Foundations of Robotics

Lec 1: Overview

主讲人 滕瀚哲
美国加州大学河滨分校
ARCS实验室博士

Outline

1. Introduction
2. Course Outline
3. Course Logistics
4. Homework

Outline

Introduction

Robots around you – categorized according to application

Introduction

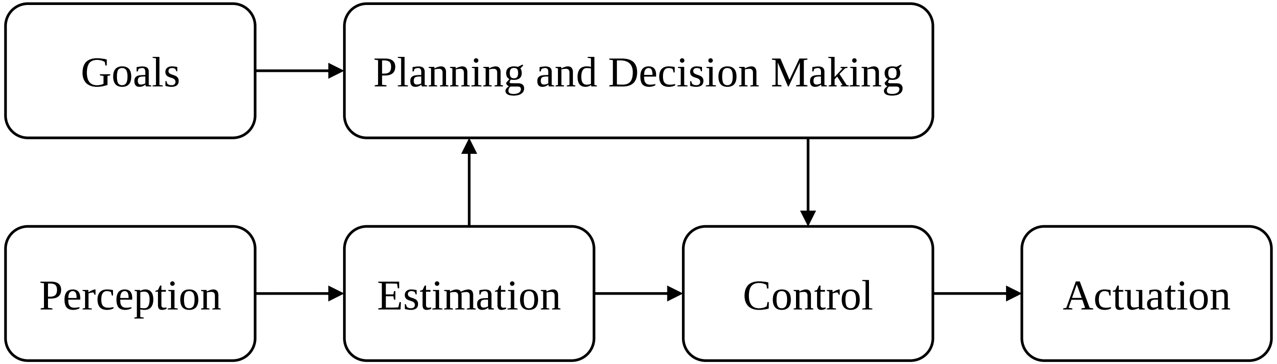
Robots around you – categorized according to mobility

Introduction

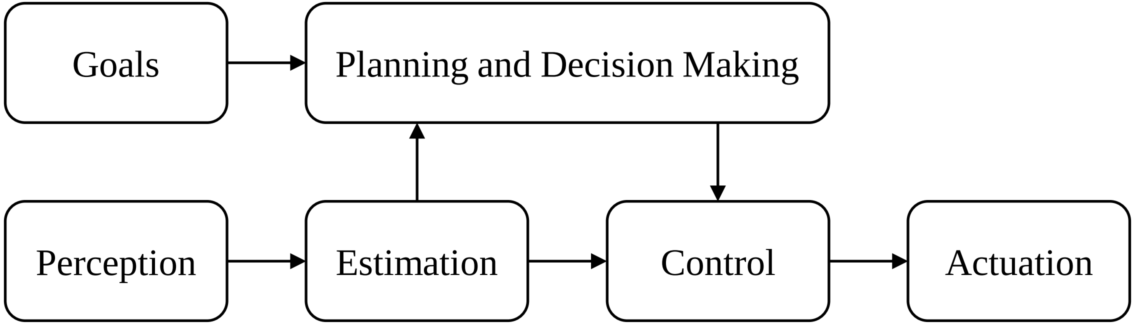
What is a robot?

Maja Matarić in her book “The Robotics Primer”:

- A robot is an **autonomous** system which exists in the **physical** world, can **sense** its environment, and can **act** on it to achieve some **goals**.



Introduction



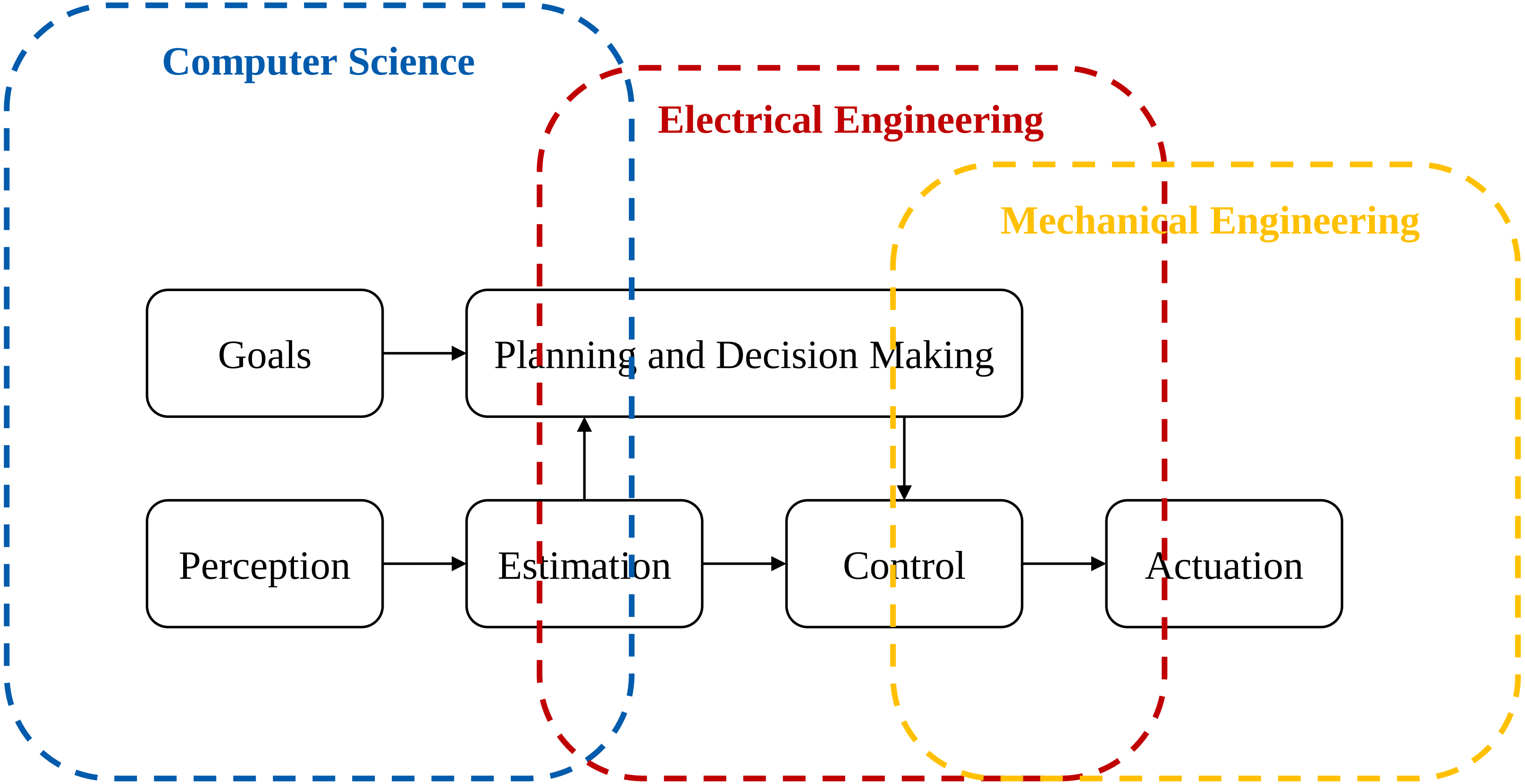
What are these components in the following examples?

a line-following robot

a robot vacuum cleaner

an autonomous driving car

Introduction



Introduction

Mechanical Engineering	frame design, force analysis, locknuts, belt friction, gear ratio, motor torque, infrared sensors	a beautiful appearance, brush pressure, docking system, wheel encoders, collision and lidar sensors	car frame, engine, fuel system, transmission, wheel encoders, perception sensors
Electrical Engineering	printed circuit board, analog/digital conversion, microcontroller, battery	power and circuits, tracking controller, non-holonomic robot model	energy system, circuits and chips, motion controller, adaptive cruise control, lane keeping
Computer Science	line-following strategy, transform goals to planning and decision-making algorithms	indoor localization, coverage path planning, trajectory generation with collision, trap recovery	motion estimation and planning, sensor fusion, scene understanding, object detection

Outline

1. Introduction
2. Course Outline
3. Course Logistics
4. Homework

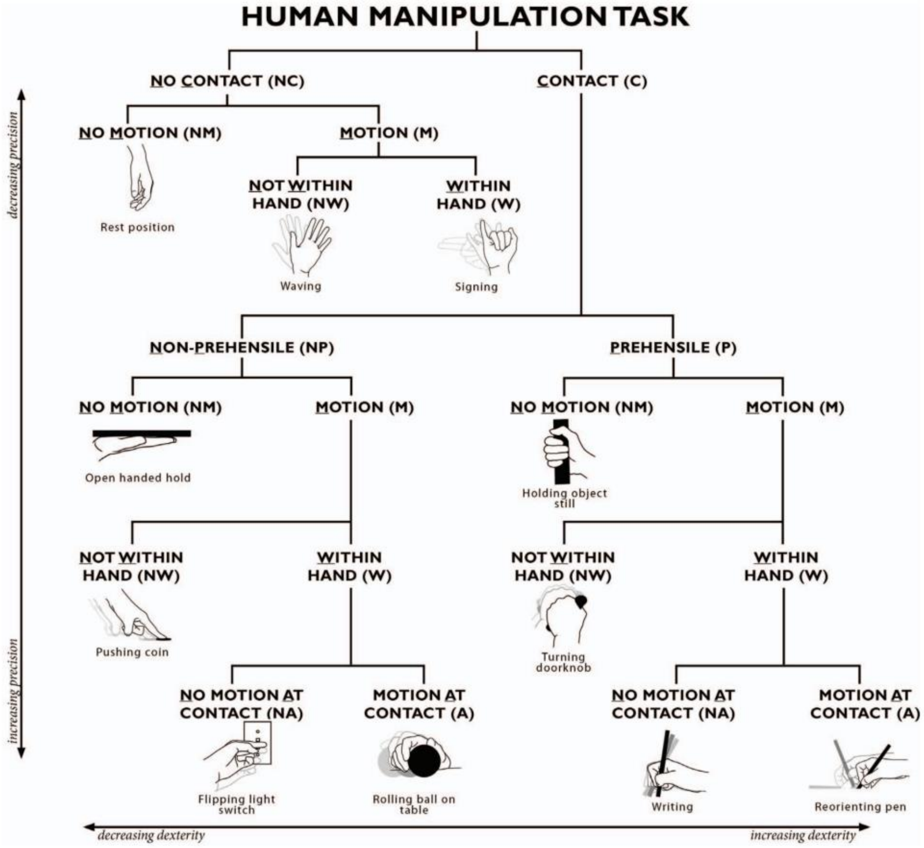
Course Outline

Example: how to control a manipulator to reach to any point you want?

- Lec 1: Overview
- Lec 2: Configuration Space
- Lec 3: Rigid-Body Motions: Rotation
- Lec 4: Rigid-Body Motions: Transformation
- Lec 5: Forward Kinematics
- Lec 6: Inverse Kinematics
- Lec 7: Trajectory Generation
- Lec 8: Motion Planning

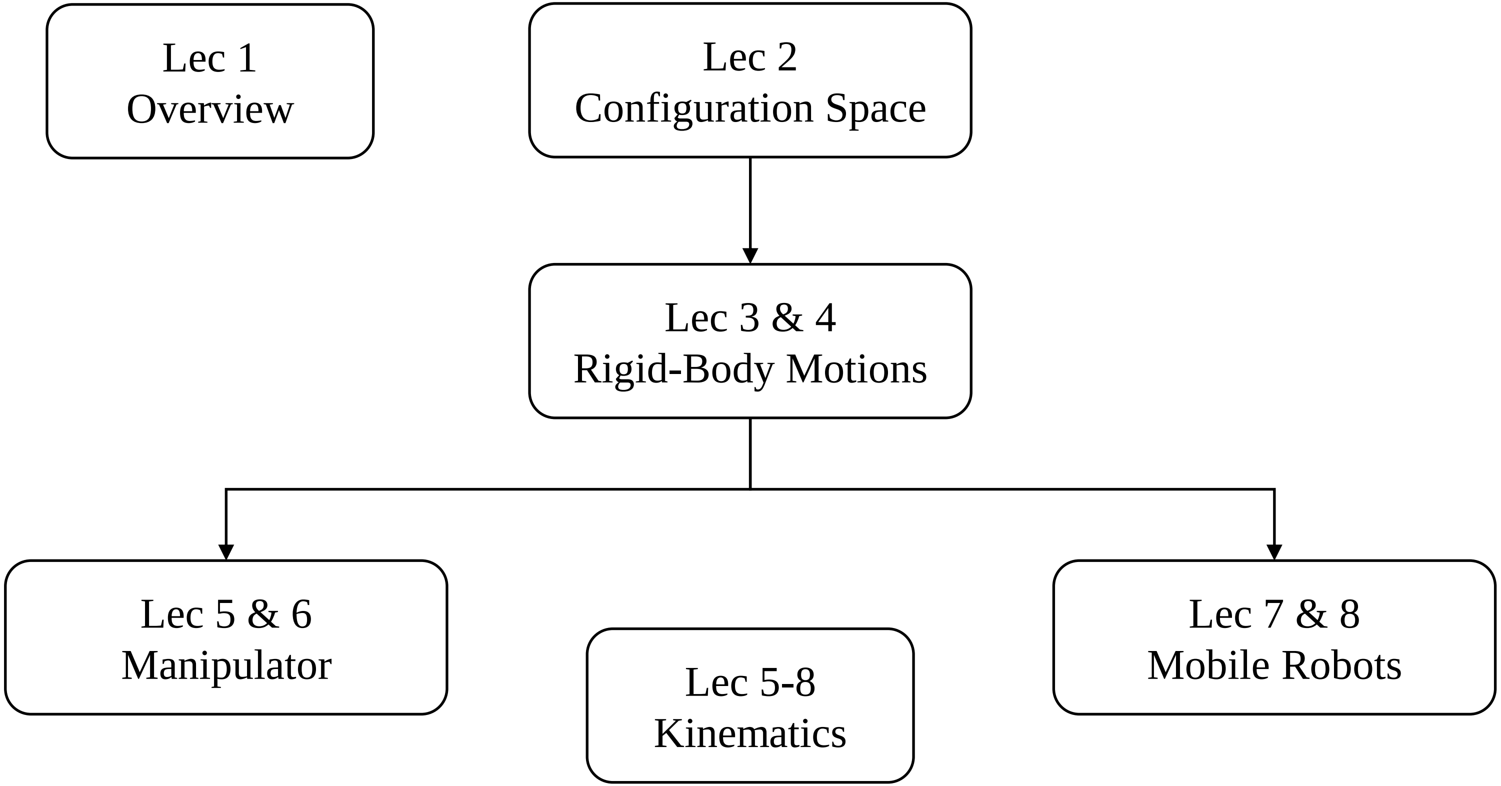
Course Outline

Future work? A long way to go!

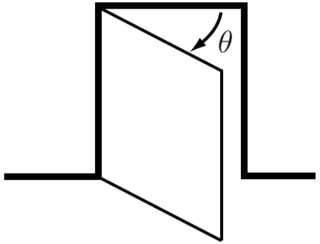


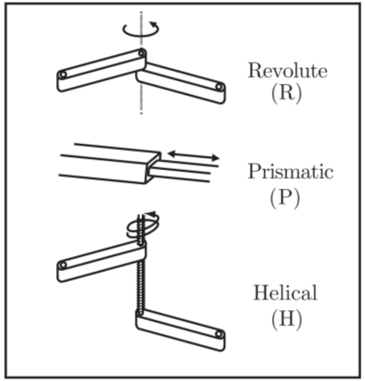
[1] I. Bullock and A. Dollar, “Classifying human manipulation behavior”

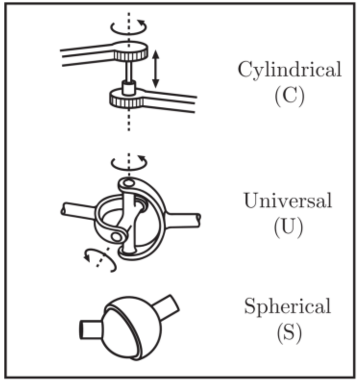
Course Outline

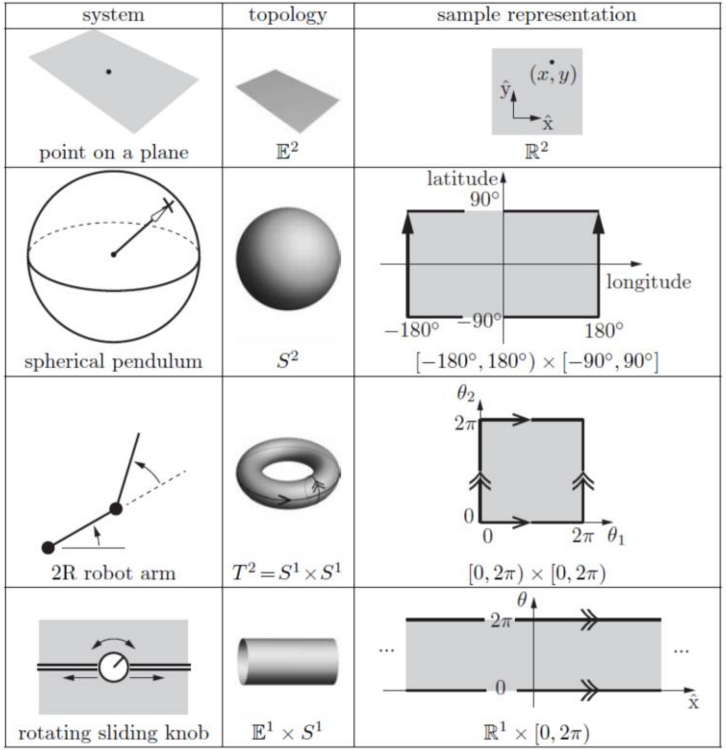


Lec 2: Configuration Space

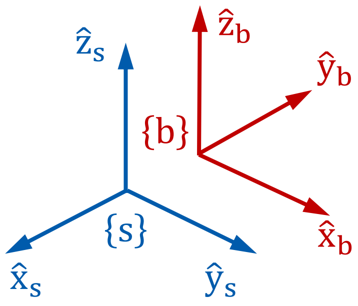








system	topology	sample representation
point on a plane	$\mathbb{E}^2$	$\mathbb{R}^2$
spherical pendulum	S^2	$[-180°, 180°) \times [-90°, 90°]$
2R robot arm	$T^2 = S^1 \times S^1$	$[0, 2\pi) \times [0, 2\pi)$
rotating sliding knob	$\mathbb{E}^1 \times S^1$	$\mathbb{R}^1 \times [0, 2\pi)$



$$R = [\hat{x}_b \;\; \hat{y}_b \;\; \hat{z}_b] = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Summary of the uses of rotation matrices

(1) represent an orientation
(2) change the reference frame (of a vector or a frame)
(3) rotate a vector or a frame (in its current frame)

Angular velocity

$$\omega = \hat{\omega}\dot{\theta}, \quad \dot{R} = [\omega]R$$

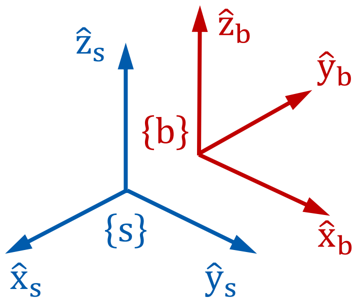
Exponential coordinate representation of rotation

$$e^{[\hat{\omega}]\theta} = I + \sin\theta\,[\hat{\omega}] + (1 - \cos\theta)[\hat{\omega}]^2 \in SO(3)$$

Bonus: other representations of rotation

- Euler angles vs. roll-pitch-yaw angles
- Unit quaternions

Lec 4: Rigid-Body Motions: Transformation



$$T = \begin{bmatrix} R & p \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & -1 & 0 & -1 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Summary of the uses of transformation matrices

(1) represent a configuration (position and orientation)
(2) change the reference frame (of a vector or a frame)
(3) displace a vector or a frame (in its current frame)

Twist (angular and linear velocity)

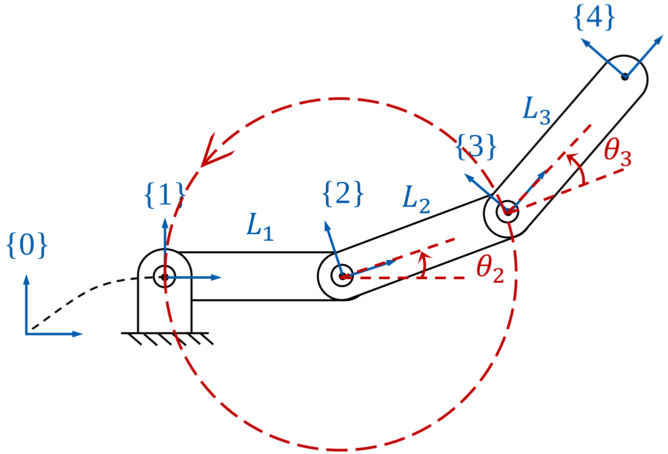
$$\mathcal{V} = \begin{bmatrix} \omega \\ v \end{bmatrix}, \quad \dot{T} = [\mathcal{V}]T$$

Exponential coordinate representation of rigid-body motions

$$e^{[\mathcal{S}]\theta} = \begin{bmatrix} e^{[\omega]\theta} & G(\theta)v \\ 0 & 1 \end{bmatrix} \in SE(3)$$

Forward Kinematics

- $\theta \in \mathbb{R}^n \quad \rightarrow \quad T(\theta) \in SE(3)$
- Given joint coordinates θ, calculate the configuration (position and orientation) $T(\theta)$ of the end-effector frame.



Two ways to solve forward kinematics

- Product of Exponentials
- D-H Parameters

Product of Exponentials

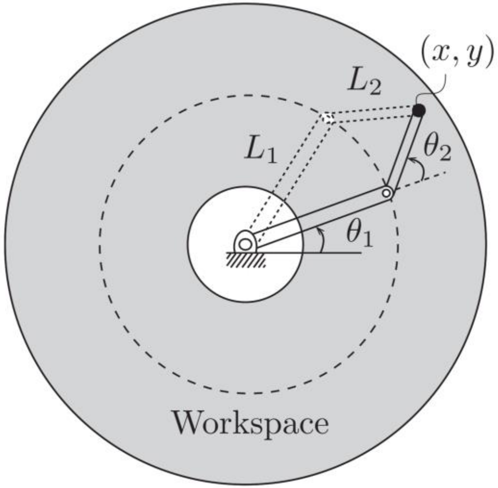
- No dependence on frame assignment
- Find exponentials of screw axes only
- $T(\theta) = e^{[\mathcal{S}_1]\theta_1} \ldots e^{[\mathcal{S}_{n-1}]\theta_{n-1}} e^{[\mathcal{S}_n]\theta_n} M$

D-H Parameters

- A minimal number of parameters associated with a convention for attaching frames
- $T(\theta) = T_{01}(\theta_1) T_{12}(\theta_2) \cdots T_{n-1,n}(\theta_n)$
- $T_{i-1,i} = \mathrm{Rot}(\hat{\mathrm{x}}, \alpha_{i-1})\ \mathrm{Trans}(\hat{\mathrm{x}}, a_{i-1})\ \mathrm{Trans}(\hat{\mathrm{z}}, d_i)\ \mathrm{Rot}(\hat{\mathrm{z}}, \phi_i)$

Lec 6: Inverse Kinematics

Inverse Kinematics
- $T(\theta) \in SE(3) \quad \rightarrow \quad \theta \in \mathbb{R}^n$
- Given a homogeneous transform $X \in SE(3)$ and forward kinematics $T(\theta)$, find solutions θ that satisfy $T(\theta) = X$.



Two ways to solve inverse kinematics
- Analytic Approach (Trigonometry)
- Numerical Approach (Newton's Method)

Analytic Approach (Trigonometry)
- Derive equations based on the geometric structure of the mechanism
- Provide zero, one or multiple solutions

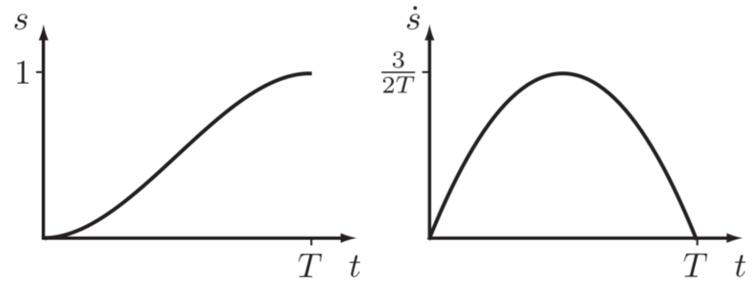
Numerical Approach (Newton's Method)
- Iteratively solve a non-linear optimization problem (require an initial guess)
- Always produce one solution (may not be optimal, best approximation)

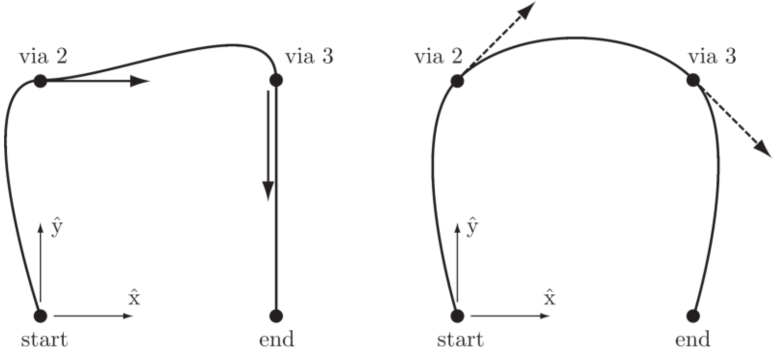
Lec 7: Trajectory Generation

Trajectory Generation
- Path: $\theta(s), s \in [0, 1]$
- Trajectory: $\theta(t), t \in [0, T]$

Polynomial Time Scaling
- Find constraints and solve for coefficients
- 3rd-order polynomials
  $s(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3, t \in [0, T]$
- 5th-order/7th-order polynomials
- Other time scaling methods

Waypoint Navigation
- Add waypoint constraints to trajectory generation





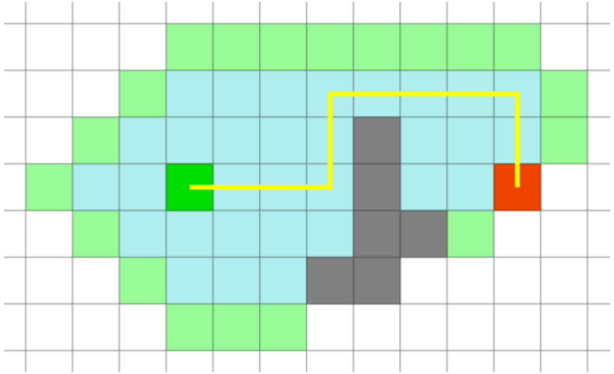
Lec 8: Motion Planning

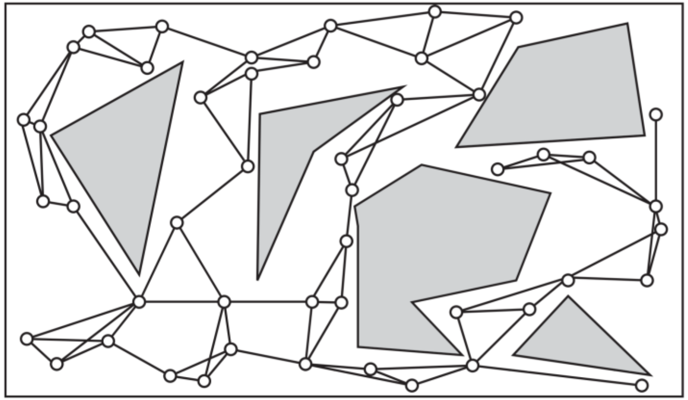
Motion Planning
- How to find a path or trajectory?
- C-space vs. C-obstacle space

Graph Search Algorithms
- A^* algorithm
- Dijkstra algorithm

Sampling-based Algorithms
- Probabilistic Roadmaps (PRM)
- Rapidly-exploring Random Trees (RRT)

Other Planning Algorithms
- Nonlinear optimization
- Virtual potential fields





Outline

1. Introduction
2. Course Outline
3. Course Logistics
4. Homework

Course Logistics

About this course

- Theoretical and practical knowledge for modern Roboticists
  - Lectures and homework build up theoretical/analytical skills
  - Lab assignments build up practical/programming skills
- Equivalent to a graduate or senior undergrad level course in universities
  - The more time and efforts you spend, the more you gain

Expected background

- Engineering mathematics
  - Calculus (linear differential equations)
  - Linear algebra
- Development experience (may be optional)
  - Ubuntu 18 + ROS Melodic (or Ubuntu 20 + ROS Noetic)
  - Python or C++ programming

Course Logistics

Textbook

- “Modern Robotics: Mechanics, Planning, and Control,” by Kevin Lynch and Frank Park
- Website: http://modernrobotics.org
  - Preprint pdf available for free
  - Supplementary videos are also available
- Official Github repository: https://github.com/NxRLab/ModernRobotics
- Please respect the math convention used in the textbook and our lecture slides

Course load

- Lecture videos (1-2 hours/week)
- Homework exercises (2-8 hours/week)
- Lab assignments (2-8 hours/week)
- Capstone project

Course Logistics

Homework

- Mostly math exercises (taken from textbook + customized ones)
- Posted with lecture videos (in a separate PDF file) and due one week after posting
- Please cite any resources/ideas not originated from you

Lab Assignments

- Run simulation in Gazebo (Turtlebot2 mobile robot + ReactorX 150 manipulator)
- A detailed instruction will be posted every week (may include sample code)
- Due one week after posting; please submit in a zip file

Grading

- Will be posted one week after submission
- You have the chance to revise and resubmit until the deadline
- Please ask our course advisor and teaching assistants for more details

Course Logistics

Week	Lecture	Homework	Lab Assignment
1	Overview	/	Easy
2	Configuration Space	Medium	Easy
3	Rigid-Body Motions: Rotation	Hard	Medium
4	Rigid-Body Motions: Transformation	Hard	/
5	Forward Kinematics	Medium	Medium
6	Inverse Kinematics	/	Hard
7	Trajectory Generation	Easy	Medium
8	Motion Planning	Easy	Medium

Outline

1. Introduction
2. Course Outline
3. Course Logistics
4. Homework

Homework

(1) Set up development environment
- A computer or virtual machine with Ubuntu 18/20 operating system
- Install ROS Melodic or Noetic accordingly
- Set up other development tools as needed (e.g., VS Code)

(2) Install Turtlebot2 packages
- https://github.com/UCR-Robotics/Turtlebot2-On-Melodic
- You may need to customize the installation script a bit if you are using Ubuntu 20
- Deliverable: be able to bring up a Turtlebot2 robot, and teleop it using keyboard

(3) Install manipulator packages
- https://github.com/UCR-Robotics/interbotix_ros_arms
- Deliverable: be able to bring up a ReactorX 150 robot arm and play with it

深蓝学院
shenlanxueyuan.com

Thanks for Listening！