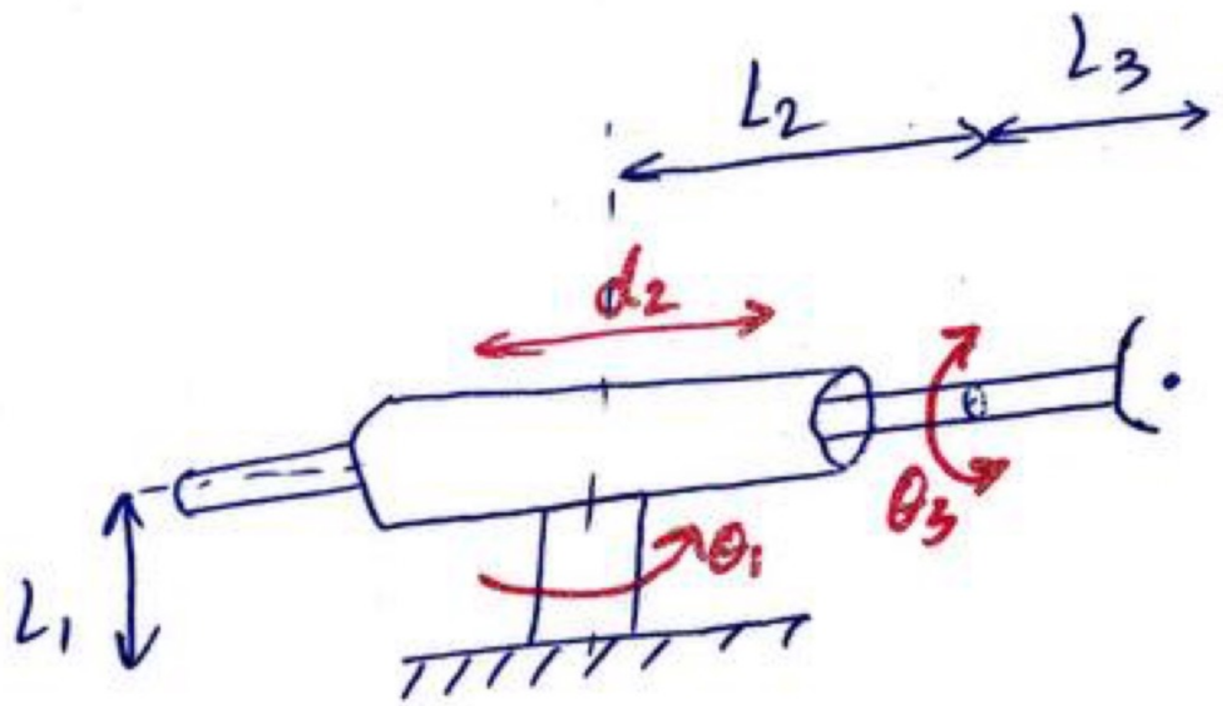
MTRN 3060: ROBOTICS and AUTOMATIONS

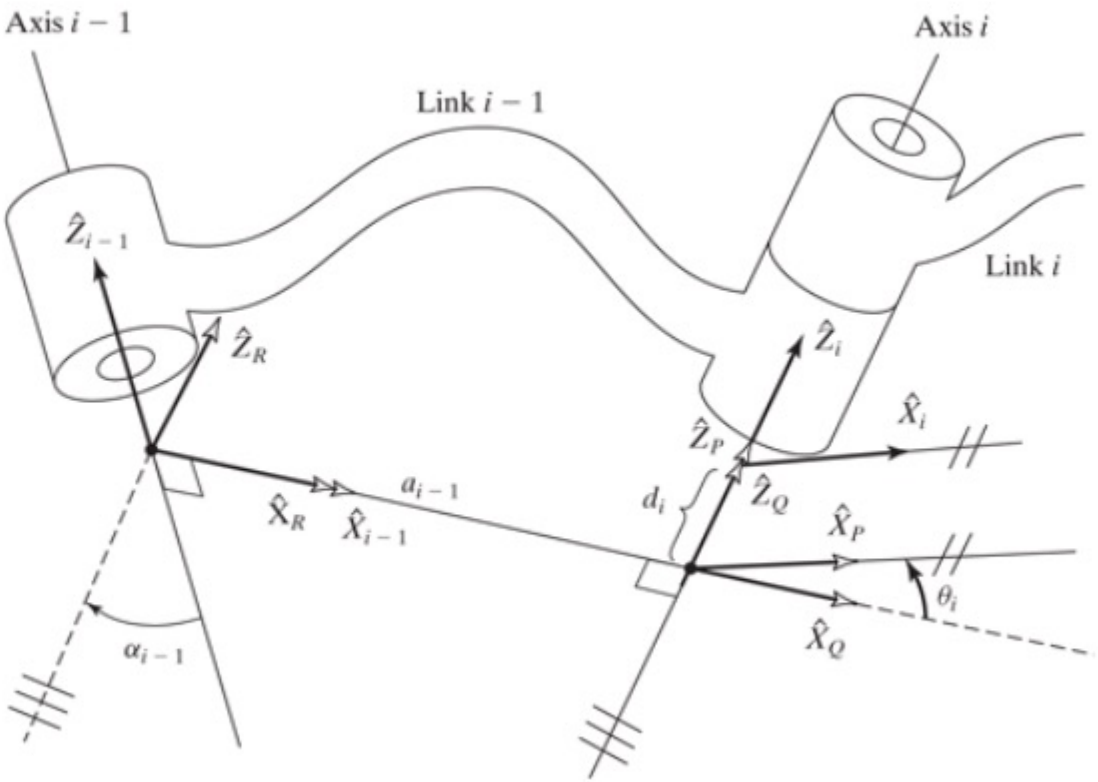
Week 2: Transformation operators, Component of Robots

Example 3 (from week 5)

Assign frames {0} to {4} for the shown non-planar RPR robotic manipulator, then find the DH parameters table



Forward Kinematics



$$ {}^{i-1}_{\ i}T = R_X(\alpha_{i-1}) D_X(a_{i-1}) R_Z(\theta_i) D_Z(d_i), $$

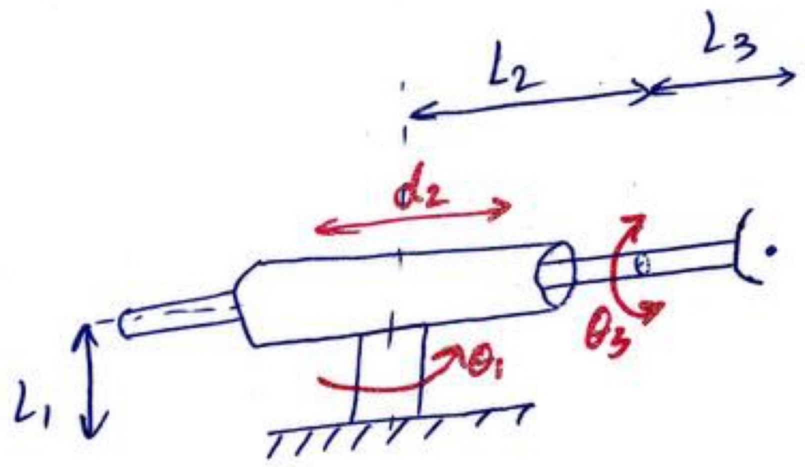
$$ {}^{i-1}_{\ i}T = \begin{bmatrix} c\theta_i & -s\theta_i & 0 & a_{i-1} \\ s\theta_i c\alpha_{i-1} & c\theta_i c\alpha_{i-1} & -s\alpha_{i-1} & -s\alpha_{i-1} d_i \\ s\theta_i s\alpha_{i-1} & c\theta_i s\alpha_{i-1} & c\alpha_{i-1} & c\alpha_{i-1} d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}. $$

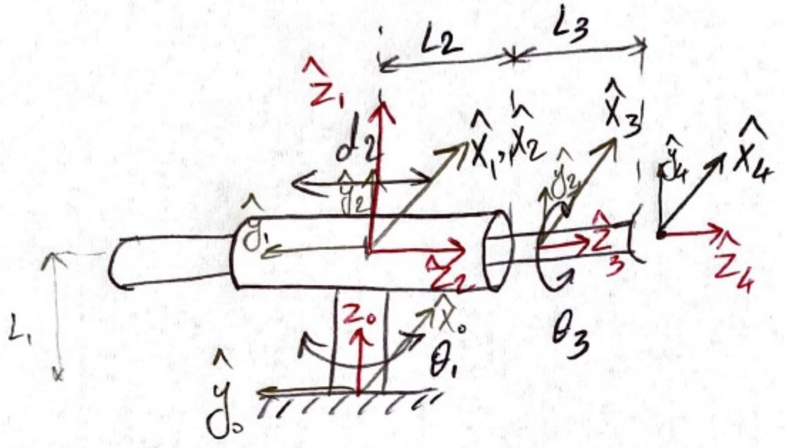
Forward Kinematics

Once the link frames have been defined and the corresponding link parameters found, developing the kinematic equations is straightforward. From the values of the link parameters, the individual link-transformation matrices can be computed. Then, the link transformations can be multiplied together to find the single transformation that relates frame {*N*} to frame {0}:

$$ {}^{0}_{N}T = {}^{0}_{1}T \, {}^{1}_{2}T \, {}^{2}_{3}T \ldots {}^{N-1}_{N}T. $$

Example





$$
{}^{i-1}_{i}T = \begin{bmatrix} c\theta_i & -s\theta_i & 0 & a_{i-1} \\ s\theta_i c\alpha_{i-1} & c\theta_i c\alpha_{i-1} & -s\alpha_{i-1} & -s\alpha_{i-1} d_i \\ s\theta_i s\alpha_{i-1} & c\theta_i s\alpha_{i-1} & c\alpha_{i-1} & c\alpha_{i-1} d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}.
$$

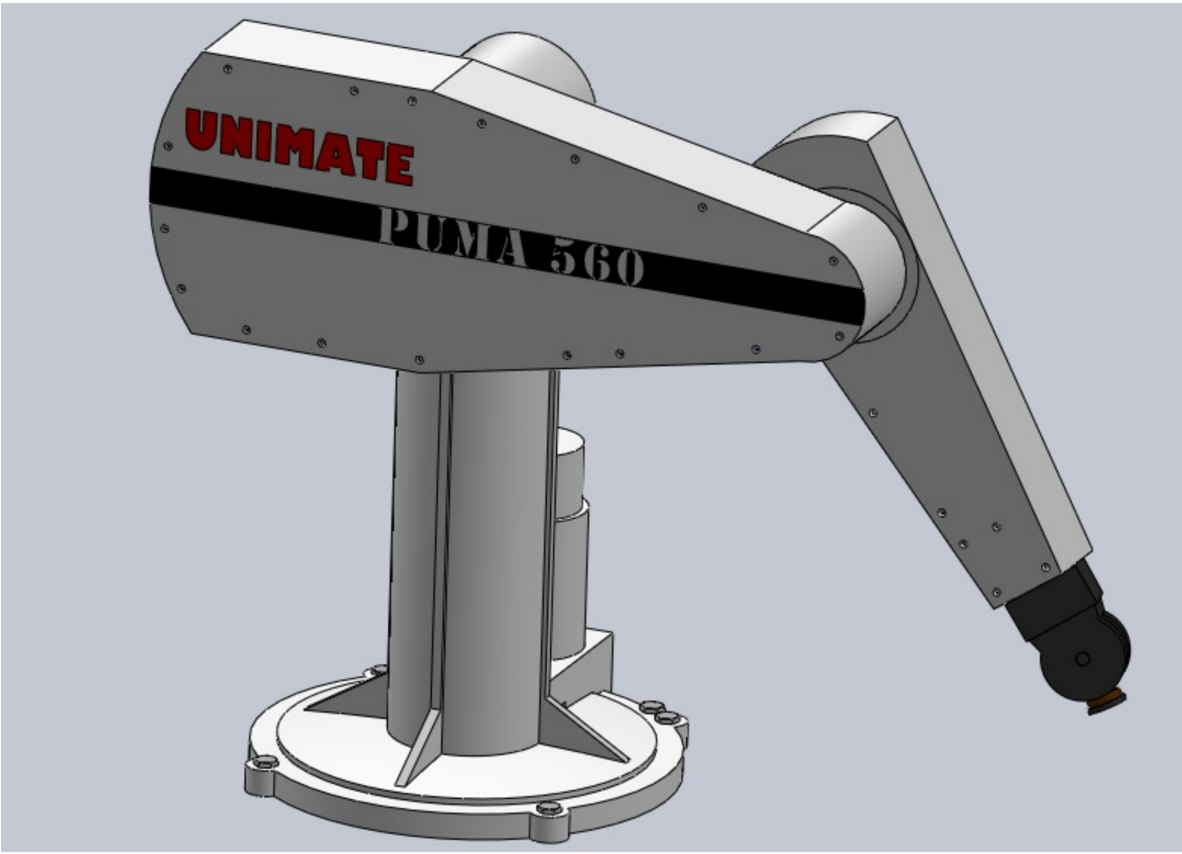
	Join Parameters		Link Parameters	
i	α_{i-1}	a_{i-1}	d_i	θ_i
1	0	0	L_1	θ_1
2	90°	0	d_2	0
3	0	0	L_2	θ_3
4	0	0	L_3	0

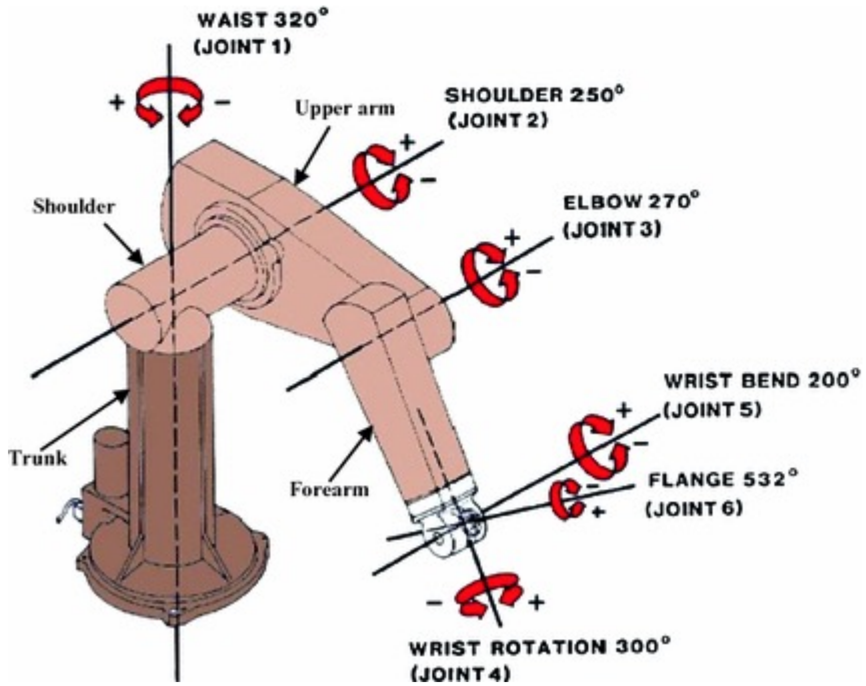
Actuator Space, Joint Space, and Cartesian Space

The position of all the links of a manipulator of n degrees of freedom can be specified with a set of *n* joint variables. This set of variables is often referred to as the n×1 joint vector. The space of all such joint vectors is referred to as **joint space**. If a robotic manipulator has 5 joints, then the join space will have 5 variables.

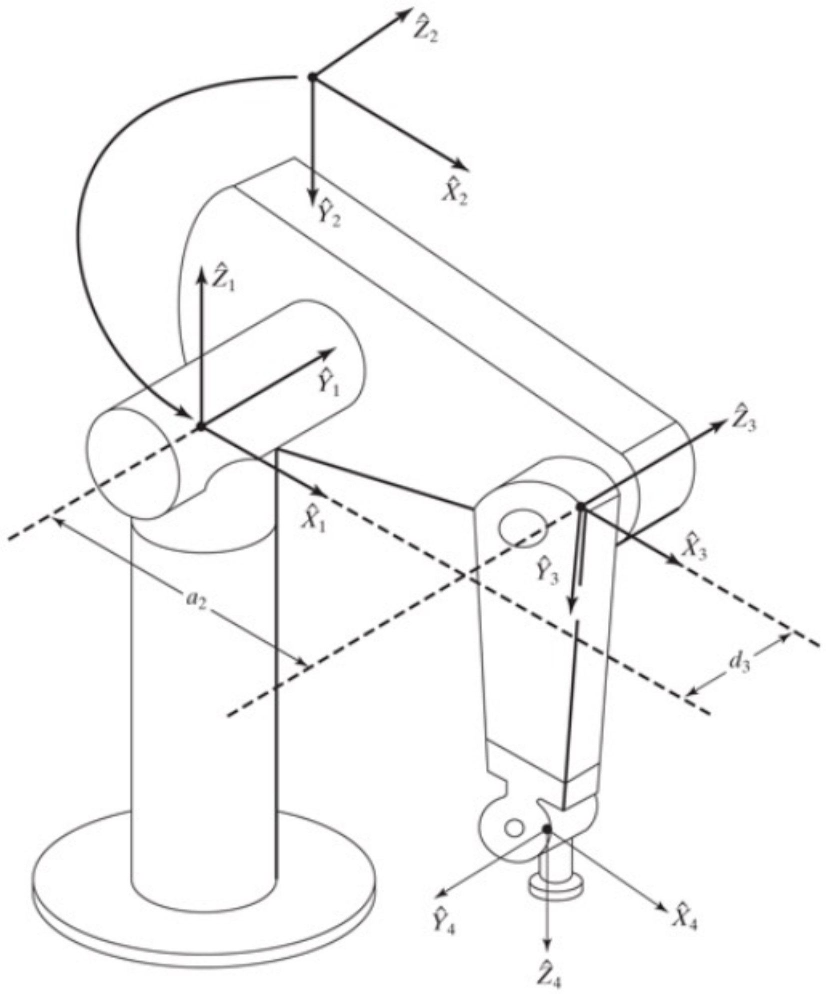
Cartesian space: is the space that contains the Cartesian variables where the end-effector can travel through within the workspace. For a planar robot, the Cartesian space will have 3 variables (Dx, Dy, Rz). For a full 3D robot, the Cartesian space will have 6 variables (Dx, Dy, Dz, Rx, Ry, Rz), which is the maximum number of Cartesian space variables.

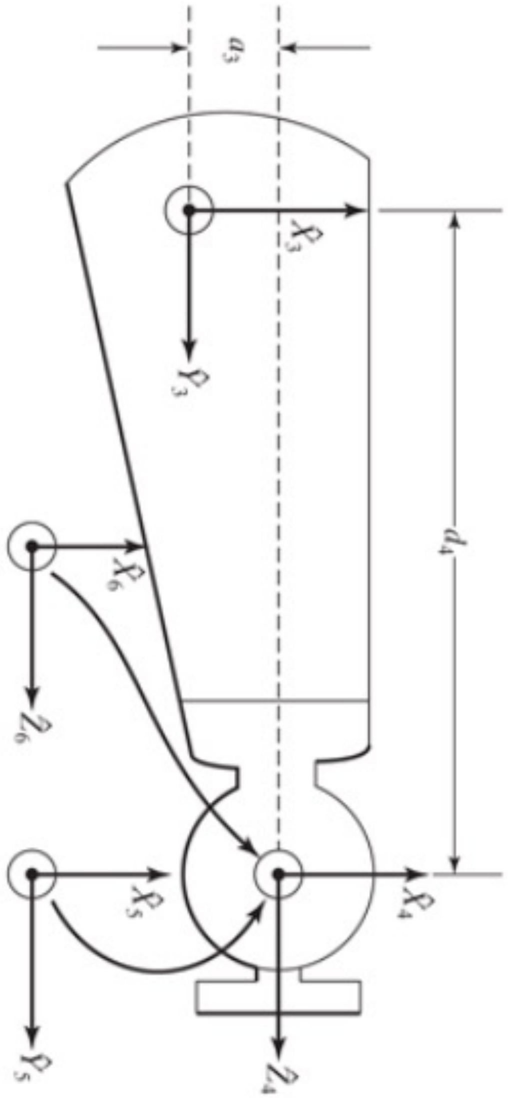
The Puma 560 Industrial Robot





The Puma 560 Industrial Robot





i	α_{i-1}	a_{i-1}	d_i	θ_i
1	0	0	0	θ_1
2	$-90°$	0	0	θ_2
3	0	a_2	d_3	θ_3
4	$-90°$	a_3	d_4	θ_4
5	$90°$	0	0	θ_5
6	$-90°$	0	0	θ_6

Link parameters of the PUMA 560

The Puma 560 Industrial Robot

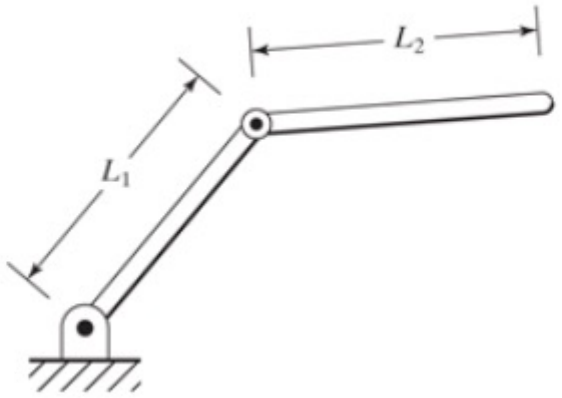
$$ {}^{0}_{1}T = \begin{bmatrix} c\theta_1 & -s\theta_1 & 0 & 0 \\ s\theta_1 & c\theta_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad {}^{1}_{2}T = \begin{bmatrix} c\theta_2 & -s\theta_2 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -s\theta_2 & -c\theta_2 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad {}^{2}_{3}T = \begin{bmatrix} c\theta_3 & -s\theta_3 & 0 & a_2 \\ s\theta_3 & c\theta_3 & 0 & 0 \\ 0 & 0 & 1 & d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

$$ {}^{3}_{4}T = \begin{bmatrix} c\theta_4 & -s\theta_4 & 0 & a_3 \\ 0 & 0 & 1 & d_4 \\ -s\theta_4 & -c\theta_4 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad {}^{4}_{5}T = \begin{bmatrix} c\theta_5 & -s\theta_5 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ s\theta_5 & c\theta_5 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad {}^{5}_{6}T = \begin{bmatrix} c\theta_6 & -s\theta_6 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -s\theta_6 & -c\theta_6 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

$$ {}^{0}_{6}T = {}^{0}_{1}T\, {}^{1}_{6}T = \begin{bmatrix} r_{11} & r_{12} & r_{13} & p_x \\ r_{21} & r_{22} & r_{23} & p_y \\ r_{31} & r_{32} & r_{33} & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

Inverse Manipulator Kinematics

Given the desired position and orientation of the tool relative to the station, how do we compute the set of joint angles which will achieve this desired result?



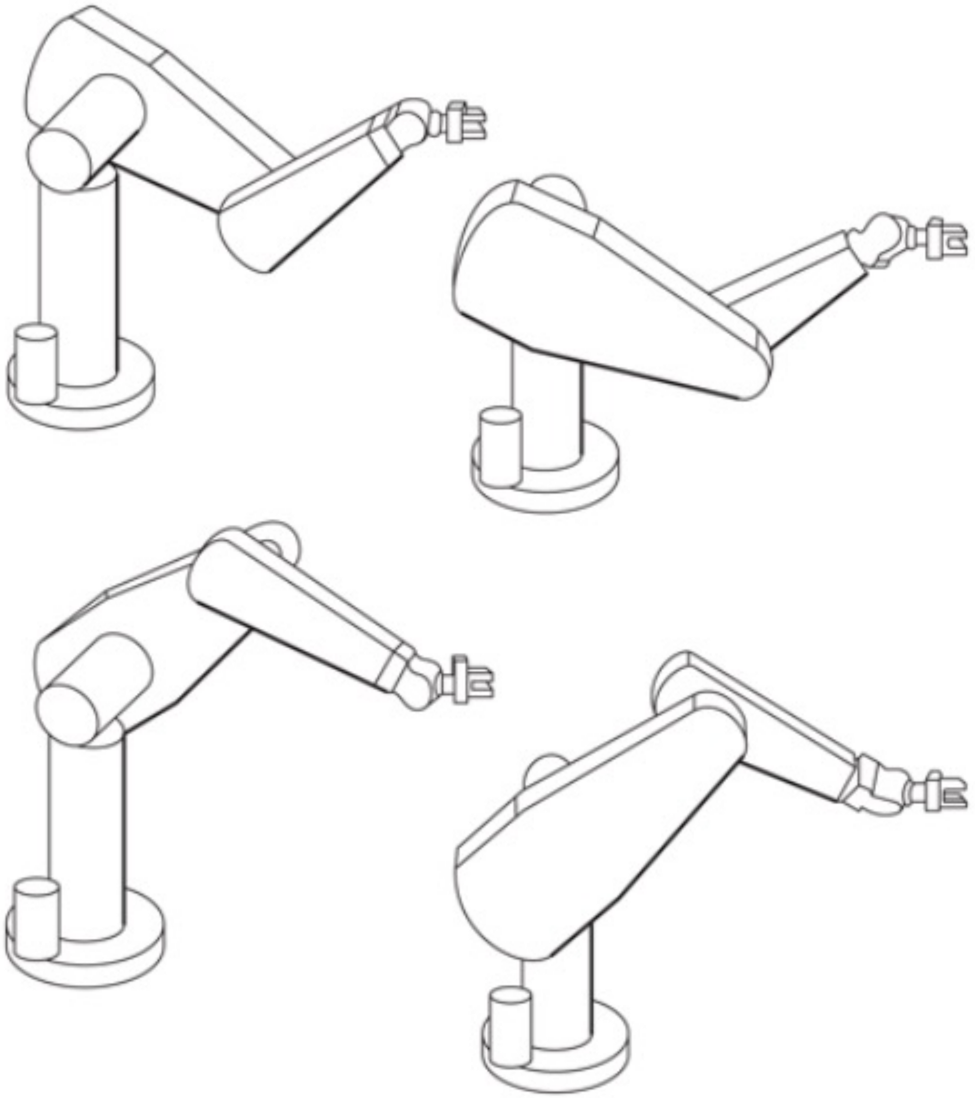
Solvability of inverse Kinematic Problems

a) Existence of a Solution
b) Workspace: is a volume of space that the end-effector can reach.
- For solution to exist, the end-effector goal must be within the workspace.

a) Degreed of Freedom (DOF): Degrees of freedom are the number of independent variable in a manipulator.
-For a solution to exist, the number of joints should be more than or equal to the number of independent Cartesian coordinates to be controlled at the end effector

Multiple Solutions

The number of solutions depends upon the number of joints in the manipulator but is also a function of the link parameters and the allowable ranges of motion of the joints.

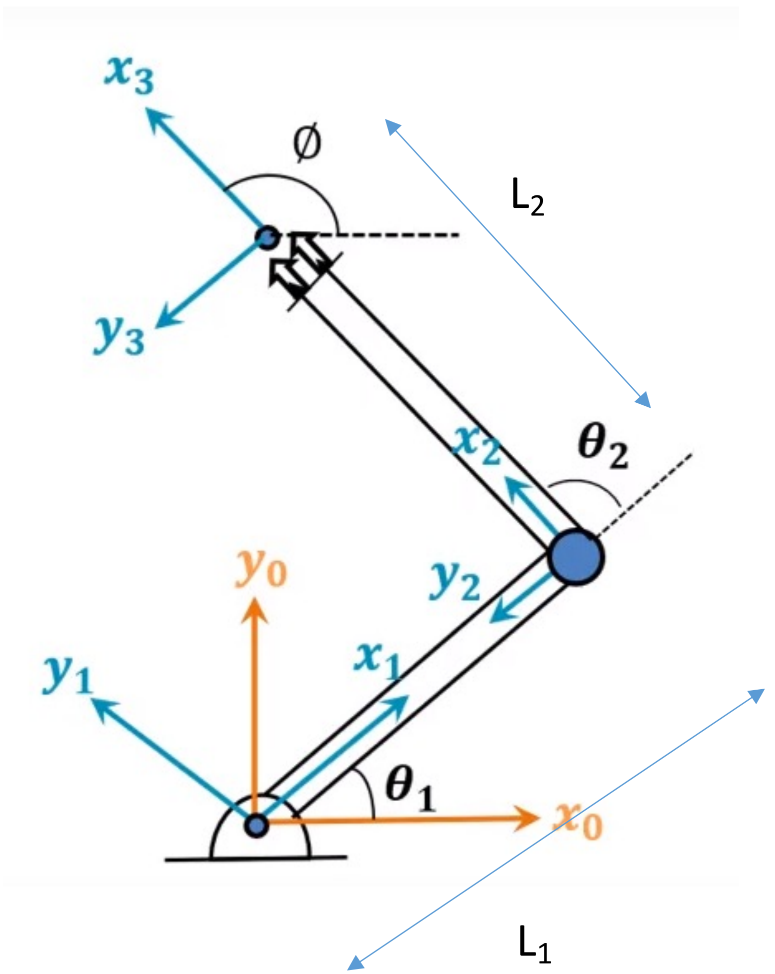


Inverse Manipulator Kinematics Solutions

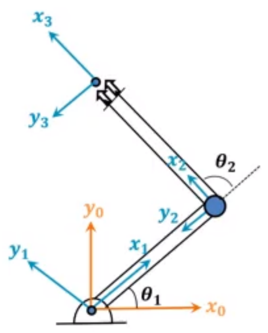
$$[{}^0_1\mathrm{T}] = \begin{bmatrix} c_1 & -s_1 & 0 & 0 \\ s_1 & c_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$[{}^1_2\mathrm{T}] = \begin{bmatrix} c_2 & -s_2 & 0 & L_1 \\ s_2 & c_2 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$[{}^2_3\mathrm{T}] = \begin{bmatrix} 1 & 0 & 0 & L_2 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$



$$[{}^0_3\mathrm{T}] = [{}^0_1\mathrm{T}][{}^1_2\mathrm{T}][{}^2_3\mathrm{T}]$$



$$[{}^0_3\mathrm{T}] = \begin{bmatrix} c_1 & -s_1 & 0 & 0 \\ s_1 & c_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c_2 & -s_2 & 0 & L_1 \\ s_2 & c_2 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & L_2 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$[{}^0_3\mathrm{T}] = \begin{bmatrix} c_1c_2 - s_1s_2 & -c_1s_2 - s_1c_2 & 0 & c_1L_1 \\ s_1c_2 + c_1s_2 & -s_1s_2 + c_1c_2 & 0 & s_1L_1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & L_2 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$[{}^0_3\mathrm{T}] = \begin{bmatrix} c_{12} & -s_{12} & 0 & c_1L_1 \\ s_{12} & c_{12} & 0 & s_1L_1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & L_2 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Trigonometric Identities:

$\cos(a+b) = c_{ab} = c_a c_b - s_a s_b$

$\cos(a-b) = c_a c_b + s_a s_b$

$\sin(a+b) = s_{ab} = s_a c_b + c_a s_b$

$\sin(a-b) = s_a c_b - c_a s_b$

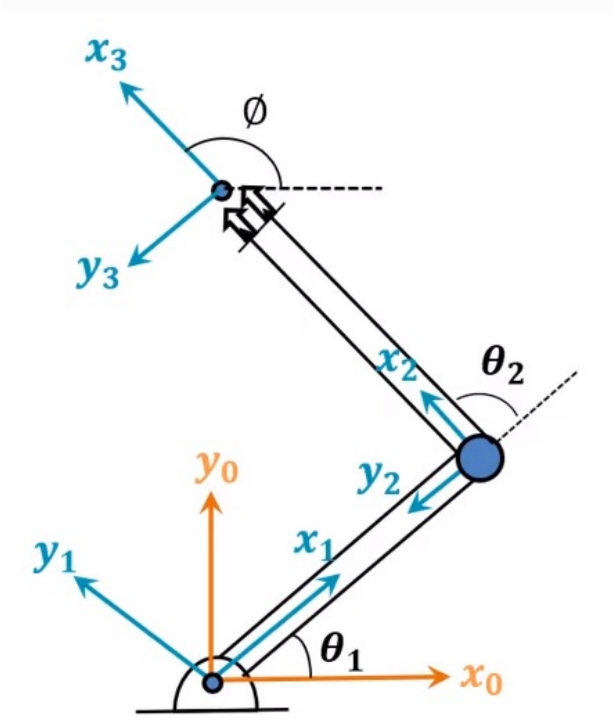
${s_a}^2 + {c_a}^2 = 1$

$$[{}^0_3\mathrm{T}] = \begin{bmatrix} c_{12} & -s_{12} & 0 & c_1L_1 \\ s_{12} & c_{12} & 0 & s_1L_1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & L_2 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$[{}^0_3\mathrm{T}] = \begin{bmatrix} c_{12} & -s_{12} & 0 & c_{12}L_2 + c_1L_1 \\ s_{12} & c_{12} & 0 & s_{12}L_2 + s_1L_1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

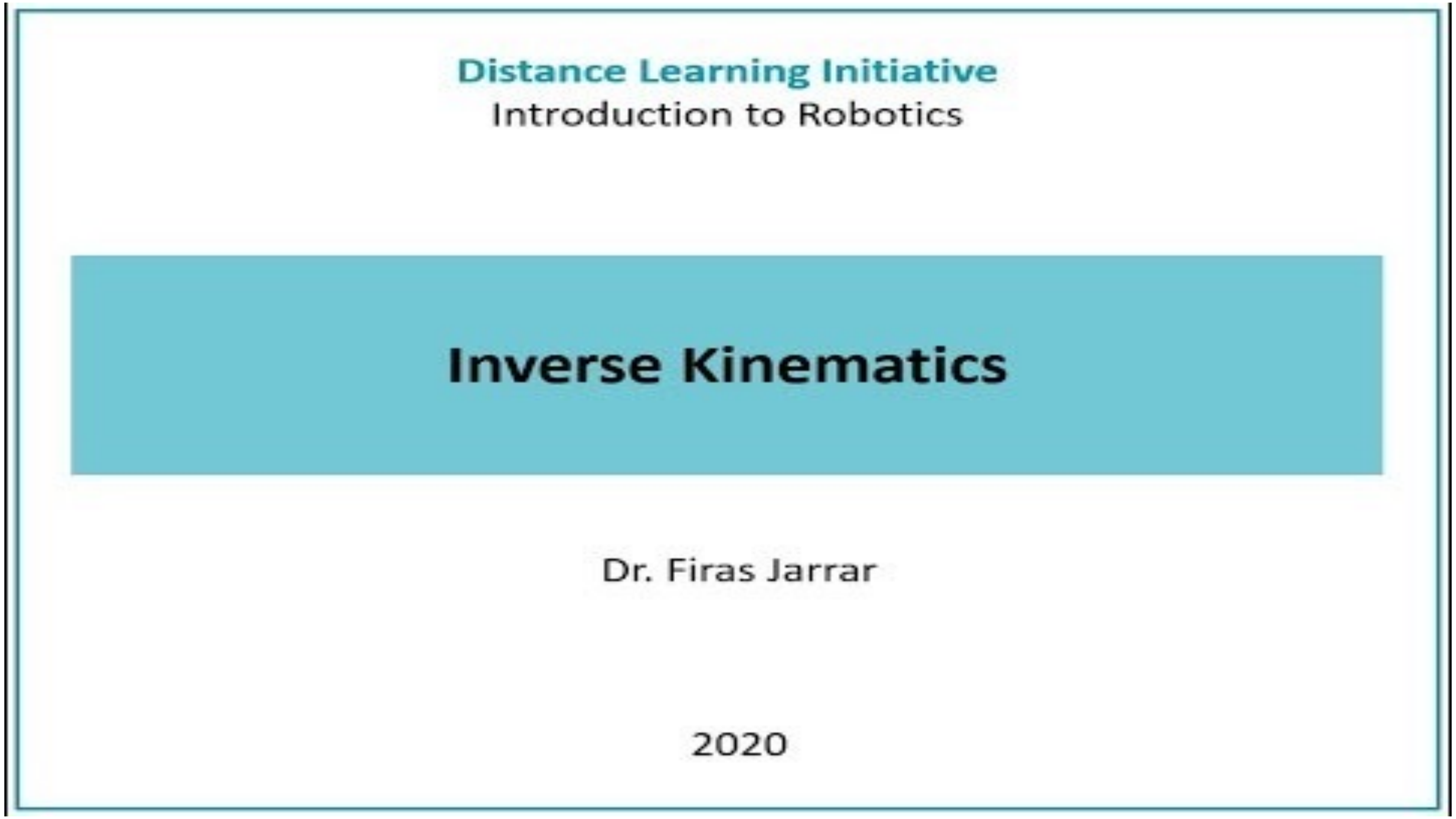
Compare this to

$$[{}^0_3\mathrm{T}] = \begin{bmatrix} r_{11} & r_{12} & r_{13} & x \\ r_{21} & r_{22} & r_{23} & y \\ r_{31} & r_{32} & r_{33} & \mathrm{z} \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} c_\emptyset & -s_\emptyset & 0 & x \\ s_\emptyset & c_\emptyset & 0 & y \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$



$$c_\emptyset = c_{12}$$
$$s_\emptyset = s_{12}$$
$$x = c_{12}L_2 + c_1L_1$$
$$y = s_{12}L_2 + s_1L_1$$

Here is how to solve the equation



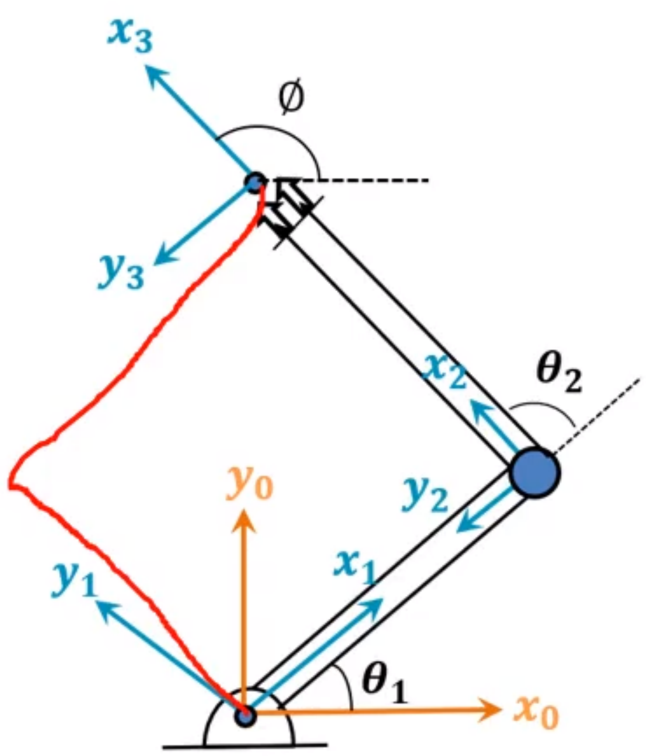
https://www.youtube.com/watch?v=4Y9hZXfEnkI&t=189s&ab_channel=Dr.FirasJarrar

$$c_2 = \frac{x^2+y^2-{L_1}^2-{L_2}^2}{2L_1L_2};$$

$$s_2 = \pm\sqrt{1-{c_2}^2}\,;$$

$$\theta_2 = \mathrm{Atan2}\left(\frac{s_2}{c_2}\right)$$

$$\theta_1 = \mathrm{Atan2}\left(\frac{y}{x}\right) - \mathrm{Atan2}\left(\frac{s_2L_2}{c_2L_2+L_1}\right)$$



x & y → c_2 → s_2 → θ_2 → θ_1

x & y → c_2 → s_2 → θ_2 → θ_1

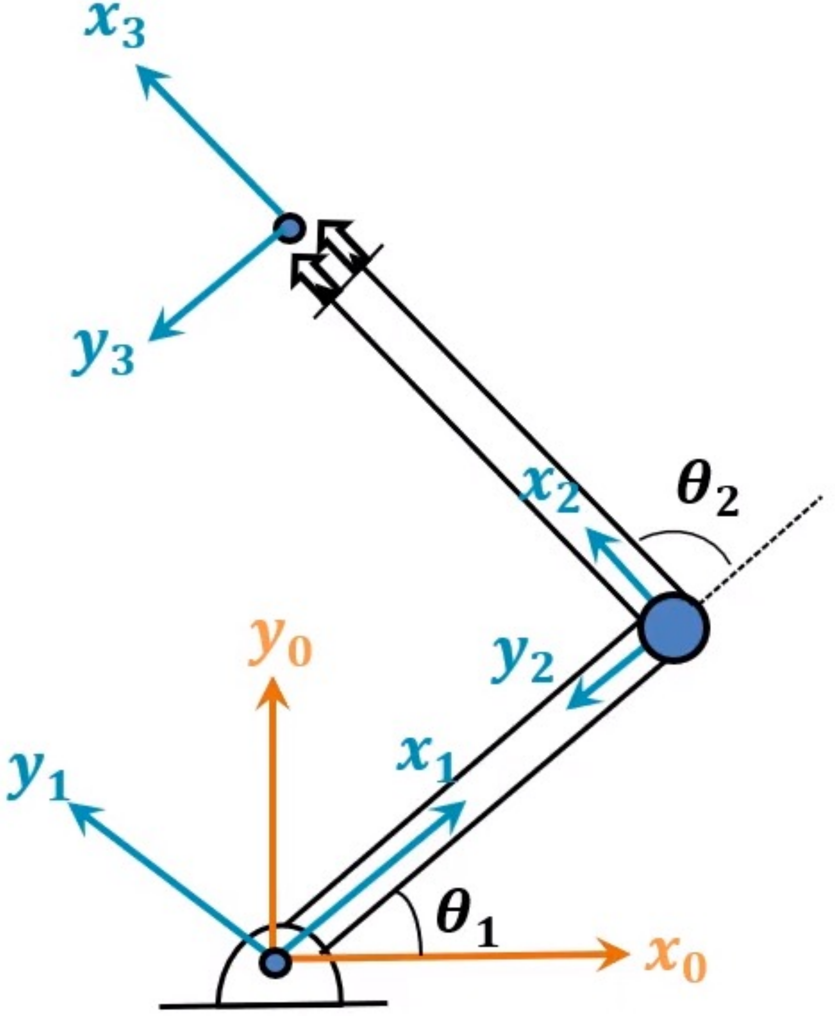
$c_2 = \frac{x^2+y^2-{L_1}^2-{L_2}^2}{\dots}$ $s_2 = \pm\sqrt{1-{c_2}^2}$ $\theta_2 = \mathrm{Atan2}\left(\frac{s_2}{\dots}\right)$ $\mathrm{Atan2}\left(\frac{y}{\dots}\right) - \mathrm{Atan2}\left(\frac{s_2L_2}{\dots}\right)$

Example:

For the planar 2DOF RR robotic arm shown in the figure, find the values of θ_1 and θ_2 required to accomplish a desired end-effector position $x = -1.12\ \text{cm}$ and $y = 24.52\ \text{cm}$.

Note that:

$L_1 = 25\ \text{cm}$ and $L_2 = 20\ \text{cm}$



$$c_2 = \frac{x^2 + y^2 - {L_1}^2 - {L_2}^2}{2L_1L_2} = \frac{(-1.12)^2 + (24.52)^2 - (25)^2 - (20)^2}{2(25)(20)} = -0.4225$$

$$s_2 = \pm\sqrt{1 - {c_2}^2} = \pm\sqrt{1 - (-0.4225\ )^2} = \pm 0.9064$$

```
>> atan2d (+0.9064,-0.4225)

ans =

  114.9916

>> atan2d (-0.9064,-0.4225)

ans =

 -114.9916
```

$$\theta_2 = Atan2\left(\frac{s_2}{c_2}\right)$$

$$\theta_2 = Atan2\left(\frac{+0.9064}{-0.4225}\right) \quad \text{or} \quad \theta_2 = Atan2\left(\frac{-0.9064}{-0.4225}\right)$$

$$\theta_2 = 115^o \quad \text{or} \quad \theta_2 = 245^o \quad (= -115^o + 360^o\ )$$

$$\theta_1 = A\tan 2\left(\frac{y}{x}\right) - A\tan 2\left(\frac{s_2 L_2}{c_2 L_2 + L_1}\right)$$

$$\theta_2 = 115^o:$$

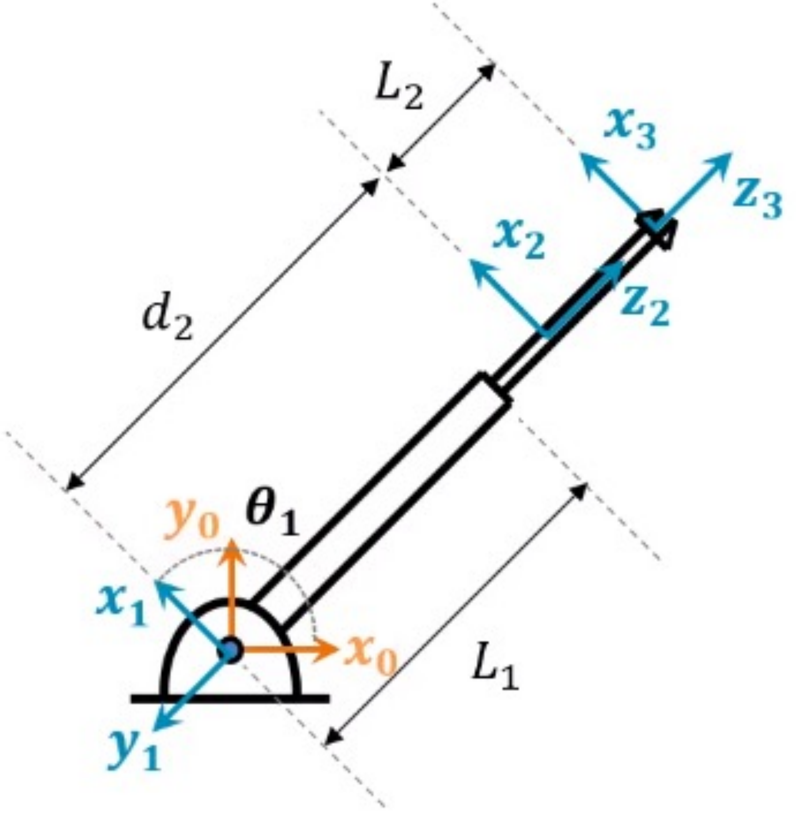
$$\theta_1 = A\tan 2\left(\frac{24.52}{-1.12}\right) - A\tan 2\left(\frac{20\sin(115)}{20\cos 115 + 25}\right) = 45^o$$

or

$$\theta_2 = 245^o$$

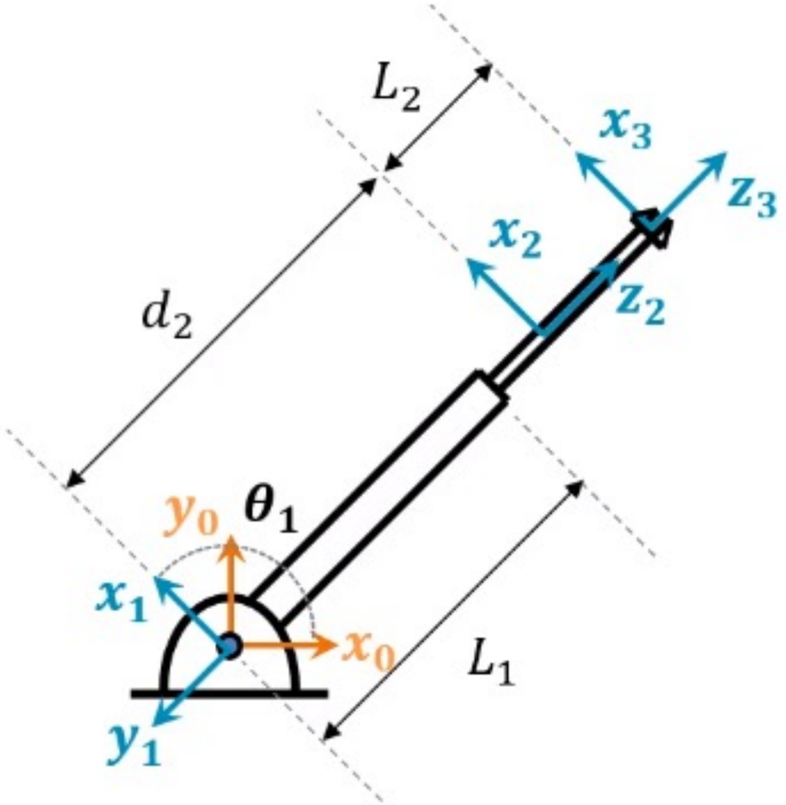
$$\theta_1 = A\tan 2\left(\frac{24.52}{-1.12}\right) - A\tan 2\left(\frac{20\sin(245)}{20\cos 245 + 25}\right) = 140^o$$

Example: Perform the inverse kinematics analysis for the planar 2 DOF RP robotic arm show in the figure?



Forward Kinematics:

$$[{}^{i-1}_{\ \ i}\mathrm{T}] = \begin{bmatrix} c_{\theta i} & -s_{\theta i} & 0 & a_{i-1} \\ s_{\theta i}c_{\alpha i-1} & c_{\theta i}c_{\alpha i-1} & -s_{\alpha i-1} & -d_i s_{\alpha i-1} \\ s_{\theta i}s_{\alpha i-1} & c_{\theta i}s_{\alpha i-1} & c_{\alpha i-1} & d_i c_{\alpha i-1} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$



$$[{}^{0}_{1}\mathrm{T}] = \begin{bmatrix} c_1 & -s_1 & 0 & 0 \\ s_1 & c_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$[{}^{1}_{2}\mathrm{T}] = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & -d_2 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$[{}^{2}_{3}\mathrm{T}] = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & L_2 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

i	a_{i-1}	α_{i-1}	d_i	θ_i
1	0	0	0	θ_1
2	0	90	d_2	0
3	0	0	L_2	0

$$[{}^{0}_{3}\mathrm{T}] = [{}^{0}_{1}\mathrm{T}][{}^{1}_{2}\mathrm{T}][{}^{2}_{3}\mathrm{T}]$$

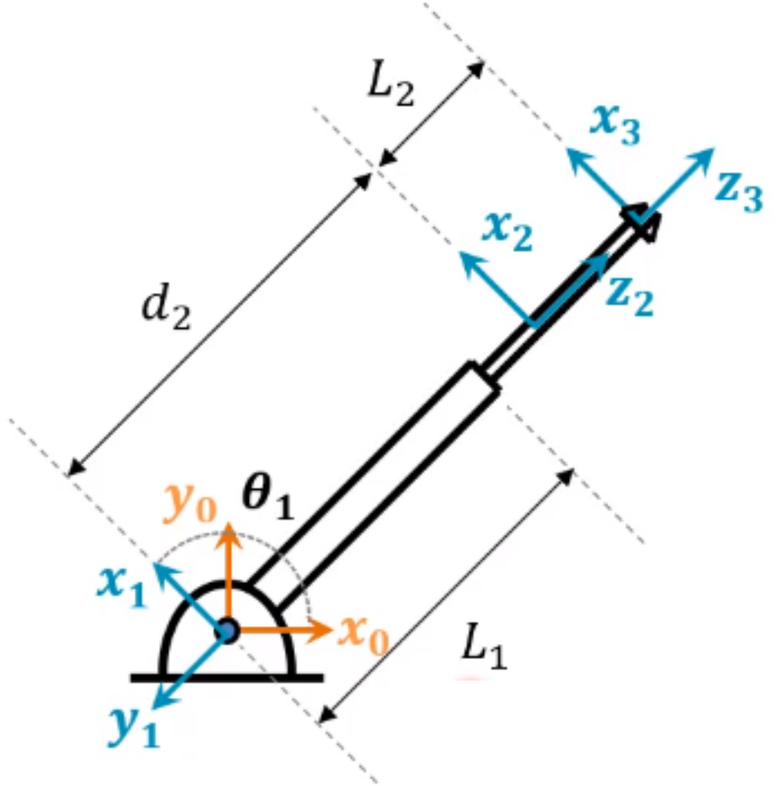
$$[{}^{0}_{3}\mathrm{T}] = \begin{bmatrix} c_1 & -s_1 & 0 & 0 \\ s_1 & c_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & -d_2 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & L_2 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$[{}^{0}_{3}\mathrm{T}] = \begin{bmatrix} c_1 & 0 & s_1 & d_2 s_1 \\ s_1 & 0 & -c_1 & -d_2 c_1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & L_2 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$[{}^{0}_{3}\mathrm{T}] = \begin{bmatrix} c_1 & 0 & s_1 & L_2 s_1 + d_2 s_1 \\ s_1 & 0 & -c_1 & -L_2 c_1 - d_2 c_1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} c_1 & 0 & s_1 & (L_2 + d_2) s_1 \\ s_1 & 0 & -c_1 & -(L_2 + d_2) c_1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Forward Kinematics:

$$[{}^0_3\mathrm{T}] = \begin{bmatrix} c_1 & 0 & s_1 & (L_2 + d_2)s_1 \\ s_1 & 0 & -c_1 & -(L_2 + d_2)c_1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$



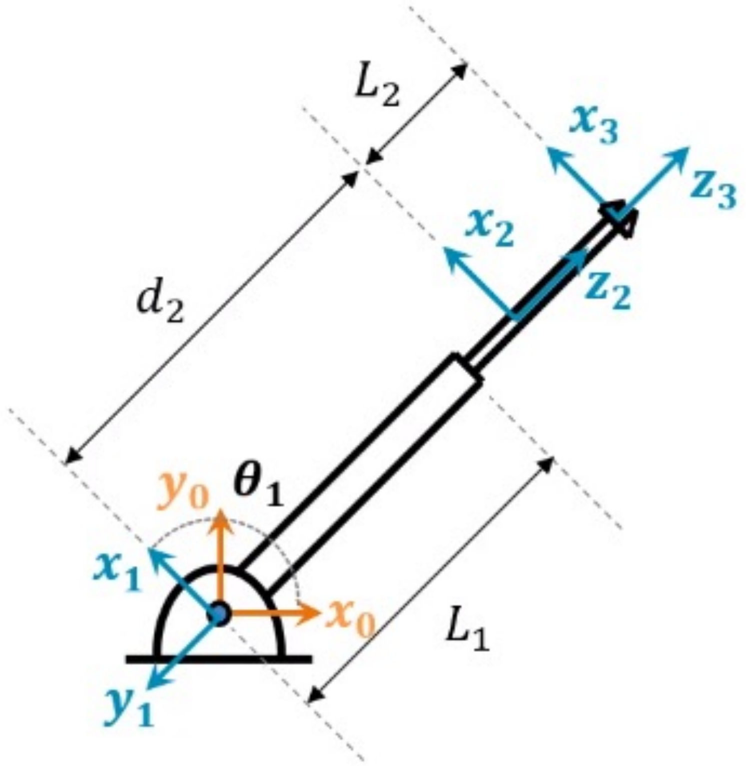
Inverse Kinematics:

$$[{}^0_3\mathrm{T}] = \begin{bmatrix} c_1 & 0 & s_1 & (L_2 + d_2)s_1 \\ s_1 & 0 & -c_1 & -(L_2 + d_2)c_1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Compare with:

$$[{}^0_3\mathrm{T}] = \begin{bmatrix} r_{11} & r_{12} & r_{13} & x \\ r_{21} & r_{22} & r_{23} & y \\ r_{31} & r_{32} & r_{33} & \mathrm{z} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$x = (L_2 + d_2)s_1$$
$$y = -(L_2 + d_2)c_1$$



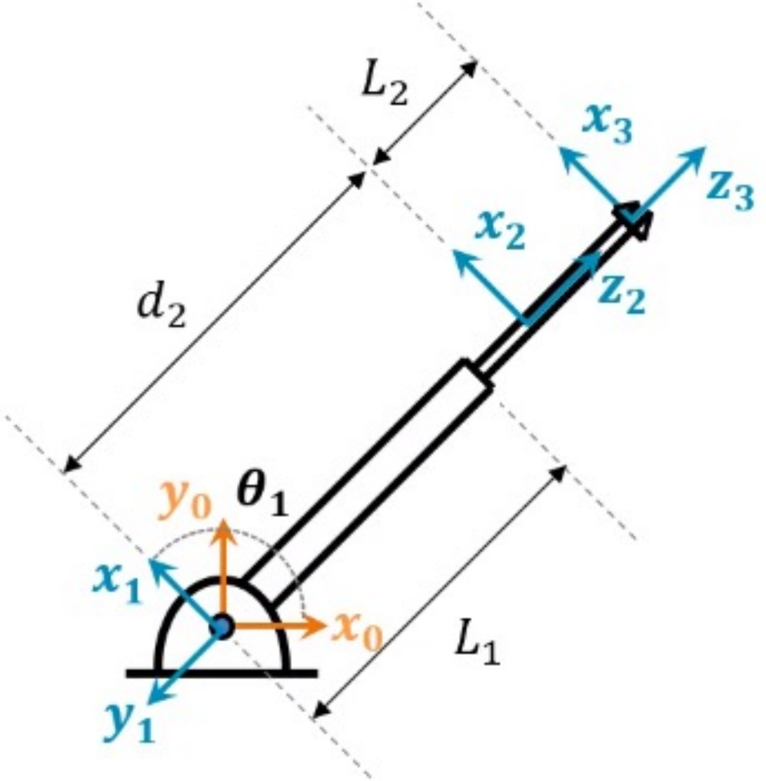
$$x = (L_2 + d_2)s_1 \qquad (1)$$
$$y = -(L_2 + d_2)c_1 \qquad (2)$$

Divide Eqn (1) by Eqn (2):

$$\frac{x}{-y} = \frac{(L_2 + d_2)s_1}{(L_2 + d_2)c_1}$$

$$\frac{x}{-y} = \tan(\theta_1)$$

$$\theta_1 = \text{Atan2}\left(\frac{x}{-y}\right)$$



$$x = (L_2 + d_2)s_1 \tag{1}$$

$$y = -(L_2 + d_2)c_1 \tag{2}$$

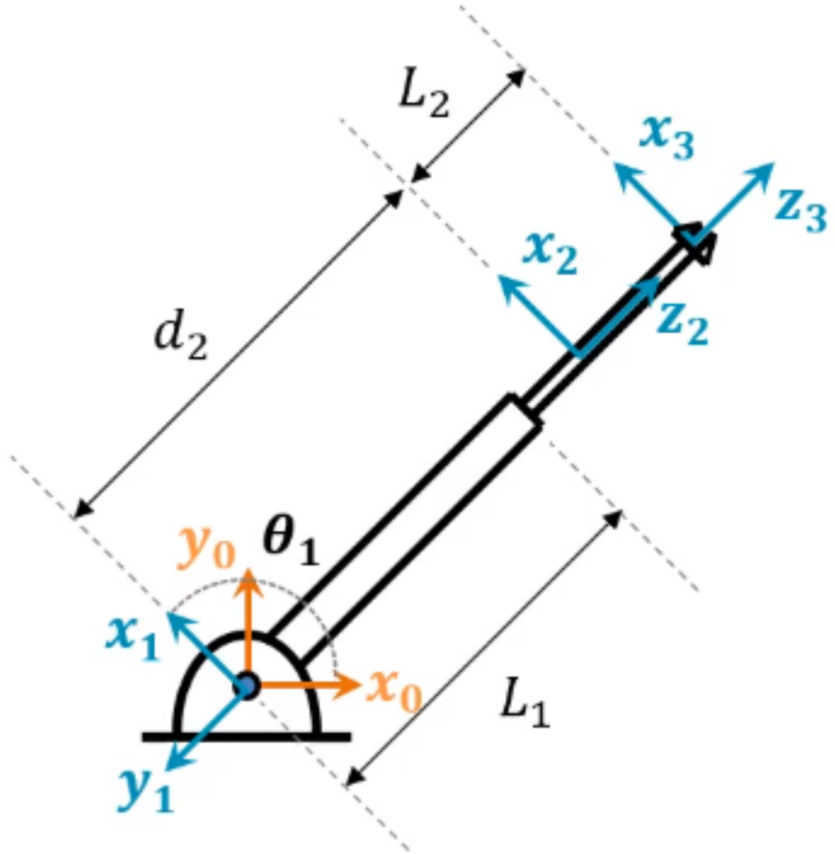
$$\boxed{\theta_1 = \mathrm{Atan2}\left(\frac{x}{-y}\right)}$$

From Eqn (1):

$$x = L_2 s_1 + d_2 s_1$$

$$d_2 s_1 = x - L_2 s_1$$

$$\boxed{d_2 = \frac{x - L_2 s_1}{s_1}}$$



Definition of Controller:

Once a robot has a motion plan, it is time to act. How do you ensure that the robot's execution follows the desired plan? Controlling a large robotic system to do a task often involves solving many smaller control problems, reaching down to the level of controlling individual motors.

Controller
The combination of hardware and software that together program and control a robotic system.
A programmable device that converts an error signal to a current/voltage command, which regulates the behavior of the dynamic system.

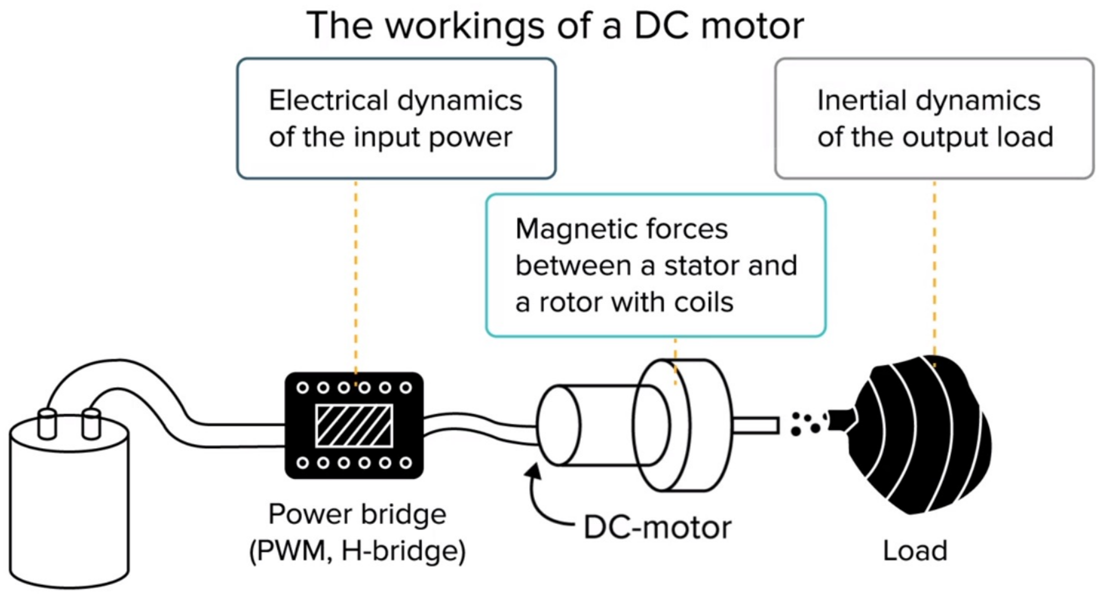
Feedforward (Open-Loop)
Uses the model of the dynamics of the controlled system to make proactive decisions on the commands to send to the robot actuators

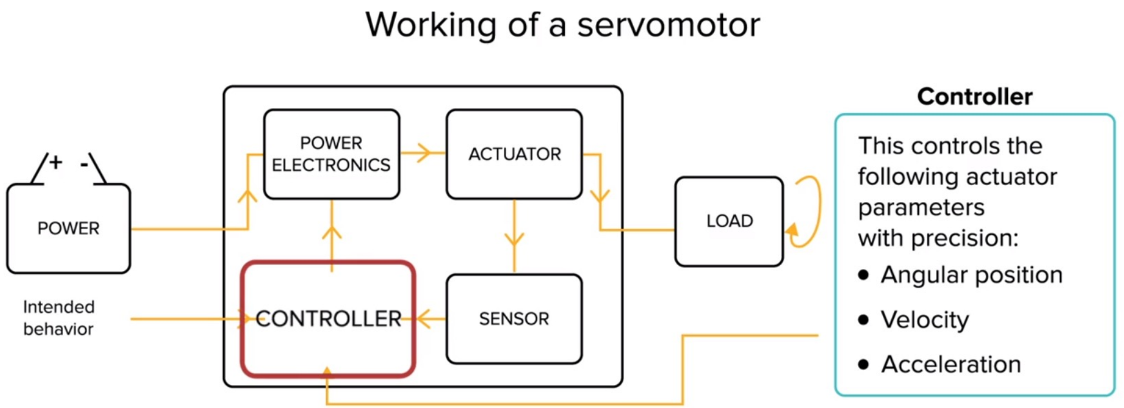
Feedback (Closed-Loop)
Uses the output of the dynamic system as an input to the control decision process. More robust differences between the real and modeled systems than open-loop control.

Control
The study of the process of regulating the output of an actuator or the behavior of a dynamical system

Control and Regulation

- Control is tightly connected with the notion of regulation
- The goal of control is to maintain a physical quantity around the desired value, even when we have imperfect knowledge of the dynamics governing that physical quantity
- For example:
  - Balancing a robot on a single wheel around the vertical with an uncertain mass
  - Maintaining a rotary actuator around a particular angle while fighting against external forces

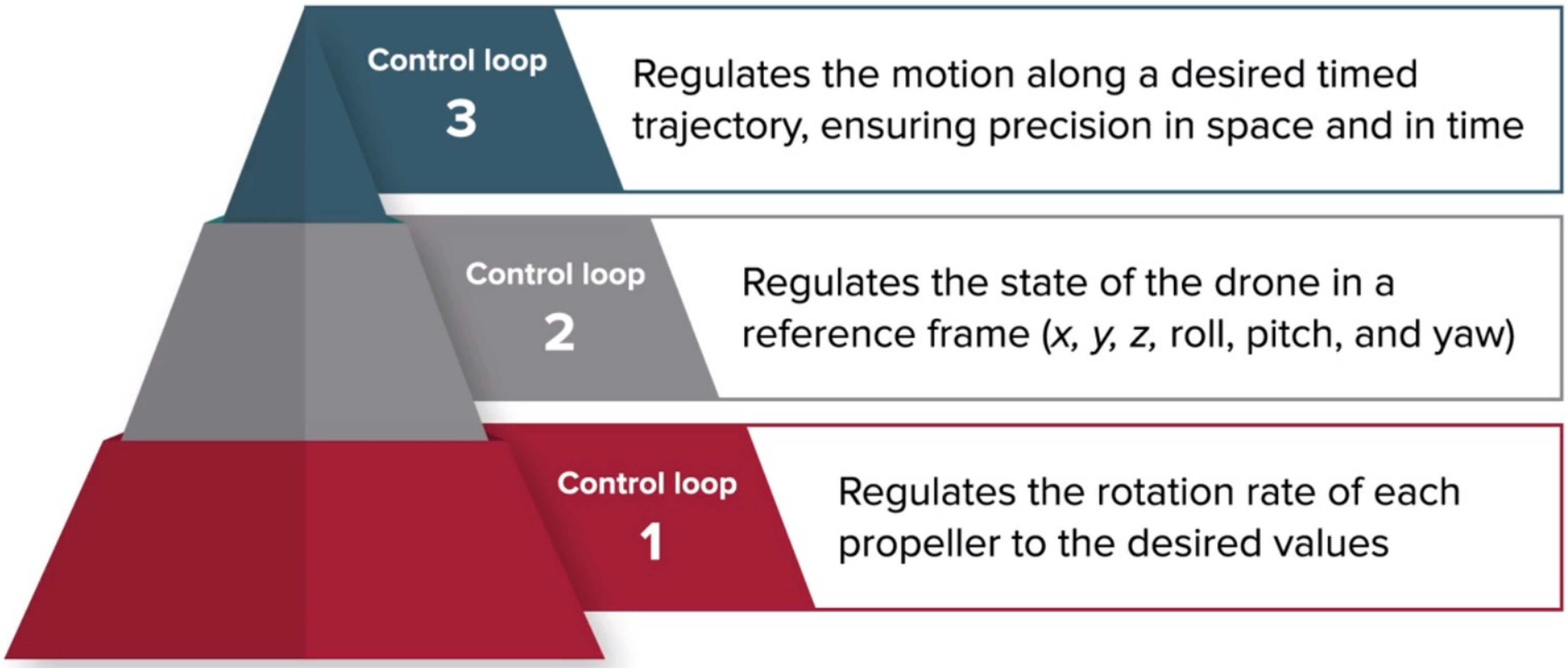




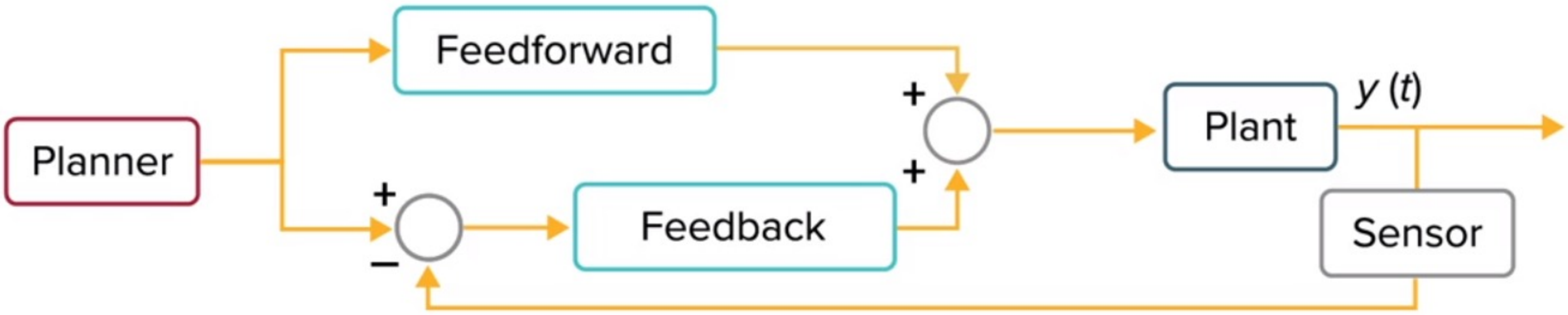
Control
The study of the process of regulating the output of an actuator or the behaviour of a dynamical system

Hierarchy of the control Loop

Example: Controlling a drone



Feedforward vs. Feedback



Feedforward or open-loop control

- Uses the model of the dynamics of the controlled system to make proactive decisions on the commands to send to the robot actuators
- Control focuses on fast decision rates possibly with simpler models

Feedback or closed-loop control

- Uses the output of the dynamic system as an input to the control decision process while overcoming the differences between the real and modeled systems
- The output signal is usually captured by a sensor that facilitates the process of regulating the output to a desired value

Feedforward vs. Feedback

Feedforward or open loop control

- Uses the model of the dynamics of the controlled system to make proactive decisions on the commands to send to the robot actuators
- Control focuses on fast decision rates possibly with simpler models
- It is proactive and can anticipate the needs of the system

VS.

Feedback or closed loop control

- Uses the output of the dynamic system as an input to the control decision process while overcoming the differences between the real and modelled systems
- The output signal is usually captured by a sensor that facilitates the process of regulating the output to a desired value
- It is reactive and this leads to delays due to latency in sensing and processing

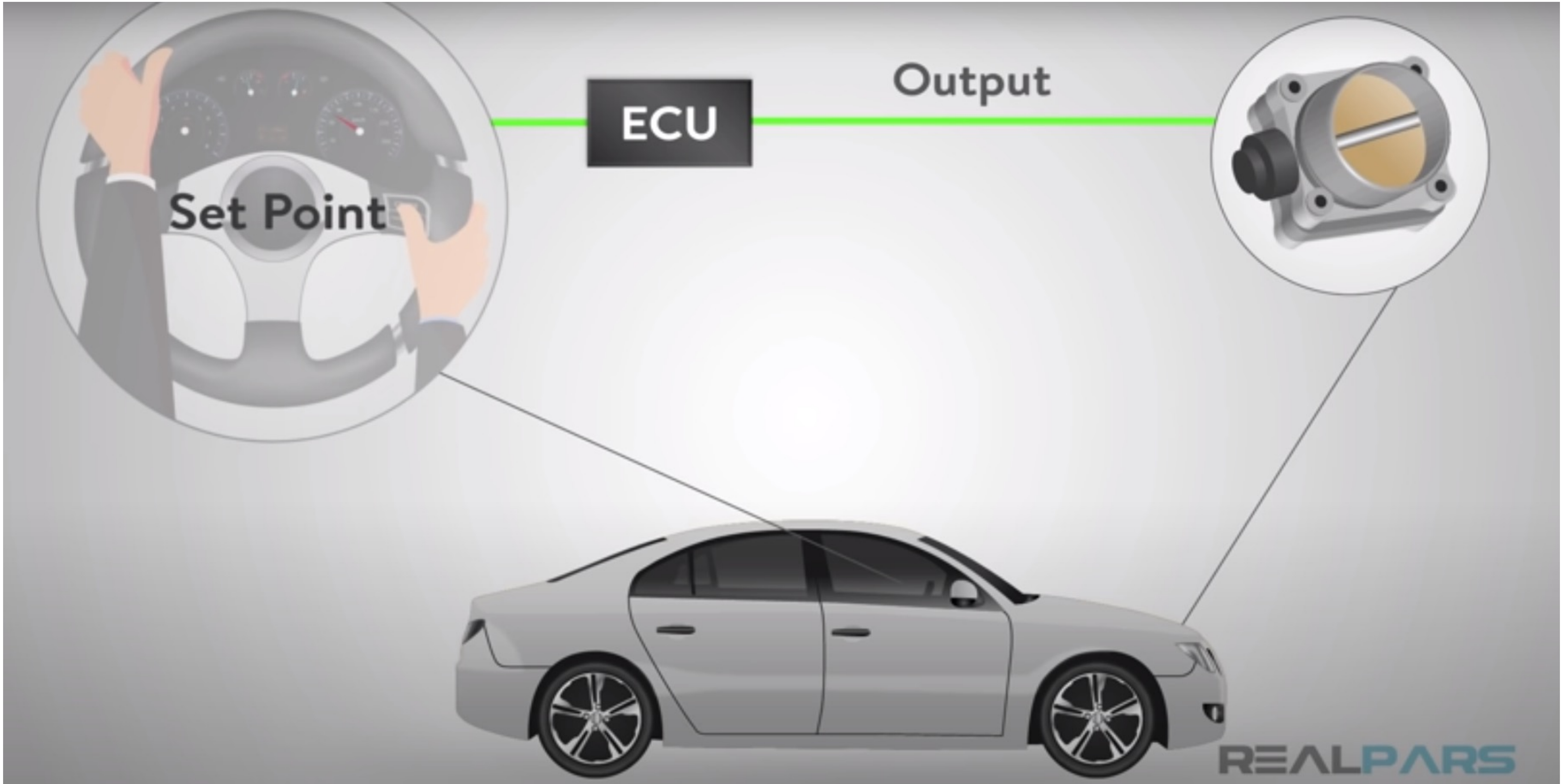
In class assignment-Submit in ilearn

In this group assignment:
1.Select a robotic system from a news article, popular culture (i.e., movie, book, comic, etc.), or your imagination that moves autonomously. Your selected robot should enable you to explore topics in control.

2. With your selected robot in mind, please write a response that includes the following points:

- What is one physical quality that needs to be regulated during the task? (e.g., position, velocity, force, etc.)
- If you were to use feedback, what sensor could the robot use to measure the physical quality? (e.g., encoder, IMU, etc.)
- Explain any hierarchies of control within this task.
- Briefly describe a control task your robot performs. Considering the task, answer the following questions:
- What would be the failure case of your system if the desired value is not sufficiently regulated?

PID controller



Types of Controllers

Proportional Integral Derivative (PID)

- Commonly used continuous controller
- Design or selection of the parameters are not trivial

Linear Quadratic Regulator (LQR)

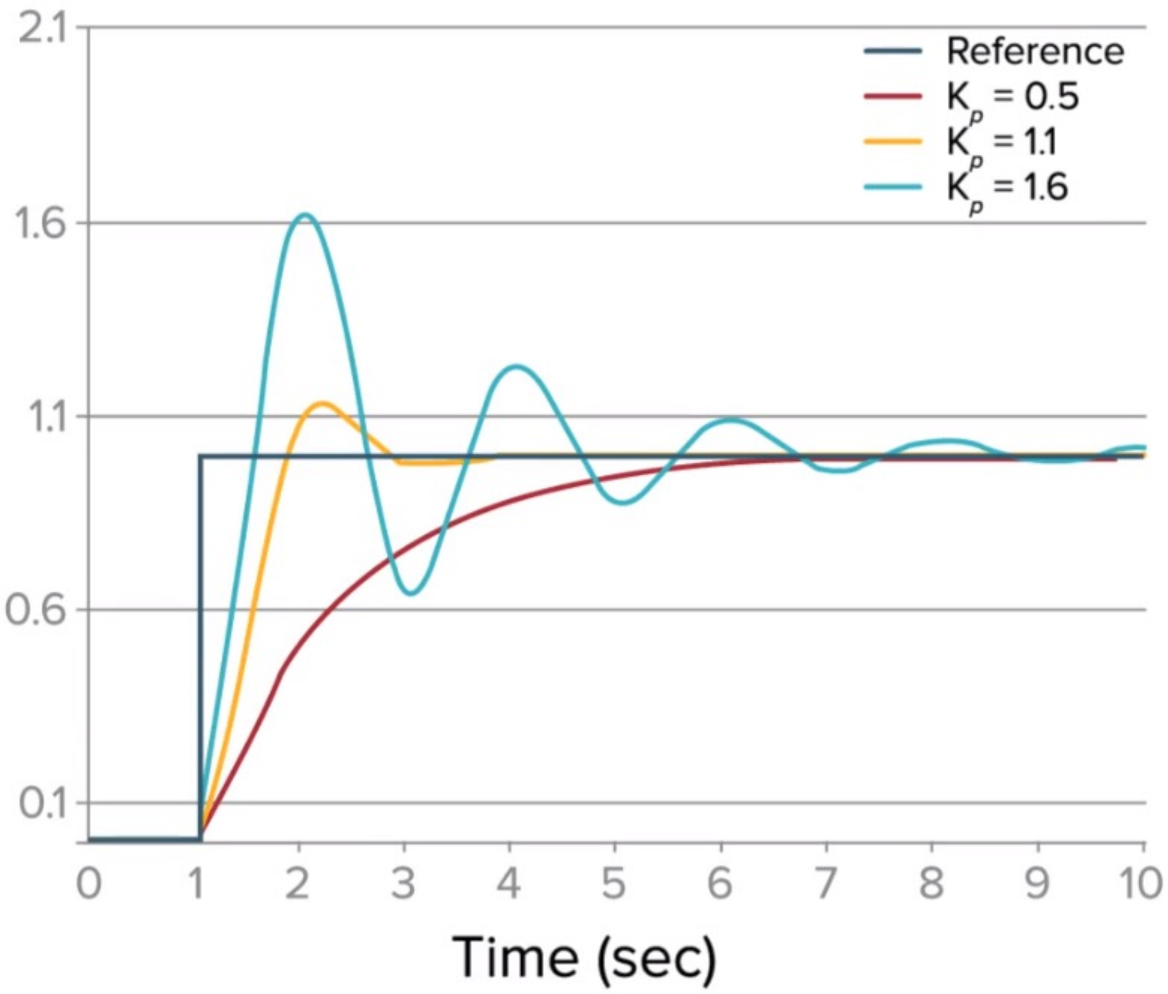
- Parameters can be selected through optimization

Model Predictive Control (MPC)

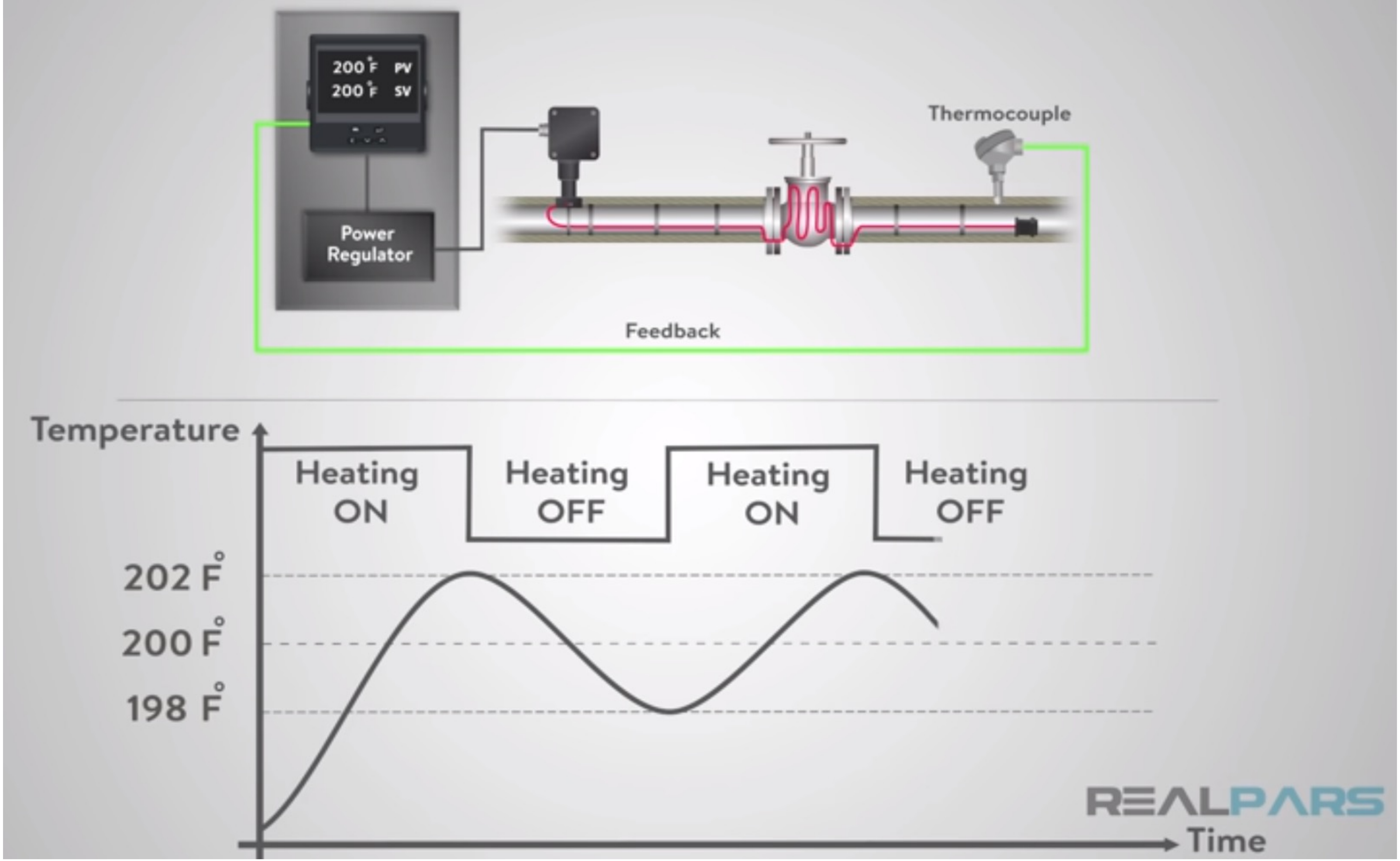
- Suitable for systems subject to non-linear dynamics and to constraints, such as actuator limits or under actuation

Proportional Controller

Proportionality constant



Source: TimmmyK. (2014). *Graph showing output on PID controller* [Illustration]. https://commons.wikimedia.org/wiki/File:PID_varyingP.jpg


200°F PV
200°F SV
Power Regulator
Thermocouple
Feedback
Temperature
Heating ON
Heating OFF
Heating ON
Heating OFF
202 F°
200 F°
198 F°
Time

Limitations of Proportional Controller

Steady-state error

- As the controller is gradual, static forces, such as friction, make it difficult or impossible to reach the target because:
  - When approaching the target, the magnitude of the control signal decreases
  - The moment the magnitude goes below the parasitic static forces, the system stops moving without reaching the target

Overshooting

- Increasing K_p makes the controller faster but also:
  - Increases overshooting
  - Encourages undesired oscillations around the target

Linear-Quadratic Regulators

Linear time-invariant (LTI) system

In a linear system, there is a proportional relationship between the input and the output

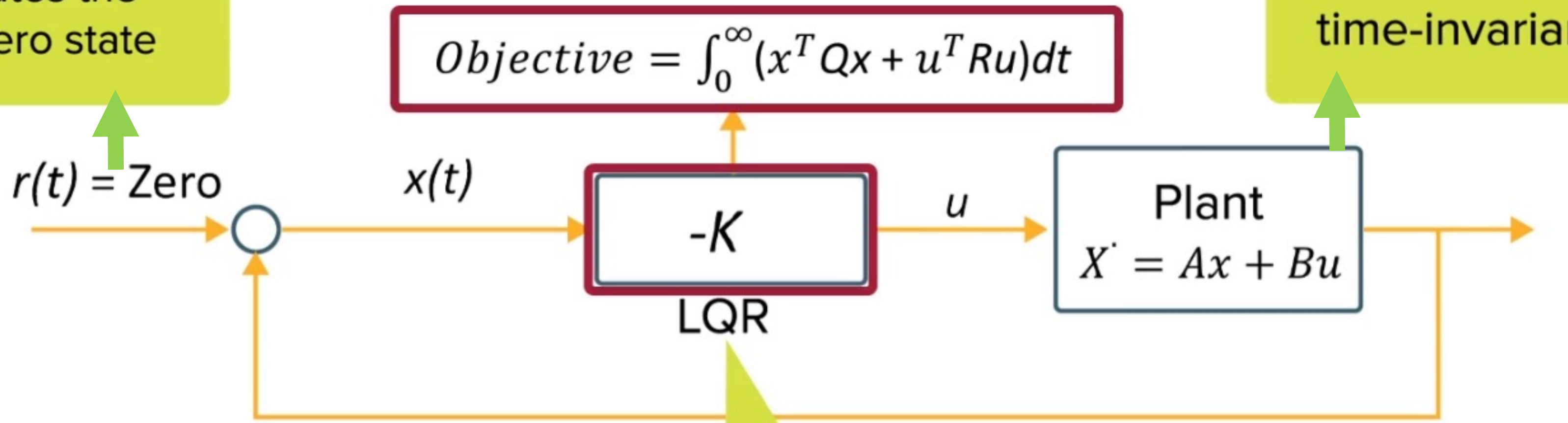
In a time-invariant system, that proportionality remains constant in time

- Linear systems are theoretical constructs
- They help explain the structure of control problems
- They are relevant to real non-linear systems
- The linear control law involves proportionality of the position and velocity, analogous to the proportional and derivative controllers in a proportional-integral-derivative (PID) controller

The problem searches for the trajectory of future states and control values that will optimize the objective

The canonical linear quadratic control setting regulates the system around the zero state

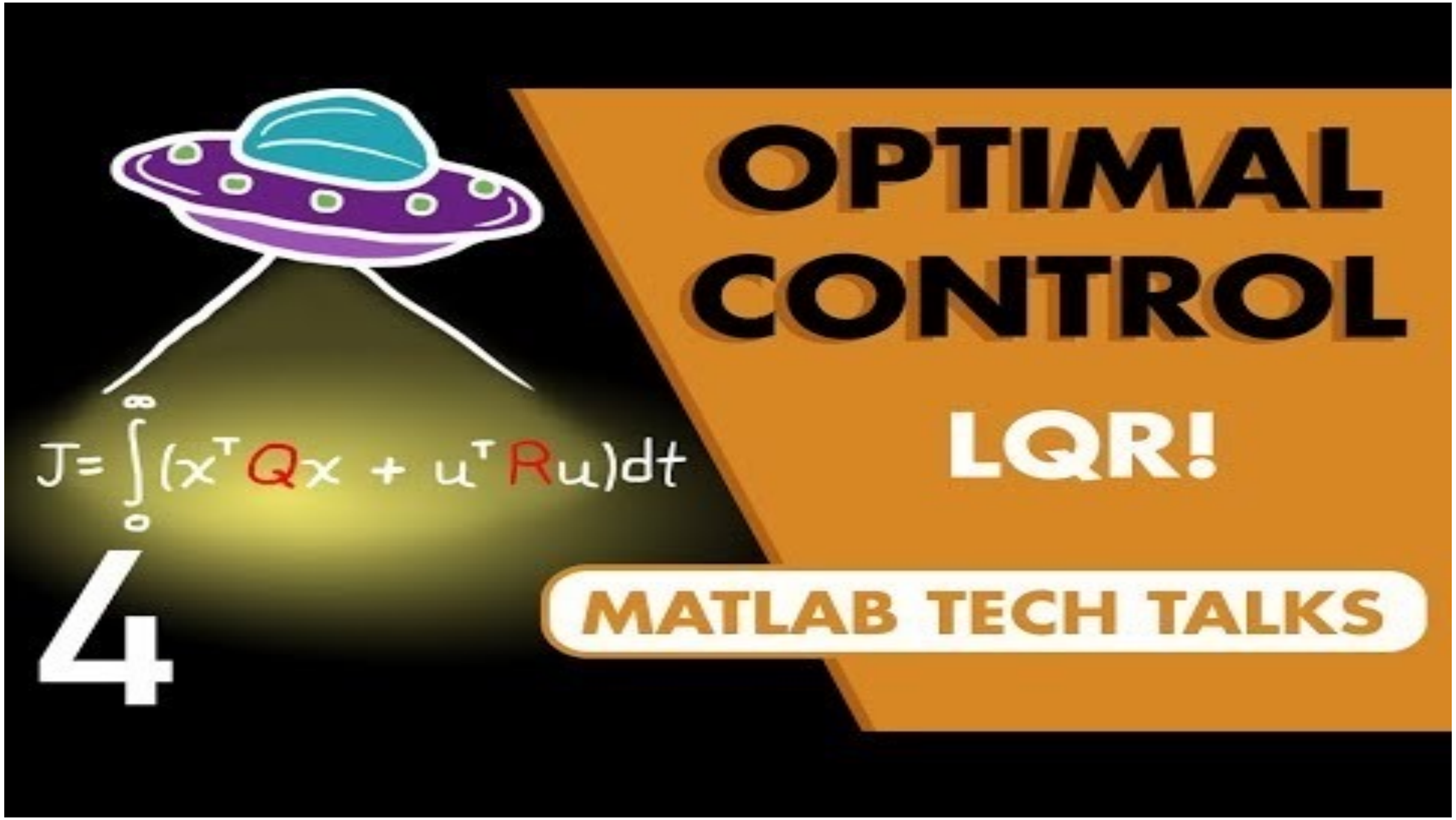
The trajectories of states and controls are constrained by the time-invariant linear dynamics



- The objective function is a quadratic function of both the state signal (matrix Q) and the control signal (matrix R)
- Think of it as a cost function that penalizes:
  - The magnitude of the state trajectory
  - The magnitude of the control trajectory

Watch at home

What Is Linear Quadratic Regulator (LQR) Optimal Control?



https://www.youtube.com/watch?v=E_RDCFOlJx4&ab_channel=MATLAB

End of Week 6