Massachusetts Institute of Technology

Part 1: Introduction to Path Planning

Lars Blackmore

CSAIL

Introductory Background

- Intro to Path Planning: AIMA Ch. 25.4
- Uninformed Search: AIMA Ch. 3
- Informed Search: AIMA Ch. 4

AIMA = Artificial Intelligence: A Modern Approach, by Russell and Norvig

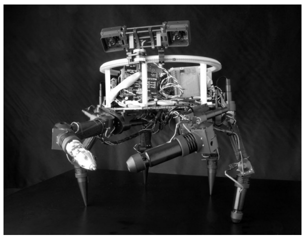
Massachusetts Institute of Technology 2

Overview

- Problem Setup
  - From workspace to configuration space
- Path Planning in Continuous Configuration Spaces
  - Potential Field Methods
- Generating Roadmaps
  - Visibility graph
  - Voronoi diagrams
  - Cell decomposition
- Part 2: Path planning with Probabilistic Roadmaps
  - Brian

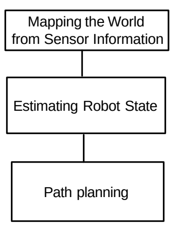
Massachusetts Institute of Technology 3

Introduction



Problem Definition

- Previous lectures:



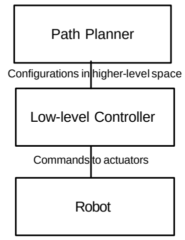
- This lecture:
- Problem Statement: Compute a **collision-free path** for a rigid or articulated object (the robot) among **static** obstacles

Problem Definition

- Inputs:
  – Model of robot
  – Initial and goal robot configurations
- Output:
  – Continuous sequence of collision-free robot configurations connecting the initial and goal configurations

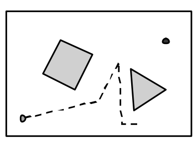
Path Planning and Modeling

- In order to make a planning problem tractable, typically abstract away complexities of robot model



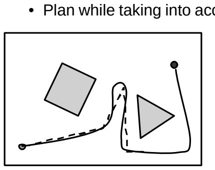
Path Planning and Modeling

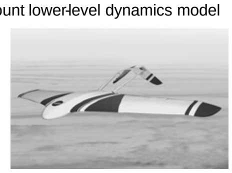
- Example: Roomba
  - Goal defined in terms of location on 2-D floor
  - 2-D location is “controllable”
  - Plan at level of 2-D location
  - Low-level controller issues motor commands



Path Planning and Modeling

- Example: UAV
  - 2-D location is not “controllable”
  - Approach A:
    - Plan a path as if 2-D location controllable
    - Try to follow this path as best possible
  - Approach B:
    - Plan while taking into account lower-level dynamics model



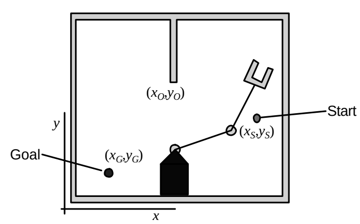


Initial Assumptions for this Lecture

- Path planning in controllable spaces
  - Later in this lecture we will talk about planning with dynamics models
- Robot state known exactly
  - Estimation algorithms (e.g. SLAM) give distributions over robot state
  - Typical approach: use **mean** of distribution
  - Later lectures will talk about path planning with uncertainty

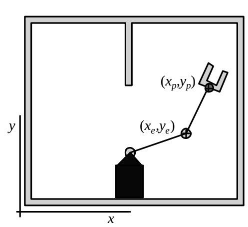
Workspace to Configuration Space

- Path planning problem usually defined in terms of workspace coordinates
  - Task: Go from start position to goal position
  - Constraint: Do not collide with any obstacles



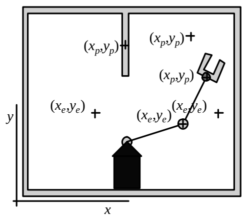
Workspace to Configuration Space

- In workspace coordinates, configuration of manipulator described by position of elbow and endpoint



Workspace to Configuration Space

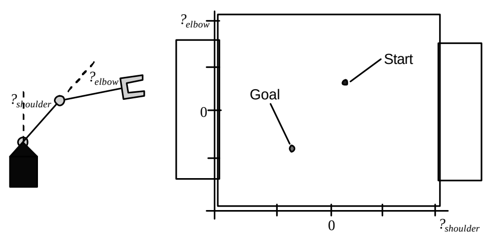
- Problem: not all elbow/endpoint configurations possible



- To reach goal, must find path that has **feasible** configurations along **entire path**

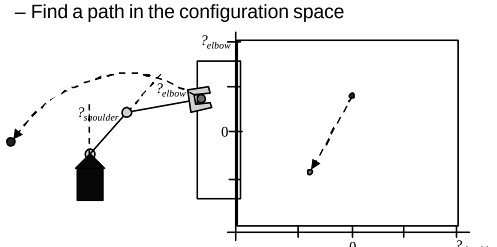
Workspace to Configuration Space

- Problem can also be represented in **configuration space**



Workspace to Configuration Space

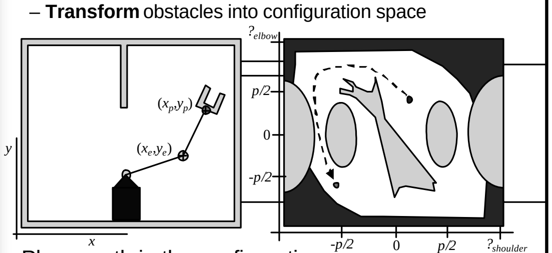
- Solution: plan in **configuration space**
  - Transform start and goal position into configuration space
  - Find a path in the configuration space



- Without obstacles, this is easy…

Workspace to Configuration Space

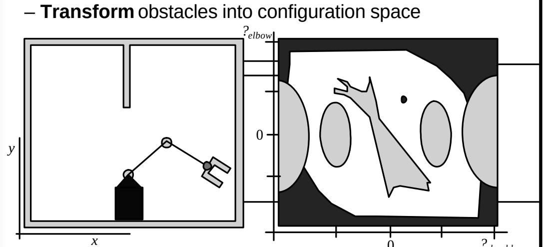
- With obstacles:
  - **Transform** obstacles into configuration space



- Plan a path in the configuration space

Workspace to Configuration Space

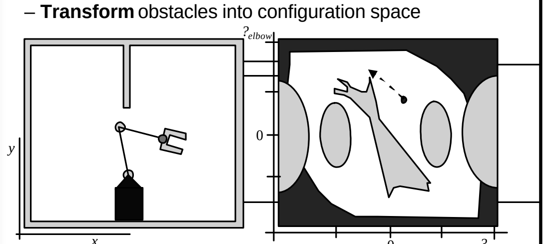
- With obstacles:
  - **Transform** obstacles into configuration space



- Plan a path in the configuration space

Workspace to Configuration Space

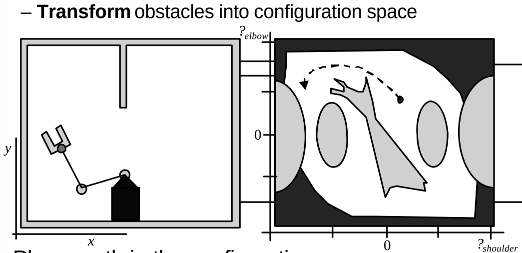
- With obstacles:
  - **Transform** obstacles into configuration space



- Plan a path in the configuration space

Workspace to Configuration Space

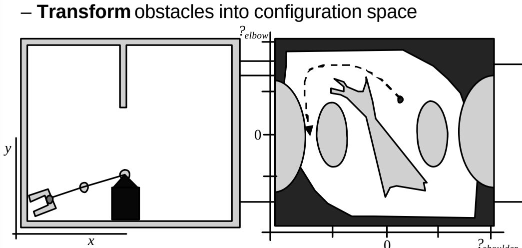
- With obstacles:
  - **Transform** obstacles into configuration space



- Plan a path in the configuration space

Workspace to Configuration Space

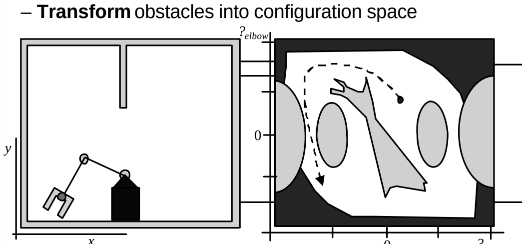
- With obstacles:
  - **Transform** obstacles into configuration space



- Plan a path in the configuration space

Workspace to Configuration Space

- With obstacles:
  - **Transform** obstacles into configuration space



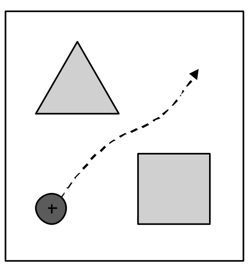
- Plan a path in the configuration space

Workspace to Configuration Space

- Configuration space represents all feasible assignments to robot state, start and goal states
  - Now we can think about how to plan in this space
- Configuration space has same dimensionality as robot state
  - Configuration space can have higher or lower dimensionality than workspace
- Other pictures of configuration spaces…

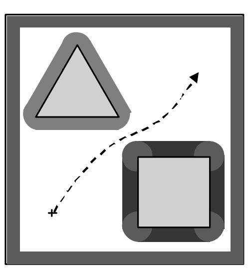
Workspace to Configuration Space

- Circular mobile robot in 2-D space



Workspace to Configuration Space

- Circular mobile robot in 2-D space



Workspace to Configuration Space

- Polygonal Robot in 2-D Space

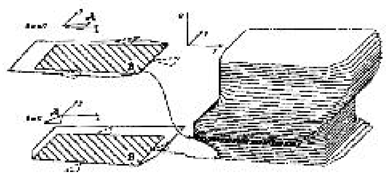


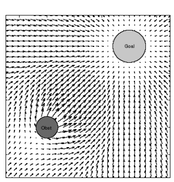
Figure 8. A and B are convex polygons. A is a free-flying object. C is represented by parameterizing each configuration q by (x,y,θ) ∈ R² × [0,2π). The represented C-obstacle is a volume bounded by patches of ruled surfaces (see Chapter 3).

Overview

- Problem Setup
  - From workspace to configuration space
- **Path Planning in Continuous Configuration Spaces**
  - **Potential Field Methods**
- Generating RoadMaps
  - Visibility graph
  - Voronoi diagrams
  - Cell decomposition
- Part 2: Path planning with Probabilistic Roadmaps
  - Brian

Potential Field Methods

- Works for continuous spaces
- Key idea:
  - Robot follows suitably defined function downhill until reaches goal
  - Function defined so that robot pushed towards goal and away from obstacles



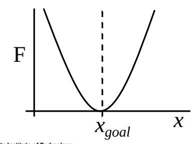
Potential Field Methods

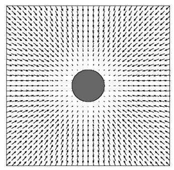
- Define potential function f

$$f = \mathrm{F} + ?$$

- Define attractive potential F for goal:
  - Example:

$$\Phi = k_1 \left\| \mathbf{x} - \mathbf{x}_{goal} \right\|^2$$

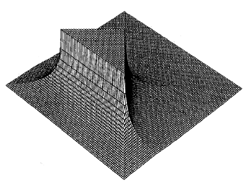


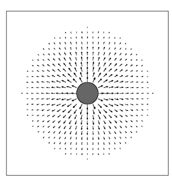


Potential Field Methods

- Example ? to guarantee no collisions?

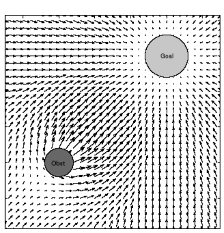
$$\Psi = \frac{k_2}{\left\| \mathbf{x} - \mathbf{x}_{obs} \right\|^2}$$





Potential Field Methods

- Overall $f = \mathrm{F} + ?$



- Robot follows gradient downhill so that $\dot{\mathbf{x}} = -\nabla f$

Potential Field Methods

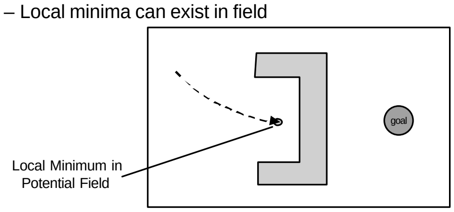
- Are we guaranteed to stop at the goal?
  - Not exactly with the functions defined so far

$$\Phi = k_1 \left\| \mathbf{x} - \mathbf{x}_{goal} \right\|^2 \qquad \Psi = \frac{k_2}{\left\| \mathbf{x} - \mathbf{x}_{obs} \right\|^2}$$

  - Solutions?
    - Small k_2 (approximate solution)
    - Logarithmic attractor function
    - Discontinuous obstacle function

Potential Field Methods

- Drawbacks?
  - Not optimal
  - Complete?
- Potential field methods are not complete:
  - Local minima can exist in field



Potential Field Methods

- Main drawback:
  - No principled way of generating functions to guarantee completeness or optimality

Overview

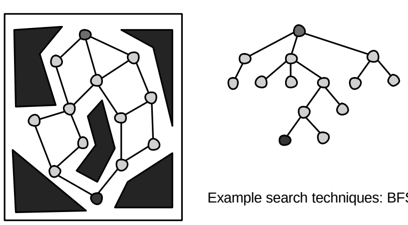
- Problem Setup
  - From workspace to configuration space
- Path Planning in Continuous Configuration Spaces
  - Potential Field Methods
- **Generating Roadmaps**
  - **Visibility graph**
  - **Voronoi diagrams**
  - **Cell decomposition**
- Path planning with Probabilistic Roadmaps
  - Brian

From Continuous Maps to Roadmaps

- Want to plan a path in the configuration space
- Two classes of approach
  - Plan directly in the configuration space
    - Infinite space (continuous)
  - Represent configuration space as a roadmap
    - Plan in the roadmap
    - Assume that low-level controller takes care of conversion from discrete map to continuous space

From Continuous Maps to Roadmaps

- Planning with roadmaps:
  - Reminder of how to search graphs



Example search techniques: BFS, DFS, A*

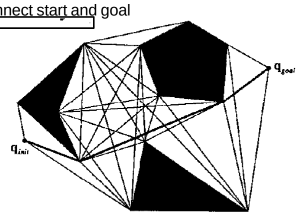
See Russell & Norvig Ch. 3 & 4

From Continuous Maps to Roadmaps

- Techniques for roadmap generation
  - Visibility graph
  - Voronoi diagram
  - Cell decomposition

From Continuous Maps to Roadmaps

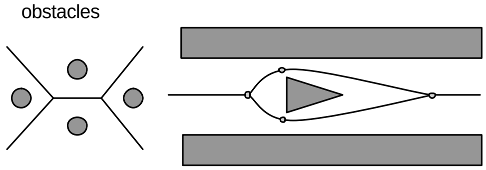
- Visibility Graph
  - Connect each obstacle corner to every other visible corner
  - Also connect start and goal



- Advantages:
  - Shortest path also shortest path in original space (in 2-D only)
- Disadvantages:
  - Not robust

From Continuous Maps to Roadmaps

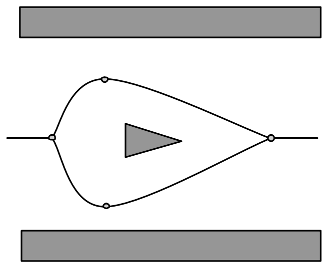
- Voronoi Graph
  - **Definition:** set of all points equidistant to two or more obstacles



- Disadvantages?
  - Hard to calculate (esp. in higher dimensional spaces)
  - Excessive clearance in wide-open spaces

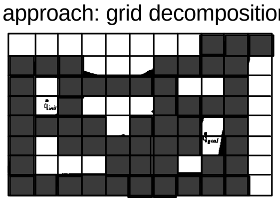
From Continuous Maps to Roadmaps

- Unnecessary clearance in wide open spaces



From Continuous Maps to Roadmaps

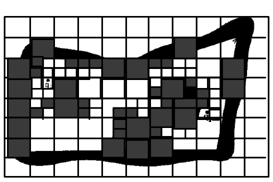
- Cell decomposition
  - Break up space into cells that are easily traversable
- Simplest approach: grid decomposition



  - Can lose completeness

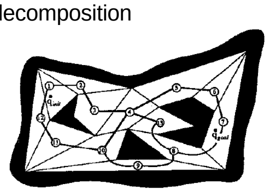
From Continuous Maps to Roadmaps

- Variable-sized cell decomposition



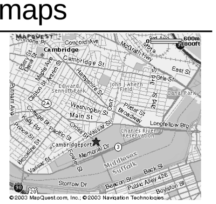
From Continuous Maps to Roadmaps

- Exact cell decomposition



- Cell decomposition drawbacks:
  - Computation can be difficult
  - Does not scale well to larger-dimensional spaces

Exploring Roadmaps



- Shortest path
  - Dijkstra's algorithm
  - Bellman-Ford algorithm
  - Floyd-Warshall algorithm
  - Johnson's algorithm

- Informed search
  - Uniform cost search
  - Greedy search
  - A* search
  - Beam search
  - Hill climbing

Overview

- Problem Setup
  - From workspace to configuration space
- Path Planning in Continuous Configuration Spaces
  - Potential Field Methods
- Generating RoadMaps
  - Visibility graph
  - Voronoi diagrams
  - Cell decomposition
- **Part 2: Path planning with Probabilistic Roadmaps**
  - **Brian**