Olasılıksal Robotik

Dr. Öğr. Üyesi Erkan Uslu

12

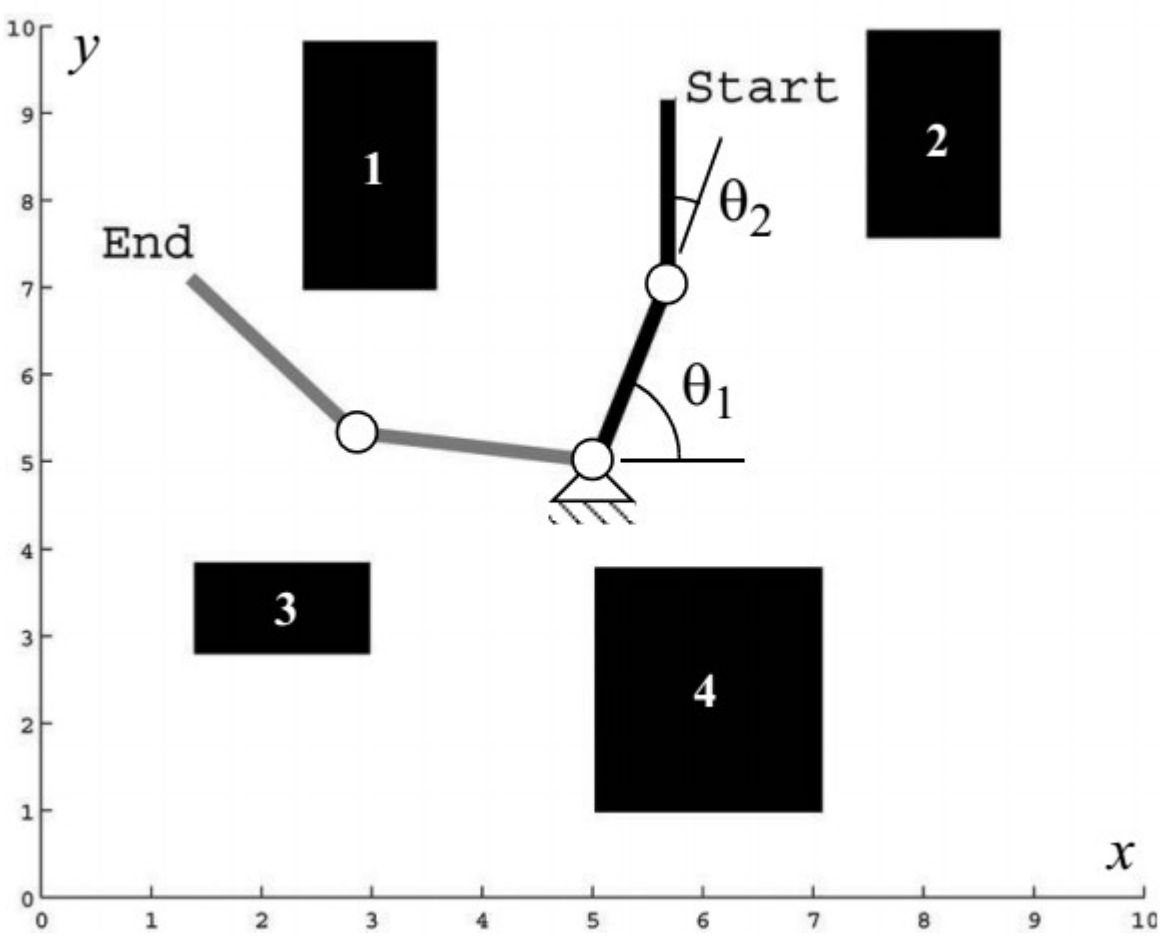
Path Planning

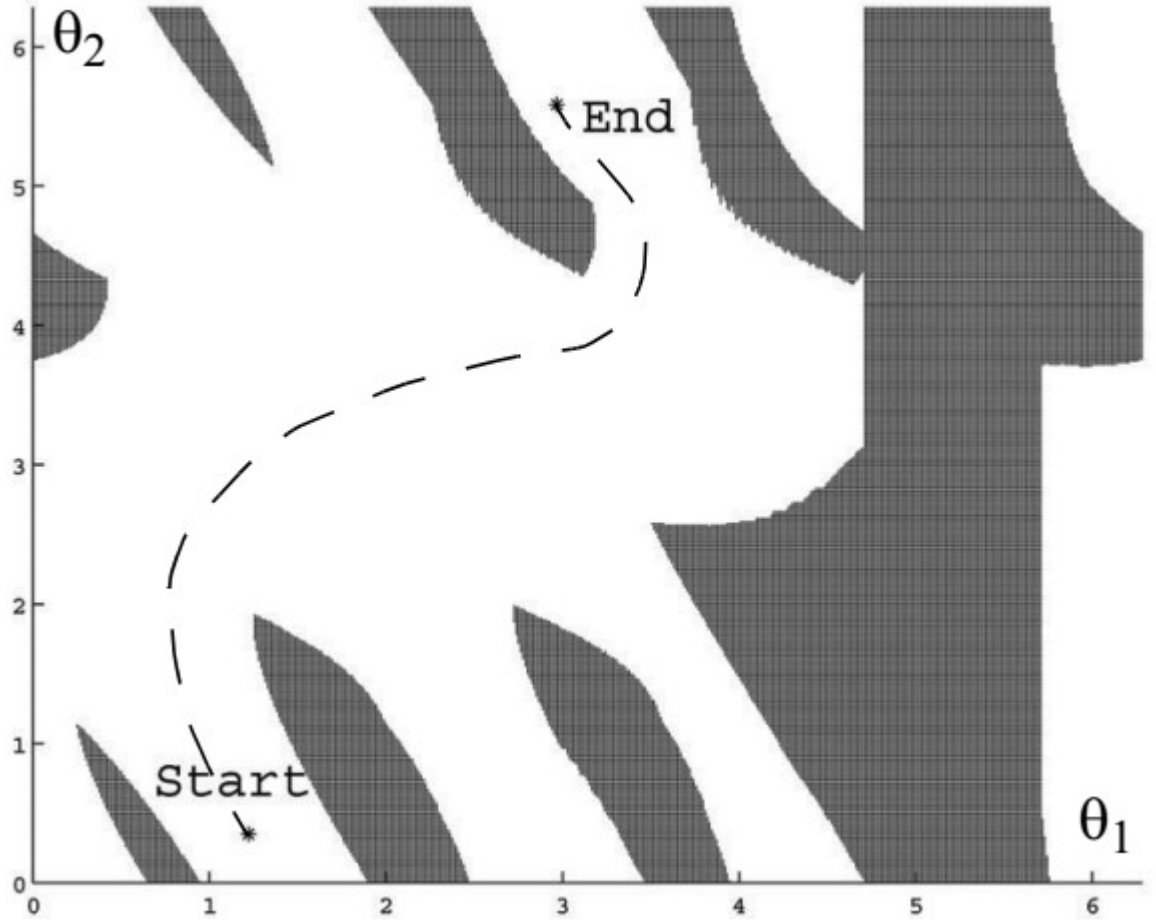
- Given
- a START pose,
- a desired GOAL pose,
- a geometric description of the ROBOT and
- a geometric description of the WORLD
- find a path that moves the robot gradually from start to goal.

Configuration Space – Robot Arm

- Suppose a robot arm has k degrees of freedom
- Every state of the robot arm can be described with k-real values
- K-values representing a point in the k dimensional space : configuration space

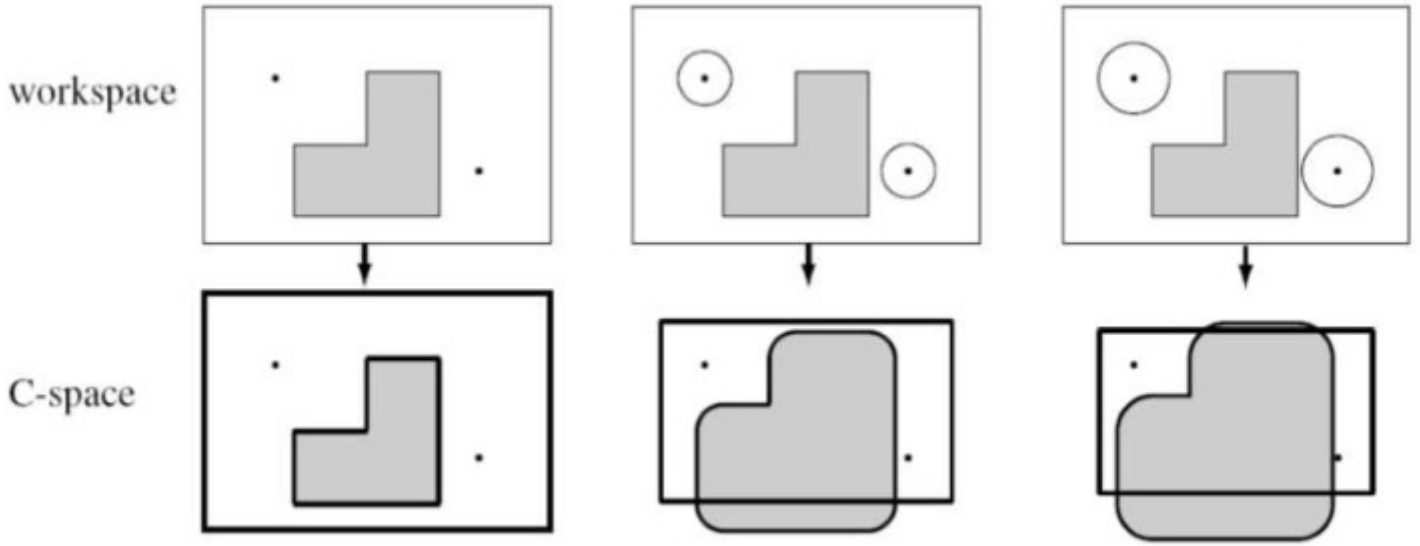
Configuration Space – Robot Arm





Configuration Space – Mobile Robot

- Assume robot is simply a point
- Each obstacle is inflated by the size of robots radius

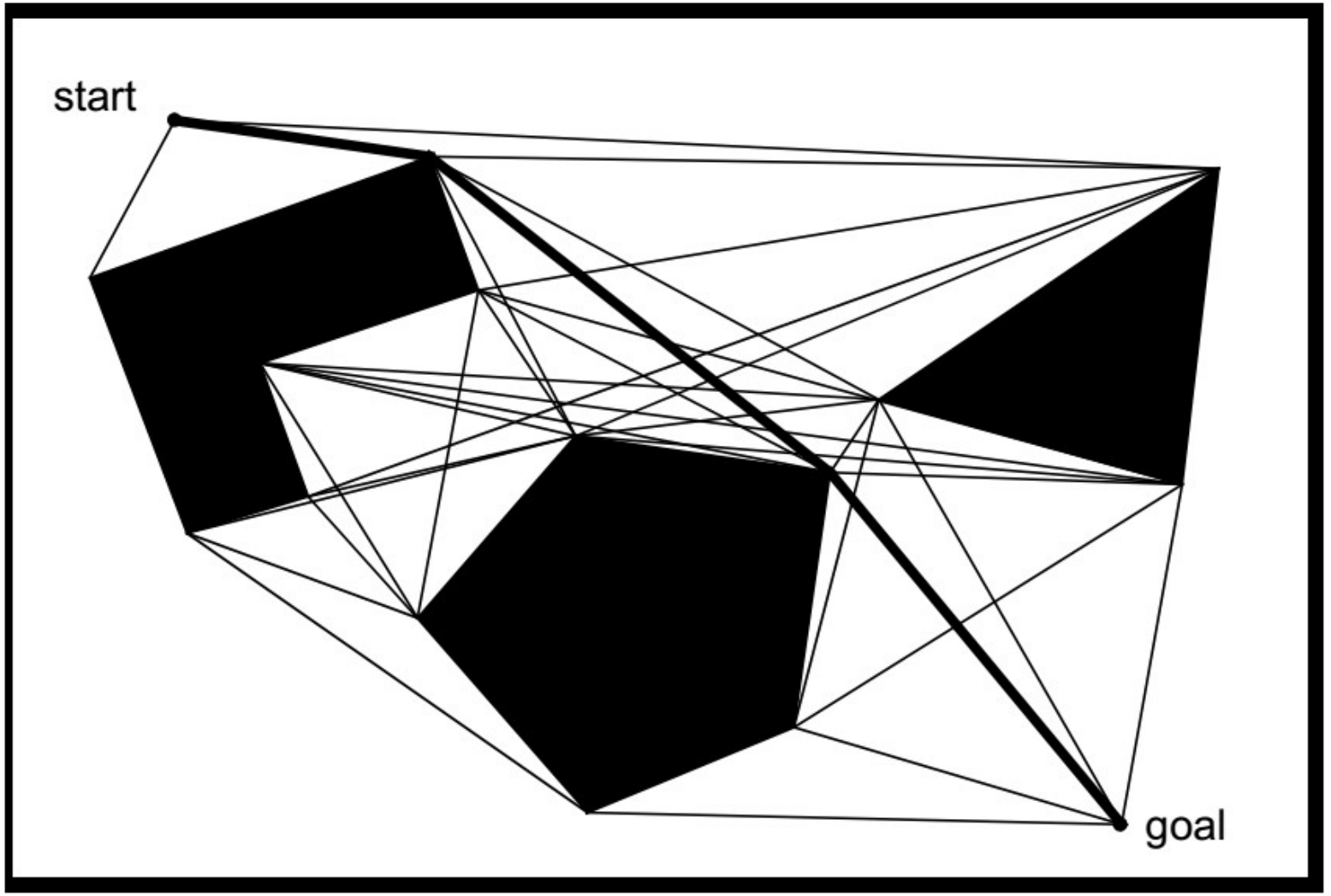


Path Planning

- Graph Search
  - Graph Construction
    - Visibility Graph
    - Voronoi Diagram
    - Exact Cell Decomposition
    - Approximate Cell Decomposition
  - Deterministic Graph Search
    - Breadth-First Search
    - Depth-First Search
    - Dijkstra’s Algorithm
    - A*
    - D*
  - Randomized Graph Search
    - Rapidly Exploring Random Trees
- Potential Field Path Planning

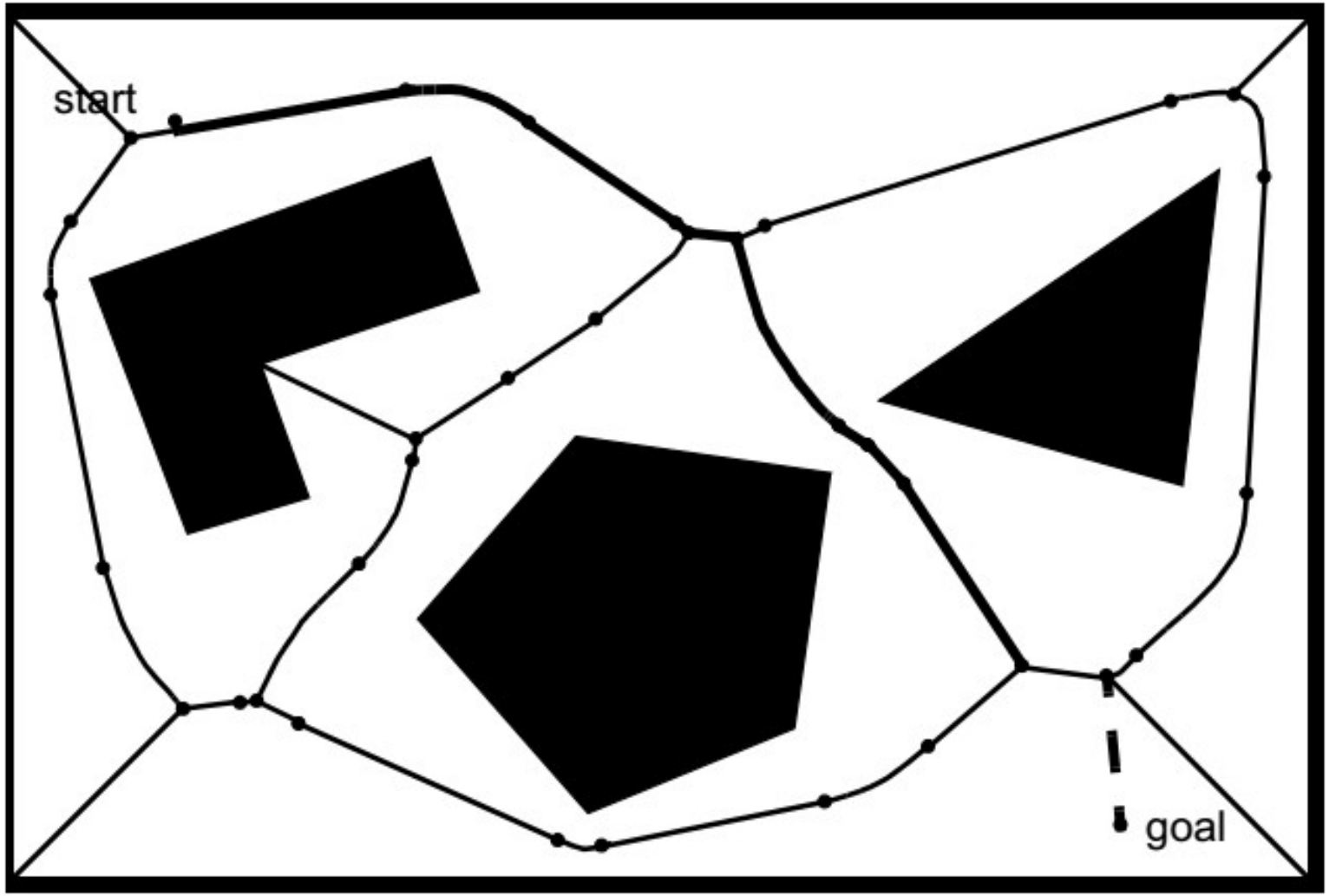
Visibility Graph

- Vertices of the graph are:
  - Start
  - Goal
  - Vertices of the obstacles visible from each other
- Visibility graphs are optimal in terms of path length



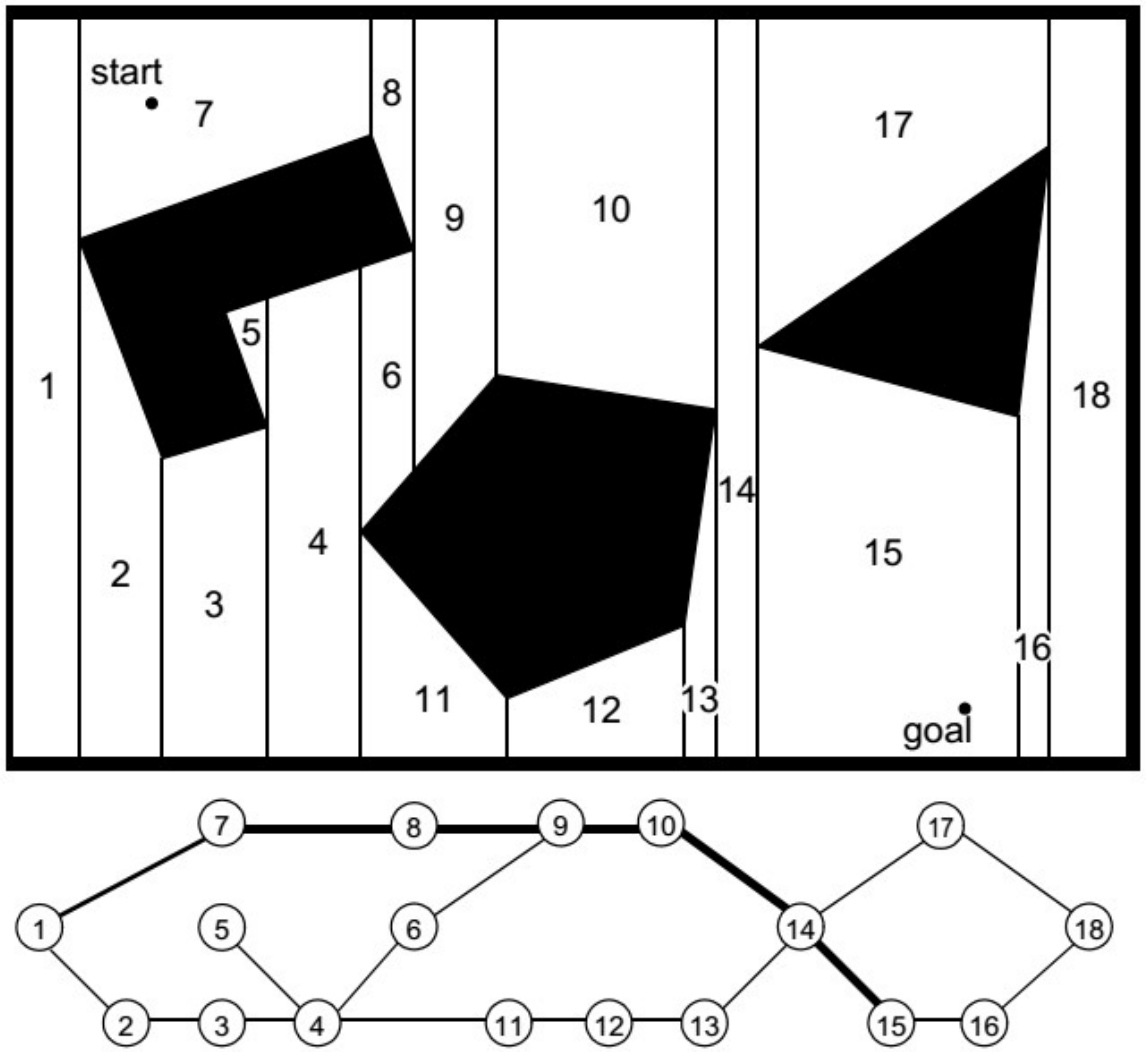
Voronoi Digram

- Maximize the distance between the robot and the obstacles
- Diagram points are equidistant from two or more obstacles



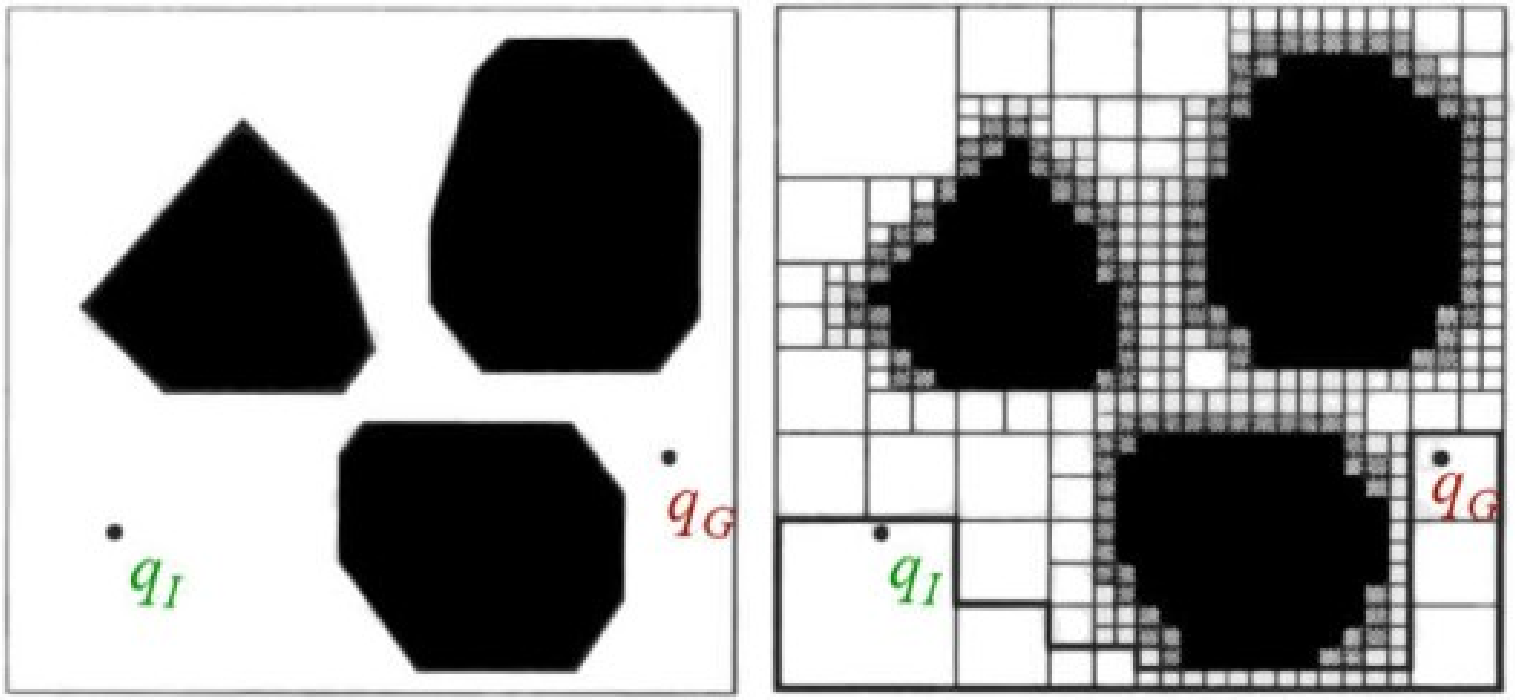
Exact Cell Decomposition

- Cell boundaries are based geometric criticality
- In extremely sparse environmets representation is efficient



Approximate Cell Decomposition

- Approximate Cell Decomposition uses cells with the same simple pre defined shape in different scale



Quadtree decomposition

Deterministic Graph Search

- Expected total cost :
- Path cost :
- Edge traversal cost :
- Heuristic cost :
- Can be defined for a node and an adjacent node

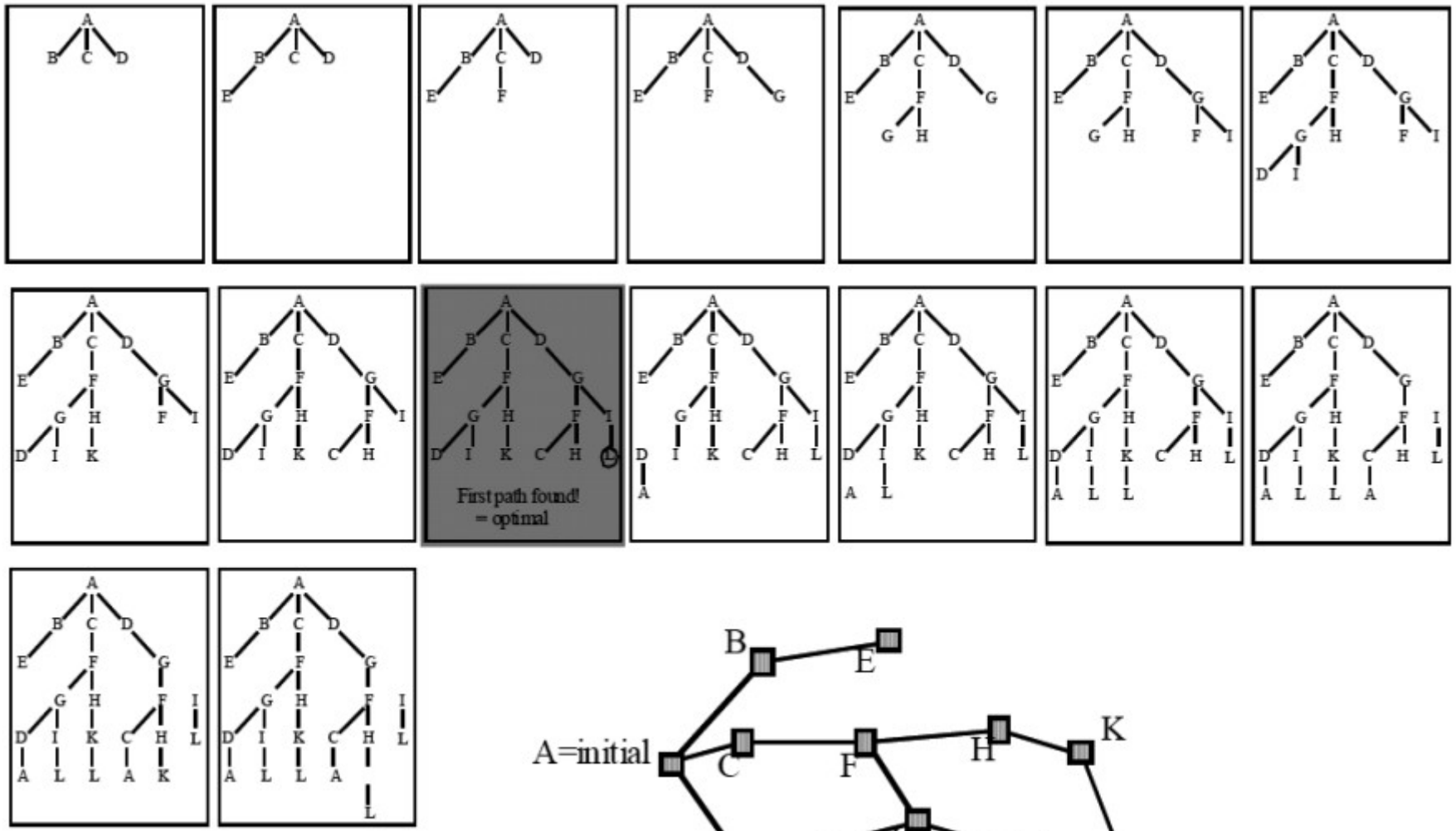
Deterministic Graph Search

- ☾ Depth-First, Breadth-First
- AND ☾ Dijkstra's Algorithm
- ☾ Optimal A*
- ☾ Suboptimal A*

Breadth-First Search

- Algorithm begins with the start node
- Explores all of its neighboring nodes
- Then, for each of these nodes, it explores all their unexplored neighbors and so on.
- This process goes as, marking a node “active”, exploring each of its neighbors and marking them “open”, and finally marking the parent node “visited
- The algorithm proceeds until it reaches the goal node where it terminates.

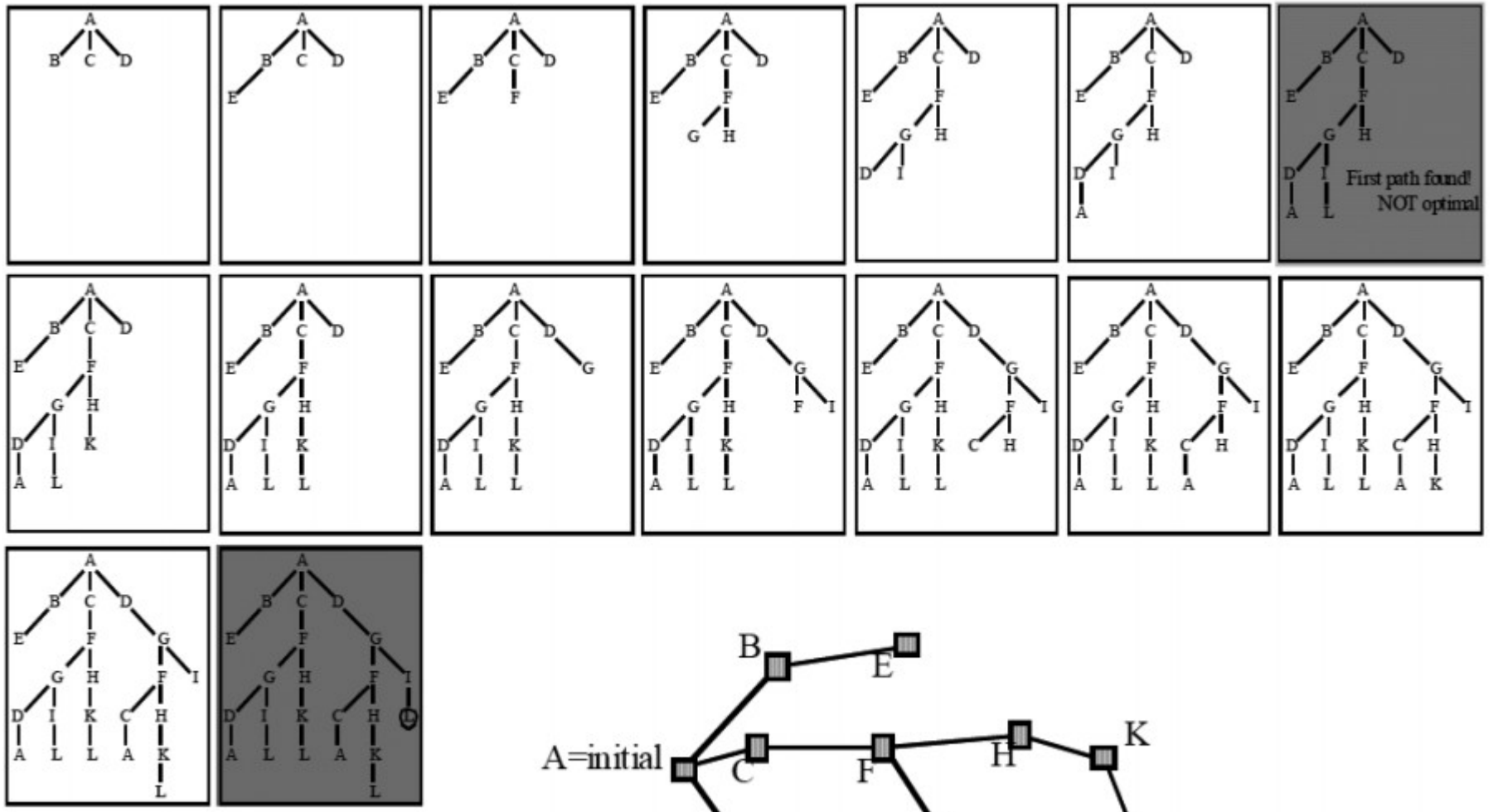
Breadth-First Search



Depth-First Search

- Depth-first search expands each node up to the deepest level of the graph
- May provide non optimal solutions
- May have better space complexity compared to breadth-first search

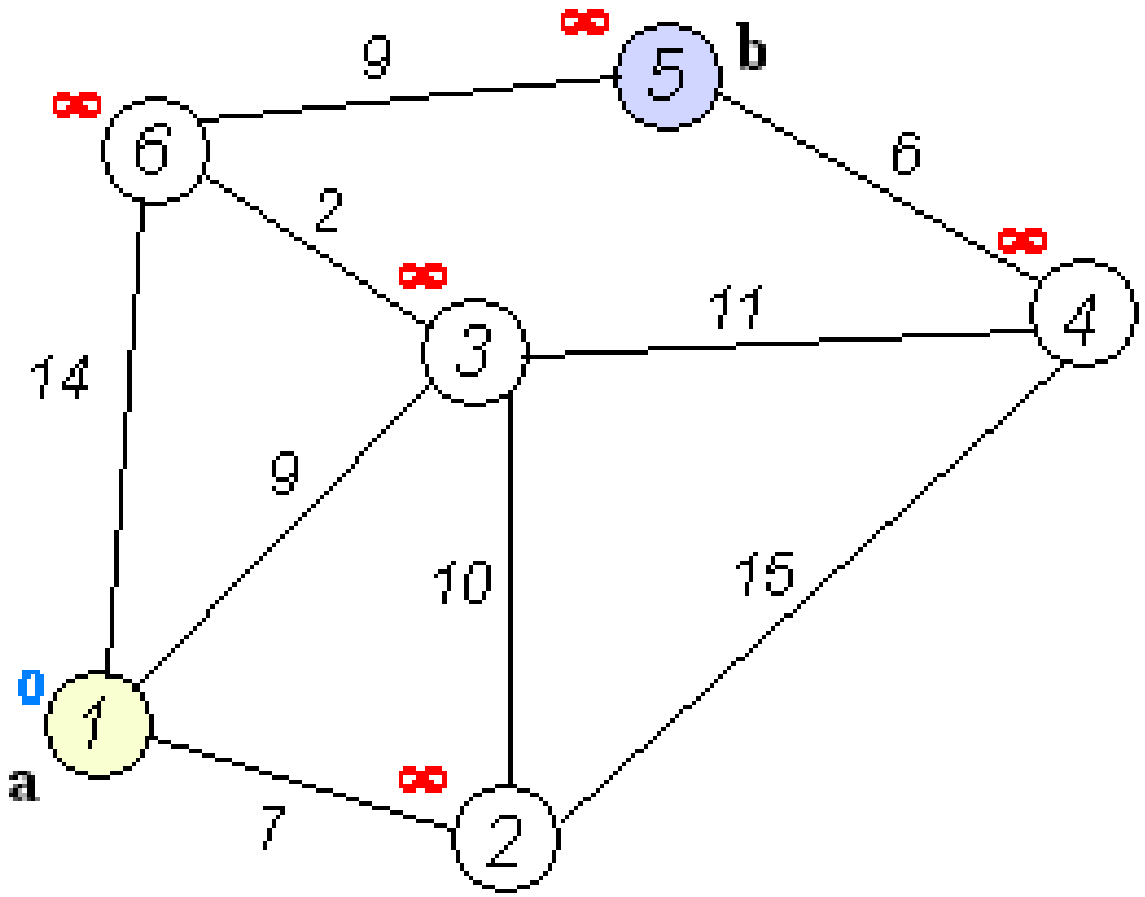
Depth-First Search



Dijkstra’s Algorithm

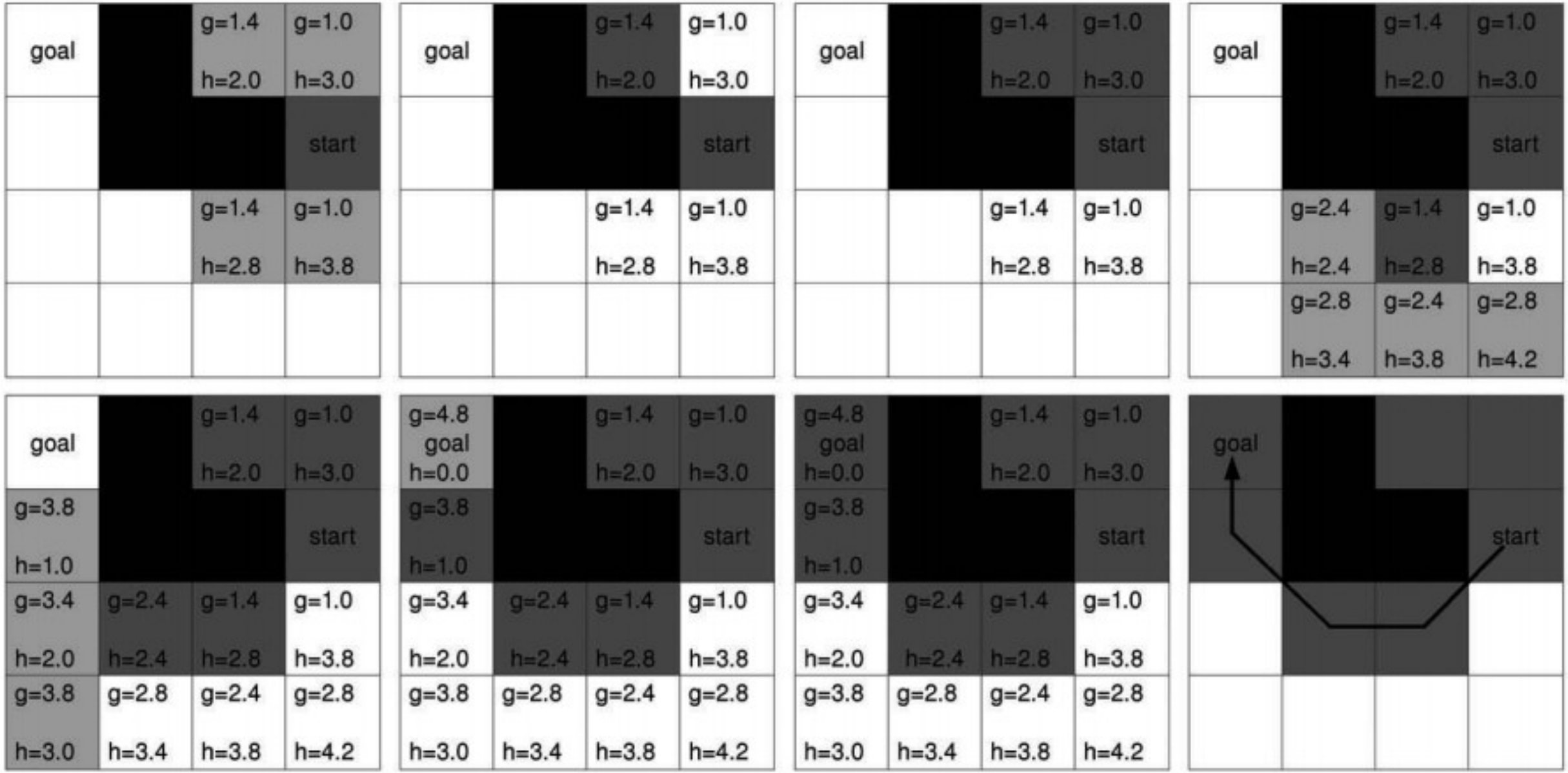
- Similar to breadth-first search
- Edge costs may assume any positive value
- Guarantees solution optimality
- No heuristic is used
- Is a greedy algorithm yet produces an optimal solution

Dijkstra’s Algorithm



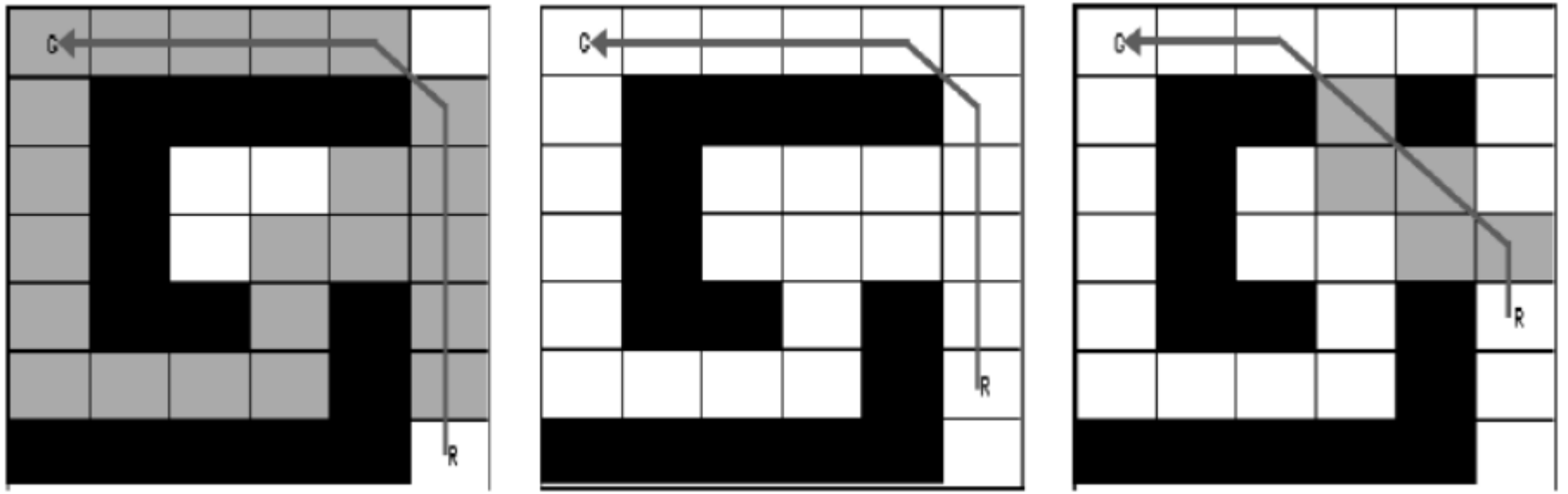
A*

- Similar to Dijkstra
- Includes an underestimated function of the cost to go
- Heuristic function is the distance from any node to goal


goal
g=1.4
h=2.0
g=1.0
h=3.0
start
g=1.4
h=2.8
g=1.0
h=3.8
g=2.4
h=2.4
g=2.8
h=3.4
g=2.8
h=3.8
g=2.8
h=4.2
g=4.8
goal
h=0.0
g=3.8
h=1.0
g=3.4
h=2.0
g=2.4
h=2.4
g=2.8
h=3.0
g=2.8
h=3.4

D*

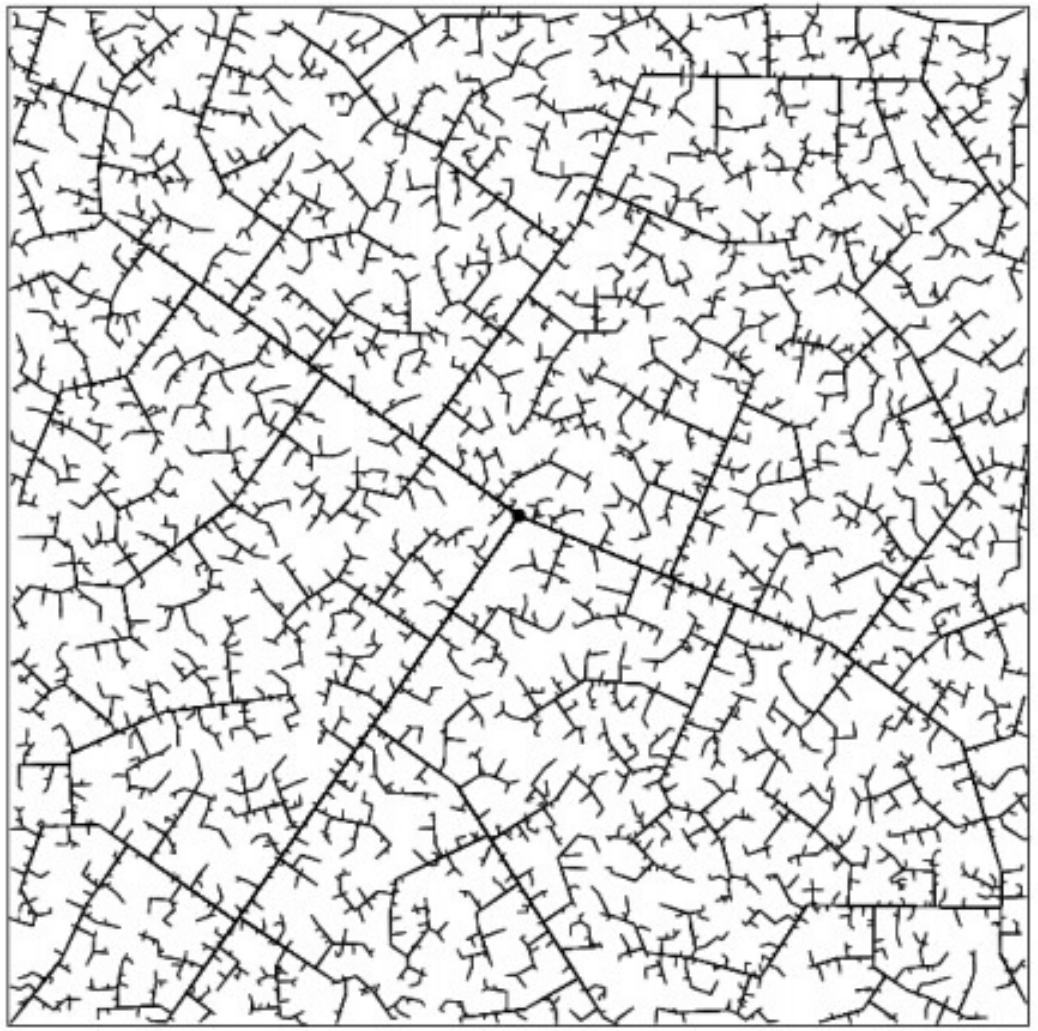
- Incremental replanning version of A*
- From start to goal A* path is calculated
- If a change occures, instead of calculating path from scratch, only effected states are recomputed



Rapidly Exploring Random Trees

- Doesn’t require graph decomposition
- Obstacle map is required
- RRTs grow a graph online during search
- Random nodes are generated and edges are grown from nearest nodes to randomly generated nodes
- Solution optimality is not guaranteed
- Deterministic completeness is not guaranteed

Rapidly Exploring Random Trees



Potential Field Path Planning

- Calculates a field/gradient across robot’s map
- The goal acts as an attractive force
- Obstacles act as repulsive forces
- Superposition of repulsive and attractive forces is applied to robot
- Smoothly guides robot to goal, simultaneously avoiding obstacles

Gradient of potential field

Potential Field Path Planning