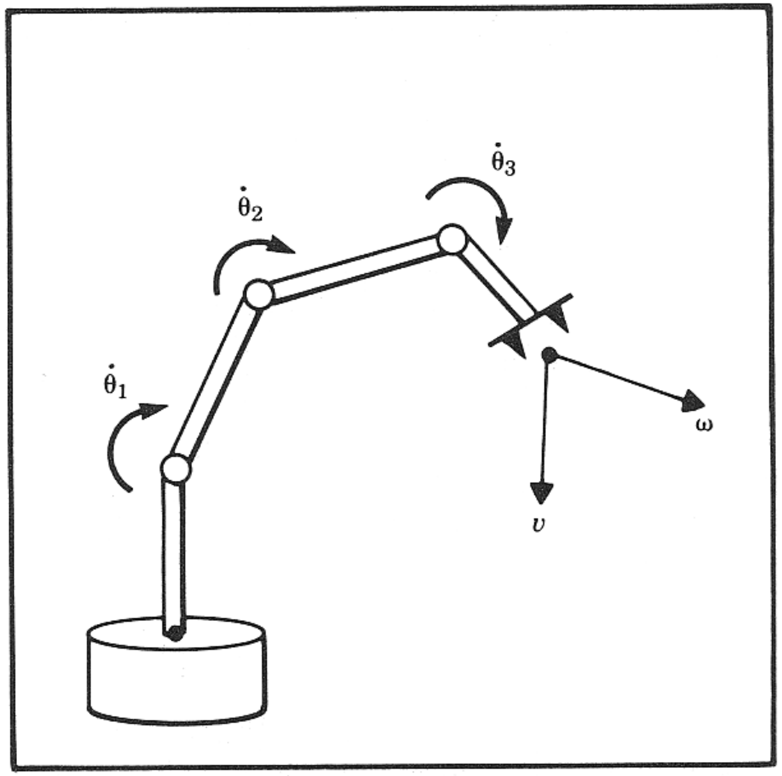


- The first line maps the contribution of the angular velocity of each joint to the linear velocity of the end effector along the x-axis

Jacobian Matrix - Introduction

- The meaning of **each column** (e.g. the first column) of the Jacobian matrix:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} = \begin{bmatrix} J_{11} & \\ J_{21} & J_v(\underline{\theta}) \\ J_{31} & \\ J_{41} & \\ J_{51} & J_\omega(\underline{\theta}) \\ J_{61} & \end{bmatrix} \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \\ \cdots \\ \dot{\theta}_N \end{bmatrix}$$



- The first column maps the contribution of the angular velocity of the first joint to the linear and angular velocities of the end effector along all the axis (x,y,z)

Jacobian – Derivation from First Principles (Virtual Work)

Forces & Torque

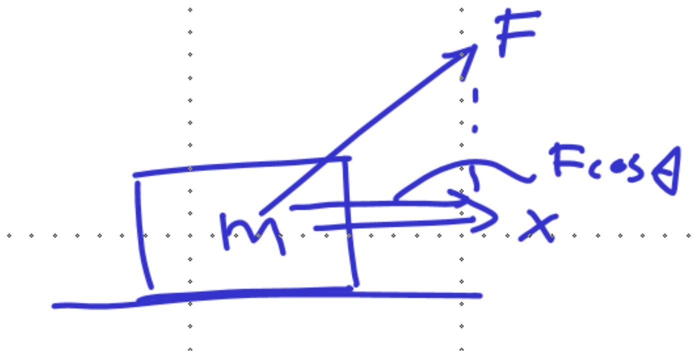
Jacobian Matrix – Derivation Using Virtual Work Principles

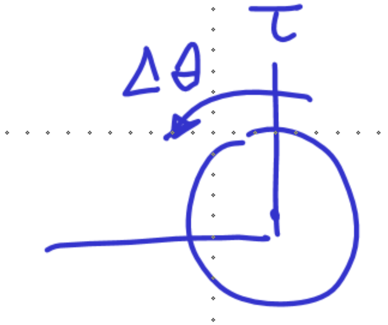
- The work applied on a mass moving in a linear fashion is the dot product of the force applied and its incremental displacement

$$W = F \cdot \Delta x = F cos\theta x$$

- In a similar fashion the work applied on a revolving mass is the do product of the torque and the incremental angular displacement

$$W = \tau \cdot \Delta\theta$$





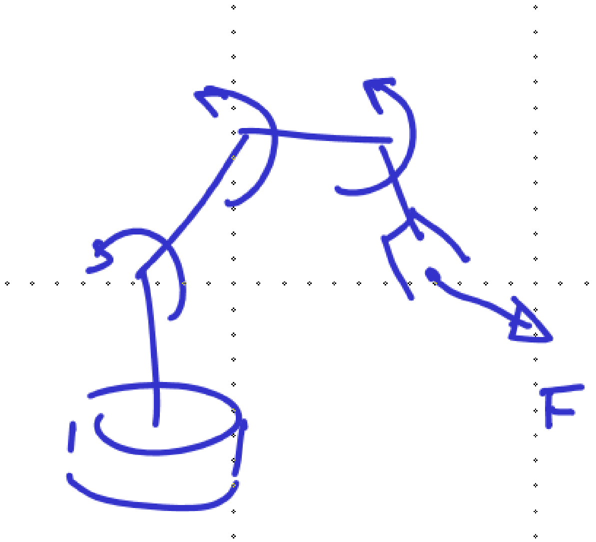
Jacobian Matrix – Derivation Using Virtual Work Principles

- Extending the two previous pervious principles to a multi joint multi link mechanism resulted in an equation which describes from one end the virtual work applied by the joint torque on the manipulator which should be equal to the work applied on its end effector by all the external loads

$$F \cdot \delta x = \tau \cdot \delta \theta$$

- Rewriting this compact equation explicitly resulted in multiple equations defined as follows

$$F_x x = \tau_1 \theta_1$$
$$\dots$$
$$M_x \theta_x = \tau_4 \theta_4$$



Jacobian Matrix – Derivation Using Virtual Work Principles

- Transposing the following compact equation

$$F \cdot \delta x = \tau \cdot \delta\theta$$
$$(F \cdot \delta x = \tau \cdot \delta\theta)^T$$

- Resulted in

$$F^T \delta x = \tau^T \delta\theta$$

- Utilizing the relationship between task space displacement and joint space displacement

$$\delta x = J\delta\theta$$

- And plugging it into the transpose equation, resulted in

$$F^T J\delta\theta = \tau^T \delta\theta$$

Jacobian Matrix – Derivation Using Virtual Work Principles

– Canceling delta theta $\delta\theta$ and transposing the following compact equation

$$F^T J \delta\theta = \tau^T \delta\theta$$

$$[\tau^T = F^T J]^T$$

– Base on the notation where

$$[AB]^T = B^T A^T$$
$$[F^T J]^T = J^T F$$

– Resulting in the equation defining the mapping between external loads and the joint torque

$$\tau = J^T F$$

Jacobian Matrix - Introduction

- Deriving the Jacobian matrix using virtual work principle

$$W = F \cdot \Delta x$$

$$= F cos\theta x$$

$$W = \tau \cdot \Delta\theta$$

$$(F \cdot \delta x = \tau \cdot \delta\theta)^T$$

$$F_x x = \tau_1 \theta_1$$

$$\vdots$$

$$M_x \theta_x = \tau_4 \theta_4$$

$$F^T \delta x = \tau^T \delta\theta$$

$$\delta x = J \delta\theta$$

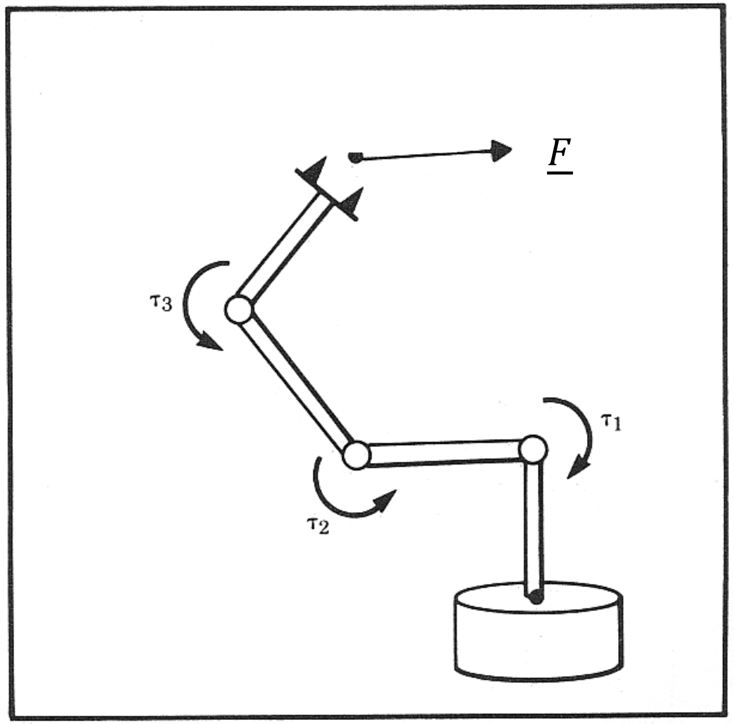
$$F^T J \delta\theta = \tau^T \delta\theta$$

Jacobian Matrix - Introduction

- In addition to the velocity relationship, we are also interested in developing a relationship between the robot joint torques ($\underline{\tau}$) and the forces and moments ($\underline{F}$) at the robot end effector (**Static Conditions**). This relationship is given by:

$$\underline{\tau} = J(\underline{\theta})^T \underline{F}$$

$$\begin{Bmatrix} \tau_1 \\ \tau_2 \\ \tau_3 \\ \tau_4 \\ \tau_5 \\ \tau_6 \end{Bmatrix} \qquad \begin{Bmatrix} F_x \\ F_y \\ F_z \\ M_x \\ M_y \\ M_z \end{Bmatrix}$$



Jacobian Matrix - Introduction

- This expression can be expanded to:

$$\begin{bmatrix} \tau_1 \\ \tau_2 \\ \dots \\ \tau_N \end{bmatrix} = \begin{bmatrix} J_f(\underline{\theta}) & J_\tau(\underline{\theta}) \end{bmatrix}^T \begin{bmatrix} F_x \\ F_y \\ F_z \\ M_x \\ M_y \\ M_z \end{bmatrix}$$

Nx1 **6xN** **6x1**

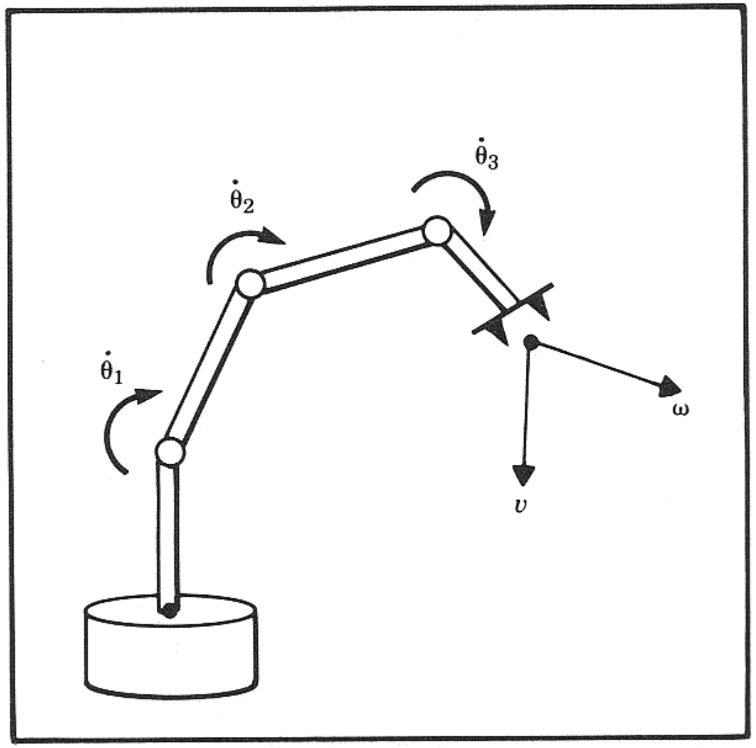
- Where:
  - $\underline{\tau}$ is a 6x1 vector of the robot joint torques
  - $J(\underline{\theta})^T$ is a 6xN Transposed Jacobian matrix
  - $\underline{F}$ is a Nx1 vector of the forces and moments at the robot end effector
  - N is the number of joints

Jacobian Matrix - Introduction

- The meaning of **<u>each line</u>** (e.g. the first line) of the Jacobian matrix:

$$\begin{bmatrix} \tau_1 \\ \tau_2 \\ \cdots \\ \tau_N \end{bmatrix} = \begin{bmatrix} J_{11} & J_{21} & J_{31} & J_{41} & J_{51} & J_{61} \\ & J_f(\underline{\theta}) & & & J_\tau(\underline{\theta}) & \\ & & & & & \end{bmatrix}^T \begin{bmatrix} F_x \\ F_y \\ F_z \\ M_x \\ M_y \\ M_z \end{bmatrix}$$

Nx1 **6xN** **6x1**



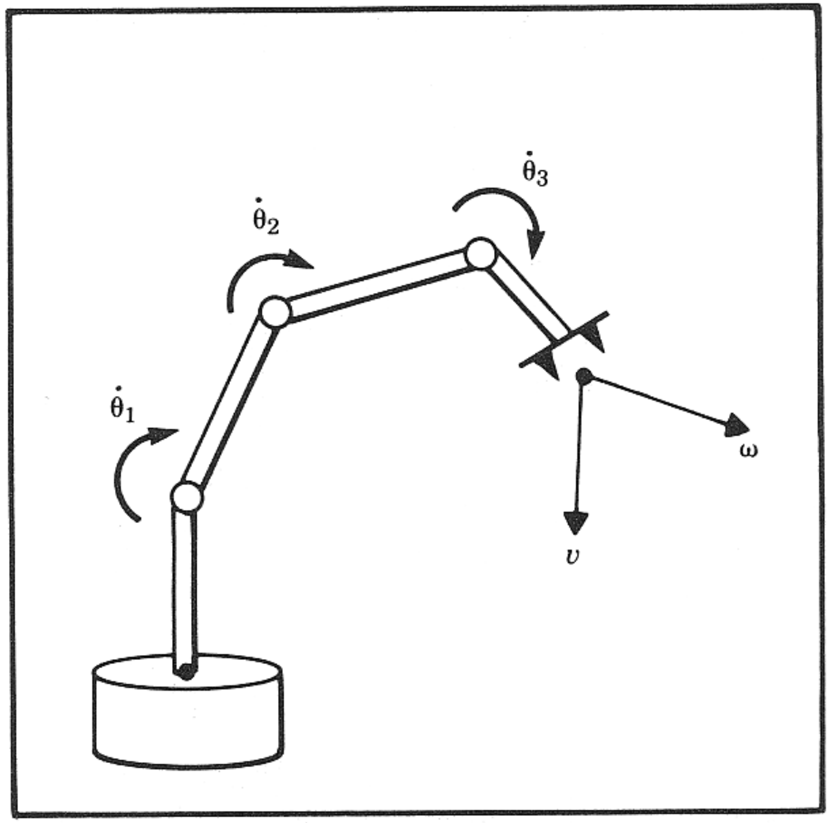
- **Action**: The first line represent how the torque applied at the first joint contributes to the forces and torques applied by the end effector
- **Reaction**: The first line maps the contribution of the partial external loads applied on the end effector to the join torque that needs to be applied to maintain static equilibriums
-

Jacobian Matrix - Introduction

- The meaning of **<u>each column</u>** (e.g. the first column) of the Jacobian matrix:

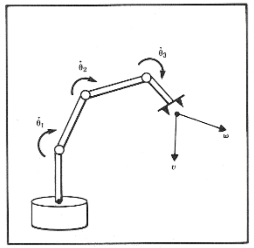
$$\begin{bmatrix}\tau_1\\ \tau_2\\ \cdots\\ \tau_N\end{bmatrix} = \begin{bmatrix} J_{11} & & \\ J_{12} & & \\ J_{13} & J_f(\underline{\theta}) & J_\tau(\underline{\theta}) \\ J_{14} & & \\ J_{15} & & \\ J_{16} & & \end{bmatrix}^T \begin{bmatrix}F_x\\ F_y\\ F_z\\ M_x\\ M_y\\ M_z\end{bmatrix}$$

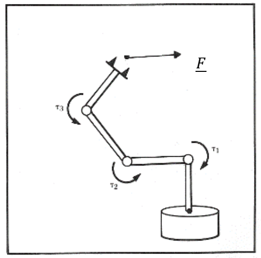
Nx1 **6xN** **6x1**



- **Action**: The first column represent what partial torque applied by each joint is required to create an equilibrium of the force alon the X- Axis
- **Reaction**: The first column maps the contribution of the partial external loads of the force along the X-axis applied on the end effector to the join torques that are needed to be applied to maintain static equilibriums

Jacobian Matrix - Derivation Methods





Explicit Method

- Differentiation the Forward Kinematics Eqs. (Method 1)

Iterative Methods Recursive Equations

- Velocity Propagation – Base to EE (Method 2)
- Force/Torque Propagation – EE to Base (Method 3)

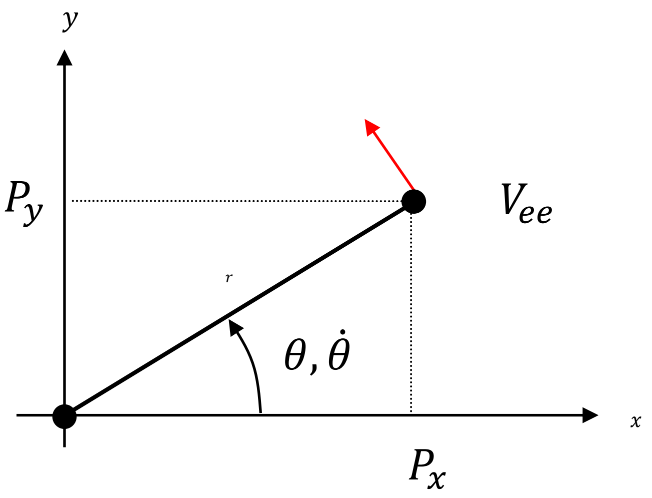
↓

Jacobian Matrix

Jacobian – R Robot (1 DOF) - Example

Jacobian Matrix by Differentiation - 1R - 1/4

- Consider a simple planar 1R robot



- The end effector position is given by

$$^0P_x = x = r\cos\theta$$
$$^0P_y = y = r\sin\theta$$

Jacobian Matrix by Differentiation - 1R - 2/4

- The velocity of the end effector is defined by

$$^{0}V_x = {}^{0}\dot{P}_x = \dot{x} = -\dot{\theta} r \sin\theta = (-\omega r \sin\theta)\dot{\theta}$$
$$^{0}V_y = {}^{0}\dot{P}_y = \dot{y} = \dot{\theta} r \cos\theta = (\omega r \cos\theta)\dot{\theta}$$

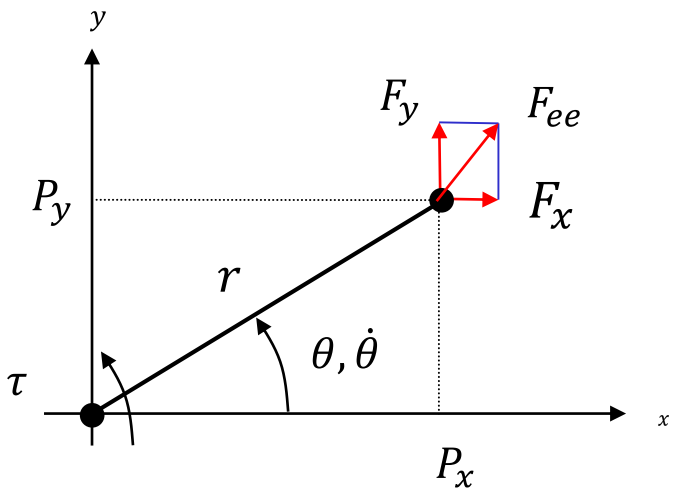
- Expressed in matrix form we have

$$\underline{\dot{x}} = J(\underline{\theta})\underline{\dot{\theta}}$$

$$\begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} = \begin{bmatrix} -r\sin\theta \\ r\cos\theta \end{bmatrix} \begin{bmatrix} \dot{\theta} \end{bmatrix}$$

2x1 **2x1** **1x1**

Jacobian Matrix by Differentiation - 1R - 3/4



- The moment about the joint generated by the force acting on the end effector is given by

$$\tau = -rF_x \sin\theta + rF_y \cos\theta$$

Jacobian Matrix by Differentiation - 1R - 4/4

- Expressed in matrix form we have

$$\underline{\tau} = J(\underline{\theta})^T \underline{F}$$

$$[\tau] = \begin{bmatrix} -r\sin\theta & r\cos\theta \end{bmatrix} \begin{bmatrix} F_x \\ F_y \end{bmatrix}$$

1x1 1x2 2x1

$$\underline{\dot{x}} = J(\underline{\theta})\underline{\dot{\theta}}$$

$$\begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} = \begin{bmatrix} -r\sin\theta \\ r\cos\theta \end{bmatrix} [\dot{\theta}]$$

Jacobian – 2R Robot (2 DOF) - Example
Jacobian – Manipulability Ellipsoid

Jacobian Matrix by Differentiation - 2R

- **Given:** Consider the following 2 DOF Planar manipulator

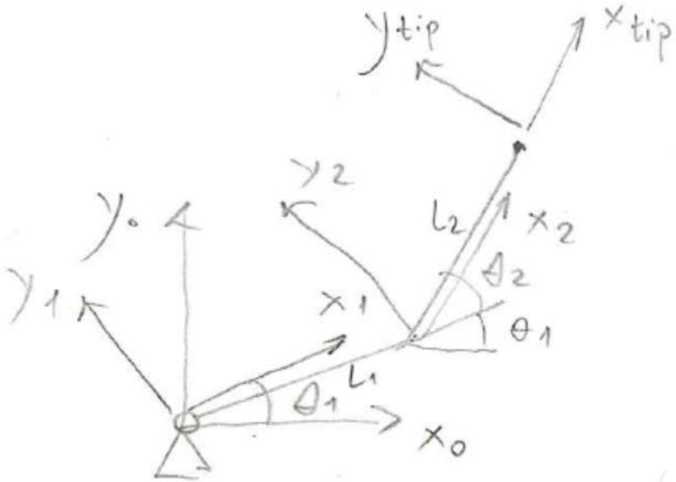
- **Problem:** Compute the Jacobian matrix that describes the relationship

$$\dot{\underline{x}} = J(\underline{\theta})\dot{\underline{\theta}} \qquad \underline{\tau} = J(\underline{\theta})^T \underline{F}$$

- **Solution:** Differentiating the forward kinematics equations

$$\underline{x} = \begin{bmatrix} x \\ y \end{bmatrix}$$

- **Result:** The end effector position and orientation is defined in the base frame by



Jacobian Matrix by Differentiation - 2R

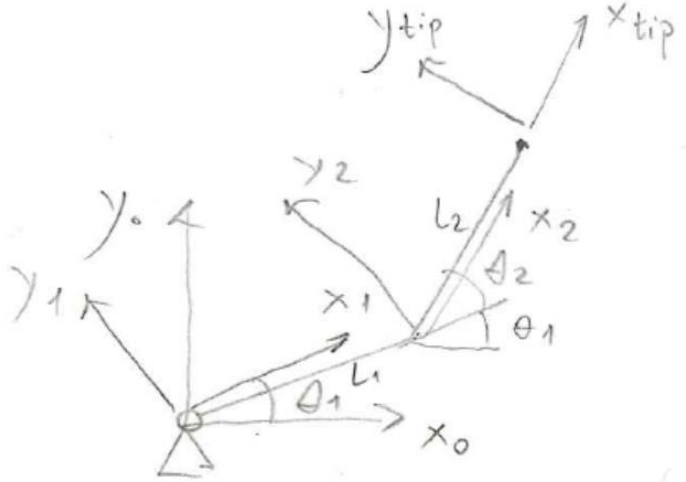
$$x_{tip} = L_1 c_1 + L_2 c_{12}$$

$$y_{tip} = L_1 s_1 + L_2 s_{12}$$

$$v_{xtip} = \frac{dx_{tip}}{dt} = -L_1 \dot{\theta}_1 s_1 - L_2 \left(\dot{\theta}_1 + \dot{\theta}_2\right) s_{12}$$

$$v_{ytip} = \frac{dy_{tip}}{dt} = L_1 \dot{\theta}_1 c_1 + L_2 \left(\dot{\theta}_1 + \dot{\theta}_2\right) c_{12}$$

$$\begin{Bmatrix} v_x \\ v_y \end{Bmatrix} = \begin{bmatrix} -L_1 s_1 - L_2 s_{12} & -L_2 s_{12} \\ L_1 c_1 + L_2 c_{12} & L_2 c_{12} \end{bmatrix} \begin{Bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{Bmatrix}$$



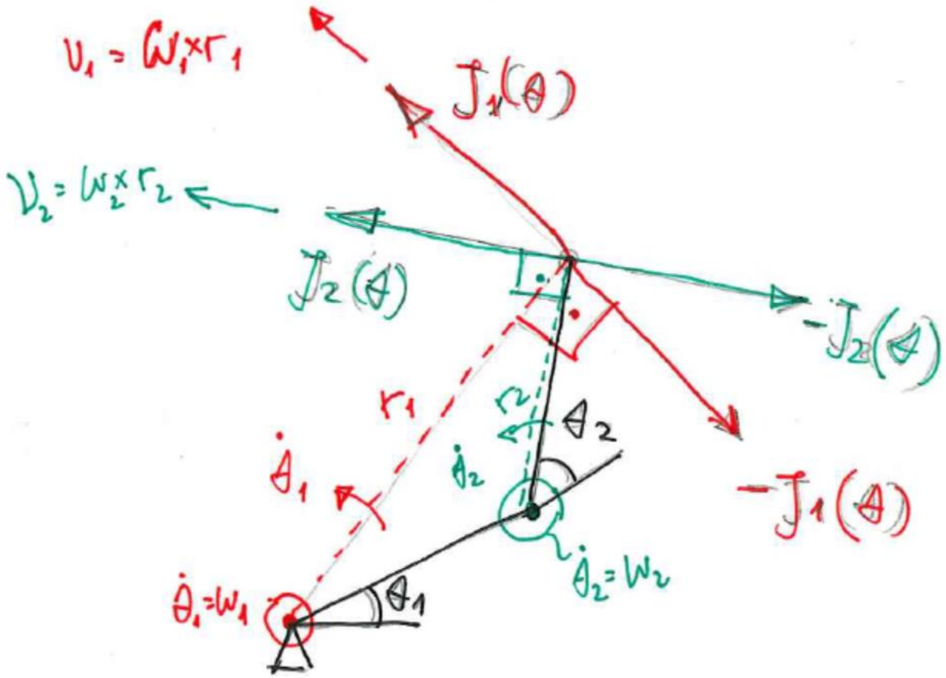
Jacobian Matrix by Differentiation - 2R

$$\begin{Bmatrix} v_x \\ v_y \end{Bmatrix} = \begin{bmatrix} J_{11} & J_{21} \\ J_{12} & J_{22} \end{bmatrix} \begin{Bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{Bmatrix}$$

- Column 1 of $J(\theta) \rightarrow J_1(\theta)$ when $\dot{\theta}_1 = 1, \dot{\theta}_2 = 0$
- Column 2 of $J(\theta) \rightarrow J_2(\theta)$ when $\dot{\theta}_1 = 0, \dot{\theta}_2 = 1$

- As long as $J_1(\theta)$ and $J_2(\theta)$ are not collinear (parallel), it is possible to generate an end effector velocity v_{tip} in any arbitrary direction in the x_0, y_0 plane by choosing appropriate joint velocities $\dot{\theta}_1$ and $\dot{\theta}_2$.

- Since $J_1(\theta)$ and $J_2(\theta)$ depend on the joint values θ_1 and θ_2, there are some configurations where $J_1(\theta), J_2(\theta)$ become collinear (parallel) (e.g. when $\theta_2 = 0$ or $\theta_2 = 180$)

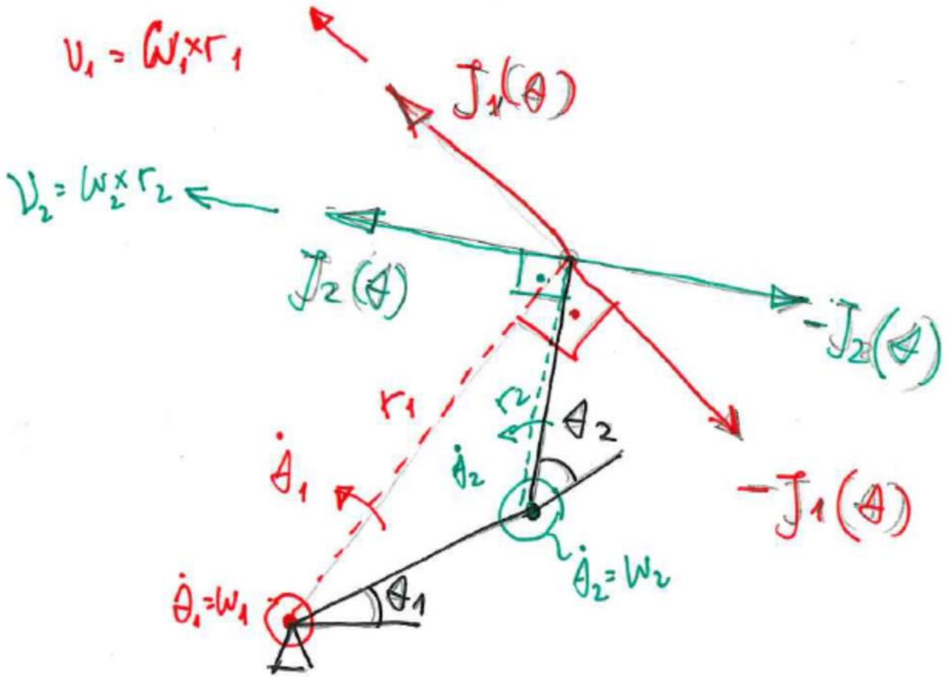


Jacobian Matrix by Differentiation - 2R

- If $\begin{Bmatrix} \theta_2 = 0 \\ \theta_2 = 180 \end{Bmatrix}$

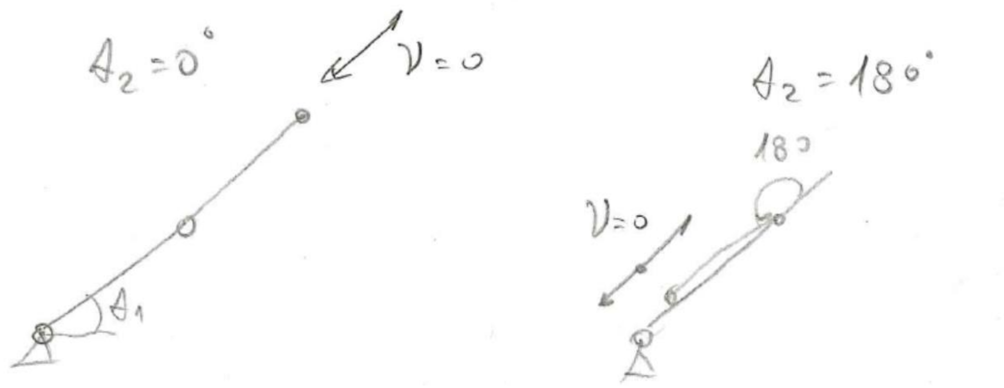
  regardless of the value of θ_1, $J_1(\theta)$ and $J_2(\theta)$ will be collinear and the Jacobian $J(\theta)$ become a singular matrix

- Such configurations are called singularities, and they are characterized by a situation where the robot's end effector is unable to generate velocities in certain directions



Jacobian Matrix by Differentiation - 2R

For any θ_1 $\begin{Bmatrix} \theta_2 = 0 \\ \theta_2 = 180 \end{Bmatrix}$ $\begin{bmatrix} J_1 \parallel J_2 \\ J_1 \parallel J_2 \end{bmatrix} \rightarrow singularities$



Jacobian Matrix by Differentiation - 2R

- Substitute $L_1 = 1; \quad L_2 = 1$
- Consider the robot at two different non-singular postures

$$\theta = \begin{bmatrix} 0 \\ \pi/4 \end{bmatrix} \qquad\qquad J\left(\begin{bmatrix} 0 \\ \pi/4 \end{bmatrix}\right) = \begin{bmatrix} -0.71 & -0.71 \\ 1.71 & 0.71 \end{bmatrix}$$

- The Jacobian can be used to map bounds on rotational speed of the joints $(\dot{\theta})$ to bounds on the end effector velocity (v_{tip})

$$\dot{\theta} = \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix} \longrightarrow \dot{X} = \begin{bmatrix} v_x \\ v_y \end{bmatrix}$$

$$\dot{X} = J(\theta)\dot{\theta}$$

Tip Velocity in Joint Space

$$\begin{bmatrix} v_x \\ v_y \end{bmatrix} = \begin{bmatrix} -0.71 & -0.71 \\ 1.71 & 0.71 \end{bmatrix} \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix}$$

Tip velocity in Task Space

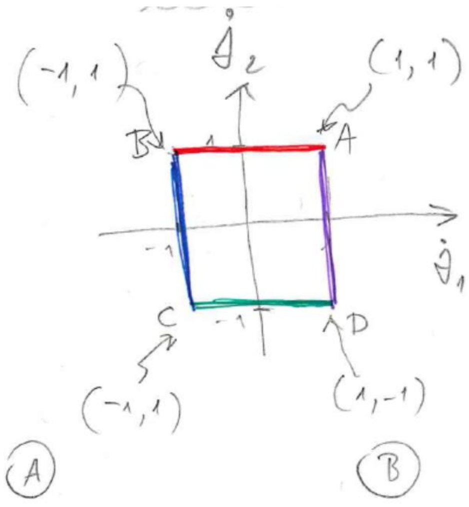
Jacobian Matrix by Differentiation - 2R

$$\dot{\theta} = \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix} \quad \dot{X} = J(\theta)\dot{\theta} \quad \dot{X} = \begin{bmatrix} v_x \\ v_y \end{bmatrix}$$

Tip Velocity in Joint Space

$$\begin{bmatrix} v_x \\ v_y \end{bmatrix} = \begin{bmatrix} -0.71 & -0.71 \\ 1.71 & 0.71 \end{bmatrix} \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix}$$

Tip velocity in Task Space

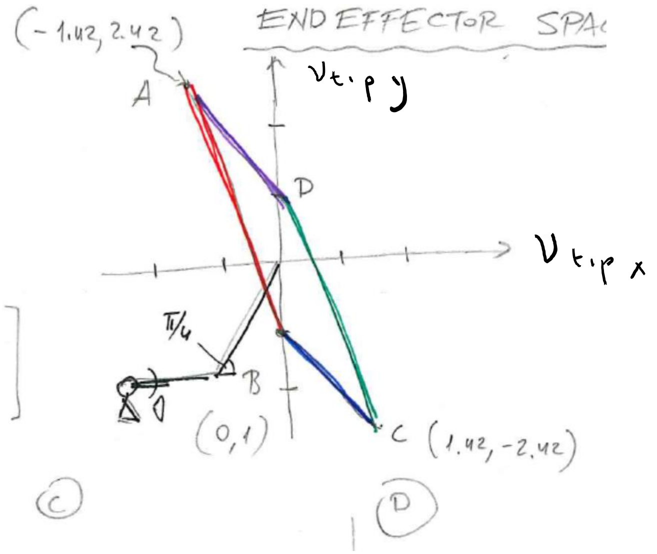


A $v_{tip} = \begin{bmatrix} -0.71 & -0.71 \\ 1.71 & 0.71 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix} = \begin{bmatrix} -1.42 \\ 2.42 \end{bmatrix}$

B $v_{tip} = \begin{bmatrix} -0.71 & -0.71 \\ 1.71 & 0.71 \end{bmatrix} \begin{bmatrix} -1 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ -1 \end{bmatrix}$

C $v_{tip} = \begin{bmatrix} -0.71 & -0.71 \\ 1.71 & 0.71 \end{bmatrix} \begin{bmatrix} -1 \\ -1 \end{bmatrix} = \begin{bmatrix} 1.42 \\ -2.42 \end{bmatrix}$

D $v_{tip} = \begin{bmatrix} -0.71 & -0.71 \\ 1.71 & 0.71 \end{bmatrix} \begin{bmatrix} 1 \\ -1 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$

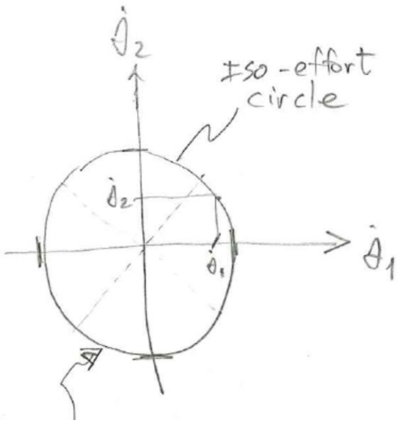


Jacobian Matrix by Differentiation - 2R

- Rather than mapping a polygon of joint velocities through the Jacobian, we could instead map a unit circle of joint velocities into the end effector velocities in the x_0, y_0 plane

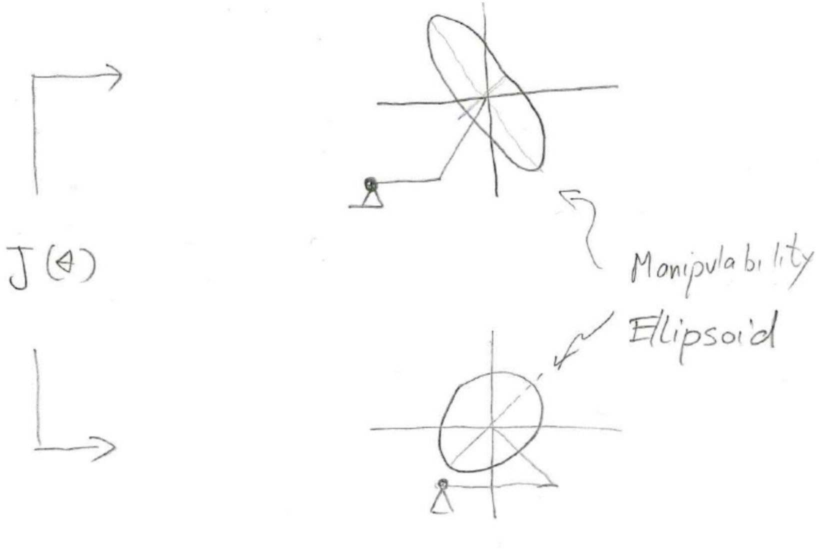
- The circle represents an iso-effort contour in the joint velocity space, where total actuator effort is considered to be the sum of squares of the joint velocities

Tip Velocity in Joint Space

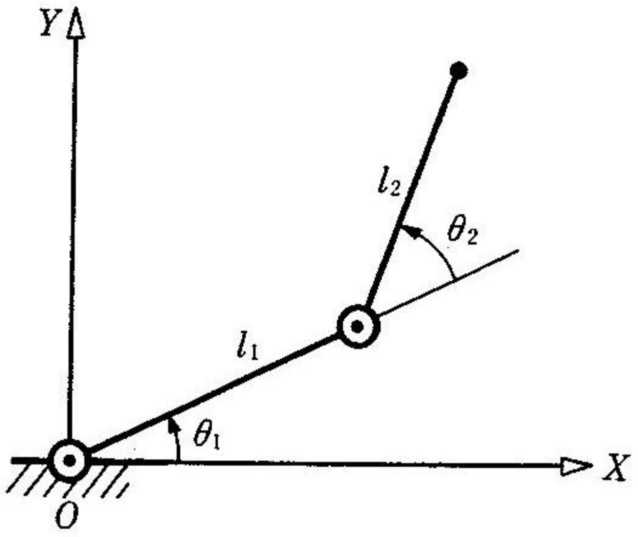


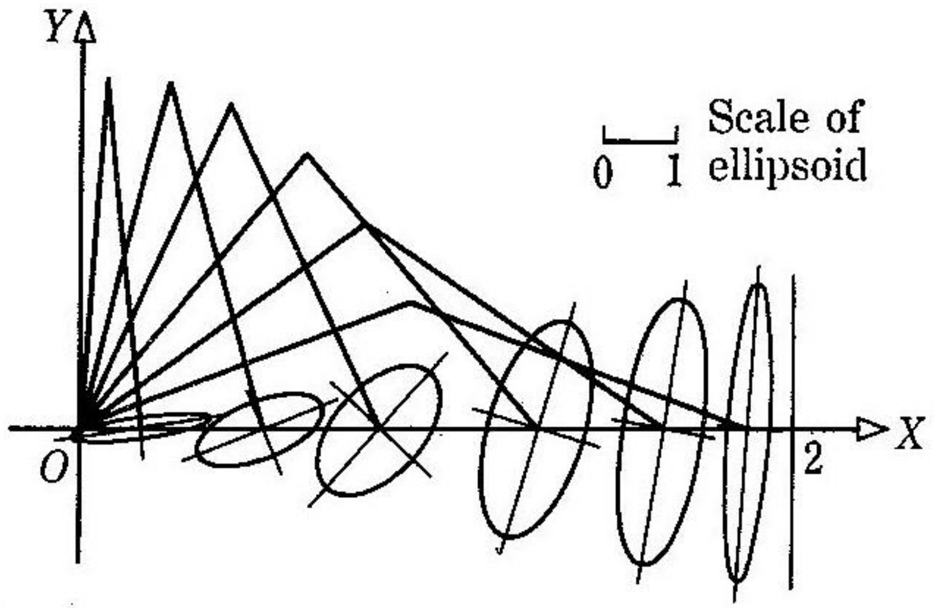
$$1 = \sqrt{\dot{\theta}_1^2 + \dot{\theta}_2^2}$$

Tip velocity in Task Space



Properties of the Jacobian - Velocity Mapping and Singularities



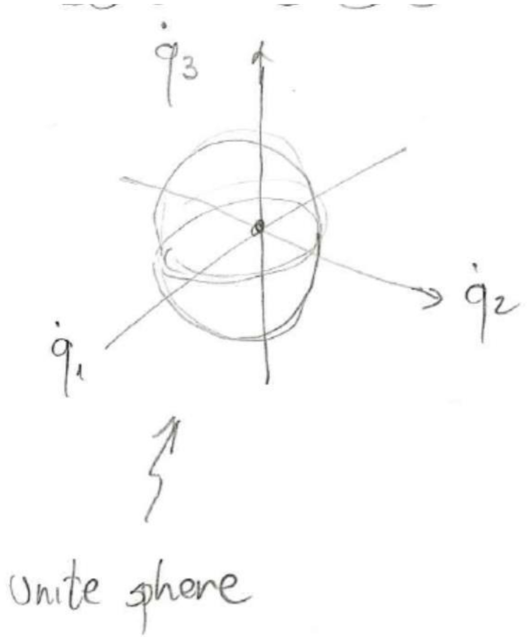


- Note: See Mathematica Simulations
  - Two Link: https://demonstrations.wolfram.com/ForwardAndInverseKinematicsForTwoLinkArm/
  - Three links : https://demonstrations.wolfram.com/ManipulabilityEllipsoidOfARobotArm/

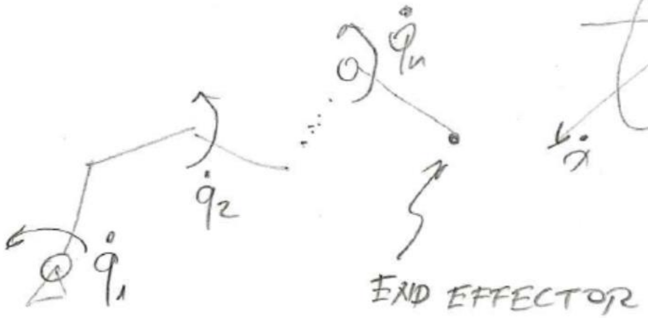
Manipulability Ellipsoid – Definition

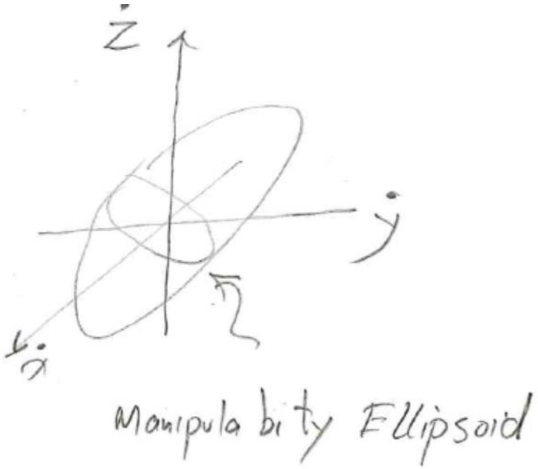
Tip Velocity in Joint Space

Tip velocity in Task Space



$$\dot{x} = J(\theta)\dot{q}$$





Manipulability Ellipsoid & Manipulability Measures – Design

- Robotic Arm Design – Mechanism Size
- Robotic Arm - Base Position – Position of the mechanism with respect to the workspace to maximize the manipulability

Jacobian – RR Robot (3 DOF) - Example

Jacobian Matrix by Differanciation - 3R - 1/4

- **Given:** Consider the following 3 DOF Planar manipulator

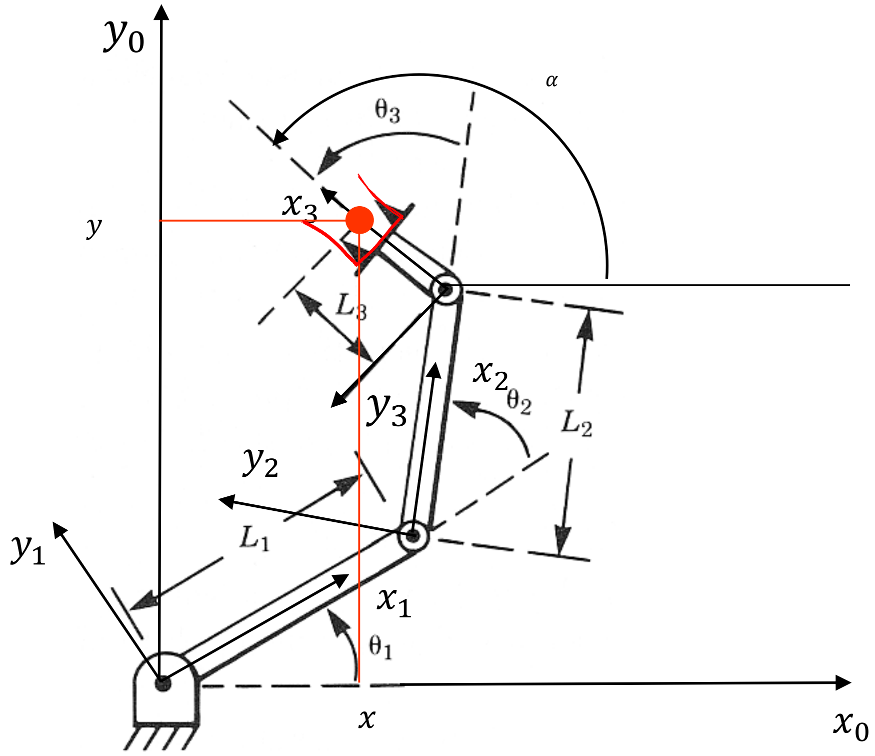
- **Problem:** Compute the Jacobian matrix that describes the relationship

$$\underline{\dot{x}} = J(\underline{\theta})\underline{\dot{\theta}} \qquad \underline{\tau} = J(\underline{\theta})^T \underline{F}$$

- **Solution:** Differentiating the forward kinematics equations

$$\underline{x} = \begin{bmatrix} x \\ y \\ \alpha \end{bmatrix}$$

- **Result:** The end effector position and orientation is defined in the base frame by



Jacobian Matrix by Differanciation - 3R - 2/4

- **Problem:** Compute the Jacobian matrix that describes the relationship

$$\underline{\dot{x}} = J(\underline{\theta})\underline{\dot{\theta}} \qquad \underline{\tau} = J(\underline{\theta})^T \underline{F}$$

- **Solution:** The end effector position and orientation is defined in the base frame by

$$\underline{x} = \begin{bmatrix} x \\ y \\ \alpha \end{bmatrix}$$

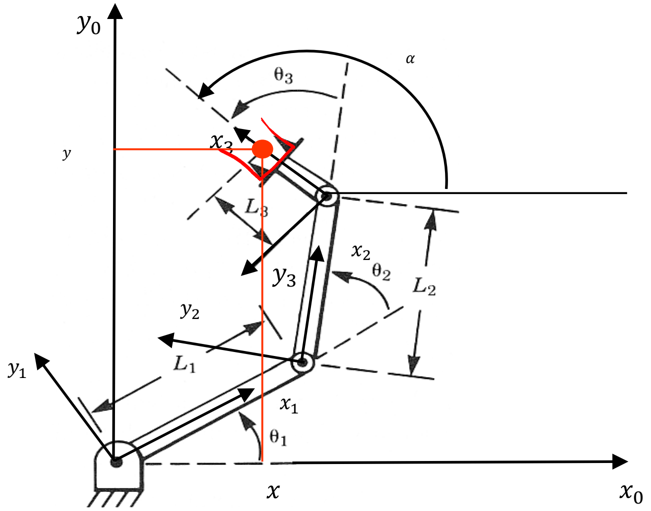
Jacobian Matrix by Differanciation - 3R - 3/4

- The forward kinematics gives us relationship of the end effector to the joint angles:

$$^{0}P_{3\ org,\ x} = x = L_1c_1 + L_2c_{12} + L_3c_{123}$$

$$^{0}P_{3\ org,\ y} = y = L_1s_1 + L_2s_{12} + L_3s_{123}$$

$$^{0}P_{3\ org,\ \alpha} = \alpha = \theta_1 + \theta_2 + \theta_3$$



- Differentiating the three expressions gives

$$\dot{x} = -L_1s_1\dot{\theta}_1 - L_2s_{12}(\dot{\theta}_1 + \dot{\theta}_2) - L_3s_{123}(\dot{\theta}_1 + \dot{\theta}_2 + \dot{\theta}_3)$$
$$= -(L_1s_1 + L_2s_{12} + L_3s_{123})\dot{\theta}_1 - (L_2s_{12} + L_3s_{123})\dot{\theta}_2 - (L_3s_{123})\dot{\theta}_3$$

$$\dot{y} = L_1c_1\dot{\theta}_1 + L_2c_{12}(\dot{\theta}_1 + \dot{\theta}_2) + L_3c_{123}(\dot{\theta}_1 + \dot{\theta}_2 + \dot{\theta}_3)$$
$$= (L_1c_1 + L_2c_{12} + L_3c_{123})\dot{\theta}_1 + (L_2c_{12} + L_3c_{123})\dot{\theta}_2 + (L_3c_{123})\dot{\theta}_3$$

$$\dot{\alpha} = \dot{\theta}_1 + \dot{\theta}_2 + \dot{\theta}_3$$

Jacobian Matrix by Differanciation - 3R - 4/4

- Using a matrix form we get

$$\underline{\dot{x}} = {}^{0}J(\underline{\theta})\underline{\dot{\theta}}$$

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\alpha} \end{bmatrix} = \begin{bmatrix} -L_1 s_1 - L_2 s_{12} - L_3 s_{123} & -L_2 s_{12} - L_3 s_{123} & -L_3 s_{123} \\ L_1 c_1 + L_2 c_{12} + L_3 c_{123} & L_2 c_{12} + L_3 c_{123} & L_3 c_{123} \\ 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \\ \dot{\theta}_3 \end{bmatrix}$$

- The Jacobian provides a linear transformation, giving a velocity map and a force map for a robot manipulator. For the simple example above, the equations are trivial, but can easily become more complicated with robots that have additional degrees a freedom. Before tackling these problems, consider this brief review of linear algebra.