Robot Localization Simulator

Prateek Garg, Kumar Ashwani, Apurv Verma

April 7, 2012

Acknowledgement

We would like to acknowledege Assistant Professor Apurva Mudgal for helping us understand the difficult Robot Localization Algorithm. His supervision made it appear all very easy. We are also thankful to the ever enthusiastic CGAL community for helping us with the various CGAL issues.

Introduction

A robot is placed at an unknown point inside a simple polygon P. The robot has a map of P and can compute visibility polygon from its current location. The robot must determine its correct location inside the polygon P at a minimum cost of travel distance.

Robot Localization Algorithm I

Input:
Map polygon P, the visibility polygon V.
Output:
The robot localizes to its actual position $h \in H$

1: Compute the set of hypotheses H.
2: **while** $|H| > 1$ **do**
3: Compute the majority-rule map P_{maj}
4: Compute the polygons G_{ij} for each pair of hypotheses, h_i and h_j
5: Compute the majority rule map K_i of G_{ij}'s
6: Find the edges on the boundary of K_i which are not on the boundary of P_{maj}
7: Draw grids and compute the set of coordinates Q_H on these edges.
8: Make instance $I_{P,H}$ of $\frac{1}{2}$ -Group Steiner Problem

Robot Localization Algorithm II

9: Solve $I_{P,H}$ to compute a half computing path $C \subset P_{maj}$
10: Half-Localize by tracing C and making observations at coordinates Q_H
11: Move back to the starting location.
12: **end while**

Geometrical Algorithms

Visibility polygon is an indispensable component in the hypothesis generation step of the algorithm. Since CGAL had no inbuilt support for computing visibility polygons we implemented the following two routines for our purposes.

- Visibility Polygon of a point inside a polygon
- Visibility Polygon of an edge of the polygon.

Visibility Polygon of a Point Inside a Polygon I

Definition

Visibility Polygon of Point: p is the bounded polygonal region of all points of the polygon visible from p.

Algorithm

1. Collect all the vertices of the polygon which are visible from the point P.
2. Iterate over the list of visible vertices and for each reflex vertex, compute the spurious vertex introduced in the visibility polygon.
3. Finally sort all the vertices in an order so that they form a simple polygon.

Examples I

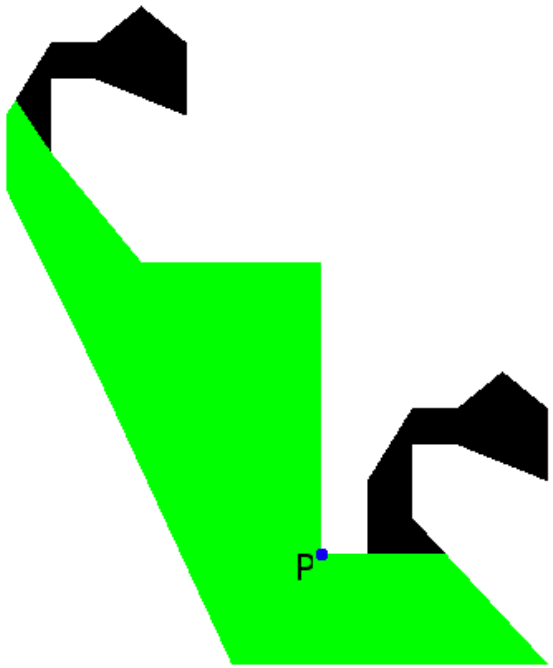

Figure: Visibility Polygon of Point

Examples II

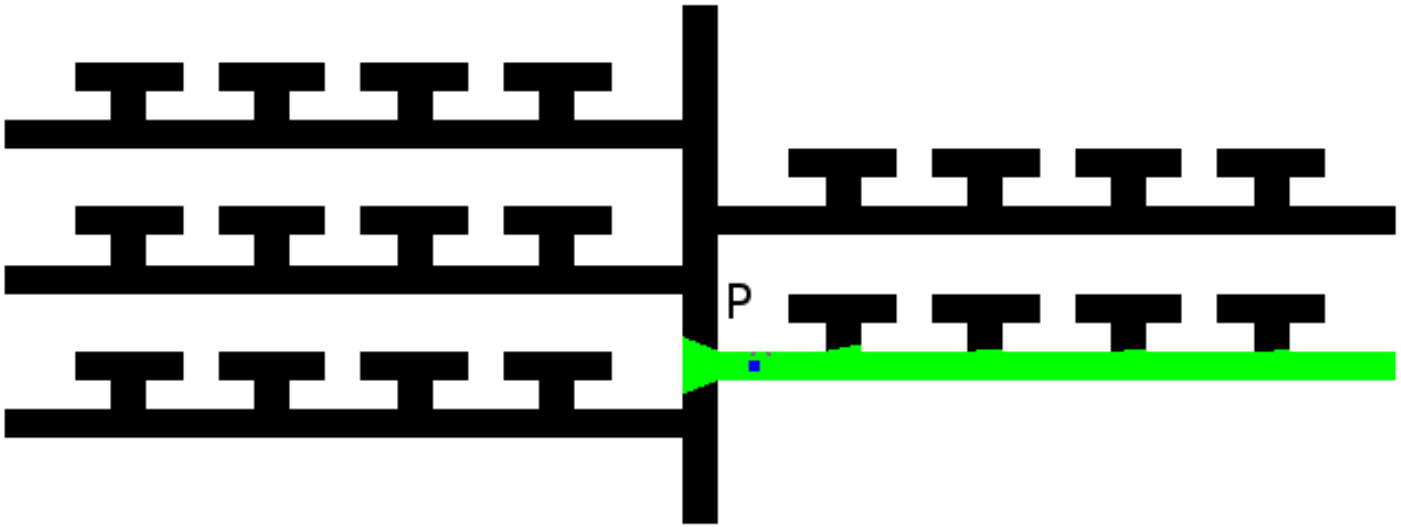

Figure: Visibility Polygon of Point

Visibility Polygon of an edge of the polygon

Definition

Visibility Polygon of Edge: e is the bounded polygonal region of all points of the polygon visible from any point on the edge e.

The algorithm for the visibility polygon of an edge has been taken from [**?**].

Visibility Polygon of an edge of the polygon

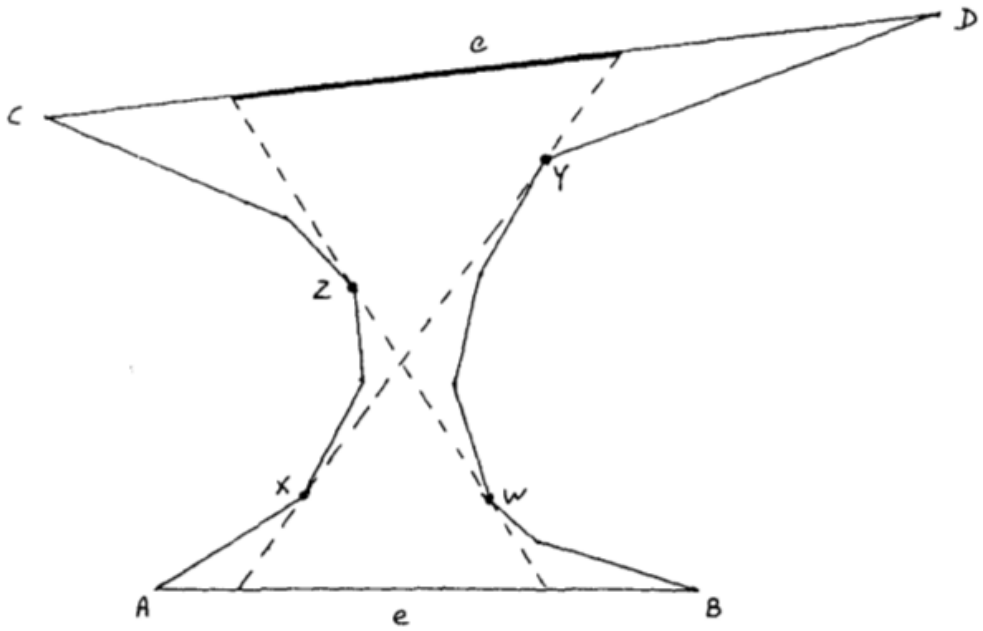

Figure: Visibility Polygon of Edge, Illustration taken from:[**?**]

Algorithm I

1. Compute the shortest path P_{AC}, from A to C and the shortest path P_{BD}, from B to D. Call this pair 1.
2. Similarly compute the shortest path P_{AD}, from A to D and the shortest path P_{BC}, from B to C. Call this pair 2.
3. Find out which of these pairs is outward convex. An outward convex pair implies an hourglass shape is formed by the two paths.
4. If none of the pairs is outward convex this means that no portion of edge CD is visible from any point on edge AB and we can completely ignore such an edge.
5. If one of the pairs is outward convex then without loss of generality, let pair 1 be the outward convex pair. Now compute the shortest paths P_{AD} and P_{BC}.

Algorithm II

6. Let X be the point where path P_{AD} and P_{AC} split and let W be the point where path P_{BD} and P_{BC} split. Let Y be the next point on the path P_{AD} and Z be the next point on the path P_{BC}. Extending XY we get one extreme point of the portion of CD visible from AB. We repeat this on other side to get the other extreme point.

Shortest Path Calculation I

For the calculation of shortest path between any two vertices of the polygon the following property was exploited.

- The shortest path must turn only at vertices of the polygon.
- It is possible to move from one vertex to the another only if they are visible to each other.

Visibility Graph I

Definition

Visibility Graph The visibility graph of a polygon can be formed as follows. Draw a vertex corresponding to each vertex in the polygon. Draw an edge between two vertices if the line joining the corresponding vertices in the polygon lies completely inside the polygon.

Visibility Graph II

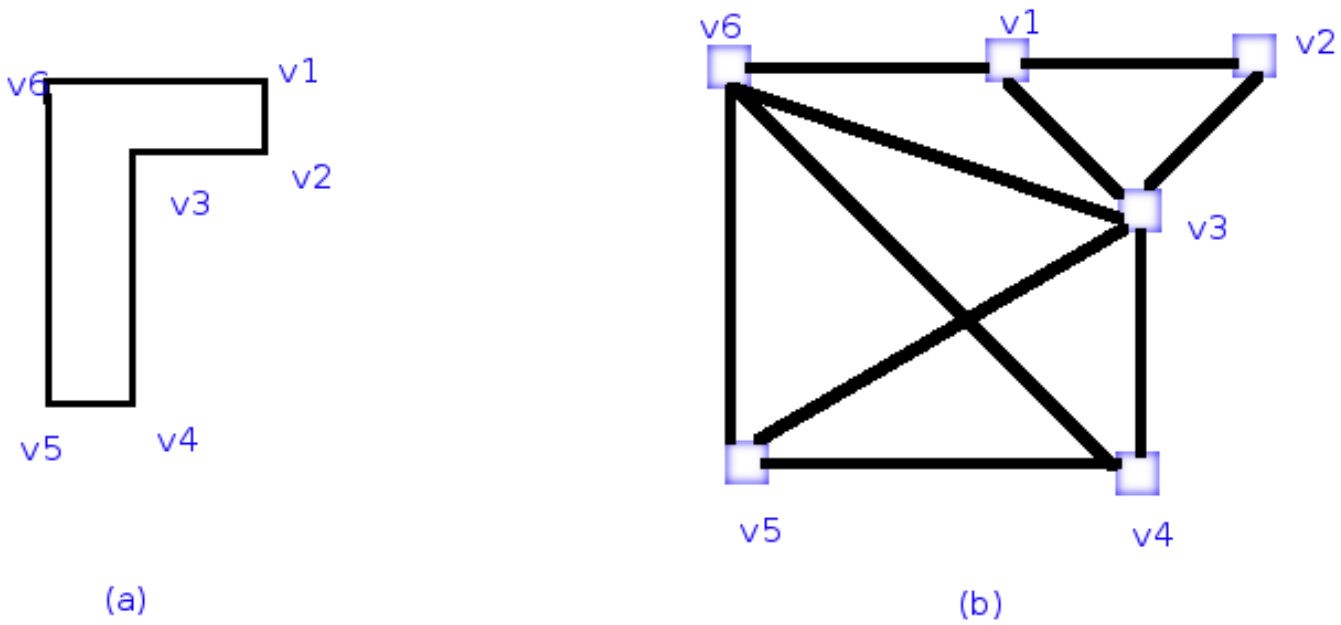

Figure: Visibility Graph

Examples I

Figure: Visibility Polygon of Edge

Examples I

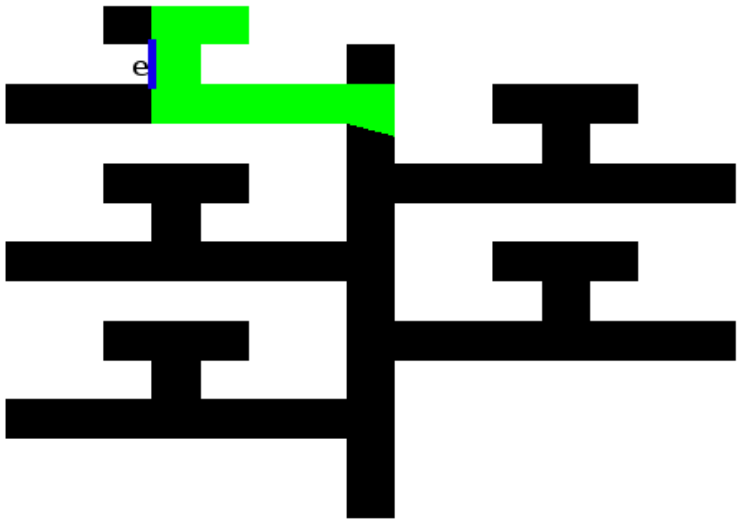

Figure: Visibility Polygon of Edge

Hypothesis Generation

Theorem

A point, P inside a simple polygon sees atleast one edge of the polygon completely.

Definition

Spurious Edge: In the visibility polygon of a point, an edge is called a spurious edge if it is obtained by extending the line joining the point P and a reflex vertex till it meets the polygon.

Theorem

The visibility polygon of a point P has atleast one edge which completely overlaps with an edge of the original polygon.

Algorithm

1. Iterate over the edges of the polygon and the edges of the map. and find an edge in the map which has the same length and orientation as an edge in the polygon.
2. Translate the visibility polygon such that the matching edge of the map polygon and the visibility polygon coincide.
3. For each of the remaining edges of the visibility polygon, check whether a complete match exists or not. If all the remaining edges match, the point where the origin was translated is added to the set of hypotheses.

Examples I

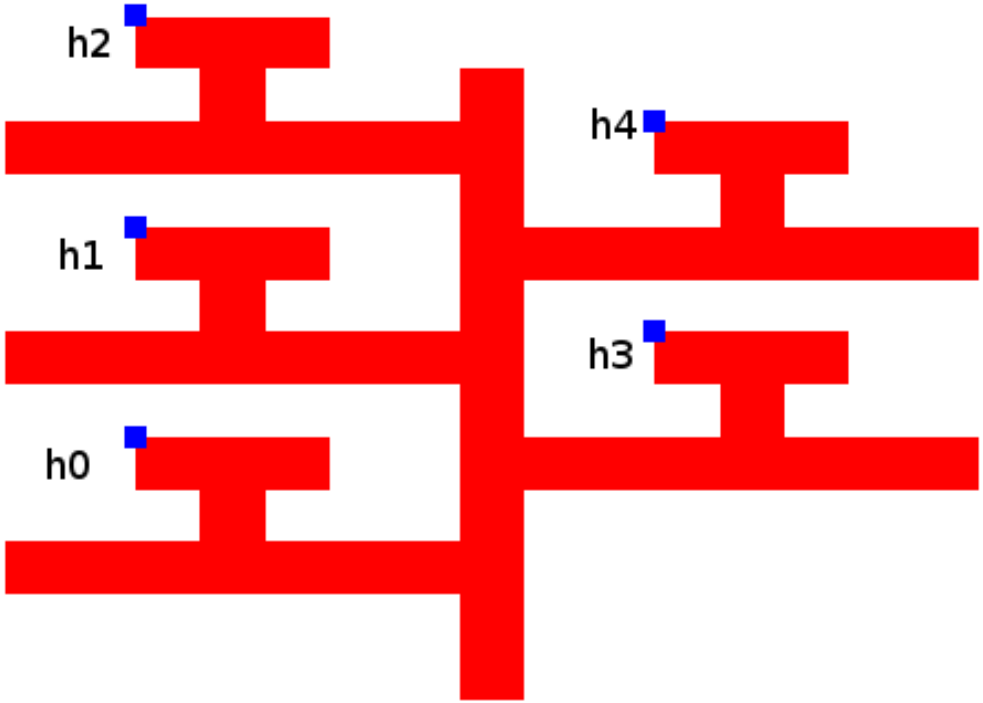

Figure: Hypothesis Generation

Examples II

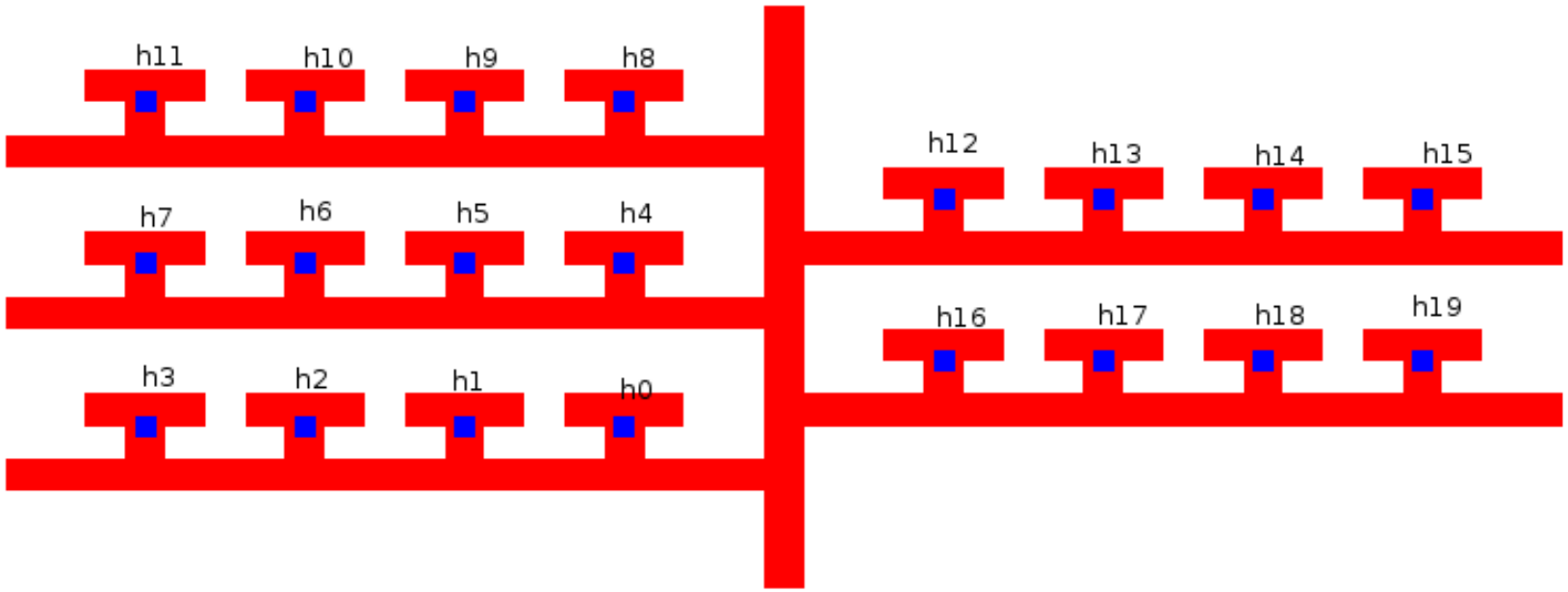

Figure: Hypothesis Generation