LECTURE 8-2

Map Making

Quote of the Week

"Genius is 1 percent inspiration and 99 percent perspiration. As a result, genius is often a talented person who has simply done all of his homework."

Thomas Edison

ANNOUNCEMENTS

- Quiz 15 on Sec. 11.1 – 11.4, Lec. 8-1 on *Thursday, 5/6/10*
- Lab 8 memo and code is due on Angel by midnight on *Thursday, 5/6/10*

OBJECTIVES

Upon completion of this lecture the student should be able to:

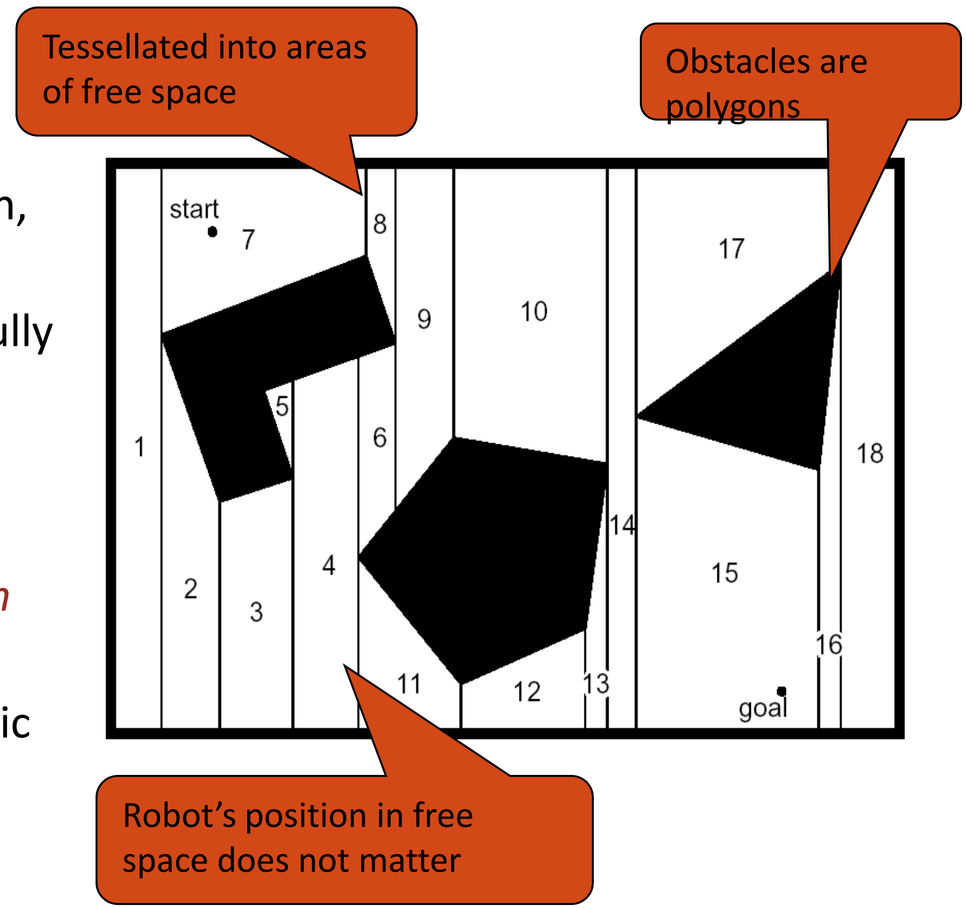
- Describe decomposition strategies for map making
- Describe the two types of formal exploration strategies

DECOMPOSITION STRATEGIES

- One method of *simplification* is to approximate the real world environment lines as a set of infinite lines
- A more dramatic form of simplification is *abstraction*
  - *A general decomposition and selection of environmental features*
- The immediate disadvantage is the loss of fidelity between the map and the real world
- It may be useful if planned carefully to capture relevant, useful features of the world while discarding all other features

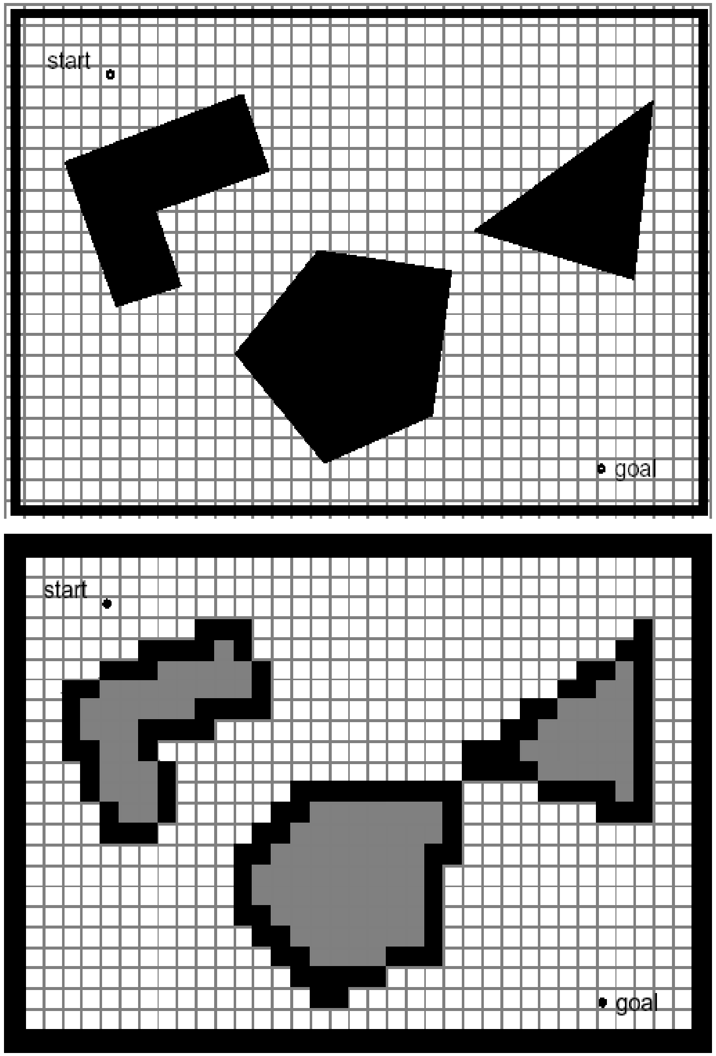
TESSELLATION DECOMPOSITION STRATEGY

- Advantage:
  - the map representation is minimized
  - With hierarchical decomposition, reasoning and planning may be computationally superior to a fully detailed world model
  - A standard, lossless form of *opportunistic decomposition* is termed *exact cell decomposition* selects boundaries between discrete cells based on geometric criticality



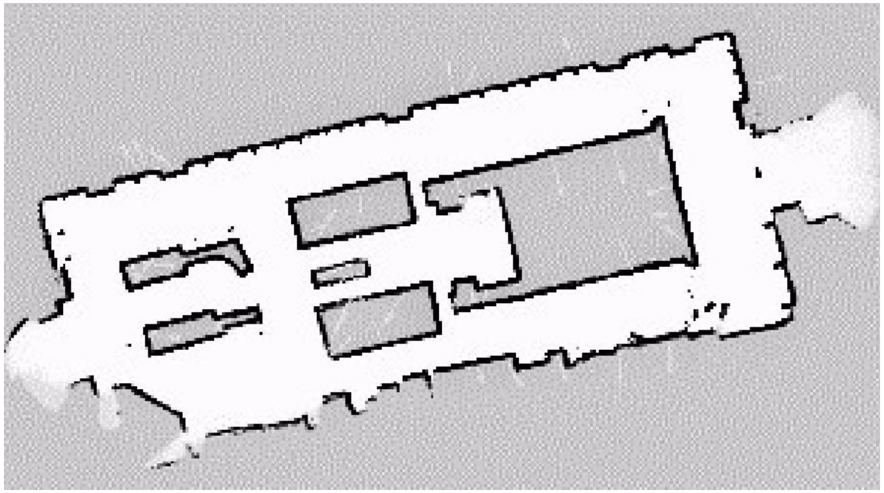
FIXED CELL DECOMPOSITION

- In *fixed cell decomposition*, the world is tessellated into a discrete approximation of the continuous map
- The key disadvantage is the inexact nature
- Narrow passages are lost in this transformation



OCCUPANCY GRID MAP REPRESENTATION

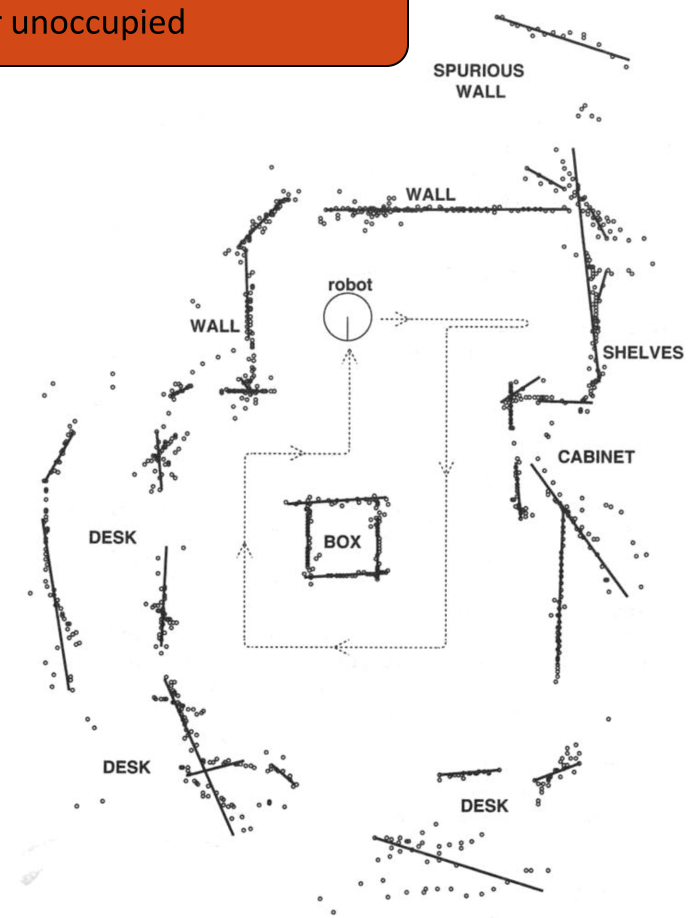
- A counter is used to determine how many times a cell is hit by a ranging sensor
- As the counter is incremented, the cell is deemed an obstacle
- The darkness of the cell is proportional to the value of the counter



OCCUPANCY GRID MAP REPRESENTATION DISADVANTAGES

- The size of the map in robot memory grows with the environment size
- Small cell sizes make the size of the memory untenable
- Not compatible with the closed-world assumption which enables large, sparse environments to have small memory requirements
- Imposes a geometric grid on the world a priori, regardless of environment details

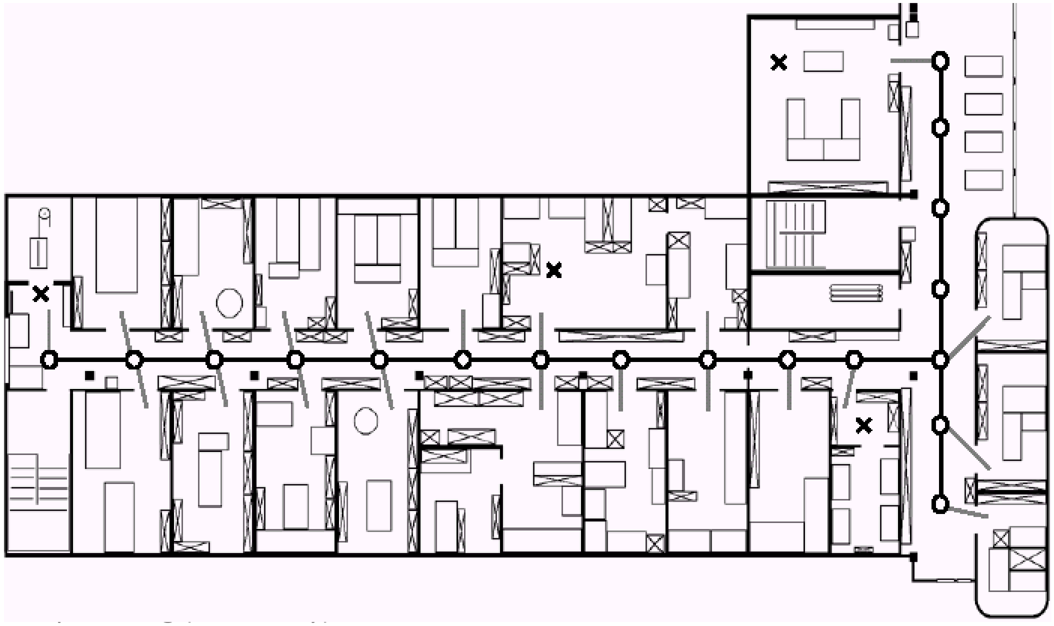
Each cell is either occupied or unoccupied



Made with sonar data

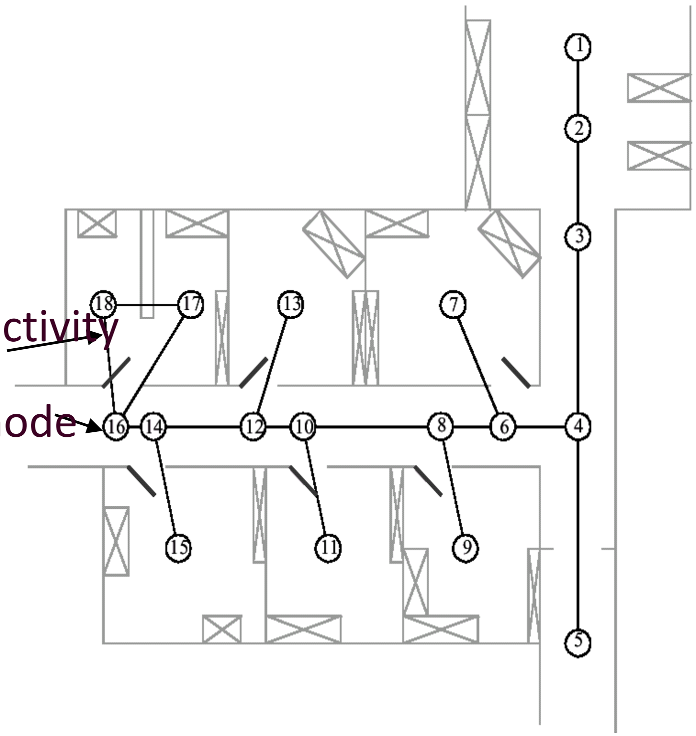
TOPOLOGICAL DECOMPOSITION

- Avoids direct measurement of geometric environmental qualities
- Concentrates on characteristics that are most relevant to robot localization



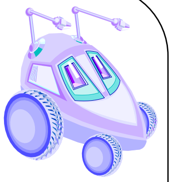
TOPOLOGICAL DECOMPOSITION

- Topological representations is a graph that specifies
  - *Nodes*
    - Areas in the world
  - *Connectivity arcs*
    - Denotes adjacent pairs of nodes
  - Adjacency is at the heart of the topological approach
  - Nodes are not of a fixed size or specifications of free space
  - Nodes document an area based on ay sensor discriminant



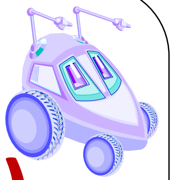
MODEL COMPLEXITY

- Some models are very elaborate
  - They take a long time to construct
  - These are kept around for a long time throughout the lifetime of the robot
  - E.g.: a detailed metric map
- Other models are simple
  - Can be quickly constructed
  - In general they are transient and can be discarded after use
  - e.g.: information related to the immediate goals of the robot (avoiding an obstacle, opening of a door, etc.)



MODELS AND COMPUTATION

- Using models require significant amount of computation
- ***Construction***
  - the more complex the model, the more computation is needed to construct the model
- ***Maintenance***
  - models need to be updated and kept up-to-date, or they become useless
- ***Use of representations***
  - complexity directly affects the type and amount of computation required for using the model
- Different architectures have different ways of handling representations



AUTONOMOUS MAP BUILDING (SLAM)

A robot that localizes successfully has the right sensors for detecting the environment and the robot ought to build its own map

- starting from an arbitrary initial point,
- a mobile robot should be able to autonomously explore the environment with its sensors,
- gain knowledge about it,
- interpret the scene,
- build an appropriate map
- and localize itself relative to this map

SIMULTANEOUS LOCALIZATION AND MAPPING (SLAM)

- When the robot does not have a map and does not know where it is and chooses to build a map as it goes along in order to localize this is called *simultaneous localization and mapping (SLAM)*
- SLAM is also referred to as concurrent mapping and localization (CML)
- *Simultaneous Localization and Mapping (SLAM)* is one of the most difficult problems specific to mobile robot systems
- SLAM is one of the most difficult tasks in robotics because it is based upon the interaction between the position updates it uses to localize and the mapping actions

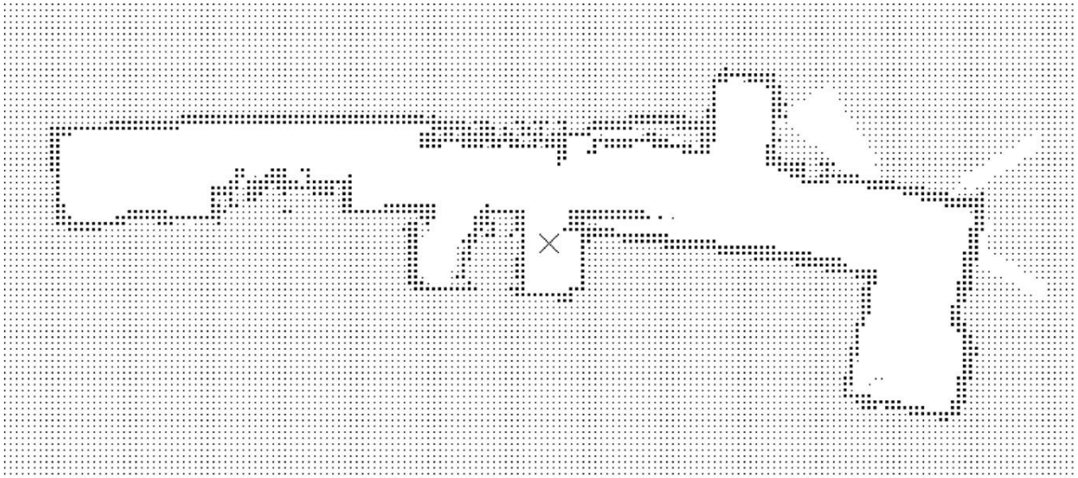
DATA ASSOCIATION PROBLEM

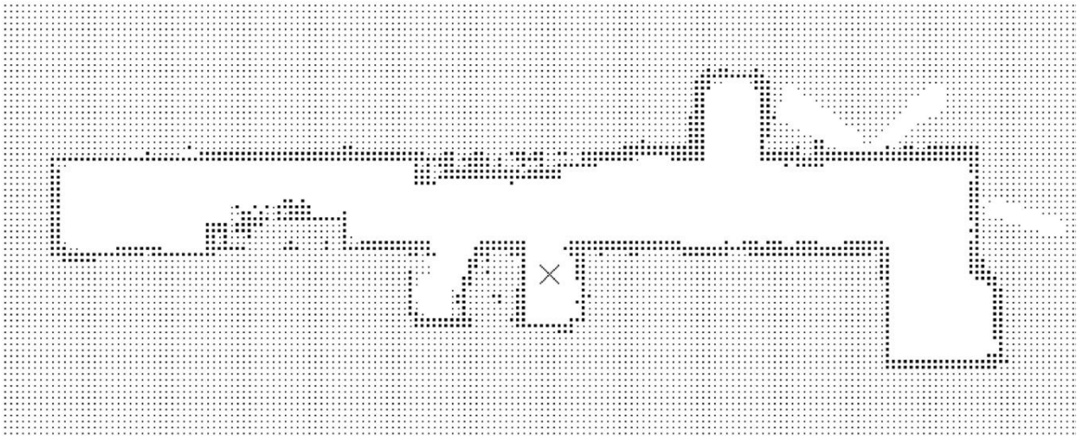
- SLAM is a difficult problem because it involves having the robot perform two ongoing and related parallel processes
- There is confusion among multiple places that look similar and therefore ambiguous
- This is the ***data association problem*** of uniquely associating the sensed data with absolute ground truth
- For topological maps
  - the robot has to contend with uniquely identifying landmarks
- For metric maps
  - The robot has to contend with odometry error and other sensor measurements
- It would be nicer if the robot had the map in order to localize or if the robot can localize when building a map

SLAM DILEMMAS

- If the robot updates its position based on an observation of an imprecisely known feature, the results position estimate becomes correlated with the feature location estimate
- The map becomes correlated with a position estimation if an observation taken from an imprecisely known position is used to update or add a feature to the map
- The complete and optimal solution has to consider correlations between *position* and *feature location estimation*
- Cross-correlated maps are called *stochastic maps*

WHEN MAPPING AND LOCALIZATION ARE NOT SIMULTANEOUS





COVERAGE

- There are two coverage problems based upon whether the robot has a map or not
- If there is a map
  - The robot searches all navigable spaces until the goal point is found
  - This is a search problem and there are algorithms in computer science and AI that are rather slow to perform this (i.e. A*)
  - Some search algorithms use *heuristics*, rules of thumb that help guide and speed up a search

NO MAP FOR COVERAGE

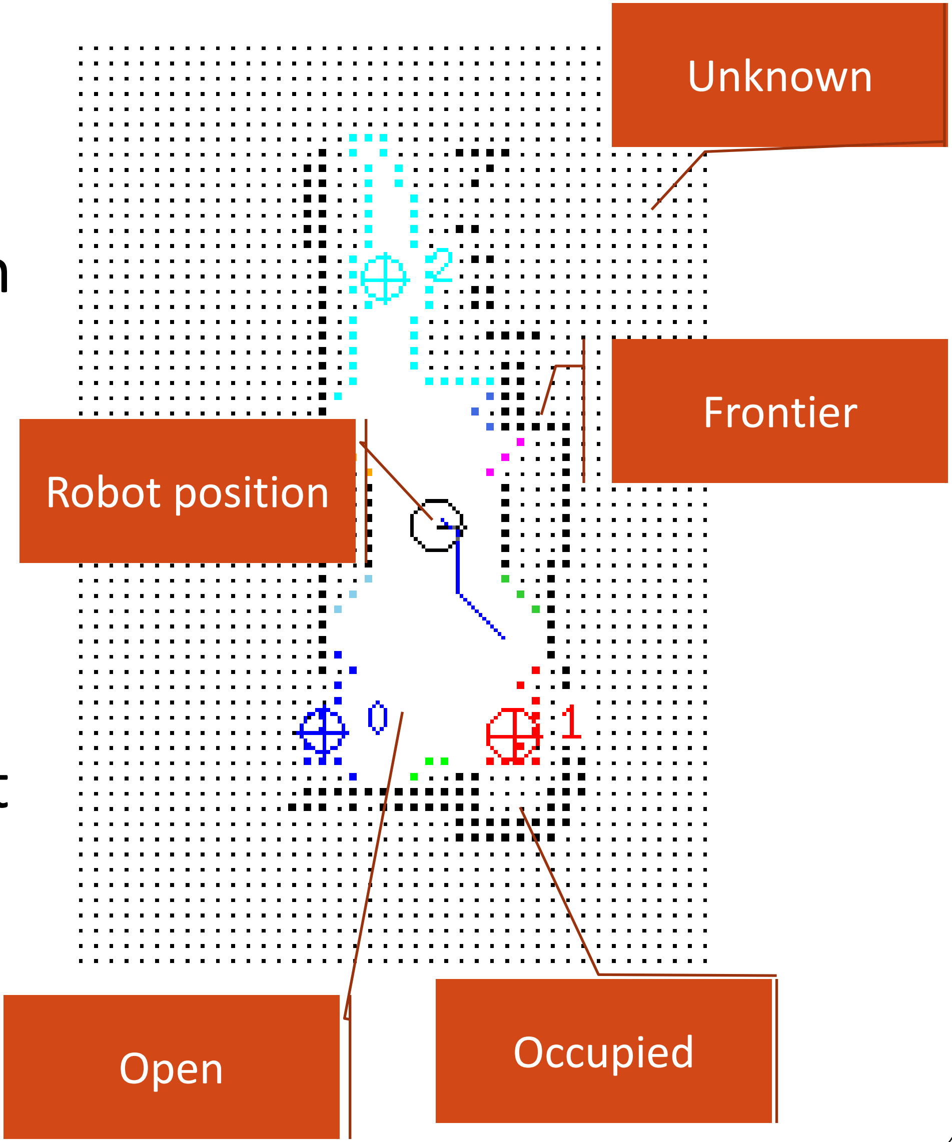
- When the robot does not have a map it has to move in a systematic fashion to find what it is looking for
- Mapping the environment first may be a better approach but that takes time
- One heuristic when the map is not known is to follow continuous boundaries or spiral out from a starting point
- The robot may also move randomly and given enough time and a closed environment it may cover the space

EXPLORATION

- Exploration attempts to answer the question *where haven't I been?*
- How do you cover an unknown environment efficiently?
- One method is random search or using a random potential field and after a long period of time, the area will be covered
- Also use an occupancy grid to explore unknown areas
- Two basic exploration methods are:
  - Frontier-based
  - Generalized Voronoi graph

FRONTIER-BASED EXPLORATION

- When a robot enters a new area, there is a boundary between each area that has been sensed and is open and the area that has not been sensed
- Boundaries are lines that form a ***frontier*** that should be explored



GENERALIZED VORONOI GRAPH METHODS

- The robot attempts to build a reduced GVG as it moves through the space to explore
- The robot attempts to maintain a path that places it equidistant from all objects it senses
- This path is the GVG edge
- The robot can then later follow the path using a behavior