Lecture 14: **Path Planning II**

Autonomous Robots, 16-200

Carnegie Mellon University Qatar

THE ROBOTICS INSTITUTE

جامعة كارنيجي ميلون قطر

Carnegie Mellon®

QATAR CAMPUS

Overview

- Logistics
- Movie
- Path Planning continued

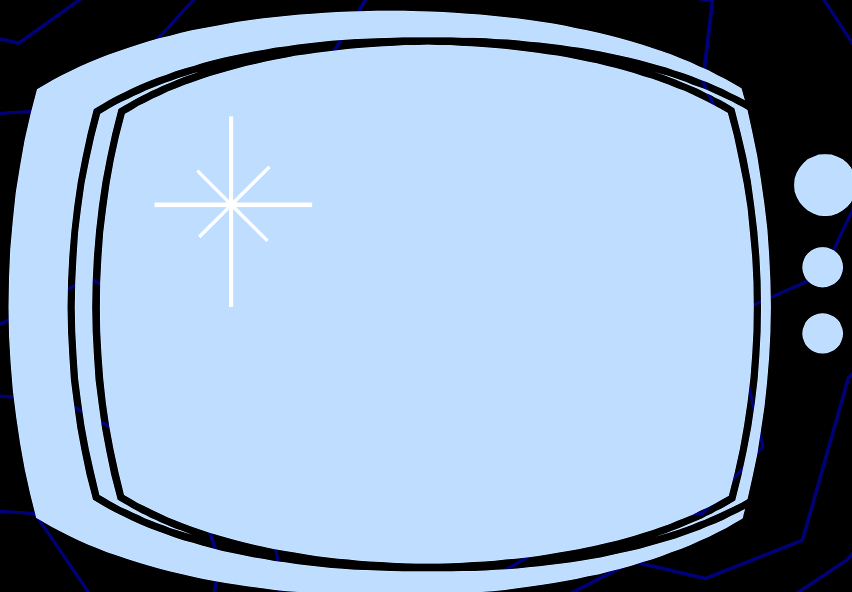
Logistics

- We will send you detailed feedback on your presentation and report as soon as we are done grading your reports
- Your project topics (title plus one paragraph) are important – come and see us if you need help with your topic
- HW #3 is due next Monday by 10:30a.m.
- Lab #3 optional second try due on Thursday during lab – sign up for slots! Timing will be very strict.
- We will complete the question portion of lab #3 tomorrow so come to the lab even if you don't want to repeat it
- Lab #4 will be assigned tomorrow

Plan

- Today:
  - Path Planning II
- Next week:
  - Coordination

Movie

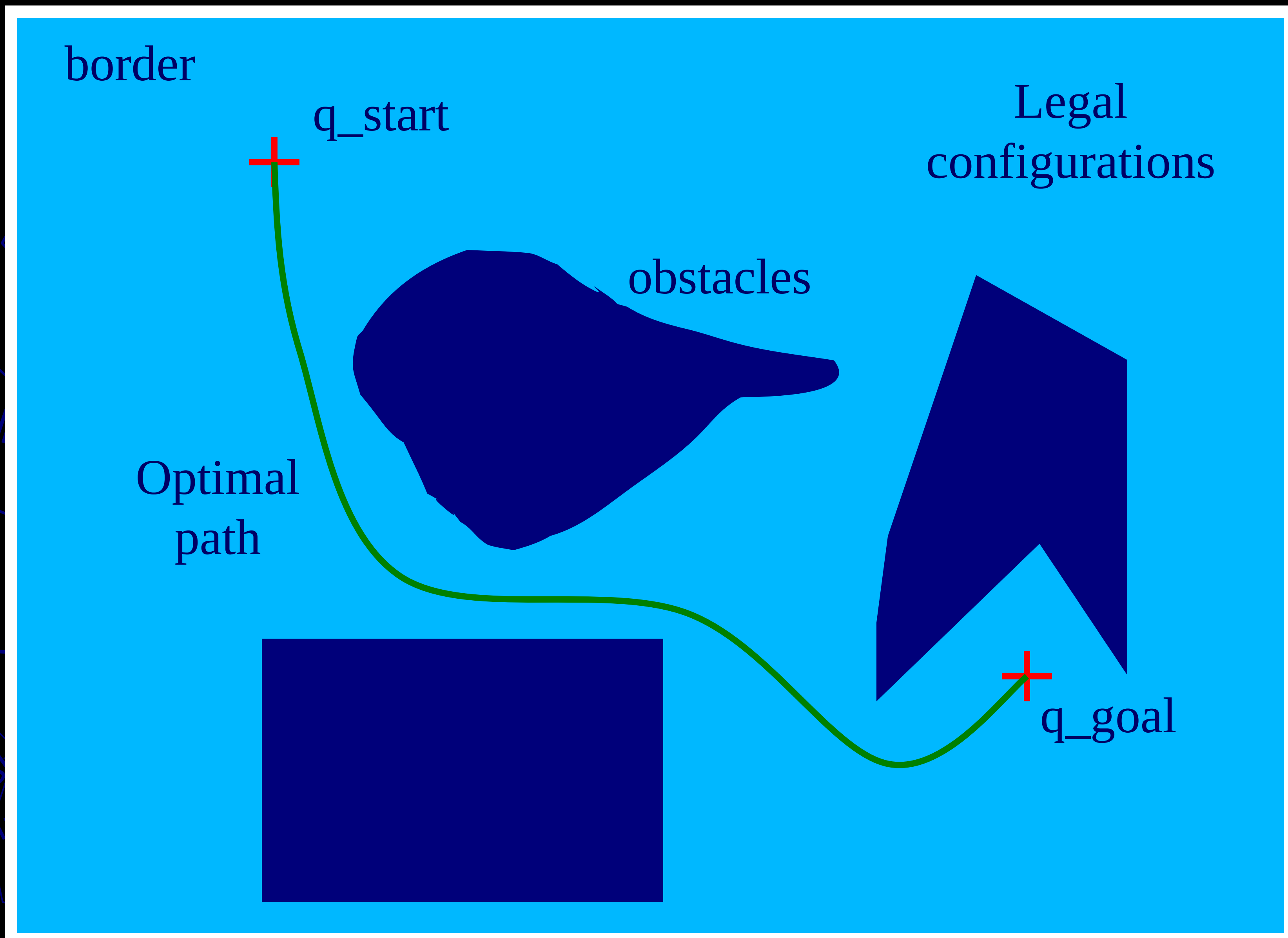


A Fun Movie ☺

Path Planning Problem

Given an initial configuration *q_start* and a goal configuration *q_goal*, we must generate the *best* continuous path of legal configurations between these two points, if such a path exists.

Illustration of Basic Definitions



C-Space Transform Steps

- For a mobile robot that *only* translates
  - Choose point on the robot.
  - Grow obstacle by translating robot around the edge of the obstacle and tracing line made by the reference point of the robot.
  - Represent the robot only by the reference point.
  - Legal configurations now consist of all non-obstacle points.
  - There is a trick to make this process easier.
- Robots that can rotate as well are usually represented by a circle.
- Once the obstacles have been grown, you can plan a path!

Path-Planning Algorithms

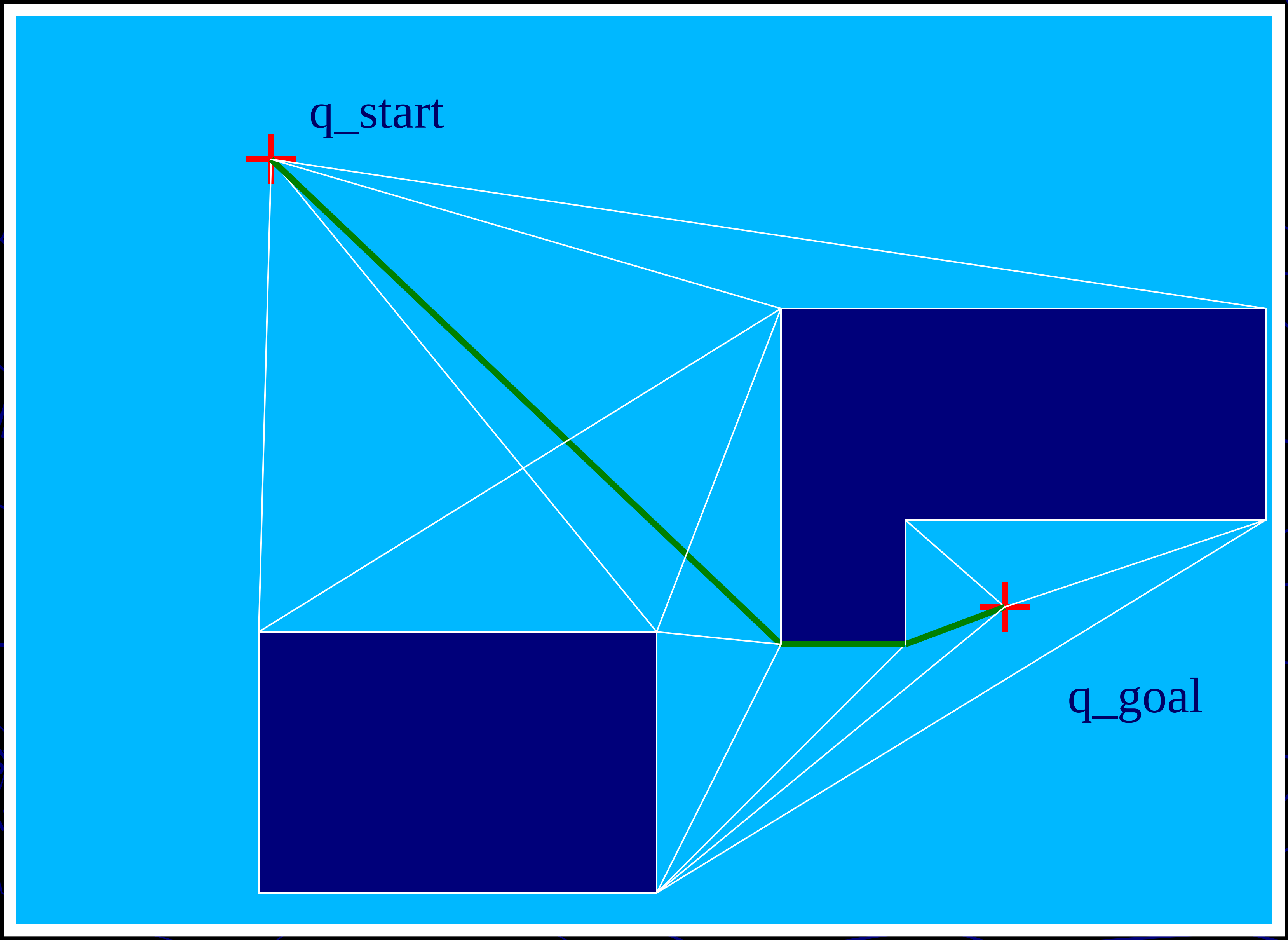
- Potential methods
- Visibility graphs
- Voronoi diagrams
- Cell decomposition

Note: We don't go over details of implementation here – just the concepts.

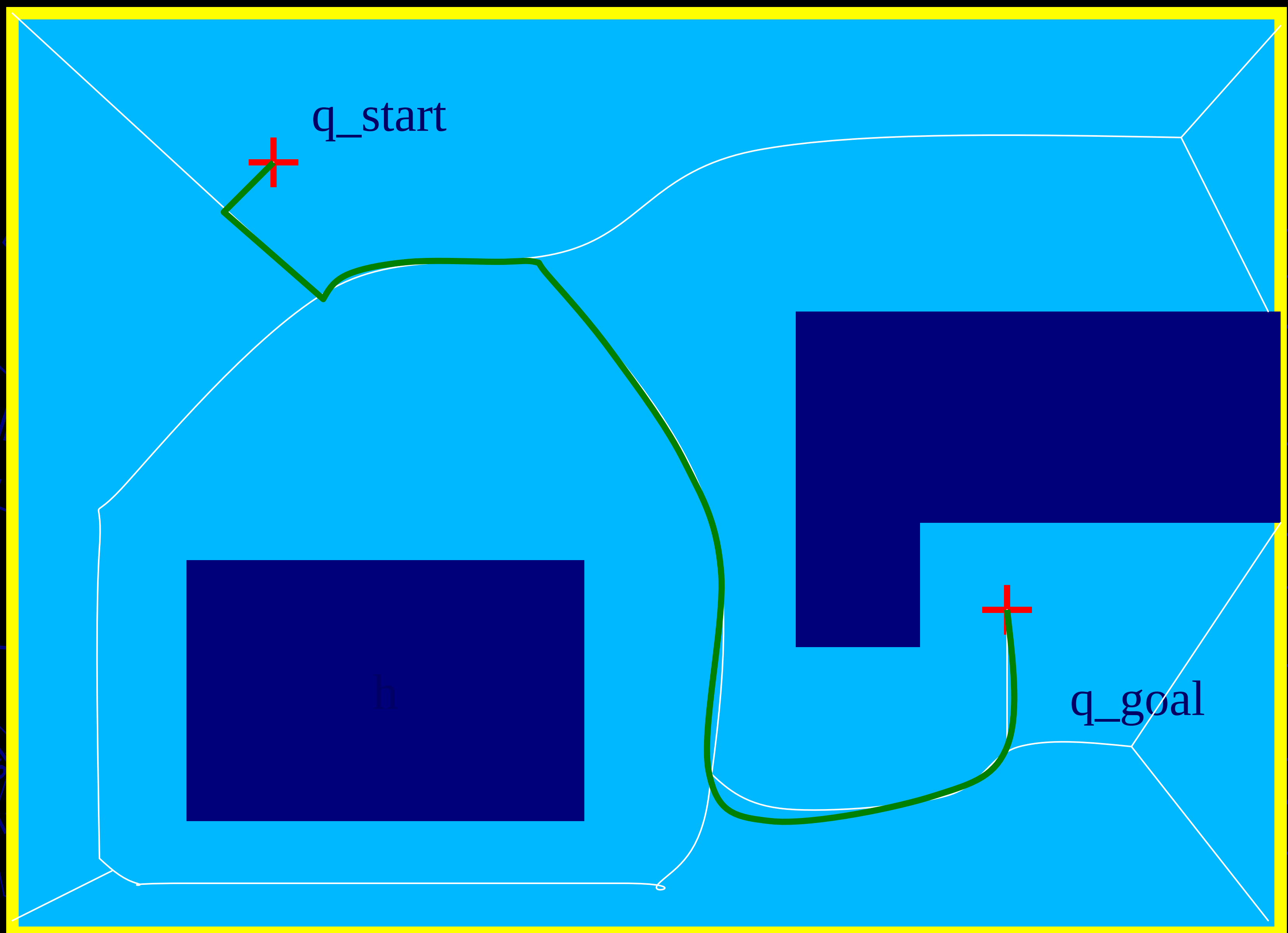
Potential Methods

- We already learnt Potential Fields – these can be used for path planning!
- Define *f(q)* such that:
  - *f* grows huge as the robot moves towards an obstacle
  - *f* grows small as the robot moves towards the goal
- Possible *f* functions:
  - $d_g(q)$ = distance from q to q_goal
  - $d_1(q)$ = distance from q to nearest obstacle
  - $f(q) = d_1(q) – d_g(q)$ OR
    $f(q) = 0.5\,\beta(d_g(q))^2 + 0.5\lambda(d_1(q))^{-2}$
- Path is given by following steepest descent on *f*

Visibility Graph Example



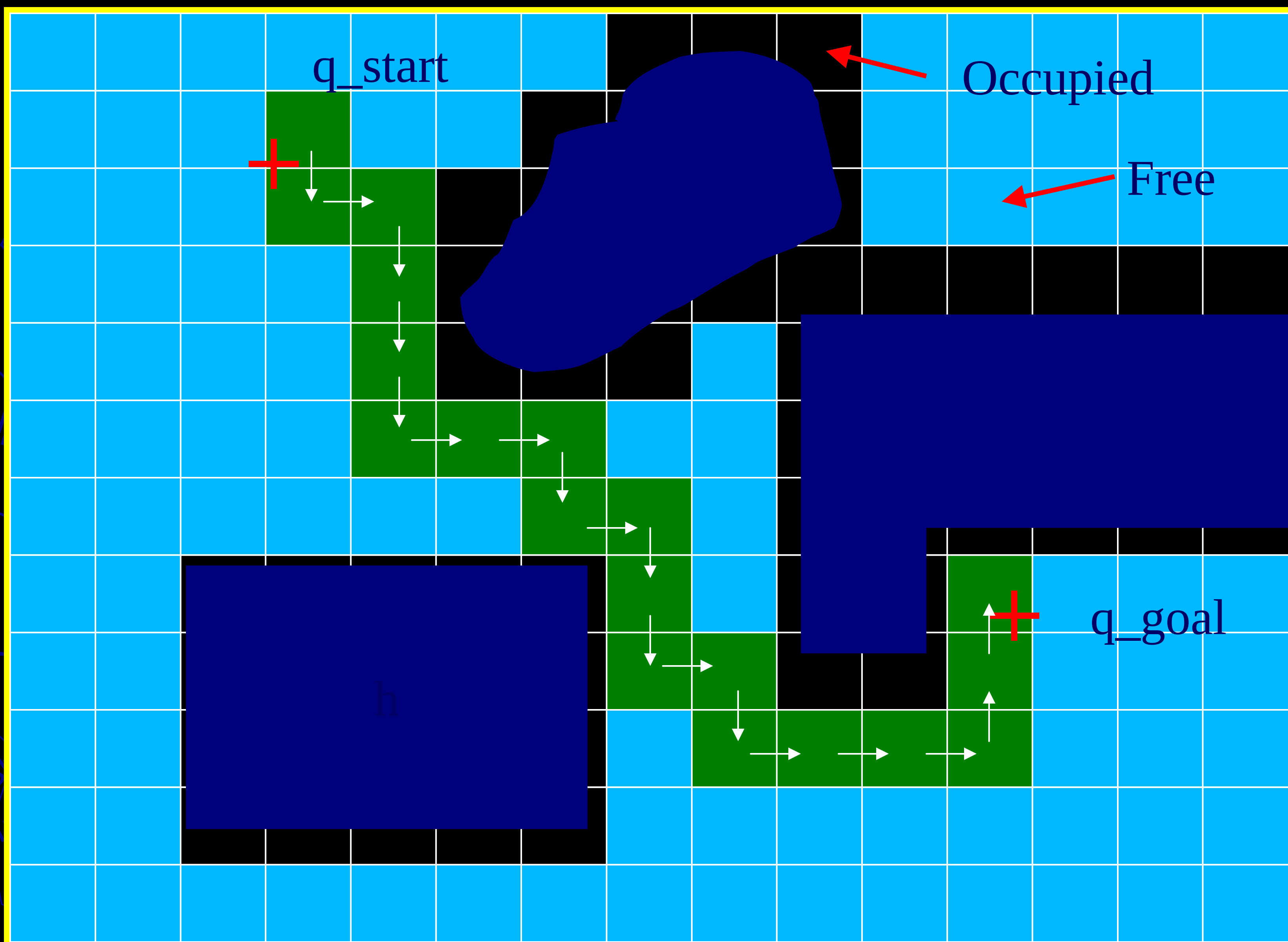
Voronoi Graph Example



Cell Decomposition

- Last approach: Divide C-space into convex polygons and plan between legal cells
- We'll use grids…but algorithm extends to general convex polygons of different sizes
- Algorithm:
  1. Start with C-space map
  2. Divide map into polygons
  3. Mark cells containing obstacles as occupied
  4. Search for path to goal using unoccupied cells

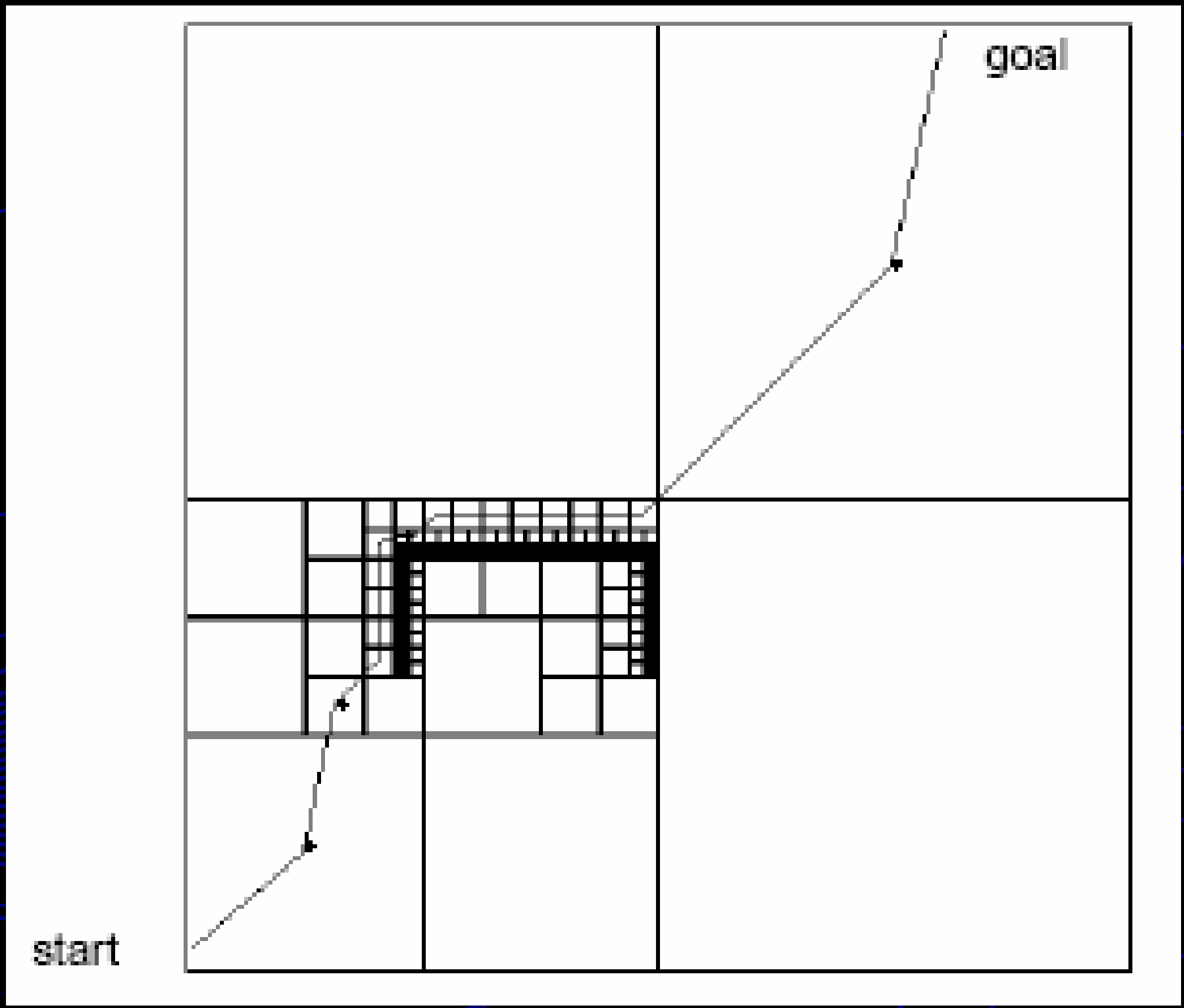
Cell Decomposition Example



Cell Decomposition in Practice

- Multiple ways to sub-divide C-space
  - Grids, Quad-trees
- Resolution is a limiting issue
  - Too fine a grid leads to long search time
  - Too coarse a grid misses paths
- May require post planning smoothing
- Requires a search algorithm
  - E.g. A* and D*

Quad-Tree Example

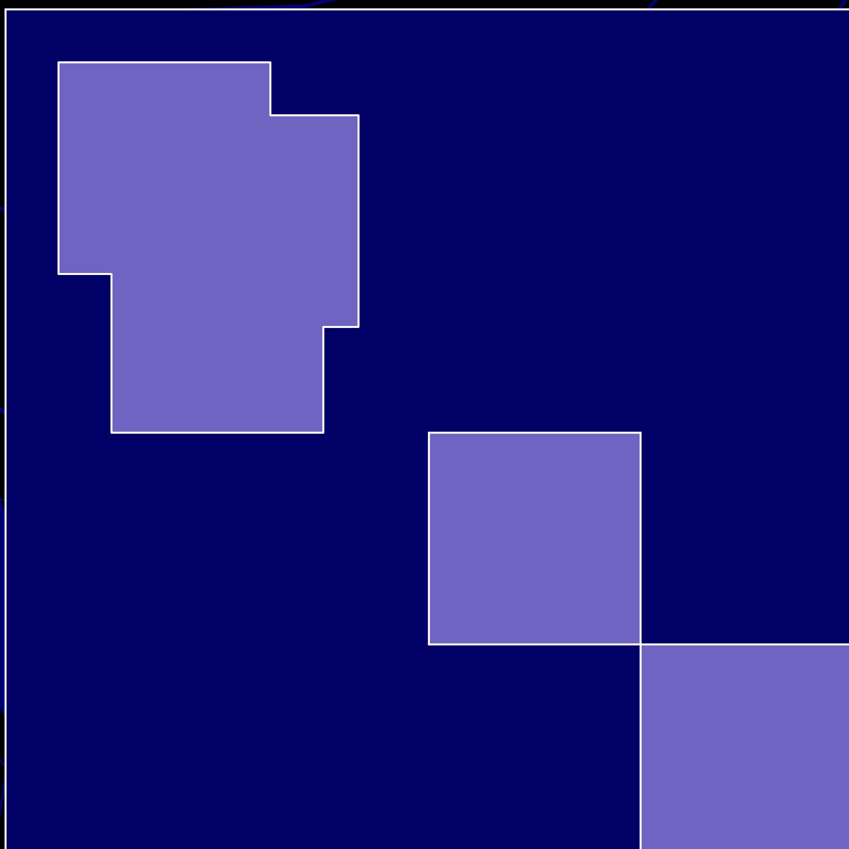


Framed-Quadtree Path Planning for Mobile Robots Operating in Sparse Environments

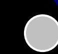
Alex Yahja, Anthony Stentz, Sanjiv Singh, and Barry L. Brumitt

Quadtree Example

Space Representation

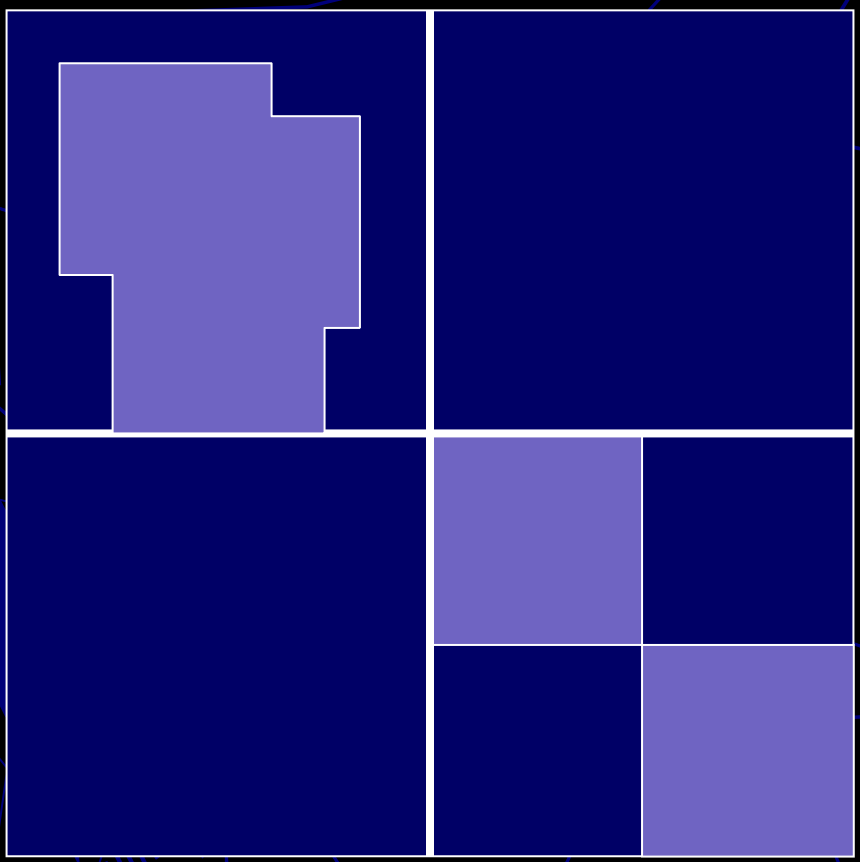


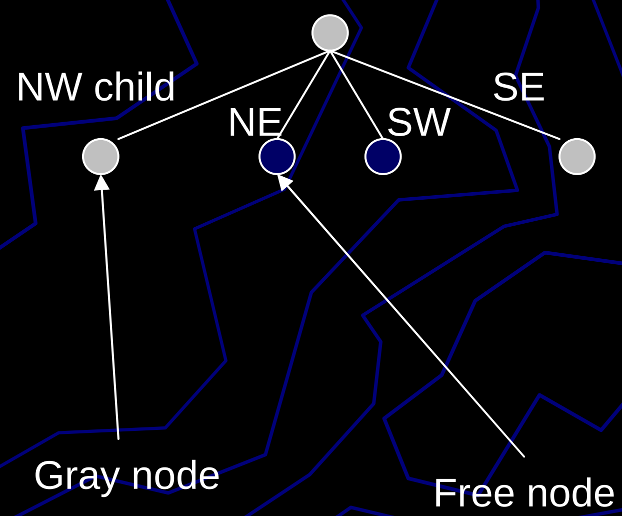
Equivalent quadtree



Russell Gayle, The University of North Carolina, Chapel Hill

Quadtree Example

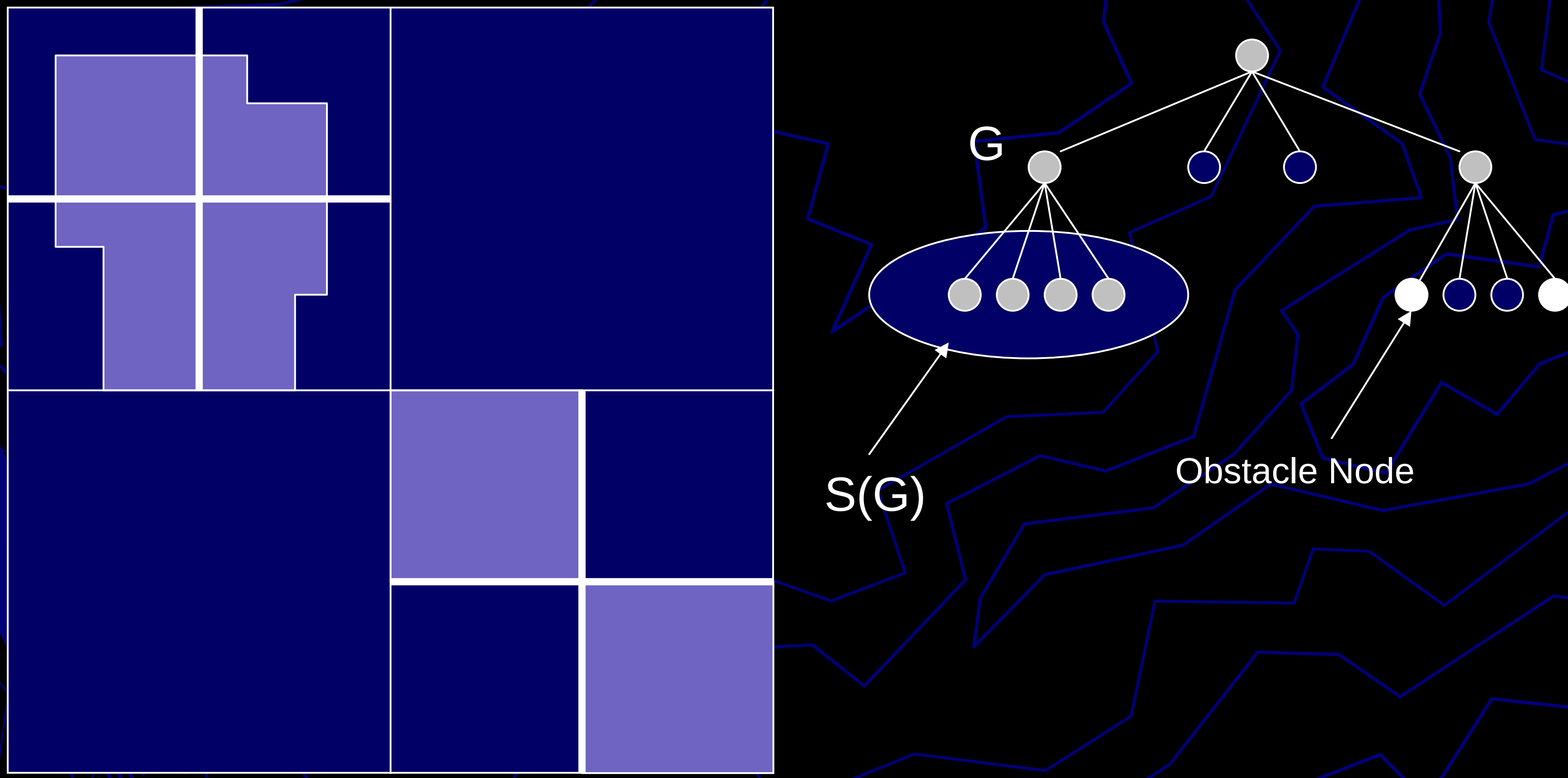




Quadtree Example

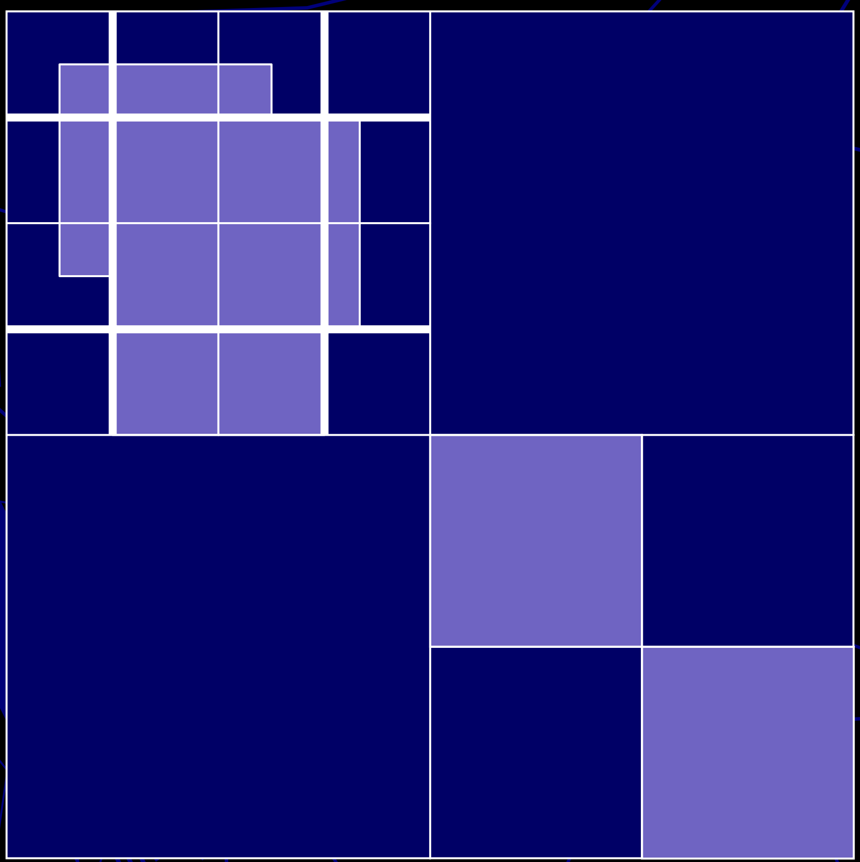
Space Representation

Equivalent quadtree

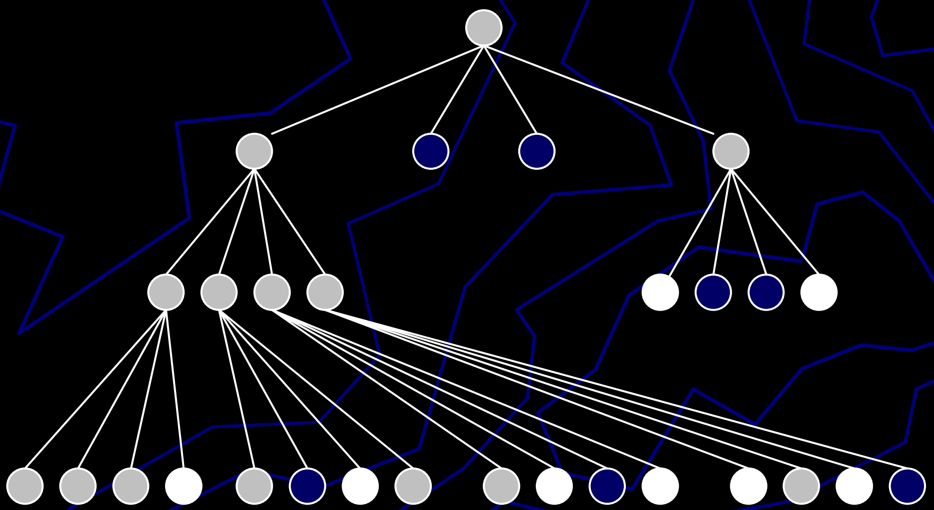


Quadtree Example

Space Representation



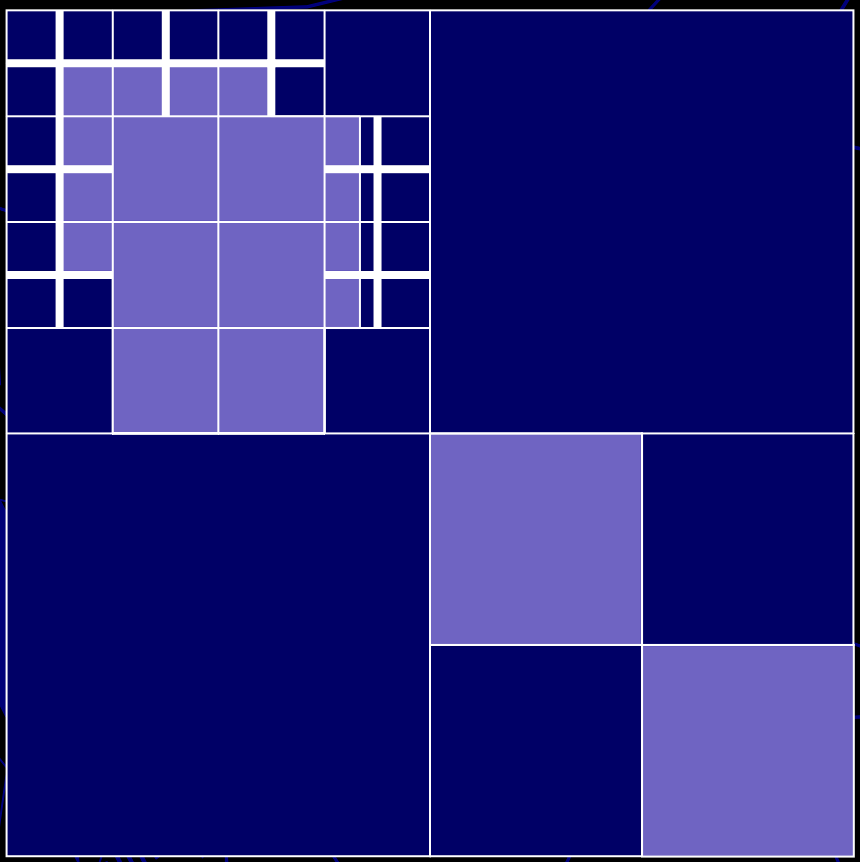
Equivalent quadtree



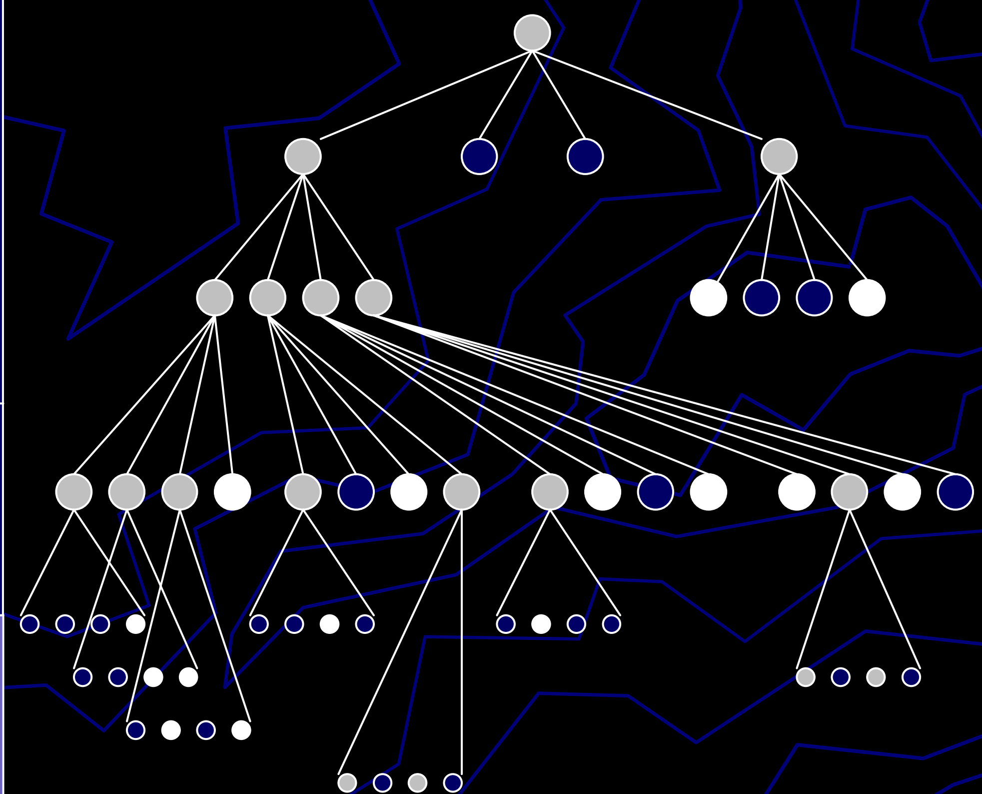
Each of these steps are examples of pruned quadtrees, or the space at different resolutions

Quadtree Example

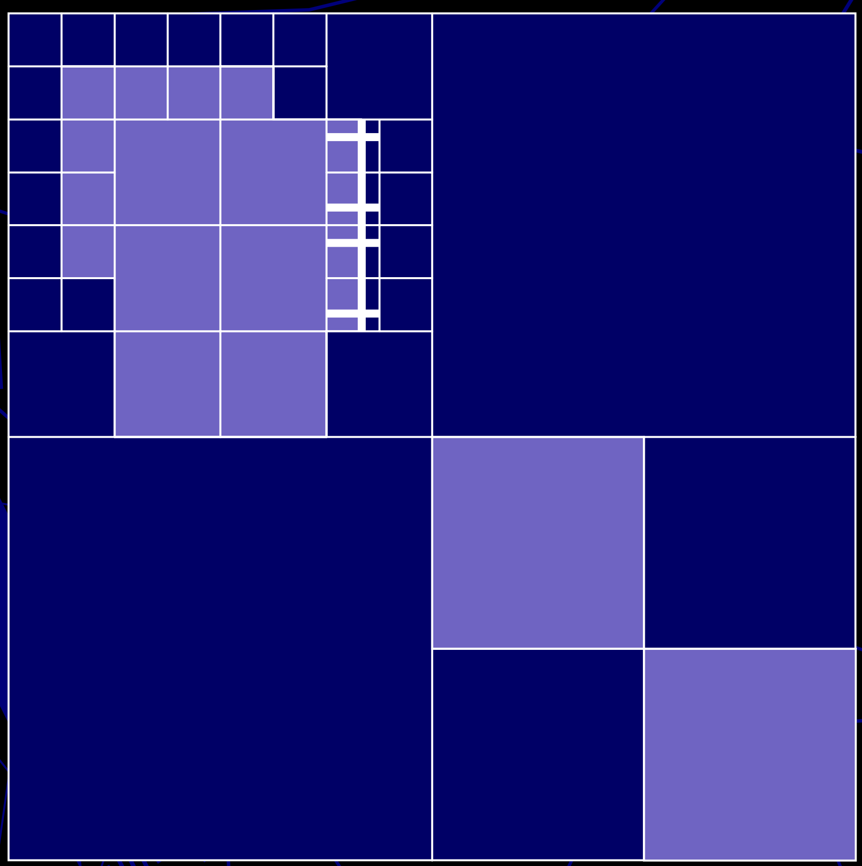
Space Representation

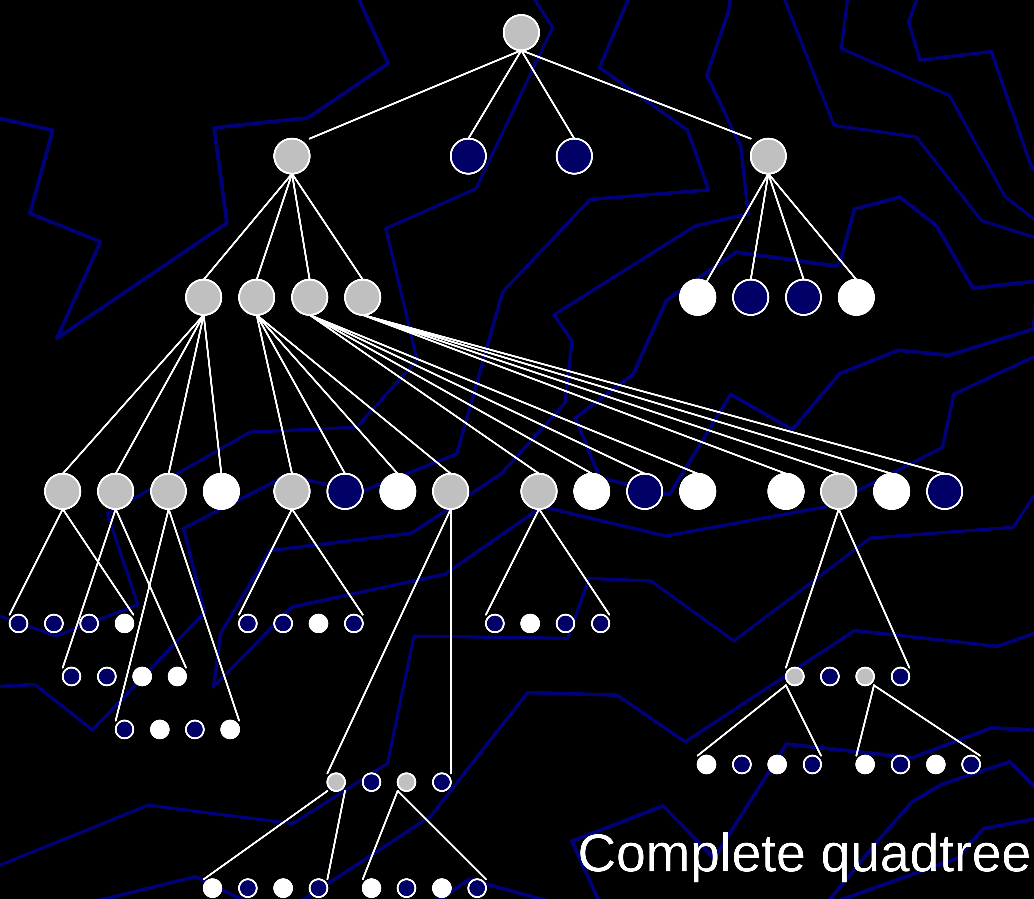


Equivalent quadtree



Quadtree Example





Quadtree-Based Path Planning

- Preprocessing
  - Step 1
    - Grow the obstacles by radius of the robot's cross section
    - Convert the result into a quadtree
  - Step 2
    - Compute a distance transform of the free nodes (from the center of the region represented by a node to the nearest obstacle)
- Given start and goal points
  - Determine the nodes S and G which contains these points
  - Compute the minimum cost path from S to G through free nodes using the A* graph search

Search

- Once you have your graph you need to search for the best path
- Several search methods can be used:
  - A* is the most popular (we will discuss this briefly on Monday)
  - Other options include random search, depth first search, breadth first search, etc.
- Good search techniques are important for a variety of reasons – you will learn more about them in 15-211 and other CS classes

A Bit of Trivia

- The concept of a Web spider was developed by **Dr. Fuzzy Mouldin**
- Implemented in 1994 on the Web
- Went into the creation of Lycos
- Lycos propelled CMU into the top 5 most successful schools
  - Commercialization proceeds
- Tangible evidence
  - Newell-Simon Hall

Dr. Michael L. (Fuzzy) Mauldin

End of slides!

See you tomorrow in lab!