

16-782 Fall'25

Planning & Decision-making in Robotics

Introduction;

What is Planning, Role of Planning in Robots

Maxim Likhachev

Robotics Institute

Carnegie Mellon University

Class Logistics

- Instructor:

Maxim Likhachev – maxim@cs.cmu.edu

- TA:

Alvin Zou – azou@andrew.cmu.edu

Barath Satheeskumar – bsathees@andrew.cmu.edu

- Website:

http://www.cs.cmu.edu/~maxim/classes/robotplanning_grad

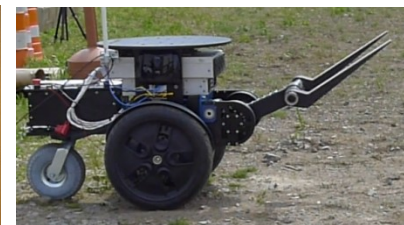
- Announcements, Questions, Recorded Lectures:

- on Piazza

- should have received an email with access info

About Me

- My Research Interests:
 - Planning, Decision-making, Learning
 - Applications: planning for complex robotic systems including aerial and ground robots, manipulation platforms, small teams of heterogeneous robots
- More info: <http://www.cs.cmu.edu/~maxim>
- Search-based Planning Lab: <http://www.sbpl.net>



About Me

- Also, currently split between CMU and [Waymo](#), where I'm heavily involved in planning for self-driving vehicles

Class Objectives at High-level

- Understand and learn how to implement most popular planning and decision-making approaches in robotics
- Understand the challenges and basic approaches to interleaving planning and execution in robotic systems
- Learn common uses of planning/decision-making in robotics
- Get a sense for doing research in the area of planning/decision-making in robotics

What is Planning?

- According to Wikipedia: *“Planning is the process of thinking about an organizing the activities required to achieve a desired goal.”*

What is Planning for Robotics?

- According to Wikipedia: *“Planning is the process of thinking about an organizing the activities required to achieve a desired goal.”*

- **Given**

- model (states and actions) of the robot(s) $M^R = \langle S^R, A^R \rangle$
- a model of the world M^W
- current state of the robot $s_{current}^R$
- current state of the world $s_{current}^W$
- cost function C of robot actions
- desired set of states for robot and world G

- **Compute a plan π that**

- prescribes a set of actions $a_1, \dots, a_K$ in A^R the robot should execute
- reaches one of the desired states in G
- (preferably) minimizes the cumulative cost of executing actions $a_1, \dots, a_K$

Few Examples

- **Given**

- model (states and actions) of the robot(s) $M^R = \langle S^R, A^R \rangle$
- a model of the world M^W
- current state of the robot $s_{current}^R$
- current state of the world $s_{current}^W$
- cost function C of robot actions
- desired set of states for robot and world G

- **Compute a plan π that**

- prescribes a set of actions $a_1, \dots, a_K$ in A^R the robot should execute
- reaches one of the desired states in G
- (preferably) minimizes the cumulative cost of executing actions $a_1, \dots, a_K$

Planning for omnidirectional robot:

What is M^R ?

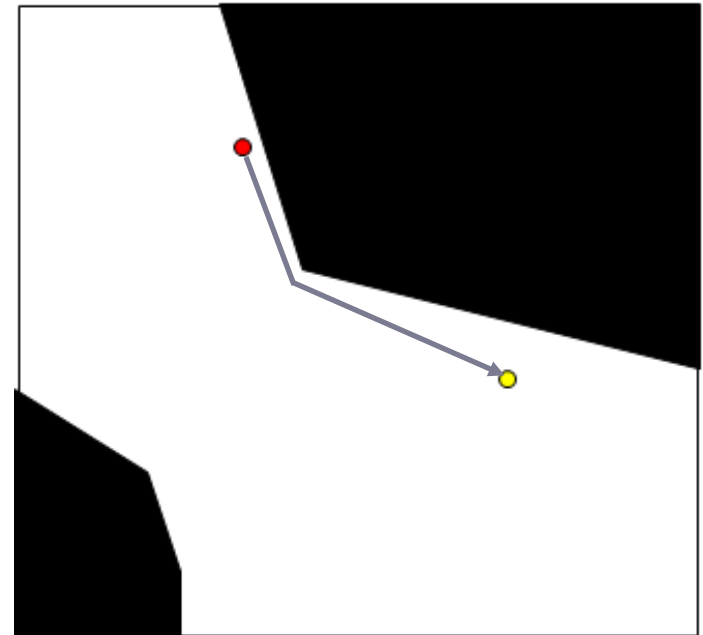
What is M^W ?

What is $s_{current}^R$?

What is $s_{current}^W$?

What is C ?

What is G ?



Few Examples

- **Given**

- model (states and actions) of the robot(s) $M^R = \langle S^R, A^R \rangle$
- a model of the world M^W
- current state of the robot $s_{current}^R$
- current state of the world $s_{current}^W$
- cost function C of robot actions
- desired set of states for robot and world G

- **Compute a plan π that**

- prescribes a set of actions $a_1, \dots, a_K$ in A^R the robot should execute
- reaches one of the desired states in G
- (preferably) minimizes the cumulative cost of executing actions $a_1, \dots, a_K$

Planning for omnidirectional drone:

What is M^R ?

What is M^W ?

What is $s_{current}^R$?

What is $s_{current}^W$?

What is C ?

What is G ?



MacAllister et al., 2013

Few Examples

- **Given**

- model (states and actions) of the robot(s) $M^R = \langle S^R, A^R \rangle$
- a model of the world M^W
- current state of the robot $s_{current}^R$
- current state of the world $s_{current}^W$
- cost function C of robot actions
- desired set of states for robot and world G

- **Compute a plan π that**

- prescribes a set of actions $a_1, \dots, a_K$ in A^R the robot should execute
- reaches one of the desired states in G
- (preferably) minimizes the cumulative cost of executing actions $a_1, \dots, a_K$

Planning for autonomous navigation:

What is M^R ?

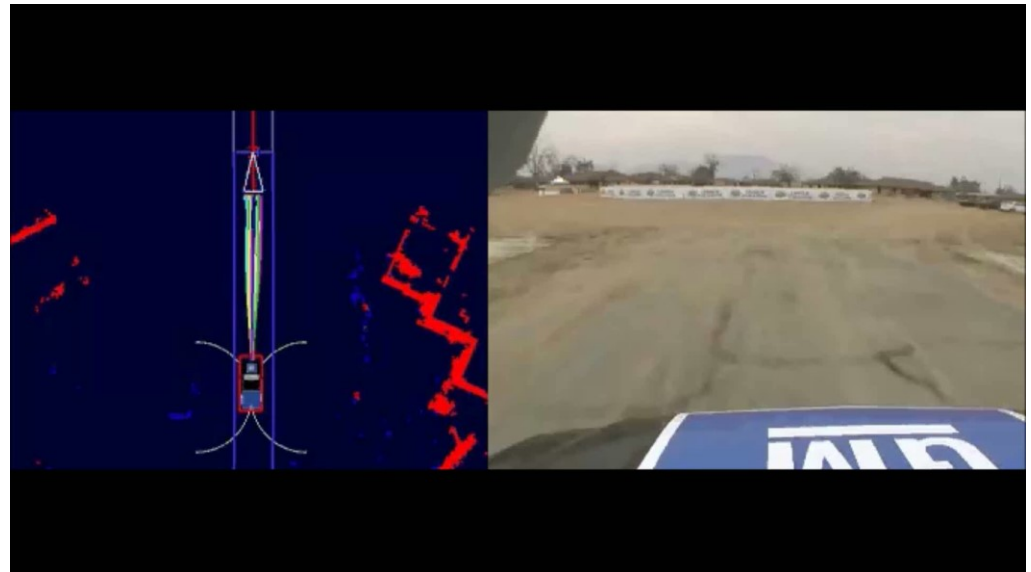
What is M^W ?

What is $s_{current}^R$?

What is $s_{current}^W$?

What is C ?

What is G ?



Likhachev & Ferguson, '09; part of Tartanracing team from CMU for the Urban Challenge 2007 race

Few Examples

- **Given**

- model (states and actions) of the robot(s) $M^R = \langle S^R, A^R \rangle$
- a model of the world M^W
- current state of the robot $s_{current}^R$
- current state of the world $s_{current}^W$
- cost function C of robot actions
- desired set of states for robot and world G

- **Compute a plan π that**

- prescribes a set of actions $a_1, \dots, a_K$ in A^R the robot should execute
- reaches one of the desired states in G
- (preferably) minimizes the cumulative cost of executing actions $a_1, \dots, a_K$

Planning for autonomous flight among people :

Narayanan et al., 2012

What is M^R ?

What is M^W ?

What is $s_{current}^R$?

What is $s_{current}^W$?

What is C ?

What is G ?



Few Examples

- **Given**

- model (states and actions) of the robot(s) $M^R = \langle S^R, A^R \rangle$
- a model of the world M^W
- current state of the robot $s_{current}^R$
- current state of the world $s_{current}^W$
- cost function C of robot actions
- desired set of states for robot and world G

- **Compute a plan π that**

- prescribes a set of actions $a_1, \dots, a_K$ in A^R the robot should execute
- reaches one of the desired states in G
- (preferably) minimizes the cumulative cost of executing actions $a_1, \dots, a_K$

Planning for a mobile manipulator robot opening a door: Gray et al., 2013

What is M^R ?

What is M^W ?

What is $s_{current}^R$?

What is $s_{current}^W$?

What is C ?

What is G ?



Few Examples

- **Given**

- model (states and actions) of the robot(s) $M^R = \langle S^R, A^R \rangle$
- a model of the world M^W
- current state of the robot $s_{current}^R$
- current state of the world $s_{current}^W$
- cost function C of robot actions
- desired set of states for robot and world G

- **Compute a plan π that**

- prescribes a set of actions $a_1, \dots, a_K$ in A^R the robot should execute
- reaches one of the desired states in G
- (preferably) minimizes the cumulative cost of executing actions $a_1, \dots, a_K$

Planning for a mobile manipulator robot assembling a birdcage: Cohen et al., 2015

What is M^R ?

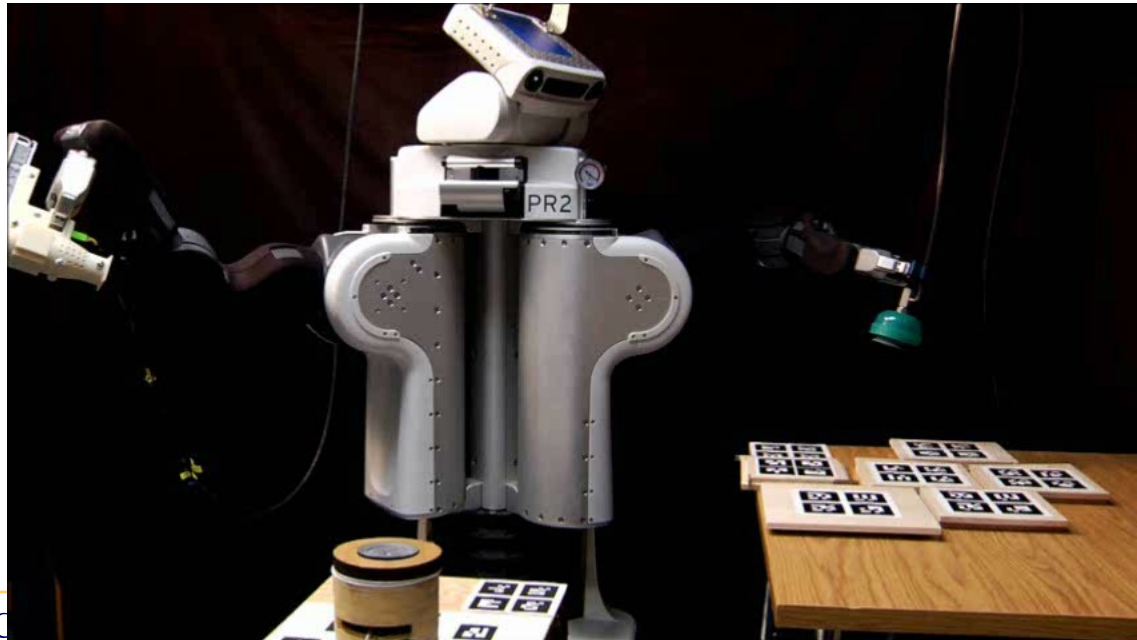
What is M^W ?

What is $s_{current}^R$?

What is $s_{current}^W$?

What is C ?

What is G ?



Few Examples

- **Given**

- model (states and actions) of the robot(s) $M^R = \langle S^R, A^R \rangle$
- a model of the world M^W
- current state of the robot $s_{current}^R$
- current state of the world $s_{current}^W$
- cost function C of robot actions
- desired set of states for robot and world G

- **Compute a plan π that**

- prescribes a set of actions $a_1, \dots, a_K$ in A^R the robot should execute
- reaches one of the desired states in G
- (preferably) minimizes the cumulative cost of executing actions $a_1, \dots, a_K$

Planning/decision-making for a mobile manipulator unloading a truck:

What is M^R ?

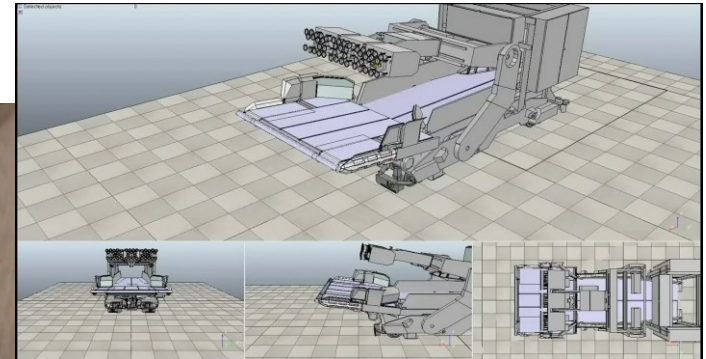
What is M^W ?

What is $s_{current}^R$?

What is $s_{current}^W$?

What is C ?

What is G ?



Assuming Infinite Computational Resources...

Where does Planning break?

Assuming Infinite Computational Resources...

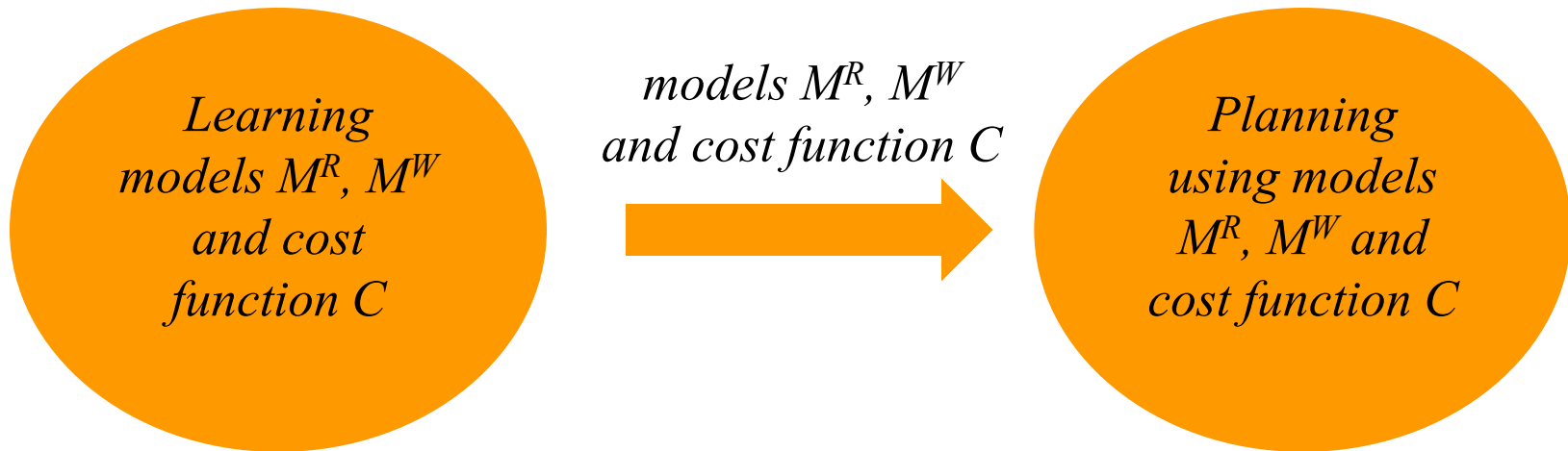
Where does Planning break?

Reliance on the knowledge/accuracy of the model!

Role of Learning in Planning?

Planning vs. Learning

Model-based approach



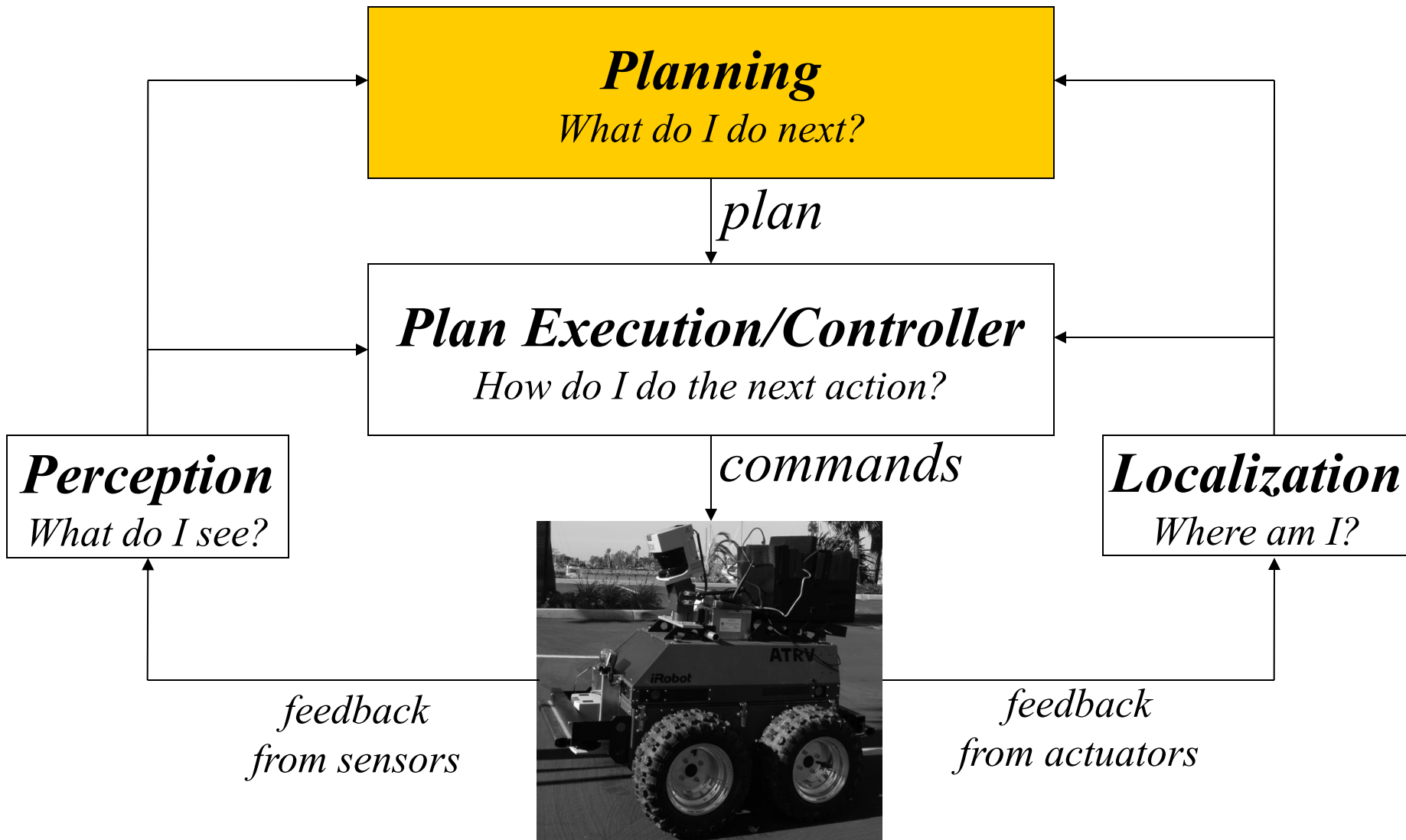
Model-free approach

Learning the mapping from “what robot sees” onto “what to do next” using rewards received by the robot (Reinforcement Learning) or demonstrations (Behavior Cloning)

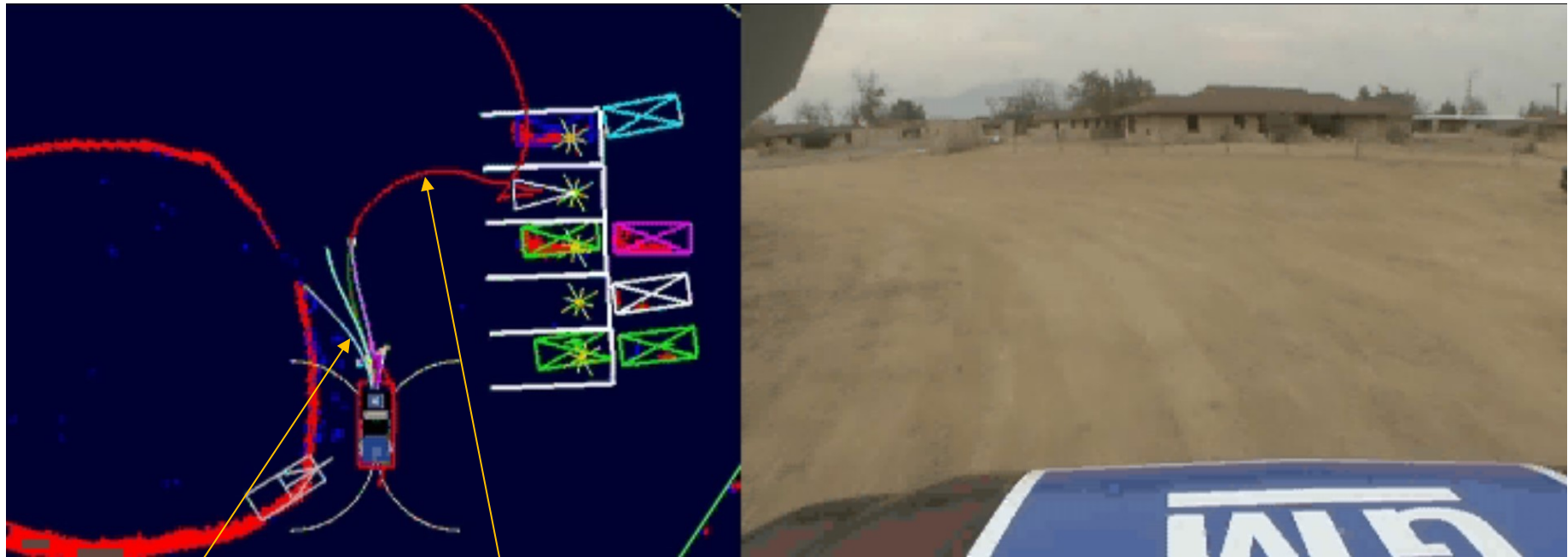
For those interested ...

- A new class called **Interleaved Planning and Learning**
 - will be offered in Spring'26
 - graduate version
 - focuses on studying the latest approaches that
 - *combine* model-based planning with modern (e.g., deep) ML
 - apply to the control of single- and multi- robot systems
 - semester-long research project (no homeworks, no exams, no pop-quizzes), with publication-level paper as outcome
 - _prerequisite: *strong knowledge in planning* (e.g., this class) and ML

Planning within a Typical Autonomy Architecture

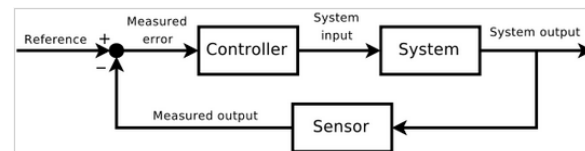


Planning vs. Trajectory Following vs. Control

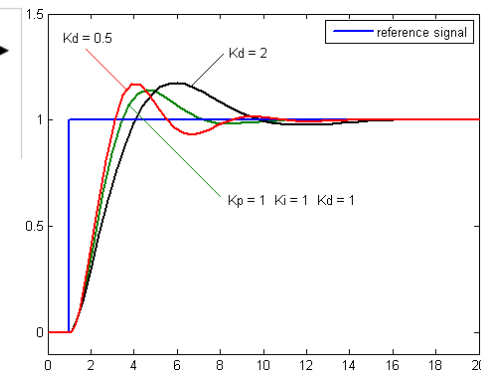


*local planning
(trajectory following)*

global planning



controller



Images from wikipedia

Class Logistics

- Books (optional):
 - Planning Algorithms *by Steven M. LaValle*
 - Heuristic Search, Theory and Applications *by Stefan Edelkamp and Stefan Schroedl*
 - Principles of Robot Motion, Theory, Algorithms, and Implementations *by Howie Choset, Kevin M. Lynch, Seth Hutchinson, George A. Kantor, Wolfram Burgard, Lydia E. Kavraki and Sebastian Thrun*
 - Artificial Intelligence: A Modern Approach *by Stuart Russell and Peter Norvig*

Class Prerequisites

- Knowledge of programming (e.g., C, C++)
- Working knowledge of data structures & basic Computer Science algorithms (e.g., graphs, linked lists, priority queues, BFS/DFS, etc.)
- Prior exposure to robotics

Class Objectives

- Understand and learn how to implement most popular planning algorithms in robotics including heuristic search-based planning algorithms, sampling-based planning algorithms, task planning, planning under uncertainty and multi-robot planning
- Learn basic principles behind the design of planning representations
- Understand core theoretical principles that many planning algorithms rely on and learn how to analyze theoretical properties of the algorithms
- Understand the challenges and basic approaches to interleaving planning and execution in robotic systems
- Learn common uses of planning in robotics
- Get a sense for doing research in the area of planning/decision-making in robotics

Tentative Class Schedule

<i>TENTATIVE SCHEDULE FOR Planning and Decision-making in Robotics CLASS</i>				
<i>Fall 2025</i>				
Date	Day	Topic	HW out	HW due
25-Aug	Mon	Introduction; What is Planning?		
27-Aug	Wed	planning representations: explicit vs. implicit graphs, skeletonization, cell decomposition & lattice-based graphs		
1-Sep	Mon	LABOR DAY - NO CLASS		
3-Sep	Wed	search algorithms: A*, Multi-goal A*, Weighted A*, Backward A*	HW1	
8-Sep	Mon	search algorithms: Heuristic functions, Multi-Heuristic A*		
10-Sep	Wed	interleaving planning and execution: Anytime heuristic search, Incremental heuristic search		
15-Sep	Mon	interleaving planning and execution: Real-time heuristic Search		
17-Sep	Wed	case study: planning for autonomous driving		
22-Sep	Mon	planning representations: PRM for continuous spaces		HW1
24-Sep	Wed	planning representations/search algorithms: RRT, RRT-Connect, RRT*	HW2	
29-Sep	Mon	case study: planning for mobile manipulators and legged robots		
1-Oct	Wed	search algorithms: Markov Property, dependent vs. independent variables, Dominance		
6-Oct	Mon	case study: planning for coverage, mapping and surveillance tasks		
8-Oct	Wed	planning representations: state-space vs. symbolic representation for task planning		HW2
13-Oct	Mon	FALL BREAK - NO CLASS		
15-Oct	Wed	FALL BREAK - NO CLASS		
20-Oct	Mon	search algorithms: planning on symbolic representations	HW3	
22-Oct	Wed	planning under uncertainty: Minimax formulation, Minimax Backward A*		
27-Oct	Mon	planning under uncertainty: Markov Decision Processes, Value Iteration, RTDP		
29-Oct	Wed	final project proposal presentations		
3-Nov	Mon	planning under uncertainty: Markov Decision Processes, Value Iteration, RTDP (cont'd)		HW3
5-Nov	Wed	planning under uncertainty: Partially-Observable Markov Decision Processes		
10-Nov	Mon	planning under uncertainty: Partially-Observable Markov Decision Processes (cont'd)		
12-Nov	Wed	multi-robot planning		
17-Nov	Mon	multi-robot planning (cont'd)		
19-Nov	Wed	TBD		
24-Nov	Mon	exam		
26-Nov	Wed	THANKSGIVING - NO CLASS		
1-Dec	Mon	learning in planning		
3-Dec	Wed	final project presentations		

Three Homeworks + Final Project

- All homeworks are individual (no groups)
- Final projects is a group project (3-5 people per group)
- Homeworks are programming assignments based on the material
- Final project is a research-like project
 - For example: to develop and implement a planner for a robot planning problem of your choice
 - Or: to extend a particular planning algorithm to improve its running time or to handle additional conditions
 - Two presentations (proposal and final) and meetings with groups

Class Structure

- Grading

Three homeworks	33%
Exam	20%
In-class pop quizzes	10%
Final project	32%
Participation	5%

- Exam is tentatively scheduled for Nov. 24
- Late Policy
 - 3 free late days
 - No late days may be used for the final project!
 - Each additional late day will incur a 10% penalty

Questions about the class?