MTRN4110 Robot Design
Week 3 – Kinematics

Liao "Leo" Wu, Lecturer

School of Mechanical and Manufacturing Engineering

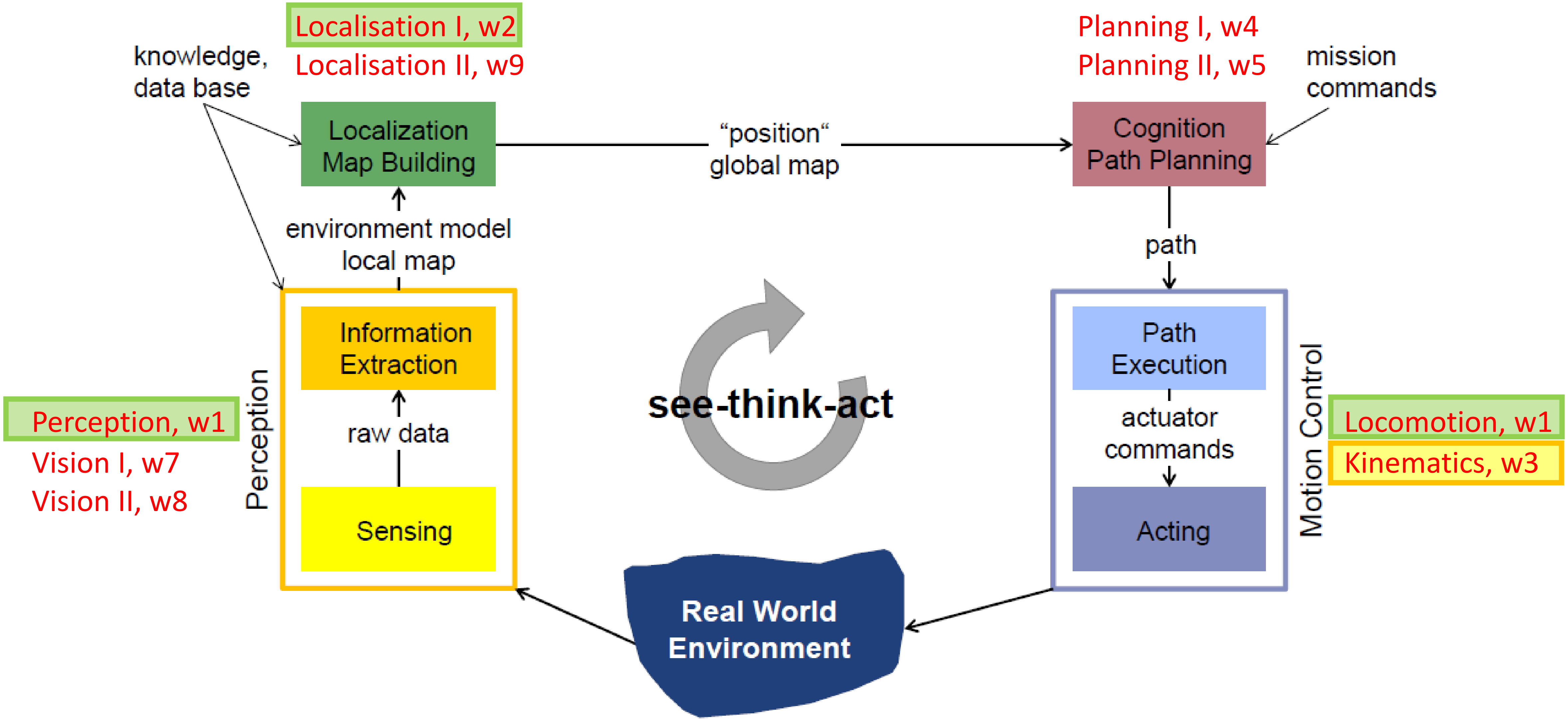
University of New South Wales, Sydney, Australia

https://sites.google.com/site/wuliaothu/

Today's agenda

- Kinematics for mobile robots
- Manoeuvrability - Revisit
- Trajectory generation
- Kinematic control

The See-Think-Act cycle



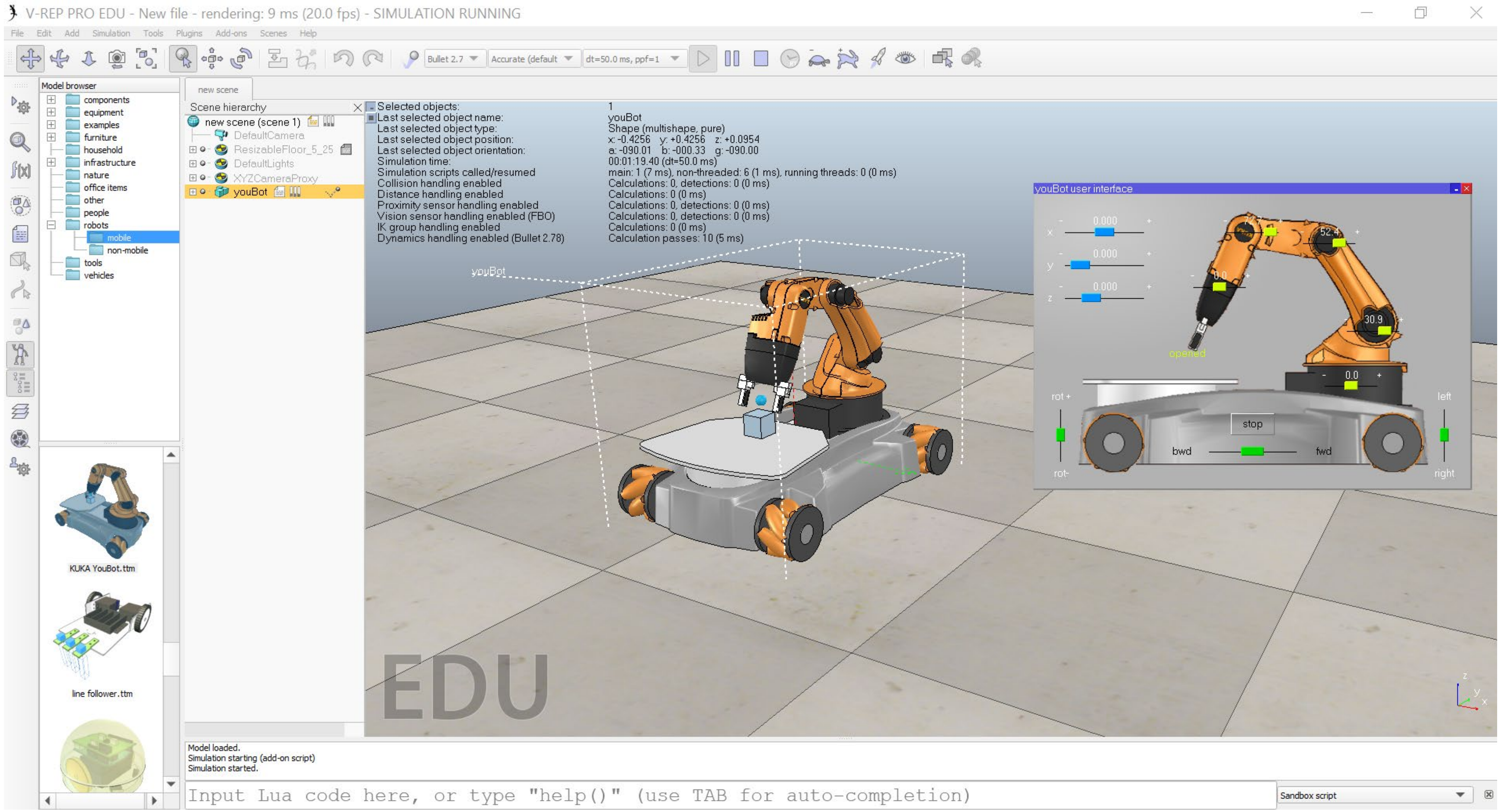
R. Siegwart, I. R. Nourbakhsh, D. Scaramuzza. Introduction to autonomous mobile robots. The MIT Press. Second edition. 2011.

Kinematics for Mobile Robots

What is Kinematics?

- A branch of mathematics that studies the motion of a body, or a system of bodies
- Concerned with positions (or angles) and velocities (translational and angular)
- Not concerned with forces or moments -> Statics and Dynamics
- Two kinematic problems are usually considered in robotics
  - Forward kinematics
    - Given the joint angles, where is the robot's tool tip?
  - Inverse kinematics
    - Given the pose of the robot's tool tip, what joint angles are required?

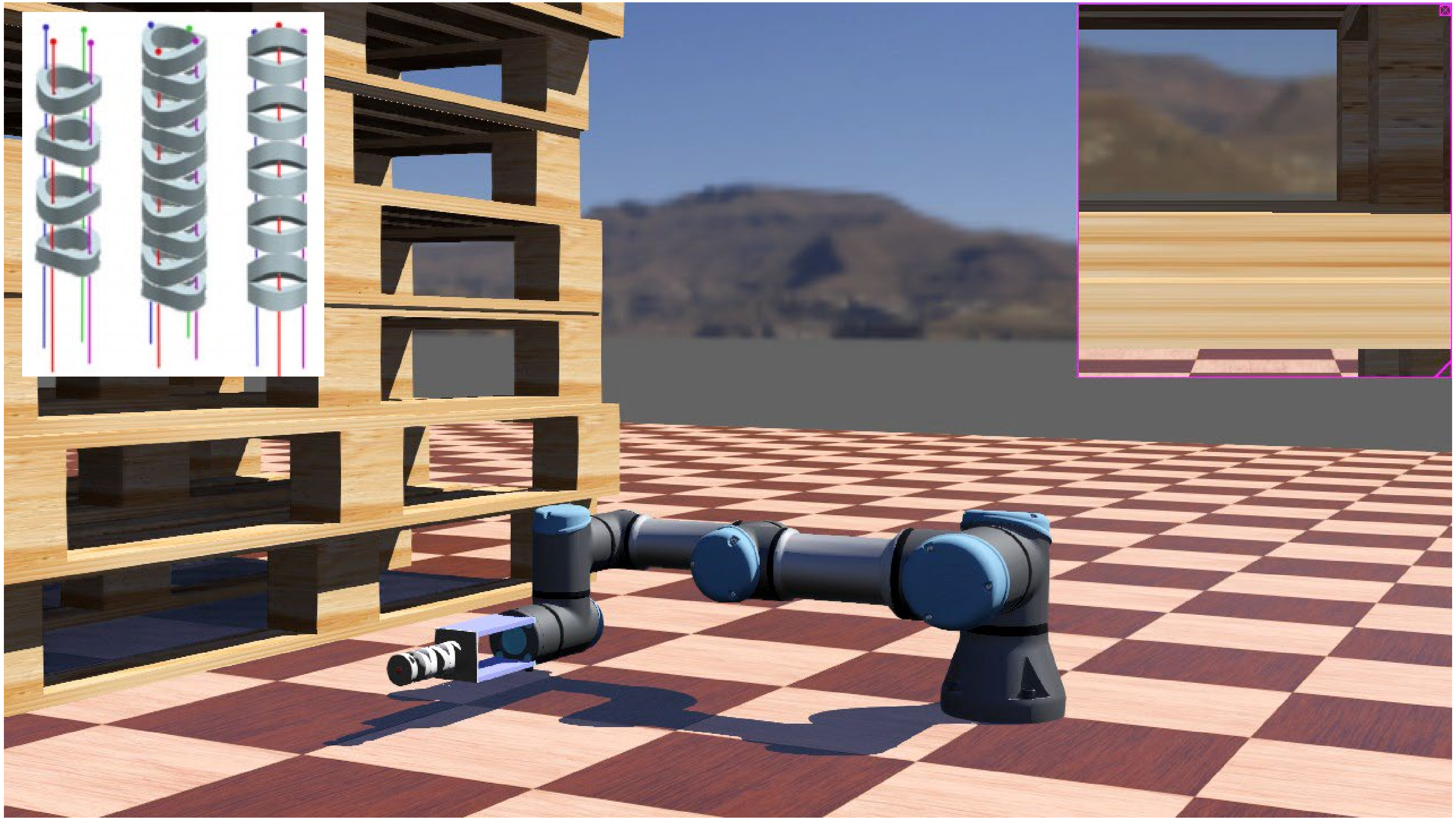
Kinematics for manipulators



https://www.coppeliarobotics.com/

Which kinematics is needed here?

https://www.sli.do/
#4110



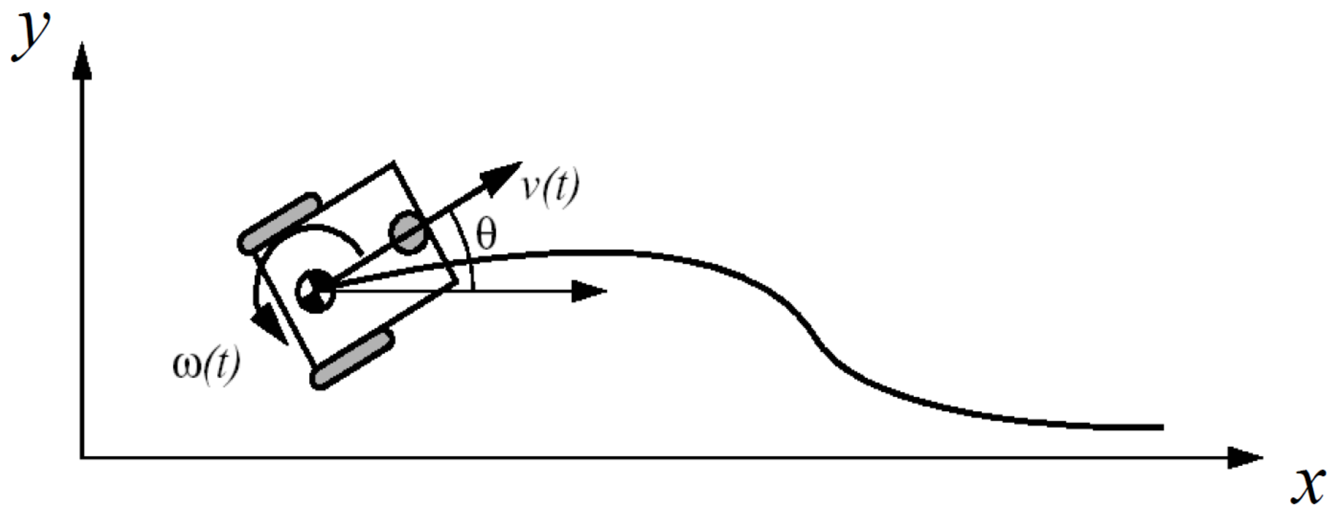
Kevin Li. Development of a Snake-like Manipulator for Minimally Invasive Surgery. 2020 slido

Which kinematics is needed in this simulation to implement the horizontal, vertical, and diagonal scanning for the snake-like robot?

ⓘ Start presenting to display the poll results on this slide.

Kinematics for mobile robots?

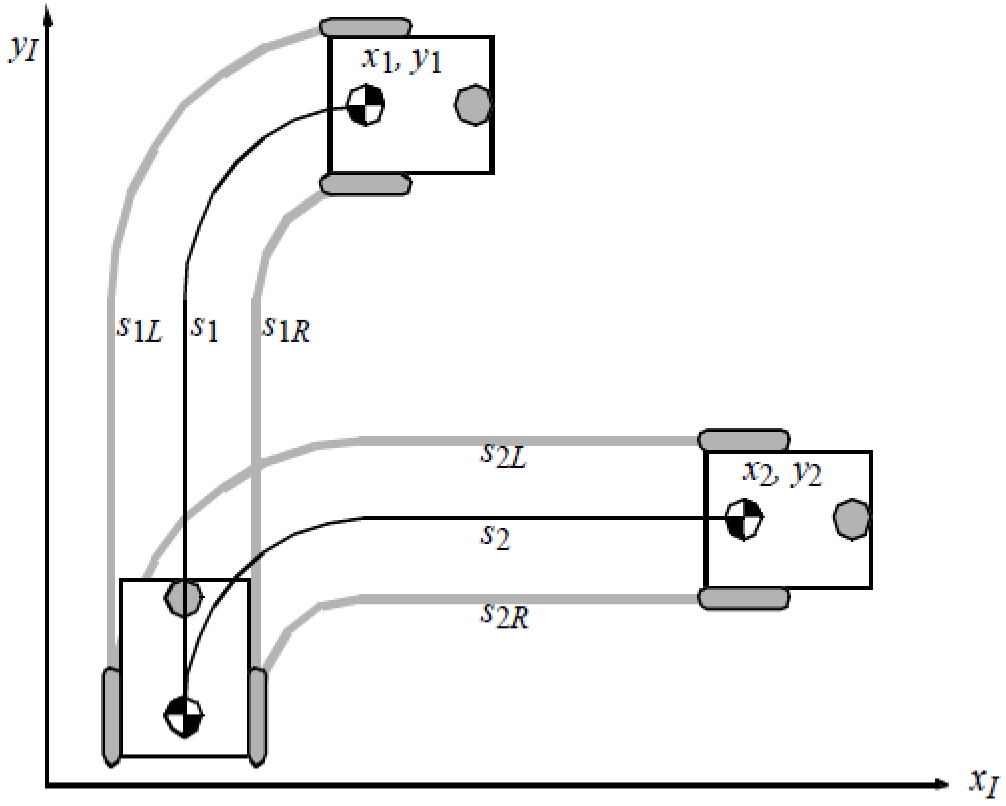
- For a differential-drive robot, is it OK to define the forward kinematics similarly to the one for manipulators as:
  - Given the travelled distance (joint) of the left and right wheels, find the position (end-effector) of the robot?



Kinematics for mobile robots?

- For a differential-drive robot, is it OK to define the forward kinematics similarly to the one for manipulators as:
  - Given the travelled distance (joint) of the left and right wheels, find the position (end-effector) of the robot?

$$s_1 = s_2, s_{1R} = s_{2R}, s_{1L} = s_{2L}$$
$$x_1 \neq x_2, y_1 \neq y_2$$



Holonomic system vs. nonholonomic system

- Holonomic system
  - All kinematic constraints can be expressed as an explicit function of position variables (and time) only.

$$f(q_1, q_2, \dots, q_n, t) = 0$$

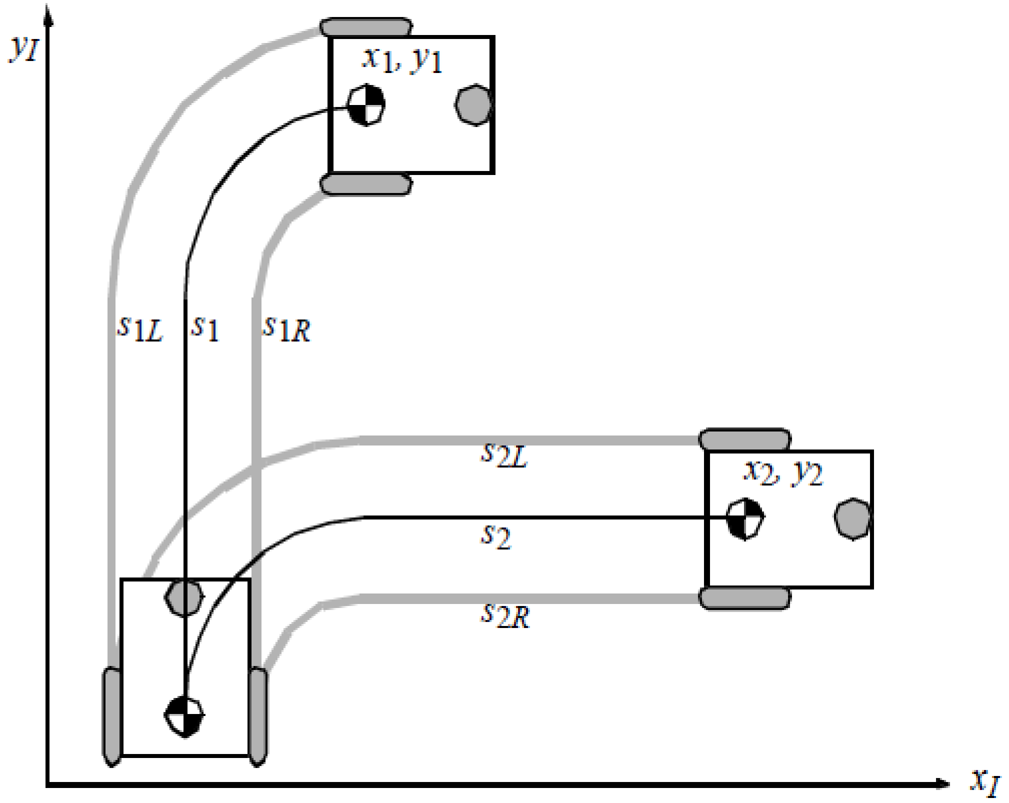
- Nonholonomic system
  - One or more kinematic constraints cannot be expressed as an explicit function of position variables (and time) only.
  - Has to involve velocity variables

$$f(q_1, q_2, \dots, q_n, \dot{q}_1, \dot{q}_2, \dots, \dot{q}_n, t) = 0$$

  - Cannot be integrated to provide a constraint in terms of position variables (and time) only.

$$s_1 = s_2, s_{1R} = s_{2R}, s_{1L} = s_{2L}$$

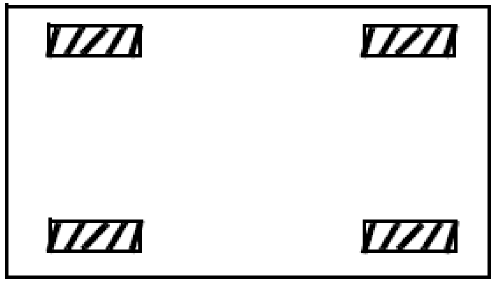
$$x_1 \neq x_2, y_1 \neq y_2$$



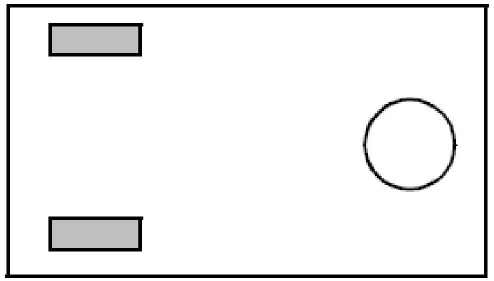
R. Siegwart, I. R. Nourbakhsh, D. Scaramuzza. Introduction to autonomous mobile robots. The MIT Press. Second edition. 2011.
https://en.wikipedia.org/wiki/Holonomic_constraints

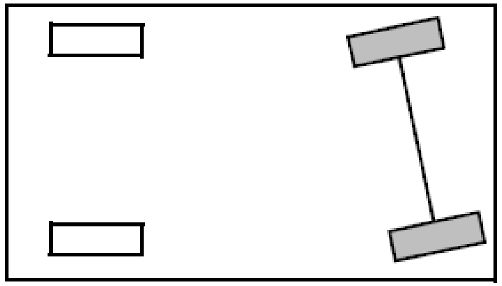
Mobile robots

- Some are holonomic systems
  - E.g., omnidirectional robots



- Some are nonholonomic systems
  - E.g., differential-drive robots, Ackermann-steering robots



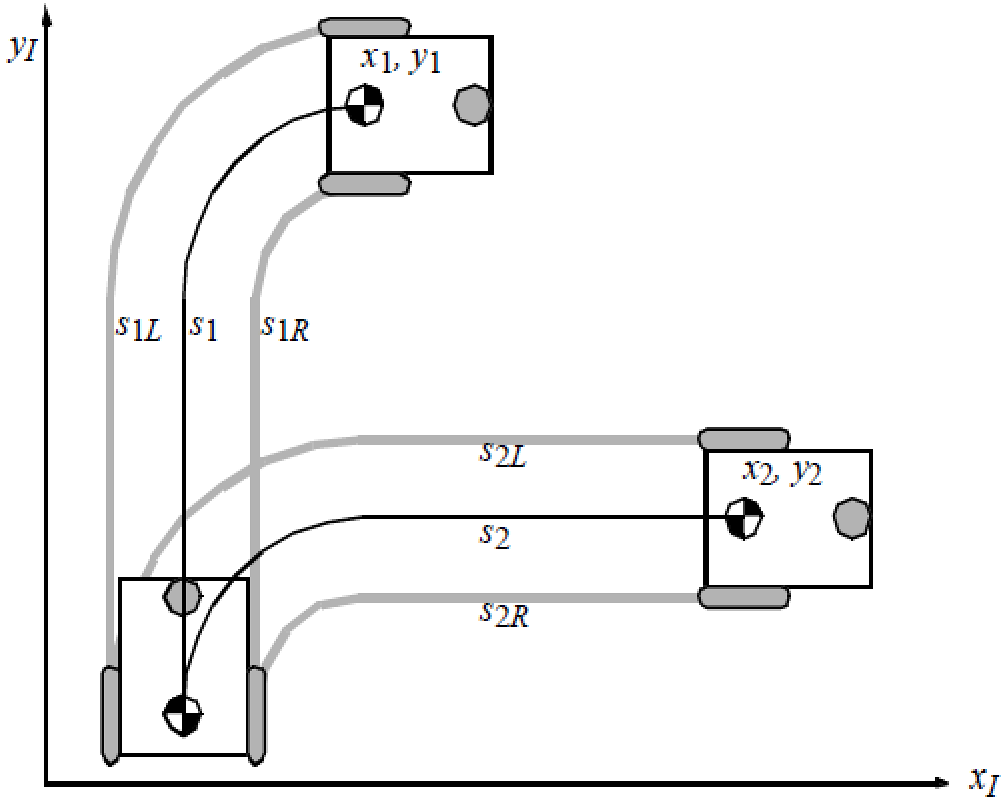


R. Siegwart, I. R. Nourbakhsh, D. Scaramuzza. Introduction to autonomous mobile robots. The MIT Press. Second edition. 2011.

Kinematics for mobile robots?

- For a differential-drive robot, can the forward kinematics be defined as:
  - Given the travelled distance (joint) of the left and right wheels, find the position (end-effector) of the robot?

$$s_1 = s_2, s_{1R} = s_{2R}, s_{1L} = s_{2L}$$
$$x_1 \neq x_2, y_1 \neq y_2$$

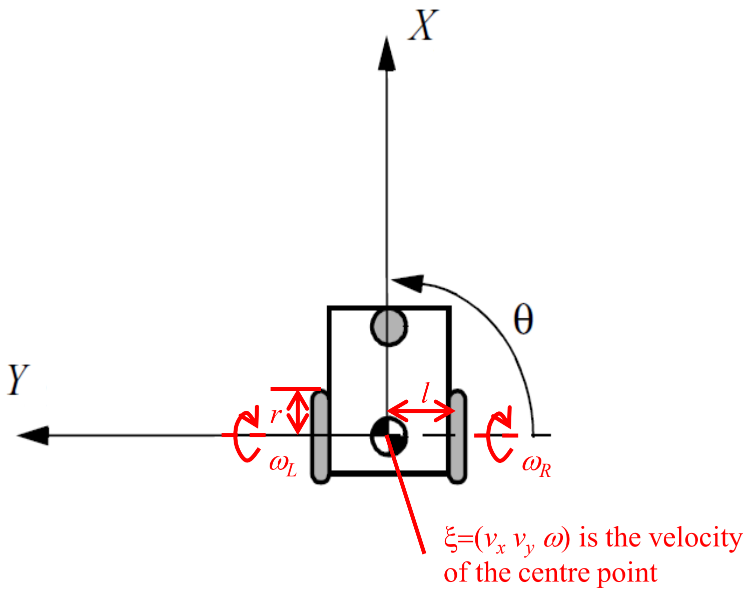


- Answer: not for nonholonomic mobile robots

Kinematics for nonholonomic mobile robots – Differential kinematics

- Forward differential (velocity) kinematics
  - Given the velocities of the actuators, what is the velocity of the robot?

Suppose both wheels have a diameter of *40mm* and spaced at *100mm*. The left wheel spins at *30deg/s*, and the right at *60deg/s*. Specify v_x , v_y , and ω. ($\pi = 3.14$) - Lecture 1



- Inverse differential (velocity) kinematics
  - Given the velocity of the robot, what are the velocities of the actuators?

Suppose both wheels have a diameter of *40mm* and spaced at *100mm*. The robot moves at $v_x = 10\pi$ *mm/s*, $v_y = 0$ *mm/s*, and $\omega = \pi/15$ *rad/s*. What are the required speeds of the left and right wheels? ($\pi = 3.14$) - Lecture 1

$$\xi = {}^{L}\xi + {}^{R}\xi = \begin{bmatrix} \frac{r \cdot \omega_L}{2} + \frac{r \cdot \omega_R}{2} \\ 0 \\ -\frac{r \cdot \omega_L}{2l} + \frac{r \cdot \omega_R}{2l} \end{bmatrix}$$

Is this in conflict with odometry? (Lecture 2 - Localisation I)

Increment

Current pose

Next pose

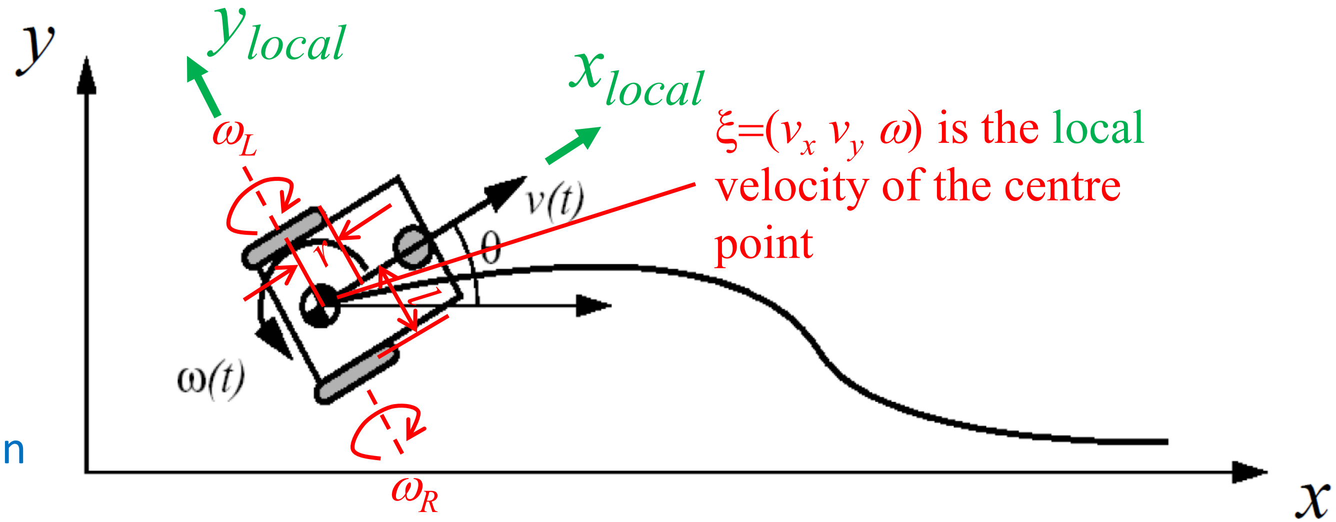
$$p(t+\Delta t) \approx p(t) + \begin{bmatrix} \Delta s \cdot \cos(\theta + \frac{\Delta\theta}{2}) \\ \Delta s \cdot \sin(\theta + \frac{\Delta\theta}{2}) \\ \Delta\theta \end{bmatrix}$$

$\Delta s \equiv \frac{r \cdot \Delta\theta_R}{2} + \frac{r \cdot \Delta\theta_L}{2}$ — Incremental linear motion

$\Delta\theta \equiv \frac{r \cdot \Delta\theta_R}{2l} - \frac{r \cdot \Delta\theta_L}{2l}$ — Incremental rotation

$\Delta\theta_R = \omega_R \cdot \Delta t$ — Incremental rotation of right wheel

$\Delta\theta_L = \omega_L \cdot \Delta t$ — Incremental rotation of left wheel



Q: Suppose a differential-drive robot is running at a constant speed. The wheels have a diameter of 40mm and spaced at 100mm. The encoders of two wheels are read twice. The differences between the two readings are 30deg and 60deg for the left and right wheels, respectively. Assume at the first reading, the robot's pos is (0mm, 0mm, 0deg). What is the robot's pose at the second reading? ($\pi = 3.14$)

R. Siegwart, I. R. Nourbakhsh, D. Scaramuzza. Introduction to autonomous mobile robots. The MIT Press. Second edition. 2011.

Is this in conflict with odometry? (Lecture 2 - Localisation I)

Rotation matrix from local frame to global frame

Current pose

Local velocity

Next pose

Sample interval

$$p(t+\Delta t) \approx p(t) + R \cdot \xi \cdot \Delta t$$

$$= \begin{bmatrix} x \\ y \\ \theta \end{bmatrix} + R \cdot \begin{bmatrix} \frac{r \cdot \omega_L \cdot \Delta t}{2} + \frac{r \cdot \omega_R \cdot \Delta t}{2} \\ 0 \\ -\frac{r \cdot \omega_L \cdot \Delta t}{2l} + \frac{r \cdot \omega_R \cdot \Delta t}{2l} \end{bmatrix}$$

$\Delta\theta_L = \omega_L \cdot \Delta t$

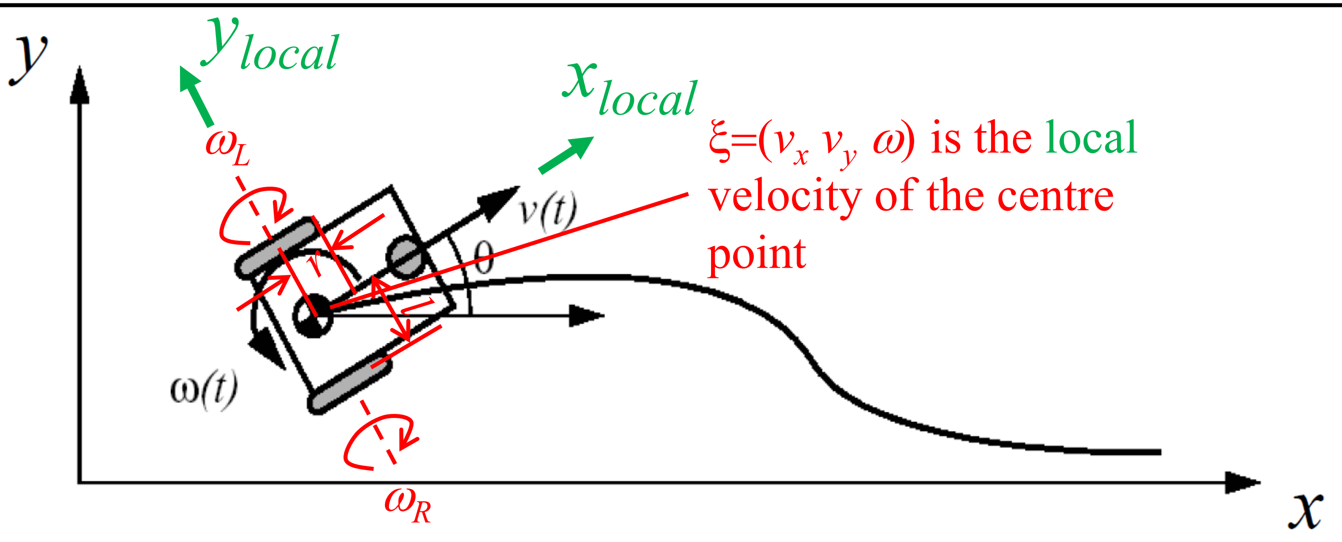
$\Delta\theta_R = \omega_R \cdot \Delta t$

$$= \begin{bmatrix} x \\ y \\ \theta \end{bmatrix} + \begin{bmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \frac{r \cdot \Delta\theta_L}{2} + \frac{r \cdot \Delta\theta_R}{2} \\ 0 \\ -\frac{r \cdot \Delta\theta_L}{2l} + \frac{r \cdot \Delta\theta_R}{2l} \end{bmatrix}$$

$$\Delta s \equiv \frac{r \cdot \Delta\theta_L}{2} + \frac{r \cdot \Delta\theta_R}{2}$$

$$\Delta\theta \equiv -\frac{r \cdot \Delta\theta_L}{2l} + \frac{r \cdot \Delta\theta_R}{2l}$$

$$= \begin{bmatrix} x \\ y \\ \theta \end{bmatrix} + \begin{bmatrix} \Delta s \cdot \cos(\theta) \\ \Delta s \cdot \sin(\theta) \\ \Delta\theta \end{bmatrix} \approx \begin{bmatrix} x \\ y \\ \theta \end{bmatrix} + \begin{bmatrix} \Delta s \cdot \cos(\theta + \frac{\Delta\theta}{2}) \\ \Delta s \cdot \sin(\theta + \frac{\Delta\theta}{2}) \\ \Delta\theta \end{bmatrix}$$



$\xi = (v_x\ v_y\ \omega)$ is the local velocity of the centre point

$$\xi = \begin{bmatrix} v_x \\ v_y \\ \omega \end{bmatrix} = \begin{bmatrix} \frac{r \cdot \omega_L}{2} + \frac{r \cdot \omega_R}{2} \\ 0 \\ -\frac{r \cdot \omega_L}{2l} + \frac{r \cdot \omega_R}{2l} \end{bmatrix}$$

Manoeuvrability - Revisit

Mobile robot manoeuvrability

- The *manoeuvrability* of a mobile robot is the combination
  - Of the mobility available
  - Plus additional freedom contributed by the steering

- *Mobility* - Ability to directly move in the environment

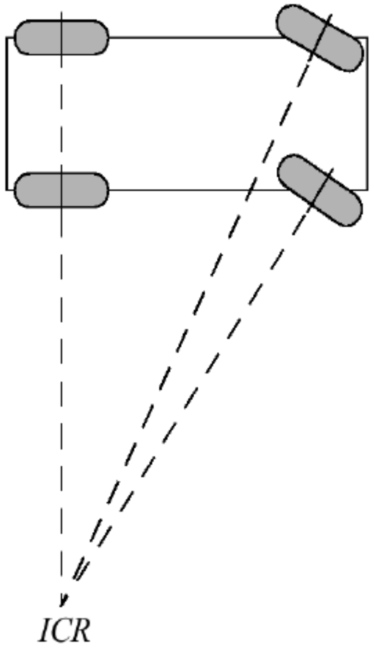
- *Steerability* - Ability to further manipulate its position, over time, by steering steerable wheels

- They can be denoted by
  - Degree of *mobility* δ_m
  - Degree of *steerability* δ_s
  - Degree of *manoeuvrability* $\delta_M = \delta_m + \delta_s$

Degree of mobility

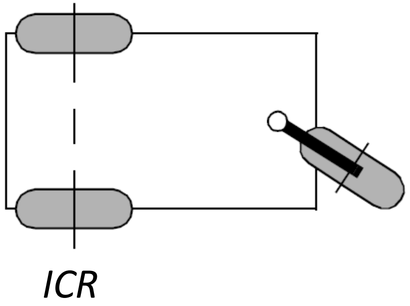
- Degrees of freedom to directly move in the environment through changes in wheel velocity.
- $\delta_m = 3 - n$ (n is the number of constraints on the position of *Instantaneous Centre of Rotation (ICR) without considering steering)*
  - *Point (2 constraints); Line (1 constraint); Plane (0 constraint)*

Ackerman-steering



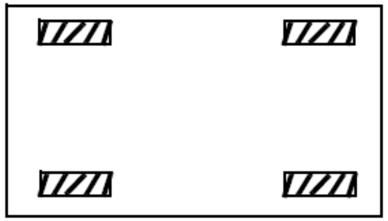
- ICR is constrained to be a fixed point
- $n = 2$
- $\delta_m = 1$

Differential-drive



- ICR is constrained to lie along a line
- $n = 1$
- $\delta_m = 2$

Omni-wheel



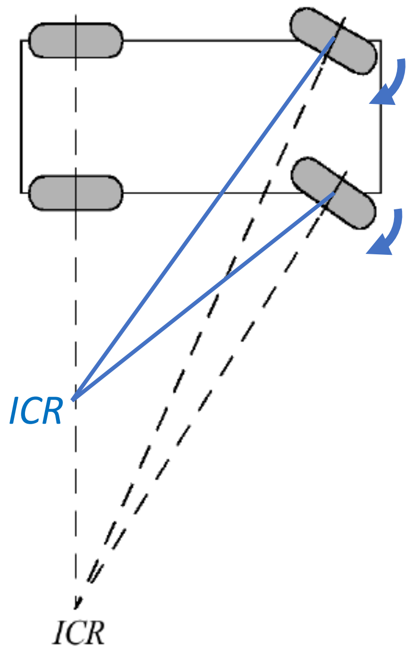
- ICR can be anywhere on the plane
- $n = 0$
- $\delta_m = 3$

R. Siegwart, I. R. Nourbakhsh, D. Scaramuzza. Introduction to autonomous mobile robots. The MIT Press. Second edition. 2011.

Degree of steerability

- The number of constraints on the position of ICR released due to the addition of steering

Ackerman-steering



- Due to the addition of two steering wheels, constraint on ICR is released from being a fixed point (2 constraints) to lying along a line (1 constraint)
- $\delta_s = 1$

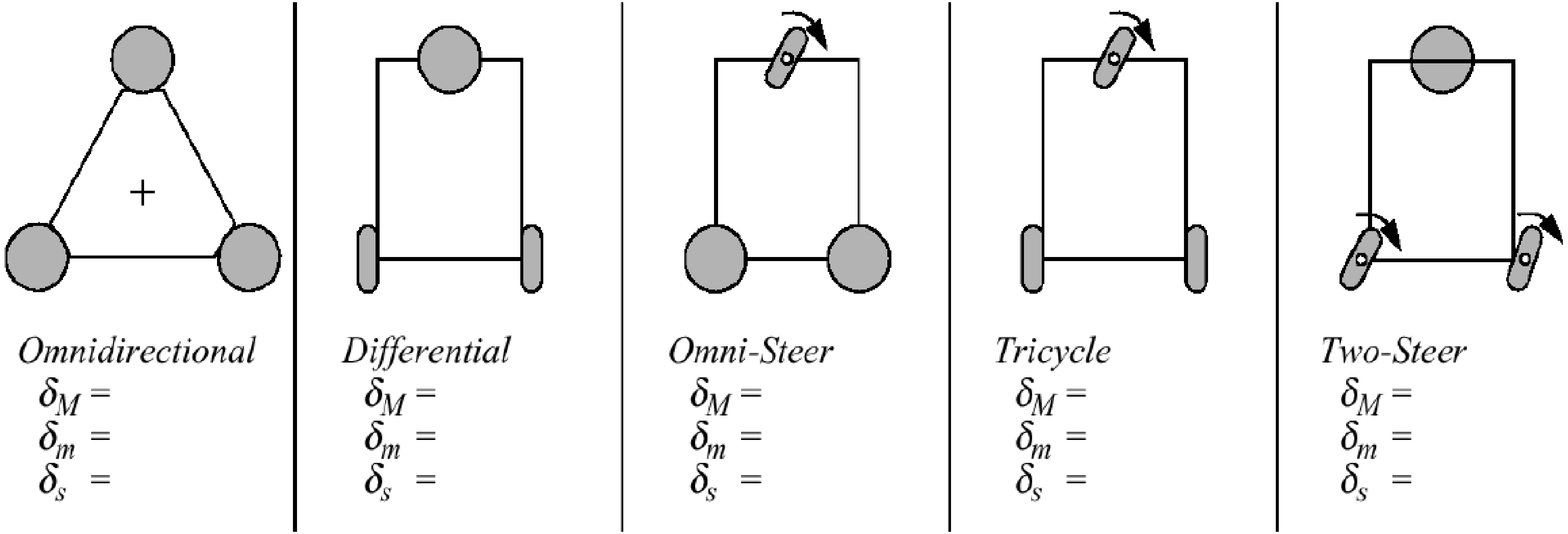
Degree of manoeuvrability

- Degree of *Manoeuvrability*: $\delta_M = \delta_m + \delta_s$
- For any robot with $\delta_M = 2$, the ICR is always constrained to lie along a line
- For any robot with $\delta_M = 3$, the ICR is not constrained and can be set to any point on the plane
- Example of $\delta_M = 1$?

Five basic configurations with three wheels

https://www.sli.do/ #4110

- $\delta_m = 3 - n$ (n is the number of constraints on the position of *ICR without considering steering*)
- δ_s is the number of constraints on the position of ICR released due to steering
- $\delta_M = \delta_m + \delta_s$



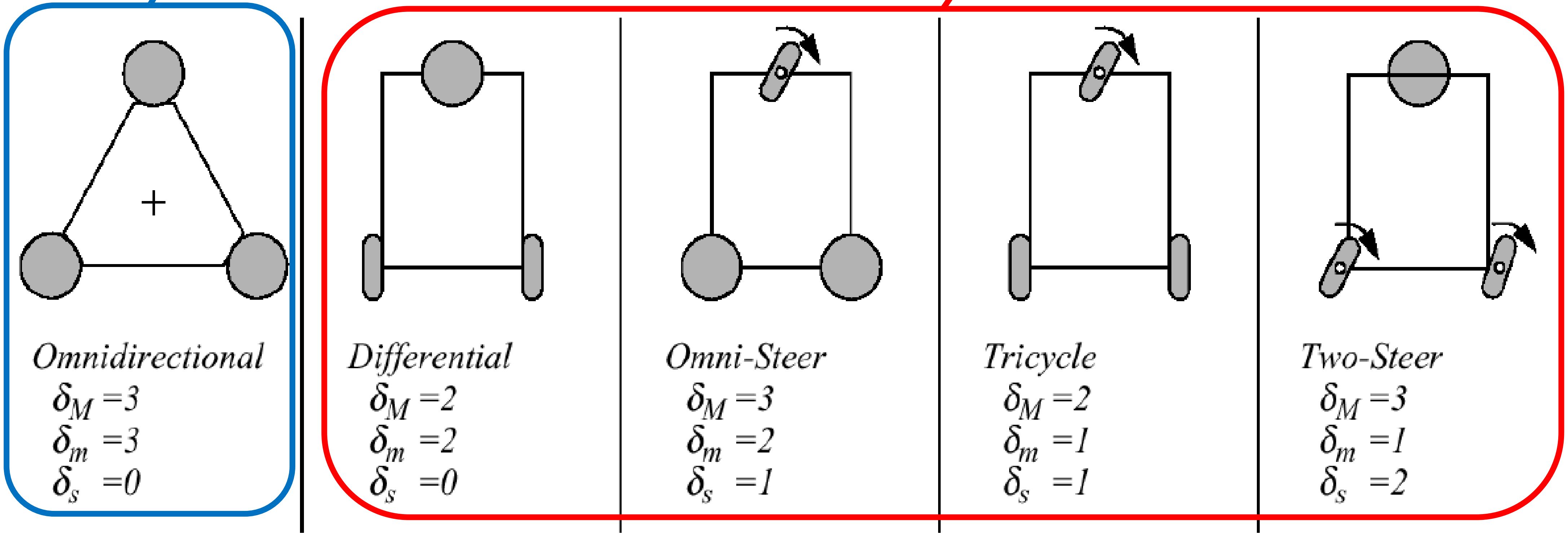
R. Siegwart, I. R. Nourbakhsh, D. Scaramuzza. Introduction to autonomous mobile robots. The MIT Press. Second edition. 2011.

What are the Manoeuvrability, Mobility, and Steerability of a two-wheel differential-drive robot?

ⓘ Start presenting to display the poll results on this slide.

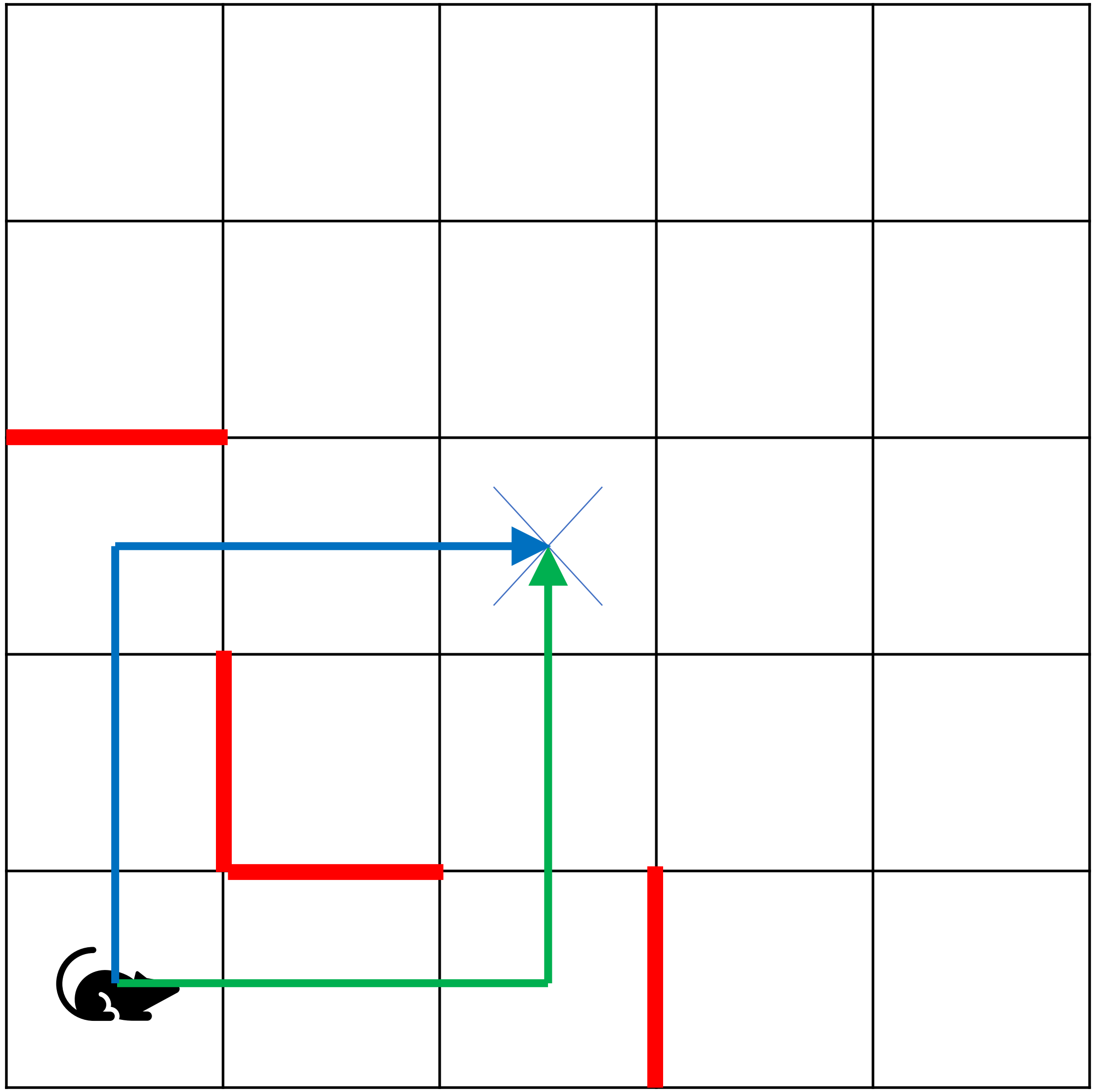
Holonomic or nonholonomic? - Another method to determine

- Holonomic systems
  - Mobility δ_m = workspace DOF
- Nonholonomic systems
  - Mobility δ_m < workspace DOF

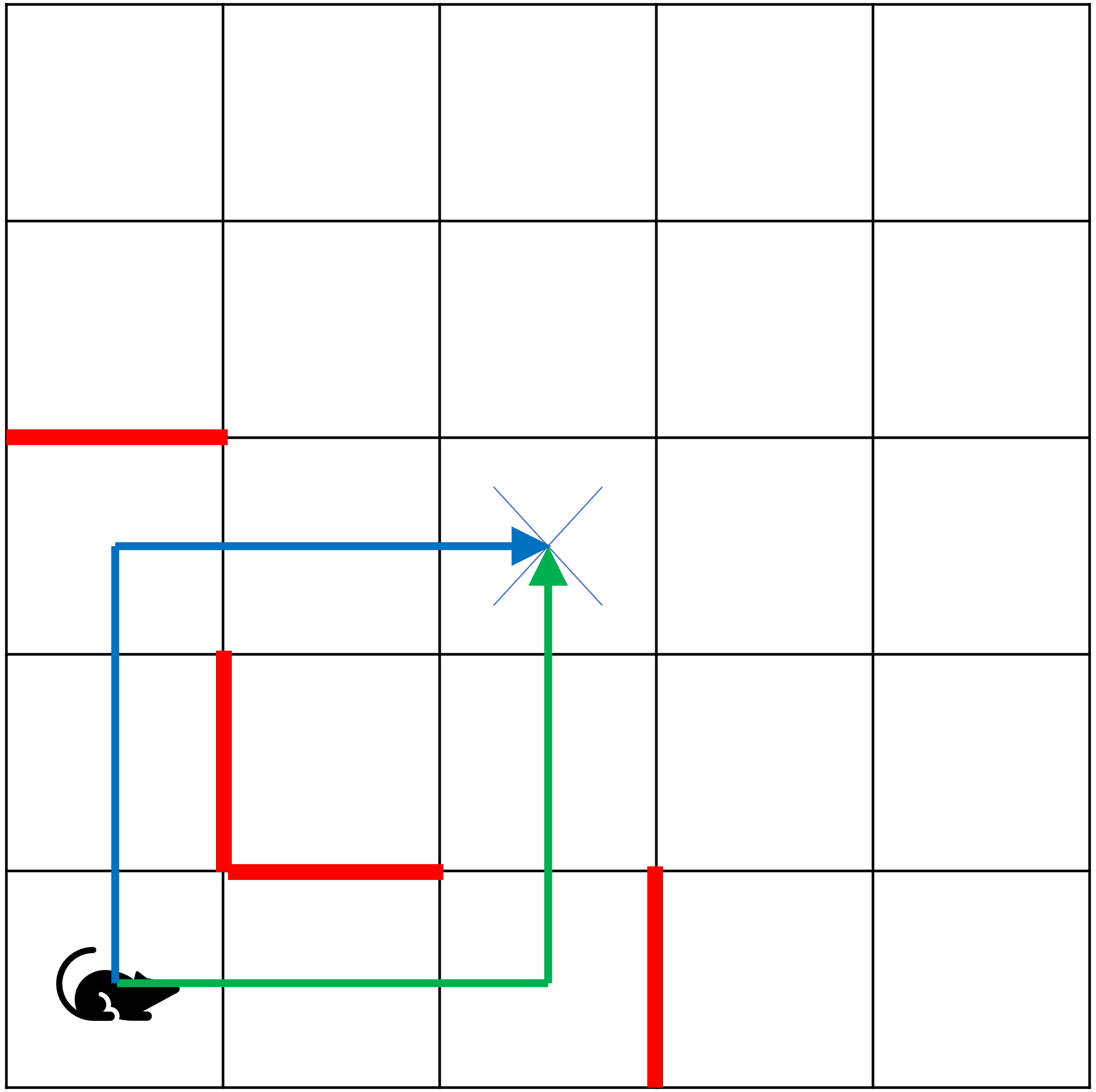


R. Siegwart, I. R. Nourbakhsh, D. Scaramuzza. Introduction to autonomous mobile robots. The MIT Press. Second edition. 2011.

For a holonomic robot, are the following two paths equally optimal?



For a nonholonomic robot, are the following two paths equally optimal?

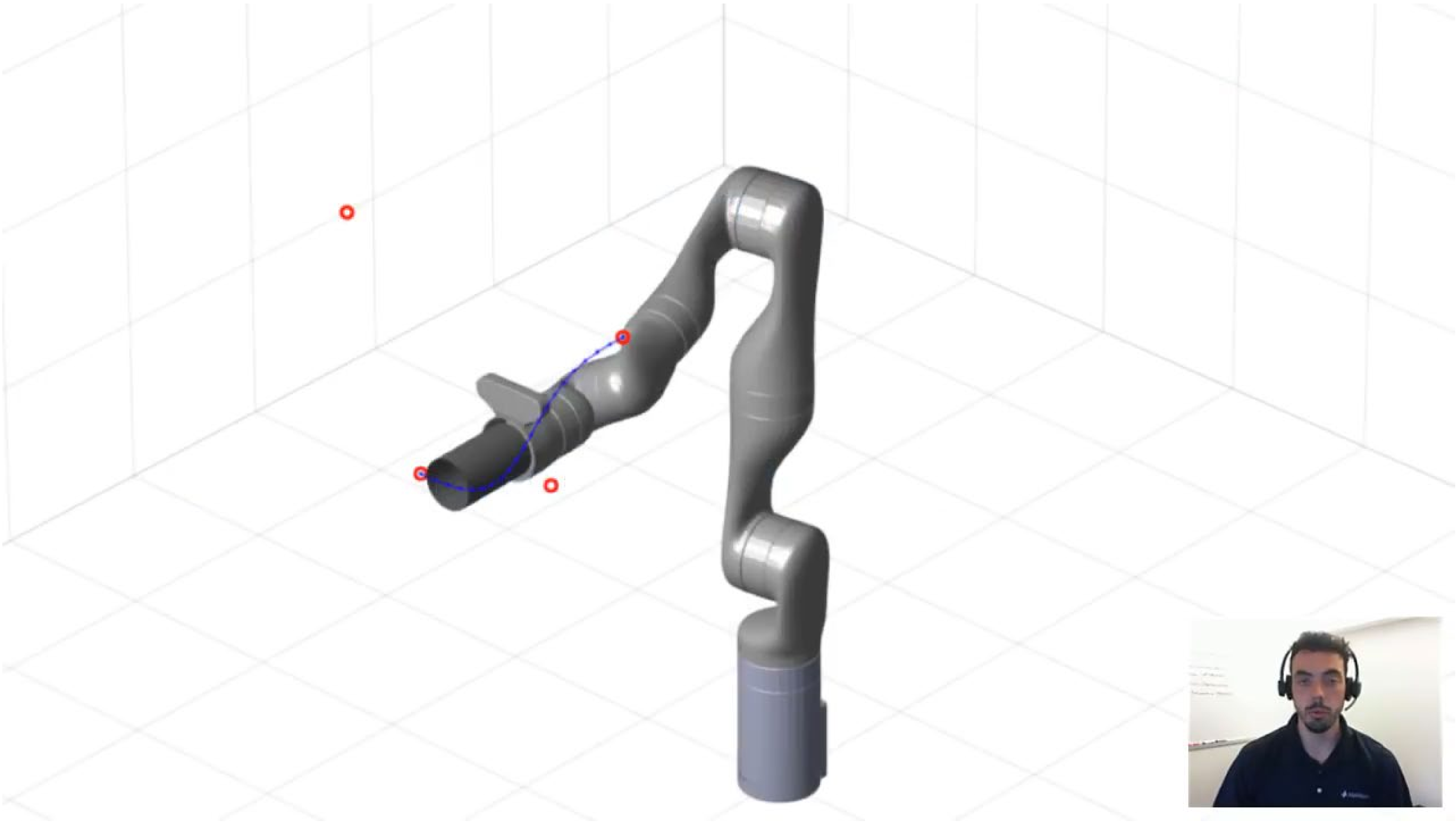


Trajectory Generation

What is the difference between a path and a trajectory?

https://robotacademy.net.au/lesson/paths-and-trajectories/

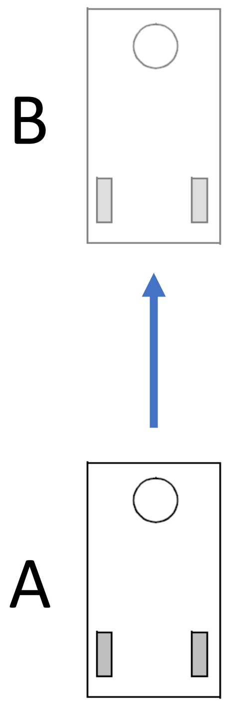
Trajectory – Example with manipulators



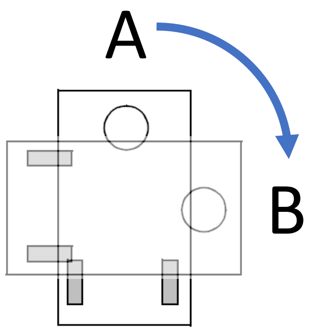
https://www.youtube.com/watch?v=Fd7wjZDoh7g&t=1s

We only consider two basic trajectories in this lecture

- Linear motion from A to B



- Rotation from A to B



Trajectory generation

- Problem statement
  - Given a start position (angle) and an end position (angle), determine a profile for the motion (position, velocity, acceleration, etc.) with respect to time.

- Methods
  - Cubic polynomial trajectory
  - Minimum time trajectory (Bang-Bang trajectory)
  - …

Trajectory generation

- Problem statement
  - Given a start position (angle) and an end position (angle), determine a profile for the motion (position, velocity, acceleration, etc.) with respect to time.

- Methods
  - **Cubic polynomial trajectory**
  - Minimum time trajectory (Bang-Bang trajectory)
  - …

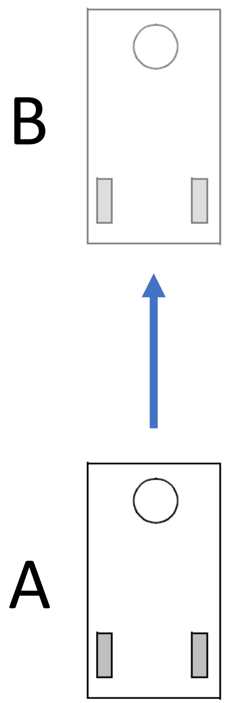
Cubic polynomial trajectory

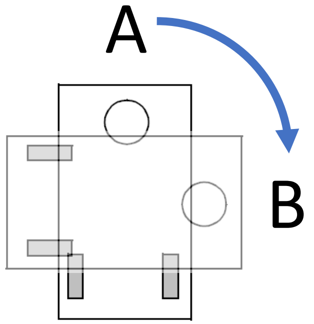
- Describing position (angle) as a cubic polynomial.

Position -> $q(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3$

Velocity -> $\dot{q}(t) = a_1 + 2a_2 t + 3a_3 t^2$

Acceleration -> $\ddot{q}(t) = 2a_2 + 6a_3 t$





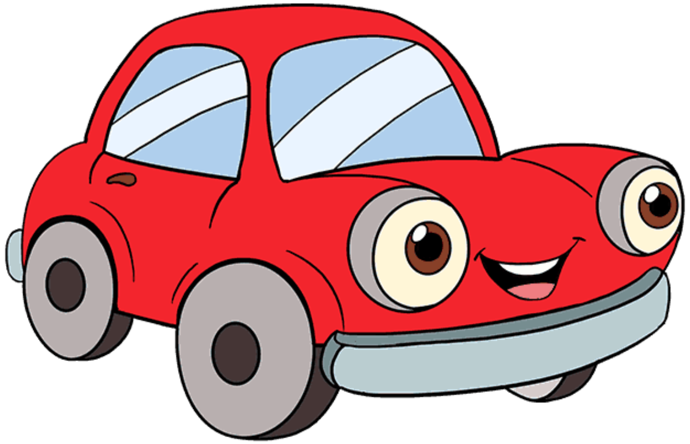
- Linear motion
- Rotation

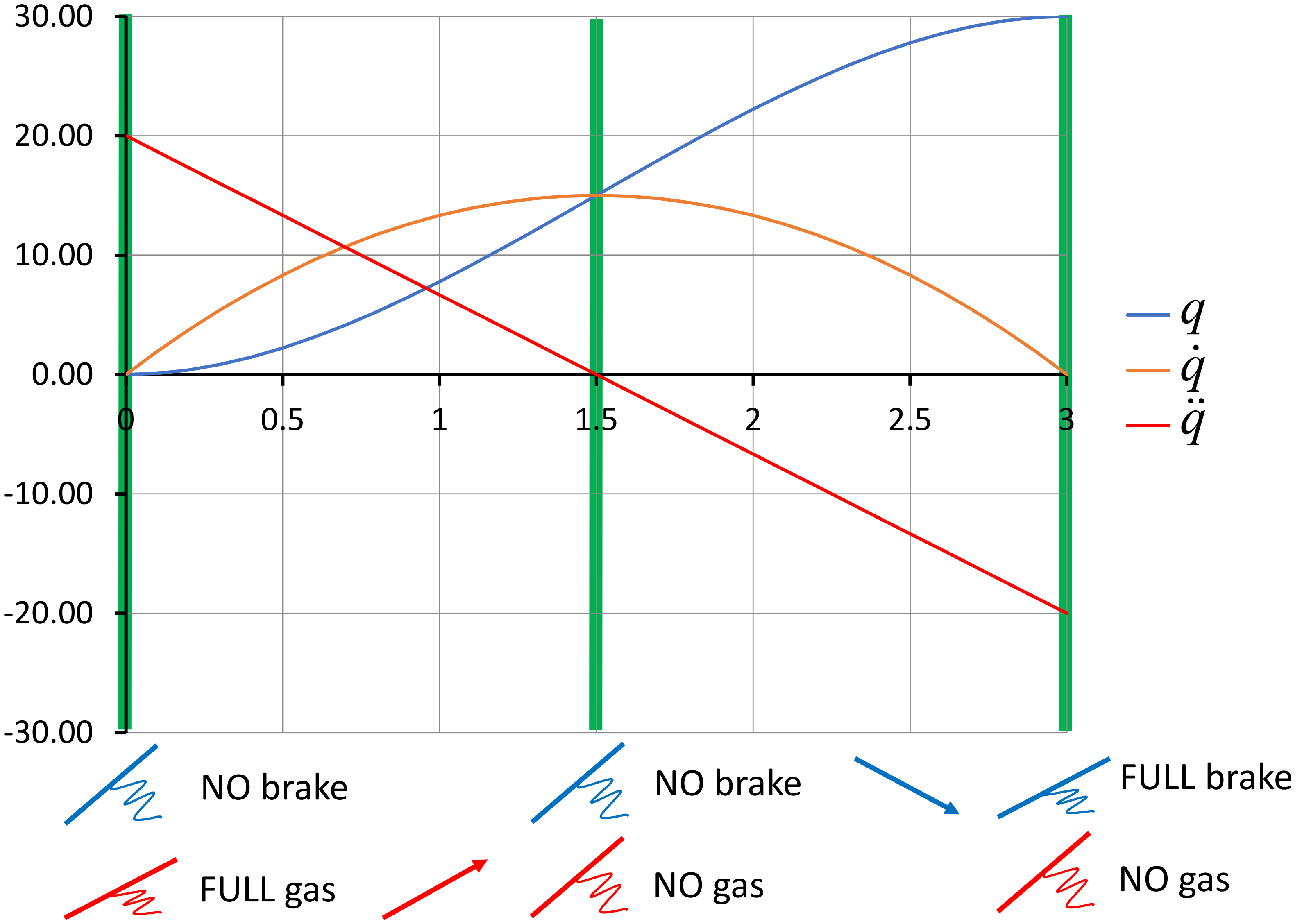
Cubic polynomial trajectory

Position -> $q(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3$

Velocity -> $\dot{q}(t) = a_1 + 2a_2 t + 3a_3 t^2$

Acceleration -> $\ddot{q}(t) = 2a_2 + 6a_3 t$





Generate a cubic polynomial trajectory?

- Describing position (angle) as a cubic polynomial.

$$q(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3$$
$$\dot{q}(t) = a_1 + 2a_2 t + 3a_3 t^2$$
$$\ddot{q}(t) = 2a_2 + 6a_3 t$$

- Find constants by setting initial and final positions and velocities and choosing a trajectory time.

- **Four variables, need four independent equations**

Cubic polynomial trajectory - Example

- Use a cubic polynomial to describe a motion from 0 to 90 degrees in 3 seconds, with zero start and stop velocities.
  - First write the two position equations:
    - $q(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3$ $\Rightarrow$

$$0 = a_0$$

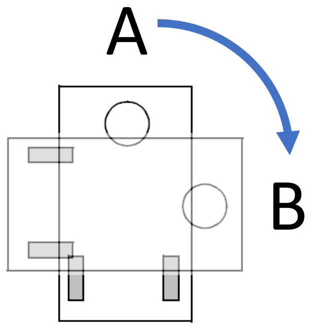
$$90 = a_0 + 3a_1 + 9a_2 + 27a_3$$

  - Then write the two velocity equations:
    - $\dot{q}(t) = a_1 + 2a_2 t + 3a_3 t^2$ $\Rightarrow$

$$0 = a_1$$

$$0 = a_1 + 6a_2 + 27a_3$$

  - Solve for a_0, a_1, a_2 and a_3



Cubic polynomial trajectory - Example

- Solve for a_0, a_1, a_2 and a_3

$$\begin{cases} 0 = a_0 \\ 90 = a_0 + 3a_1 + 9a_2 + 27a_3 \\ 0 = a_1 \\ 0 = a_1 + 6a_2 + 27a_3 \end{cases} \Rightarrow a_0 = 0, a_1 = 0$$

$$q(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3$$
$$\dot{q}(t) = a_1 + 2a_2 t + 3a_3 t^2$$
$$\ddot{q}(t) = 2a_2 + 6a_3 t$$

$$\begin{cases} 90 = 9a_2 + 27a_3 \\ 0 = 6a_2 + 27a_3 \end{cases} \Rightarrow a_2 = 30, a_3 = -6.67$$

$$q(t) = 30t^2 - 6.67t^3$$
$$\dot{q}(t) = 60t - 20t^2$$
$$\ddot{q}(t) = 60 - 40t$$

Homework: Write a program for cubic polynomial trajectory generation.

Cubic polynomial trajectory - Summary

Advantages	Disadvantages

• **Spatial and temporal accuracy**
• **Enable smooth connection of trajectories**
    • given positions and velocities at connection points
	• **Doesn't readily facilitate minimum time operations**
	    • Not using full actuator capability
	• **Smoothness**
	    • Infinite jerks (derivative of acceleration) at start and end

Trajectory generation

- Problem statement
  - Given a start position (angle) and an end position (angle), determine a profile for the motion (position, velocity, acceleration, etc.) with respect to time.

- Methods
  - Cubic polynomial trajectory
  - **Minimum time trajectory (Bang-Bang trajectory)**
  - …

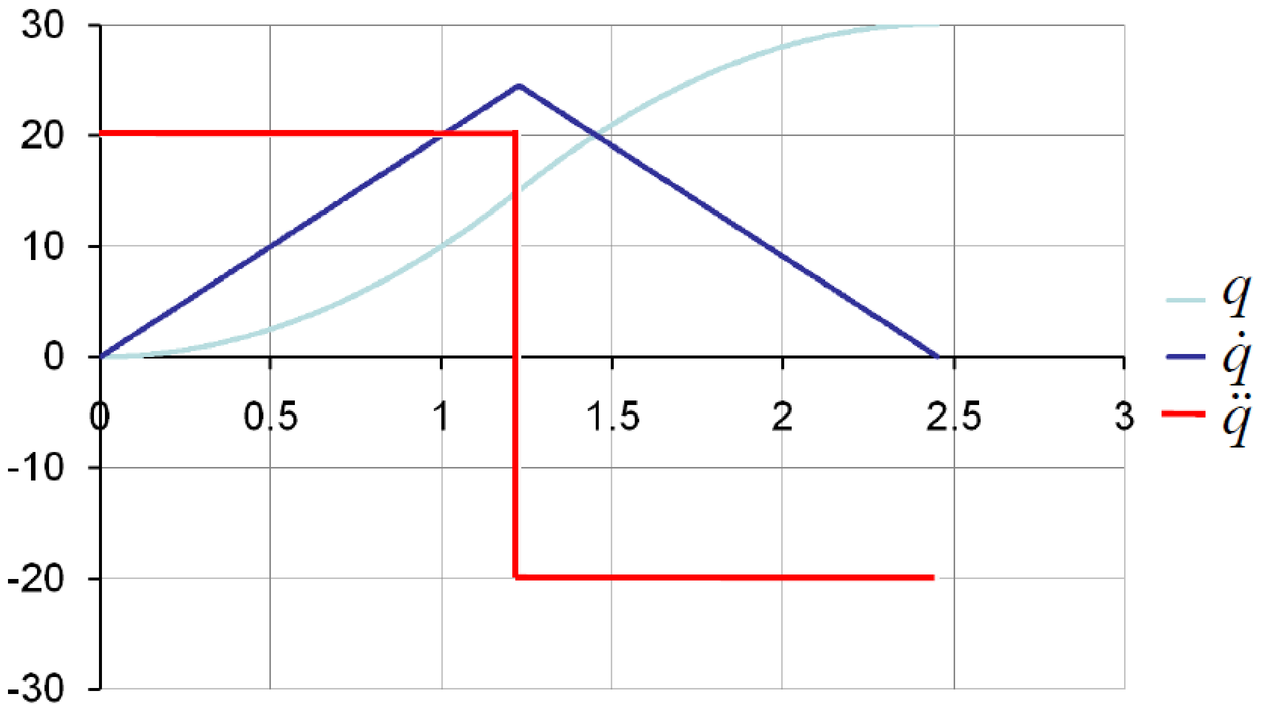
Minimum time trajectory (Bang-Bang trajectory)

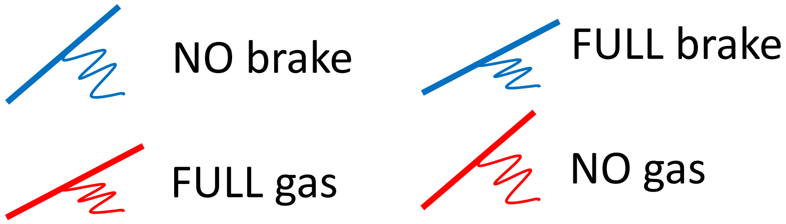
https://www.youtube.com/watch?v=FwYjHyHoimU

Minimum time trajectory (Bang-Bang trajectory)

- Minimum time trajectory is achieved by using maximum positive acceleration until:
  - Maximum velocity is reached, OR
  - Minimum braking distance is reached

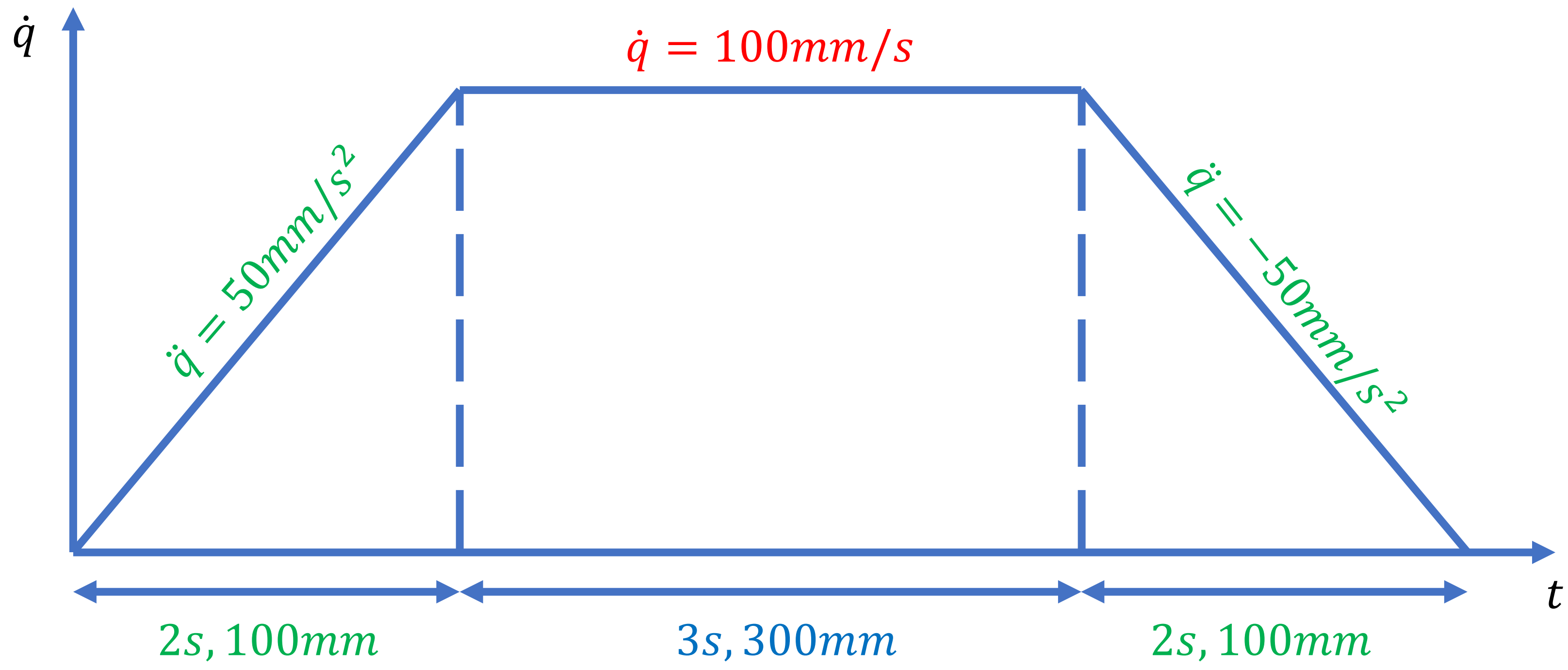
- Then switch to maximum negative acceleration

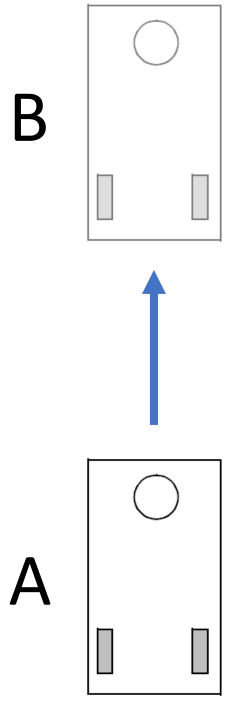




Bang-Bang trajectory - Example

- Using a bang-bang trajectory, how long does it take a mobile robot to move from rest to rest through *500mm*, if the maximum acceleration is *50mm/s*^{2} with a maximum velocity of *100mm/s*?



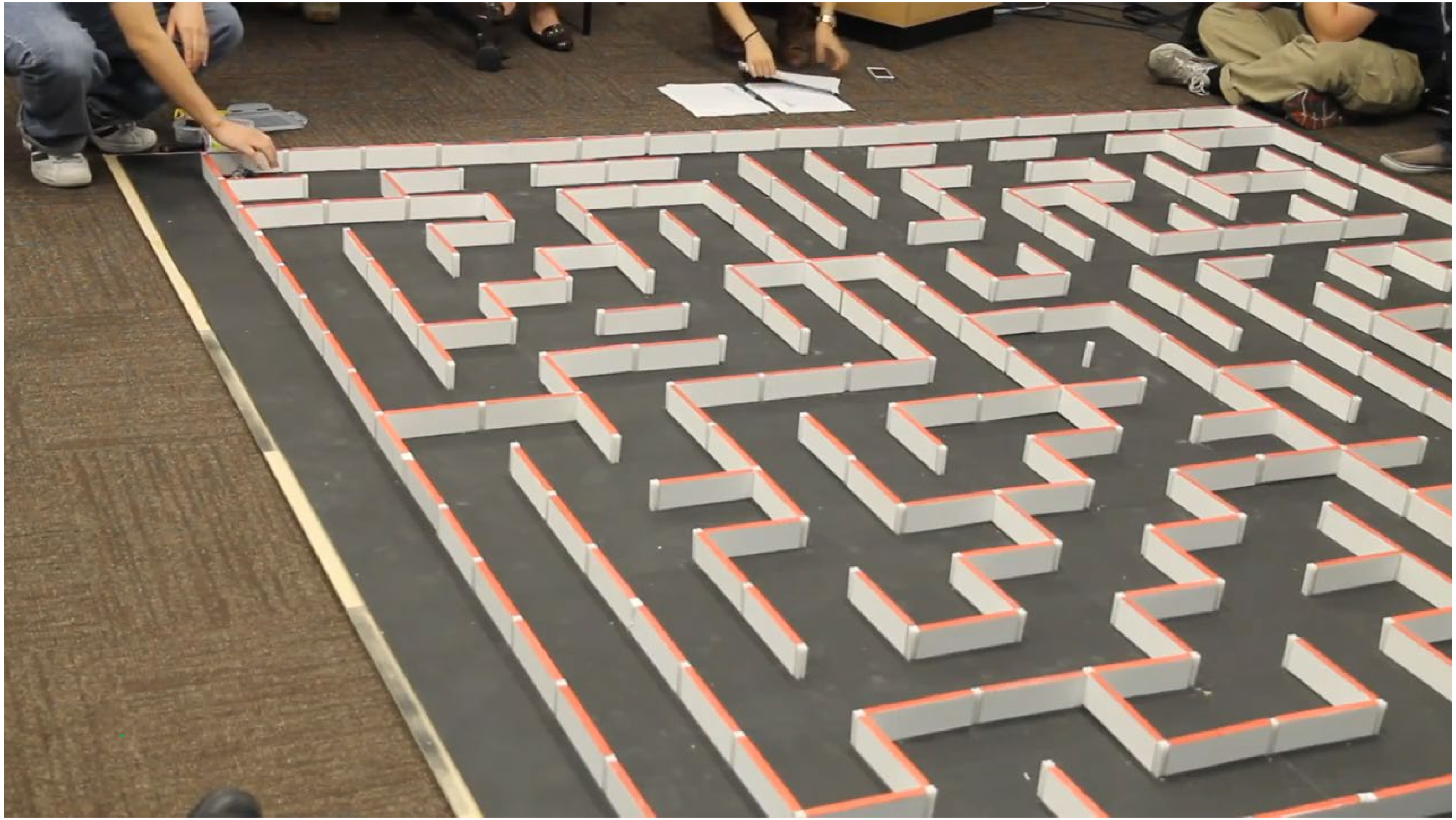


Minimum time trajectory (Bang-Bang trajectory) - Summary

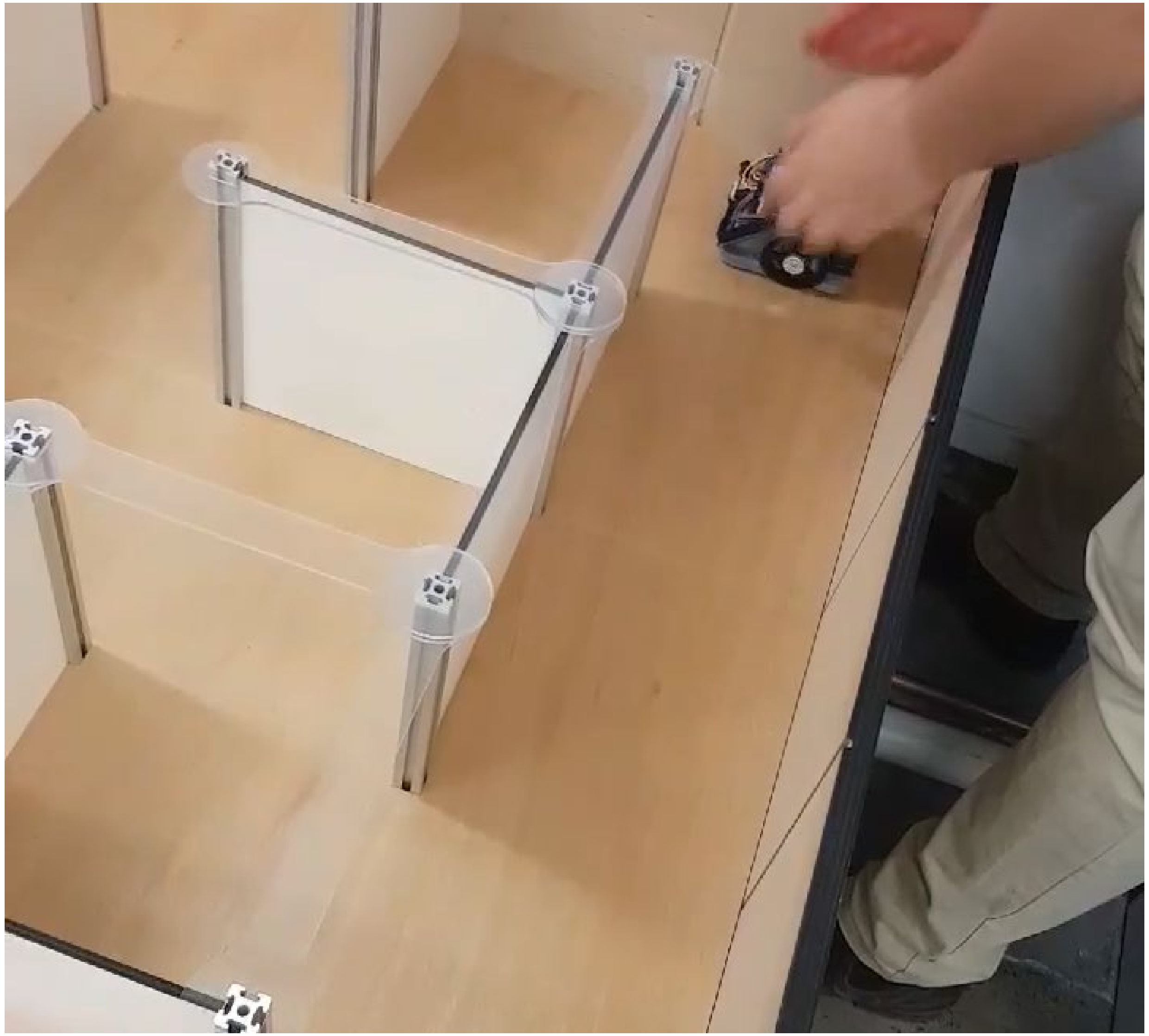
Advantages	Disadvantages

• **Simple algorithm**
 	
• **Sometimes achieves "minimum" time**
    • Assumes we know acceleration limit
	• **Doesn't always achieve "minimum" time**
	    • Acceleration limit may not be known or may change
	• **Need to use fine time step or handle change from +ve to –ve acceleration carefully**
	    • Otherwise trajectory will not land precisely at required position/angle
	• **Smoothness**
	    • Unbounded jerks (derivative of acceleration) at start, middle, and end

What trajectory is used here?



https://youtu.be/aXMcEDy-ly8



Leigh Huang, Yuen Chan, Chris Wong, Matthew Buffa

Kinematic Control

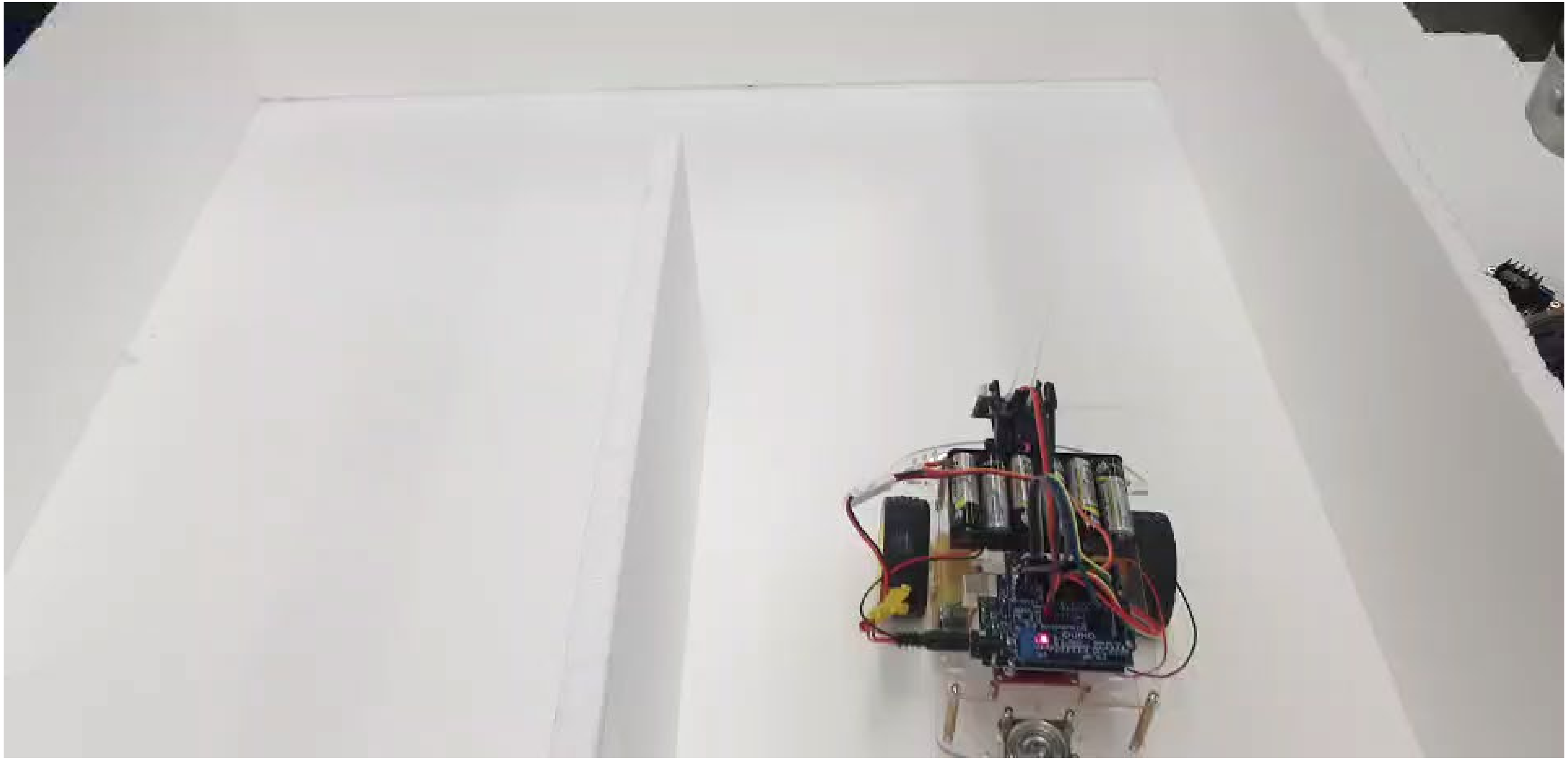
Kinematic control

- Open-loop control
- Feedback control
  - Bang-Bang control
  - PID control
  - …

Open-loop control

- Given a trajectory, generate a series of commands and send to the actuators, and then execute the commands

Open-loop control - Example



Open-loop control - Summary

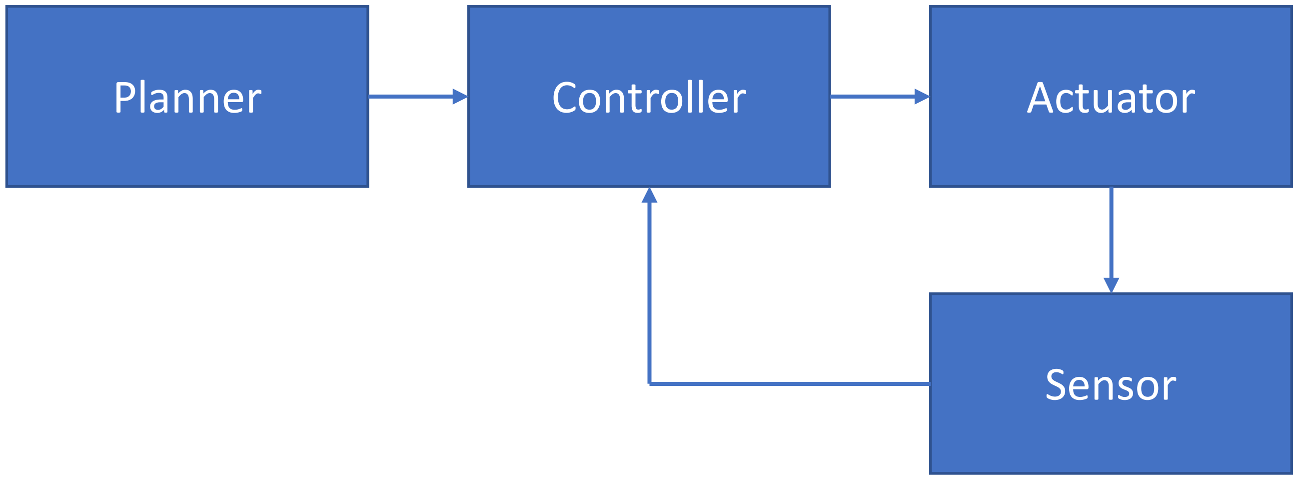
- Easy to implement

- Accuracy of control relies on accuracy of model
  - Calibration may improve the accuracy!

- Does not adapt to unexpected changes of the environment

Feedback control

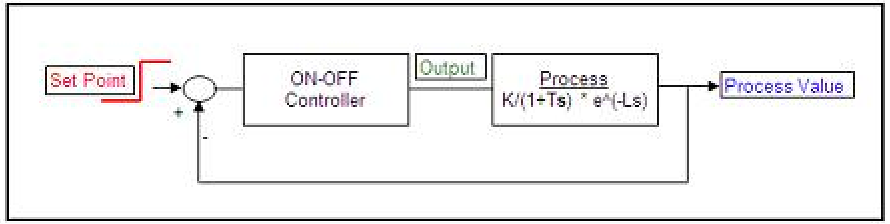
- Use the measurement of sensors to adjust the commands generated by the controller and then send to the actuators.



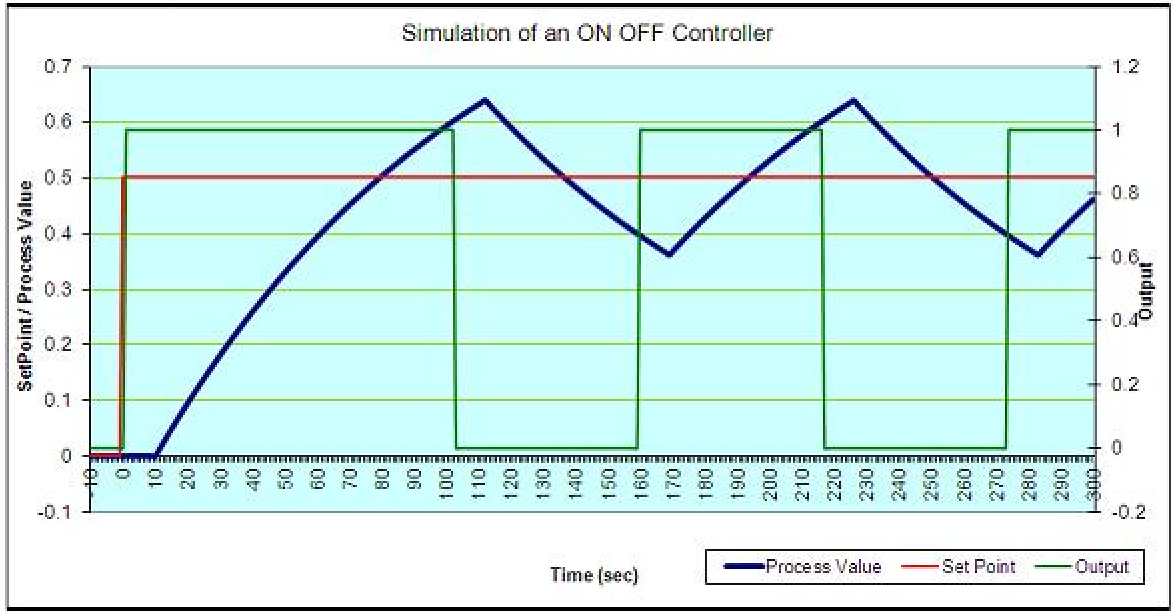
Bang-Bang control / On-Off control / Hysteresis control

Process	
Gain (K)	1
Time Constant (T)	100
Delay (L)	10

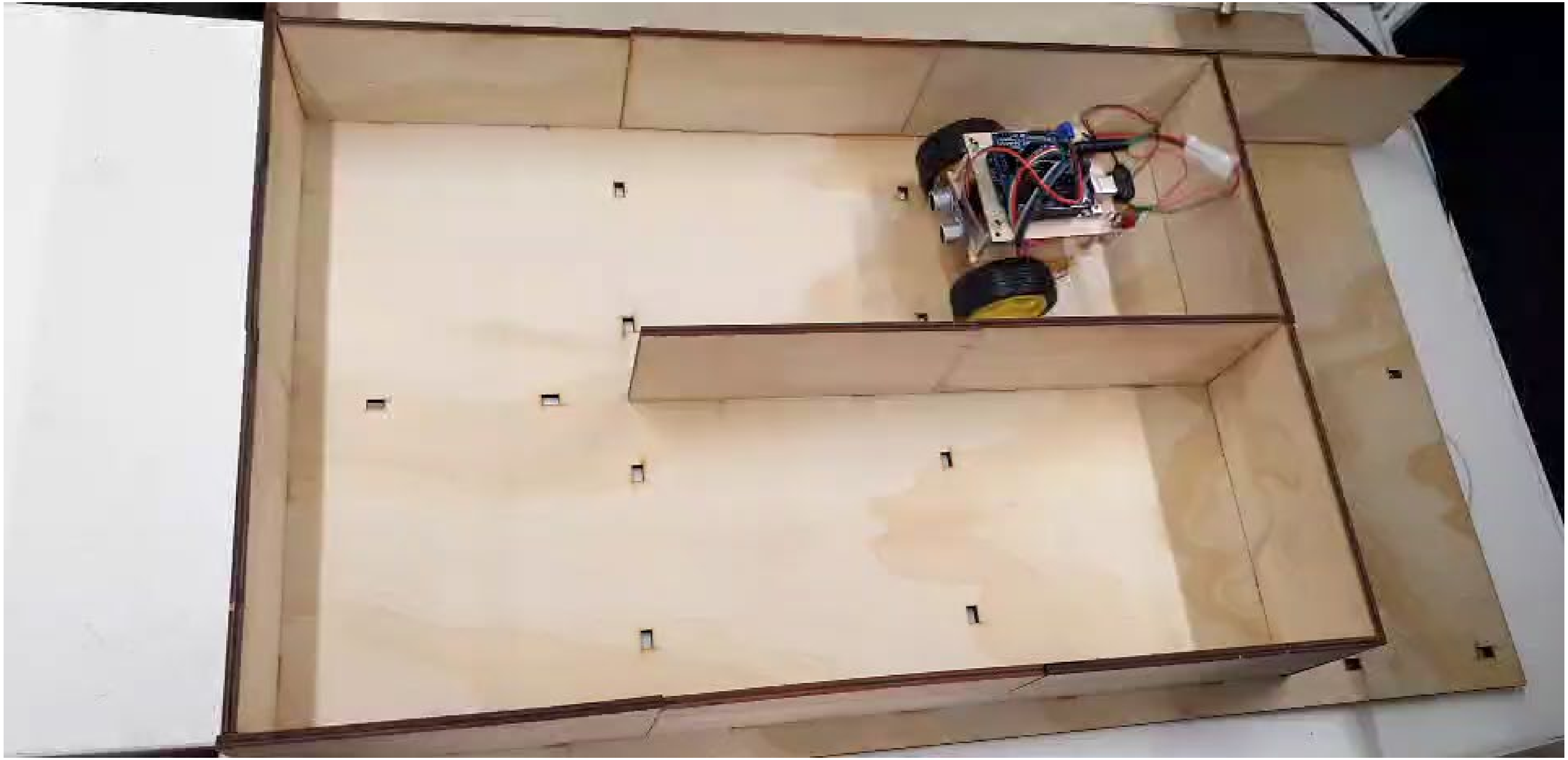
Controller	
Hysterisys	10%



(All times in seconds)



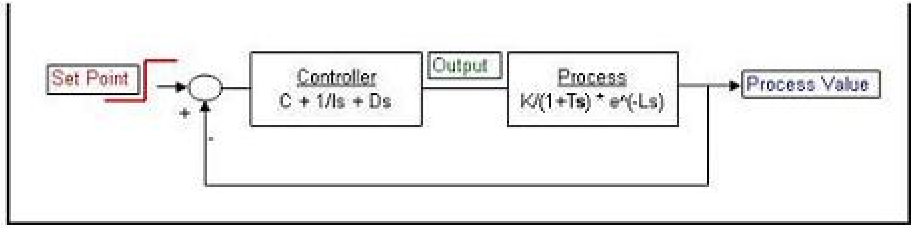
Bang-Bang control - Example



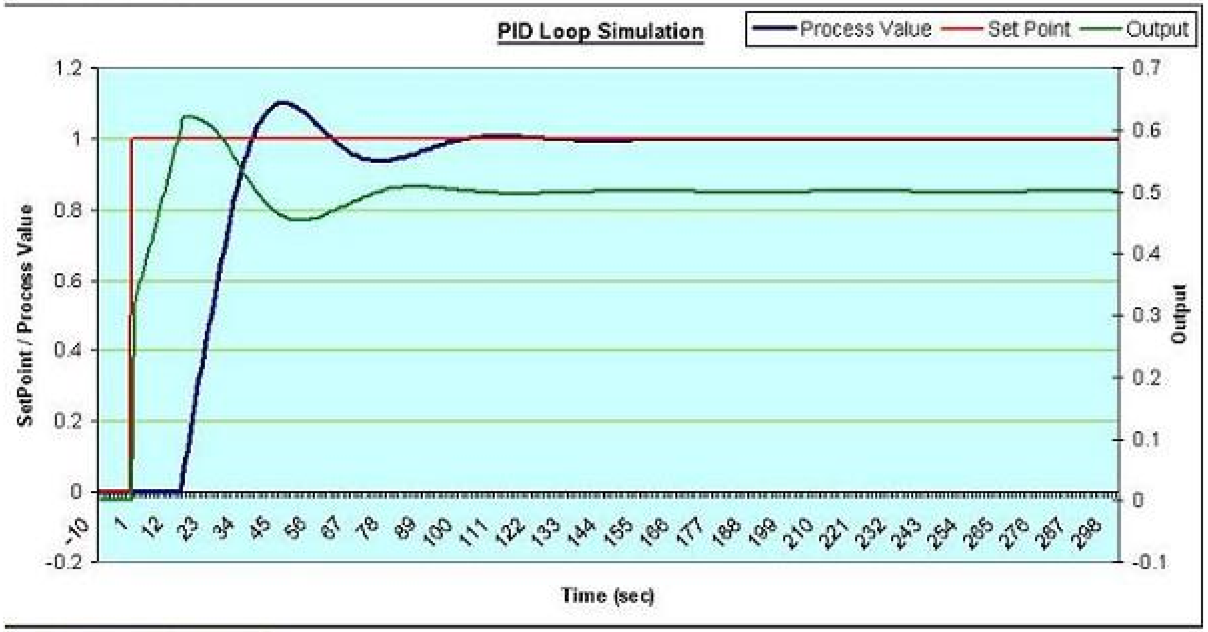
PID control

Process	
Gain (K)	2
Time Constant (T)	10
Delay (L)	15

Controller	
Gain (C)	0.3
Integral (I)	50
Derivative (D)	

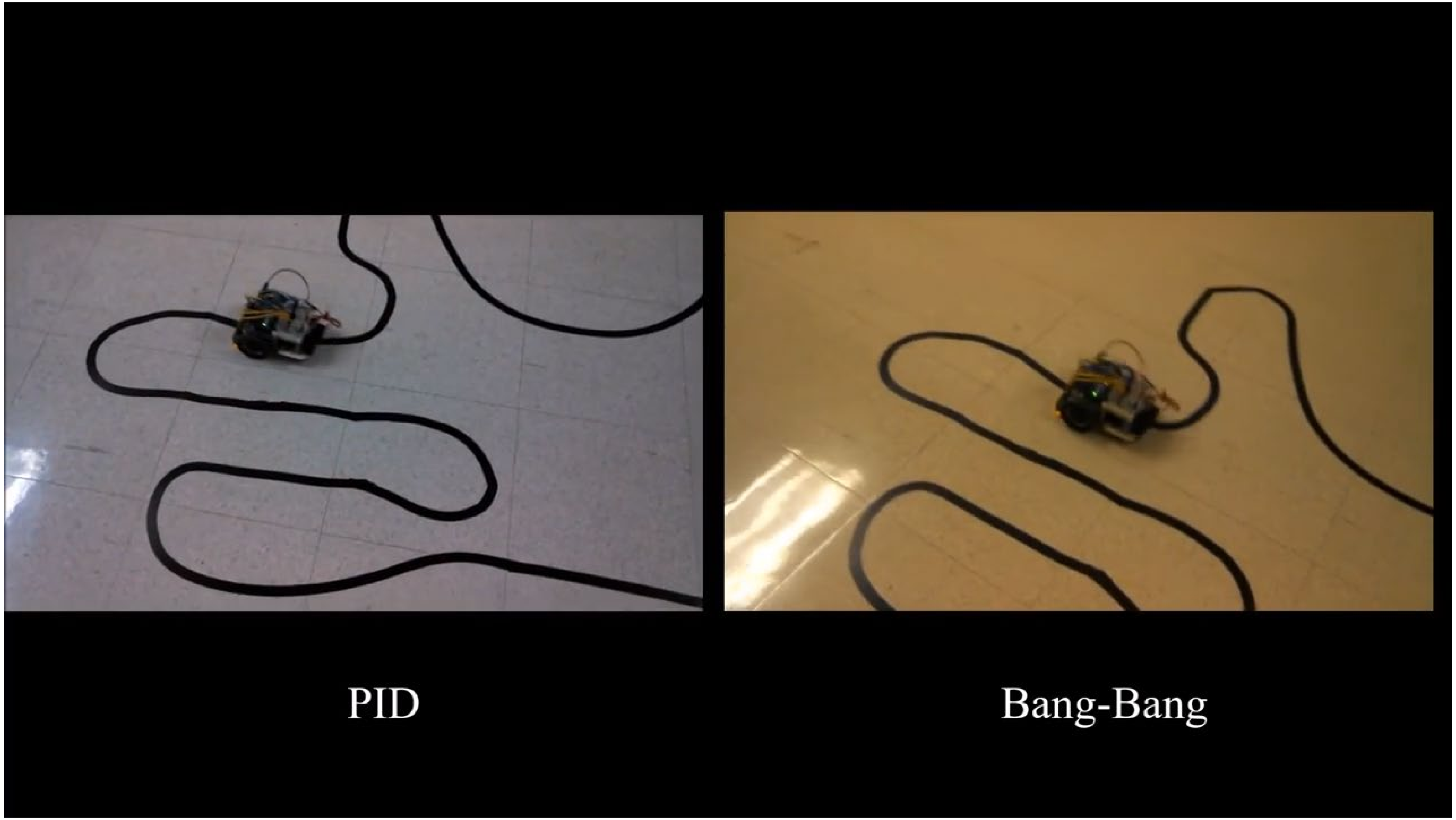


(All times in seconds)



http://www.engineers-excel.com

PID vs. Bang-Bang



https://www.youtube.com/watch?v=t3R9cPq1aYk

PID control - example

UNSW SYDNEY

slido

Is feedback control needed in a real Micromouse Competition?

ⓘ Start presenting to display the poll results on this slide.

What we have learnt today

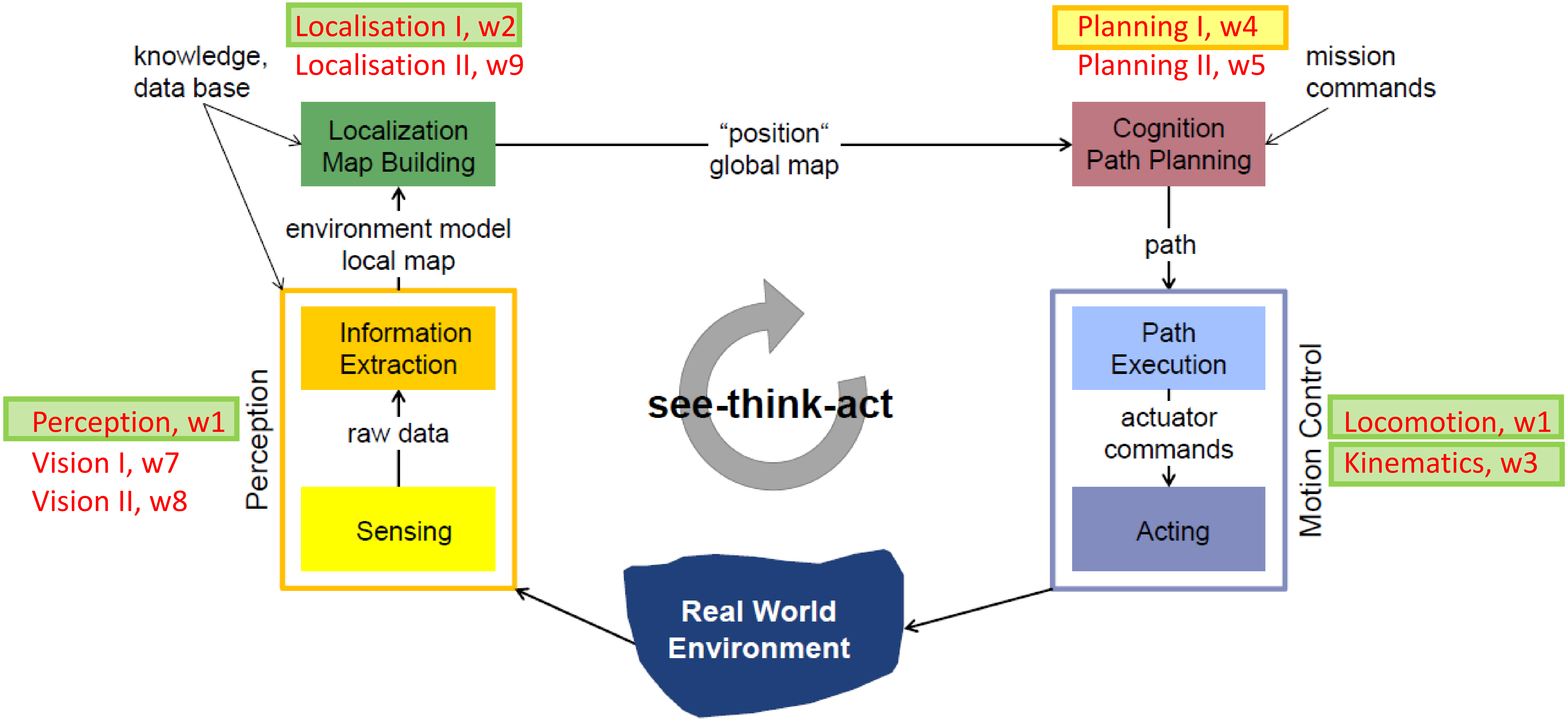
- Differential (velocity) kinematics is usually studied for nonholonomic robots

- Nonholonomic robots are robots whose mobility δ_m < workspace DOF

- Different trajectories can be generated for a planned path
  - Cubic polynomial trajectory
  - Bang-Bang trajectory
  - …

- Different control methods can be used for executing a trajectory
  - Open-loop control
  - Bang-Bang control
  - PID control
  - …

Next week: Planning I



R. Siegwart, I. R. Nourbakhsh, D. Scaramuzza. Introduction to autonomous mobile robots. The MIT Press. Second edition. 2011.