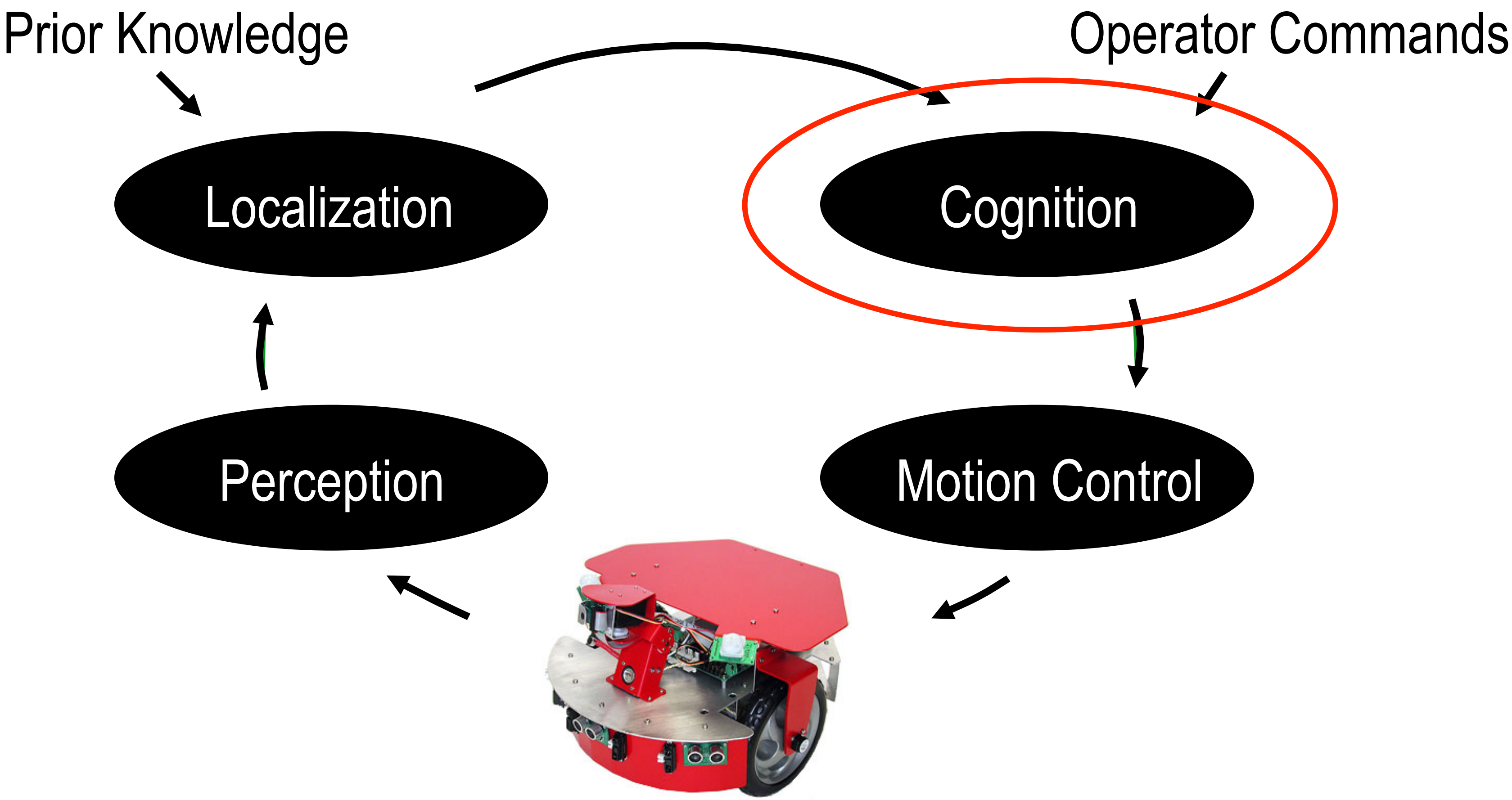
COS 495 - Lecture 18
Autonomous Robot Navigation

Instructor: Chris Clark

Semester: Fall 2011

Figures courtesy of Siegwart & Nourbakhsh

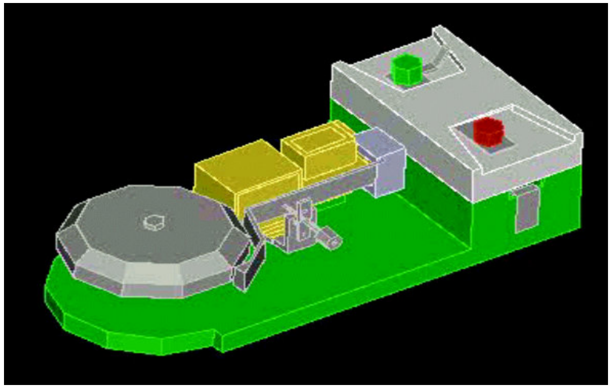
Control Structure



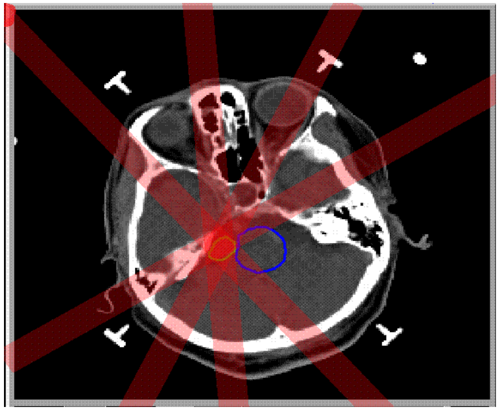
Introduction to Motion Planning

1. MP Overview
2. Metrics
3. The Configuration Space
4. General Approach to MP

MP Overview

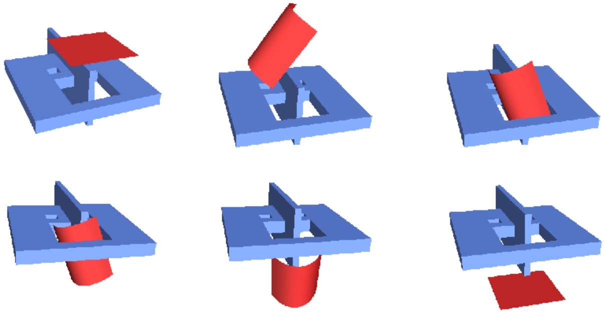


Assembly Planning, Latombe



Cross-Firing of a Tumor, Latombe

Tomb Raider 3 (Eidos Interactive)



Deformable Objects, Kavraki

MP Overview

- Goal of robot motion planning
  - Construct a collision-free path from some initial configuration to some goal configuration for a robot within a workspace containing obstacles.

MP Overview

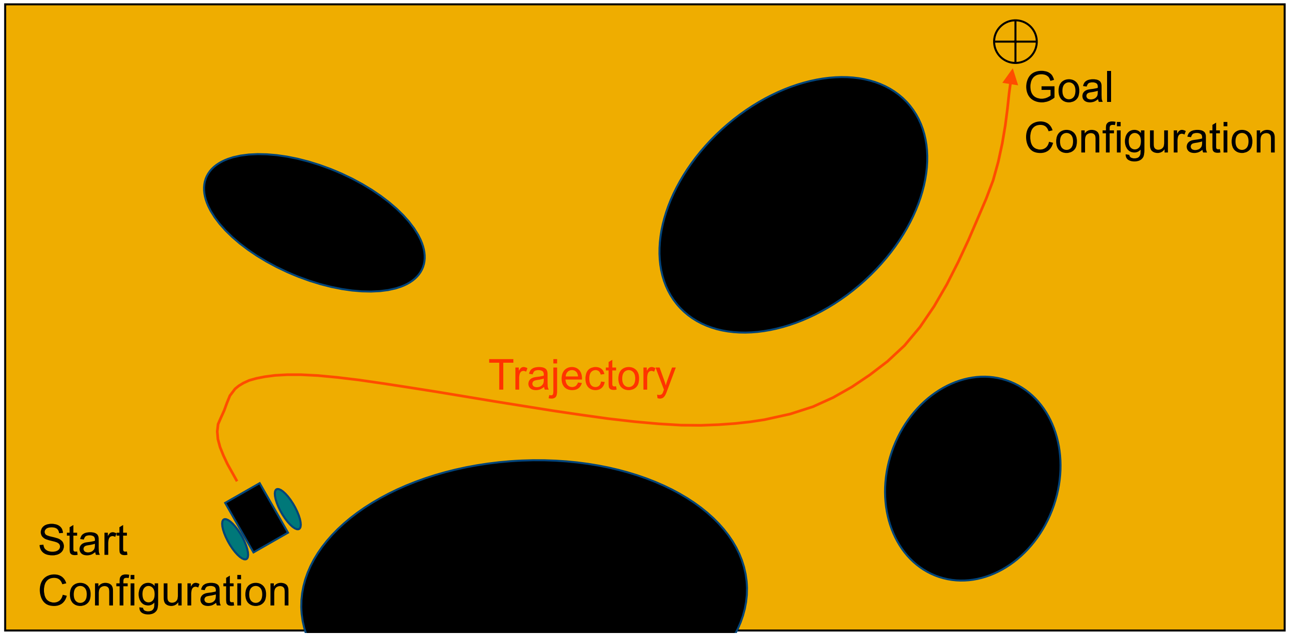
- Inputs
  - Geometry of robots and obstacles
  - Kinematics/Dynamics of robots
  - Start and Goal configurations
- Outputs
  - Continuous sequence of configurations connecting the start and goal configurations

MP Overview

- Assumptions
  - A model of the environment is provided.
    - Partial or full
  - A model of the robot is provided.
    - Kinematic constraints
    - Dynamic constraints

MP Overview

- Example:



MP Overview

- Extensions
  - Moving obstacles
  - Multiple robots
  - Movable objects
  - Assembly planning
  - Goal is to acquire information by sensing
  - Nonholonomic constraints
  - Dynamic constraints
  - Stability constraints
  - Uncertainty in model, control and sensing
  - Exploiting task mechanics (under-actuated systems)
  - Physical models and deformable objects
  - Integration with higher-level planning

Introduction to Motion Planning

1. MP Overview
2. Metrics
3. The Configuration Space
4. General Approach to MP

Metrics

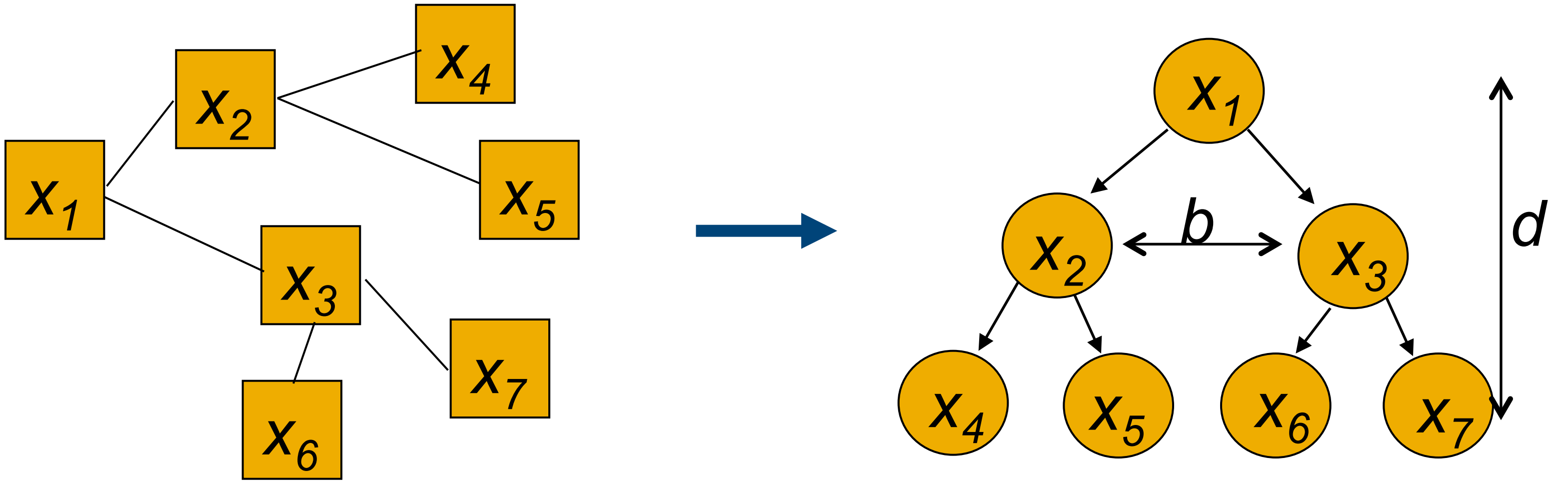
- Metrics for which to compare planning algorithms:
  1. Speed or Complexity
  2. Completeness
  3. Optimality
  4. Feasibility of solutions

Metrics

1. Speed or Complexity
   - Often, planners are compared based on the running time of an algorithm.
   - Example: Planner A outperformed Planner B in that it took half the time to solve the same planning problem.

Metrics

1. Speed or Complexity
   - Planners are also compared based on the complexity of the algorithm
     - Example: $O(b^d)$ where b is the breadth and d is the depth of a search tree.



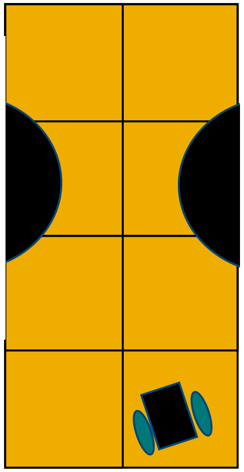
Metrics

2. Completeness
   - A *complete* algorithm is one that is guaranteed to find a solution if one exists, or determine if no solution exists.
   - Time Consuming!
   - Example:
     - An exhaustive search will search every possible path to see if it is a feasible solution.

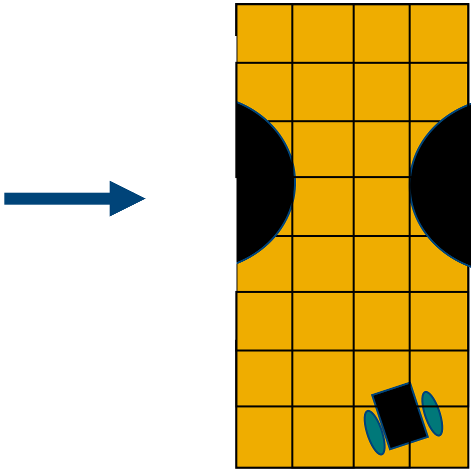
Metrics

2. Completeness

- A *resolution complete* planner discretizes the space and returns a path whenever one exists in the discretized representation.



No Solution



Solution!

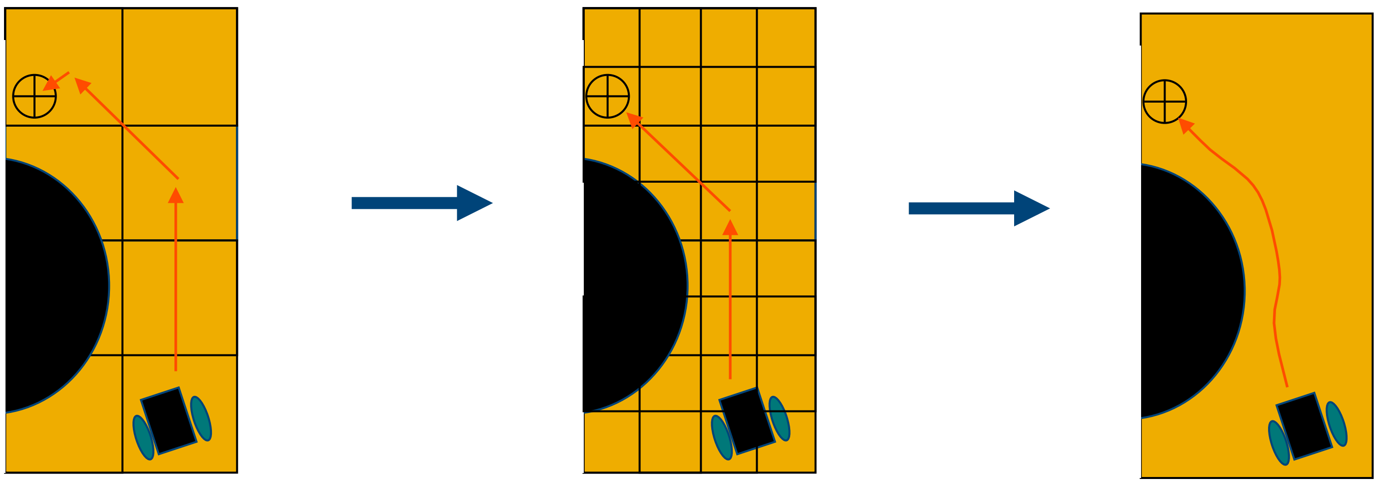
Metrics

2. Completeness

- A *probabilistically complete* planner returns a path with high probability if a path exists. It may not terminate if no path exists.
  - E.g. P(failure) → 0 as time → ∞
- Weaker form of completeness, but usually faster.

Metrics

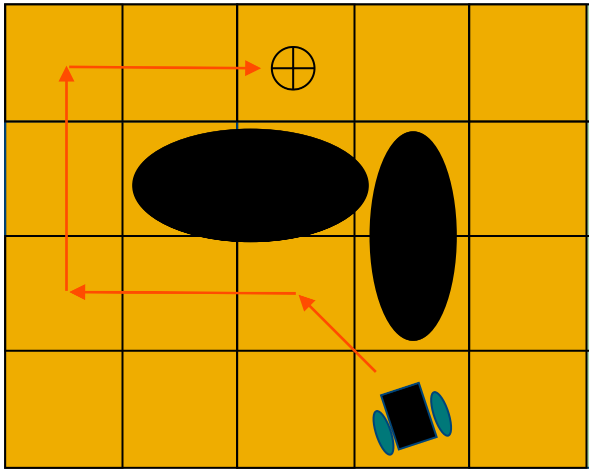
3. Optimality
   - Resolution of Discretization can lead to sub-optimal solutions



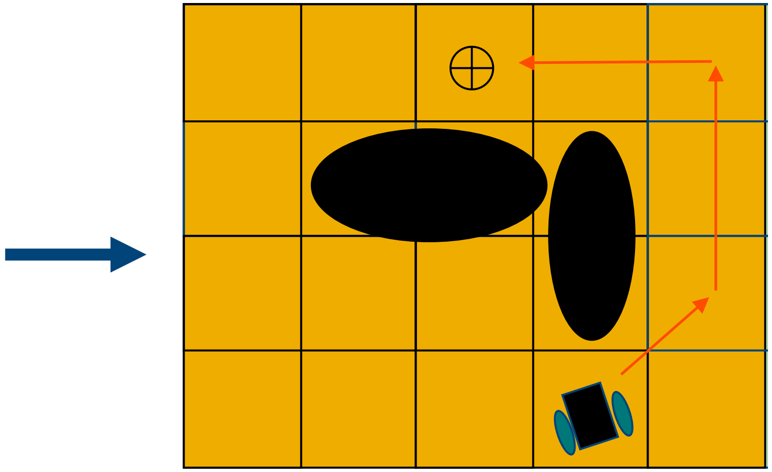
Metrics

3. Optimality

- Some algorithms will only guarantee sub-optimal solutions (e.g. Greedy Search).



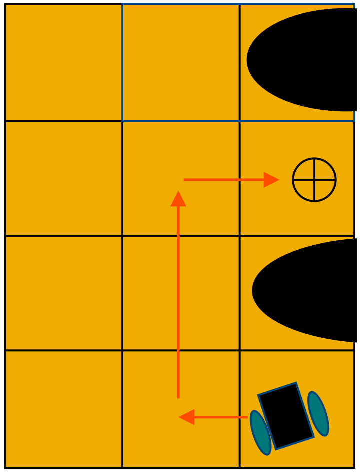
Sub-Optimal



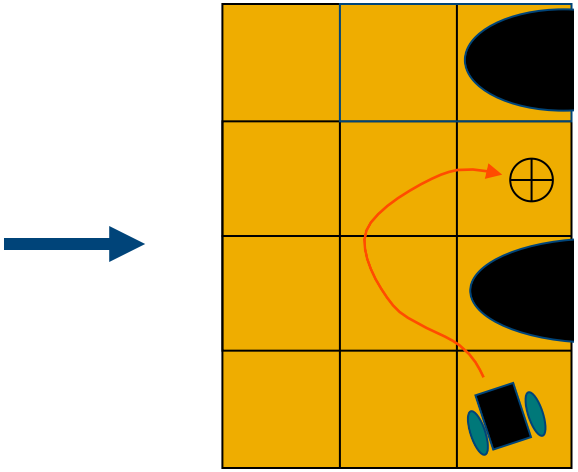
R. Optimal

Metrics

4. Feasibility of Solutions
   - Not all planners take into account the exact model of the robot or environment.
   - E.g. Non-differential drive robot



Infeasible



Feasible

Metrics

- A complete planner usually requires exponential time in the number of degrees of freedom, objects, etc.
- Theoretical algorithms
  - Strive for completeness and minimal worst-case complexity
  - Difficult to implement
- Heuristic algorithms
  - Strive for efficiency in common situations
  - Use simplifying assumptions
  - Weaker completeness
  - Exponential algorithms that work in practice

Introduction to Motion Planning

1. MP Overview
2. Metrics
3. The Configuration Space
4. General Approach to MP

The Configuration Space

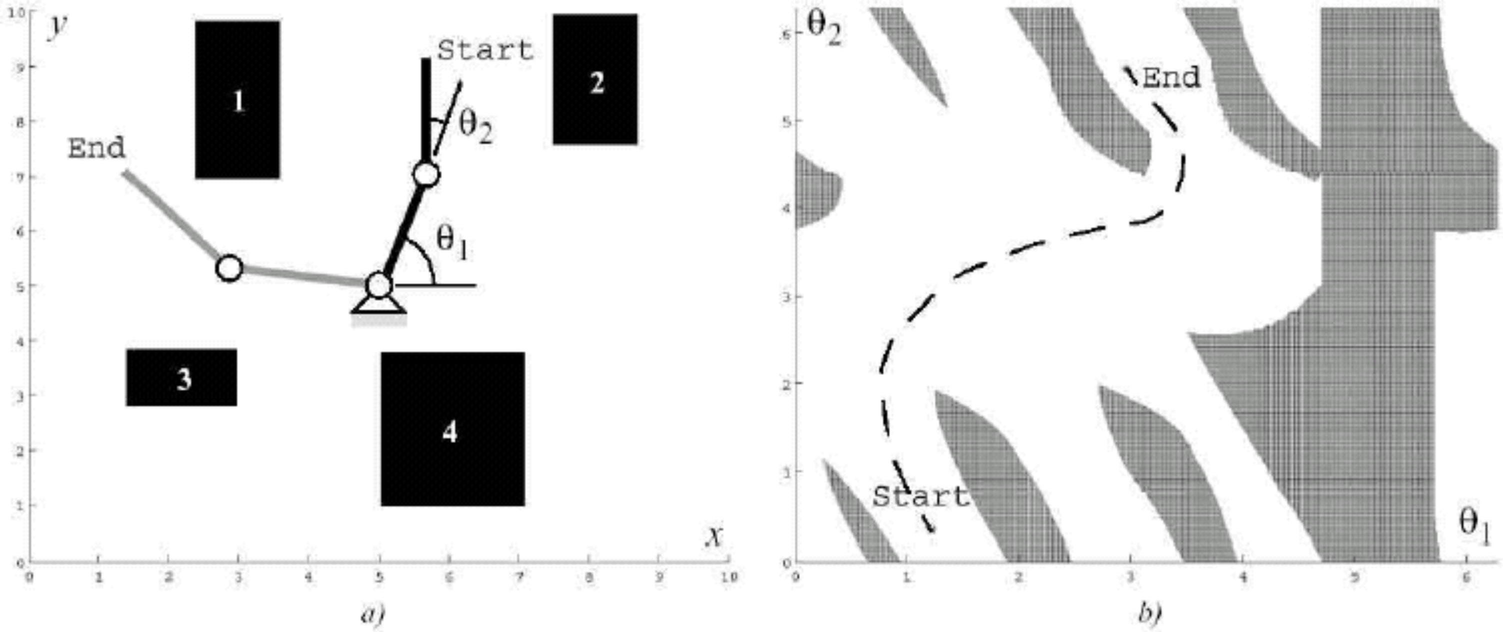
- To facilitate motion planning, the configuration space was defined as a tool that can be used with planning algorithms.

The Configuration Space

- A configuration q will completely define the state of a robot (e.g. mobile robot x, y, θ)
- The configuration space C, is the space of all possible configurations of the robot.
- The free space $F \subseteq C$, is the portion of the free space which is collision-free.
- The goal of motion planning then, is to find a path in F that connects the initial configuration q_{start} to the goal configuration q_{goal}

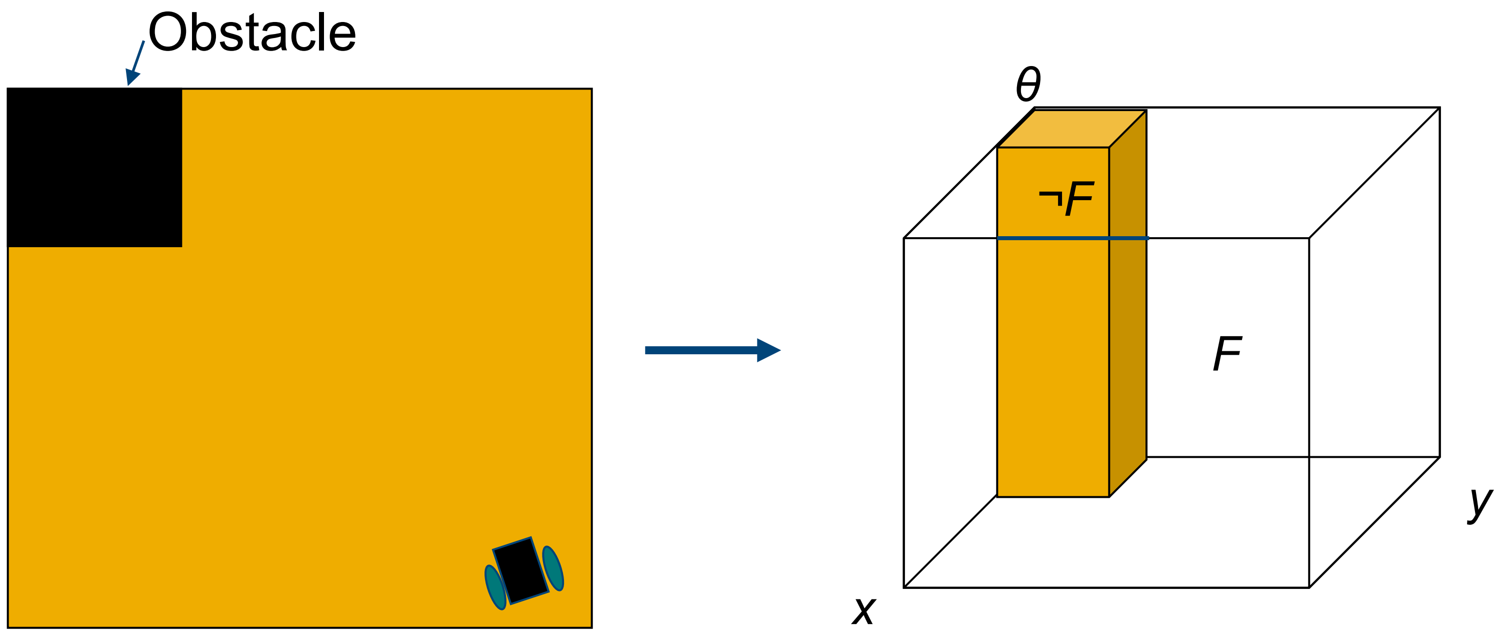
The Configuration Space

- Example 1: 2DOF manipulator:



The Configuration Space

- Example 2: Mobile Robot



Workspace

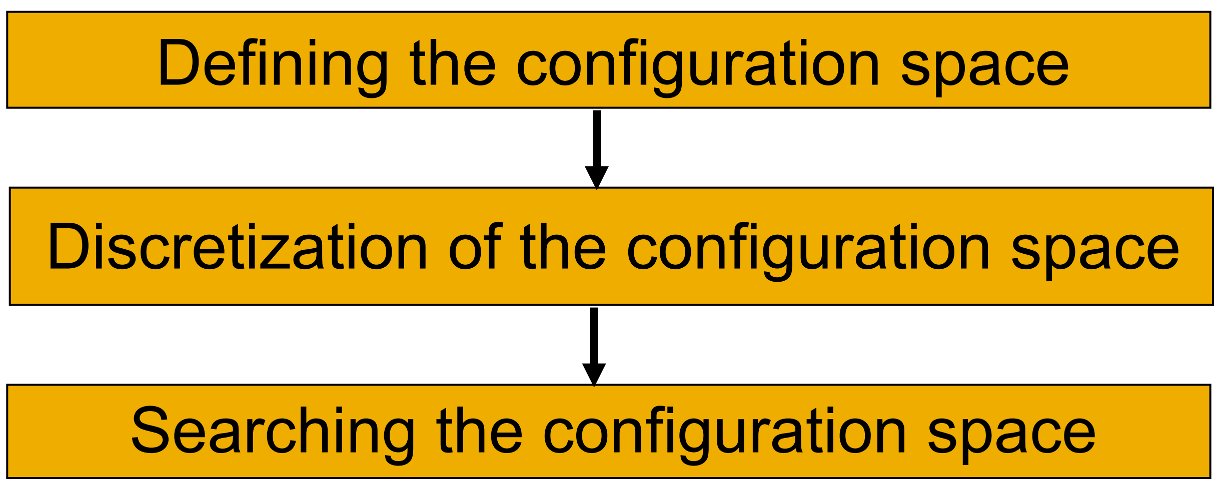
Configuration Space

Introduction to Motion Planning

1. MP Overview
2. Metrics
3. The Configuration Space
4. General Approach to MP

Motion Planning: General Approach

- Motion planning is usually done with three steps:

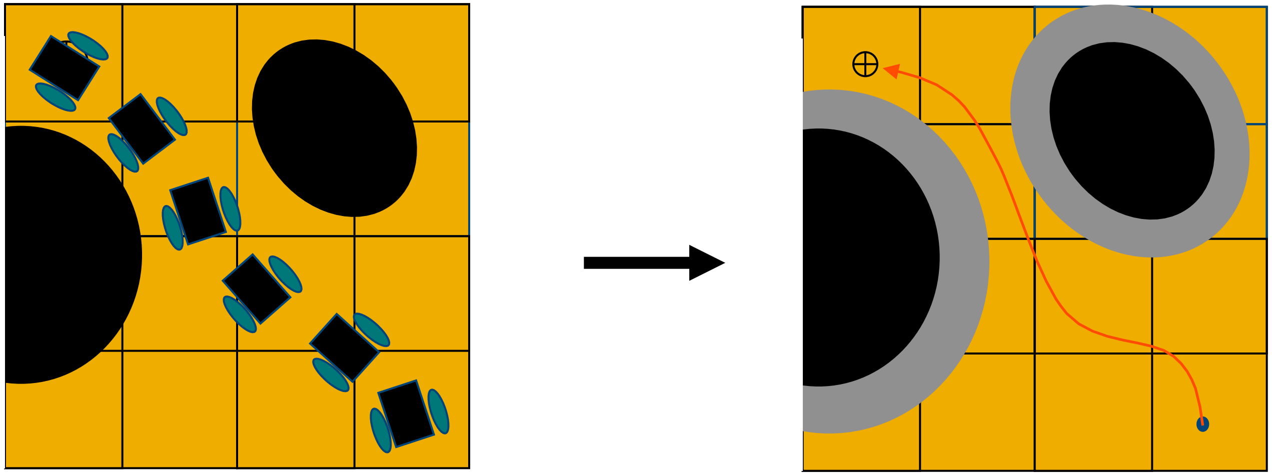


Motion Planning: Defining the Configuration Space

- The configuration space *C* must include all DOF's that capture all the configurations of the robot with constraints.
- Sometimes we plan with a subspace of *C*.
  - Example 1:Include 3DOF for a mobile robot in static environment - (x,y,θ).
  - Example 2: Include only 2DOF for a mobile robot in static environment - (x,y).
  - Example 3: Include 5DOF for a mobile robot in dynamic environment - (x,y,θ,v,t).

Motion Planning: Defining the Configuration Space

- Plan paths for a point robot
  - Instead of using a robot of fixed dimensions/size, "grow" the obstacles to reflect how close the robot can get.

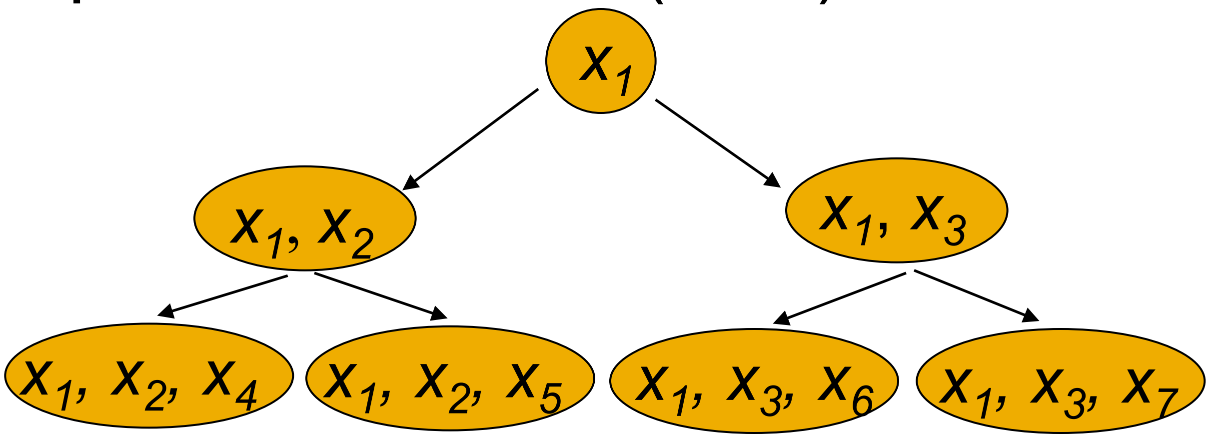


Motion Planning: Discretizing the Configuration Space

1. Roadmap
   - Represent the connectivity of the free space by a network of 1-D curves
2. Cell decomposition
   - Decompose the free space into simple cells and represent the connectivity of the free space by the adjacency graph of these cells
3. Potential field
   - Define a function over the free space that has a global minimum at the goal configuration and follow its steepest descent

Motion Planning: Searching the Configuration Space

- Given a discretization of *F* (e.g. a topological map), a search of the discretized map can be carried out using a Graph search or gradient descent, etc.
- Example: Tree Search (DFS)



Motion Planning: Searching the Configuration Space

- Example: Multi Robot MP