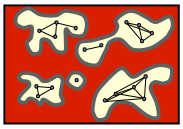
Last lecture

- Multiple-query PRM
- Expansive spaces

Uniform sampling

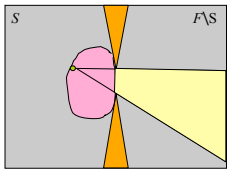
- All-pairs path planning



- **Theorem 1** : A roadmap of $\frac{16\ln(1/\gamma)}{\varepsilon\alpha}+\frac{6}{\beta}$ uniformly-sampled milestones has the correct connectivity with probability at least $1-\gamma$.

Definition: (ϵ,α,β)-expansive

- The free space F is $(\varepsilon,\alpha,\beta)$-expansive if
  - Free space F is ε-good
  - For each subset S of F, its β-lookout is at least α fraction of S. ε,α,β are in (0,1]



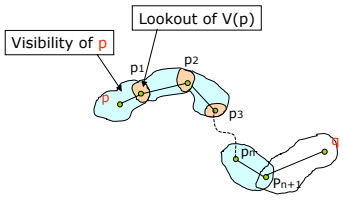
F is ε-good → ε=0.5

β-lookout → β=0.4

$\frac{\text{Volume}(\beta\text{-lookout})}{\text{Volume}(S)}$ → α=0.2

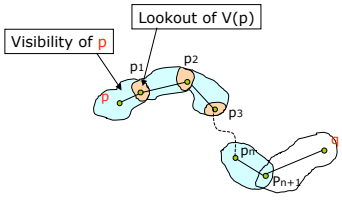
F is $(\varepsilon, \alpha, \beta)$-expansive, where ε=0.5, α=0.2, β=0.4.

Definition: Linking sequence



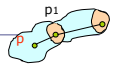
P_{n+1} is chosen from the lookout of the subset seen by p, $p_1,\ldots,p_n$

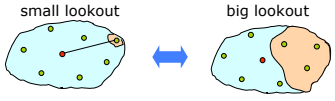
Definition: Linking sequence



P_{n+1} is chosen from the lookout of the subset seen by p, $p_1,\ldots,p_n$

Size of lookout set



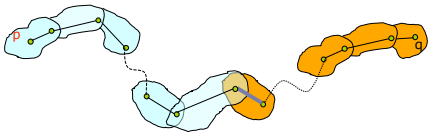


A C-space with larger lookout set has higher probability of constructing a linking sequence.

Lemmas

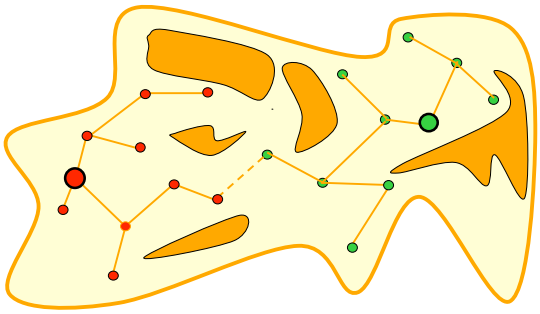
- In an expansive space with large ε, α, and β, we can obtain a linking sequence that covers a large fraction of the free space, with high probability.

Space occupied by linking sequences



Single-Query PRM

Probabilistic Roadmap (PRM): single query



Lazy PRM

- *Path Planning Using Lazy PRM*, R. Bohlin & L. Kavraki, 2000.

Precomputation: roadmap construction

- Nodes
  - Randomly chosen configurations, which may or may **not** be collision-free
  - No call to **CLEAR**
- Edges
  - an edge between two nodes if the corresponding configurations are close according to a suitable metric
  - no call to **LINK**

Query processing: overview

1. Find a shortest path in the roadmap
2. Check whether the nodes and edges in the path are collision-free.
3. If yes, then done. Otherwise, remove the nodes or edges in violation. Go to (1).

We either find a collision-free path, or exhaust all paths in the roadmap and declare failure.

Query processing: details

- Find the shortest path in the roadmap
  - A* algorithm
  - Dijkstra's algorithm
- Check whether nodes and edges are collisions free
  - **CLEAR**(q)
  - **LINK**(q_0, q_1)

Expansive space tree

- *Path Planning in Expansive Configuration Spaces*, D. Hsu, J.C. Latombe, & R. Motwani, 1999.

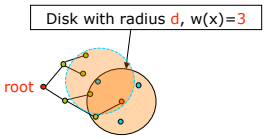
Overview

1. Grow two trees from Init and Goal configurations.
2. Randomly sample nodes around existing nodes.
3. Connect a node in the tree rooted at Init to a node in the tree rooted at the Goal.

Expansion + Connection

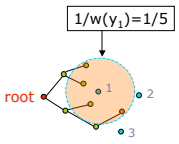
Expansion

1. Pick a node x with probability 1/w(x).
2. Randomly sample k points around x.
3. For each sample y, calculate w(y).



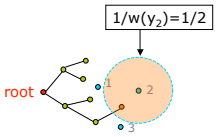
Expansion

1. Pick a node x with probability 1/w(x).
2. Randomly sample k points around x.
3. For each sample y, calculate w(y).



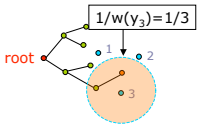
Expansion

1. Pick a node x with probability 1/w(x).
2. Randomly sample k points around x.
3. For each sample y, calculate w(y).



Expansion

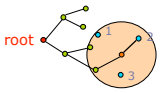
1. Pick a node x with probability 1/w(x).
2. Randomly sample k points around x.
3. For each sample y, calculate w(y).



Expansion

1. Pick a node x with probability 1/w(x).
2. Randomly sample k points around x.
3. For each sample y, calculate w(y). If y

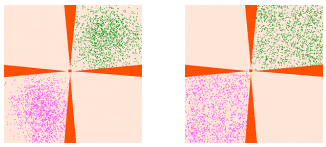
   (a) collision free; (b) can sees x,
   then add y into the tree.



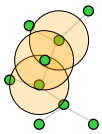
Sampling distribution

- Weight $w(x)$ = no. of neighbors
- Roughly $\Pr(x) \sim 1 / w(x)$

Effect of weighting



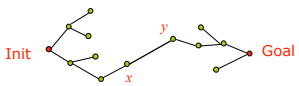
unweighted sampling weighted sampling



Connection

- If a pair of nodes (*i.e.*, x in Init tree and y in Goal tree) and distance(x,y)<L, check if
  x can see y

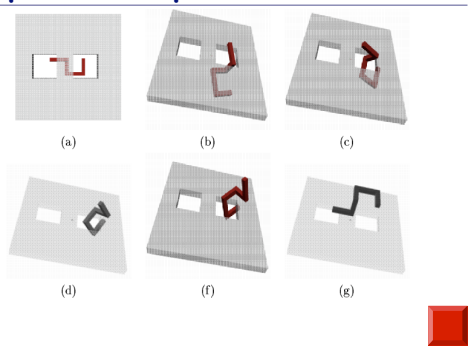
  YES, then connect x and y



Termination condition

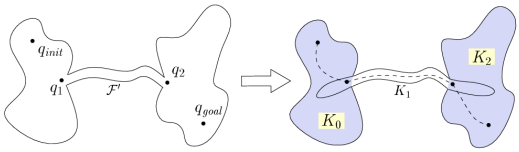
- The program iterates between **Expansion** and **Connection**, until
  - two trees are connected, or
  - max number of expansion & connection steps is reached

Computed example



Extensions

- Accelerate the planner by automatically generating intermediate configurations to decompose the free space into expansive components.



Extensions

- Accelerate the planner by automatically generating intermediate configurations to decompose the free space into expansive components.
- Use geometric transformations to increase the expansiveness of a free space, *e.g.*, widening narrow passages.

Extensions

- Accelerate the planner by automatically generating intermediate configurations to decompose the free space into expansive components.
- Use geometric transformations to increase the expansiveness of a free space, *e.g.*, widening narrow passages.
- Integrate the new planner with other planner for multiple-query path planning problems.

Questions?

Sampling a Point Uniformly at Random

Positions

- Unit interval
  Pick a random number from [0,1]

- Unit square

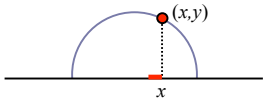
- Unit cube

Intervals scaled & shifted

- What shall we do?

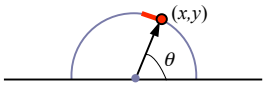
If x is a random number from $[0,1]$, then $7x$-2.

Orientations in 2-D



- Sampling
  - Pick x uniform at random from [-1,1]
  - Set $y = \sqrt{1-x^2}$
- Intervals of same widths are sampled with equal probabilities

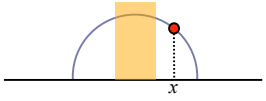
Orientations in 2-D



- Sampling
  - Pick θ uniformly at random from [0, 2π]
  - Set $x = \cos\theta$ and $y = \sin\theta$
- Circular arcs of same angles are sampled with equal probabilities.

What is the difference?

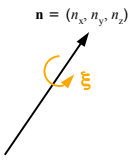
- Both are uniform in some sense.
- For sampling orientations in 2-D, the second method is usually more appropriate.



- The definition of uniform sampling depends on the task at hand and not on the mathematics.

Orientations in 3-D

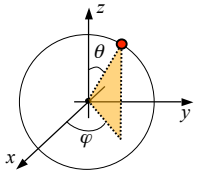
- Unit quaternion
  $(\cos\xi/2, n_x\sin\xi/2, n_y\sin\xi/2, n_z\sin\xi/2)$ with $n_x^2 + n_y^2 + n_z^2 = 1$.
- Sample $\mathbf{n}$ and ξ separately
- Sample ξ from [0, 2π] uniformly at random



Sampling a point on the unit sphere

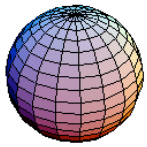
- Longitude and latitude

$$\begin{cases} n_x = \sin\theta\cos\varphi \\ n_y = \sin\theta\sin\varphi \\ n_z = \cos\theta \end{cases}$$



First attempt

- Choose θ and φ uniformly at random from [0, 2π] and [0, π], respectively.

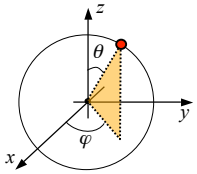


Better solution

- Spherical patches of same areas are sampled with equal probabilities.
- Suppose U_1 and U_2 are chosen uniformly at random from $[0,1]$.

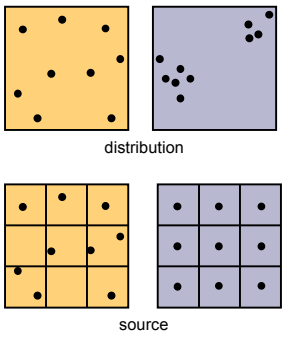
$$\begin{cases} n_z = U_1 \\ n_x = R\cos(2\pi U_2) \\ n_y = R\sin(2\pi U_2) \end{cases}$$

where $R = \sqrt{1-U_1^2}$



Sampling distribution vs. sampling source

- Sampling strategy
  - Distribution
    e.g., uniform, Gaussian
  - Source
    e.g., pseudo-random, quasi-random, lattice



Summary

```
repeat
   sample a configuration q with a suitable
   sampling strategy
   if q is collision-free then
     add q to the roadmap R
     connect q to existing milestones
return R
```

- Sampling strategy
- Connection strategy