MENG 3065 - MODULE 7

Artificial Intelligence: A Modern Approach
Chapter 26 Robotics

HUMBER

WE ARE HUMBER

Outline

- Robots
- Robot Hardware
- What kind of problem is robotics solving?
- Robotic Perception
- Planning and Control
- Planning Uncertain Movements
- Reinforcement Learning in Robotics
- Humans and Robots
- Application Domains

Robots

- physical agents that perform tasks by manipulating the physical world
- equipped with effectors such as legs, wheels, joints, and grippers
- equipped with sensors, which enable them to perceive their environment
- Maximizing expected utility for a robot means choosing how to actuate its effectors to assert the right physical forces
- partially observable and stochastic
- usually model their environment with a continuous state space
- Robotic learning is constrained because the real world operates at real time

Robot Hardware

Types of robots from the hardware perspective

- **Manipulators**: just robot arms
  - Large payload (assembling cars)
  - Wheelchair mountable arms
- **Mobile robots** are those that use wheels, legs, or rotors to move about the environment
  - Quadcopter drones are a type of unmanned aerial vehicle (UAV)

Other kinds of robots include prostheses, exoskeletons, robots with wings, swarms, and intelligent environments in which the robot is the entire room.

Robot Hardware

(a)

(b)

(a) An industrial robotic arm with a custom end-effector. Image credit: Macor/123RF.
(b) Kinova® JACO® Assistive Robot arm mounted on a wheelchair. Kinova and JACO are trademarks of Kinova, Inc.

Robot Hardware

(a) (b)

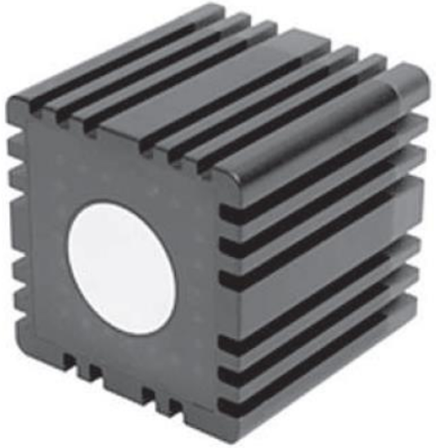
(a) NASA’s Curiosity rover taking a selfie on Mars. Image courtesy of NASA.
(b) A Skydio drone accompanying a family on a bike ride. Image courtesy of Skydio.

Robot Hardware

Sensing the world

- **Passive sensors**, such as cameras, are true observers of the environment: they capture signals that are generated by other sources in the environment
- **Active sensors**, such as sonar, send energy into the environment
- **Range finders** are sensors that measure the distance to nearby object
- **Stereo vision** relies on multiple cameras to image the environment from slightly different viewpoints, analyzing the resulting parallax in these images to compute the range of surrounding objects
- Autonomous cars often use scanning lidars (short for light detection and ranging): active sensors that emit laser beams and sense reflected beam
- Radar is often the range finding sensor of choice for air vehicles

Robot Hardware



(a)

(b)

(a) Time-of-flight camera; image courtesy of Mesa Imaging GmbH.
(b) 3D range image obtained with this camera. The range image makes it possible to detect obstacles and objects in a robot's vicinity. Image courtesy of Willow Garage, LLC.

Robot Hardware

Producing motion

- **Actuator:** The mechanism that initiates the motion of an effector
- **Electric actuator** uses electricity to spin up a motor
- **Hydraulic actuators** use pressurized hydraulic fluid (like oil or water)
- **Pneumatic actuators** use compressed air
- Actuators are often used to move joints, which connect rigid bodies (links)
- In revolute joints, one link rotates with respect to the other. In prismatic joints, one link slides along the other.

What kind of problem is robotics solving?

Robotics use a three-level hierarchy:

- The **task planning level** decides a plan or policy for high-level actions
- Then **motion planning** is in charge of finding a path that gets the robot from one point to another, achieving each subgoal
- **Control** is used to achieve the planned motion using the robot’s actuators

Preference learning is in charge of estimating an end user’s objective, and people prediction is used to forecast the actions of other people in the robot’s environment

Robotic Perception

Perception is the process by which robots map sensor measurements into internal representations of the environment

Good internal representations for robots have three properties:

1. They contain enough information for the robot to make good decisions.
2. They are structured so that they can be updated efficiently.
3. They are natural in the sense that internal variables correspond to natural state variables in the physical world.

The posterior $\mathbf{P}(\mathbf{X}_t\ \mathbf{z}_{1:t},\ a_{1:t-1})$ is the sensor model, a probability distribution over all states that captures what we know from past sensor measurements and controls.

$$\begin{aligned}&\mathbf{P}(\mathbf{X}_{t+1} \mid \mathbf{z}_{1:t+1}, a_{1:t})\\&= \alpha \mathbf{P}(\mathbf{z}_{t+1} \mid \mathbf{X}_{t+1}) \int \mathbf{P}(\mathbf{X}_{t+1} \mid \mathbf{x}_t, a_t)\, P(\mathbf{x}_t \mid \mathbf{z}_{1:t}, a_{1:t-1})\, d\mathbf{x}_t\,.\end{aligned}$$

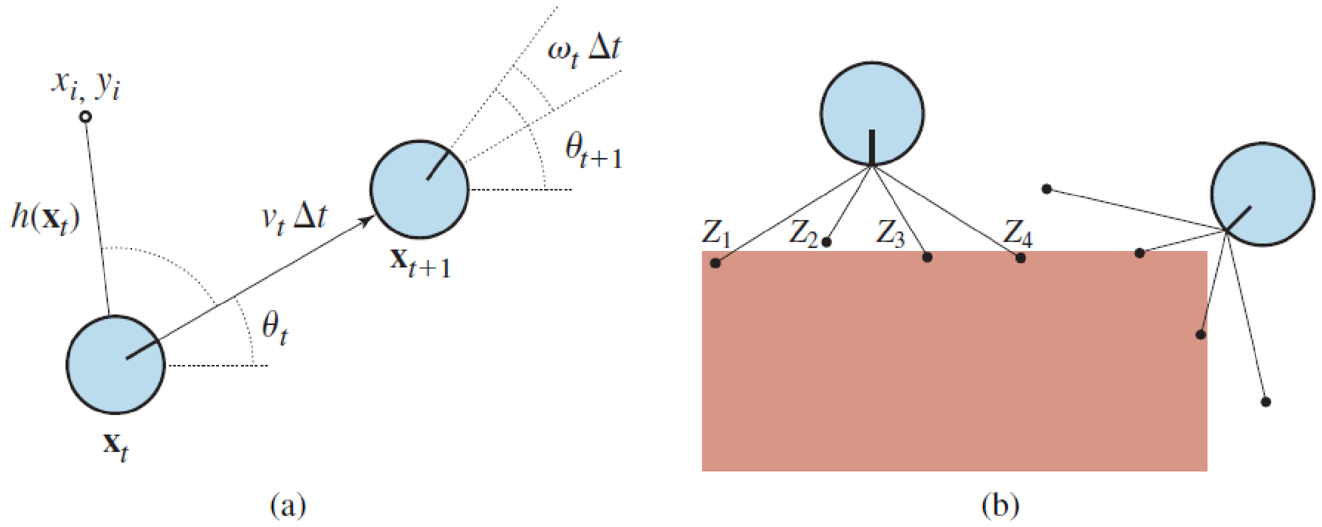
$\boldsymbol{P}(\boldsymbol{X}_{t+1} \mid \mathbf{x}_t,\ a_t)$ is called *the* **transition model** or **motion model**,

Robotic Perception

Localization and mapping

- Localization is the problem of finding out where things are—including the robot itself. Localization using particle filtering is called Monte Carlo localization, or MCL
- MCL algorithm is an instance of the particle-filtering algorithm
- The Kalman filter is the other major way to localize. A Kalman filter represents the posterior $\mathbf{P}(\mathbf{X}_t \mid \mathbf{z}_{1:t}, a_{1:t-1})$ by a Gaussian. A Kalman filter that linearizes via Taylor expansion is called an extended Kalman filter (or EKF).
- The problem of needing to know the identity of landmarks is an instance of the data association problem
- In some situations, no map of the environment is available, this problem has been studied extensively under the name simultaneous localization and mapping, abbreviated as SLAM.
- SLAM problems are solved using many different probabilistic techniques, including the extended Kalman filter

Robotic Perception



(a) A simplified kinematic model of a mobile robot. The robot is shown as a circle with an interior radius line marking the forward direction. The state $\mathbf{x}_t$ consists of the (x_t, y_t) position (shown implicitly) and the orientation θ_t. The new state $\mathbf{x}_{t+1}$ is obtained by an update in position of $v_t\Delta_t$ and in orientation of $\omega_t\Delta_t$. Also shown is a landmark at (x_i, y_i) observed at time t.

(b) The range-scan sensor model. Two possible robot poses are shown for a given range scan (z_1, z_2, z_3, z_4). It is much more likely that the pose on the left generated the range scan than the pose on the right.

Robotic Perception

Other types of perception

- Robots also perceive temperature, odors, sound, and so on
- Can be estimated using variants of dynamic Bayes networks
- Required for such estimators are conditional probability distributions that characterize the evolution of state variables over time, and sensor models that describe the relation of measurements to state variables.
- Trend in robotics is growing towards representations with well-defined semantics
- Probabilistic techniques, statistical techniques and simpler solutions
- To help decide which approach to take, experience working with real physical robots is your best teacher..

Robotic Perception

Monte Carlo localization, a particle filtering algorithm for mobile robot localization

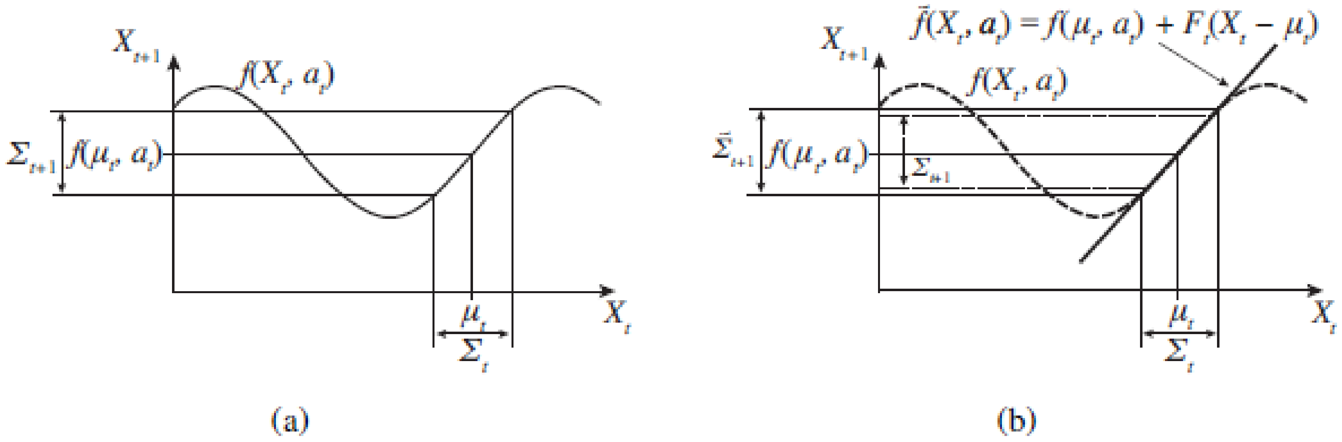
(a)Initial, global uncertainty.
(b)Approximately bimodal uncertainty after navigating in the (symmetric) corridor.
(c)Unimodal uncertainty after entering a room and finding it to be distinctive

WE ARE HUMBER

Robotic Perception

1. **Initial, Global Uncertainty:**
   1. When the robot starts its journey or enters an unfamiliar environment, there is a high level of uncertainty about its precise location.
   2. The algorithm uses a large number of particles (representative points in the space) to cover the entire map, reflecting this initial uncertainty.
2. **Approximately Bimodal Uncertainty After Navigating in the (Symmetric) Corridor:**
   1. As the robot navigates through a symmetric corridor, the uncertainty may decrease, but due to the symmetry, there might be multiple plausible locations.
   2. The probability distribution may become bimodal, indicating that there are two or more possible positions with relatively high likelihoods.
3. **Unimodal Uncertainty After Entering a Room and Finding It to be Distinctive:**
   1. Upon entering a distinctive room or environment, the sensor measurements are likely to provide more distinct information.
   2. As a result, the probability distribution becomes more focused and unimodal, indicating that the robot has a clearer idea of its location.
   3. The algorithm adjusts the particles, converging towards a single location that aligns with the observed features in the room.

Robotic Perception



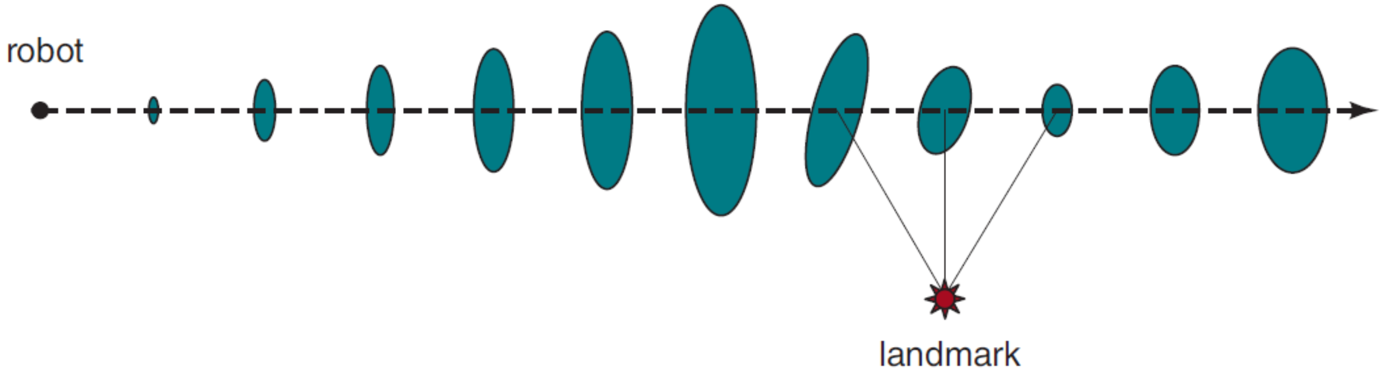
One-dimensional illustration of a linearized motion model:

(a) The function f, and the projection of a mean μ_t and a covariance interval (based on Σt) into time $t+1$.

(b) The linearized version is the tangent of f at μ_t. The projection of the mean μ_t is correct. However, the projected covariance $\tilde{\Sigma}_{t+1}$ differs from Σ_{t+1}.

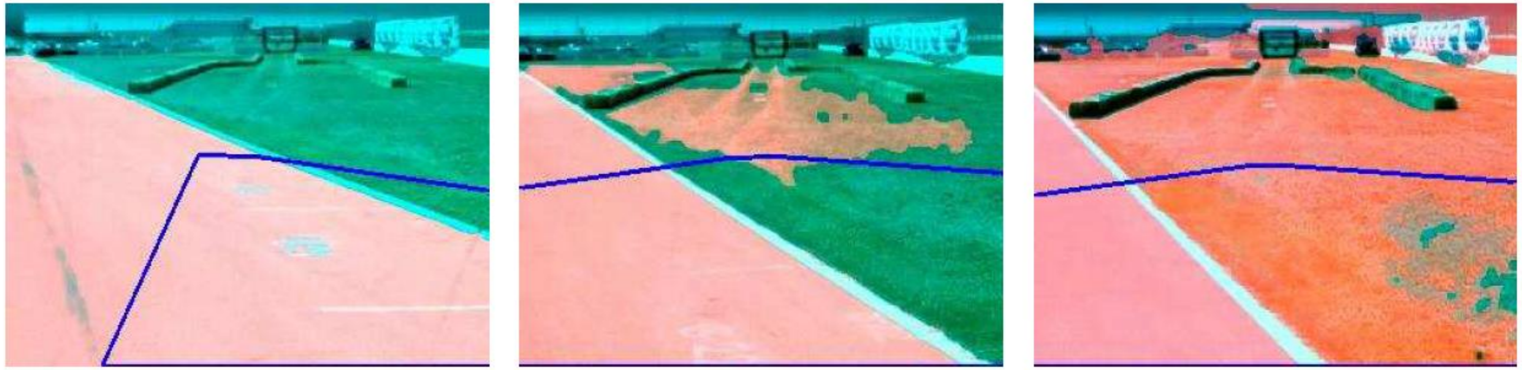
WE ARE HUMBER

Robotic Perception



Localization using the extended Kalman filter. The robot moves on a straight line. As it progresses, its uncertainty in its location estimate increases, as illustrated by the error ellipses. When it observes a landmark with known position, the uncertainty is reduced

Robotic Perception



(a) (b) (c)

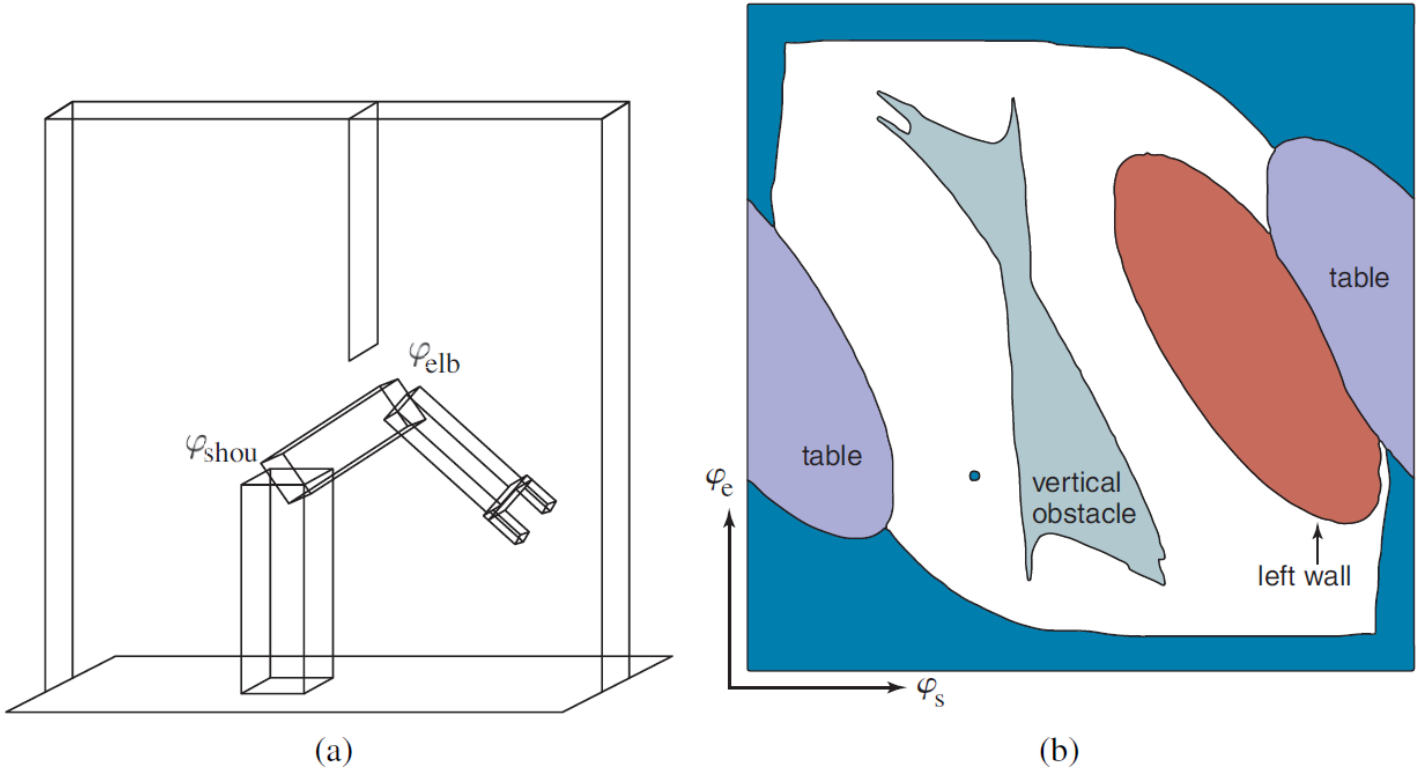
Sequence of “drivable surface” classifications using adaptive vision.
(a) Only the road is classified as drivable (pink area). The V-shaped blue line shows where the vehicle is heading.
(b) The vehicle is commanded to drive off the road, and the classifier is beginning to classify some of the grass as drivable.
(c) The vehicle has updated its model of drivable surfaces to correspond to grass as well as road. Courtesy of Sebastian Thrun

Planning and Control

- **Path** as a sequence of points in geometric space that a robot (or a robot part, such as an arm) will follow
- The task of finding a good path is called **motion planning**
- Once we have a path, the task of executing a sequence of actions to follow the path is called **trajectory tracking control**
- **Trajectory** is a path that has a time associated with each point on the path
- Configuration space
- The physical space that a robot moves about in is called the **workspace**
- **Configuration space**, or C-space: representation scheme in which all the points
- that comprise the robot are represented as a single point in an abstract multidimensional space

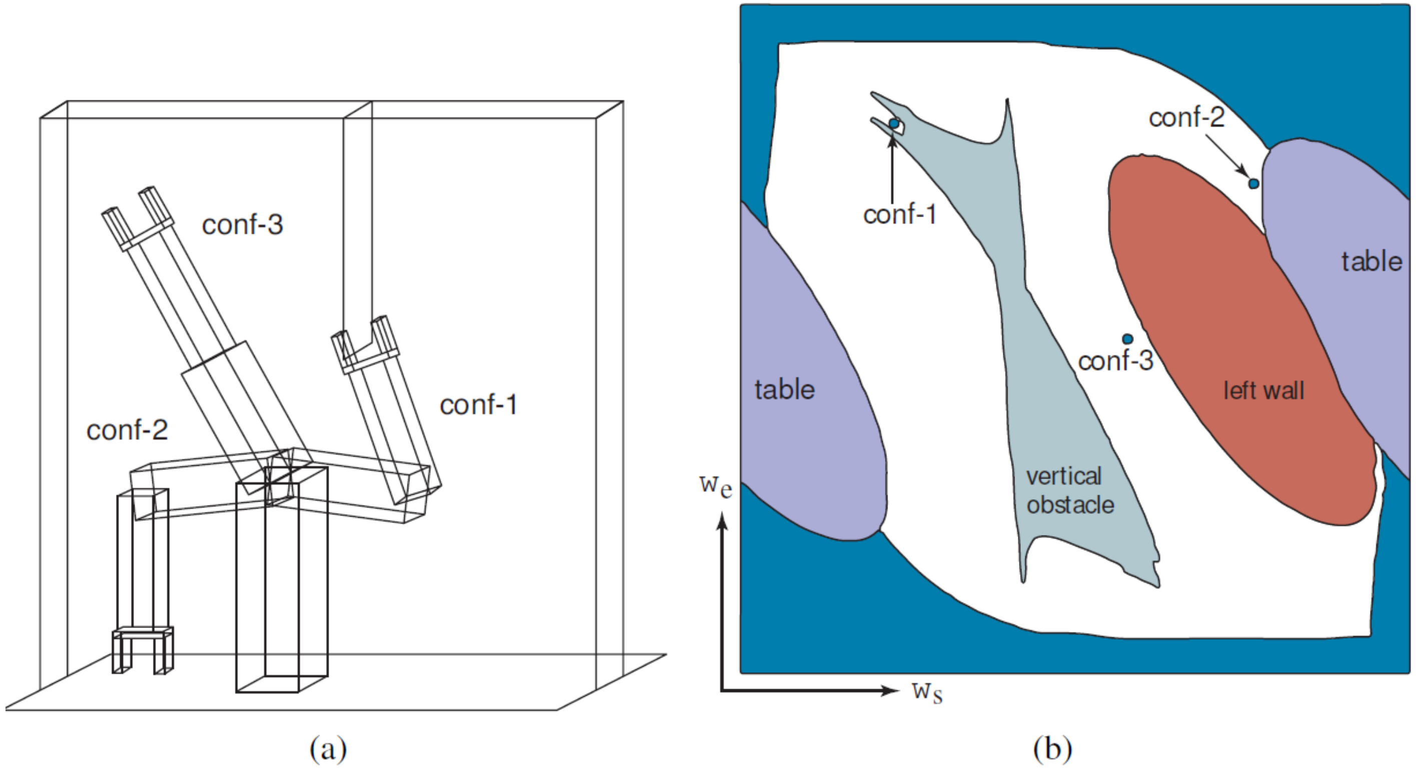
WE ARE HUMBER

Planning and Control



(a) Workspace representation of a robot arm with two degrees of freedom. The workspace is a box with a flat obstacle hanging from the ceiling.
(b) Configuration space of the same robot. Only white regions in the space are configurations that are free of collisions. The dot in this diagram corresponds to the configuration of the robot shown on the left.

Planning and Control



Three robot configurations, shown in workspace and configuration space.

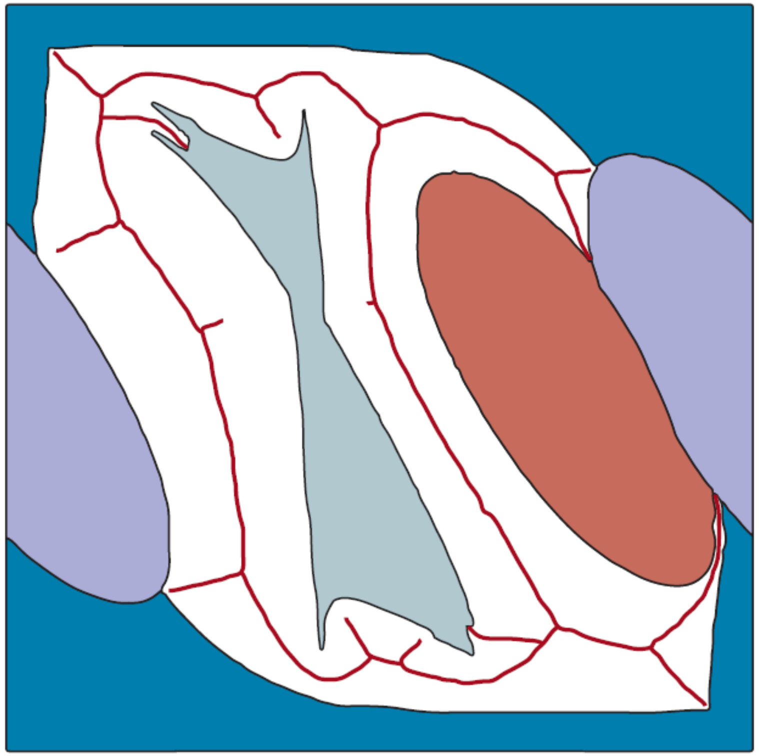
WE ARE HUMBER

Planning and Control

Motion planning

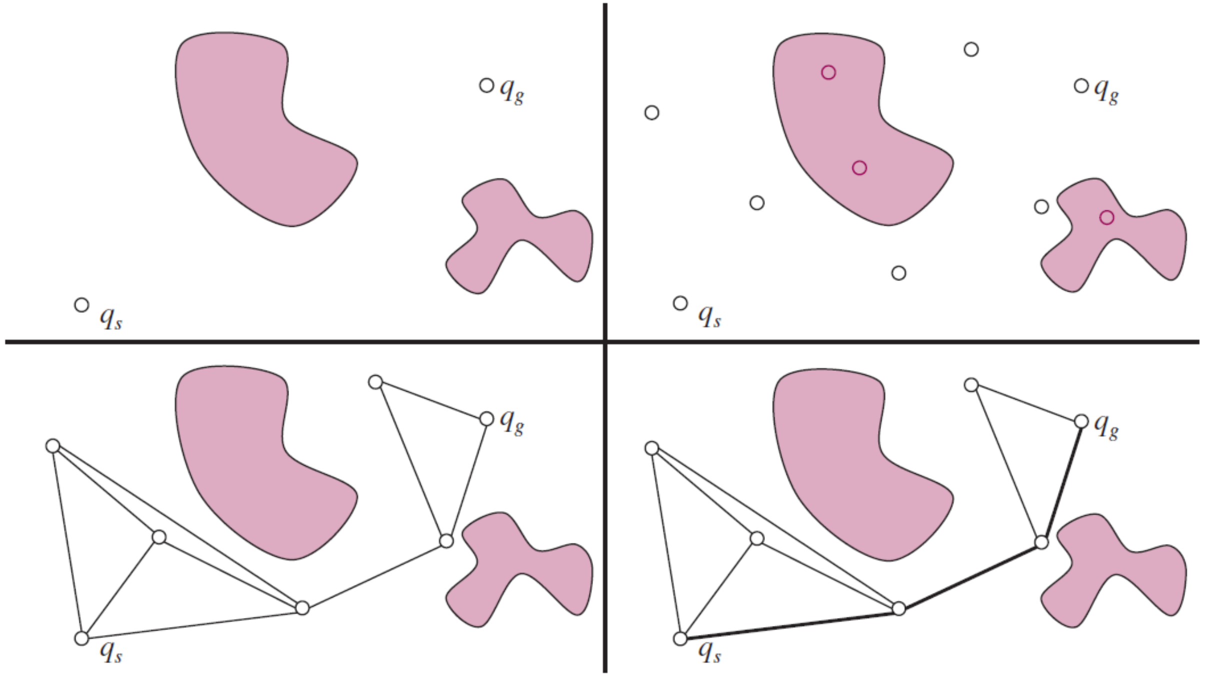
- quintessentially continuous-state search problem & sometimes referred to as the piano mover's problem
- Once we have a path, the task of executing a sequence of actions to follow the as a single point in an abstract multidimensional space
  - a workspace *world* W in either $\mathbb{R}^2$ for the plane or $\mathbb{R}^3$ for three dimensions,
  - an *obstacle region* $O \subset W$,
  - a robot with a configuration space C and set of points $A(q)$ for $q \in C$,
  - a starting configuration $q_s \in C$, and
  - a goal configuration $q_g \in C$.
- some ways of solving the motion planning problem
  - Visibility graphs
  - Voronoi diagrams
  - Cell decomposition
  - Randomized motion planning
  - Rapidly-exploring random trees (RRT)
  - Trajectory optimization for kinematic planning

Planning and Control



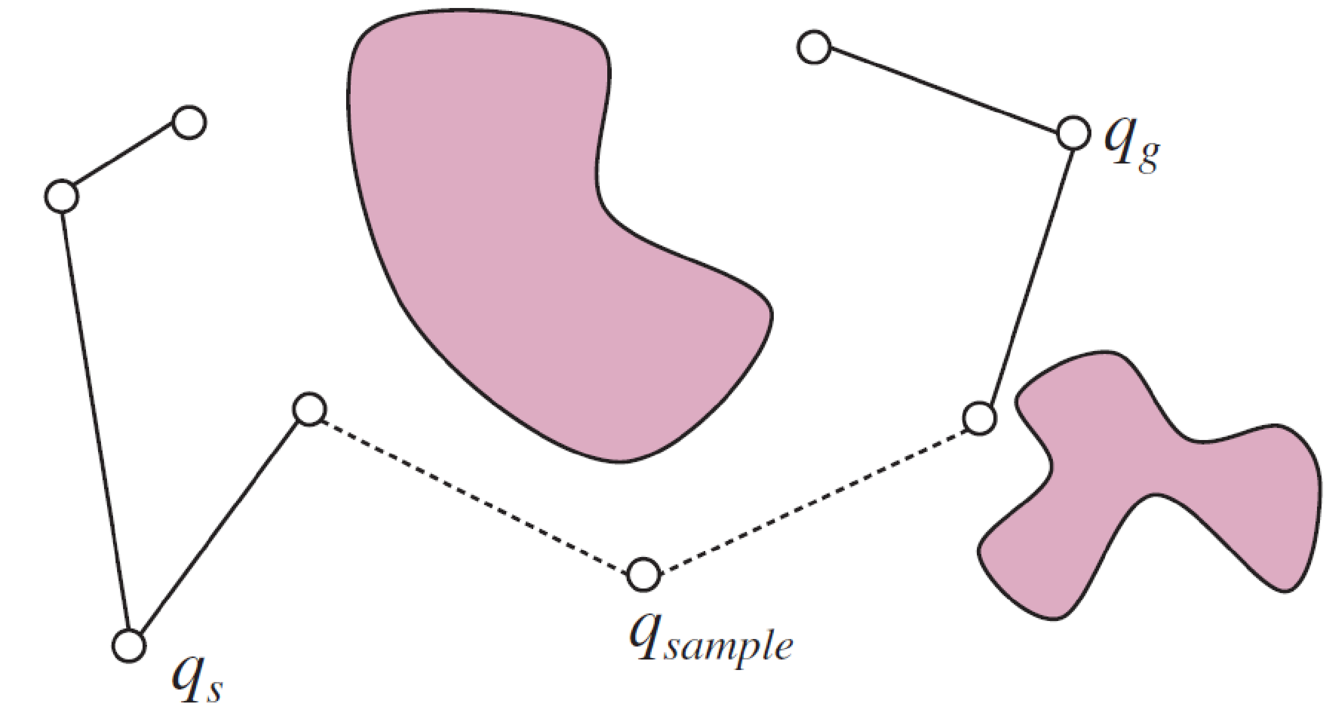
A Voronoi diagram showing the set of points (black lines) equidistant to two or more obstacles in configuration space

Planning and Control



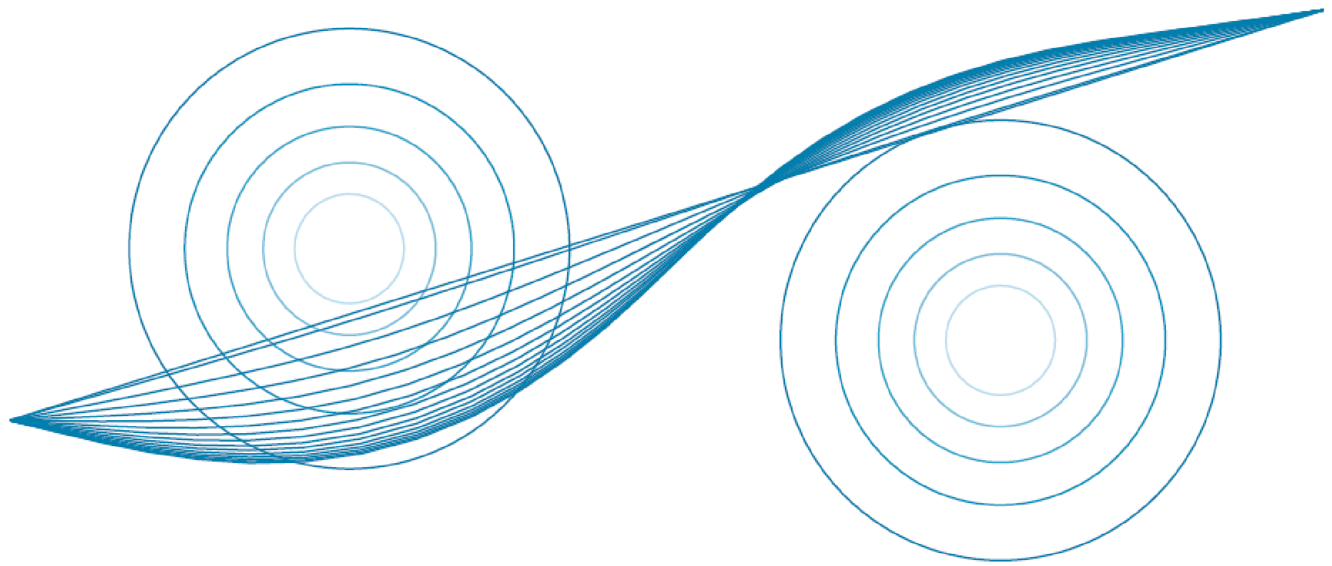
The **bidirectional RRT algorithm** constructs two trees (one from the start, the other from the goal) by incrementally connecting each sample to the closest node in each tree, if the connection is possible. When a sample connects to both trees, that means we have found a solution path.

Planning and Control



The probabilistic roadmap (PRM) algorithm. **Top left**: the start and goal configurations. **Top right**: sample M collision-free milestones (here $M = 5$). **Bottom left**: con- nect each milestone to its k nearest neighbors (here $k = 3$). **Bottom right**: find the shortest art to the goal on the resulting graph.

Planning and Control



Trajectory optimization for motion planning. Two point-obstacles with circular bands of decreasing cost around them. The optimizer starts with the straight line trajectory, and lets the obstacles bend the line away from collisions, finding the minimum path through the cost field.

Planning and Control

Trajectory tracking control

- **From configurations to torques for open-loop tracking**: Our path $\tau(t)$ gives us configurations. The robot starts at rest at $q_s = \tau(0)$. From there the robot's motors will turn currents
- The robot a function f that computes the effects torques have on the configuration.

- If the robot is at configuration q and velocity $\dot{q}$, and applied torque u, that would lead to acceleration $\ddot{q} = f(q, \dot{q}, u)$.
- f^{-1} is the **inverse dynamics**, telling us what torque to apply if we want a particular acceleration, which leads to a change in velocity and thus a change in dynamic state.

- The path τ was created as a sequence of points,

$$u(t) = f^{-1}(\xi(t), \dot{\xi}(t), \ddot{\xi}(t))$$

- A controller that provides force in negative proportion to the observed error is known as a proportional controller or P controller for short. The equation for the force is:

$$u(t) = K_P(\xi(t) - q_t)$$

WE ARE HUMBER

Planning and Control

Trajectory tracking control

- A controller is said to be **stable** if small perturbations lead to a bounded error between the robot and the reference signal
- **strictly stable** if it is able to return to and then stay on its reference path upon such perturbation
- **PD controller**: The letter ‘P’ stands again for proportional, and ‘D’ stands for derivative

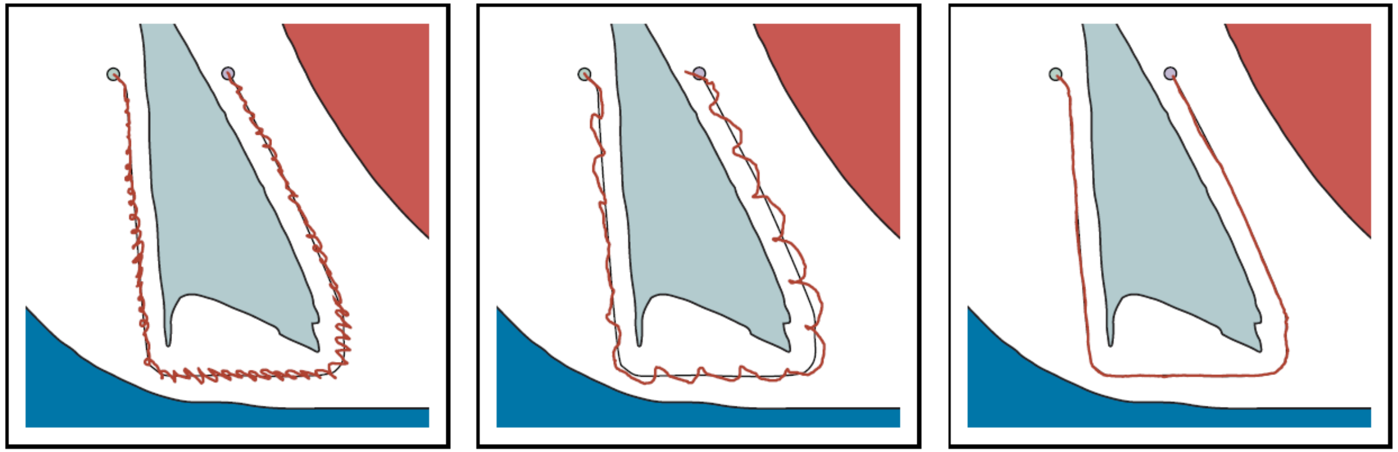
$$u(t) \;=\; K_P(\xi(t) - q_t) + K_D(\dot{\xi}(t) - \dot{q}_t)\ .$$

PID controller (for proportional integral derivative)

computed torque control:

$$u(t) \;=\; \underbrace{f^{-1}(\xi(t),\dot{\xi}(t),\ddot{\xi}(t))}_{\textit{feedforward}} + \underbrace{m(\xi(t))\left(K_P(\xi(t)-q_t)+K_D(\dot{\xi}(t)-\dot{q}_t)\right)}_{\textit{feedback}}\ .$$

Planning and Control



(a) (b) (c)

Robot arm control using (a) proportional control with gain factor 1.0, (b) proportional control with gain factor 0.1, and (c) PD (proportional derivative) control with gain factors 0.3 for the proportional component and 0.8 for the differential component. In all cases the robot arm tries to follow the smooth line path, but in (a) and (b) deviates substantially from the path.

Planning Uncertain Movements

- Robots deal with the continuous state space by turning it into a discrete space
- Robots deal with uncertainty in the current state by choosing the most likely state from the probability distribution produced by the state estimation algorithm

- instead of deterministic plans, uncertainty calls for policies
  - online replanning
  - model predictive control (MPC): plan for a shorter time horizon, but replan at every time step

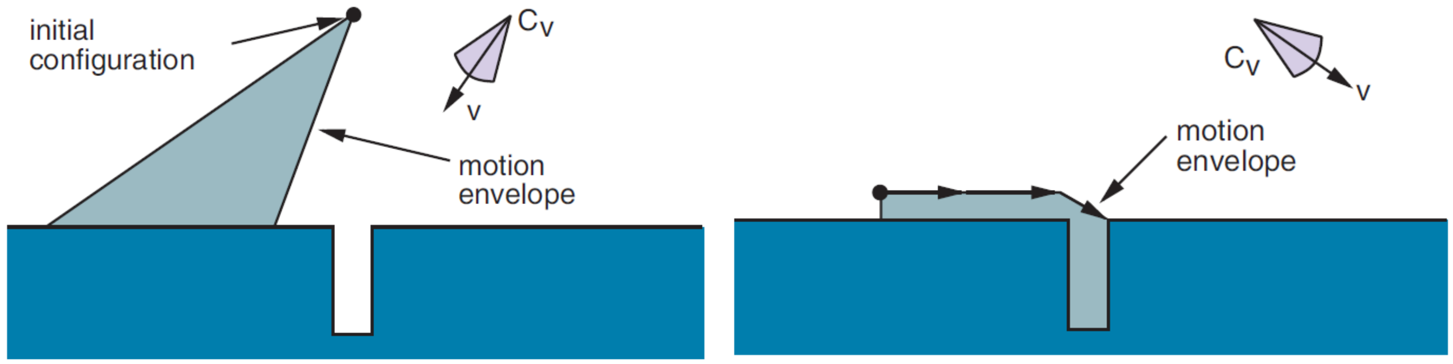
- uncertainty calls for information gathering actions
- better captured by the POMDP framework

- Guarded movements: explicitly defined information gathering actions
  - Each guarded motion consists of (1) a motion command and (2) a termination condition, which is a predicate on the robot’s sensor values saying when to stop

Planning Uncertain Movements

Techniques beyond guarded movements

- change the cost function to incentivize actions we know will lead to information
- coastal navigation heuristic: requires the robot to stay near known landmarks
- incorporate the expected information gain (reduction of entropy of the belief) as a term in the cost function



The first motion command and the resulting envelope of possible robot motions.

The second motion command and the envelope of possible motions. Even with error, we will eventually get into the hole.

WE ARE HUMBER

Reinforcement Learning in Robotics

- One challenge of RL in robotics is the continuous nature of the state and action spaces

- boils down to how we might reduce the real world sample complexity
  - the number of interactions with the physical world that the robot needs before it has learned how to do the task.
- model-based reinforcement learning
- sim-to-real transfer: transferring policies that work in simulation to real world
  - Use model as a simulator for policy search

End-to-end learning:

- More recent techniques have experimented with fitting local models, planning with them to generate actions, and using these actions as supervision to fit a policy, then iterating to get better and better models around the areas that the policy needs.

Exploiting other information

- use higher-level motion primitive
- metalearning or transfer learning

Humans and Robots

Coordination problem: optimizing reward when there are people acting in the same environment as the robot.

- When the human and the robot are on the same team, this turns into **collaboration**
- optimizing for what people actually want.

Coordination

- formulate coordination with a human is to model it as a game between the robot and the human
  - state of the environment (robot & human), $x = (x_R, x_H)$;
  - each agent can take actions, u_R and u_H respectively;
  - each agent has an objective that can be represented as a cost, J_R and J_H:
  - each objective depends on the state and on the actions of *both* agents: $J_R(x, u_R, u_H)$ and $J_H(x, u_H, u_R)$.
- Three important aspects complicate this game:
  - **Incomplete information game** don't know each other's objective
  - state and action spaces are continuous
  - human's behavior might not always be well-characterized as a solution to the game

Humans and Robots

Predicting human action

- The robot can create a model for $P(u_H \mid x, J_H)$, for instance using the softmax

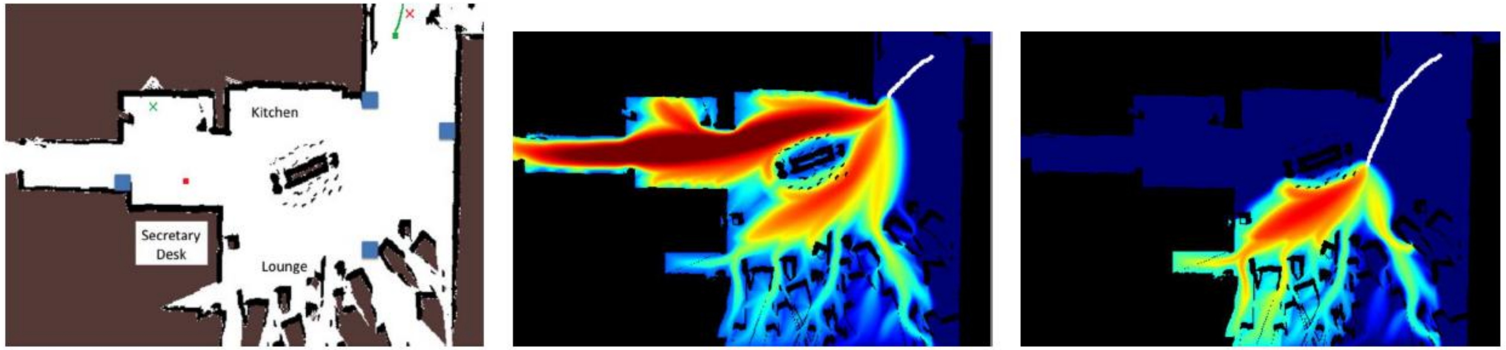
$$P(u_H \mid x, J_H) \propto e^{-Q(x, u_H; J_H)}$$

- $Q(x, u_H; J_H)$ the Q-value function corresponding to J_H (the negative sign is there because in robotics we like to minimize cost, not maximize reward).

Human predictions about the robot

- the robot does not know the human's objective and the human, in turn, does not know the robot's objective
- Collaboration: A special case of the game when human and the robot are on the same team, same goal or objective: $J_H = J_R$.
- a **joint agent** whose actions are tuples of human–robot actions, (u_H, u_R) and who optimizes for $J_H(x, u_H, u_R) = J_R(x, u_R, u_H)$
- compute the optimal plan or policy for the joint agent
- Human don't follow perfectly laid plan
    - **model predictive control (MPC)** is the answer

Humans and Robots



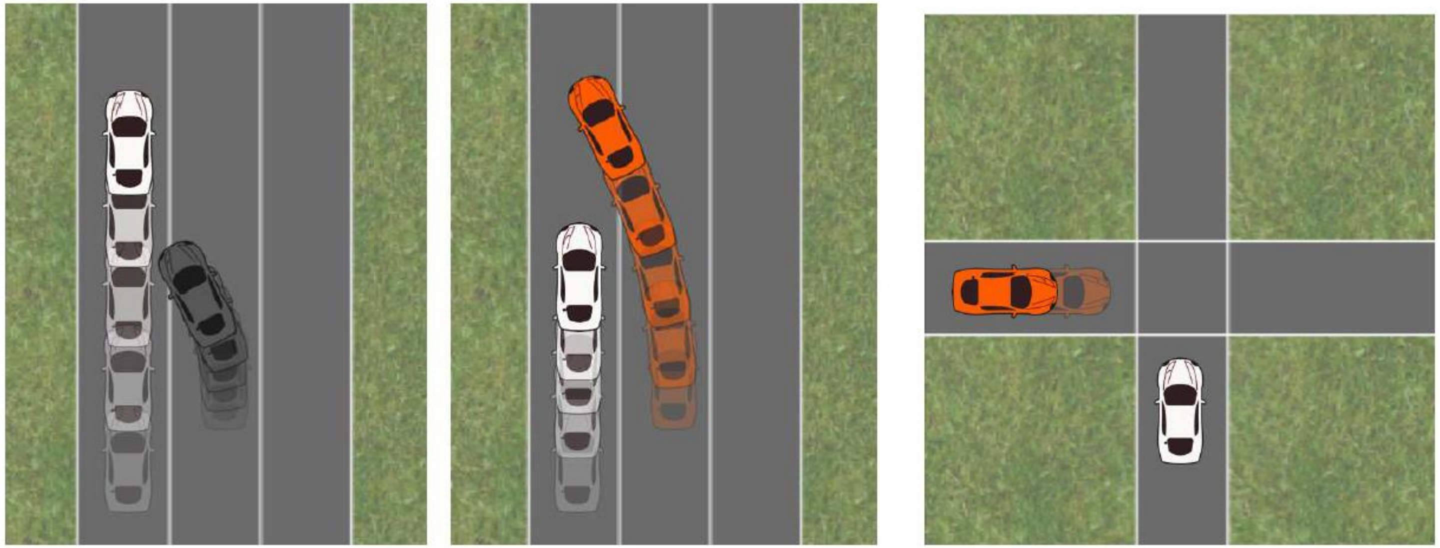
(a) (b) (c)

Making predictions by assuming that people are noisily rational given their goal: the robot uses the past actions to update a belief over what goal the person is heading to, and then uses the belief to make predictions about future actions.

(a) The map of a room.
(b) Predictions after seeing a small part of the person's trajectory (white path);
(c) Predictions after seeing more human actions: the robot now knows that the person is not heading to the hallway on the left, because the path taken so far would be a poor path if that were the person's goal.

Images courtesy of Brian D. Ziebart. See Ziebart *et al.* (2009).

Humans and Robots



(a) (b)

(a) Left: An autonomous car (middle lane) predicts that the human driver (left lane) wants to keep going forward, and plans a trajectory that slows down and merges behind. Right: The car accounts for the influence its actions can have on human actions, and realizes it can merge in front and rely on the human driver to slow down.
(b) That same algorithm produces an unusual strategy at an intersection: the car realizes that it can make it more likely for the person (bottom) to proceed faster through the intersection by starting to inch backwards. Images courtesy of Anca Dragan.

Humans and Robots

Humans as black box agents

- human is merely some agent whose policy π_H "messes" with the environment dynamics.

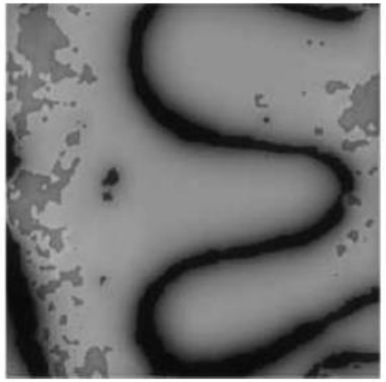
- The robot does not know π_H , but can model the problem as needing to act in an MDP with unknown dynamics

Learning to do what humans want

- Preference learning: Learning cost functions

- Learning policies directly via imitation
  - imitation learning or behavioral cloning
  - Challenge: generalization to new states
  - action the robot should take at each state: $D = \{(x_i, u_i)\}$
  - robot can run supervised learning to fit a policy $\pi : x \longmapsto u$, and execute that policy.
  - Address: interleave collecting labels and learning, reinforcement learning
  - Adversarial Training

Humans and Robots

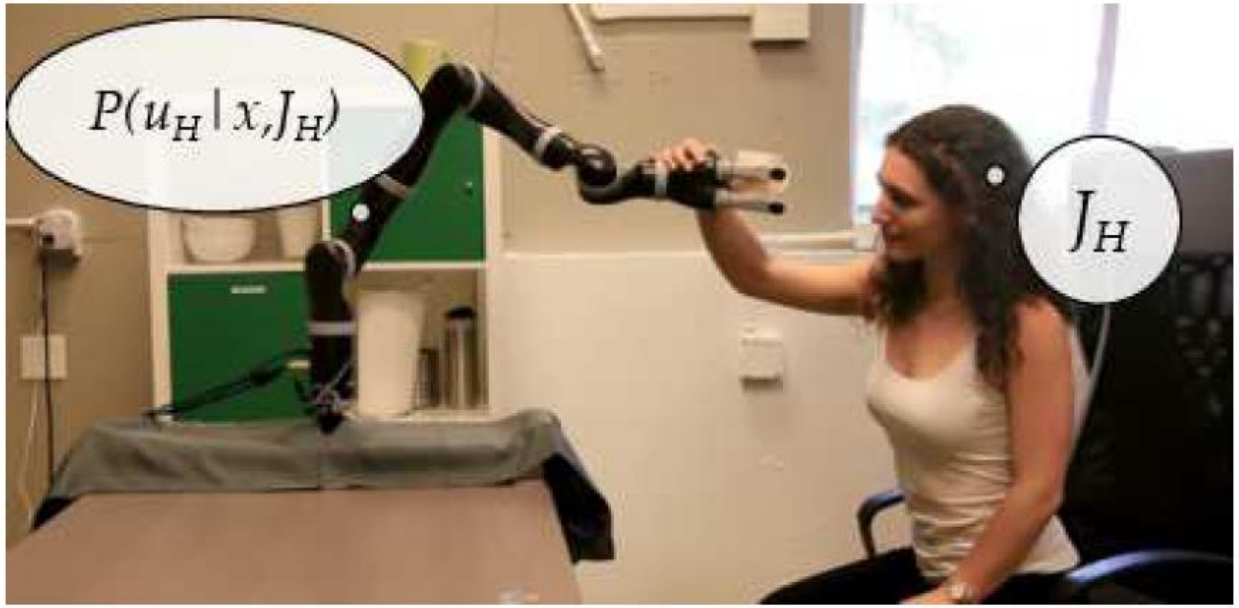


Left: A mobile robot is shown a demonstration that stays on the dirt road.

Middle: The robot infers the desired cost function, and uses it in a new scene, knowing to put lower cost on the road there.

Right: The robot plans a path for the new scene that also stays on the road, reproducing the preferences behind the demonstration.

Humans and Robots



A human teacher pushes the robot down to teach it to stay closer to the table. The robot appropriately updates its understanding of the desired cost function and starts optimizing it

Application Domains

- **Home care:** Robots have started to enter the home to care for older adults and people with motor impairments
- **Health care:** Robots assist and augment surgeons, enabling more precise, minimally invasive, safer procedures with better patient outcomes
- **Services:** Mobile robots help out in office buildings, hotels, and hospitals
- **Autonomous cars**
- **Entertainment:** Disney has been using robots (under the name animatronics) in their parks since 1963.
- **Exploration and hazardous environments:** Robots have gone where no human has gone before, including the surface of Mars.
- **Industry:** The majority of robots today are deployed in factories, automating tasks that are difficult, dangerous, or dull for humans.

Home Care Robots (Social Impact)

- **Role of Robots in Home Care:**
  - Robots are increasingly used to assist older adults and people with motor impairments in daily tasks, including medication reminders, mobility assistance, and companionship.
- **Implications for Society:**
  - **Positive Impact:** Improved quality of life, enhanced independence for older adults, reduced burden on family caregivers.
  - **Challenges:** Emotional dependency on robots, potential for reduced human interaction, privacy concerns, and ethical considerations on robot use in personal spaces.
- **Example:** The ElliQ robot provides social interaction and health monitoring for the elderly.

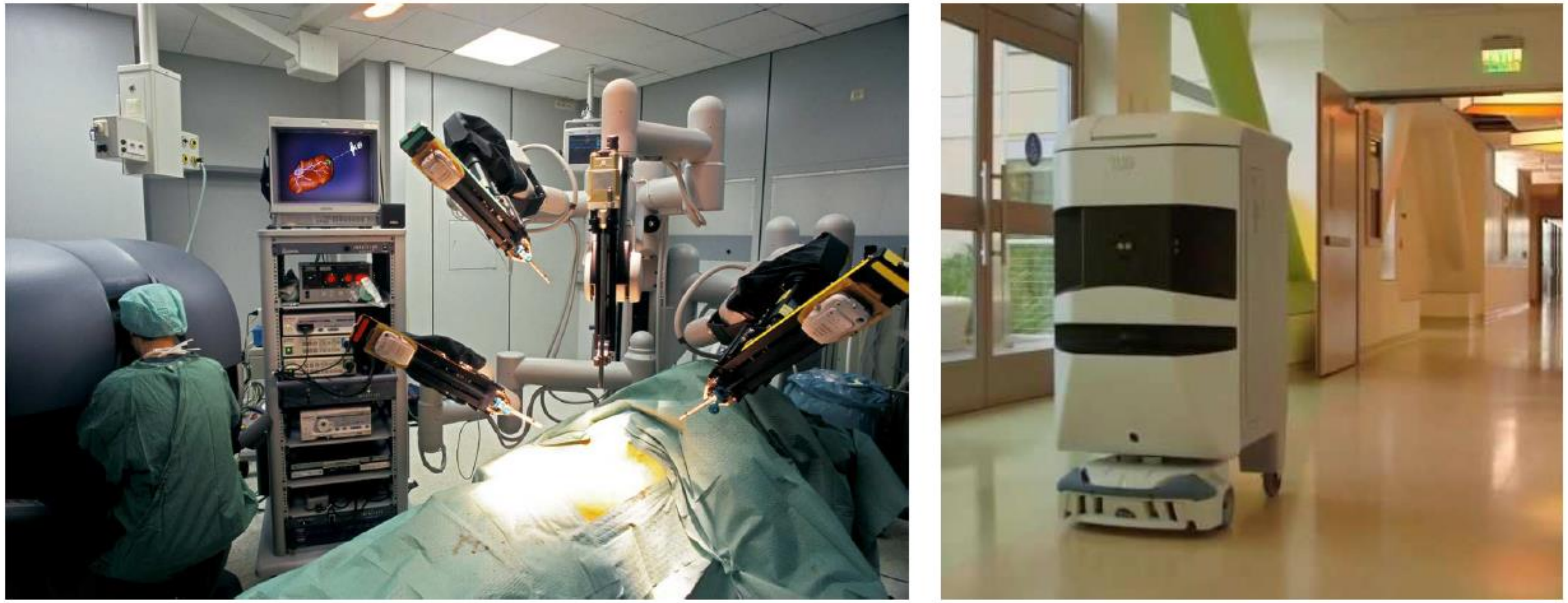
Healthcare Robotics (Social Impact)

- **Robotic Assistance in Healthcare:**
  - AI-enhanced robots assist surgeons with precise, minimally invasive procedures, improving patient outcomes and recovery times.
  - Robots aid in patient monitoring, rehabilitation, and elderly care.
- **Implications for Society:**
  - **Positive Impact:** Enhanced precision in surgery, improved patient safety, and reduced recovery time.
  - **Challenges:** High costs, training requirements, potential job displacement, and ethical considerations regarding AI decision-making in care.
- **Example:** The Da Vinci Surgical System for minimally invasive surgeries.

Service Robots (Social Impact)

- **Service Roles in Offices, Hotels, and Hospitals:**
  - Mobile robots provide assistance with deliveries, cleaning, and patient support in service industries.
- **Implications for Society:**
  - **Positive Impact:** Increased efficiency, reduced manual labor, enhanced customer and patient experience..
  - **Challenges:** Job displacement, cybersecurity risks, and dependency on technology for basic tasks.
- **Example:** Relay delivery robots in hospitals deliver supplies and medications to patient rooms.

Application Domains



(a) (b)

(a) Surgical robot in the operating room
(b) Hospital delivery robot.

Autonomous Vehicles (Social & Environmental Impact)

- **Impact of Autonomous Cars:**
  - Self-driving cars promise safer roads and reduce accidents, while also minimizing human error.
- **Implications for Society:**
  - **Social Impact:** Increased road safety, reduced traffic congestion, potential transformation of public transportation.
  - **Environmental Impact:** Reduced emissions through optimized driving patterns, potential reduction in car ownership.
- **Challenges:** High initial costs, ethical challenges in decision-making algorithms, cybersecurity concerns.
- **Example:** Waymo’s self-driving cars being tested in urban areas.

Application Domains

(a)

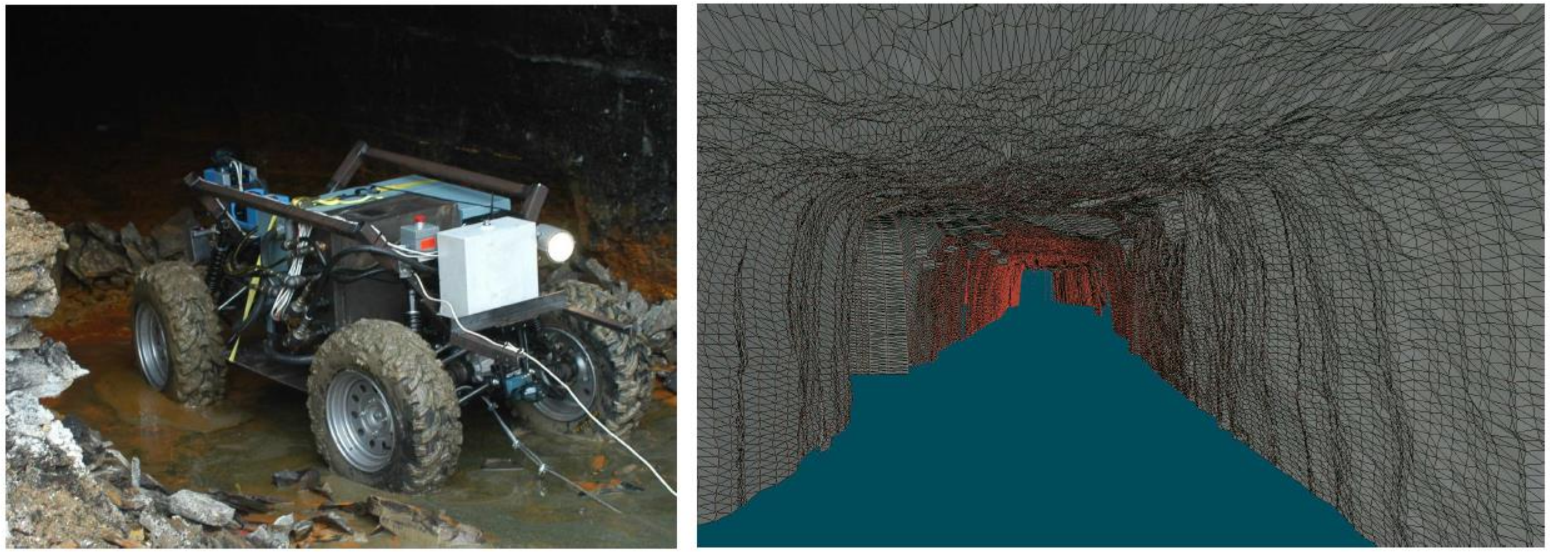
(b)

(a) Autonomous car BOSS which won the DARPA Urban Challenge.
(b) Aerial view showing the perception and predictions of the Waymo autonomous car (white vehicle with green track). Other vehicles (blue boxes) and pedestrians (orange boxes) are shown with anticipated trajectories. Road/sidewalk boundaries are in yellow.

Robotics in Hazardous Environments (Environmental Impact)

- **Robots in Exploration and Hazardous Tasks:**
  - Robots perform tasks in extreme environments, such as space exploration, underwater exploration, and nuclear decontamination.
- **Implications for the Environment:**
  - **Positive Impact:** Ability to explore new frontiers without human risk, safe handling of hazardous materials, contributes to environmental monitoring.
  - **Challenges:** High energy consumption, waste generated from used robots in harsh conditions, potential environmental disturbance.
- **Example:** Mars rovers exploring and collecting data on Mars.

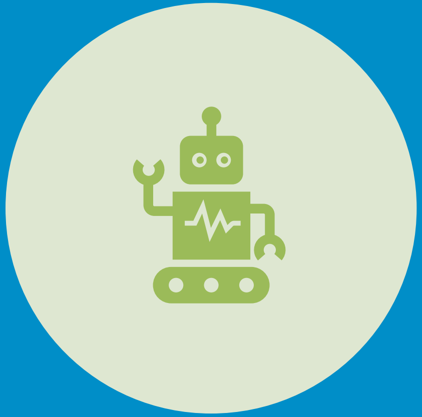
Application Domains



(a) (b)

(a) A robot mapping an abandoned coal mine.
(b) A 3D map of the mine acquired by the robot.

Balancing Innovation with Responsibility



Implications: AI-driven robotics can transform various sectors, offering vast benefits but also posing societal and environmental challenges.

Ethical Responsibility: Engineers and technologists need to address ethical concerns and environmental sustainability.

Future Outlook: Continued research and development should focus on minimizing negative impacts while maximizing societal benefits.

Summary

- The most common types of robots are manipulators (robot arms) and mobile robots.
- They have sensors for perceiving the world and actuators that produce motion, which then affects the world via effectors
- The general robotics problem involves stochasticity (which can be handled by MDPs), partial observability (which can be handled by POMDPs), and acting with and around other agents (which can be handled with game theory
- Robotic perception concerns itself with estimating decision-relevant quantities from sensor data
- Probabilistic filtering algorithms such as particle filters and Kalman filters are useful for robot perception
- A path found by a search algorithm can be executed using the path as the reference trajectory for a PID controller,
- Planning under uncertainty unites perception and action by online replanning (such as model predictive control) and information gathering actions that aid perception