Motion Planning

Stephen J. Guy

March 2, 2020

Many images from *Lavalle, Planning Algorithms* and *Peter Abbeel*

1

Recap

- Why use spatial-data structures
  - Exploit locality in the environment
- BSP-tree vs KD-tree
  - BSP allows arbitrary cuts
- Efficient nearest neighbor search?
  - Store all agents in kd-tree
  - Intersect kd-tree with circle centered on agent in question

2

2

HW3 – Motion Planning

- Posted already?
- Check-in: March 16
- Due: March 30

- Should be able to finish check-in from toady's lecture

- Very different than HW2!
  - For good or ill …

Burn-out?

- I know HW2 was difficult
  - Material is straightforward, but
  - Different from other type of course work
  - Lots of parameter tuning & tweaking
  - Small errors cause big issues
- HW3
  - You have a long time to complete
  - Still parameter tuning, but less difficult
  - Not as much ~~difficult~~ fun new math
  - A lot of fun new algorithms ☺

Motion in Games/Movies

- Natural Phenomena (Passive)
  - Water, Ocean
  - Fire, Smoke
  - Snow
  - Cloth

Physically Based Animation

- Planned Phenomena (Active)
  - Walking, Running
  - Animal locomotion
  - Herding, flocking
  - Planning in complex environments
  - Intent

Motion Planning

5

5

Motion Planning

- Problem
  - Given start state X_S and goal state X_G
  - Find a path leading from start to goal
- Difficulties
  - Need to avoid obstacles
  - Systems may be "underarticulated"
    - Can not move along any coordinate at will
    - Cars, Tanks, Airplanes
    - Humans, Animals, Robots (e.g. joint limits)

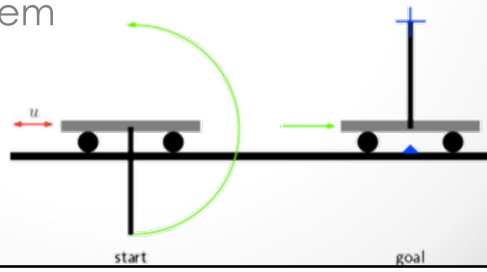
6

6

Motion Planning in Robotics

- Piano movers problem

- Cart-pole problem

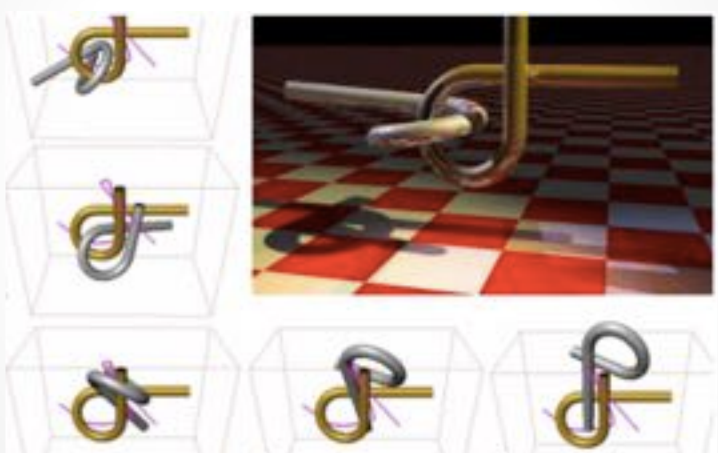


7

7

Examples

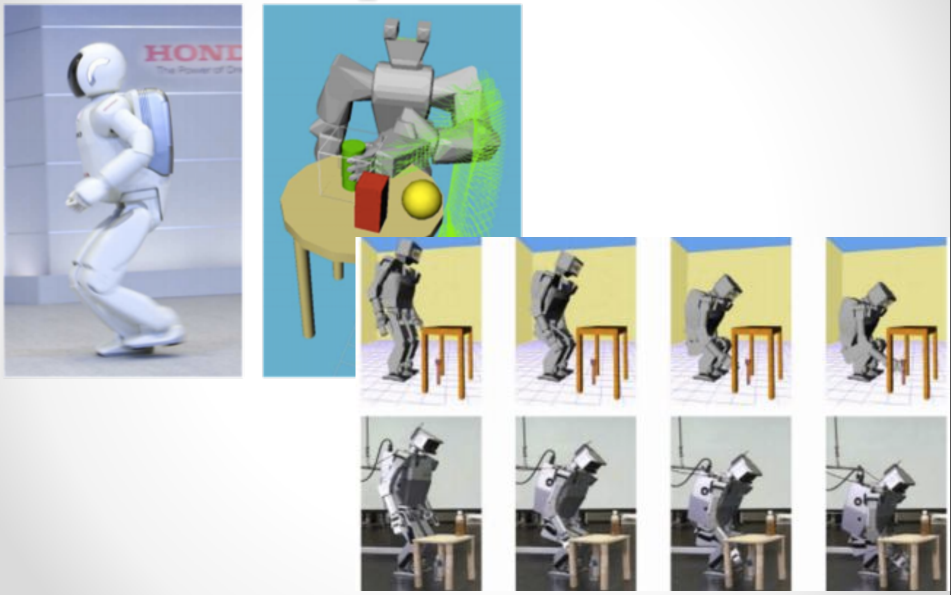
- Alpha puzzle



8

8

Examples - Robots



9

Outline

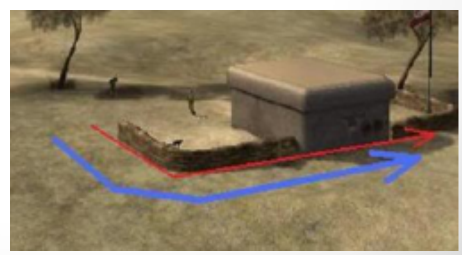
- Configuration Space
- Probabilistic Roadmaps
  - Sampling
  - Collision Checking
- Rapidly-exploring Random Trees (RRTs)
- Smoothing

10

10

Representing Paths

- Issues with raw position
  - Doesn’t account for an agent’s extent
  - Doesn’t account for orientation
  - Doesn’t allow arbitrary changes in configuration
- What is “correct” path around obstacle
  - Depends on whose taking the path

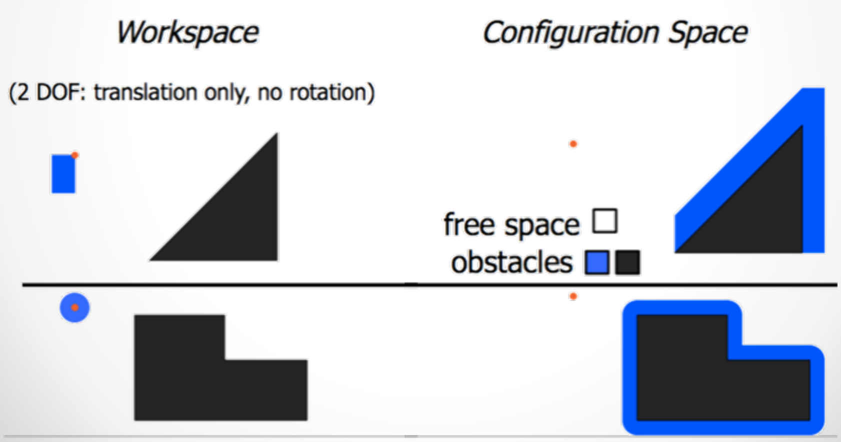


11

11

Configuration Space (C-Space)

- = {x | x is pose of agent}
- Key idea: Replace obstacles with configuration space obstacles!

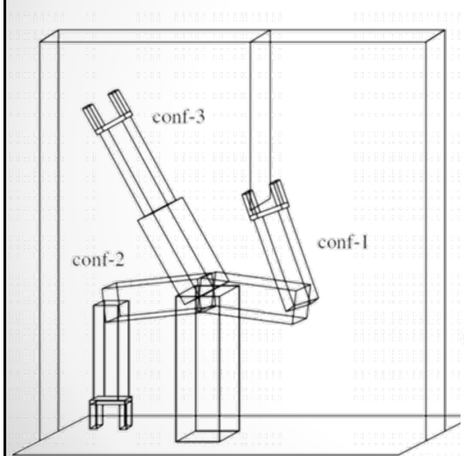


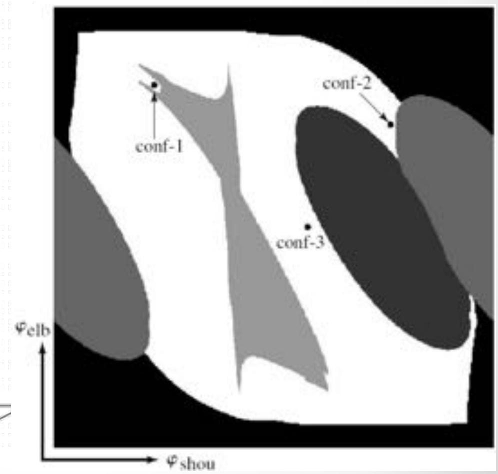
12

12

C-Space Example

- Works on arbitrarily complex agents!



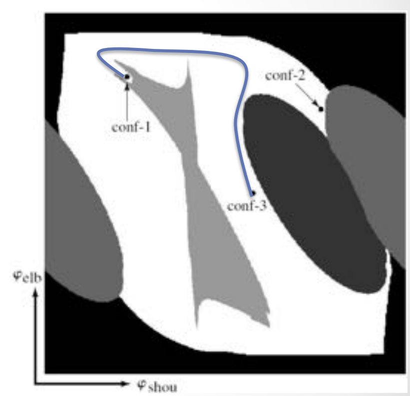


13

13

Planning a Path

- How to get from one configuration to another
- Must find path in free space of c-space
  - o Path in c-space that doesn't intersect any c-obstacles



14

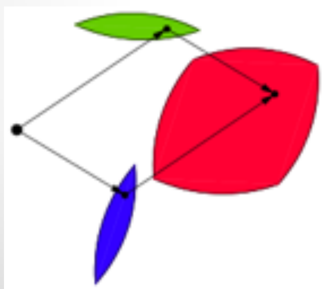
14

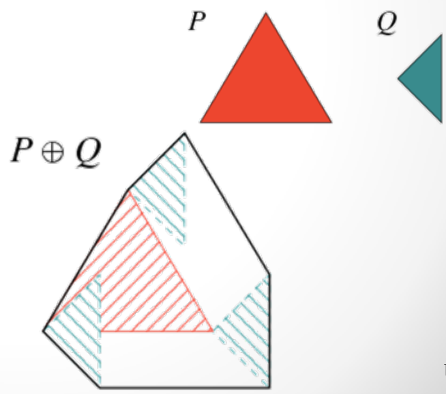
Minkowski Sums

- Geometric concept which can help compute C-space
- **Minkowski sum** is set of sum of every point in two sets

$$A + B = \{\mathbf{a} + \mathbf{b} \mid \mathbf{a} \in A,\ \mathbf{b} \in B\}$$

  - Can be computed by a convolution



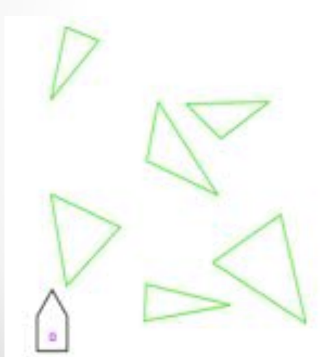


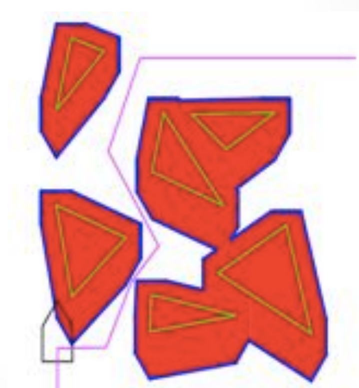
15

15

Minkowski Sum for C-Space

- Obstacle is A
- Robot is B
  - B’ is B reflected around reference point
- If only translation is allowed C-space is $A \oplus B'$





16

16

Roadmaps

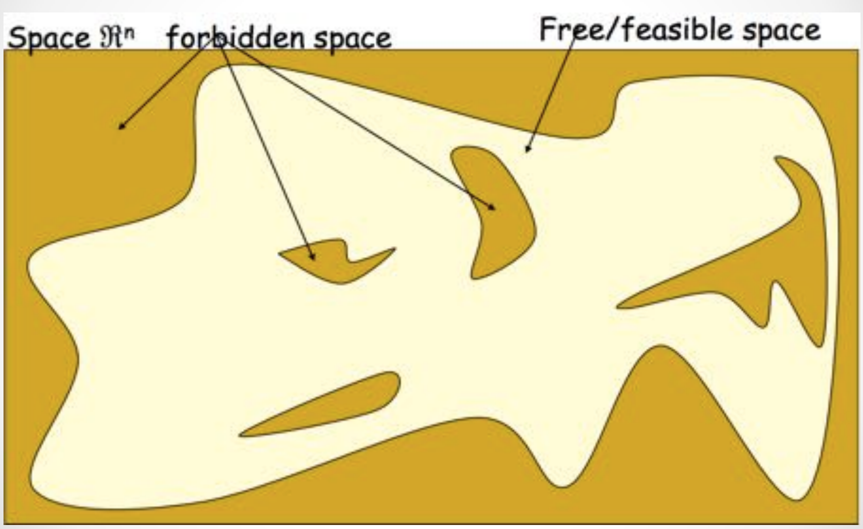
- Given a start and goal position
- And a representation of Configuration space

- How can we plan a collision free path from start to goal?

17

17

Roadmaps

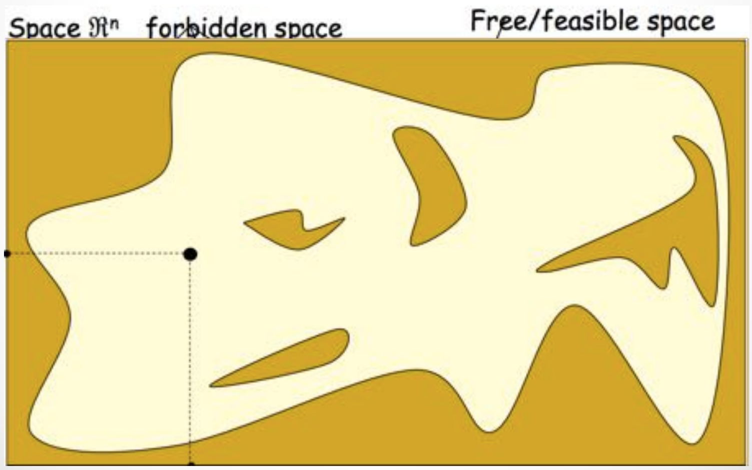


18

18

Probabilistic Roadmap

- Sampling random configurations
  - Choose each coordinate at random

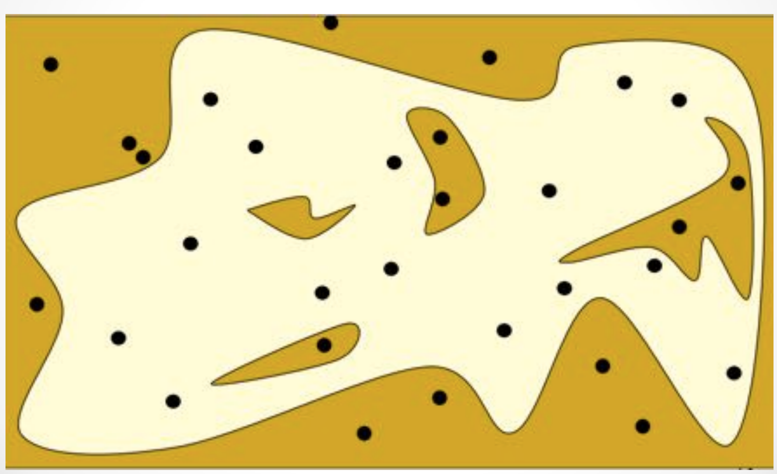


19

19

Probabilistic Roadmap (PRM)

- Randomly sample configurations

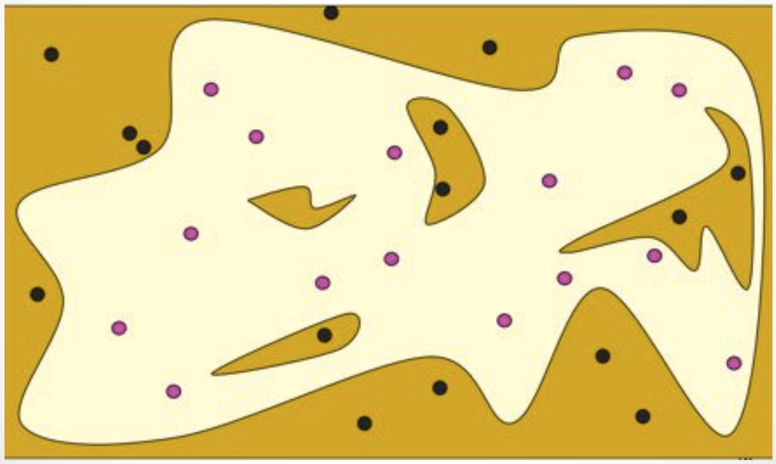


20

20

Probabilistic Roadmap (PRM)

- Test configurations for collisions

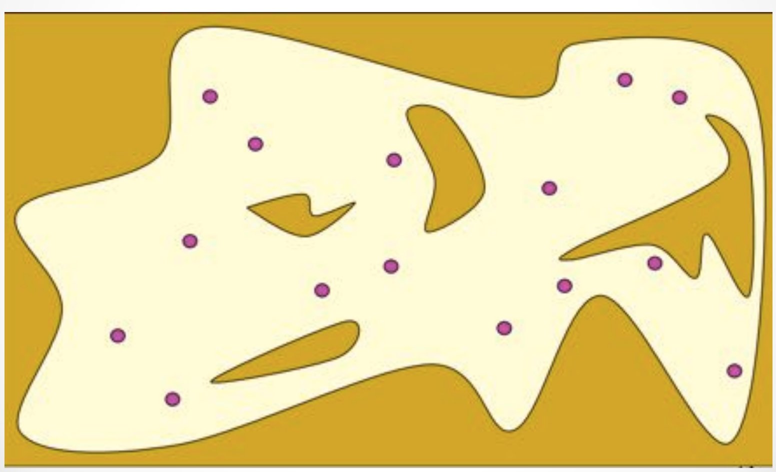


21

21

Probabilistic Roadmap (PRM)

- Collision-free states are saved as milestones

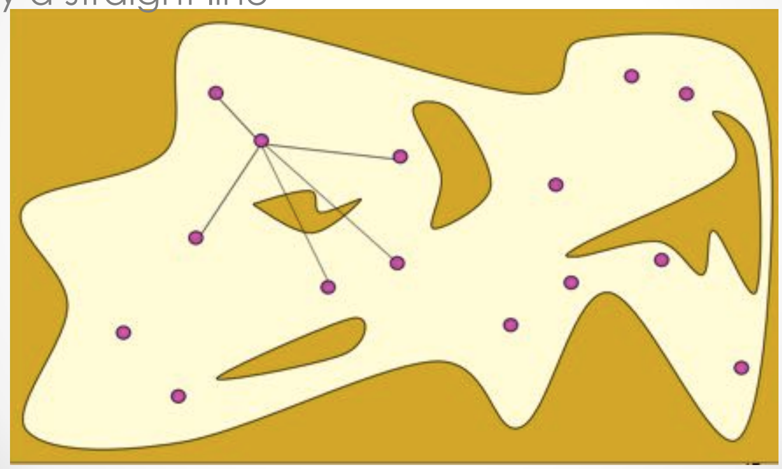


22

22

Probabilistic Roadmap (PRM)

- Each milestone is linked to its nearest neighbors by a straight line

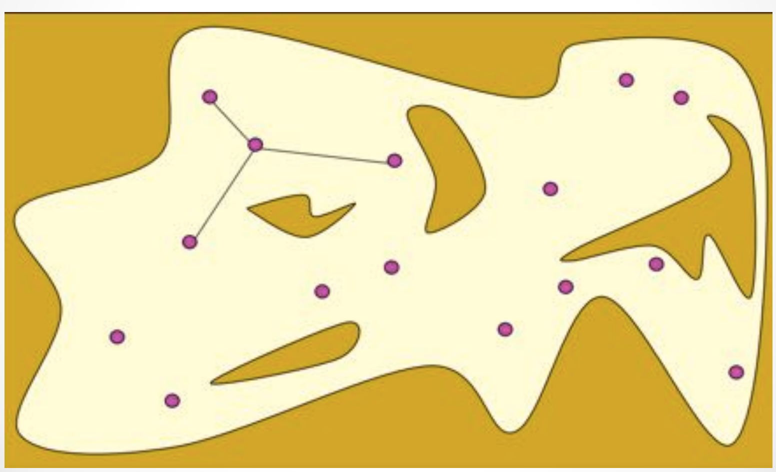


23

23

Probabilistic Roadmap (PRM)

- Straight lines connect neighboring milestones

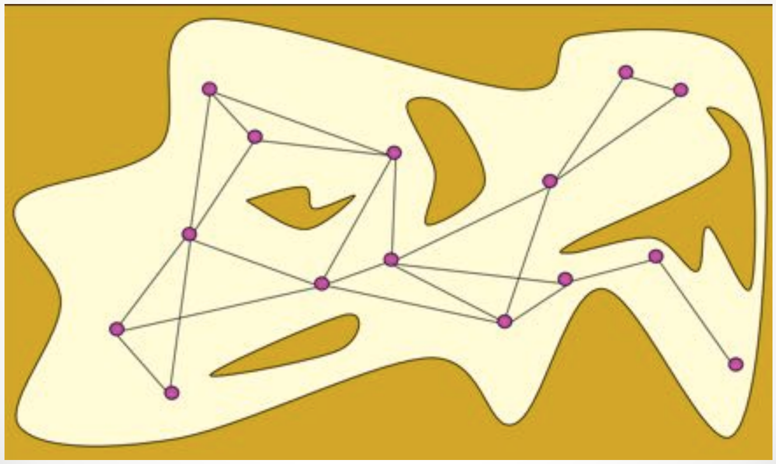


24

24

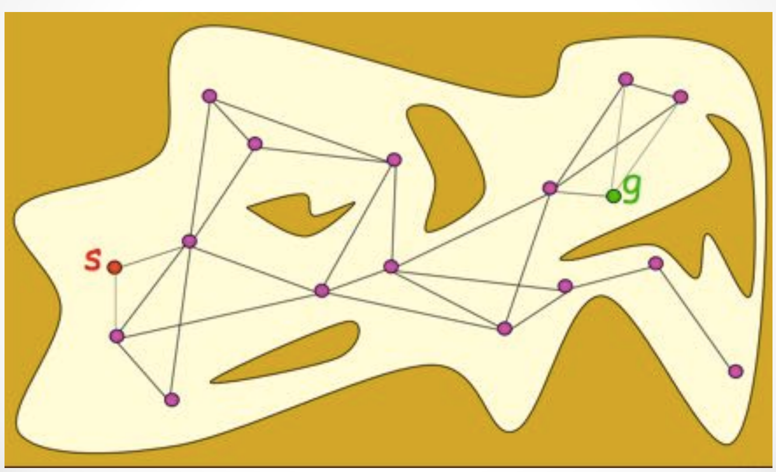
Probabilistic Roadmap (PRM)

- Collision-free links of local paths from PRM



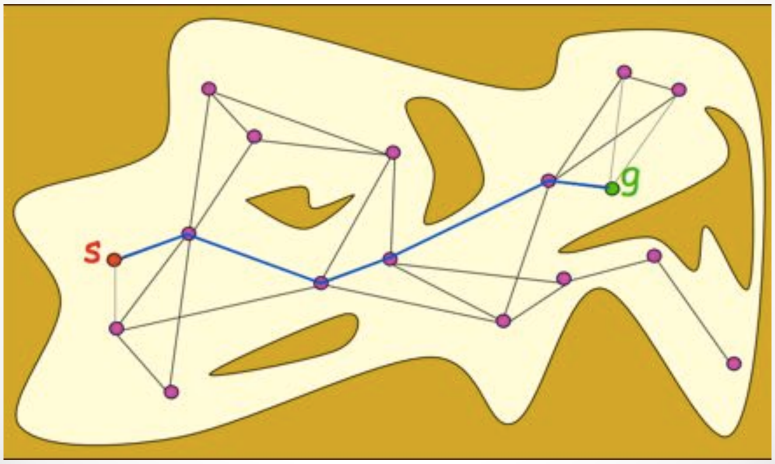
Probabilistic Roadmap (PRM)

- Start and goal retained as milestones



Probabilistic Roadmap (PRM)

- PRM searched for path from s to g

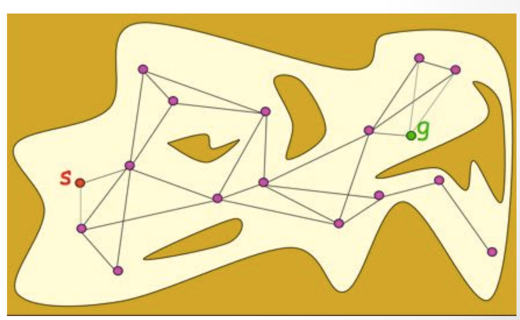


27

27

Finding Path in PRM

- Graph search algorithms
  - Dijkstra's / Uniform Cost Search
  - A*



28

28

PRM – Overview

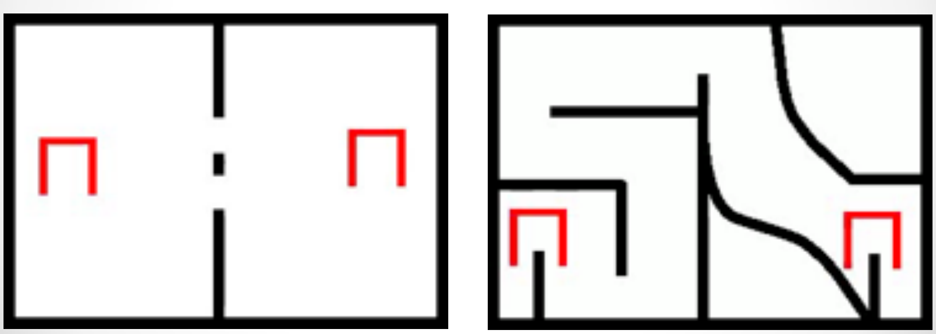
- Initialize sets of points with X_S and X_G
- Randomly sample points in configuration space
- Connect nearby points if they can be reached from each other
- Find path from X_S to X_G
  - o Alternatively, track connected components. Stop when X_S and X_G are connected

29

29

PRM - Examples

- Bruno Adorno - Automation and Robotics Laboratory (LARA) - University of Brasilia



30

30

PRM Difficulties

- Connecting neighboring points:
  - Only easy for holonomic systems
    - Systems where you can move in any direction from any state
  - General solution requires a Boundary Value Problem
    - E.g., Car, Tank, Bicycle

$$\begin{aligned} \min_{u,x} \quad & \|u\| \\ \text{s.t.} \quad & x_{t+1} = f(x_t, u_t) \quad \forall t \\ & u_t \in \mathcal{U}_t \\ & x_t \in \mathcal{X}_t \\ & x_0 = x_S \\ & X_T = x_G \end{aligned}$$

Typically solved without collision checking; later verified if valid by collision checking

31

31

PRM Challenges (cont.)

- Collision Checking
  - Determining if local path intersects c-space
  - Often the bottleneck in planning (see Lavalle book)
- Sampling
  - Sample uniformly at random
    - Same strategies as in stochastic animation
  - Bias sampling based on prior knowledge
    - E.g., more samples near narrow passages

32

32

PRM Analysis

- Advantages:
  - Probabilistically complete
    - Probability of finding a solution approaches one over time (if solution exists)
- Issues:
  - Required to solve 2-point boundary value problem
  - Graph is built over all state-space without focus on generating a path

33

33

Next Several Lectures

- Search & Maps
- Multi-agent planning
  - Centralizing
  - Distributed

34

34