Probabilistic Robot Path Planning

Manuela M. Veloso

Carnegie Mellon University
Computer Science Department

15-780 Graduate AI – Spring 2013

Readings:

- Real-time randomized path planning for robot navigation, James Bruce and Manuela Veloso. In *Proceedings of IROS-2002.*

Robot Motion Planning

- A mobile robot needs to *navigate:*
  - Navigation is carrying out locomotion primitives to move between points
  - Navigation includes avoiding obstacles.
- We need to define:
  - The state – a model of the environment
  - The actions – a model of the robot's motion primitives

Deterministic Path Planning

- A*
  - Discretize the state
  - Enumerate a set of actions
  - Search
    - Generate successors of states
    - Use admissible heuristic
  - Partially successful
- Extensions of A*
  - D*, etc

Probabilistic Path Planning

- Continuous state spaces
- Continuous actions

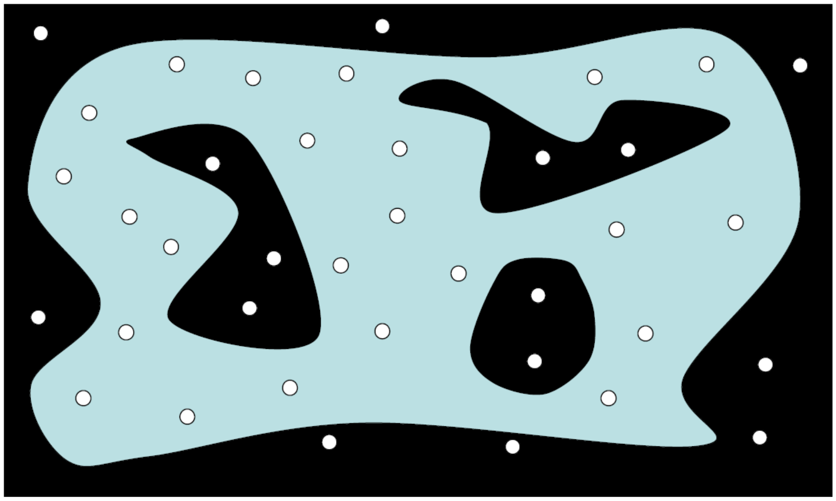
- PRM (Kavraki & many successors)
- RRT (Lavalle & many successors)
  - ERRT (Bruce & many other variations)

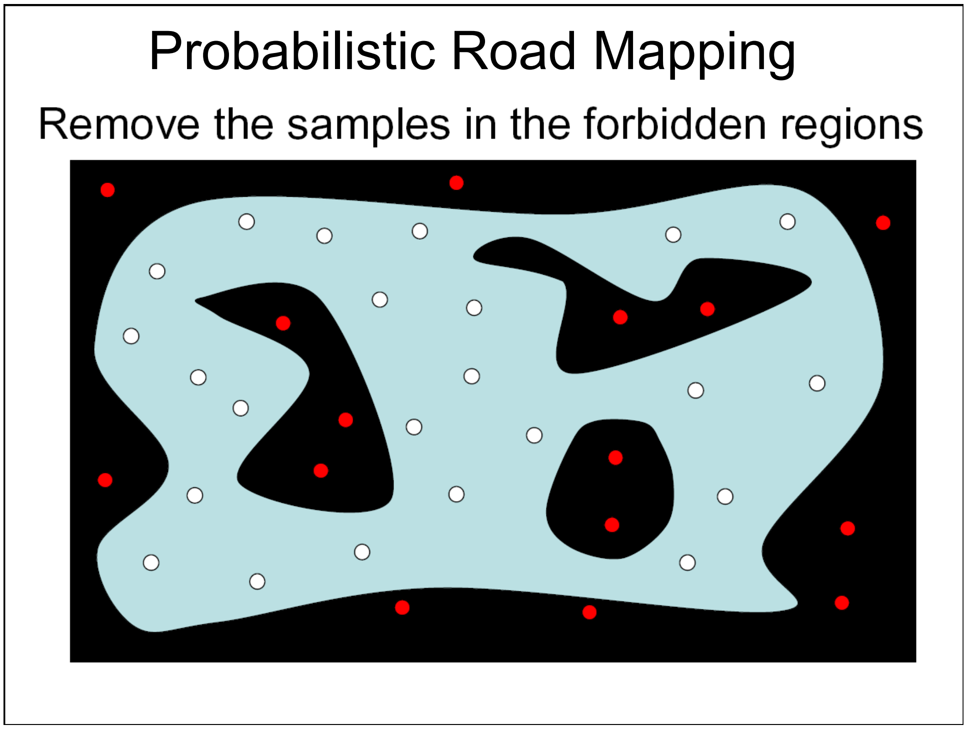
PRM – Probabilistic Roadmap

- Separate planning into two stages
  - “Learning” Phase
    - random samples of free configurations (vertices)
    - Attempt to connect pairs of nearby vertices with a local planner
    - if a valid plan is found, add an edge to the graph
  - Query Phase
    - find local connections to graph from initial and goal positions

Probabilistic Road Mapping

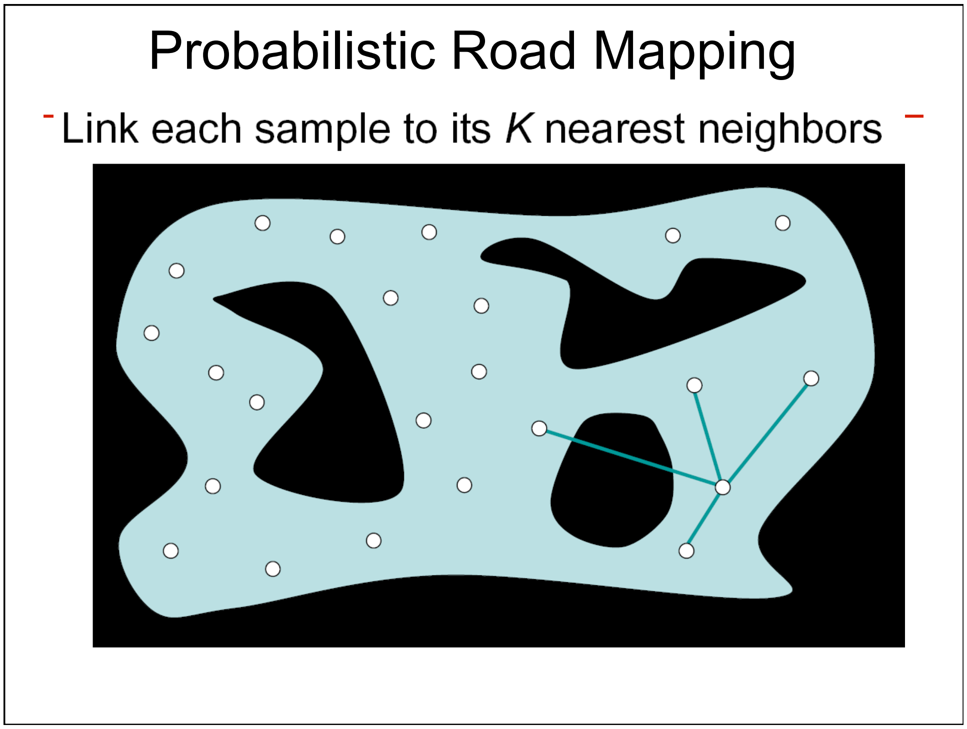
Sample random locations





Probabilistic Road Mapping

Remove the samples in the forbidden regions

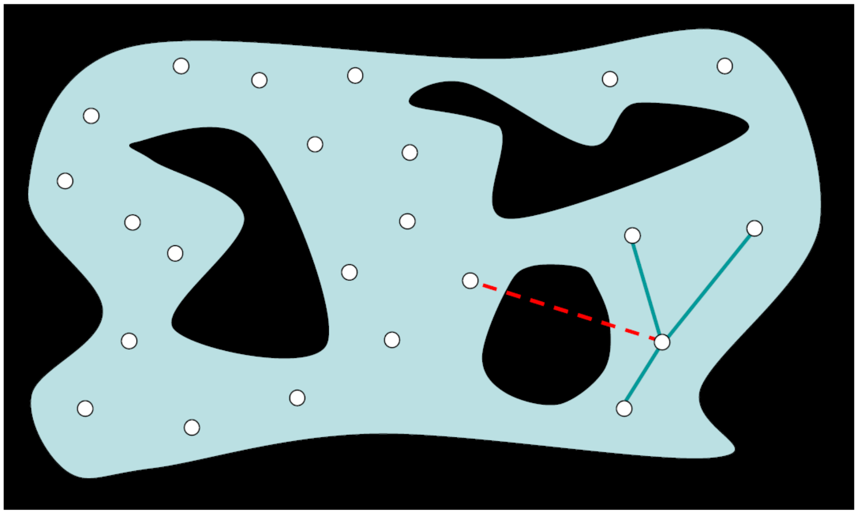


Probabilistic Road Mapping

Link each sample to its *K* nearest neighbors

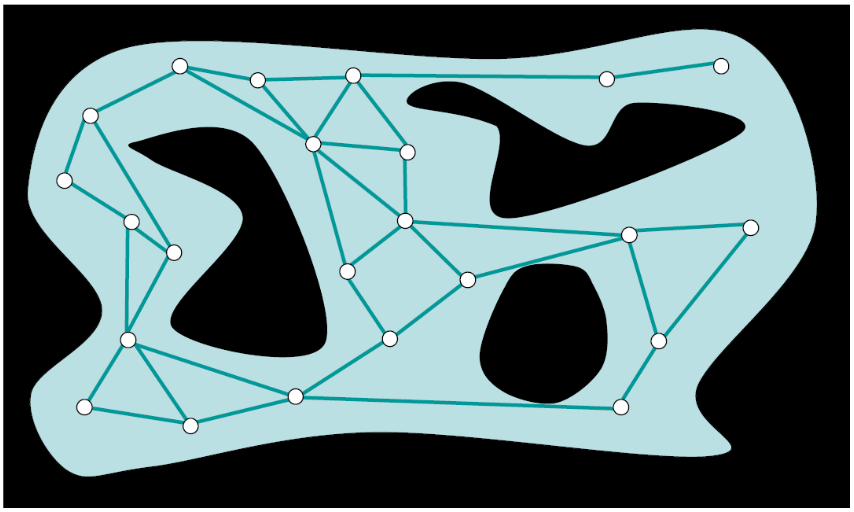
Probabilistic Road Mapping

Remove the links that cross forbidden regions



Probabilistic Road Mapping

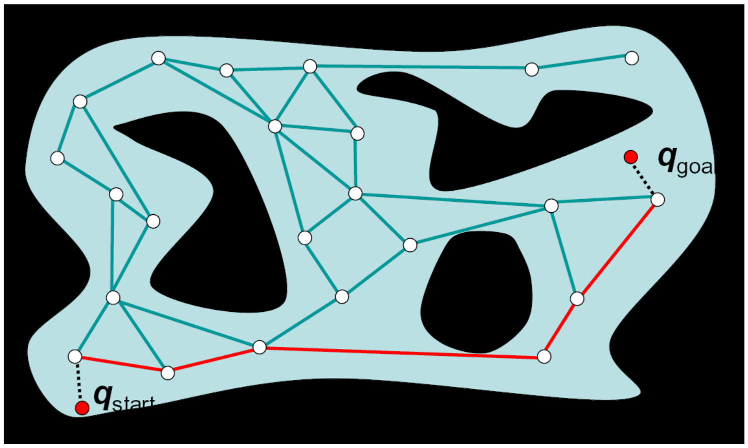
Remove the links that cross forbidden regions



The resulting graph is a *probabilistic roadmap (PRM)*

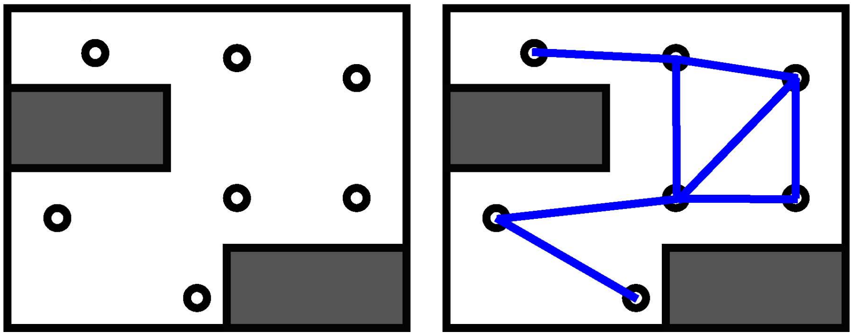
Probabilistic Road Mapping

Link the start and goal to the PRM and search using A*



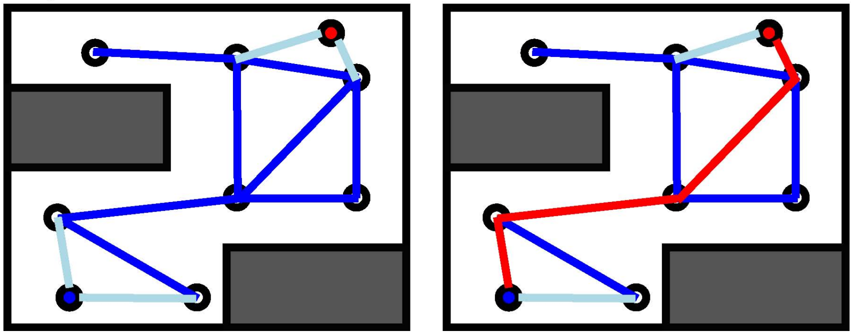
PRM Example – Learning Phase

Learning Phase:



PRM Example – Query Phase

Query Phase:



PRM Discussion

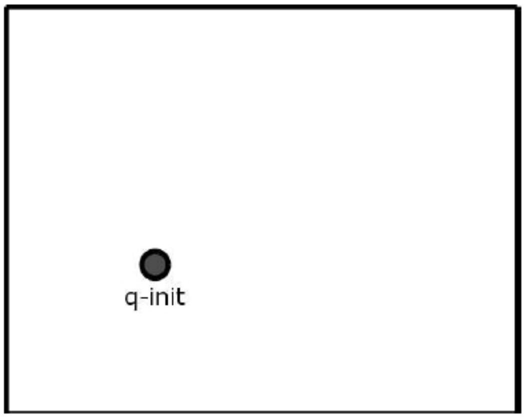
- Very interesting approach
  - Continuous spaces
- General learning phase
  - Not targeted at specific initial and goal states
- Not optimal path generated – probabilistically optimal
- How to pick sample configurations?
  - Uniform – simple, but may be slow
  - Non-uniform – (probabilistically) choose configurations that are in the neighborhood of nodes that are unconnected
    - Weight nodes by the “difficulty” of the region
- Efficient local planner to connect robot position to PRM samples

Rapidly Exploring Random Trees

- RRT
  - Explore continuous spaces efficiently
    - No need for an artificial grid
  - Basic for probabilistically complete planner
- RRT uses random search

Basic RRT Example

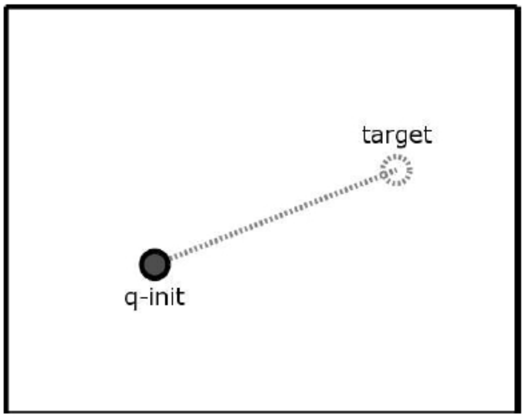
(1) Start with the initial state as the root of a tree



Basic RRT
Just Search, No use of Goal

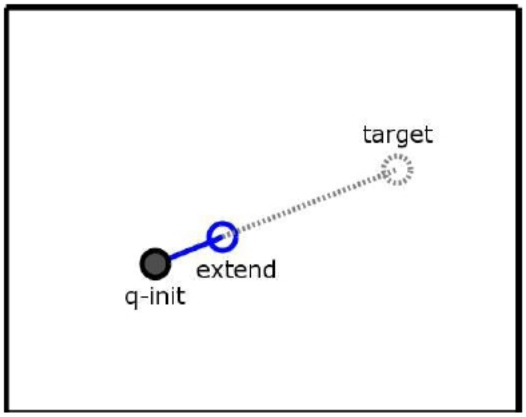
(2) Pick a random state in the environment

(3) Find the closest node in the tree

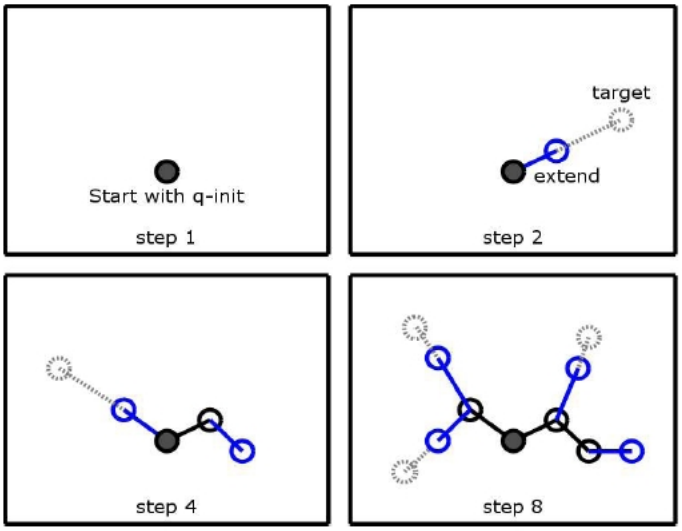


Basic RRT Search

(4) Extend that node toward the target

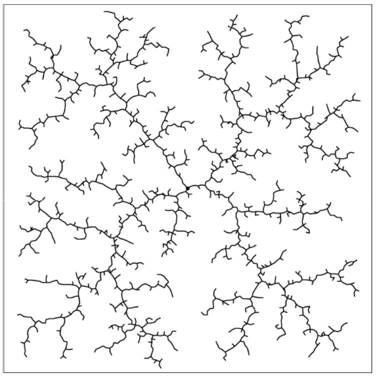


Basic RRT Search Example



RRT Basic Algorithm

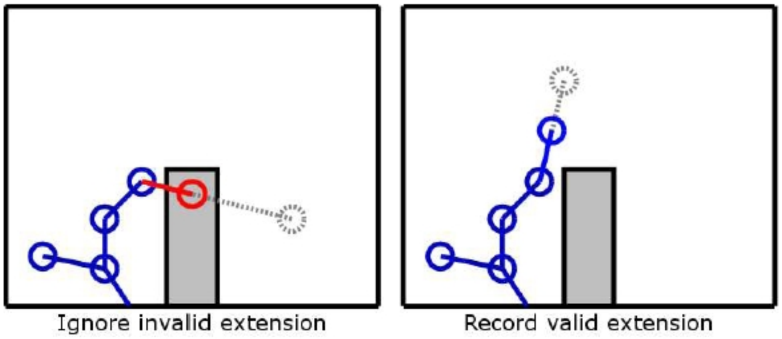
Pick a random point in the search space
Try to connect the current tree to that point
Continue until goal is reached (or stuck)



- Pick point ***p*** stochastically
- Find nearest node ***q*** in search tree
- Extend search from ***q*** in direction of point ***p***
- Continue until goal reached or limit on number of nodes
- If time (or goal not reached) can retry with different random seed

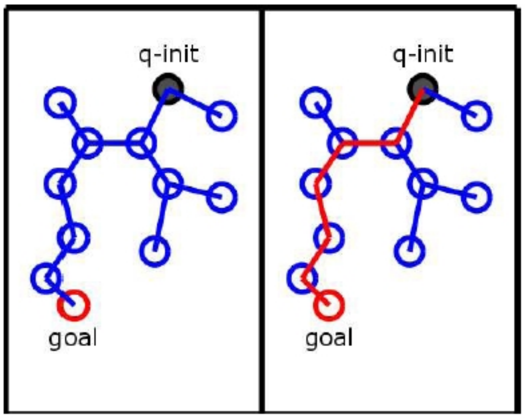
RRT with Obstacles

- Ignore extensions which hit obstacles
- Resulting tree contains *only* valid paths



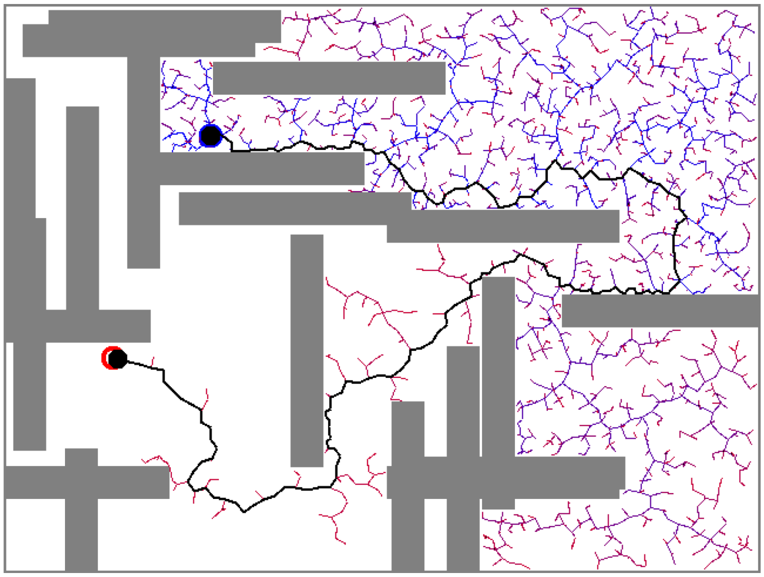
RRT As a Planner

- Once a node of the tree is a *goal*, the plan is the path back up the tree



RRT-GoalBias Algorithm

1) Start with initial state as root of tree
2) Pick a random target state
   - Goal configuration with probability p
   - Random configuration with probability 1-p
3) Find the closest node in the tree
4) Extend the closest node toward the target
5) Goto step 2



ERRT – RRT with Replanning

(Bruce & Veloso 2002)

Introduce past path(s) as a bias!

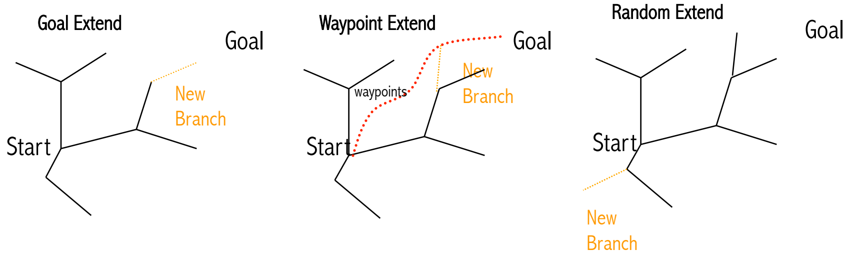
1) Start with initial state as root of tree
2) Pick a random target state
   - o Goal configuration with probability p
   - o **Random item from waypoint cache with probability q**
   - o Random configuration with probability 1-q-p
3) Find the closest node in the tree
4) Extend the closest node toward the target
5) Goto step 2

ERRT: Replanning with Advice

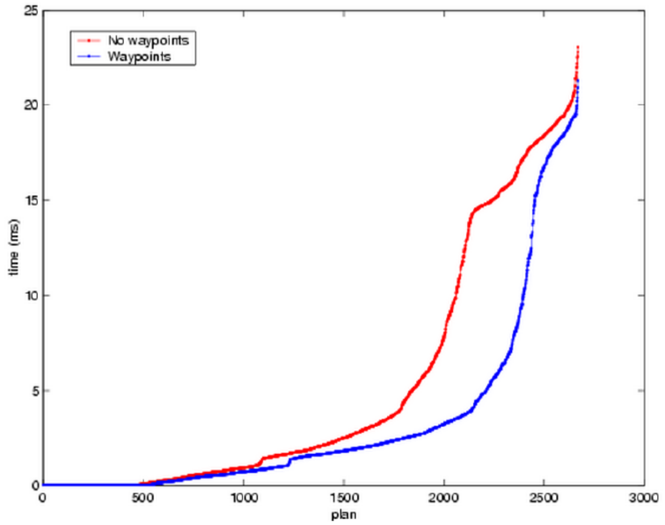
Probability p : Extend closest node in tree towards goal

Probability q : Extend closest node in tree towards random cache point

Probability $1\text{-}p\text{-}q$: Extend closest node towards a random point



Waypoints [Results]



Planning and Replanning

- Environments and planning
  - Value of p?
- Dynamic environments
- When failure, what to do?

Other RRT Variations

- Use any heuristics to guide RRT (Urmson)
- Model uncertainty with particles (Melchior)
- Balanced growth sampling (Zickler)

Summary

- PRM
  - Sampling and search among sample nodes
- Planning with RRT
  - Extend towards random target, or towards goal
  - High p – few known obstacles
  - Low p – many known obstacles
- Replanning with ERRT
  - Extend towards random target, goal, or past plan
  - High q – small dynamics (no state change)
  - Low q – high dynamics (lots of state change)
  - ERRT – bias to use previous plan; but could be any other bias
- RRT and ERRT – probabilistic convergence