Chapter 8: Maps and related tasks

Harm Dermois Joris Stork

January 23, 2011

Robots and maps

Sensorial maps

- Image based mapping
- Spacial occupancy representations
- Geometric maps

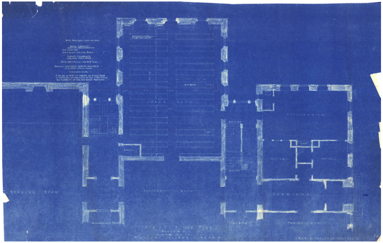
Topological Maps

Multiple Robots

Maps for robots

Human maps:

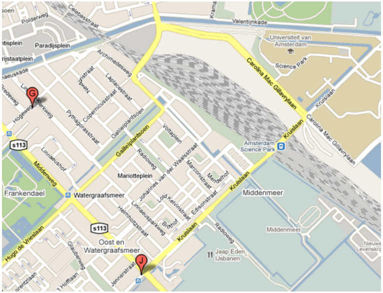
- Often unavailable;
- Often incomplete:
  - human relevant data;
  - robot relevant data;



Maps for robots

Human maps:

- Often unavailable;
- Often incomplete:
  - human relevant data;
  - robot relevant data;
- Wrong level(s) of abstraction: human oriented;



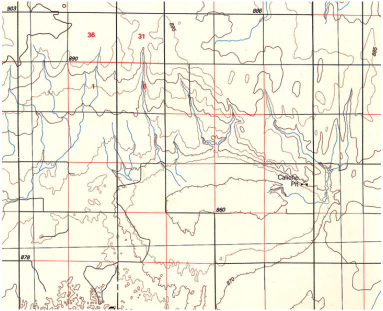
Maps by robots

- Building a robot friendly map is very difficult and tedious.
- Robots are good candidates to build maps with and for their own sensory suite;

Conclusion: Design robots to autonomously construct, update and validate maps destined for robot use.

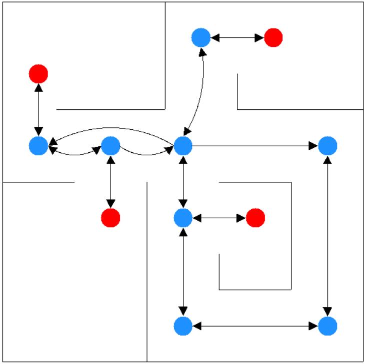
Map paradigms: metric vs. topological

- Metric
  - Sensorial
  - Geometric



Map paradigms: metric vs. topological

- Metric
  - Sensorial
  - Geometric
- Topological
  - Local relational
  - Topological
  - Semantic



Direction of map hierarchy

- Giralt et al.: metric to topological.
- Kuipers and Levitt : topological to metric. Low level topological landmarks as starting point.

Types of data

- (Derived) spacial occupancy;
- (Direct) sensor measurements in relation to position. e.g. olfaction.

Sensorial maps

- Represent sensor measurements against odometry;
- Collection of measurements: $[I_i(x_i, y_i, \theta_i)]$

Image based mapping

The challenge:

- how to sample the set of possible measurements, $\{I_i\}$;
- how to turn the samples into a continuous I.

Harm Dermois, Joris Stork

Li: street panoramas

Li et al.: robots builds graph representing street network:

- edges = streets;
- nodes = intersections;

Robot collects by:

- moving in a closed loop, always turning left;
- recording panoramas of left and right sides of streets;
- concluding a loop by identifying previously recorded street side;

Bourque: robot sightseeing

Bourque et al.: robots builds graph nodes corresponding to panorama shots, in a less constrained environment, by:

- choosing sample (panorama) points based on models of human attention;
- using "alpha backtracking" to make trade-off between distance to next sample point and optimality of next sample point;

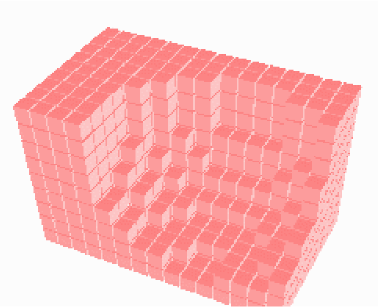
Spacial occupancy grid

Pioneered by Elfes and Moravec

- Grid of pixels

Spacial occupancy grid

- Grid of pixels
- Volume of voxels



Spacial occupancy: data represented

- Fill pixels / voxels with degree of occupancy data.
- More refined: fill pixels / voxels with probability of occupancy data.

Spacial occupancy: probabilistic approach

- Example of a laser sensor: probability of an actual distance z for a given laser reading r computed using Bayes' theorem:

$$P(z|r) = \frac{P(z)P(r|z)}{P(r)}$$

Spacial occupancy: probabilistic approach

- Example of a laser sensor: probability of an actual distance z for a given laser reading r computed using Bayes' theorem:

$$P(z|r) = \frac{P(z)P(r|z)}{P(r)}$$

- Generalised:

$$P(W|R) = \frac{P(W)P(R|W)}{P(R)}$$

with $P(R_i) = \sum_j P(R_i|W_j)P(W_j)$

Spacial occupancy: probabilistic approach

The result: *maximum a posteriori* (MAP). World model that most reasonably estimates environment according to Bayesian approach.

Spacial occupancy: probabilistic approach

The result: *maximum a posteriori* (MAP). World model that most reasonably estimates environment according to Bayesian approach.
Considerations:

- very general: no assumed model, deals with multiple sensors;
- requires accurate probabilistic model of the sensors;
- requires a lot of memory for the occupancy map;
- measurement locations/times discarded: geometric accuracy reduced;
- important to avoid accumulated positional errors - e.g. by iteratively recomputing position;
- needs an exploration policy: e.g. random or towards “unkown’ ’areas.

Spacial occupancy: Markov models

Markov localisation: estimating robot's location based on sensor data by maintaining probability density grid for the robot's environment, with each cell representing a possible robot pose.

Geometric maps

Accurate, with two assumptions:

- sensor data is suitable
- environment is suitable

Geometric maps: exploration

Challenge is exploration. Includes searching for:

- a goal position;
- route with specific properties;
- “covering” a space;
- occupancy.

Geometric maps: reach goal

Papadimitriou and Yannakakis's bug-like algorithm for reaching known goal from known origin in unkown environment with obstacles:

- move "towards" line connecting origin and goal;
- if not possible, move in arbitrary direction;

Useful in certain simple types of environment, notably where obstacles are:

- rectilinear;
- nonintersecting;
- aligned with world coordinates.

In more general environments no bound is possible.

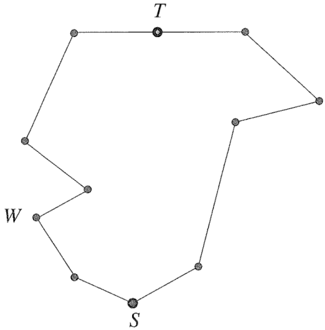
Geometric maps: geometric representations

Chosen geometric representation influences applicable algorithms. Important representation is that of “street polygons”.

Geometric maps: Street polygons

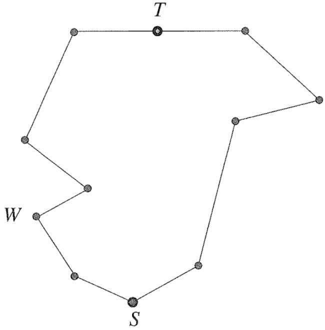
Polygon such that:

- there is a start vertex S and end vertex T
- vertices and lines categorised as “left” or “right” with respect to line segment from S to T;
- every vertex on either side is visible to some vertex on the other



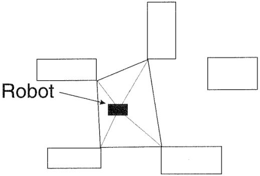
Geometric maps: Street polygons

- Kleinberg developed an algorithm that finds the "*optimal* L_1" shortest path from S to T.
- Datta, Icking and Klein developed an algorithm applicable to a generalisation of the street, the "G-street".

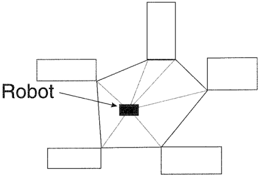


Geometric maps: Street polygons

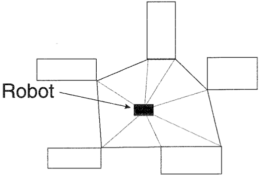
Exploring for occupancy.



(a)



(b)

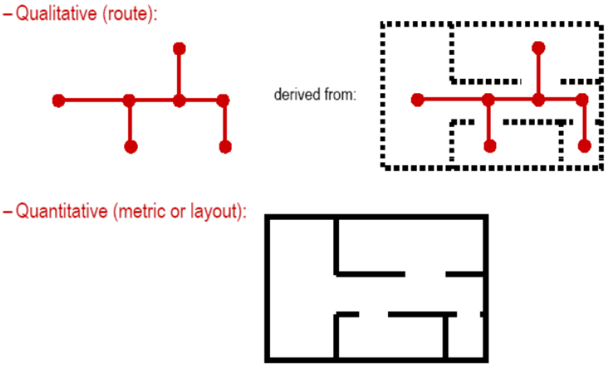


(c)

Topolgical Maps

Topological maps: describes the environment as a graph that connectcs specific locations in the world and represents them as nodes(vertices).

- Because metric representations cost too much memory to maintain in the long run.
- Easy to understand for humans.



The Graph

Landmarks and Edges

- The nodes on the graphs are landmarks or features of the environment.
- The edges are paths between the different nodes.
- Landmarks can be artificial or natural.(junctions, signs)
- Landmarks can look the same so you need to make sure you dont use two or more nodes to represent the same landmark.
- The graph can be extended by enumerating the edges incident to the node entered. Edge you traveled along is 0 and enumerate clockwise. This enumeration is local cause it depends on the edge the robot moved over.
- Landmarks need to be unique to good landmarks.

Marker Based Exploration

Marker Based Exploration: can be used when no prior information about the environment available and there aren't enough unique landmarks.

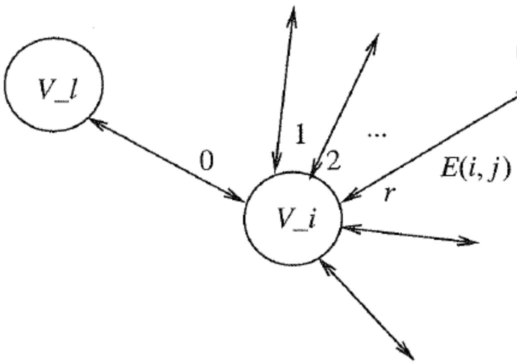
- The robot needs to have something to mark where it has already been.(spray paint, bread crumbs)
- Here we choose marks which it can pick up, drop and recognize.
- Using marks to explore has an $O(N^3)$. All nodes are on one straight line.

Marker Based Exploration

- Iteratively builds up the known graph by traveling along the incident edges of a node.
- v_i is the node where the robot is currently at. v_j is the node where the robot is moving to. $E_{i,j}$ is the edge between the two nodes.
- In the transition function r stands for move along the given edge.
- Transition function need to follow these properties. If $(v_i, E_{i,j}, r) = v_j$ and $(v_j, E_{i,j}, s) = v_k$, then $v_j, E_{i,j}, -s) = v_i$
  Moves are invertible and can be retraced.
- $t \neq -s$ then $v_j, E_{i,j}, -s) = v_i$ and $(v_j, E_{i,k}, s) = v_j$ are not valid. To avoid redundant and degenerate paths.
- Subgraph S for explored edges and Nodes. U is the for unexplored sub graph.

Marker Based Exploration

Edge Ordering: A correct graph.



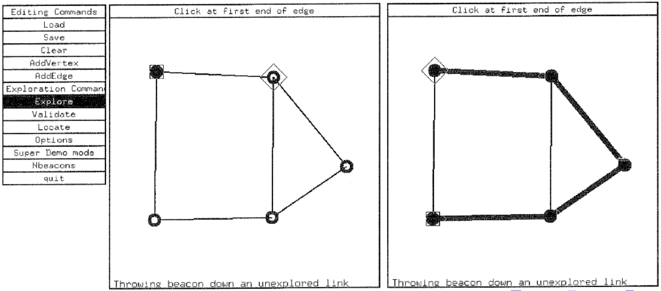
Marker Based Exploration

Operations of the robot

- Move along edge r.
- Each marker can be in 3 different states[pickup,putdown,null]
- At each vertex the robot can see two things [present, not-present]
- Robot can determine the relative positions of the edges by enumerating the edges.
- Entering the same vertex from a different edge gives two different orderings. The robot needs to make a global ordering.

Marker Based Exploration Example

- First validate all explored nodes. So all nodes in graph S. Make sure there aren't any doubles by looking for markers.
- Explore new nodes. If there is no marker found at a certain node v add it the subgraph S and add the edge which was taken aswell.
- Enumerate all edges incident to the new node and add them to U.
- Do this till subgraph U is empty.



Multiple Robots

Why would you use multiple Robots

- **Improved Robustness**: A multirobot can, in principle, keep functioning even if one indiviudual robots dail completely.
- **Improved effiency**: It is possible for a group of robots to accomplish a search or exploration task faster than an equivalent single robot.
- **Alternative Algorithms**: For some tasks, the availability of multiple robots allows feadible or guaranteed algorithms to be implemented when no such algorithm is available for a single robot system.

Multiple Robots in Practice

Problems

- **Where are teh other robots?**: Rendezvous with other robots
- **Partitioning**: Finding a good way to distribute the work amongst the robots.
- **Multi-robot planning**: Prevent the trajectories of the robots to collide.
- **Merging the data from the indivual team members**: Need to be close proximity and Sensor fusion problems.

Rendezvous

Rendezvous: Is a having two or more robots meet at an appointed place and time.

- A rendezvous is needed for robots that can only communicate in close proximities, but may also be needed to exchange objects between robots.
- When Multiple robots try to complete a task collaborotavely without prior knowledge. They need to to exchange information while they are still working at the task at hand.
- If they dont meet they cannot benefit from what others have already learned.

Too many rendezvous

Problems:Robots mustn't devote too much energy to rendezvous to stay efficient.

- The extent to which the two robots agree on their perceptions of the environment.
- The degree of synchronization of the robots can attain expressed as the likelihood that an appointed rendezvous at a common location will fail owning to a failure to arrive at the same time
- The extent of the commonality between the region of space the robots have explored.(Can't share if the parts are completely diffent)

Map Fusion

Map Fusion: is needed to make the collaborative efforts worthwhile when the problem needs a long term map.

- Complexity of the map-merging depends on the, Odemetry error, the fidelity of the sensing and the richness of the evironment.
- Fusing maps is mostly done by cross correlation. This depends on the fact that the individual maps overlap "sufficiently'.
- Done by rotation and translating the given maps to minimize the difference between them.

Exploration with Multiple robots

The basic idea of the algorithm: Given Robots are only allowed to communicate when they are in the same node.

- Split all the work between all the robots and have them explore their own part of the graph.
- Plan rendezvous, to harmonize the information they got till then and make a single consistent representation of the environment they are in.
- Redevide the work and repeat this till everything is known.

links

http://www.csupomona.edu/~ftang/courses/CS499/notes/navigation3.pdf