# Robotics Terminology


Dr. Atul Thakur

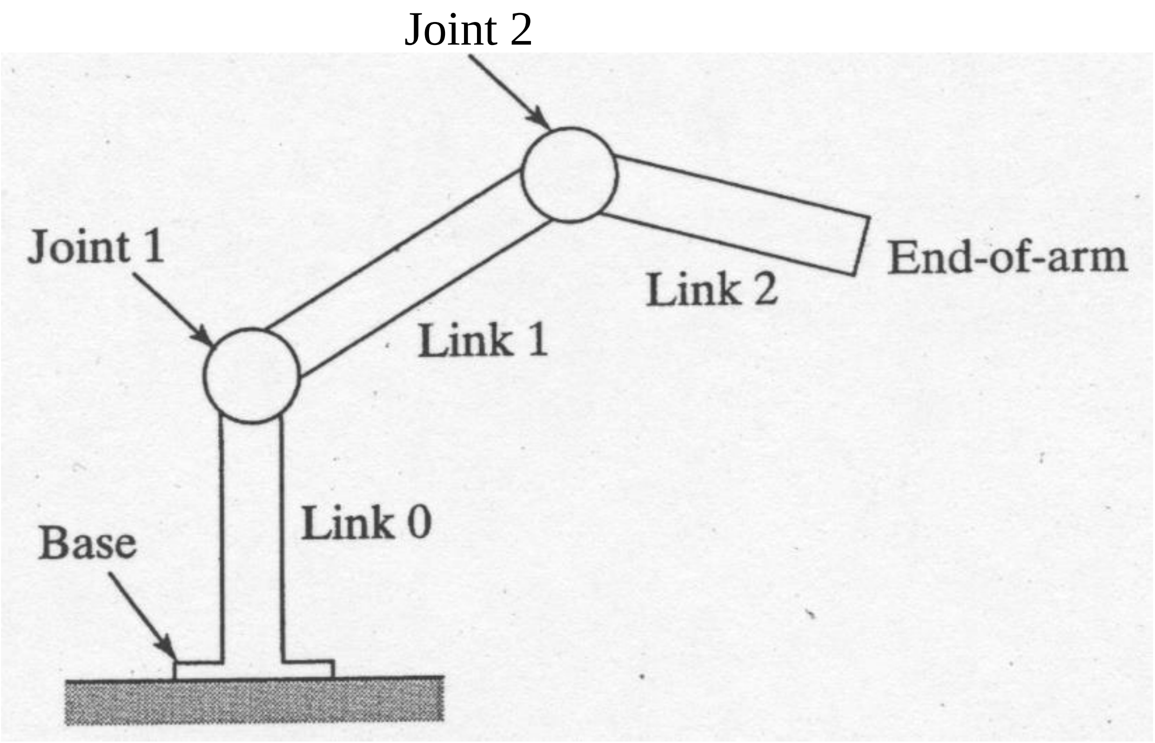
Mechanical Engineering Department

IIT Patna

# Parts of Stationary Robot

- Mechanisms (Mechanical Structure)
- End Effecters
- Tools
- Controllers
- Actuators
- Sensors
- Programming Interface

# Mechanism Anatomy

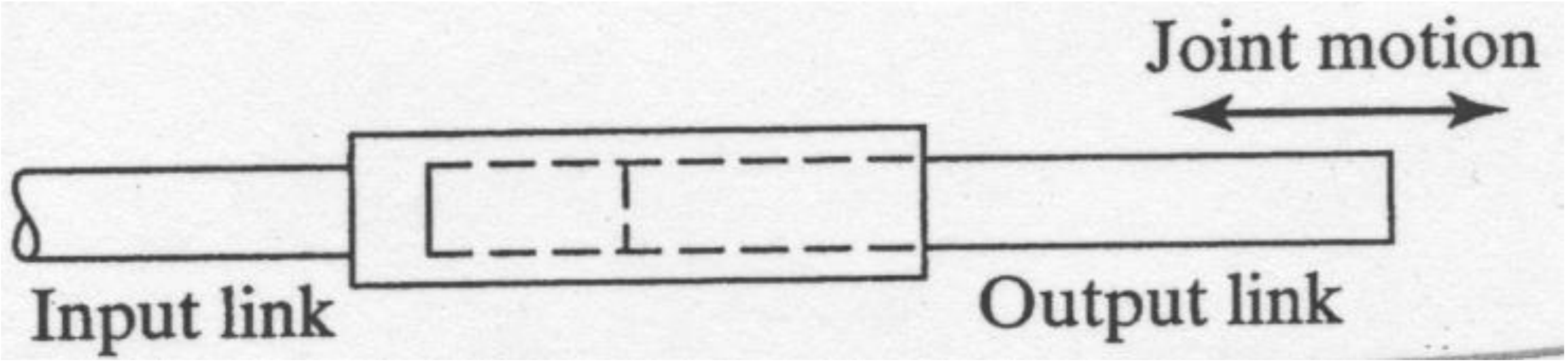


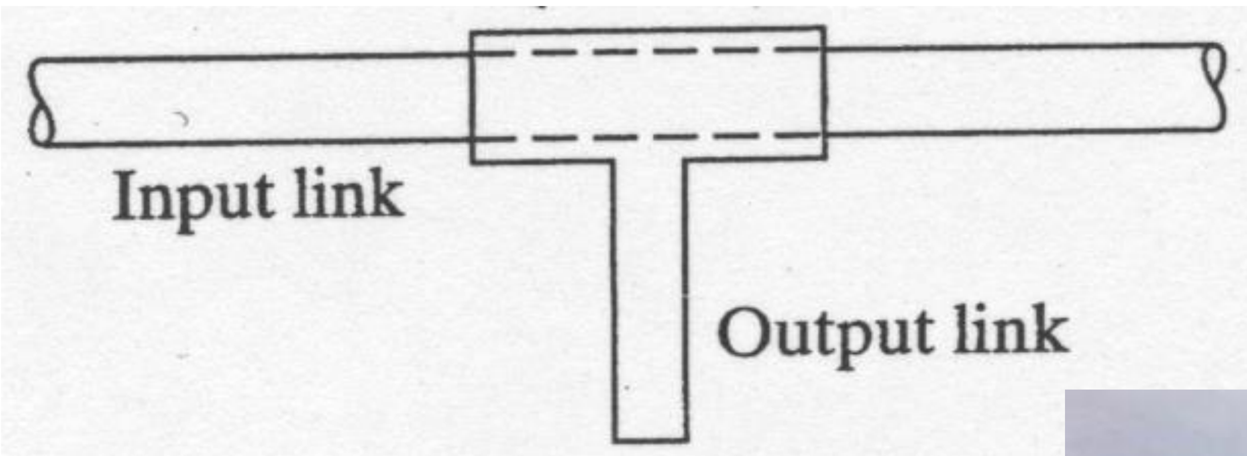

# Links and Joints

- Links are rigid parts
- Joints permit relative motion between links

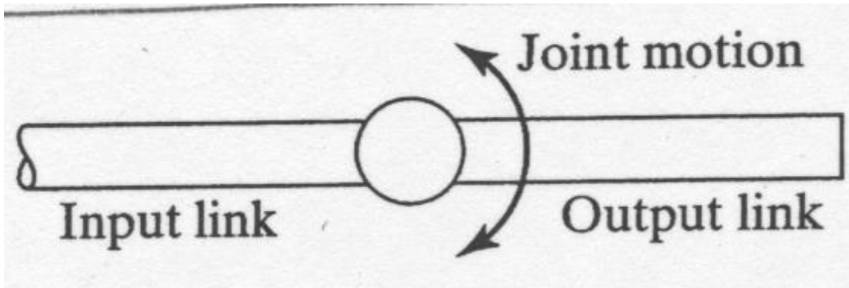
# Types of Joints

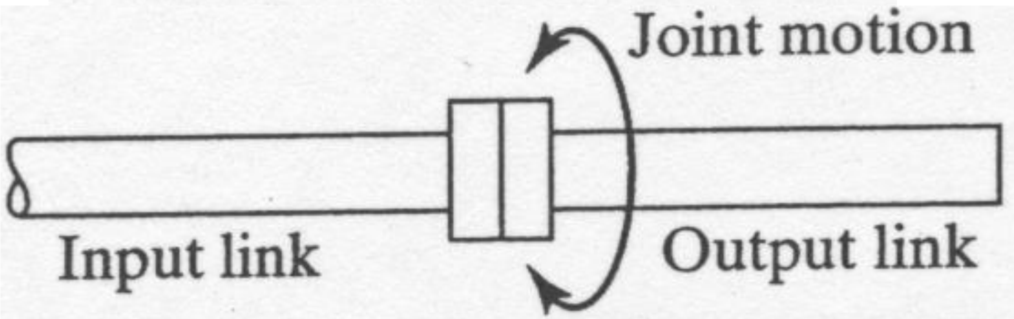
# Prismatic Joint

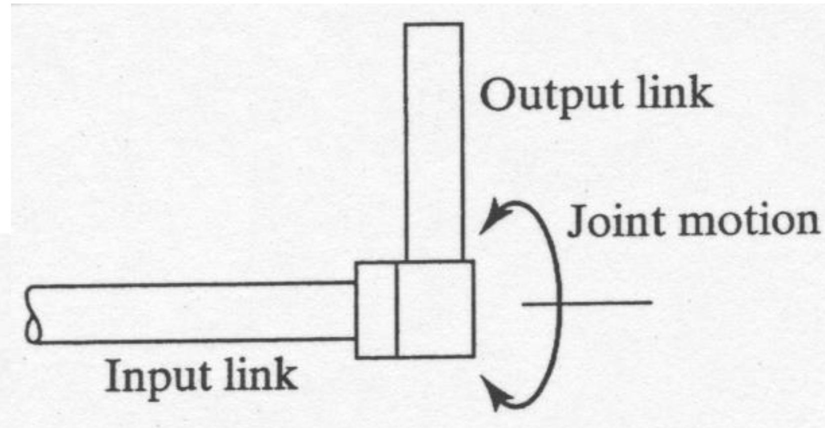



# Revolute Joint




# Degrees of Freedom in Mechanisms

How many degrees of freedom (DOF) are needed to position and orient an object in space?

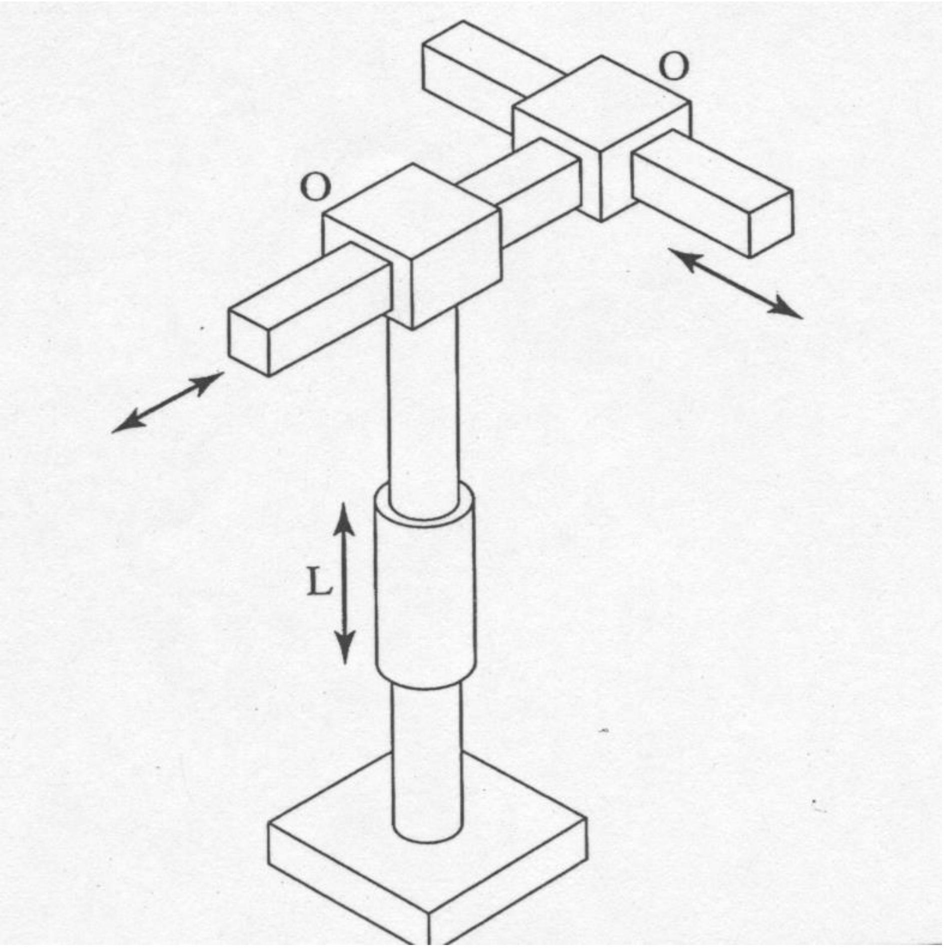
# Desired Degrees of Freedom

- 3 Dimensional Case
  - 3 DOF for positioning (x, y, z)
  - 3 DOF for orientation (pitch, yaw, roll)
- 2 Dimensional Case (Planar case)
  - 2 DOF for positioning (x, y)
  - 1 DOF for orientation (tilt)

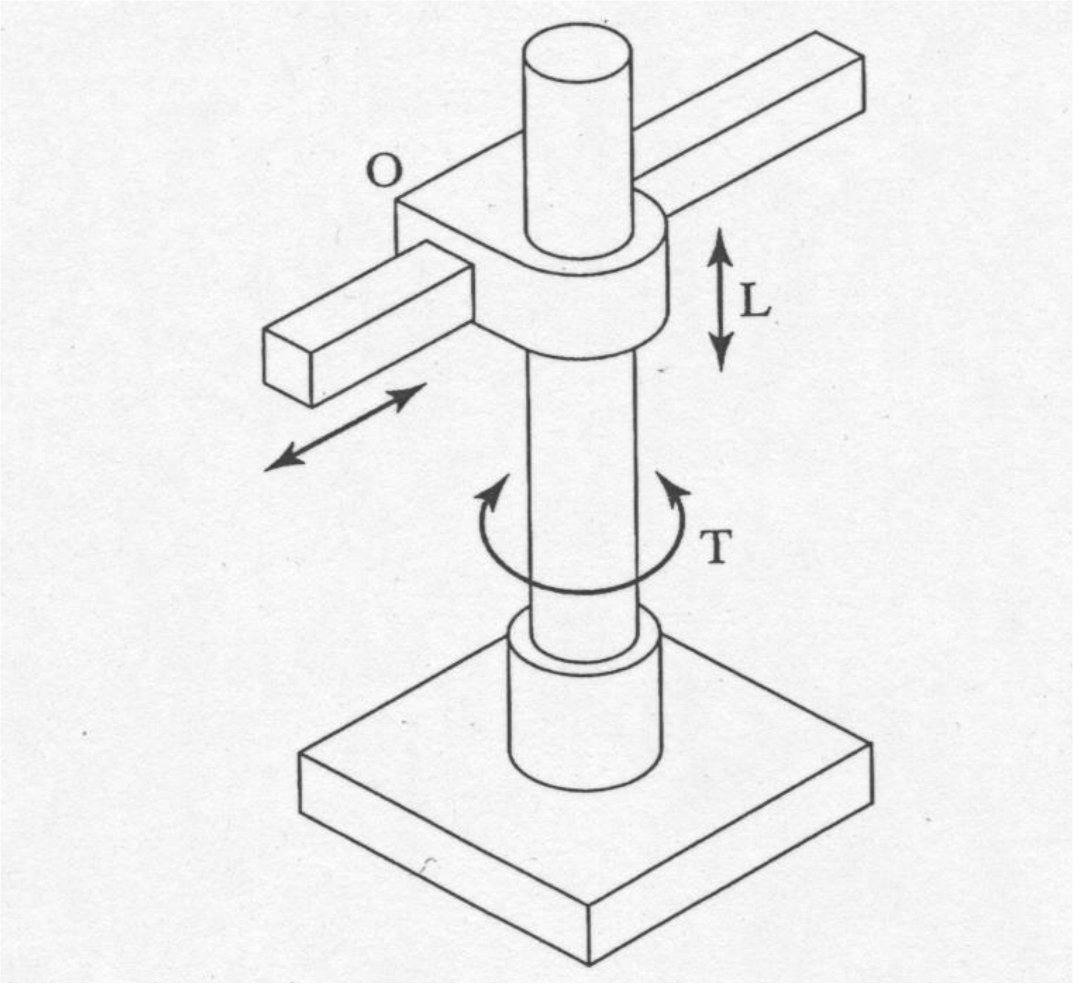
# Mechanism Types

- Two different category
  - Arm
    - Two to three degrees of freedom
  - Wrist
    - One to three degrees of freedom

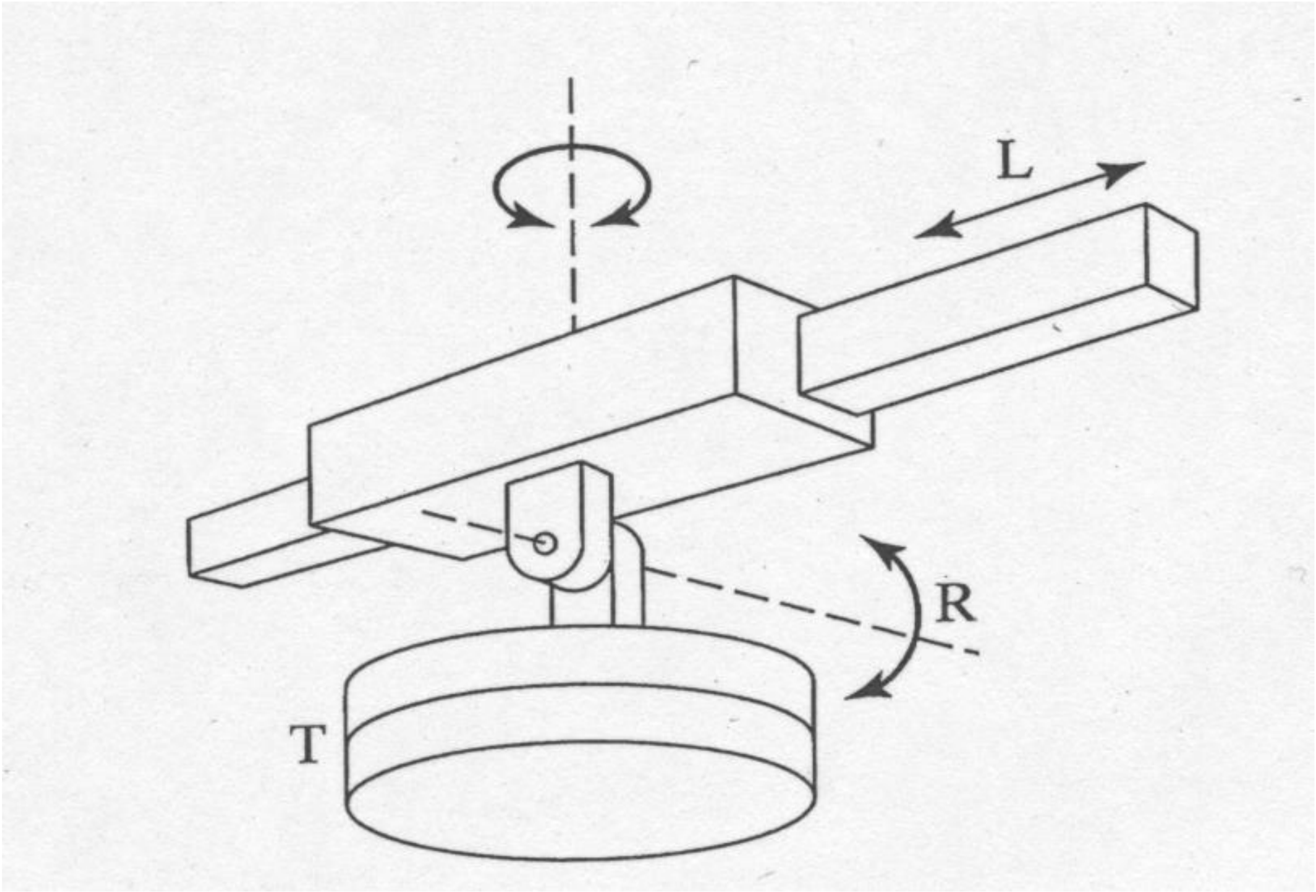
# Arm: Cartesian Configuration



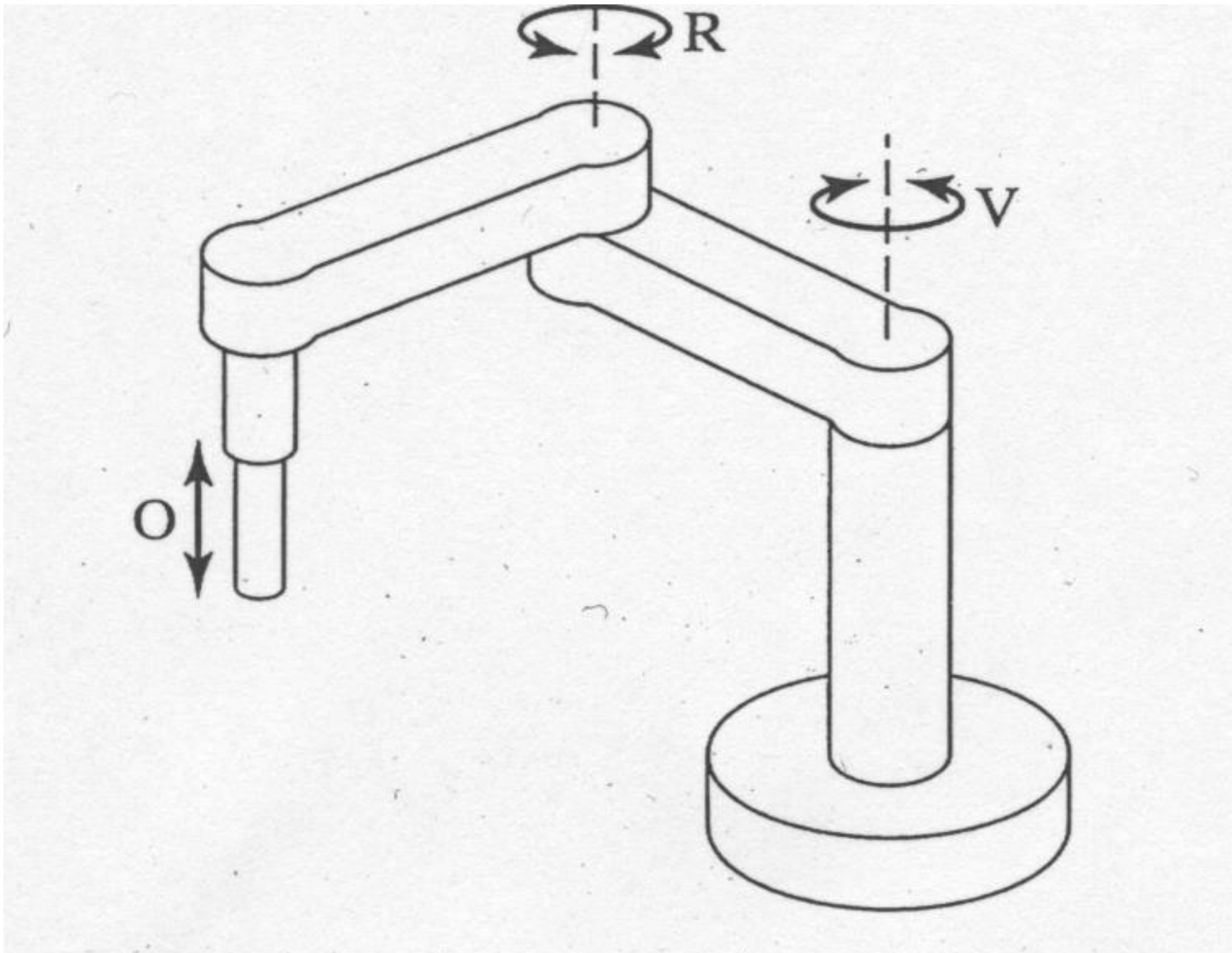
# Arm: Cylindrical Configuration



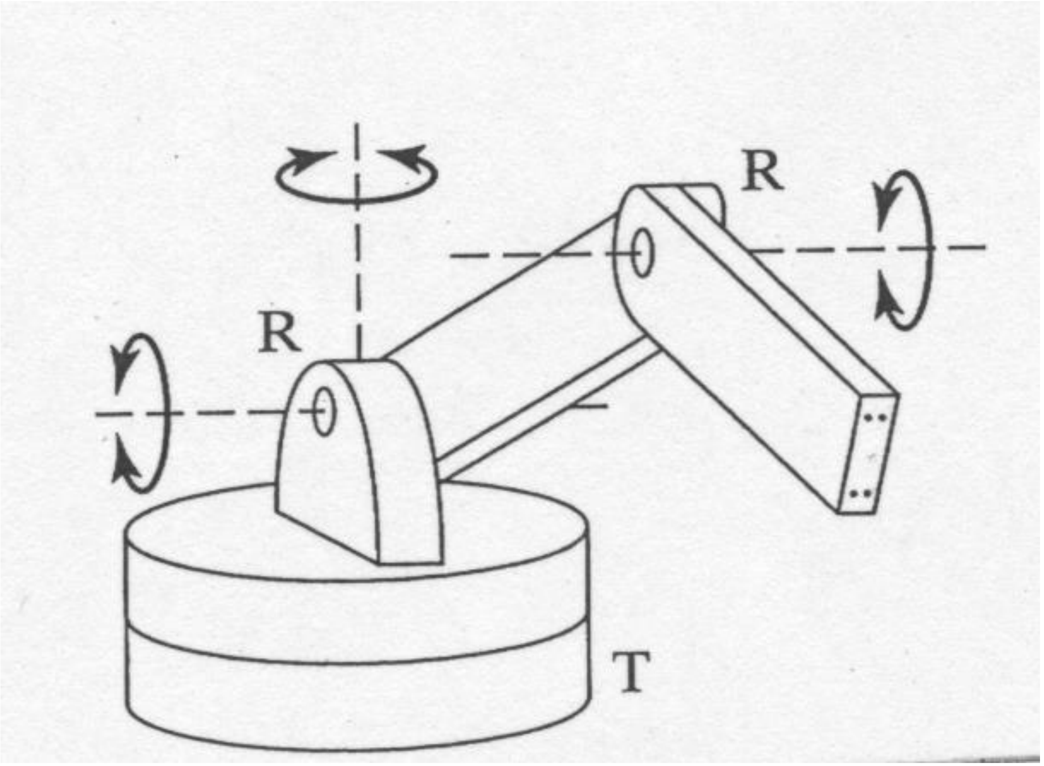
# Arm: Polar Configuration



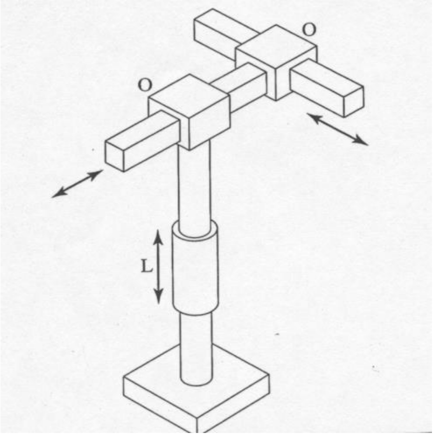
# Arm: SCARA Configuration

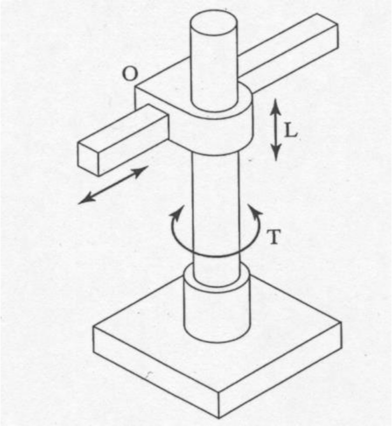


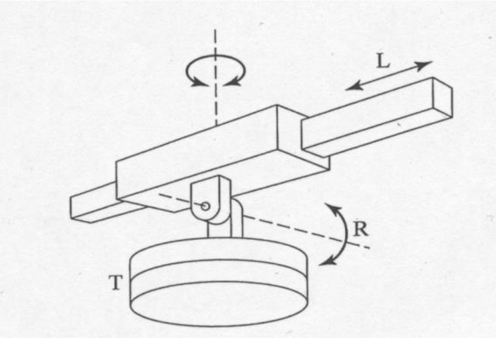
# Arm: Articulated Configuration

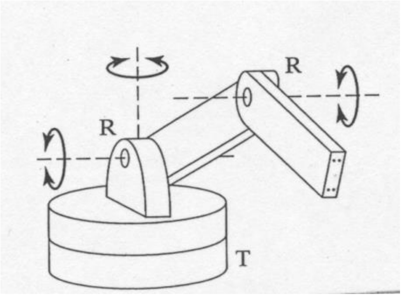


# Comparison of Arm Configurations



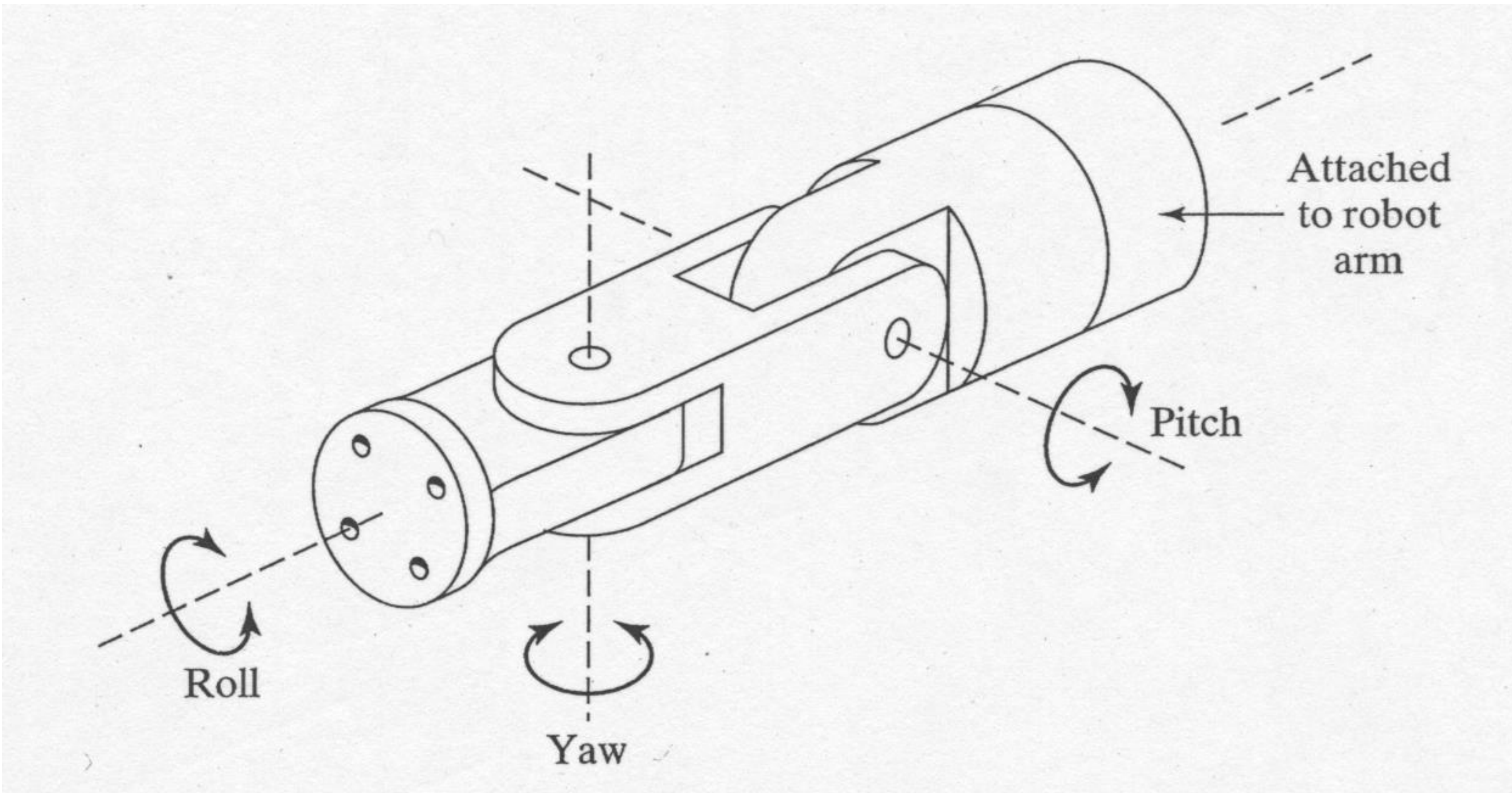



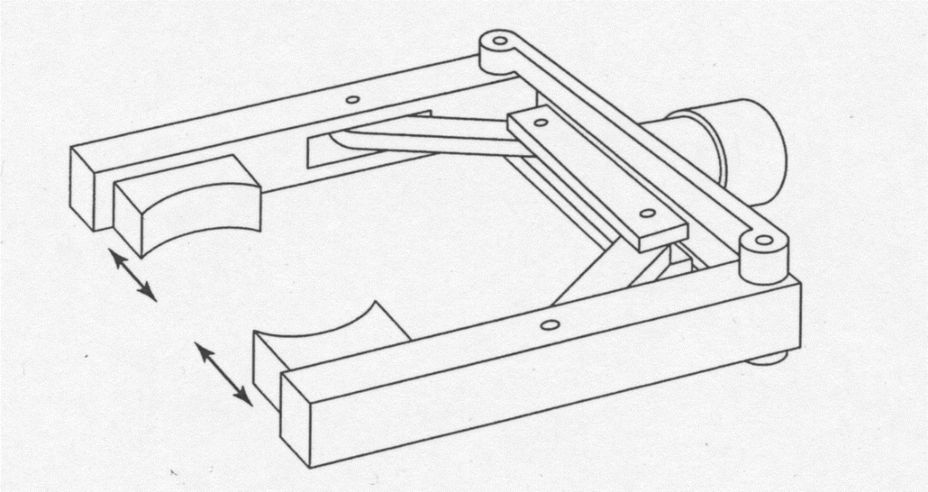

More Revolute Joints

# 3 DOF Wrist



# End Effectors

- Mechanical grippers
- Vacuum gripper
- Magnetic gripper
- Gripper with Adhesive strips



# Tools

- Welding guns
- Spray paint guns
- Spindle for drilling and milling
- Screw drivers
- Heating torch

# Robot Controllers

- Limited Sequence Control
  - Run one Joint at a time
- Point to Point Control
  - No synchronization of joint motion
- Continuous Path Control
  - Synchronized motion of joints

# Commonly Used Actuators

- Electric motors
  - Most commonly used
- Pneumatic cylinders
  - Used in small compact robots
- Hydraulic cylinders
  - Used in very large robots

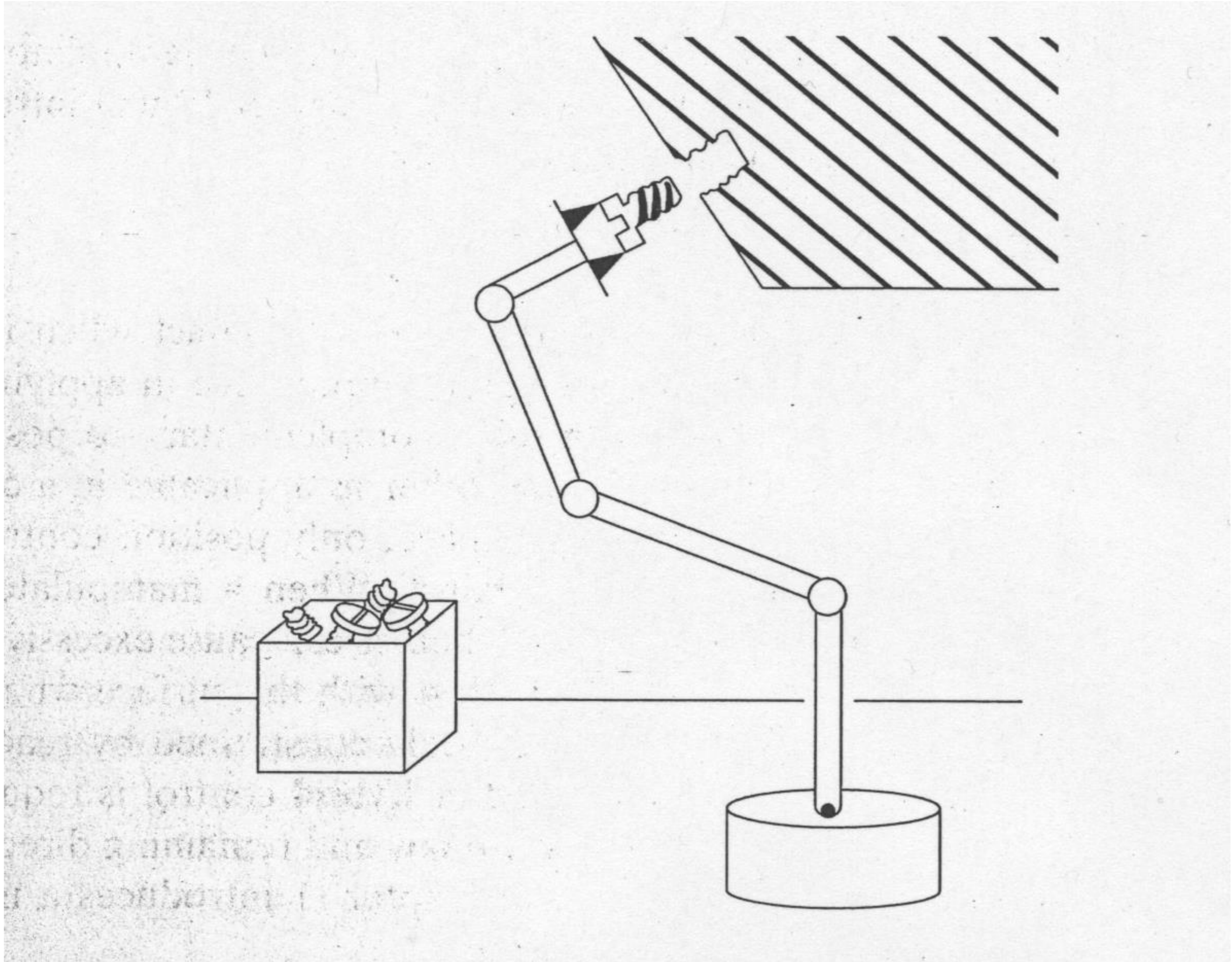
# Commonly Used Sensors

- Proximity sensor (also sometimes called range sensor)
- Tactile sensor (also sometimes called contact sensor)
- Machine vision

# Robot Programming

- Manual lead through
- Powered lead through
- Motion programming

# A Typical Task



# What do we need?

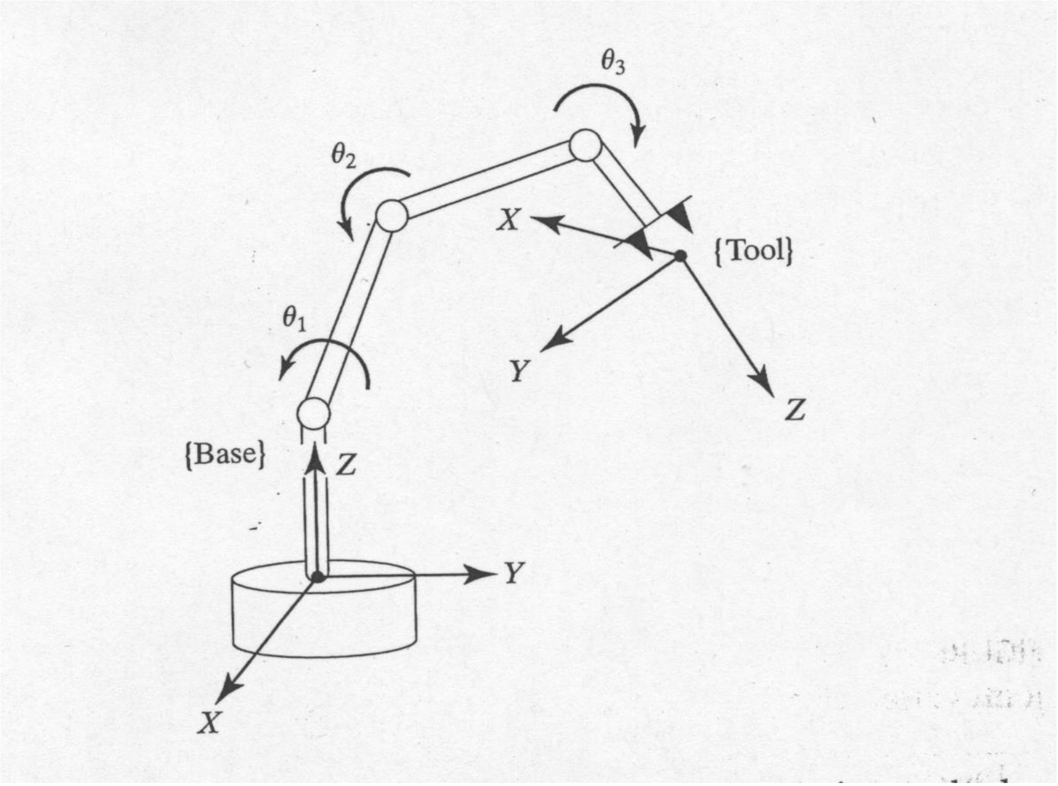
- How to move the robot through space to complete the specified task
  - What torques and forces to apply on the joints?

# Modeling

- The underlying mathematical model that describes the robot behavior
  - Will be needed to support off line programming
  - Will be needed to design robots
    - Selecting motors, link lengths and cross sections

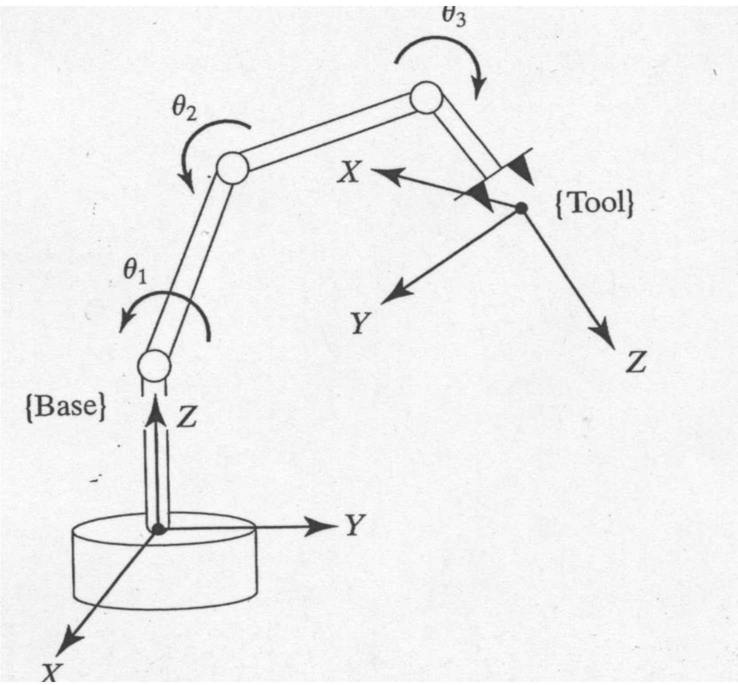
# Forward Kinematics

- Given joint parameters, determine the final end effecter location



# Inverse Kinematics

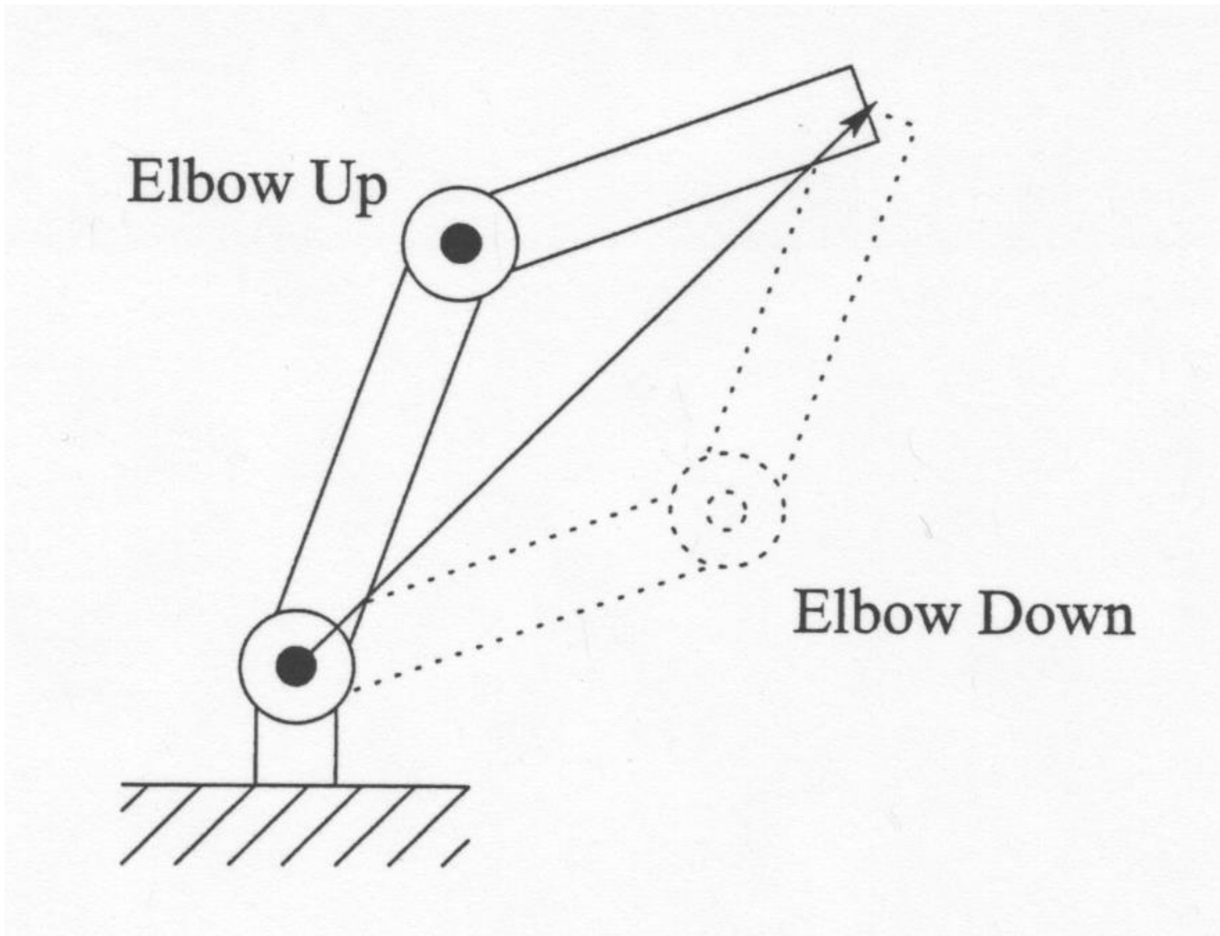
- Given desired end effecter position and orientation determine the joint parameters



# Inverse Kinematics Solutions

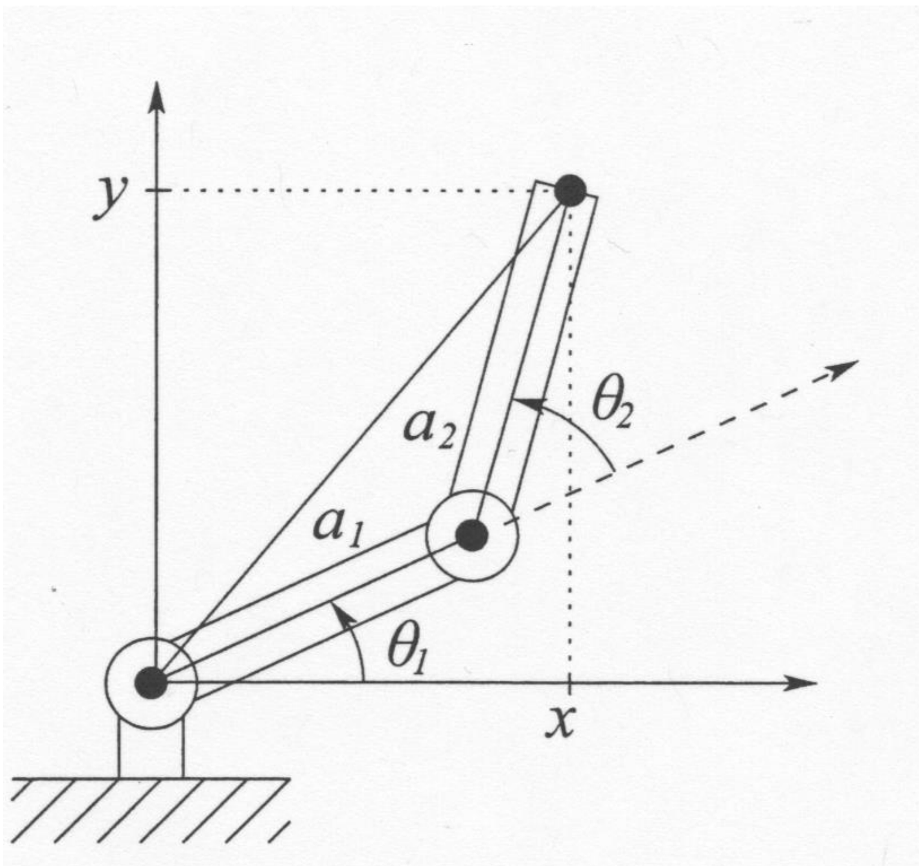
- Inverse kinematics may produce
  - One solution
  - Multiple solution
  - No solution

# Multiple Solution Case



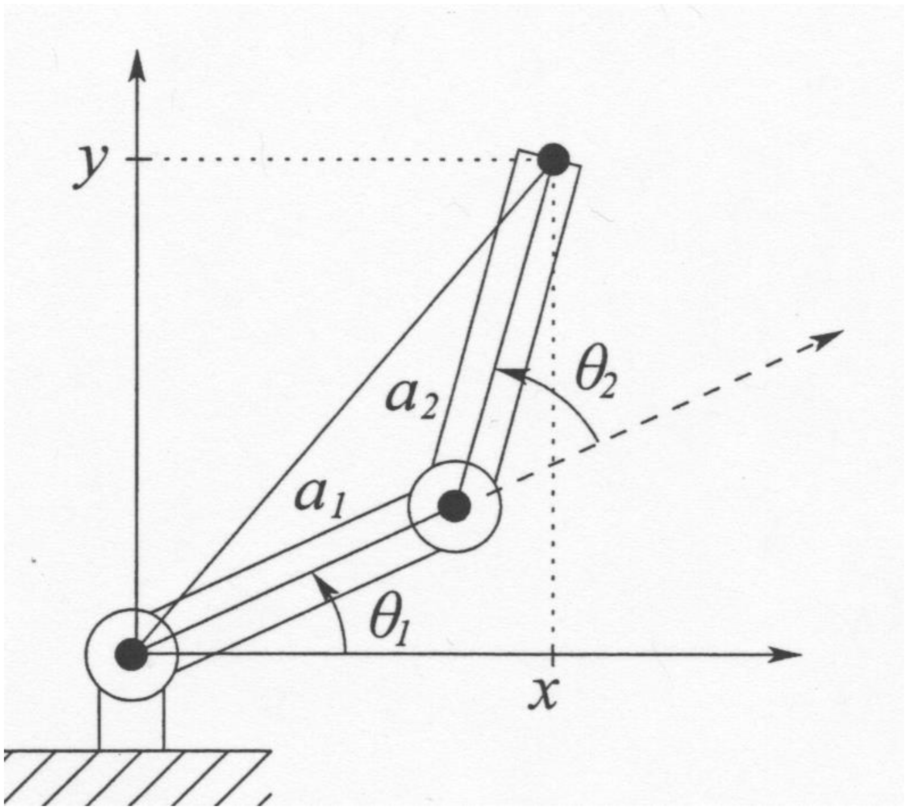
# No Solution Case

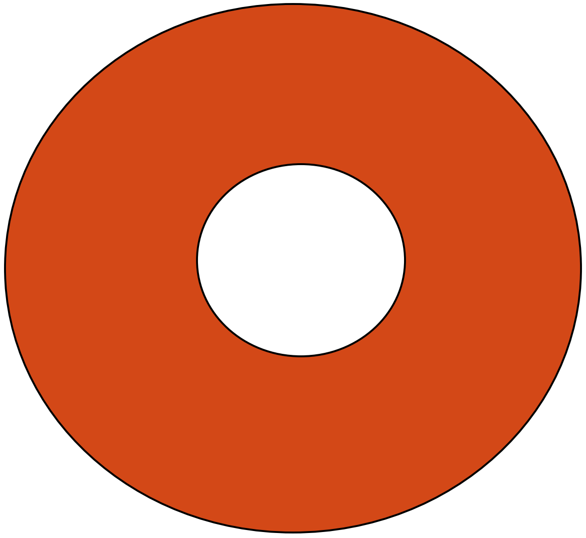
The selected location is outside the robot workspace



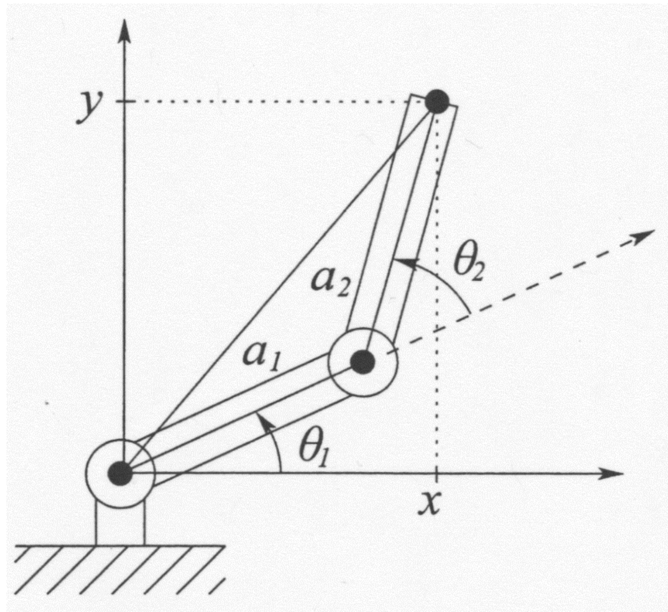
# Workspace

- The set of locations that can be reached by the robot



# Example

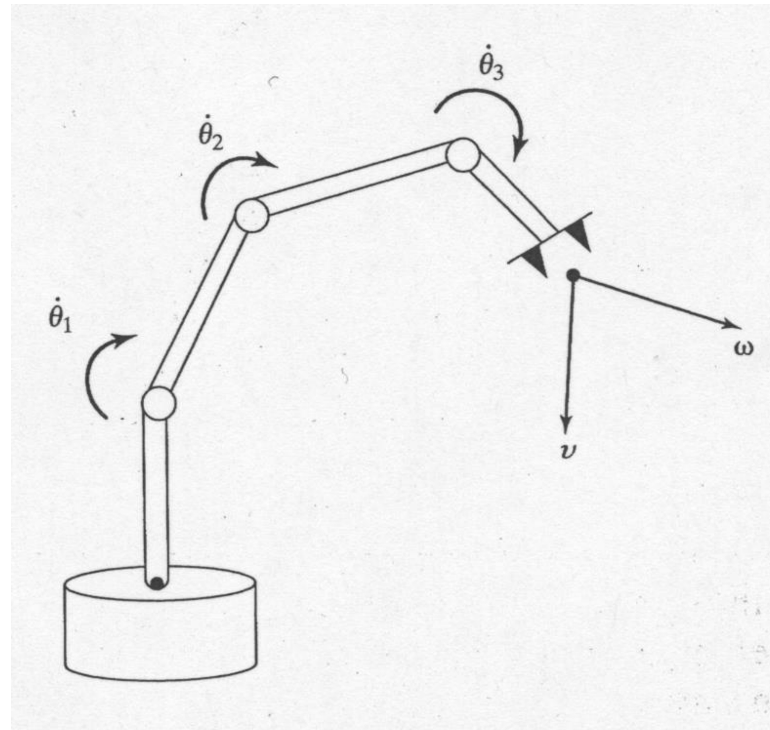



$$x = a_1 \cos\theta_1 + a_2 \cos(\theta_1 + \theta_2)$$

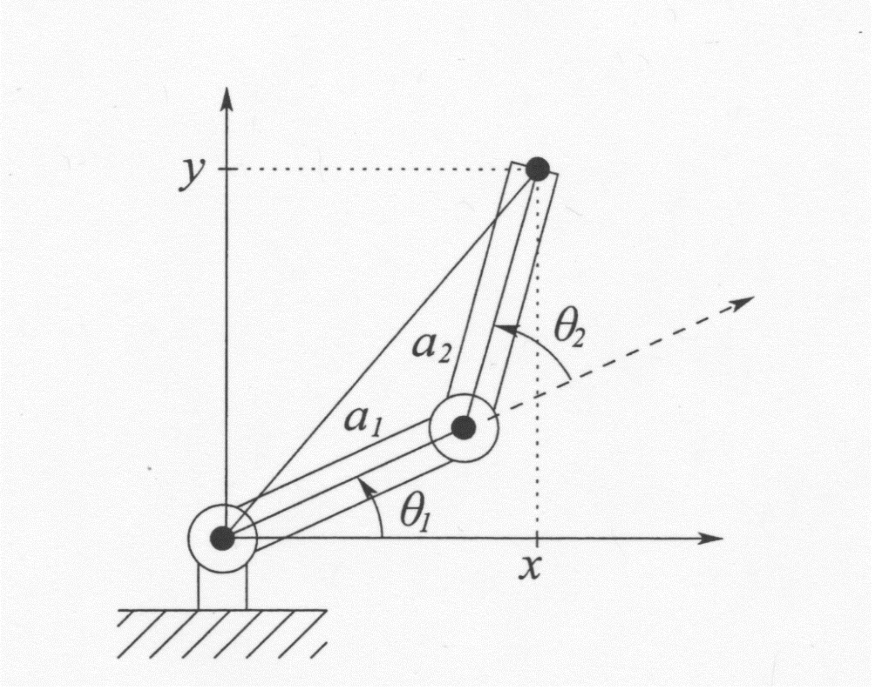
$$y = a_1 \sin\theta_1 + a_2 \sin(\theta_1 + \theta_2)$$

# Jacobian: Relating Velocities with Joint Velocities

Given joint velocities determine the desired end effecter velocities



# Example




Recall

$$x = a_1 \cos\theta_1 + a_2 \cos(\theta_1 + \theta_2)$$

$$y = a_1 \sin\theta_1 + a_2 \sin(\theta_1 + \theta_2)$$

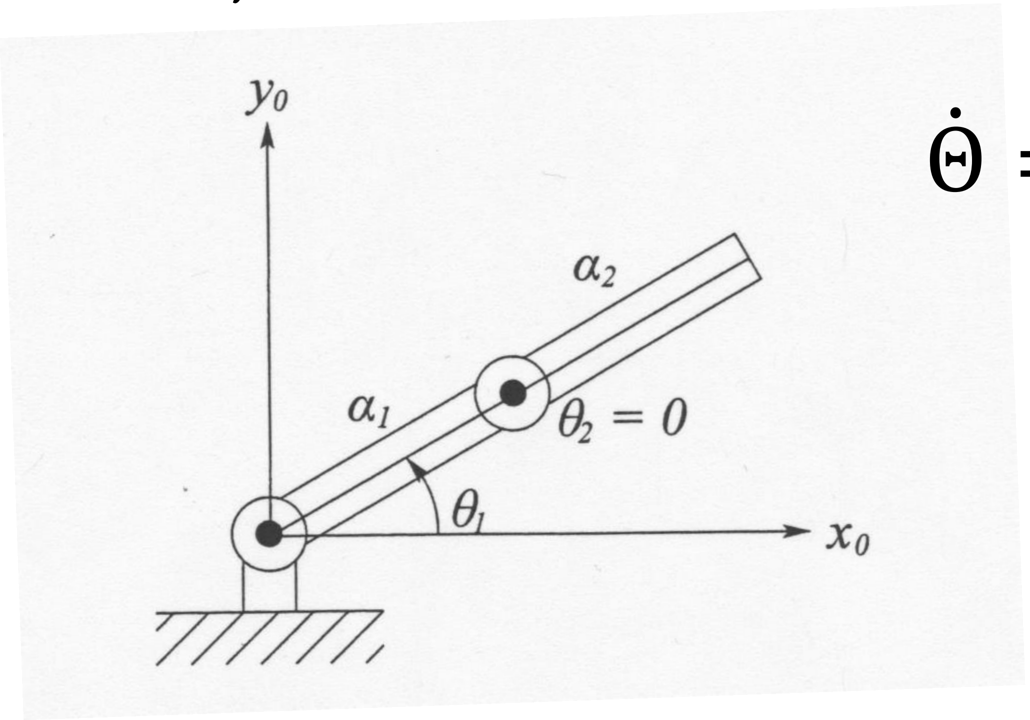
$$\Rightarrow \begin{pmatrix} \dot{x} \\ \dot{y} \end{pmatrix} = \begin{pmatrix} -a_1 \sin\theta_1 - a_2 \sin(\theta_1 + \theta_2) & -a_2 \sin(\theta_1 + \theta_2) \\ a_1 \cos\theta_1 + a_2 \cos(\theta_1 + \theta_2) & a_2 \cos(\theta_1 + \theta_2) \end{pmatrix} \begin{pmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{pmatrix}$$

Jacobian

$$\dot{x} = J\dot{\Theta}$$

# Achieving Desired End Effecter Velocities

- Jacobian matrix needs to be inverted to determine the joint velocities to achieve the desired end effecter velocity
  - This is not always possible
    - Matrix may not be invertible




$$\dot{\Theta} = J^{-1}\dot{x}$$

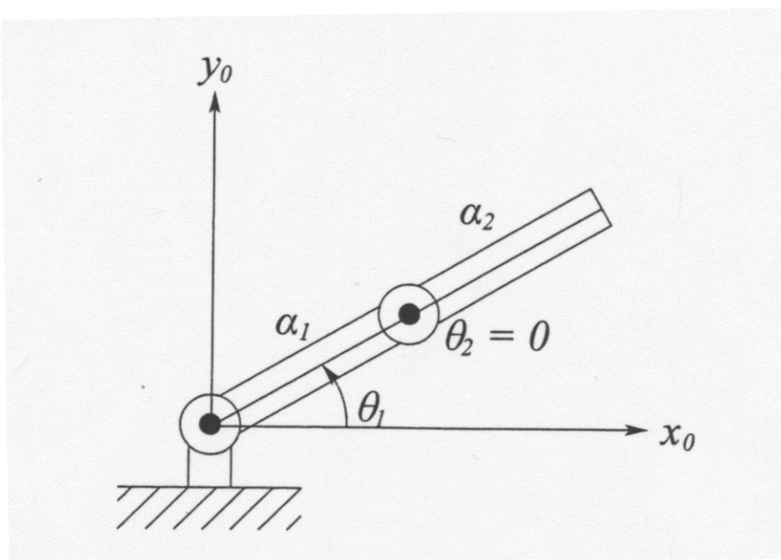
# Singularity

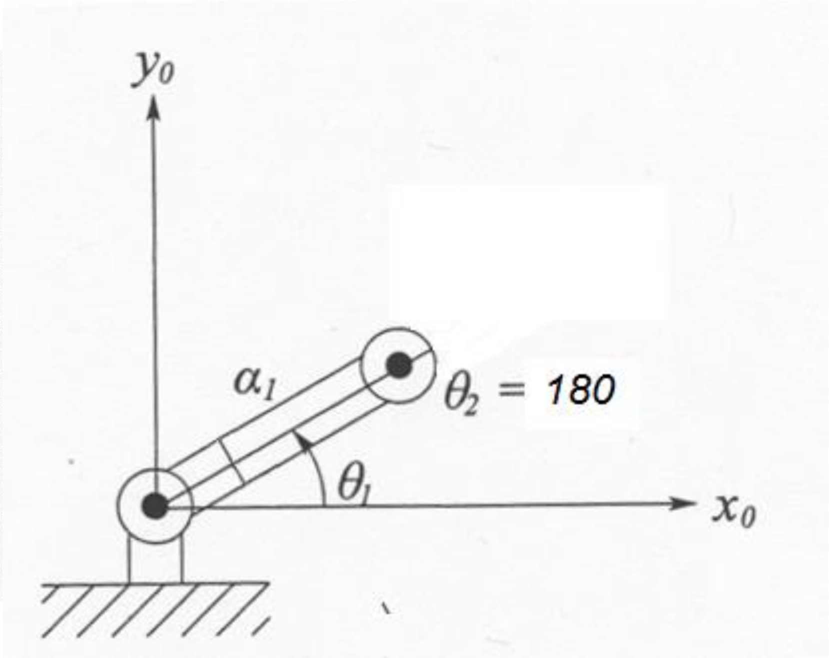
$$\dot{\Theta} = J^{-1}\dot{x}$$

Can it be guaranteed that J is always invertible?

$$\det(J) = a_1 a_2 \sin(\theta_2)$$

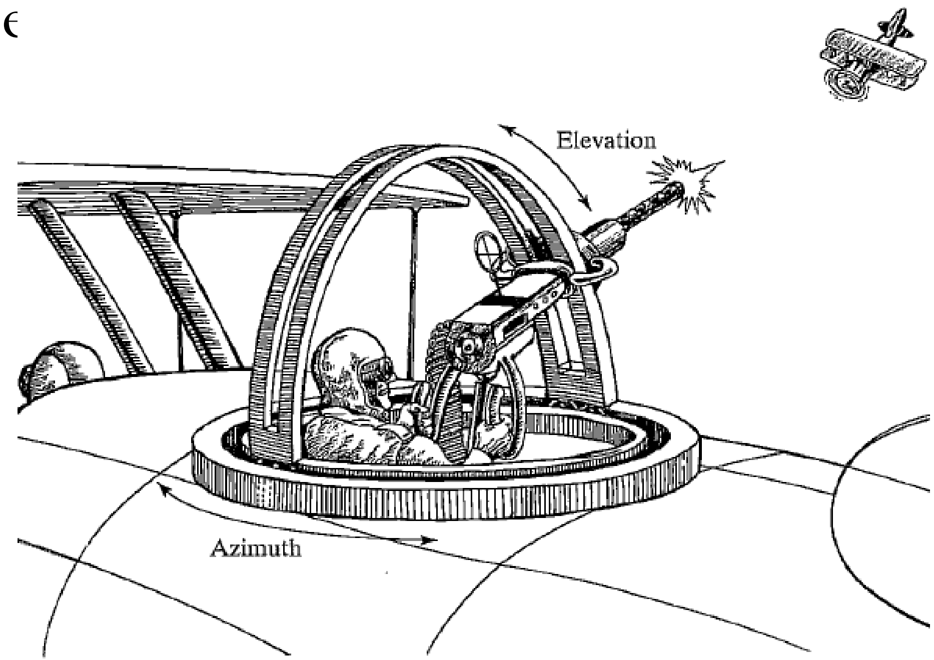
At $\theta_2$=0 or 180 degrees *J* can't be inverted



# Singularity

- Losing effective degrees of freedom
  - Cannot specify desired velocity
- Two types of singularitie
  - Work-space boundary
  - Workspace-interior

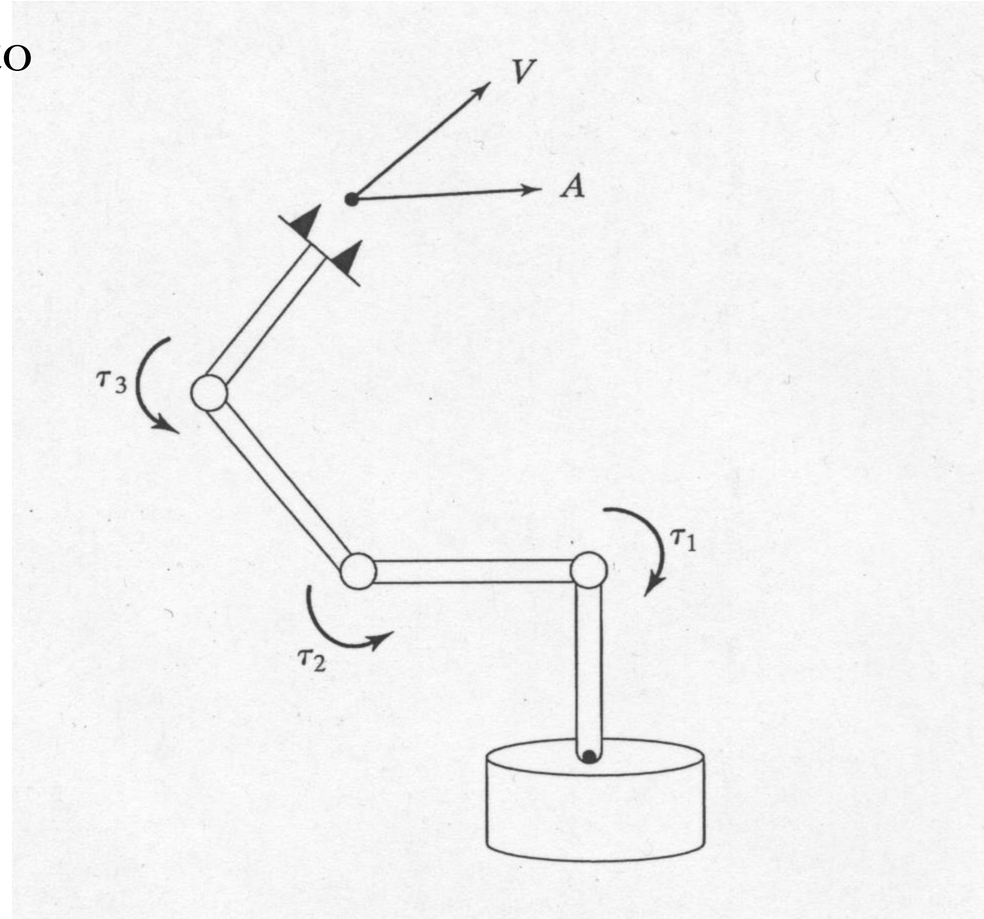



At elevation of zero degrees no effect of changing azimuth

Craig, 2005

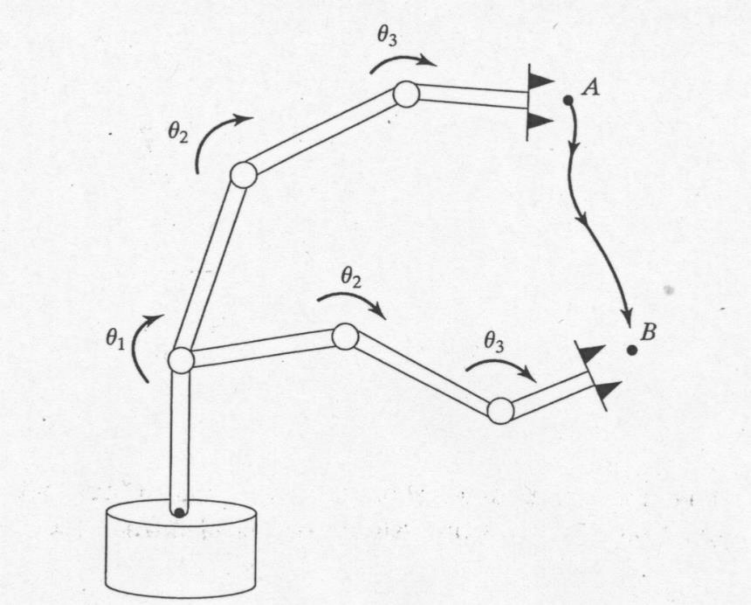
# Dynamics

What forces and torque need to be applied to joints to achieve the desired velocities and accelerations ?



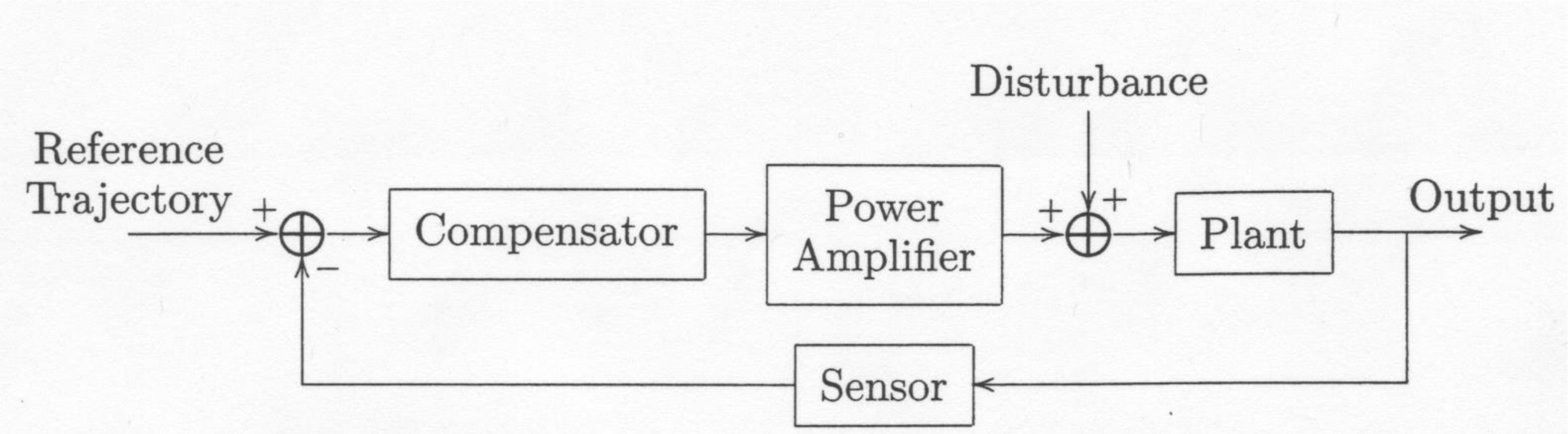
# Trajectory Generation

- How to trace a path through the space at the specified velocities



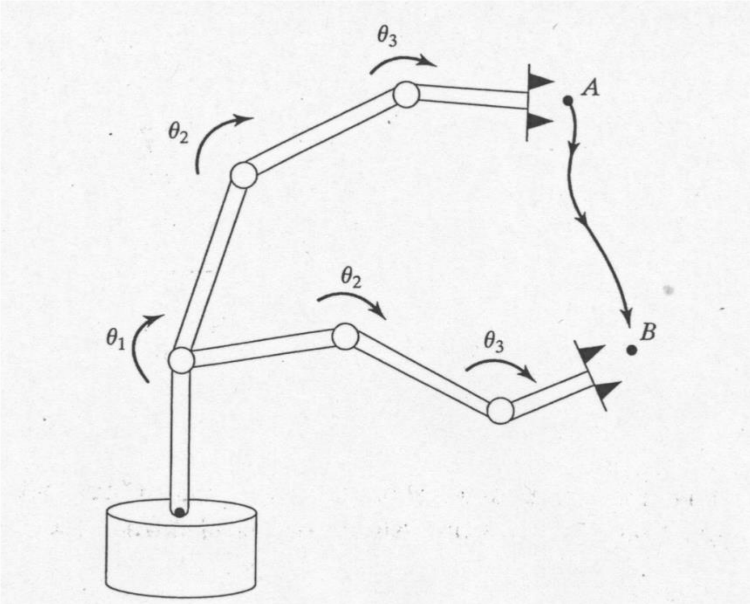
# Open Loop vs Closed Loop Control

- If we have a perfect model, then we can just position motors at the desired location with no feedback
- But this does not work in practice



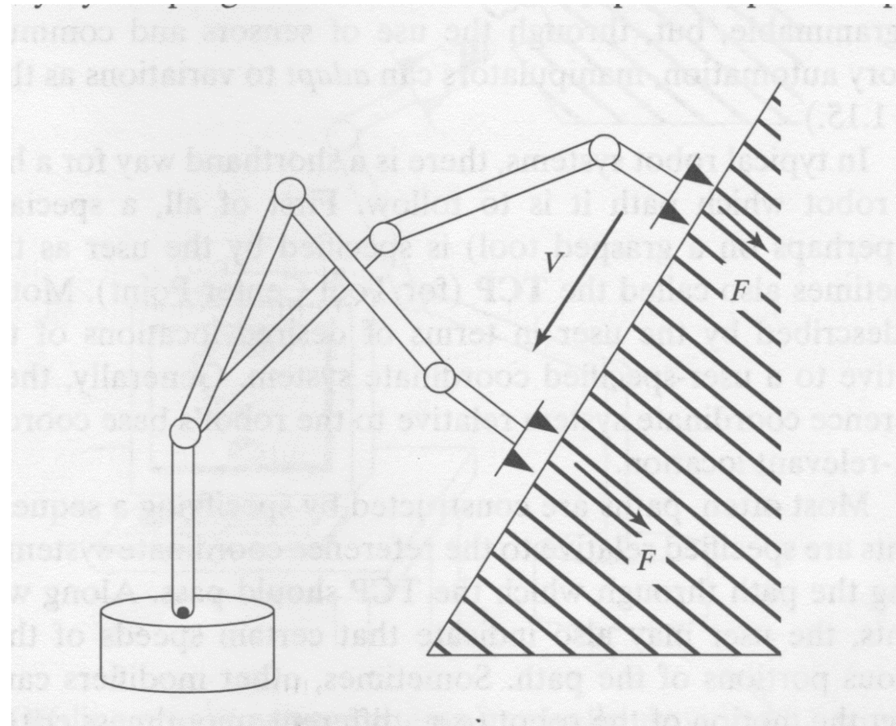
# Position Control

- How to compensate for errors and inaccuracies

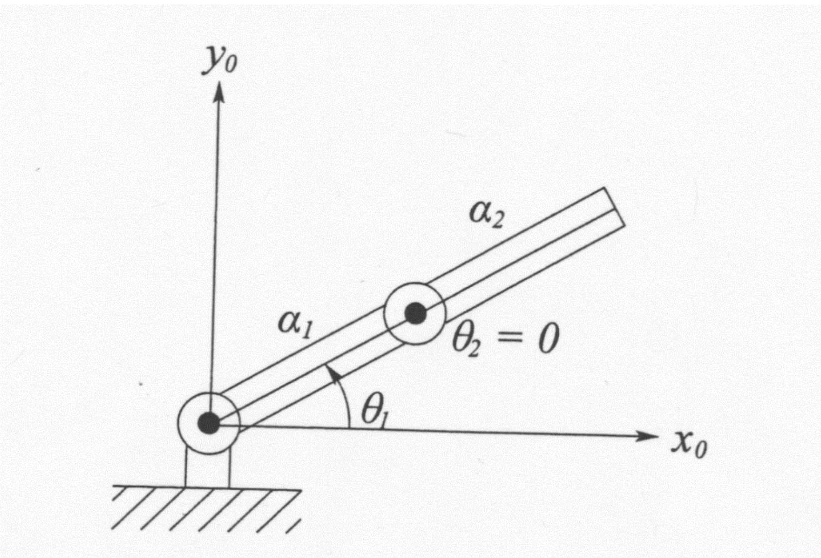


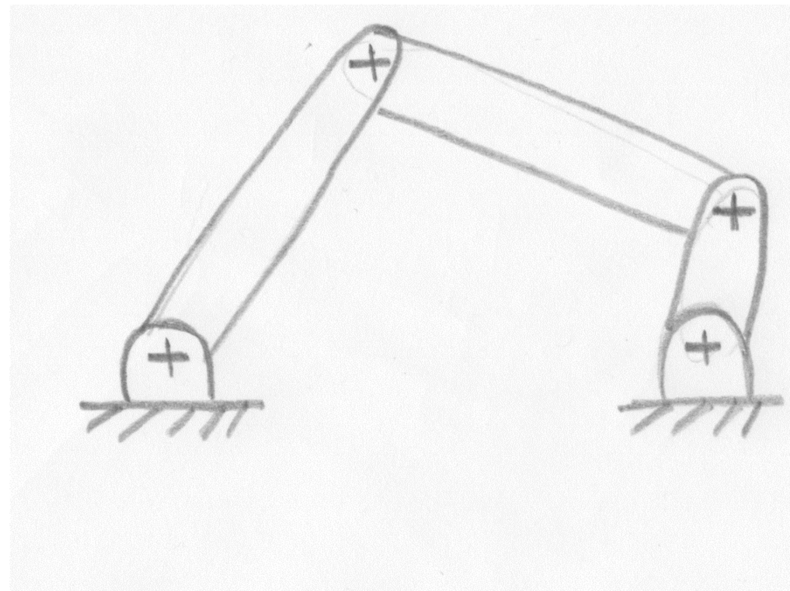
# Force Control

- Control force to ensure that robot can handle delicate objects and move more constrained surfaces



# Serial Mechanisms vs Closed Loop Mechanisms



# Accuracy vs Repeatability

- Accuracy is a measure of how close the robot reaches to the programmed point in the workspace
- Repeatability is measure of how close the robot reaches to the point previously reached by the robot (for the same programmed point)