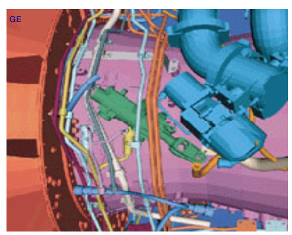
CS 603 - Path Planning

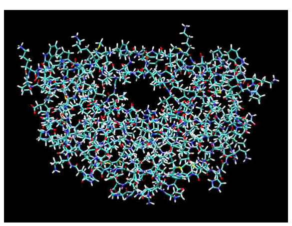
Rod Grupen

Why Path Planning?

Why Motion Planning?

Virtual Prototyping
Character Animation
Structural Molecular Biology
Autonomous Control





Origins of Motion Planning

- T. Lozano-Pérez and M.A. Wesley: "An Algorithm for Planning Collision-Free Paths Among Polyhedral Obstacles," 1979.
- introduced the notion of configuration space (c-space) to robotics
- many approaches have been devised since then in configuration space

Completeness of Planning Algorithms

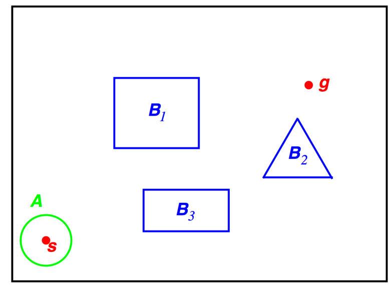
a **complete planner** finds a path if one exists

resolution complete – complete to the model resolution

probabilistically complete

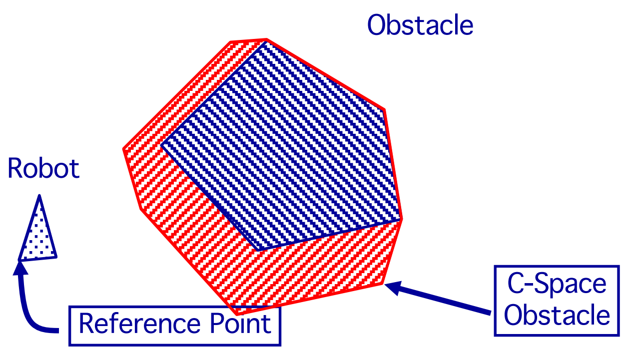
Representation

...given a moving object, *A*, initially in an unoccupied region of freespace, ***s***, a set of stationary objects, B_i, at known locations, and a goal position, ***g***, ...



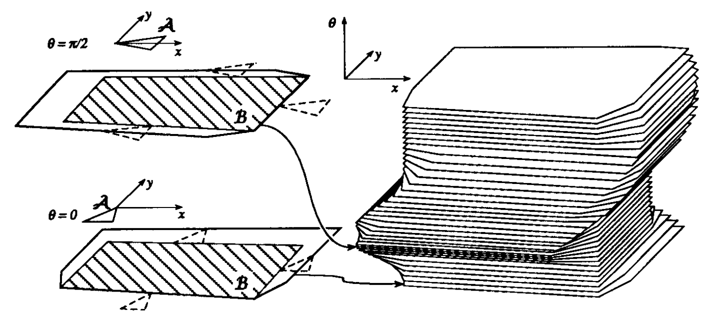
find a sequence of collision-free motions that take *A* from ***s*** to ***g***

Mapping to Configuration Space - Translational Case (fixed orientation)



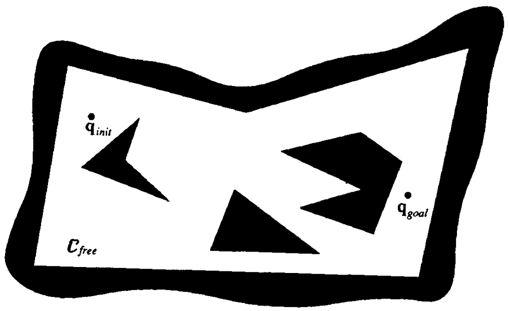
changing θ ?

Obstacles in 3D (x,y,θ)

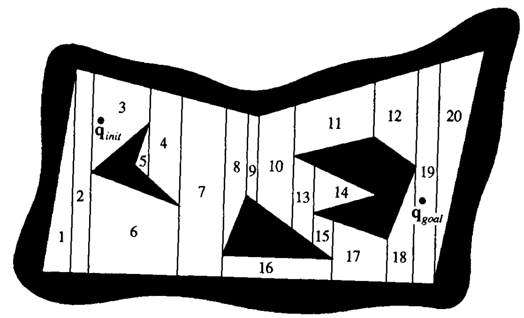


Jean-Claude Latombe

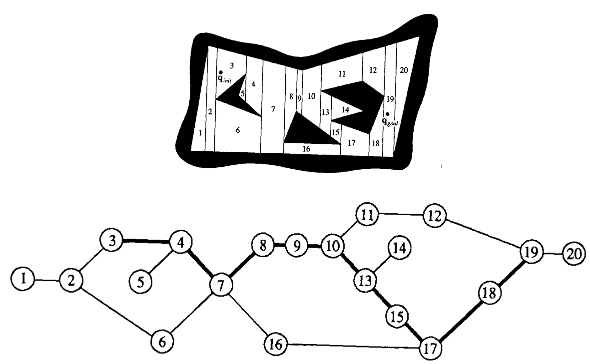
Exact Cell Decomposition



Exact Cell Decomposition

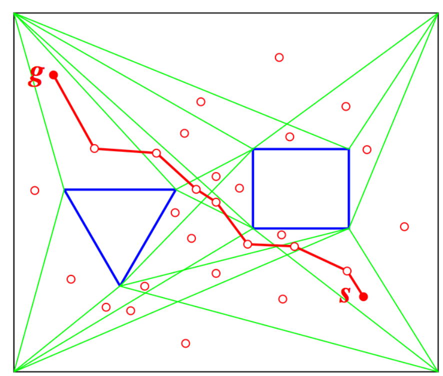


Exact Cell Decomposition



Jean-Claude Latombe

Representation – Simplicial Decomposition

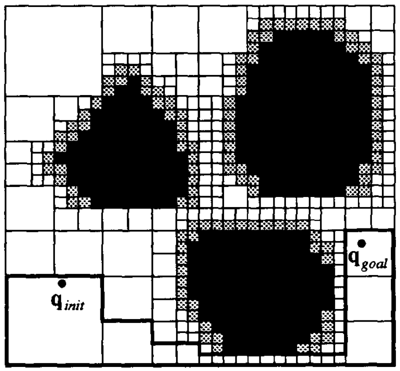


Schwartz and Sharir

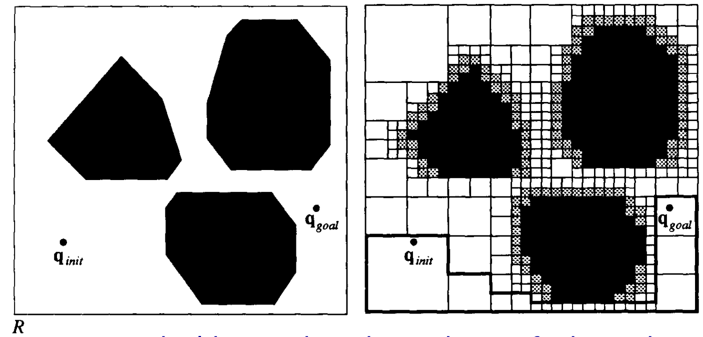
Lozano-Perez

Canny

Approximate Methods: 2^n-Tree



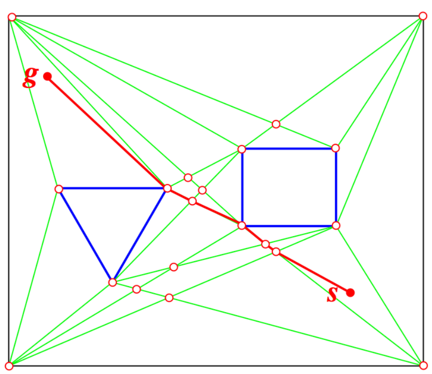
Approximate Cell Decomposition



again…build a graph and search it to find a path

Jean-Claude Latombe

Representation – Roadmaps

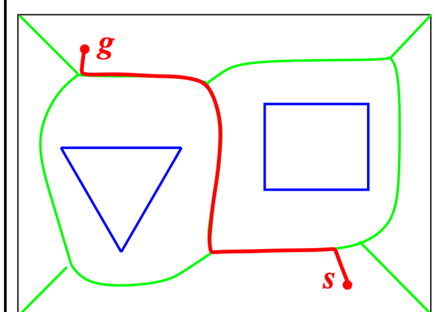


Visibility diagrams:

unsmooth

sensitive to error

Roadmap Representations



Voronoi diagrams

a "*retraction*"

…the continuous freespace is represented as a network of curves…

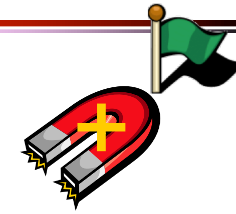
Jean-Claude Latombe

Summary

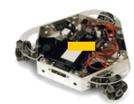
- Exact Cell Decomposition
- Approximate Cell Decomposition
  - graph search
  - next: potential field methods
- Roadmap Methods
  - visibility graphs
  - Voronoi diagrams
  - next: probabilistic road maps (PRM)

state of the art techniques

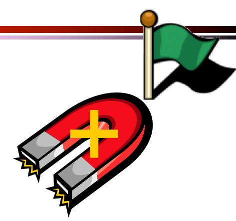
Attractive Potential Fields



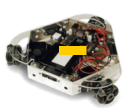
Oliver Brock



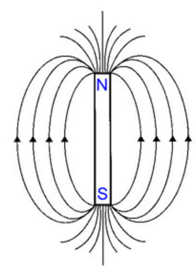
Repulsive Potentials



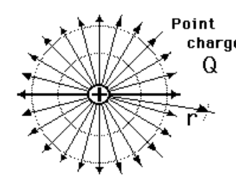
Oliver Brock



Electrostatic (or Gravitational) Field



depends on direction

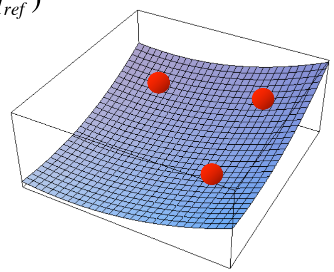


$$\mathbf{F} = \frac{k \cdot q_1 \cdot q_2}{r^2}$$

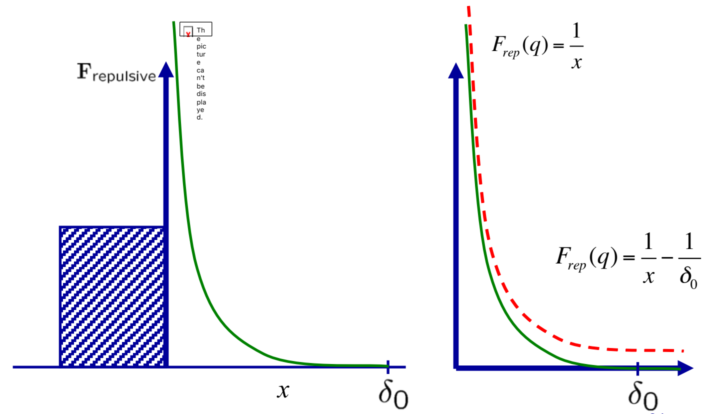
Attractive Potential

$$\phi_{\text{att}}(q) = \frac{1}{2} k \, (q - q_{ref})^T (q - q_{ref})$$

$$F_{\text{att}}(q) = -\nabla \phi_{\text{att}}(q)$$
$$= -k \, (q - q_{ref})$$



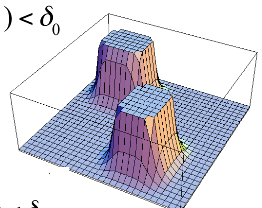
A Repulsive Potential



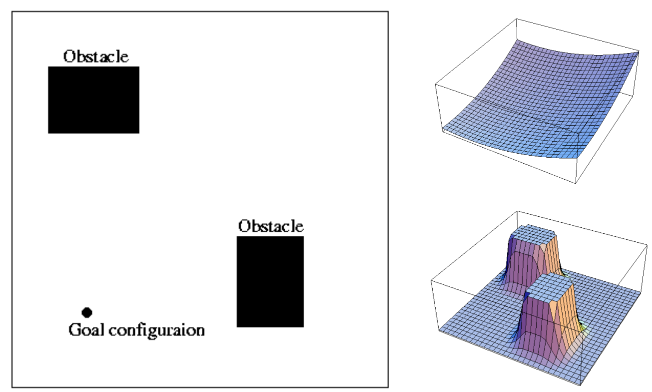
Repulsive Potential

$$\phi_{\text{rep}}(q) = k \left(\frac{1}{(q - q_{obst})} - \frac{1}{\delta_0} \right)^2 \quad if((q - q_{obs}) < \delta_0$$
$$= 0 \quad otherwise$$

$$F_{\text{rep}}(q) = -\nabla \phi(q)$$
$$= \begin{cases} -k \left(\frac{1}{(q - q_{obs})} - \frac{1}{\delta_0} \right) & if((q - q_{obs}) < \delta_0 \\ 0 & otherwise \end{cases}$$

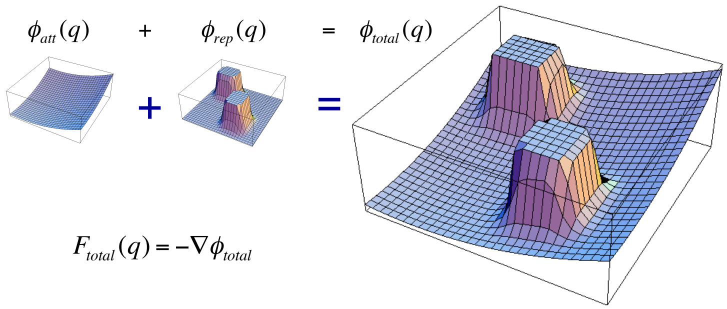


Sum Attractive and Repulsive Fields



Artificial Potential Function

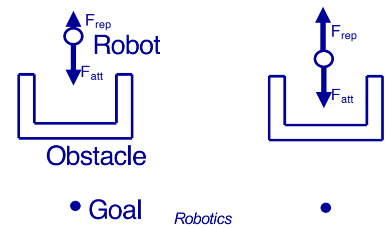
$\phi_{att}(q) + \phi_{rep}(q) = \phi_{total}(q)$



$$F_{total}(q) = -\nabla\phi_{total}$$

Potential Fields

- Goal: avoid local minima
- Problem: requires global information
- Solution: **Navigation Function**



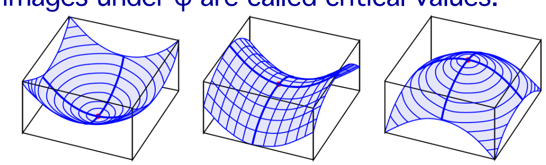
Navigation Functions

Analyticity – navigation functions are analytic because they are infinitely differentiable and their Taylor series converge to $\phi(q_0)$ as q approaches q_0

Polar – gradients (streamlines) of navigation functions terminate at a unique minima

Navigation Functions

Morse - navigation functions have no degenerate critical points where the robot can get stuck short of attaining the goal. Critical points are places where the gradient of ϕ vanishes, i.e. minima, saddle points, or maxima and their images under ϕ are called critical values.



Admissibility - practical potential fields must always generate bounded torques

The Hessian

multivariable control function, $f(q_0, q_1, \ldots, q_n)$

$$\frac{\partial^2 f}{\partial \mathbf{Q}^2} = \begin{bmatrix} \frac{\partial^2 f}{\partial q_1^2} & \frac{\partial^2 f}{\partial q_1 \partial q_2} & \cdots & \frac{\partial^2 f}{\partial q_1 \partial q_n} \\ & \vdots & \vdots & \\ \frac{\partial^2 f}{\partial q_n \partial q_1} & \frac{\partial^2 f}{\partial q_n \partial q_2} & \cdots & \frac{\partial^2 f}{\partial q_n^2} \end{bmatrix}$$

if the Hessian is positive semi-definite over the domain Q, then the function f is *convex* over Q

Harmonic Functions

if the trace of the Hessian (the Laplacian) is 0

$$\nabla^2 \phi = \frac{d^2\phi}{dx_1^2} + \ldots + \frac{d^2\phi}{dx_n^2} = 0$$

then function ϕ is a *harmonic* function

laminar fluid flow, steady state temperature distribution, electromagnetic fields, current flow in conductive media

Properties of Harmonic Functions

$$\nabla^2 \phi = \frac{d^2\phi}{dx_1^2} + \ldots + \frac{d^2\phi}{dx_n^2} = 0$$

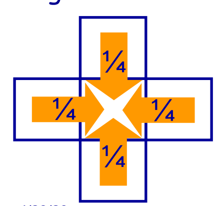
Min-Max Property -

...in any compact neighborhood of freespace, the minimum and maximum of the function must occur on the boundary.

Properties of Harmonic Functions

$$\nabla^2 \phi = \frac{d^2\phi}{dx_1^2} + \ldots + \frac{d^2\phi}{dx_n^2} = 0$$

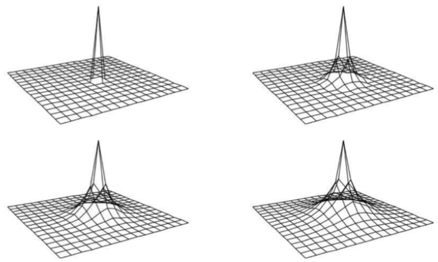
Mean-Value - up to truncation error, the value of the harmonic potential at a point in a lattice is the average of the values of its 2n Manhattan neighbors.



analog & numerical methods

Numerical Relaxation

Jacobi iteration



$$u^{(k+1)}(x_i, y_j) = \frac{1}{4}\Big(u^{(k)}(x_{i+1}, y_j) + u^{(k)}(x_{i-1}, y_j) + u^{(k)}(x_i, y_{j+1}) + u^{(k)}(x_i, y_{j-1})\Big)$$

Harmonic Relaxation: Numerical Methods

Gauss-Seidel

$$u^{(k+1)}(x_i, y_j) = \frac{1}{4}\left[u^{(k)}(x_{i+1}, y_j) + u^{(k+1)}(x_{i-1}, y_j) + u^{(k)}(x_i, y_{j+1}) + u^{(k+1)}(x_i, y_{j-1})\right]$$

Successive Over Relaxation

$$u^{(k+1)}(x_i, y_j) = u^{(k)}(x_i, y_j) + \frac{\omega}{4}\left[u^{(k)}(x_{i+1}, y_j) + u^{(k+1)}(x_{i-1}, y_j) + u^{(k)}(x_i, y_{j+1}) + u^{(k+1)}(x_i, y_{j-1}) - 4u^{(k)}(x_i, y_j)\right]$$

Properties of Harmonic Functions

$$\nabla^2 \phi = \frac{d^2\phi}{dx_1^2} + \ldots + \frac{d^2\phi}{dx_n^2} = 0$$

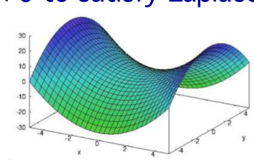
Hitting Probabilities - if we denote p(x) at state x as the probability that starting from x, a random walk process will reach an obstacle before it reaches a goal—p(x) is known as the hitting probability

greedy descent on the harmonic function minimizes the hitting probability.

Minima in Harmonic Functions

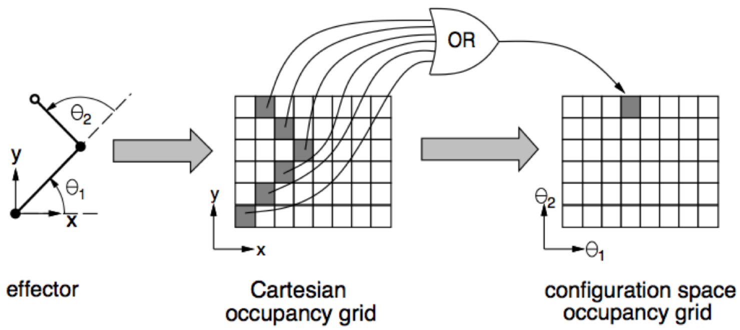
$$\nabla^2 \phi = \frac{d^2\phi}{dx_1^2} + \ldots + \frac{d^2\phi}{dx_n^2} = 0$$

for some i, if $\partial^2\phi/\partial x_i^2 > 0$ (concave upward), then there must exist another dimension, j, where $\partial^2\phi/\partial x_j^2 < 0$ to satisfy Laplace's constraint.



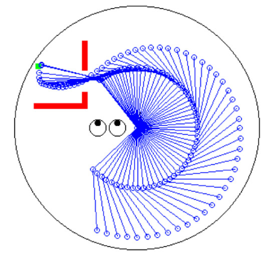
therefore, if you're not at a goal, there is always a way downhill... ...there are no local minima...

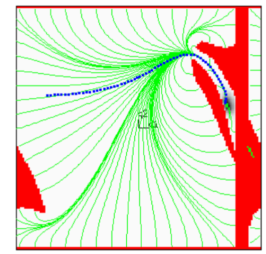
Configuration Space



Harmonic Functions for Path Planning

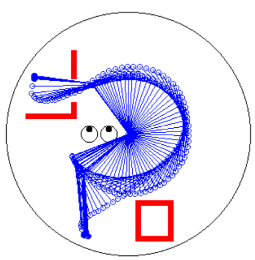
$$\nabla^2 \phi = \frac{d^2\phi}{dx_1^2} + \ldots + \frac{d^2\phi}{dx_n^2} = 0$$

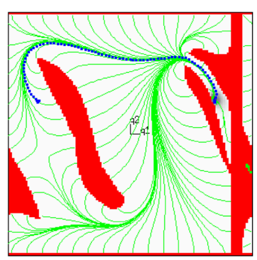




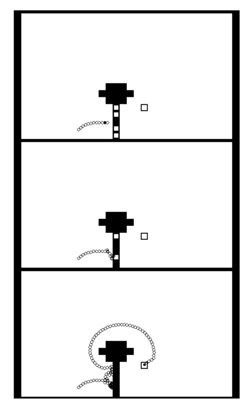
Harmonic Functions for Path Planning

$$\nabla^2 \phi = \frac{d^2\phi}{dx_1^2} + \ldots + \frac{d^2\phi}{dx_n^2} = 0$$





Reactive Admittance Control



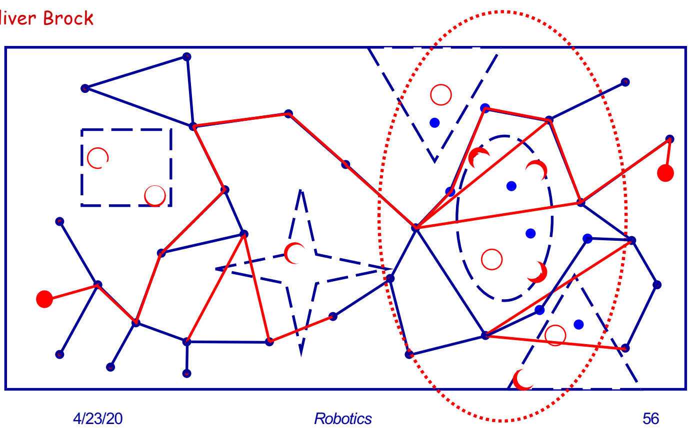
ok, back to graphical methods…

Probabilistic Roadmaps (PRM)

- Construction
  - Generate random configurations
  - Eliminate if they are in collision
  - Use local planner to connect configurations
- Expansion
  - Identify connected components
  - Resample gaps
  - Try to connect components
- Query
  - Connect initial and final configuration to roadmap
  - Perform graph search

Probabilistic Roadmaps (PRM)

Oliver Brock



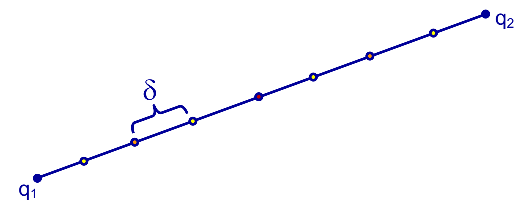
Sampling Phase

- Construction
  - R = (V,E)
  - repeat *n* times:
    - generate random configuration
    - add to V if collision free
    - attempt to connect to *neighbors* using *local planner*, unless in same connected component of R

Path Extraction

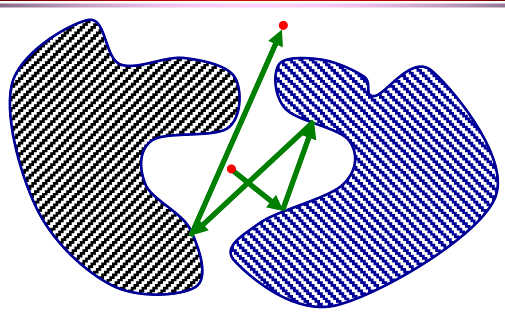
- Connect start and goal configuration to roadmap using local planner
- Perform graph search on roadmap
- Computational cost of searching negligible compared to construction of roadmap

Local Planner



tests up to a specified resolution δ!

Another Local Planner



perform random walk of predetermined length;
choose new direction randomly after hitting obstacle;
attempt to connect to roadmap after random walk

Summary: PRM

- Algorithmically very simple
- Surprisingly efficient even in high-dimensional C-spaces
- Capable of addressing a wide variety of motion planning problems
- One of the hottest areas of research
- Allows probabilistic performance guarantees

Variations of the PRM

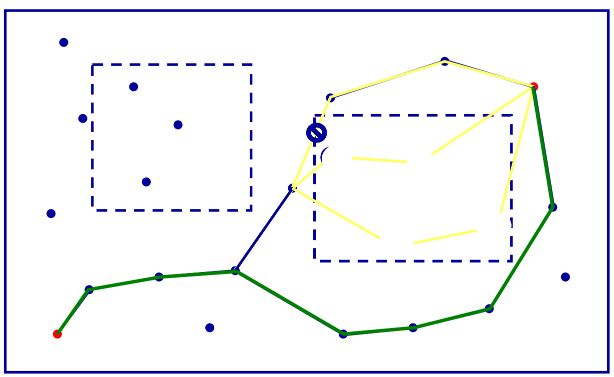
- Lazy PRMs
- Rapidly-exploring Random Trees

Lazy PRM

observation: pre-computation of roadmap takes a long time and does not respond well in dynamic environments

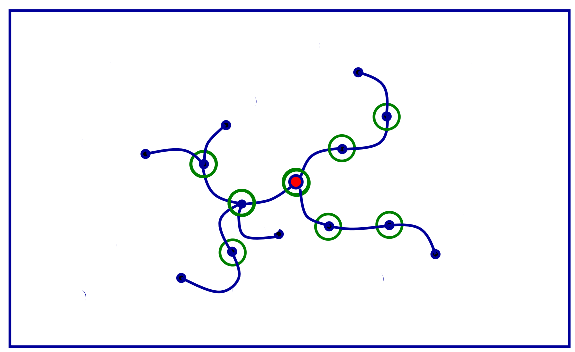
Lazy PRM

Oliver Brock



Rapidly-Exploring Random Trees (RRT)

Oliver Brock



Rapidly-Exploring Random Trees (RRT)

Steven LaValle