Sampling-Based Robot Motion Planning

Lydia Kavraki
Department of Computer Science
Rice University
Houston, TX
USA

Motion planning: classical setting

Go from Start to Goal without collisions and while respecting all robot constraints.

Motion Planning

- Is done in a continuous world and with constrained motions.
- Needs to know robot and world geometry.
- Needs to know robot and world physics.
- Must be accurate and predictive to work in practice.

Some notes:

- More powerful motion planning simplifies the task planner.
- More accurate motion planning simplifies motion execution.
- Motion planning is limited by model accuracy and complexity.

Motion planner is part of a replanning loop

Bekris et al.

Motion planning is hard

Problem	Complexity
Sofa Mover (3 DOF)	$O(n^{2+\varepsilon})$ not implemented
Piano Mover (6 DOF)	Polynomial – no practical algorithm known
n Disks in the Plane	NP-hard
n Link Planar Chain	PSPACE-Complete
Generalized Mover	PSPACE-Complete
Shortest Path for a Point in 3D	NP-hard
Curvature Constrained Point in 2D	NP-hard
Simplified Coulomb Friction	Undecidable

Exact, approximate, and heuristic methods

Method	Advantage	Disadvantage
Exact	theoretically insightful	impractical
Cell Decomposition	easy	does not scale
Control-Based	online, very robust	requires good trajectory
Potential Fields	online, easy	slow or fail
Sampling-based	fast and effective	cannot recognize impossible query

Exact, approximate, and heuristic methods

Method	Advantage	Disadvantage
Exact	theoretically insightful	impractical
Cell Decomposition	easy	does not scale
Control-Based	online, very robust	requires good trajectory
Potential Fields	online, easy	slow or fail
$\boxed{\text{Sampling-based}}$	fast and effective	cannot recognize impossible query

Outline of this talk

- Basic concepts and definitions.
- Examples of sampling-based planners:
  - Roadmap planner
  - Tree-based planner
- Underlying key components.
- OMPL and future challenges in motion planning.

Outline of this talk

- Basic concepts and definitions.
- Examples of sampling-based planners:
    - Roadmap planner
    - Tree-based planner
- Underlying key components.
- OMPL and future challenges in motion planning.

Basic concepts and definitions

- Workspace
- Robot
- State space
- Path
- Planning Instance/Simple Setup
- Query/Problem

Workspace

- The workspace is the environment that the robot operates in.
- The boundary of the workspace determines the obstacles.

Robot

- The robot is defined by:
  - Geometry
  - Parameters or Degrees of Freedom (DOF)
    - Different settings for the parameters embed the geometry in different ways into the workspace.

State space

- The parameter space for the robot is called the state space S.

- A point in this space is a state.

Free state space

- A state is free if the corresponding embedding of the robot's geometry lies in the workspace.

- The subspace of free configurations is free state space S_{free}.

- S_{free} can be very complex even for seemingly simple systems.

- This complexity is the main difficulty in motion planning.

Paths

- A path is continuous mapping in C

$$\pi : [0, L] \rightarrow S_{free}$$

- L is the length of the path.

- The path is collision-free if for all t

$$\pi(t) \in S_{free}$$

Planning instance/Simple setup

- A planning instance consists of:
  - Robot (S-space and embedding).
  - Workspace.
  - Constraints.

Query/Problem definition

- A problem or query is
  - Given two states, q_0 and q_f.

PROBLEM:

Determine if there is a collision-free path between q_0 and q_f.

Outline of this talk

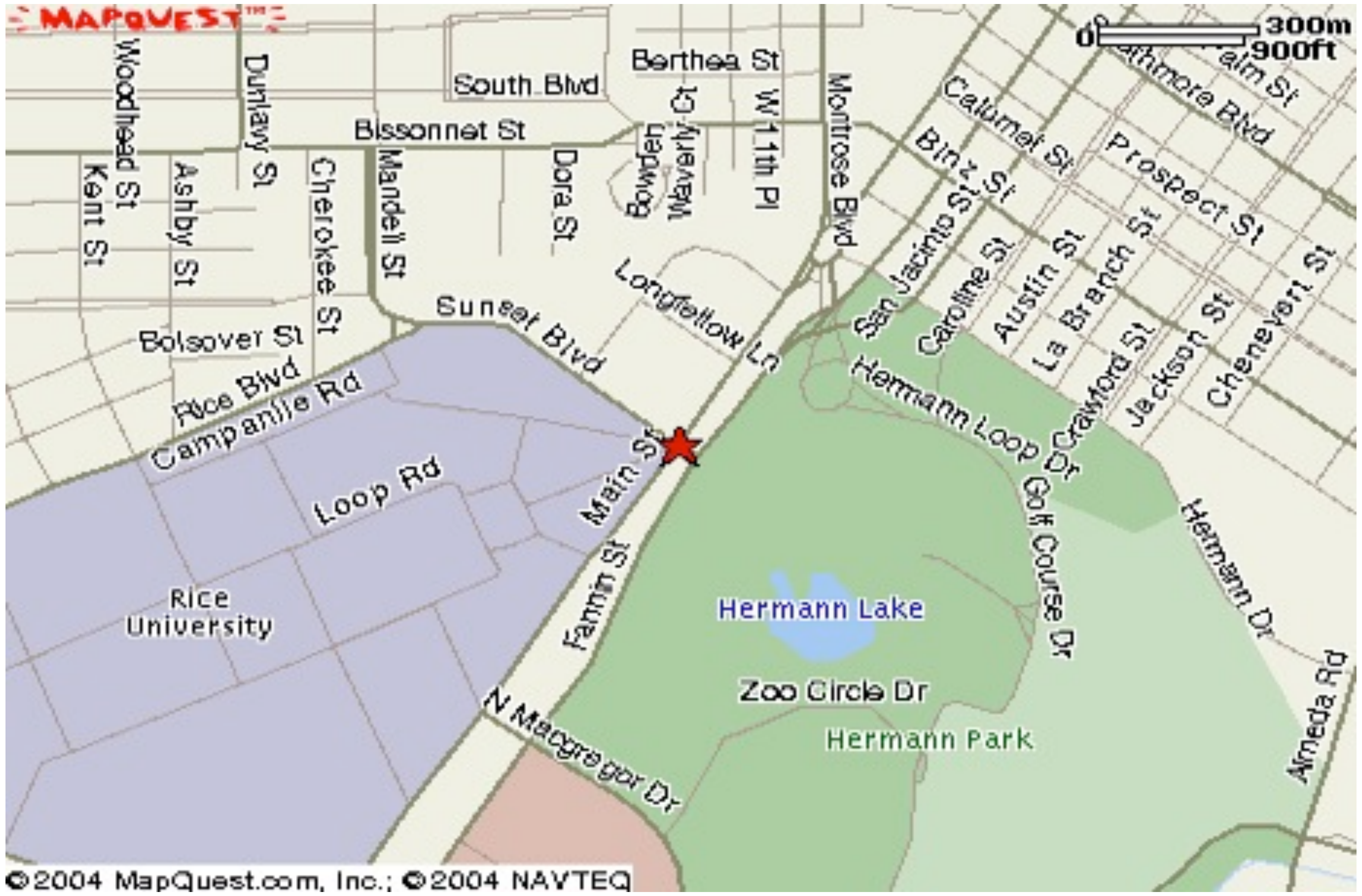
- Basic concepts and definitions.
- Examples of sampling-based planners:
  - Roadmap planner
  - Tree-based planner
- Underlying key components.
- OMPL and future challenges in motion planning.

Probabilistic Roadmap Planner (PRM)

Kavraki, Svestka, Overmars and Latombe 96

PRM

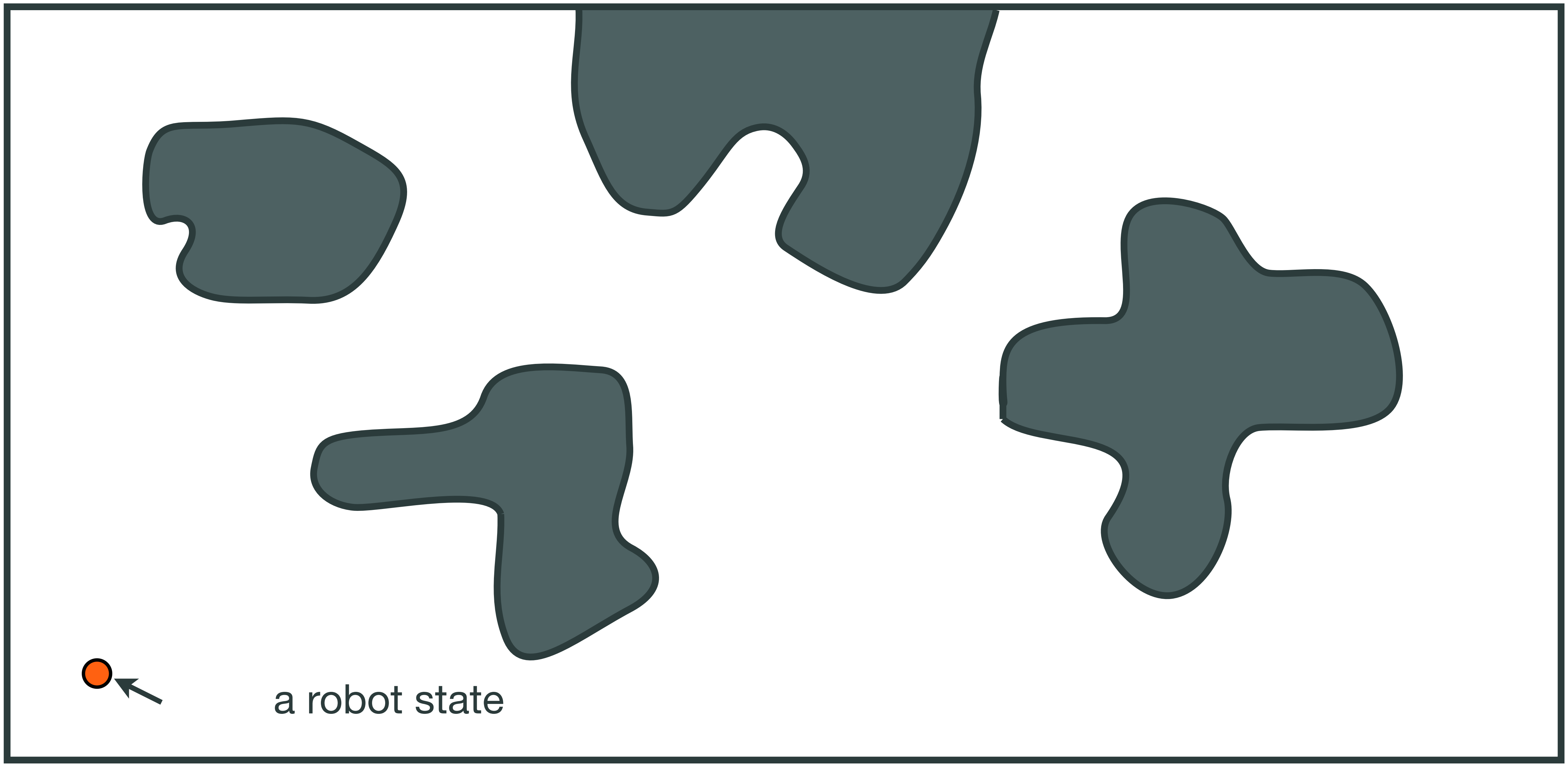
- Uses random sampling.
- Uses simple local planner.
- Builds a roadmap of the state space.



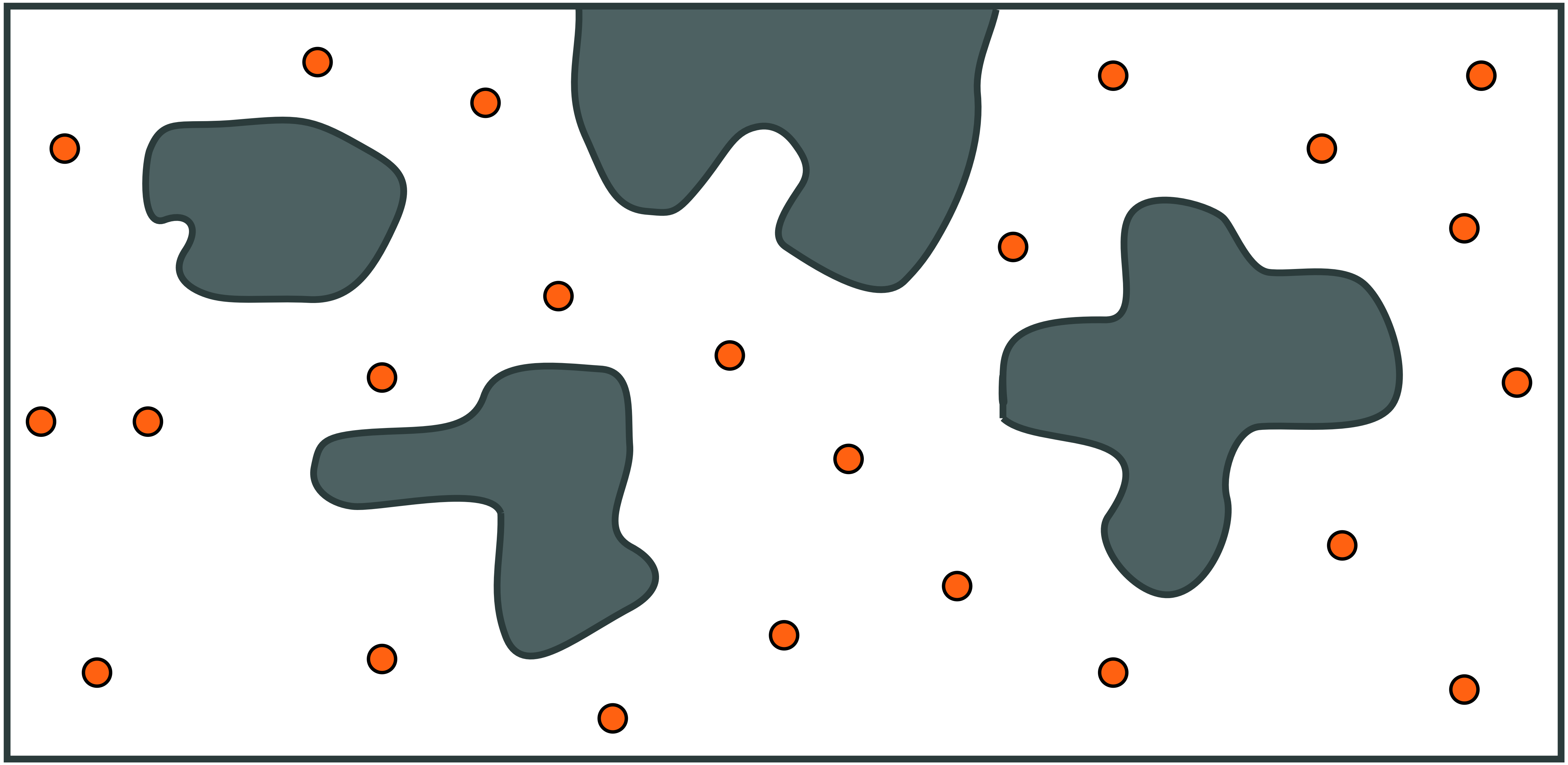
PRM

- Illustrate with an easy planning instance/problem set up.
    - Robot is a point in 2D.
    - Robot moves freely.
    - Simple example used for illustration only.
- Isolate primitive techniques.
- Generalize.

Point robot in 2-D

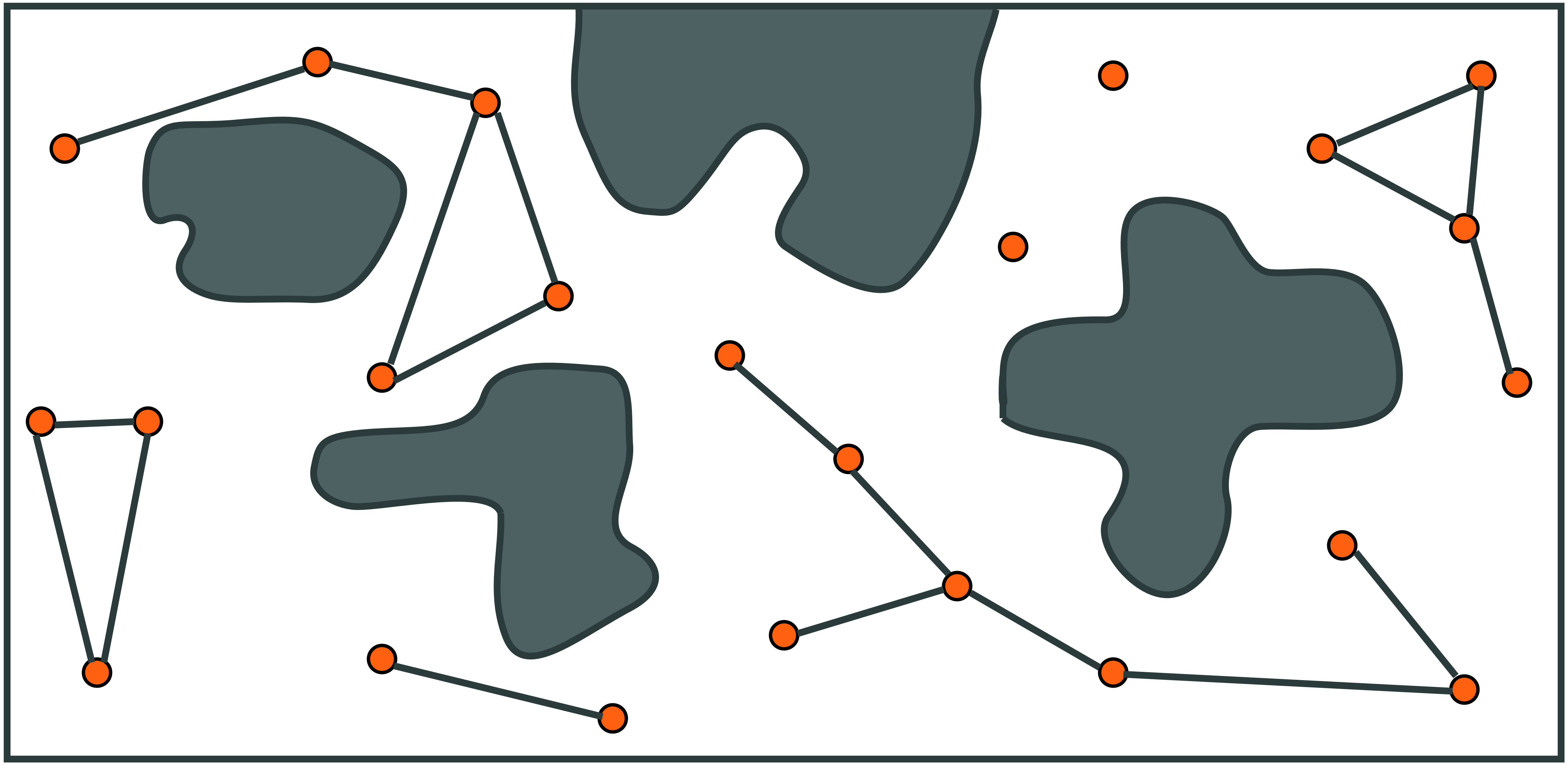


Operation of PRM



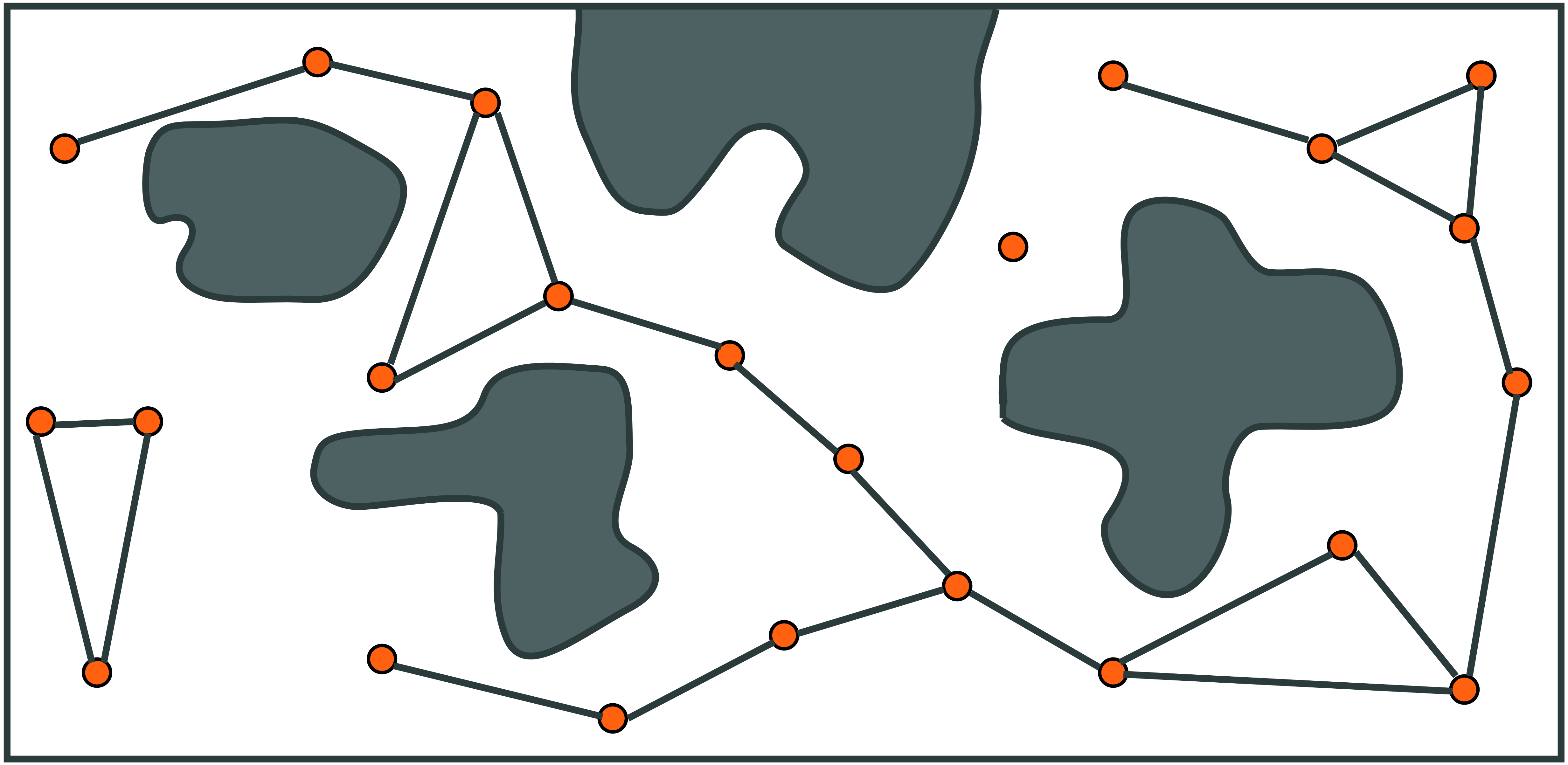
: **nodes,** random states

Operation of PRM



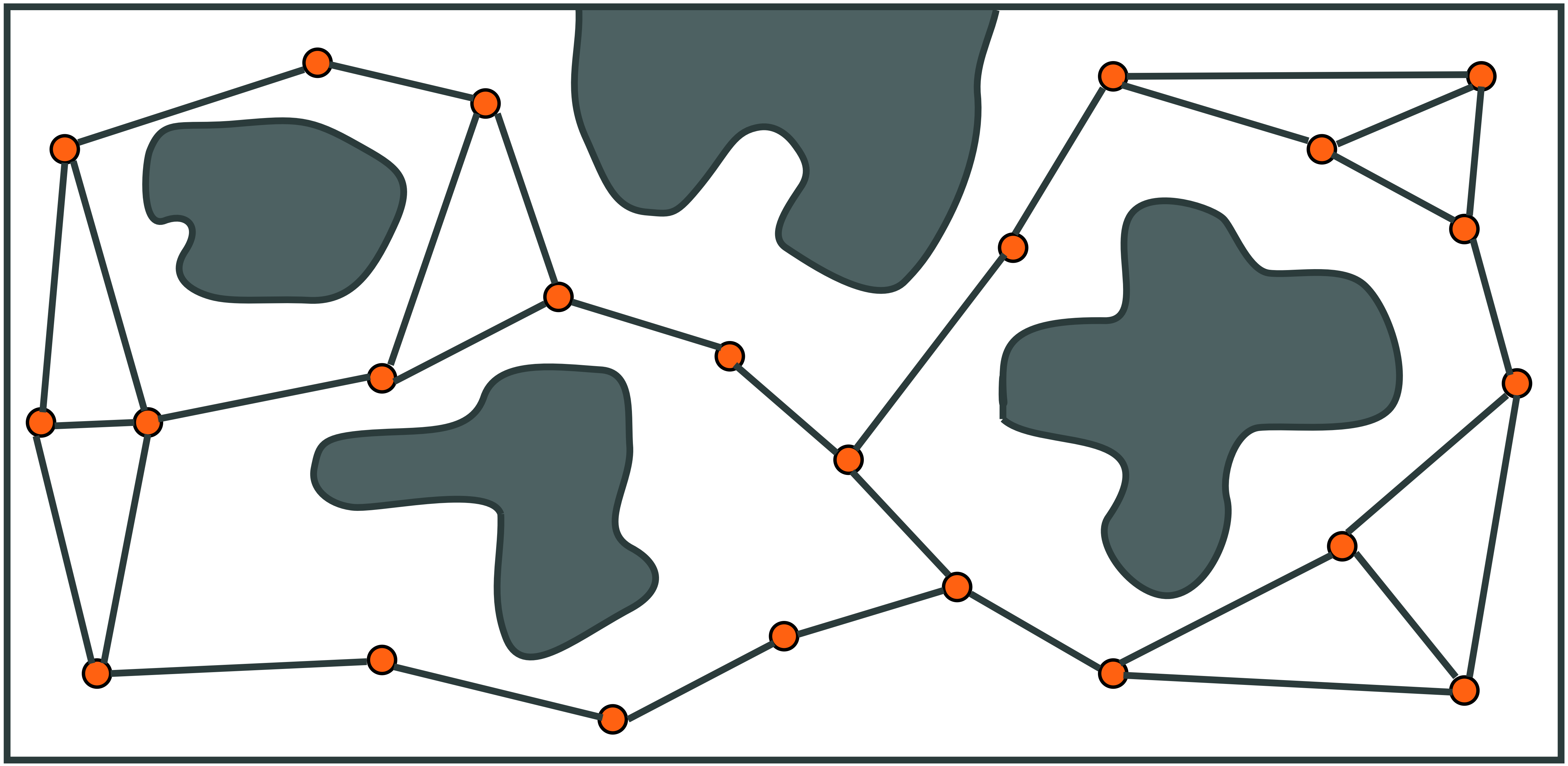
:edges, paths computed by local planner

Operation of PRM



:edges, paths computed by local planner

Operation of PRM

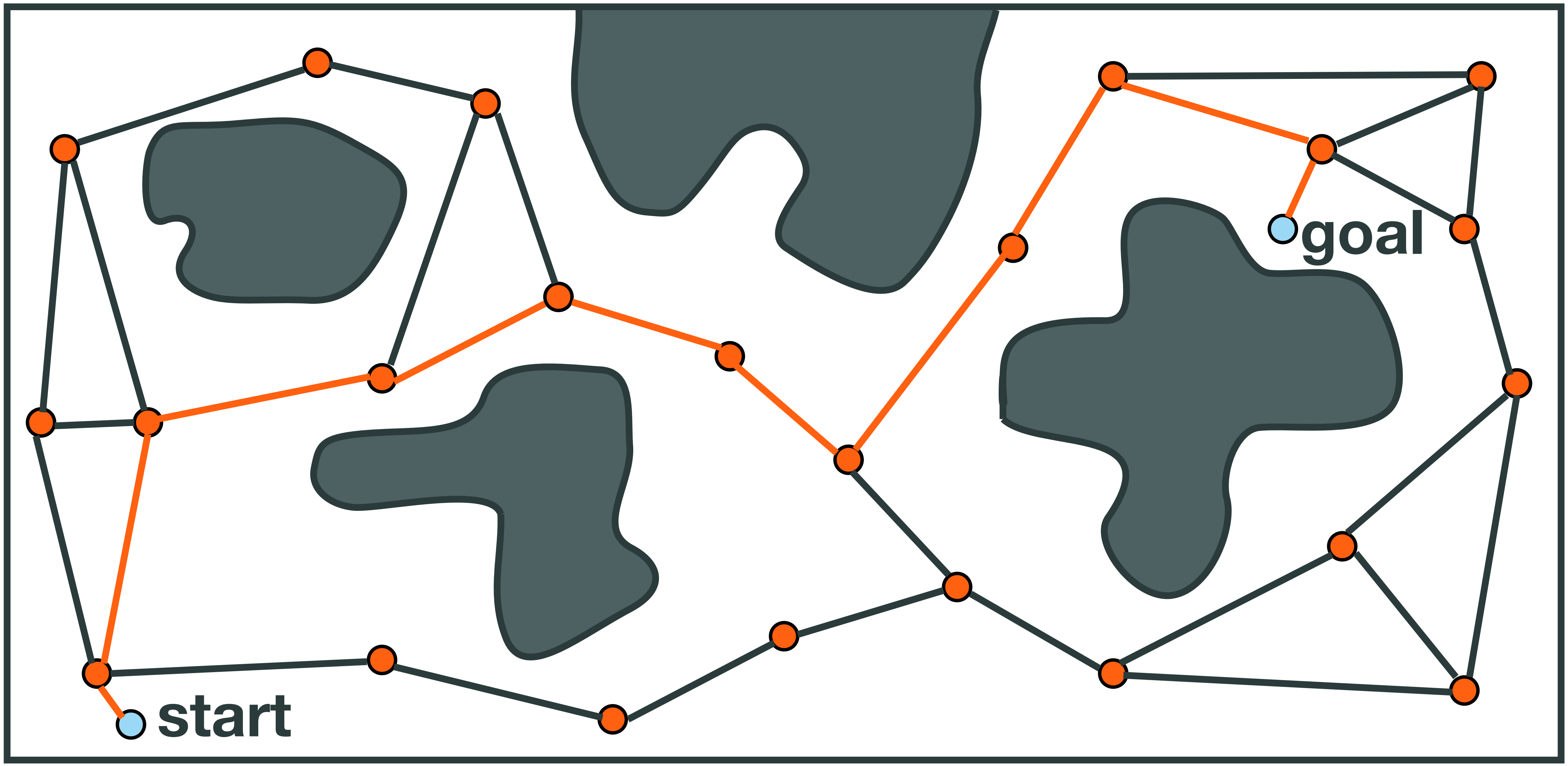


:edges, paths computed by local planner

Queries

- Given a roadmap G and query q_0, q_f
  - Connect q_0 and q_f to G.
  - Check to see if there is a path in G.

Answering Queries



plan a path:

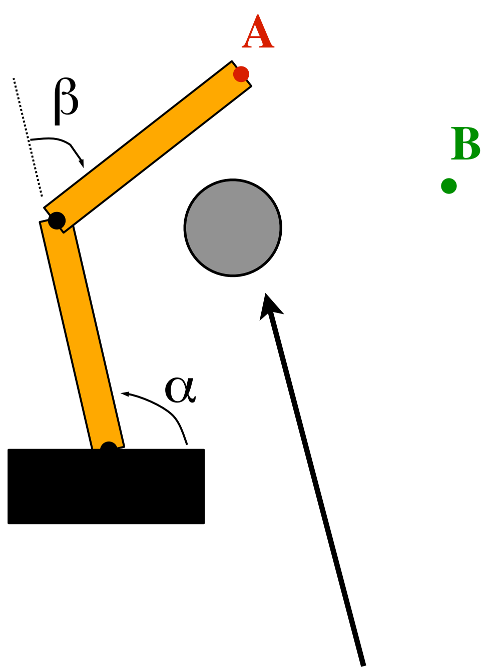
1. connect start & goal to roadmap
2. perform graph search

Primitive Techniques

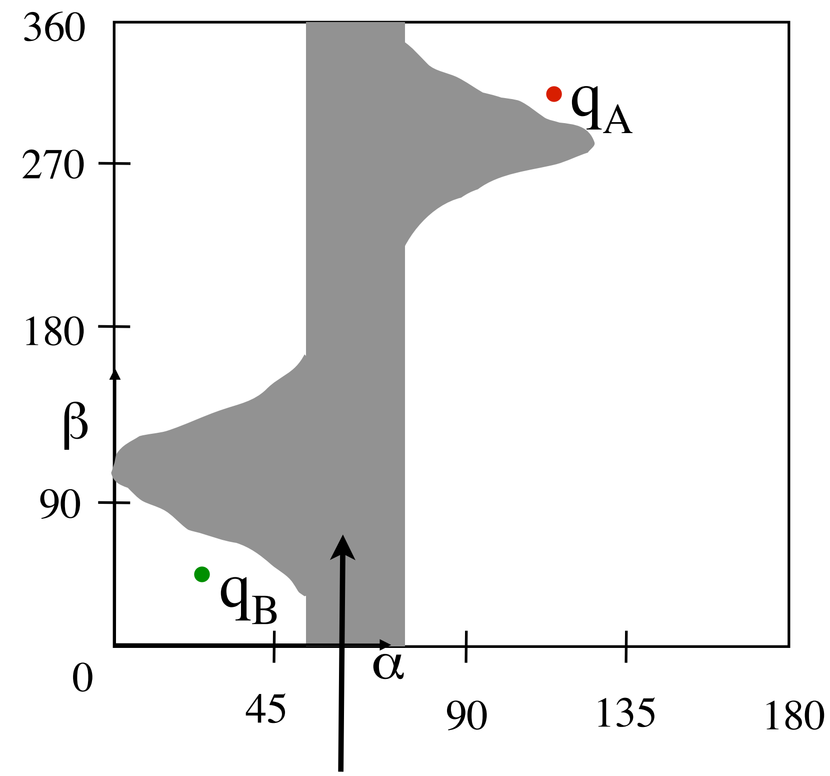
- **Select Sample:** (in the example) Uniform sampling to get milestones.
- **Connect:** (in the example) Local planner uses “straight lines.”
- **Store in some data structure:** (in the example) A graph.
    - A roadmap is finite graph G=(V,E)
    - V is a subset of S_{free}.
    - (s_1,s_2) in E implies that the local planner found a path.

Why use sampling?

- S_{free} is impractical to represent explicitly.



Obstacle in workspace

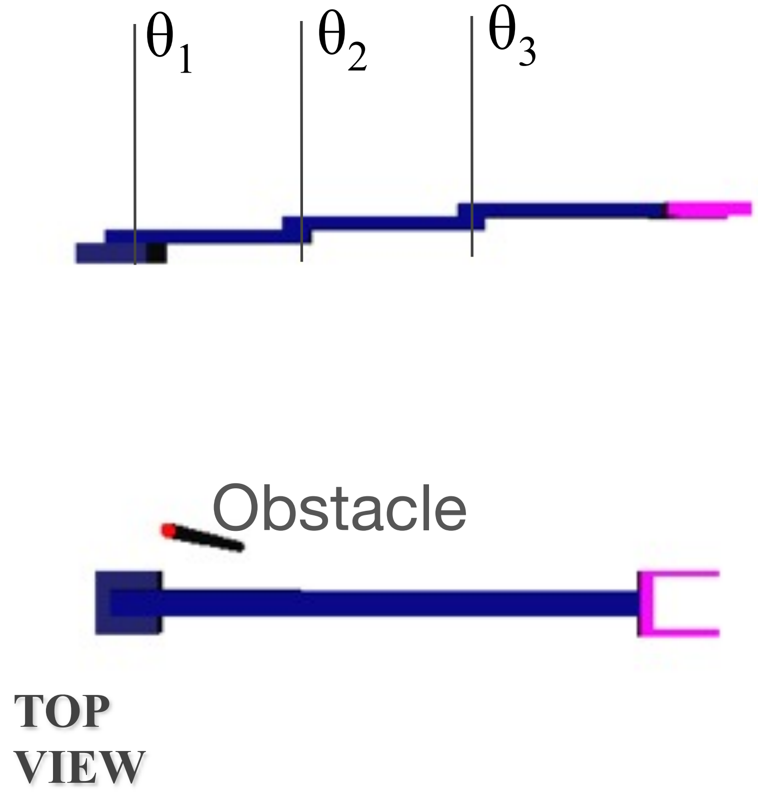


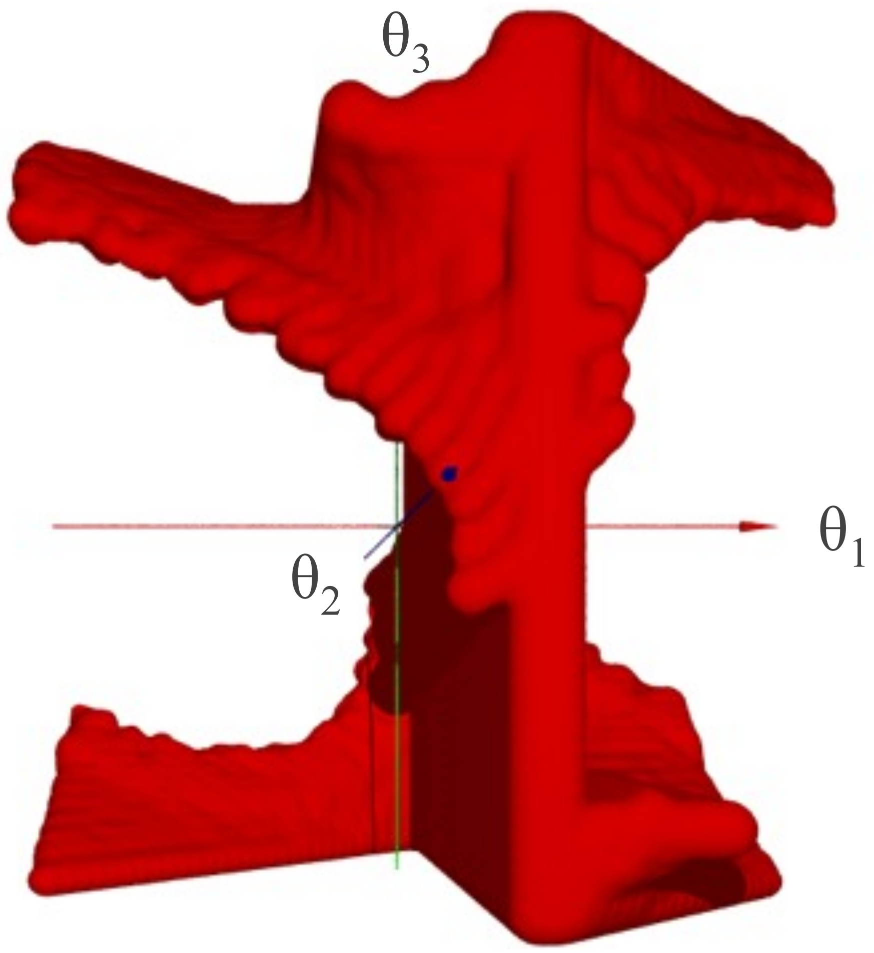
Obstacle in state space

-

Why use sampling?

- S_{free} is impractical to represent explicitly.





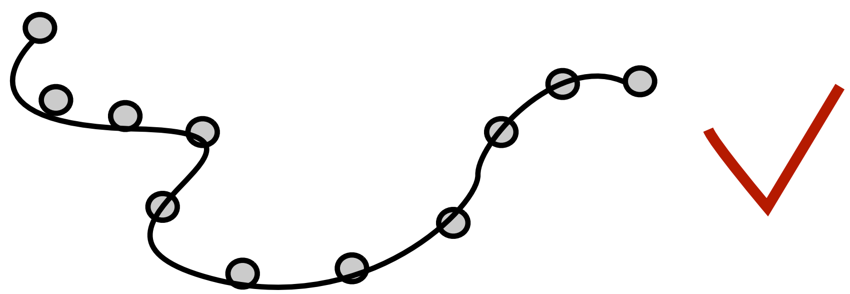
-

Why use sampling?

- S_{free} is impractical to represent explicitly.
- Sampling can be very efficient.
- Resulting data structure can be very compact.

Connecting samples

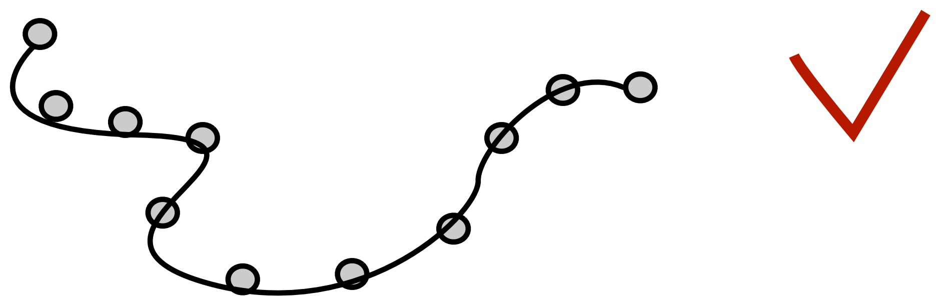
- An example of a simple planner:
  - Computes the straight line path between q1, q2.
  - Checks to see if it is valid.
  - If so, returns SUCCESS and the path.
  - Otherwise, returns FAIL.



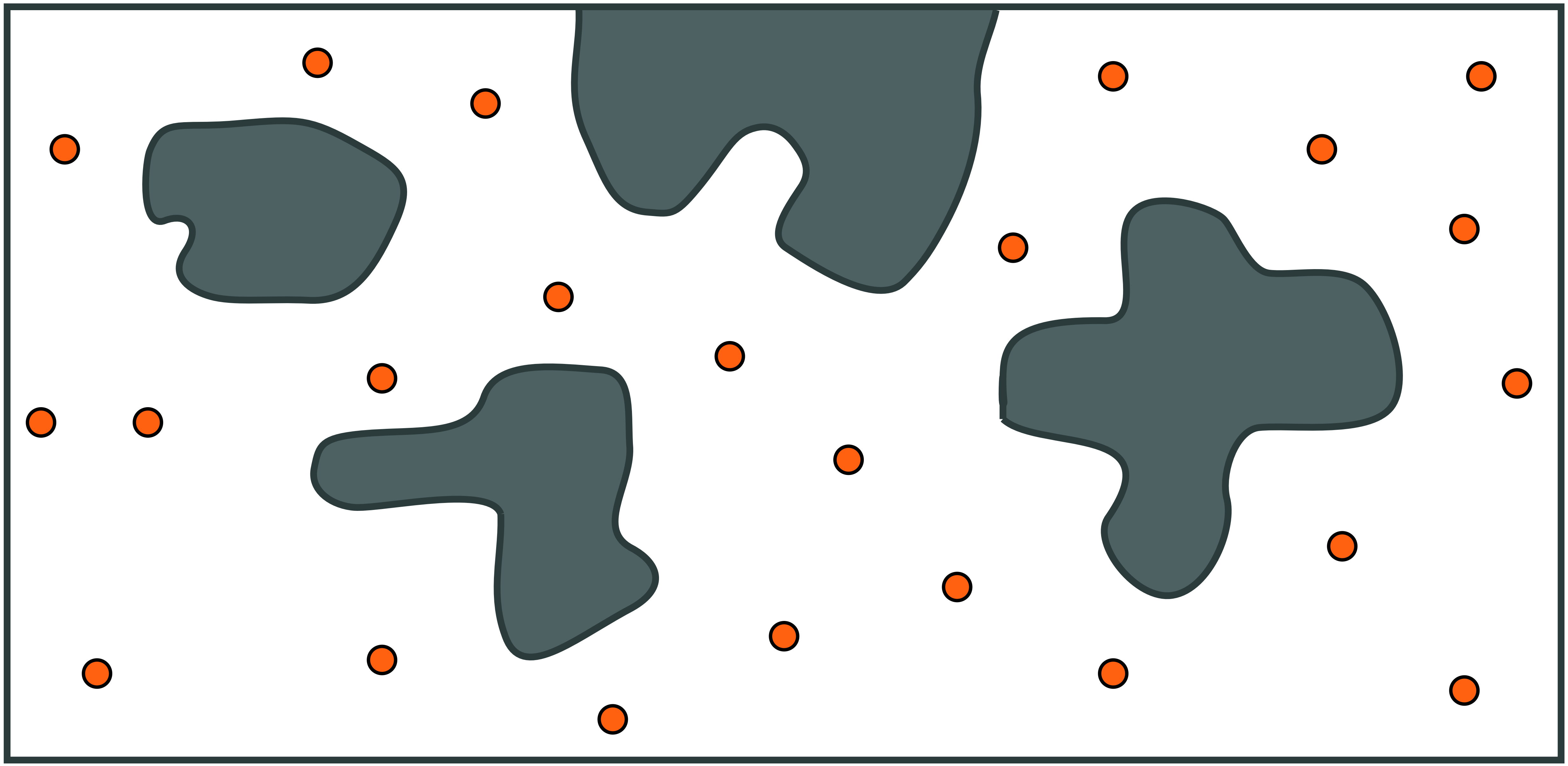
May fail often

State validity checker

- For states
    - Use e.g., collision checking, check any bounds
- For paths
    - State validation along a path is done by recursive refinement.
    - Bounds on clearance are combined with bounds on motion to cover the path with open balls or find a collision.

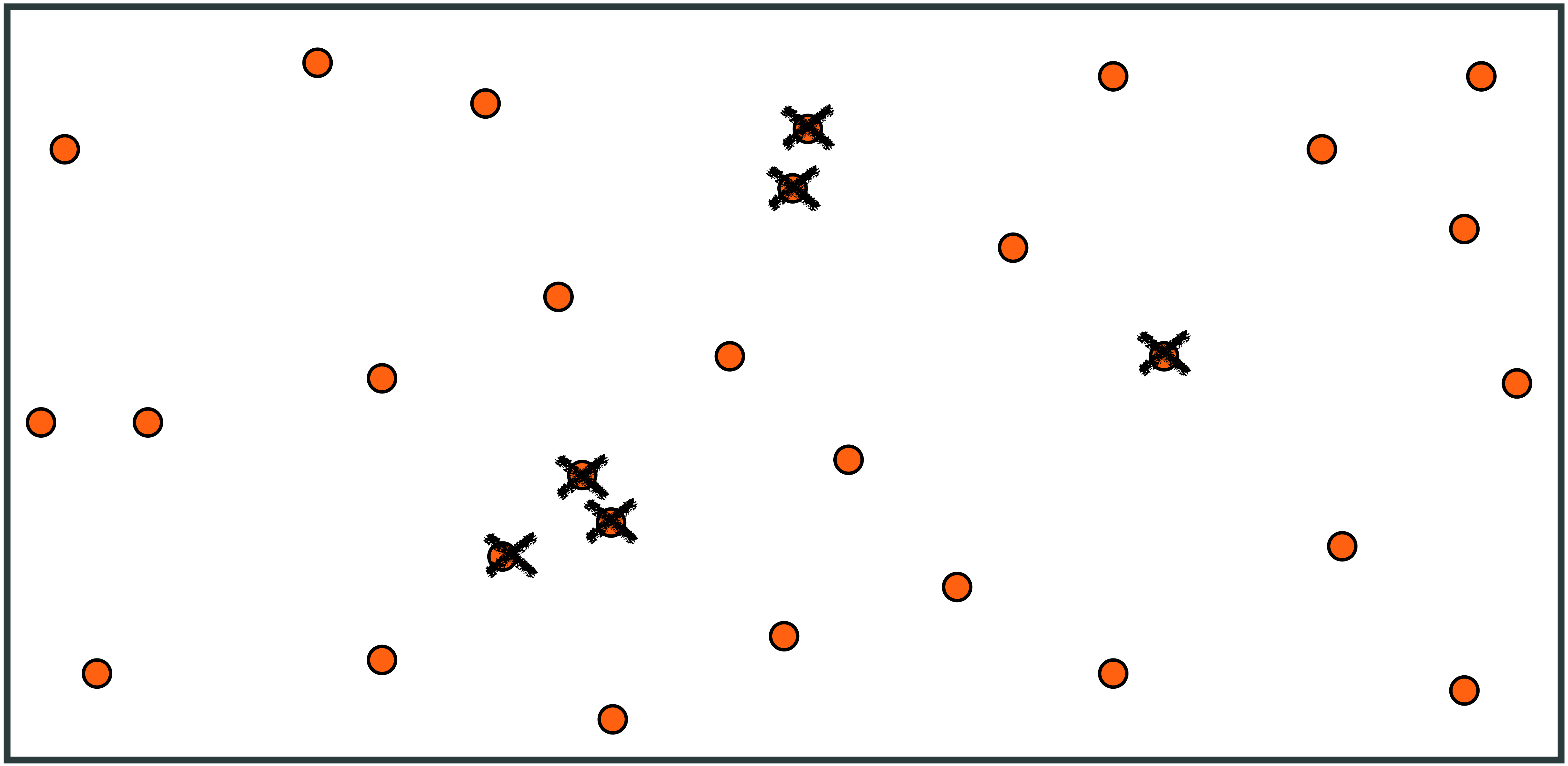


Operation of PRM



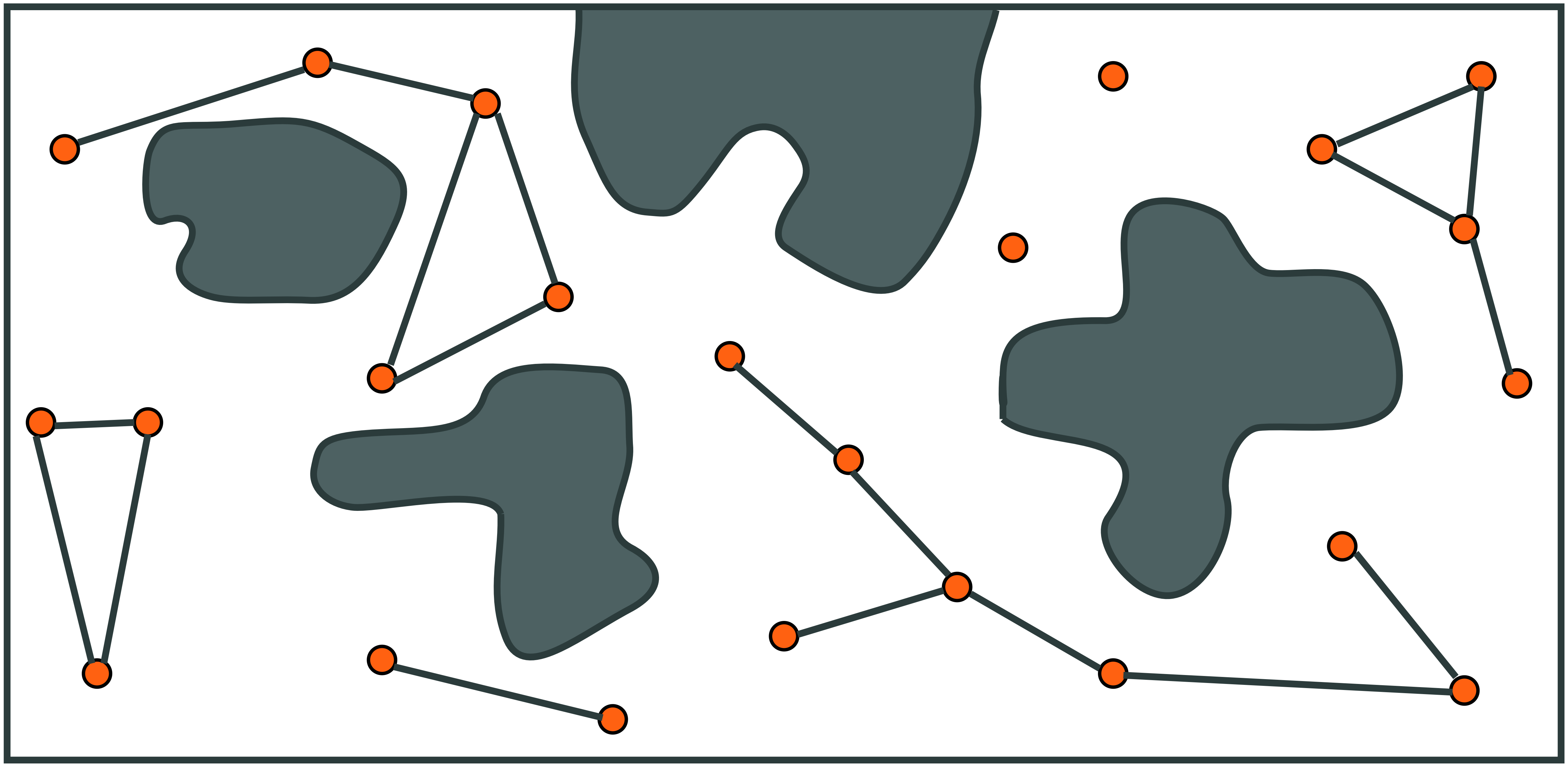
: **nodes,** random states

Operation of PRM



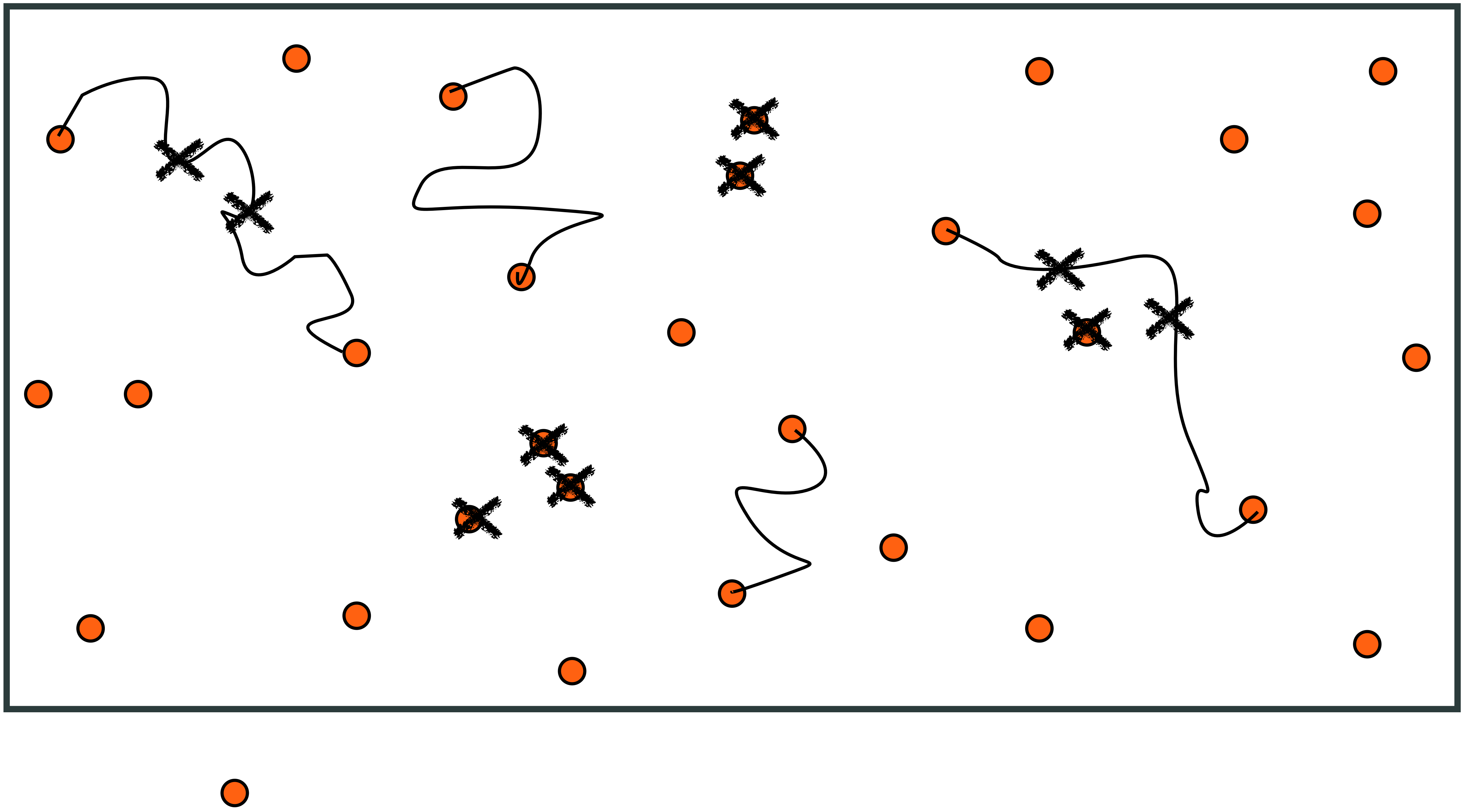
: states

Operation of PRM

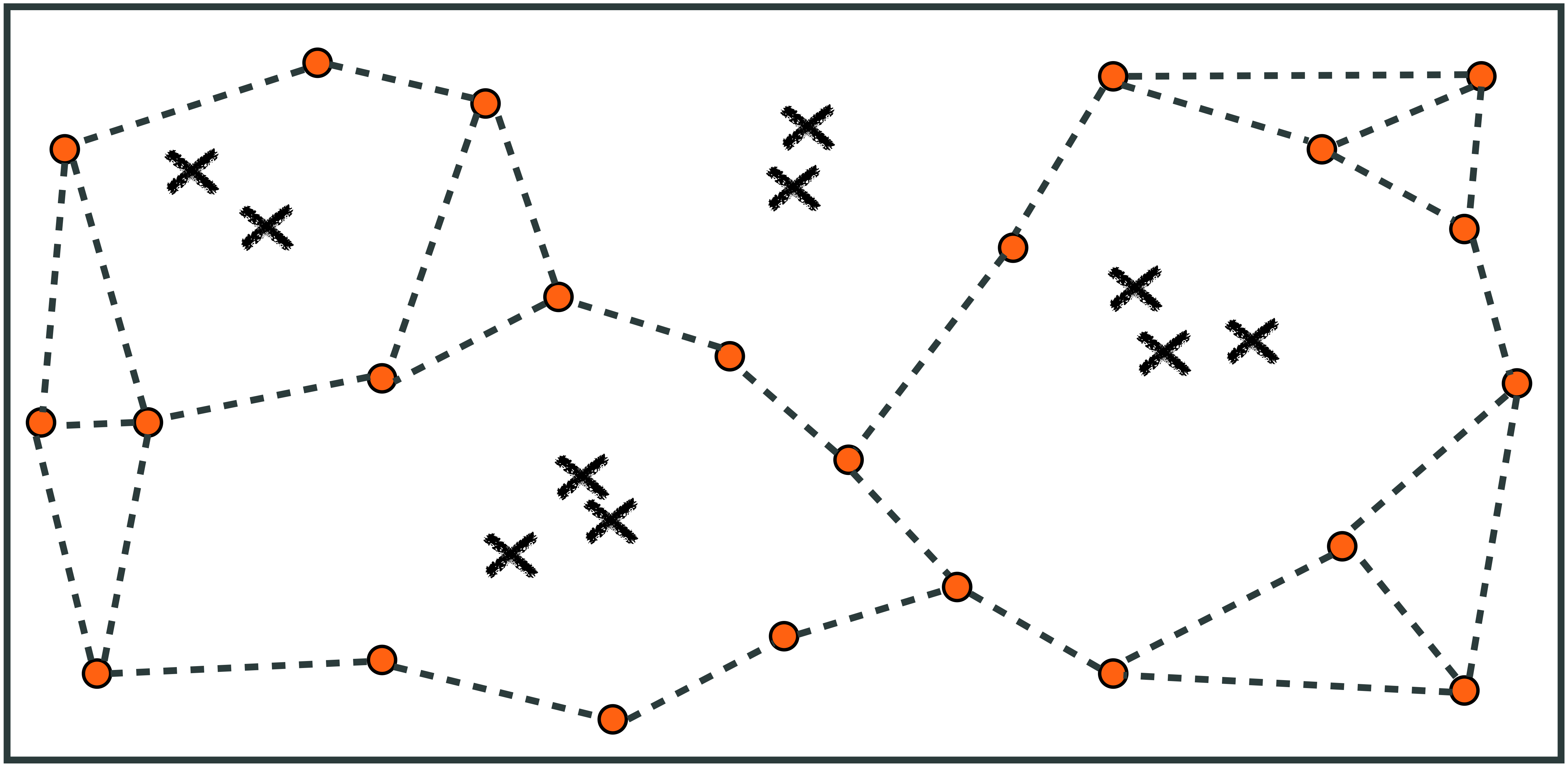


:edges, paths computed by local planner

Operation of PRM

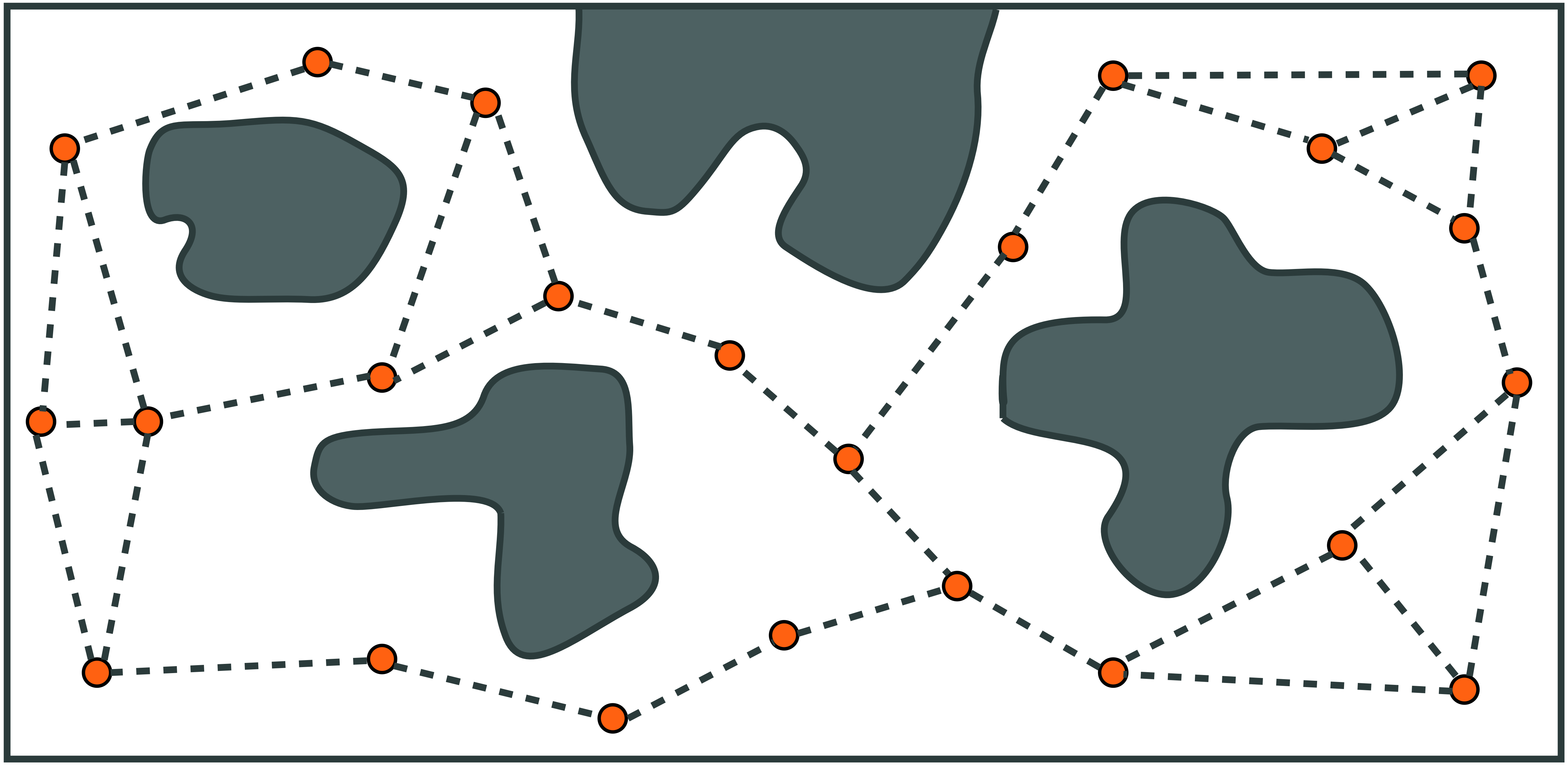


Operation of PRM



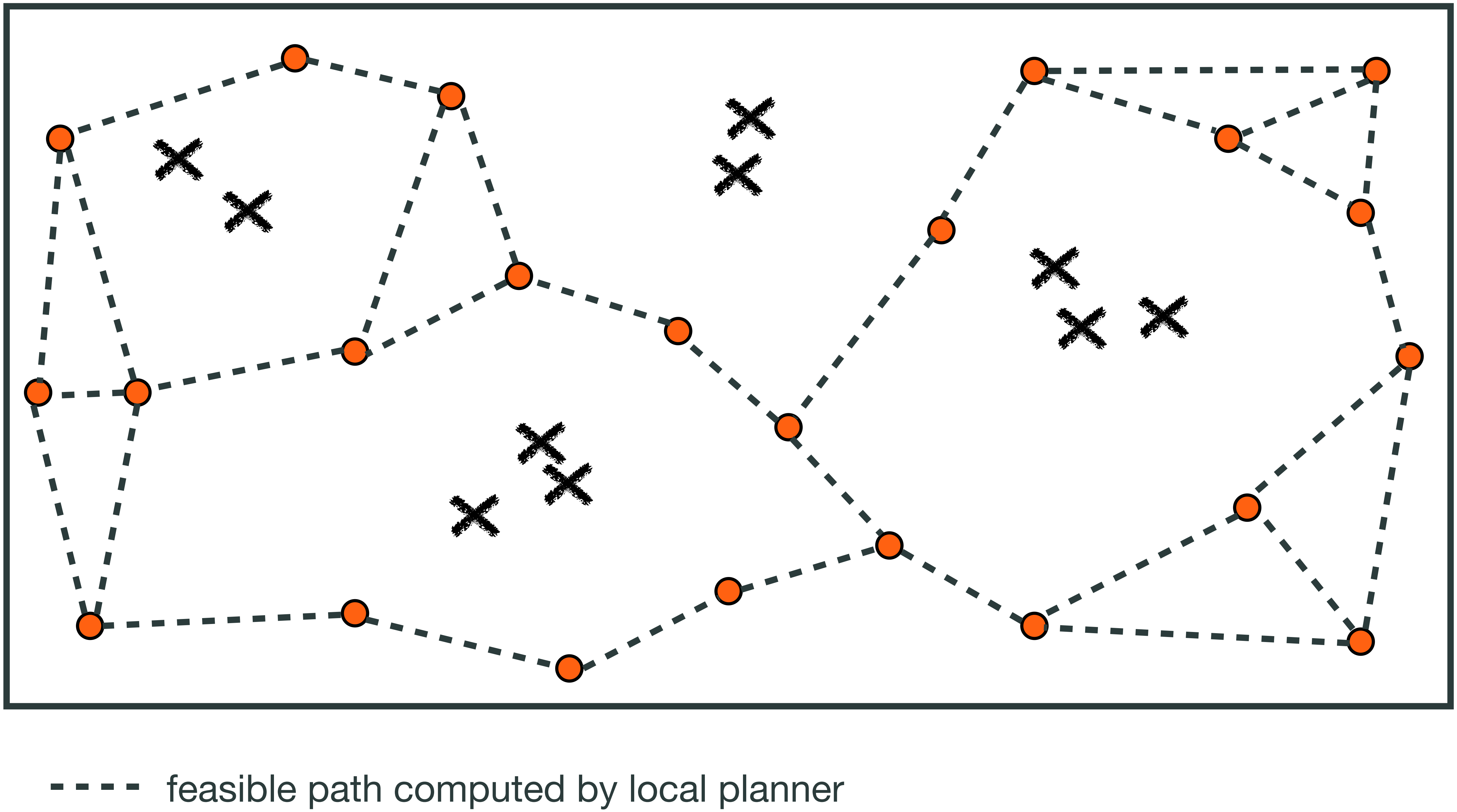
- - - - feasible path computed by local planner

Operation of PRM



feasible path computed by local planner

Operation of PRM

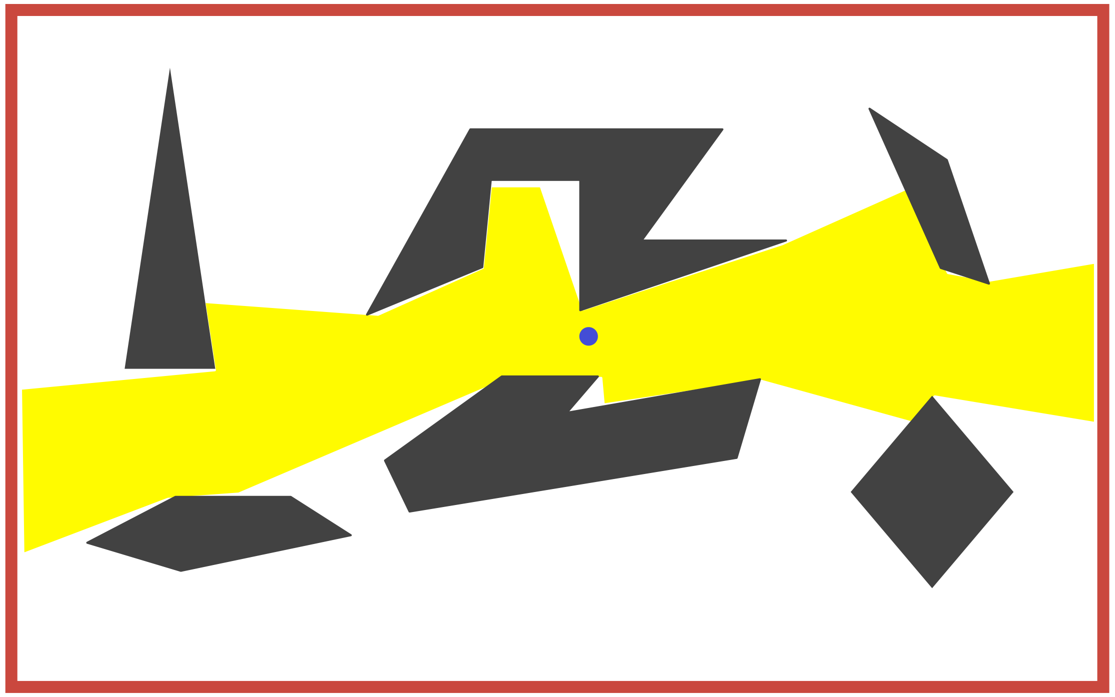


Completeness of PRM

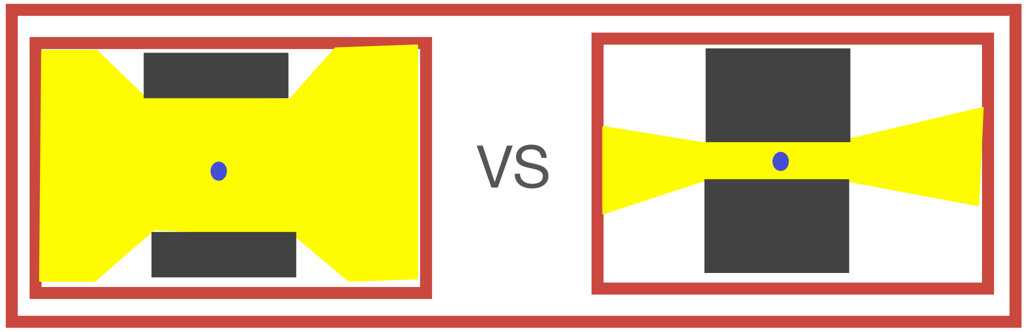
- If no path exists, then PRM cannot find the path.
- But… if a path exists, it is possible PRM fails to find it.
- PRM is not complete but instead is probabilistically complete.

Theoretical Analysis of PRM (1/2)

[Kavraki et al 96, 98, 00, 03, 07]



ε-goodness property



- Tradeoff: planner may fail with probability α
- Number of nodes/states:

$$N \approx \frac{1}{\epsilon}[log(\frac{1}{\epsilon}) + log(\frac{4}{\alpha})]$$

- Important: Performance related to properties of the space

Theoretical Analysis of PRM (2/2)

- We sacrifice completeness for speed
- Probabilistic completeness
- Novel analysis and performance guarantees

$$Pr(failure) = f(e^{-cN})$$

- How much can the assumptions be relaxed?

Primitive techniques

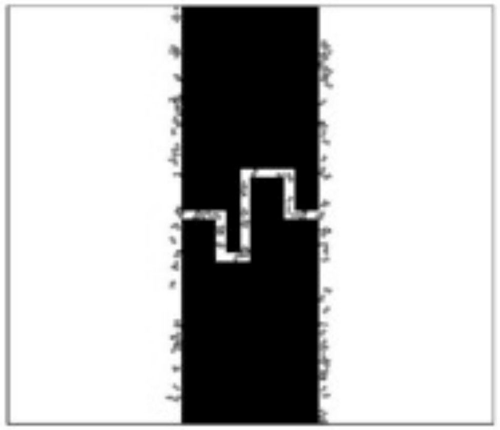
Primitives

- **Select Sample:** Uniform sampling is general but not the most efficient.
  - Optimal selection remains elusive.
- **Connect:** Connect all to all is general but not efficient.
  - Neighbors
  - Notion of “straight line” or other local plan needs to be adapted.
- **Store efficiently**

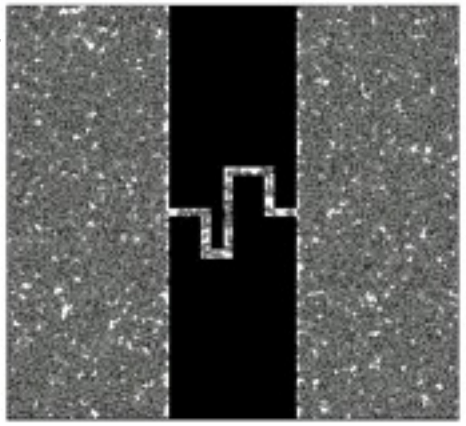
Primitives

- **Select Sample:** Uniform sampling is general but not the most efficient.
    - Optimal selection remains elusive.
- **Connect:** Connect all to all is general but not efficient.
    - Neighbors
    - Notion of “straight line” or other local plan needs to be adapted.
- **Store efficiently**

Several sampling strategies



VS



- Gaussian sampling [Overmars et al]:
    - Places samples close to objects.
    - Distribution is Gaussian around the obstacle boundary.
- Medical-axis sampling [Amato et al].
- Bridge Test sampling for narrow corridors [Hsu et al].
- Quasi-Random sampling [LaValle et al].
- Selective sampling [Kavraki el al].

Recent study confirmed it is one of the most critical parts of the planner [Hsu, Latombe 1998].

Primitives

- **Select Sample:** Uniform sampling is general but not the most efficient.
  - Optimal selection remains elusive.
- **Connect:** Connect all to all is general but not efficient.
  - Neighbors
  - Notion of “straight line” or other local plan needs to be adapted.
- **Store efficiently**

Several connection strategies

- Nearness: Try to connect each configuration to a constant number of "nearby" configurations.
  - nearest neighbors by kd-trees, k-NN, k-ANN
  - random neighbors may be helpful
- Component technique: Only test edges which reduce the number of connected components in the roadmap.

Svestka, Overmars 96

Primitives

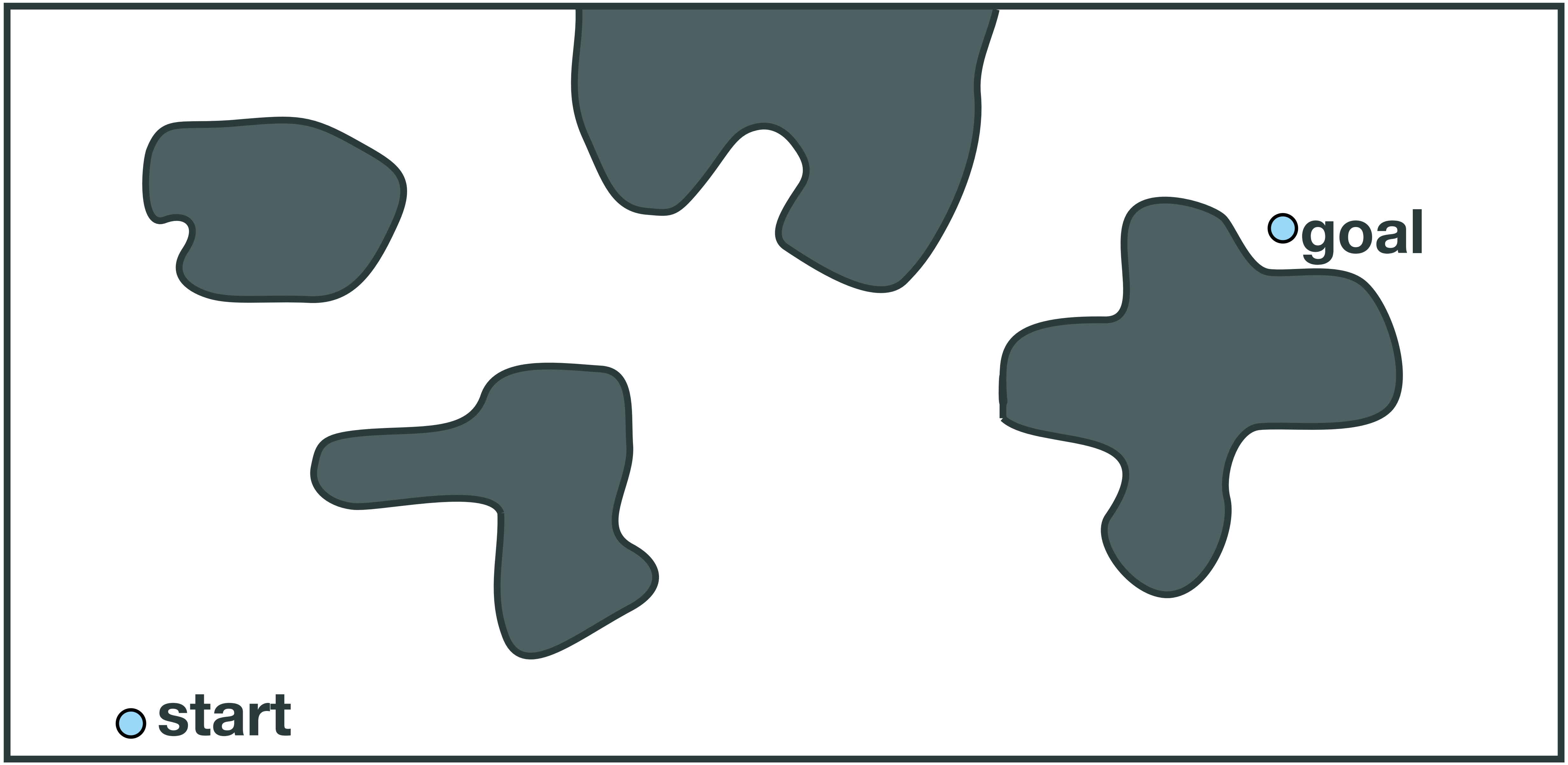
- **Select Sample:** Uniform sampling is general but not the most efficient.
  - Optimal selection remains elusive.
- **Connect:** Connect all to all is general but not efficient.
  - Neighbors
  - Notion of “straight line” or other local plan needs to be adapted.
- **Store efficiently**

Outline of this talk

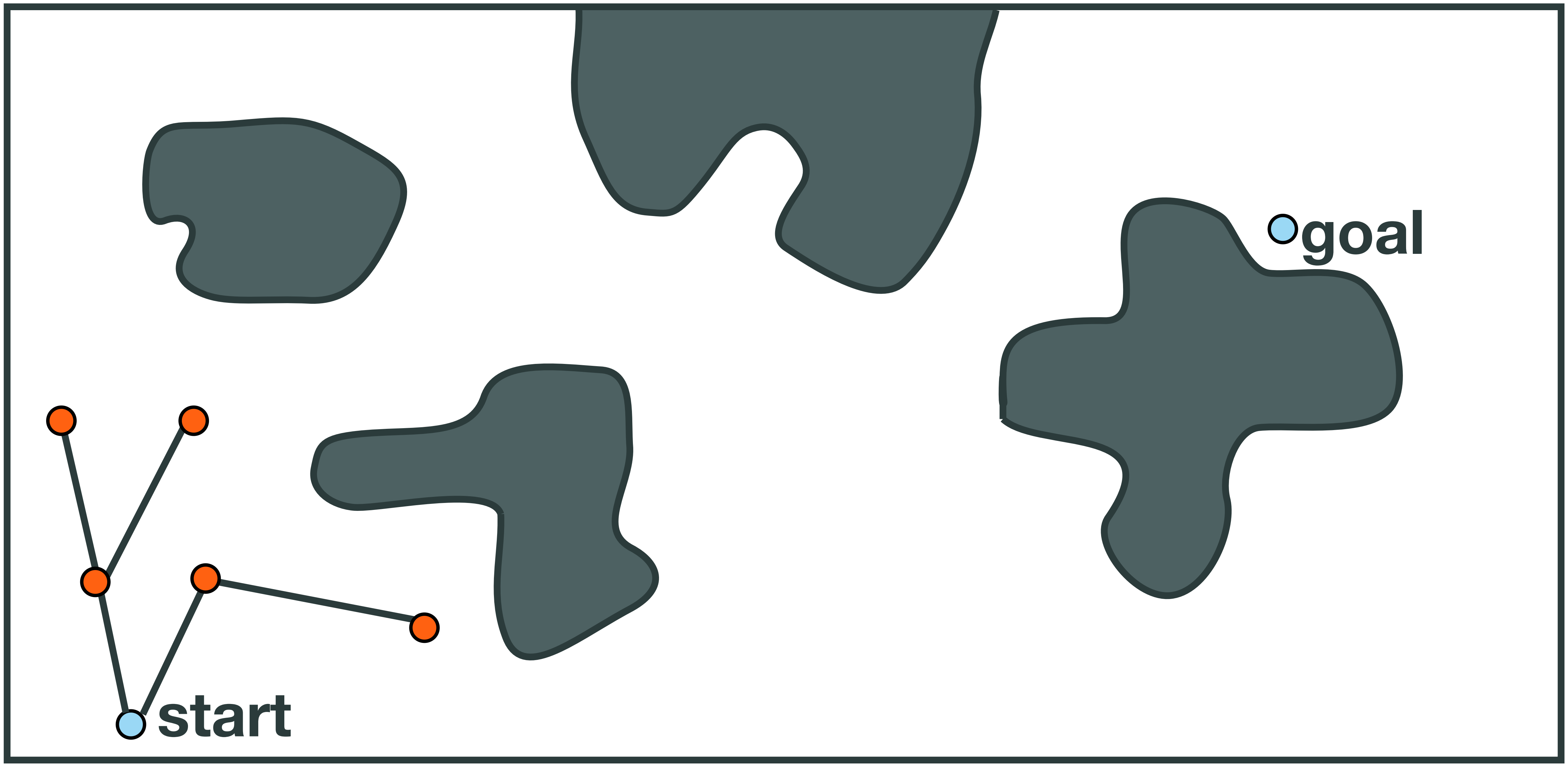
- Basic concepts and definitions.
- Examples of sampling-based planners:
    - Roadmap planner
    - Tree-based planner
- Underlying key components.
- OMPL and future challenges in motion planning.

A generic sampling-based tree planner

Sampling-based tree planner operation

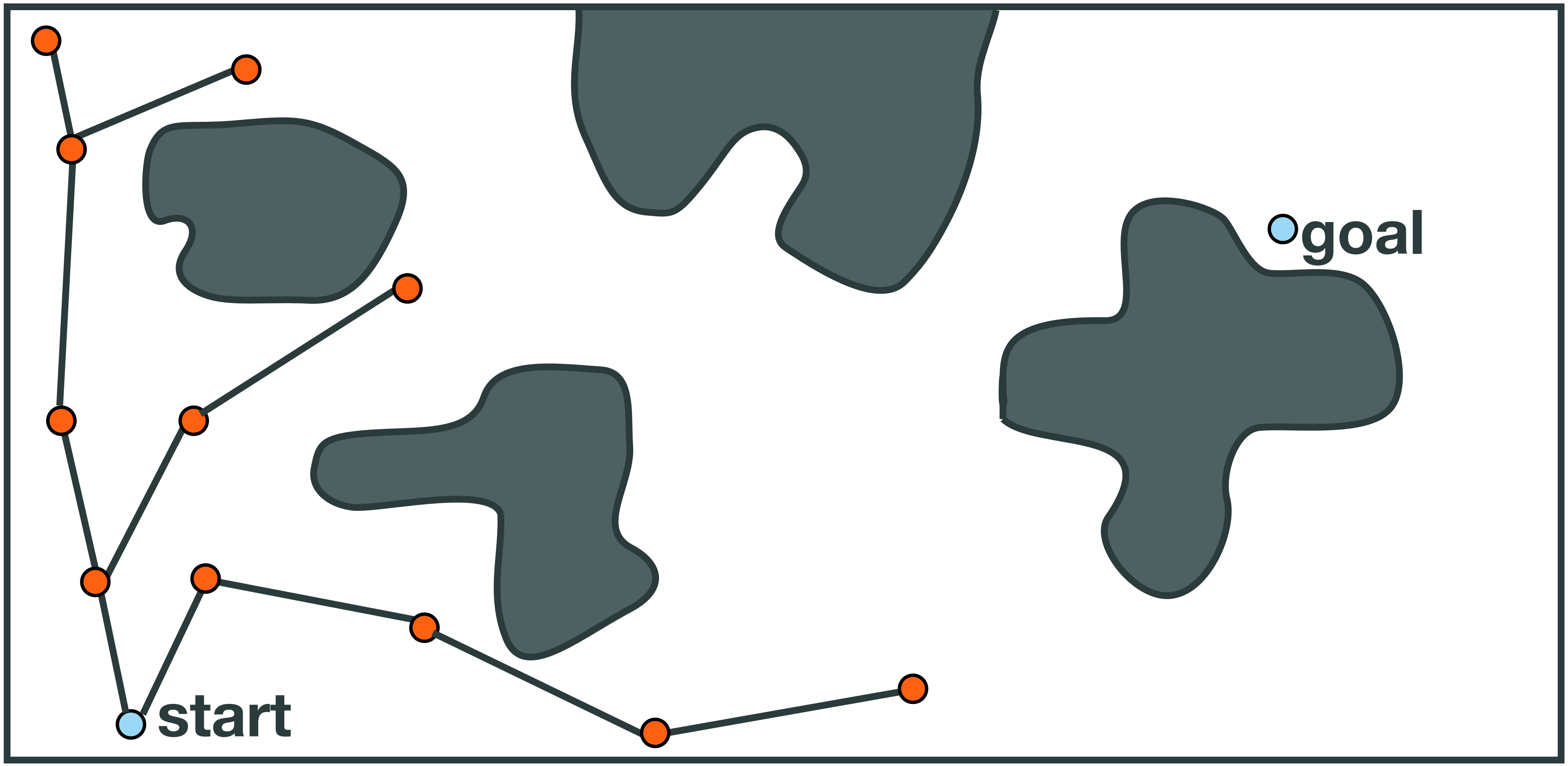


Sampling-based tree planner operation

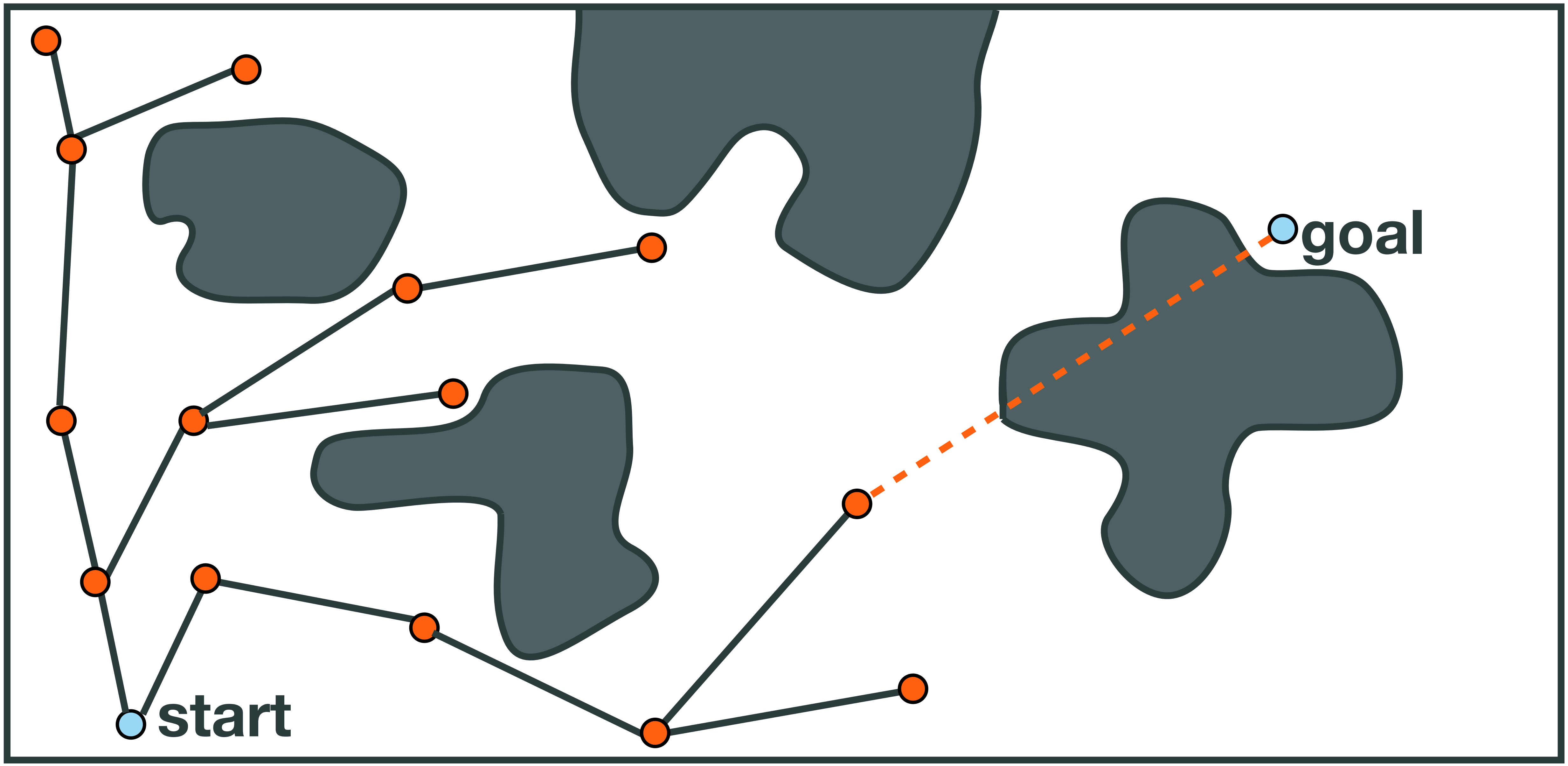


grow random tree from **start**

Sampling-based tree planner operation

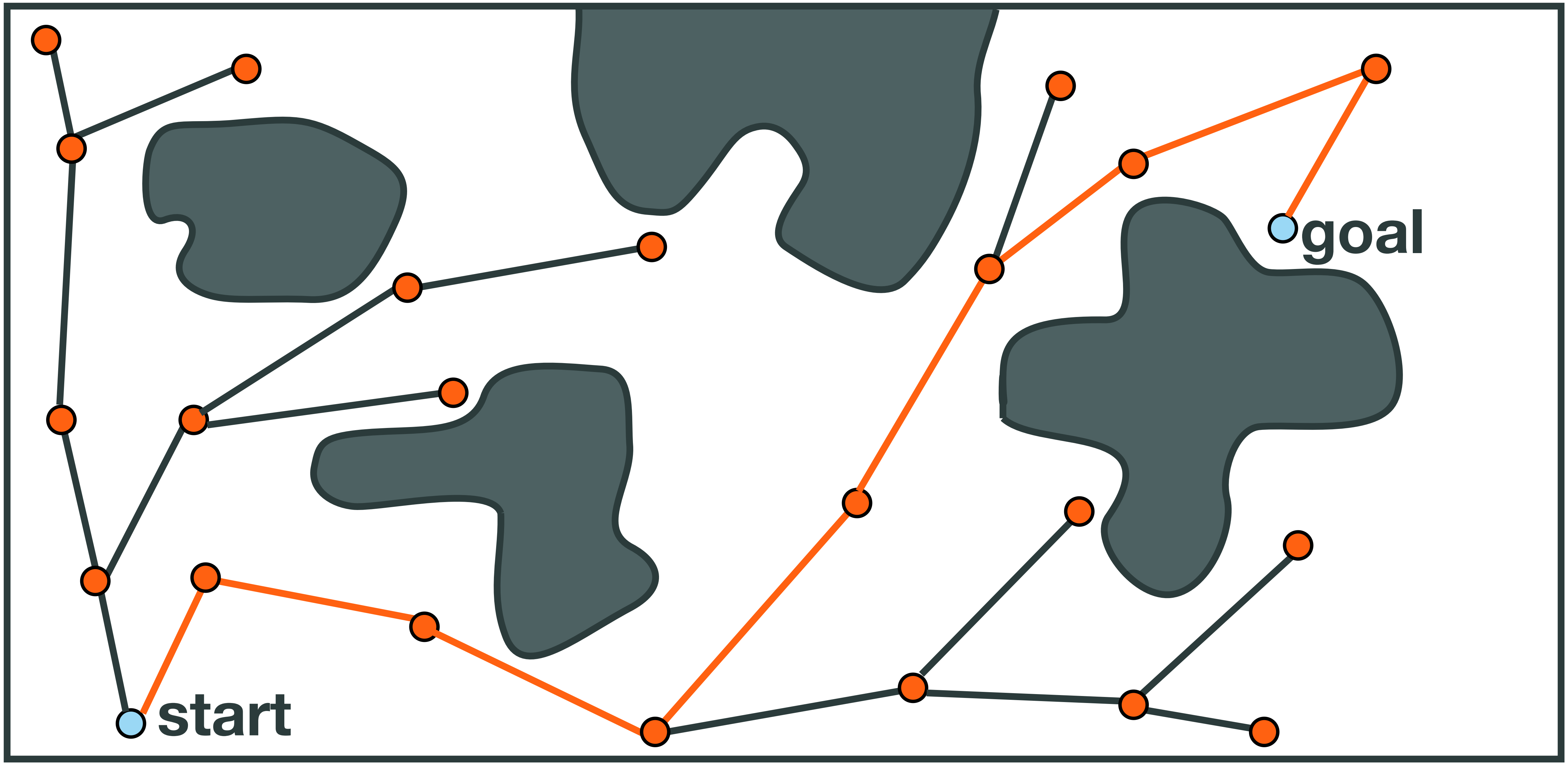


Sampling-based tree planner operation



: occasionally attempt to connect tree to **goal**

Sampling-based tree planner operation



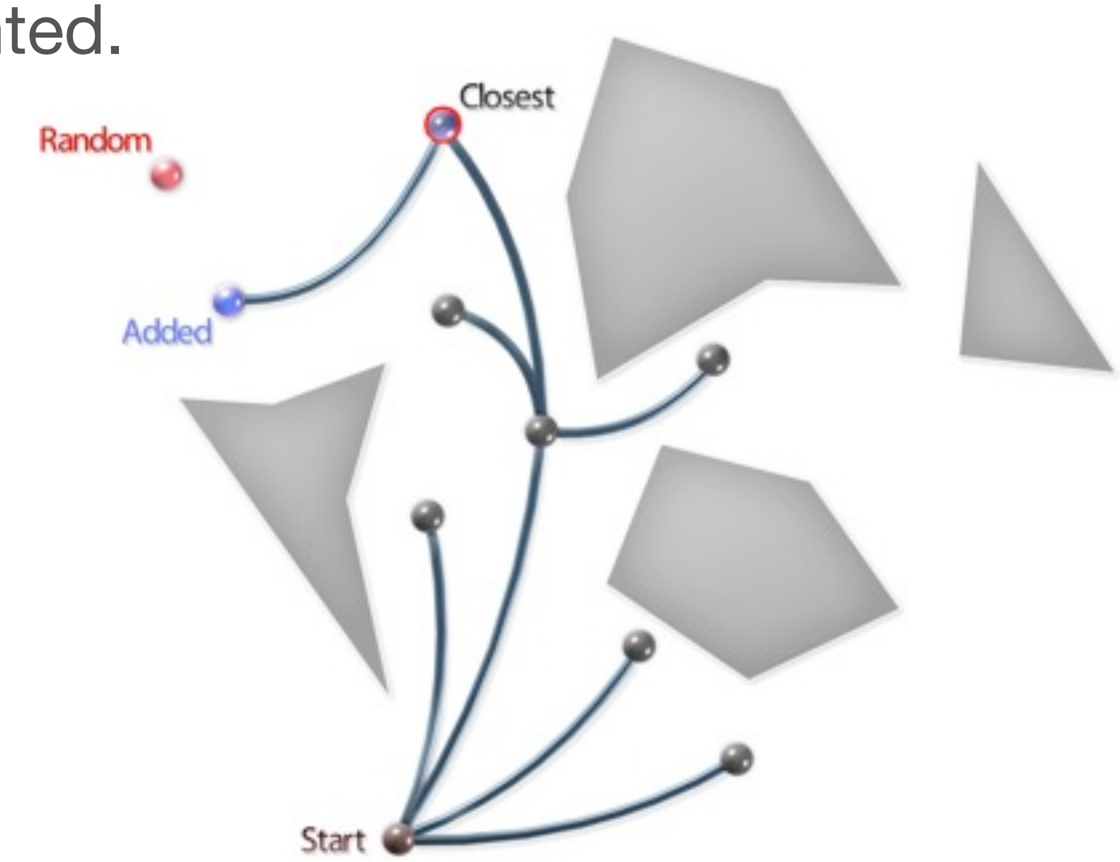
- Repeat until **goal** is connected to tree.
- Bi-directional trees are possible when considering only geometric constraints.

Primitives

- Select Sample
- Expand from the sample
- Store efficiently

Rapidly Exploring Random Trees (RRT)

- Uses proximity query to guide construction (Voronoi Bias).
- Uses propagation instead of connection.
- Powerful heuristic for single-query planning.
- Bi-directional search can be implemented.

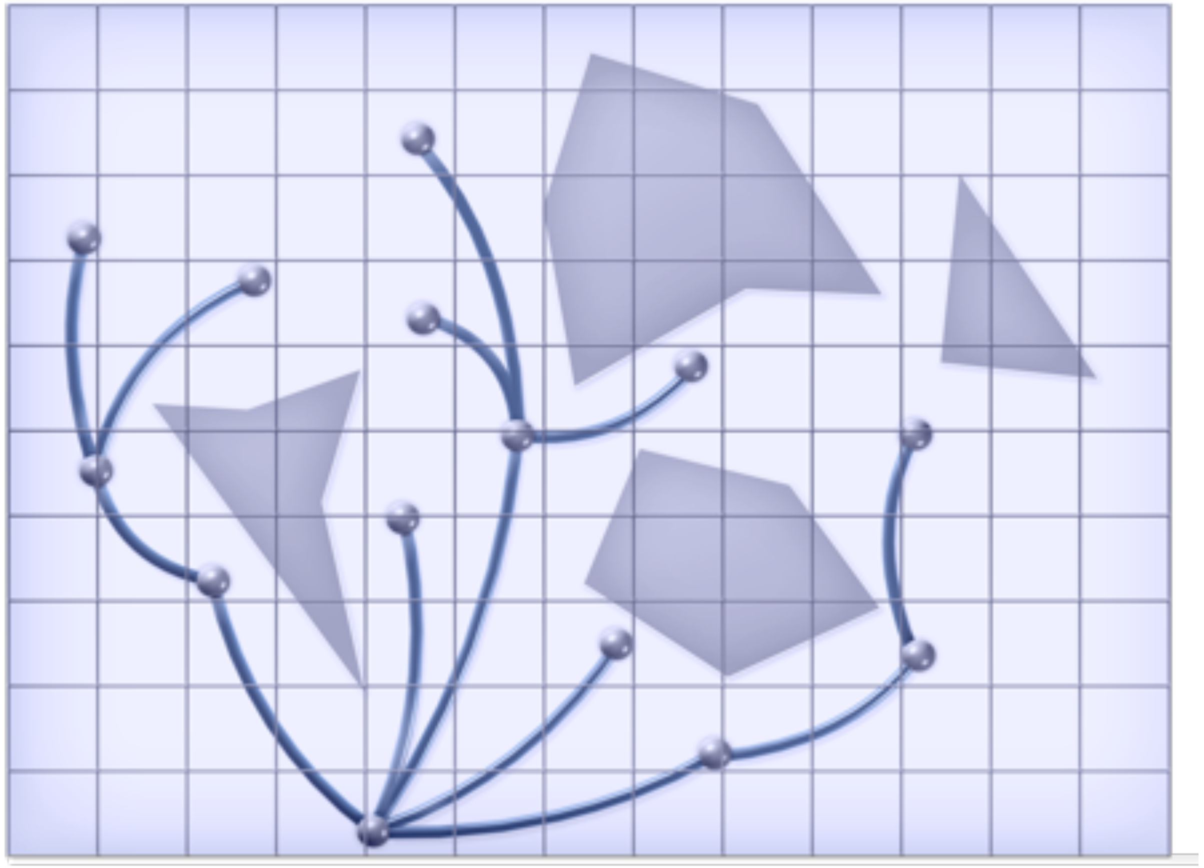


[Lavalle, Kuffer 1999, 2000]

Expansive Trees (EST and SBL)

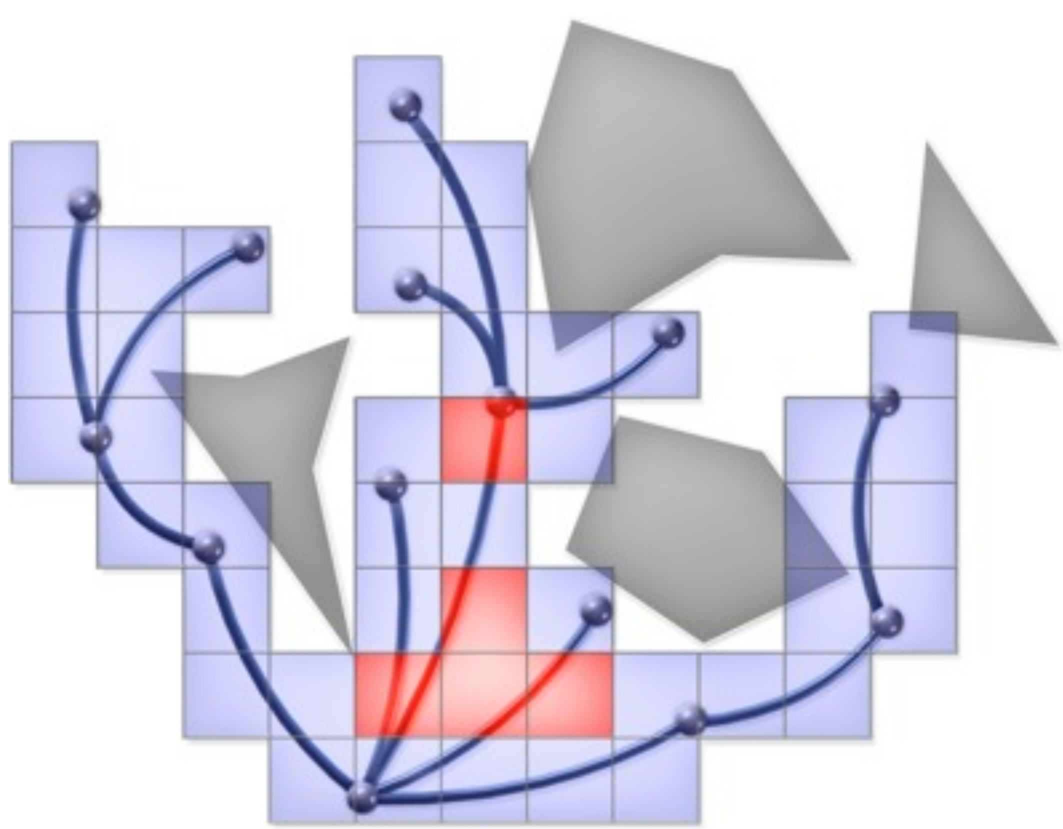
- EST: Uses density of nodes to guide expansion (density bias). [Hsu and Latombe, 1997, 1999]

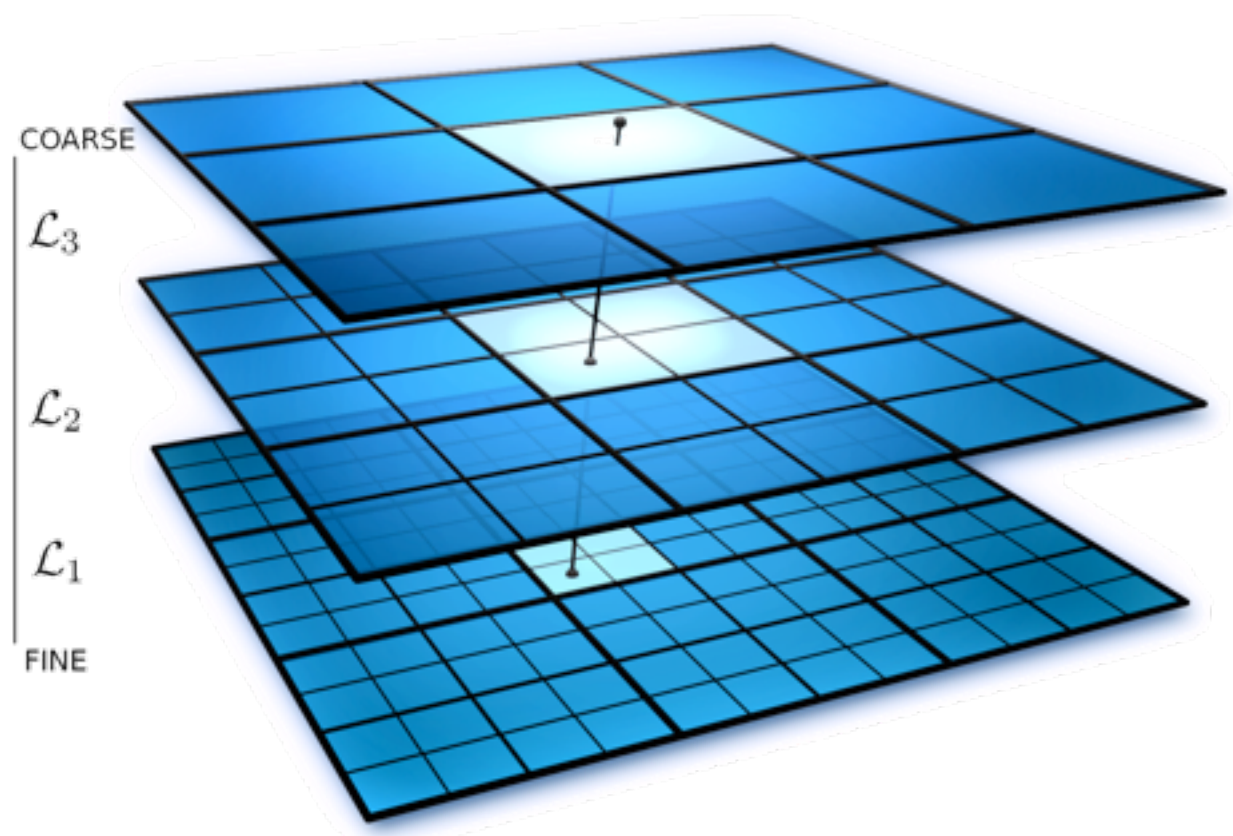
- SBL: Uses some coverage estimates and density of nodes. [Sanchez and Latombe, 2001]



KPIECE

- Keeps tract of coverage by using discretization and by distinguishing the boundary from the covered space.
- Keeping of coverage can be done in a hierarchical fashion.
- Projections my be used.

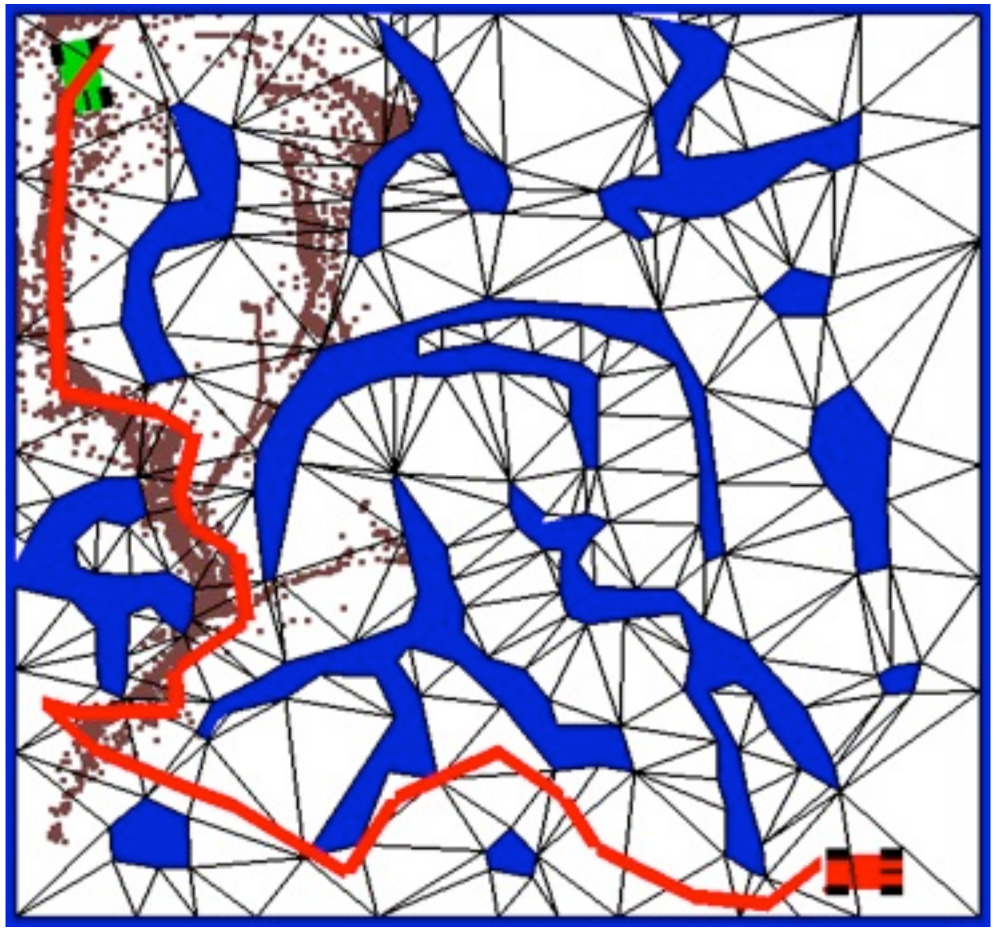




[Șucan, Kavraki 2008]

SyCLoP

- Using a discrete lead to help guide the expansion of the tree



Plaku and Kavraki, 2008

Performance improvements for trees

- Bi-directional search.
- Lazy collision checking.
- Goal biasing.
- Accounting for constrained manifolds.
- Employing motion primitives.
- and many others.

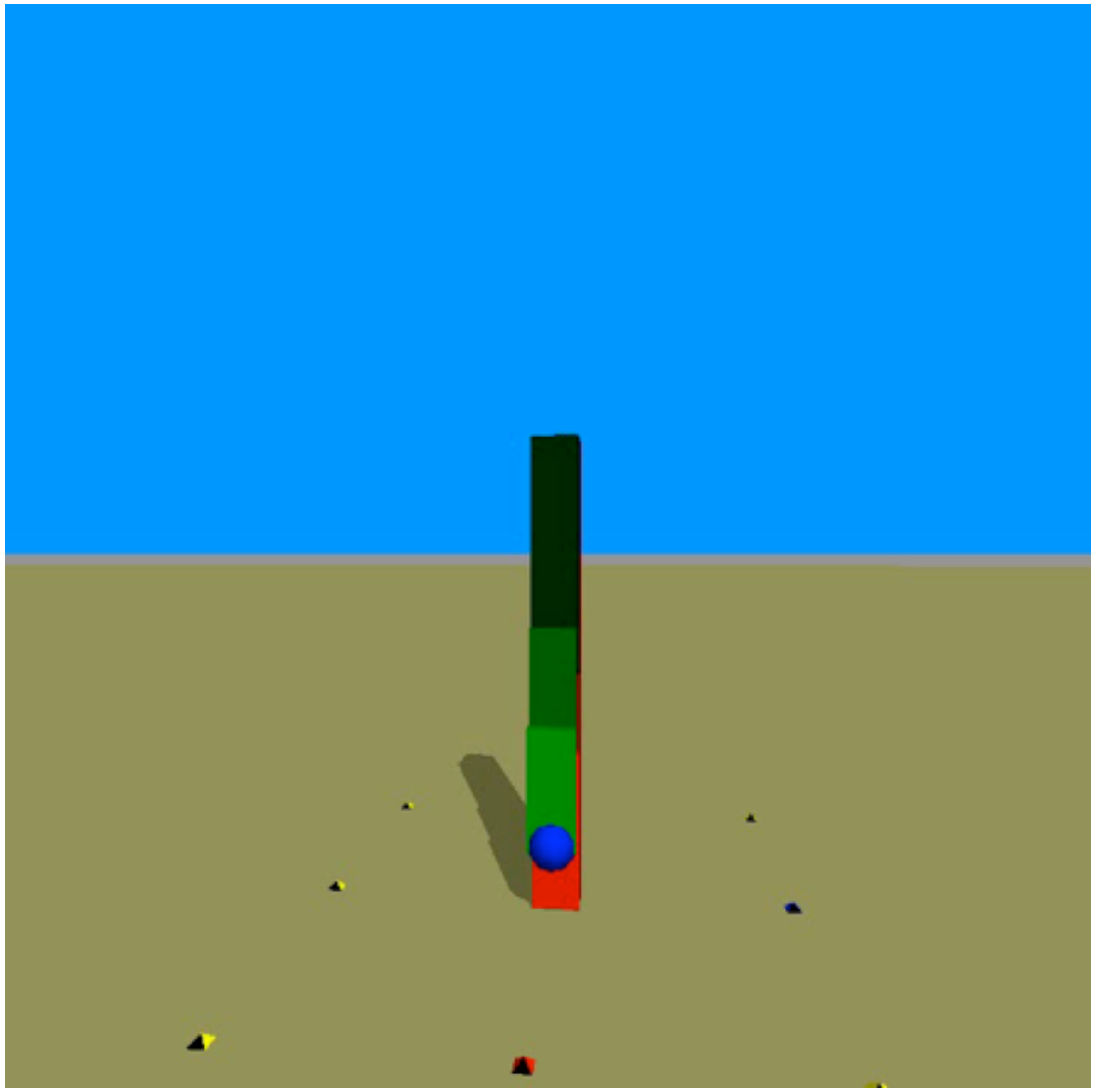
Planning with Dynamics:
Trees offer an advantage

Planning with dynamics

Bekris et al.

Ladd et al.

Planning with dynamics



Ladd et al.

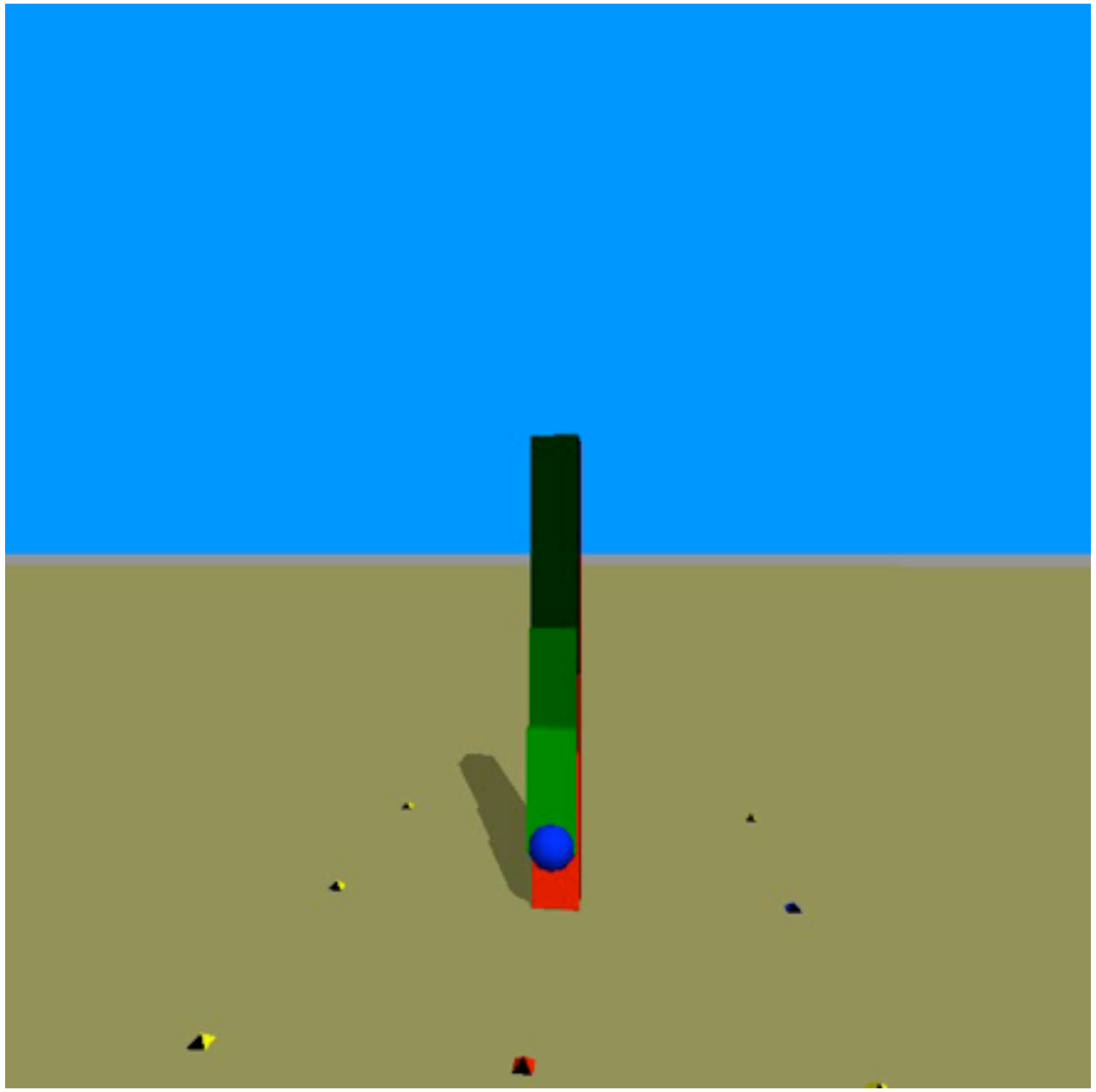
Bekris et al.

Planning with dynamics

Bekris et al.

Ladd et al.

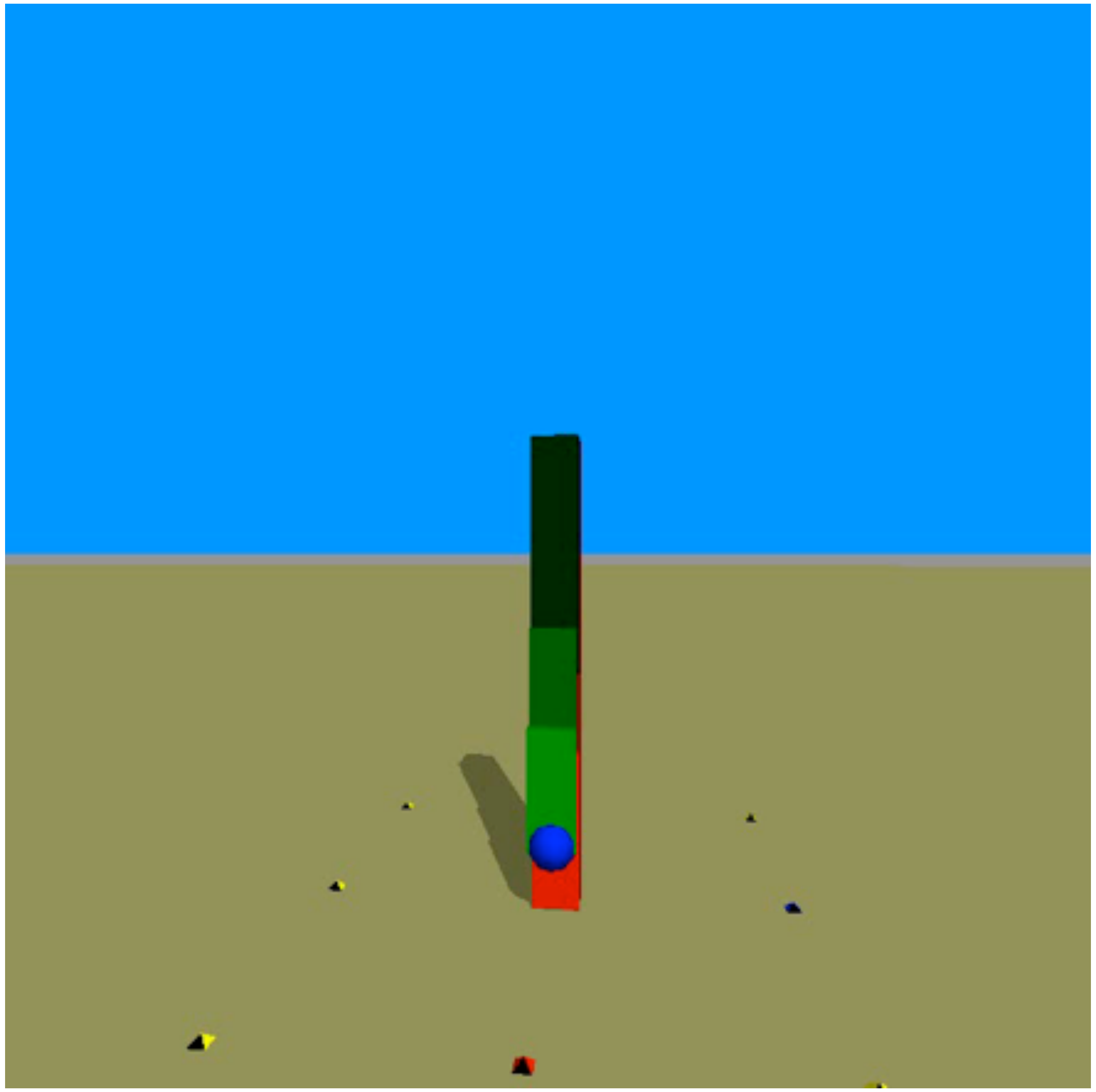
Planning with dynamics



Ladd et al.

Bekris et al.

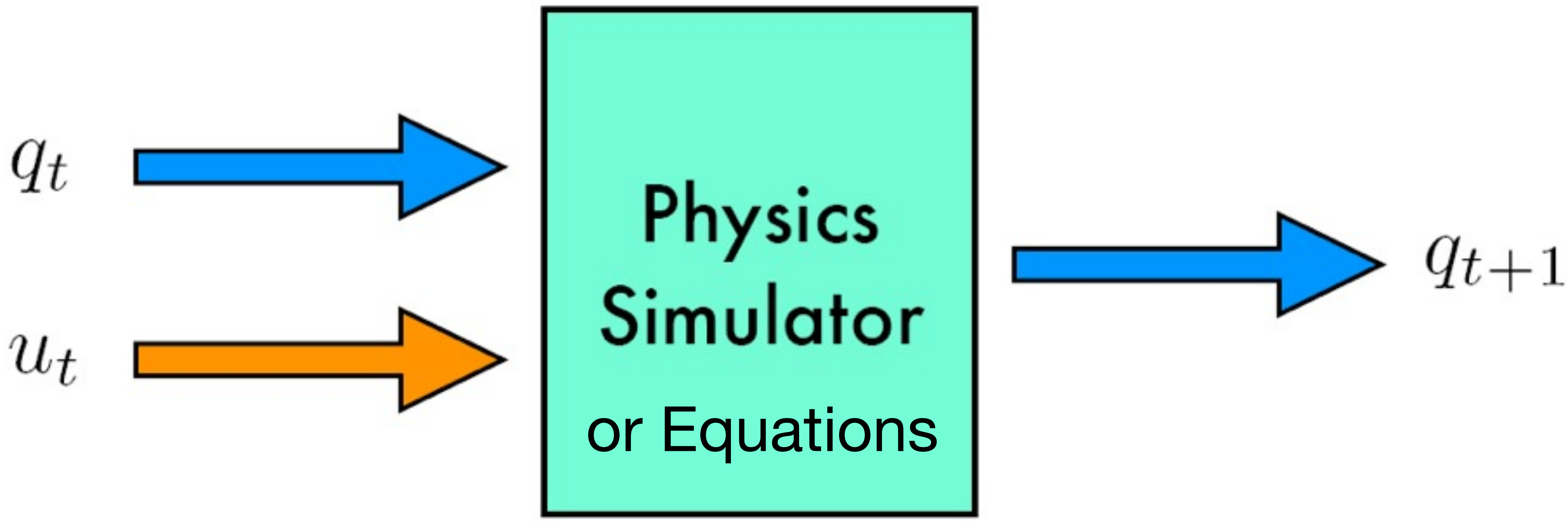
Planning with dynamics



Ladd et al.

Bekris et al.

Physical Systems Planning



$$F(q_t, u_t) = q_{t+1}$$

Space of controls is defined

Physical system planning

Given

1. an initial state $q_0 \in Q$

2. a goal set $G \subset Q$

The discrete physical systems planning problem is to compute a sequence $u_0, \ldots, u_N$ such that:

$$F(q_i, u_i) = q_{i+1}$$

and $q_{N+1} \in G$ is contained in the goal set.

Planning with dynamics

- Adding dynamics is essential to increase physical realism.
- Techniques from control theory can be used to create better paths or reduce differential equation integrations.
- Metrics tend to work poorly.
- Efficient planning for systems with dynamics is still fairly open: sampling-based tree planners offer an advantage.

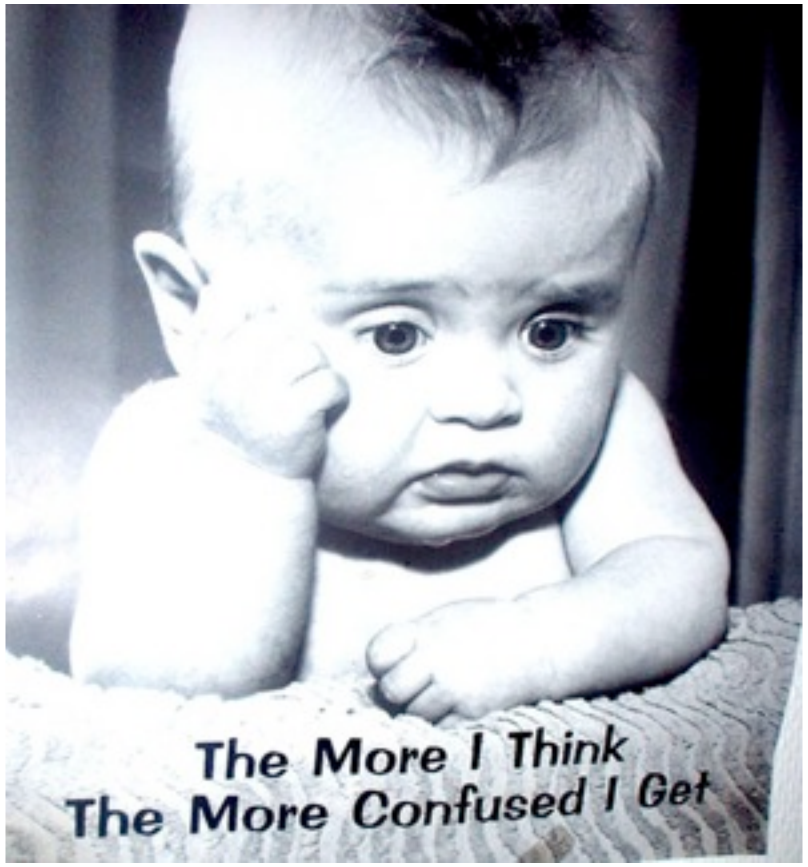
Primitives

- **Select Sample**
- **Expand from the sample**
- **Store efficiently**

These primitives are combined with various optimizations.

Variations of tree sampling-based planners

EST [Hsu et al.’97, ‘00]
RRT [Kuffner, LaValle ’98]
RRT-Connect [Kuffner, LaValle '00]
SBL [Sanchez, Latombe ’01]
Guided EST [Phillips et al. ‘03]
PDRRT [Ranganathan, Koenig '04]
SRT [Plaku et al. '05]
DDRRT [Yershova et al. ’05]
ADDRRT [Jaillet et al. ’05]
RRT-Blossom [Kalisiak, van Panne ’06]
PDST [Ladd, Kavraki ‘06]
Utility RRT [Burns, Brock ’07]
GRIP [Bekris, Kavraki ’07]
Multiparticle RRT [Zucker et al. ’07]
TC-RRT [Stillman et al. '07]
RRT-JT [Vande Wege et al '07]
DSLX [Plaku, Kavraki, Vardi '08]
KPIECE [Șucan, Kavraki '08]
RPDST [Tsianos, Kavraki '08]
BiSpace [Diankov et al. '08]
GRRT [Chakravorty, Kumar '09]
IKBiRRT [Berenson et al.'09]
CBiRRT [Berenson et al.'09]
J+RRT [Vahrenkamp '09]
RRT* [Karaman et al, 10]
and many others



Sampling-based planning (many possibilities)

- Core operations
  - state sampling
  - connection strategy
  - ......
- Common optimizations
  - bi-directional
  - goal-biasing
  - .....

Sampling-based planning (many possibilities)

- Core operations
  - state sampling
  - connection strategy
  - ......
- Common optimizations
  - bi-directional
  - goal-biasing
  - .....

Need for a systematized approach: OMPL

Benefits

- A repository of planners: choose the right planner and right parameters for that planner.
- Compare new planners to existing ones.
- Develop significantly more complex specialized planners.
- Enable challenging research.
- Support education of new scientists.

Challenges

- Uncertainty.
- Manipulation of rigid and flexible objects.
- Parallel Linkages.
- Dynamics.
- Hybrid planning.
- Real-time planning.
- and other.

THANK YOU

Acknowledgements: Work at the Kavraki Lab on sampling-based planners has been supported by NSF