COMP417
Introduction to Robotics and Intelligent Systems

Global Localization: MDL

Mobile Robotics Lab at McGill University

Global Localization

- Recall, incremental localization (state estimation) refines a starting guess.
- Global localization start *ab initio* (i.e without a good prior).
- Analogy with function minimization.
- Localization problems (as usual) presupposes a map and find a position.
- This algorithm is about action selection: probabilistic filtering can be used in parallel.

Mobile Robot Pose Estimation Revisited

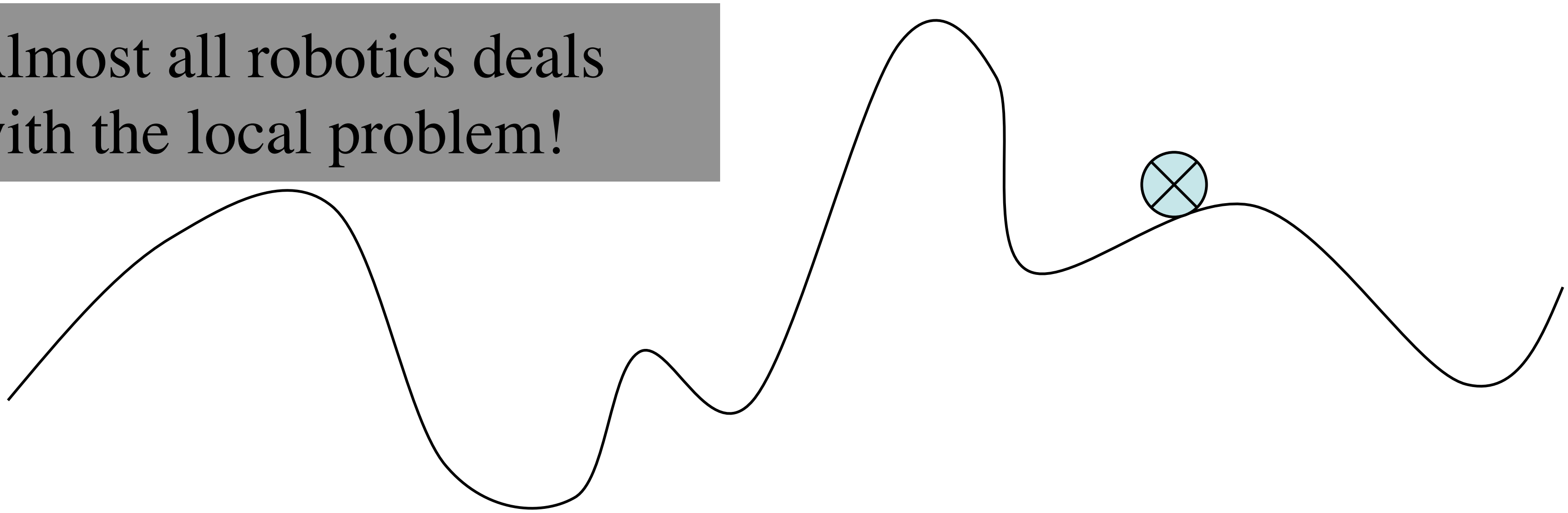
- Recall, to answer the question: where am I?

  directly analogous to global vs local function optimization

  A. **Globally**, assuming I could be "anywhere".

  B. **Locally**, assuming I know I am in a specified region but want an accurate position fix.

Almost all robotics deals with the local problem!



Overview

- **Introduction**
  - Minimum Distance Localization (MDL)
    - Localization with efficient travel (LET).
- **Part 1**
  - Complexity result, Deterministic Algorithm
- **Part 2: Randomized Algorithms**
  - Common Overlay Localization (COL) Algorithm
    - Useful Region Localization (URL) Algorithm
- **Experimental Evaluation**

Problem Statement

- Localize a robot with no prior position estimate.
  - a) How difficult is this problem, formally?
  - b) What is an *efficient* algorithm?
- "Global Localization" [Book Sec 8.4]
  - Also known more informally as the kidnapped robot problem.

How hard the the problem?

- In general, observations from a single position may be ambiguous.
- To uniquely localize, most move around and take more observations.
- Complexity measure: what is the minimum amount of travel needed to localize, in the worst case?

Formalism & Assumptions

- To consider the *feasibility* of the global problem, we start with an idealized model:
  1. 2D world
  2. Point robot
  3. Assume the environment is a polygon P without obstacles
  4. Assume our robot has a perfect map
  5. Assume our robot has a perfect range sensor with infinite maximum range.
  6. Perfect compass (known orientation)
- The region seen by the robot at any time is a **visibility polygon**
  - The key observation is the set of vertices visible at any time.
- We can divide the environment into regions within which we seem the same set of vertcies: **visibility cell decomposition.**

Example

- Intuition: you wake up in a hotel room with memory of getting there, what room is it?
  - Check what end of the hall it is.
  - Check what floor it is.

- Robot example:
  - System gets rebooted.
  - Range data from current location is ambiguous.
  - Where are we?
  - More around to reduce pose uncertainty.
    - Saw some examples of this already in this session.

Global Problem (now our focus)

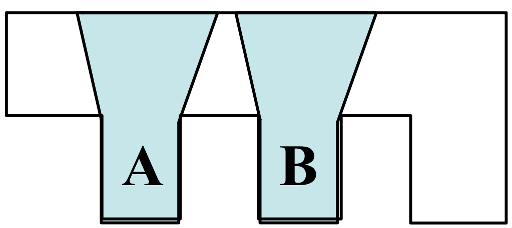
What if* no *approximate pose estimate is available? (i.e. a uniform prior)

- Find the best match of our observations over the entire known map.

- Problem: Observations from a single viewpoint may be ambiguous: two offices may look alike.

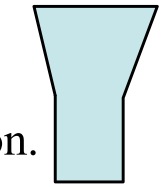
  This is a risk even with perfect noise-free maps & sensors.

- We may need to <u>combine multiple observations</u> to determine our position uniquely.

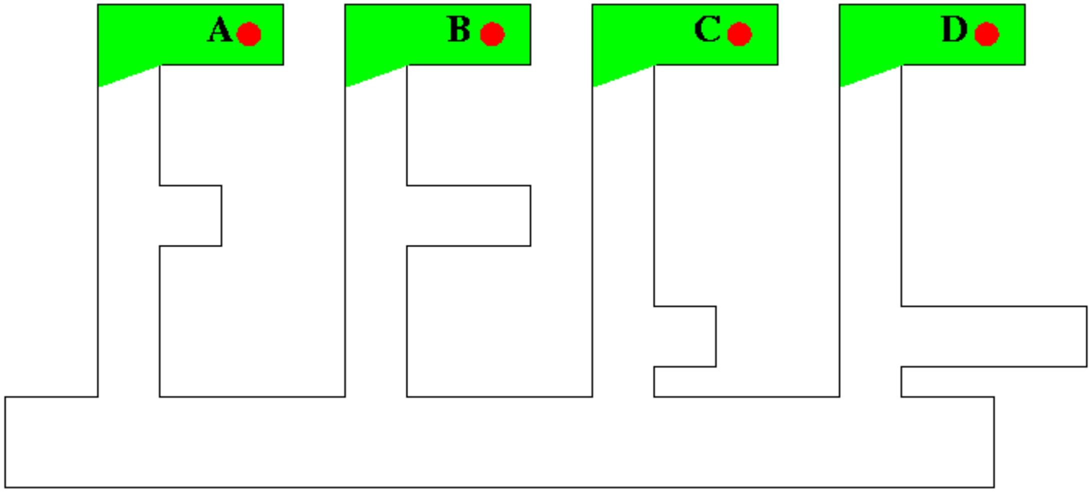


Positions **A** and **B** are indistinguishable without moving into the "hallway" above.

Common visibility polygon.



Minimum Distance Localization



- Mobi [illegible]
  locati [illegible] nment *P*.
- Robo [illegible] its
  environment.

- **Objective:**

Problem statement

- Determine where we are.
- But the problem is not really what are my precise coordinates, but

  Which of several different hypothetical locations am in in?
- If we can resolve ambiguity, the precise position is trivial given our (idealized) assumptions.
- Localization in thus case is thus combinatorial, not metric.

Assumptions

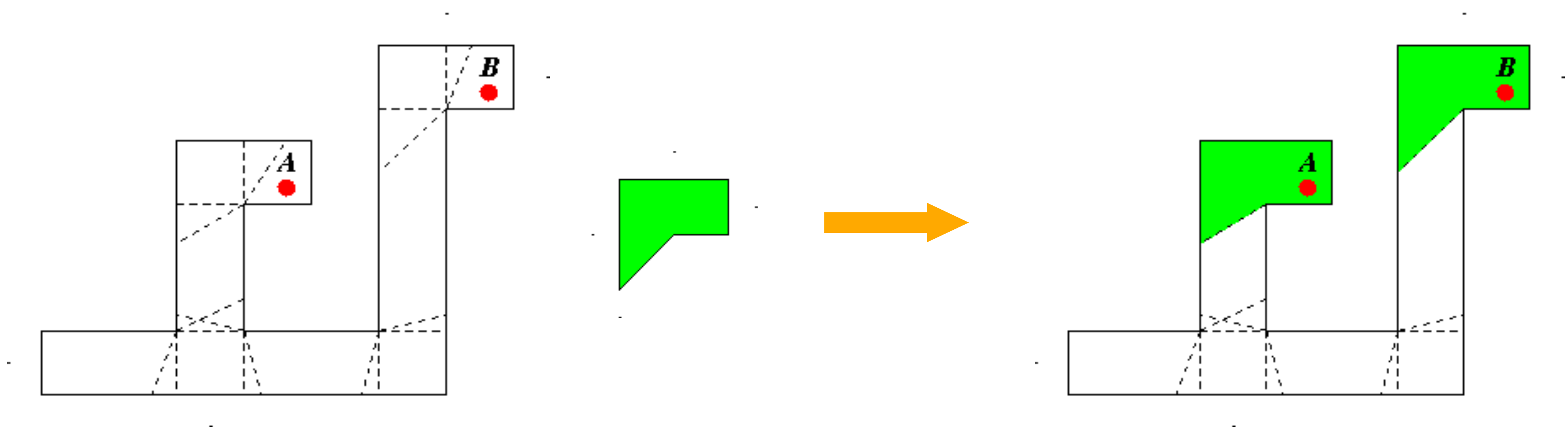
- Robot’s equipped with a finite-distance range sensor.
  - Note: prior theoretical results assumed unbounded range.
  - In reality, with a noisy (i.e. real) sensor, the problem can only become <u>more difficult.</u>
- Sensed data consists of an ordering of vertices and edges seen by the robot (visibility polygon).
- Robot is assumed to be a point robot moving in a static *2D*, obstacle-free environment.
  - Obstacles have not effect on the result, but make it more tedious.
- Robot is able to determine its orientation.

Basic Approach

- Determine set of *hypothetical locations* that match robot's initial observations.
- Compute next sensing location from where distinguishing landmarks may be seen.
- Travel to this location and make observations that disambiguate the hypotheses.
- Eliminate incorrect hypotheses.

Theoretical Results

- **MDL** expressed as optimal localizing decision tree is NP-hard.
  - (Even with unbounded range.)
- **Precomputation:**
  - *Hypothesis Generation:* Computes the set of hypothetical locations matching the robot's initial observations.

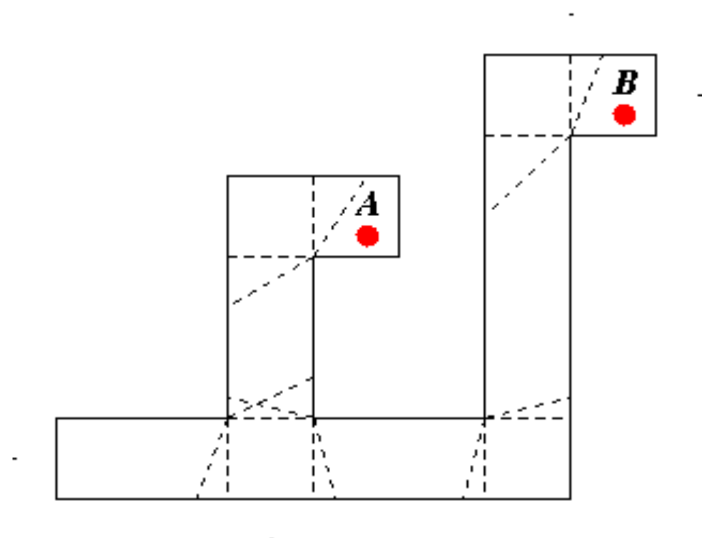


Competitive Ratio

- Ratio of solution quality to the optimal.
- Normally used for hard problems.

Results

– *Hypothesis Elimination:* Rules out incorrect hypotheses by directing the robot to travel to the nearest distinguishing visibility cell.



– Competitive ratio: $(k\text{-}1)d$, k is number of hypotheses, d is length of an optimal verification tour.

– Complexity: $O(n^6)$, n is number of vertices in P.

Hardness of MDL

- Even for a point robot in a polyhedral environment, and with perfect range sensor the global localization problem is NP-hard.
  - There is a reduction to **abstract decision tree**, a previously-established NP-hard problem.

- Can we simply use suitable heuristics to get good performance?

Approximating MDL

- A simple greedy algorithm:
  - visit the closest place that tells me something about where I am.
  - Next, visit the next closest place that tells me something.

  This seems intuitive and has actually been used in practice.

This approach can be shown to be *exponentially* worse that optimal.

Approximating MDL accurately

- Polynomial time strategies with provably good performance can be constructed [Dudek, Romanik, Whitesides].
  - For h alternative hypothesis about where we are, we can be sure the path length is never more than (h-1)optimal.
  - Cost of computing the solution is polynomial.

Oops

Did not to the rest of this slide set in class

While these results are satisfying, they have 2 key shortcomings.

- The computational cost of computing a solution is high.
- The observation that need to be made may not be practical.
  - It may be necessary to visit very small cells that cannot be easily reached given realistic odometry errors.

Intractability of the Global Problem

Resolving ambiguity?

- We compute a decision tree that will allow us to optimally decide what to do if we find ourself in an ambiguous location.
  - The height of this decision tree gives the length of the longest information-gathering path, and we seek to minimize this.
- We will refer to this problem of determining pose using a <u>minimum length path</u> through a known map as the *global localization problem*.