An Approximation Algorithm for the Robot Localization Problem

Sven Koenig[1] Apurva Mudgal[2] Craig Tovey[2]

[1] Computer Science Department
University of Southern California
skoenig@usc.edu

[2] College of Computing *
Georgia Institute of Technology
{apurva, ctovey}@cc.gatech.edu

April 11, 2004

Abstract

Localization is a fundamental problem in robotics. The robot possesses line-of-sight sensors, a compass, and a map of its polygonal environment; it must determine its location at a minimum cost of travel distance. Localization is NP-hard [3], even to minimize within a $c \log n$ factor [15], where n is the number of polygon vertices. No approximation algorithm for the problem has been known. We give a strongly polynomial time $O(\log^2 n \log r)$-factor approximation algorithm, where r is the number of reflex vertices. Technical features of the algorithm include a new edge-visibility based partition decomposition of the plane, and the idea of repeatedly planning travel on a "majority-rule" map, which permits a plan to be a route rather than a decision tree.

1 Introduction

Localization is a fundamental task in mobile robotics. A robot is equipped with a compass and map of its environment but does not know its current location ("kidnapped robot problem"). Its sensors tell it the distances to the obstacles surrounding it. The task of the robot is to uniquely identify its current location (if possible). There are often many locations in the environment that are consistent with the current sensor information, especially in self-similar environments such as corridor environments. The robot thus typically has to move in the environment to collect additional sensor information. This way, it can rule out locations that are inconsistent with the sensor information, until only one possible location is left. At this point in time, the robot is localized. Ideally, the robot should localize with a small travel distance to guarantee that it localizes quickly since its sensing and computation time are typically negligible and its localization time is thus directly proportional to its travel distance.

Localization is important for mobile robots because they often get turned off and moved to a different location for maintenance and then need to determine their current location once they get turned on again. Robots also can become confused about their location because of accumulated odometer error [**?**]. In these contexts, localization eliminates the need for complex and expensive positioning systems [2], such as indoor systems of radio beacons.

We study localization in the usual robot navigation algorithm framework: the environment is a simple two-dimensional polygon P which is completely known to the robot [10]. This is a slightly simplified but reasonable environment model that fits, for example, corridor environments well. The robot is a point robot with perfect actuation and sensing. It has a compass on board that tells it its orientation relative to the environment. It also has distance sensors on board that tell it the distance to the nearest edge of P in every direction. Its actuators allow it to move within P in any direction without kinematic constraints. This is a simplified but reasonable robot model. Lasers, for example, are sensors with prop-

*This research is partly supported by NSF awards under contracts IIS-99427, IIS-00907, and ITR/AP-01131. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the sponsoring organizations and agencies or the U.S. government.

erties close to the ones assumed here. They are long distance sensors that measure the distance to the obstacles surrounding the robot in increments of fractions of degrees and with little noise [6, 2].

Previous Work Despite the considerable attention it has received in the robotics literature (e.g. [2, 16, 14, 12, 13, 6], localization has been the subject of much less theoretical work. Guibas *et al.* [10] devise an algorithm to output all possible locations inside P that are consistent with a single observation V of the robot. They compute a special decomposition of P into cells called the "visibility cell decomposition". Using this preprocessing, they give a scheme that generates possible locations in $O(\log n+m+A)$ time where n is the number of vertices in P, m is the number of vertices in the visibility polygon and A is the size of the set of possible locations.

All previous work on devising a localization strategy has focused on the competitive analysis criterion first introduced into the realm of navigation problems by Papadimitriou and Yannakakis [11]. Kleinberg *et al.* [9] give an $O(n^{\frac{2}{3}})$ competitive algorithm for localization on a geometric tree which asymptotically improves on the "spiral search" technique of Baeza-Yates *et al.* [1]. Dudek *et al.* [3] give a polynomial time algorithm that causes the robot to travel distance at most $2(k-1)d$ in the polygonal model, where k denotes the size of the set of possible locations generated by the algorithm of [10] and d denotes the cost of a minimum verification tour. They also show that the problem is hard to approximate within factor k in the competitive analysis sense.

In the problem studied in [11], the map is unknown and thus worst-case analysis is meaningless. In the localization problem, the map is known, so the worst-case criterion has meaning. Indeed, we believe that this criterion better matches the roboticist's concerns with guaranteed rapid localization, rather than with comparisons against an omniscient verifier. To minimize worst-case travel, the optimal next move depends on the information gathered so far. Thus localization requires a *strategy* of information acquisition and movement. For any starting location, the strategy can be represented by a decision tree [3]. Essentially this is because information is discrete, even though movement and sensor data are continuous – either the data are consistent with a hypothesized location or they are not. A strategy's cost is the maximum incurred on any path from the root to a leaf.

To localize in polygon P, it is NP-hard to minimize worst-case travel cost [3], even to within a $c\log n$ factor [15], where n is the number of vertices of P. No approximation algorithm has previously been known for localization. We give a strongly polynomial $O(\log^2 n \log r)$-factor approximation algorithm, where r is the number of reflex vertices (vertices at which the interior angle exceeds π).

Our algorithm features a new planar partition decomposition based on a new edge-visibility decomposition. The algorithms cited above ([10, 3]) employ a vertex-visibility decomposition [10], which does not actually utilize all of the information acquired by the sensors. For example, if only a portion of the interior of an edge were visible through a gap formed by other edges, then this information would not be utilized in the vertex decomposition. But for an approximation guarantee, we have to compare against the best a robot could do, and we cannot restrict that robot's performance by not letting it utilize all available information. Our new decomposition may be useful in developing other approximation algorithms for robot motion problems.

1.1 Algorithm Overview

Here we describe the main ideas of the algorithm. As soon as the robot's sensors scan from the initial location, the set H of hypotheses, i.e. of locations in P consistent with information gathered so far, is size at most r [10]. At a cost of a $\log r$ approximation factor, reduce the problem to the HALF-LOCALIZE problem, which is to reduce the set H to at most half its size. If a robot is blocked while attempting to move along a path which is unblocked (does not cross an edge of P) with respect to at least $|H|/2$ of the hypotheses, then the robot has half-localized. It is not hard to prove that HALF-LOCALIZE can be solved optimally by such a majority-rule path (Corollary 2). Majority-rule finesses the troublesome decision trees, an idea that may be useful in other algorithm design problems where information gathering and decision-making are interleaved ($\alpha|H|$ for any $\alpha \leq .5$ works). The visibility partition decomposition provides, in polynomial time, a partition of the plane such that the information acquired from all points within the interior of a cell is the same with respect to all hypotheses (Theorem 2). This decomposition is built from multiple copies of visible-edge decompositions. Modulo the infinitesimal additional cost to enter the cell interiors, it suffices for paths to be piecewise linear between edges and vertices of the cells, except the starting point which may be interior. Discretizing the edges for sufficiently small ϵ would lead to an algorithm polynomial in the encoding length, but we seek a strongly polynomial algorithm. Ideally we would like a polynomial size set of points that contains all the breakpoints of the optimal paths, but such sets are exponentially large. Instead, we construct a polynomial size set of possible breakpoints which yield paths within a factor 5 of optimal. We then reduce HALF-LOCALIZE at $O(1)$-factor cost to a polynomial size $\frac{1}{2}$-group Steiner planar problem, which finally is solved within factor $O(\log^2 n)$ by combining the algorithms of [7, 5, 4].

2 Assumptions and Definitions

We assume a mobile point robot placed on a flat two-dimensional surface. The robot is equipped with a compass and line-of-sight sensors. The robot is constrained to lie inside or on the boundary of a n-vertex simple polygon P on the surface. Call P the *map polygon*, and write $\mathcal{P}$ for the set of points lying inside or on the boundary of P. The robot can move in any desired direction on the surface.

Definition 1 *Two points $p, q \in \mathcal{P}$ are visible from each other if the straight line segment joining them does not intersect the exterior of P. The visibility polygon $V(p)$ is the polygon consisting of all points visible from p. $V(p) = \phi$ if p lies outside P.*

The sensors can provide the robot with the visibility polygon $V(p)$ with respect to its current position p.

A hypothesis $h \in H$ is *active* if the robot has not yet ruled out h as a candidate for its initial position p_0. We abuse notation and denote the set of active hypotheses at any time by H. Whether H refers to the initial set of hypotheses or the set of currently active hypotheses will be clear from context.

LOCALIZE(P,H): Devise a strategy by which the robot can correctly eliminate all but one hypothesis from H, thereby determining its exact initial position $p_0 \in H$. The robot should travel a distance as small as possible to achieve this.

Definition 2 *Point q is at coordinate p relative to point c if $q = c + p$. Given the set of hypotheses H and a coordinate p, the visibility partition $H(p)$ is the partition of H given by the following equivalence relation: $h_1 \sim h_2$ iff $V(h_1 + p) = V(h_2 + p)$.*

$V(p, G)$ denotes the common visibility polygon for all hypotheses in class G of $H(p)$. $G(p, V)$ denotes the class of $H(p)$ with visibility polygon V. Let $C(p, S)$ denote the distance traveled by a robot initially located at $p \in P$ and guided by strategy S before it *localizes* i.e., determines its initial position p_0.

Definition 3 *The worst-case cost of a strategy S for a set of hypotheses H is defined as $W(H, S) = max_{p \in H} C(p, S)$.*

An optimal strategy for LOCALIZE(P,H) is the strategy with minimum worst-case cost $W(H, S)$. $OPT(P, H)$ denotes the cost of optimal strategy.

3 Half-localize

HALF-LOCALIZE(P,H): Devise a strategy by which the robot can correctly eliminate at least half of the hypotheses in H. The robot should travel a distance as small as possible to achieve this.

The worst-case cost $W(H, S)$ of a strategy for HALF-LOCALIZE(P,H) is defined as $max_{h \in H} C(h, S)$, where $C(h, S)$ denotes the distance traveled by a robot initially at h and guided by S before it *half-localizes* i.e., eliminates at least half the hypotheses from H. HALF-OPT(P,H) denotes the cost of an optimal strategy for HALF-LOCALIZE(P,H).

Lemma 1 *HALF-OPT(P,H') $\leq$ $OPT(P, H') \leq OPT(P, H)$ if $H' \subseteq H$.*

Lemma 2 *Let P be the map polygon and H the initial set of hypotheses. A robot guided by strategy* **RHL** *localizes by traveling at most $O(\log |H|)$OPT(P,H) distance by repeatedly using the optimal strategy for HALF-LOCALIZE(P,H).*

Proof: As the number of hypotheses reduces by at least half after each phase, the robot localizes in $m \leq \lceil \log |H| \rceil = \lceil \log k \rceil$ phases. By lemma 1, the distance traveled by the robot in each phase is at most 2HALF-OPT(P,H_i)$\leq$ 2OPT(P,H). Therefore the total worst-case travel distance $\leq m$OPT(P,H)$\leq$ $O(\log |H|)$OPT(P,H).■

Data : P the map polygon, H the set of active hypotheses
Result : The robot localizes to its initial position $p_0 \in P$

while $|H| > 1$ **do**
  Algorithm $\mathcal{A}(P, H)$;
  begin
    Compute $D(P, H), P^*_H$(sections 4.2,5);
    Compute $Q_{P,H}$ (section 6);
    Form instance $\mathcal{I}_{P,H}$ (section 7);
    Solve $\mathcal{I}_{P,H}$ using $\mathcal{A}'$ (theorem 4);
    Convert the solution to a halving curve C (lemma 15);
  end
  Half-localize by tracing curve C (lemma 6);
  Move back to the initial location $p_0 \in P$;
end

Algorithm 1: Strategy RHL

4 A New Decomposition

First we define a decomposition which will be used as a building block for the partition decomposition.

4.1 Visible edge decomposition

Let P denote a n-vertex simple polygon in the plane. An edge e of P is *visible* from a point $p \in \mathcal{P}$ if at least one interior point (i.e., a point except the end points) of e is visible from p. For a point $p \in \mathcal{P}$, the *edge skeleton* $E^*(p)$ is the subset of edges of P visible from p.

Definition 4 *A visible edge cell C is a maximally connected subset of $\mathcal{P}$ such that the edge skeletons $E^*(p) = E^*(q)$ for any two points p, q contained in C. A visible edge decomposition is a partition of $\mathcal{P}$ into visible edge cells.*

For an edge e of P and a point $p \in \mathcal{P}$, $e(p)$ denotes the subset of points on e visible from p. It is easy to show that $e(p)$ is a line segment. We denote the (possibly) two end points of $e(p)$ by $e_1(p)$ and $e_2(p)$ respectively.

Figure 1: Visible Edge Decomposition

For points $p, q \in P$, $Ray(p, q)$ denotes a ray starting from p which is collinear with line segment pq and is directed away from p. Each ray partitions the plane into two regions which we call left and right based on the ray's direction. For each reflex vertex r and each edge e visible from r, we introduce two rays $Ray(e_1(r), r)$ and $Ray(e_2(r), r)$ in the interior of P (see Fig 4.1(a)). We assume that these rays are line segments by restricting them to the interior of P. We write $E(P)$ for the partition of P formed by these line segments (Fig 4.1(b)). By a cell $C \in E(P)$ we mean a maximally connected region containing no line segments. By the boundary $B(C)$ of a cell we mean the set of line segments bounding it.

Lemma 3 *$E(P)$ satisfies the following:*
(i) The cells of $E(P)$ are convex polygonal regions in the interior of P, and
(ii) $E^*(p) = E^*(q)$ for every two points p and q contained in a cell $C \in E(P)$.

Proof: It is obvious that cells of $E(P)$ will be simple polygons. Let C be a cell with non-convex boundary. Let r be a reflex vertex on $B(C)$. Clearly r cannot lie in the interior of P, as otherwise the two rays intersecting at r will further divide C. On the other hand, if r lies on P then r must be a vertex of P. But then the rays formed by the two edges of P adjacent to r

will divide C. Hence $B(C)$ must be polygonal and convex.

For the second part, take two points p, q contained in C such that $E^*(p) \neq E^*(q)$. Let e be an edge in $E^*(p)$ but not in $E^*(q)$. Since $B(C)$ is convex, the line segment l joining p and q is also contained in C. Let x be a point on pq such that $e(x)$ reduces to a single point (such a point must exist). Let v_1 and v_2 be the vertices of P touching the line segment $xe(x)$ from left and right respectively. Let v_1 be the vertex closer to x than v_2. Then v_1 is a reflex vertex of P and $e(v_1)$ has $e(x)$ as one of its end points. Therefore $Ray(e(x), v_1) \in \mathcal{P}$ intersects pq at x and hence subdivides C (a contradiction). ■

Theorem 1 *$E(P)$ is a visible edge decomposition of P. $E(P)$ can be computed in time polynomial in n and has cardinality $O(n^4)$.*

Proof: The first part follows from lemma 3. As $E(P)$ contains at most two rays for every pair of a reflex vertex and an edge, it has at most $O(n^2)$ rays. Since at most $O(n^4)$ regions can be formed by $O(n^2)$ lines, this is an upper bound on the cardinality of $E(P)$. The decomposition can be computed in polynomial time using obvious algorithms. ■

4.2 Visibility Partition Decomposition

Let P be the map polygon and $H = \{h_1, h_2, \ldots, h_k\}$ the set of active hypotheses.

Definition 5 *For a set of hypotheses H, a visibility partition cell C is a maximally connected subset of the coordinate plane such that the visibility partitions $H(p) = H(q)$ for any two points p, q contained in C. A visibility partition decomposition is a partition of the coordinate plane into visibility partition cells.*

Let P_i denote a copy of the map polygon P in the coordinate plane such that hypotheses h_i coincides with the origin **0**. For each pair of distinct hypotheses $H_{ij} = \{h_i, h_j\}$ we write $D(P, H_{ij})$ for the visibility partition decomposition with respect to H_{ij}. The visibility partition decomposition $D(P, H)$ for H is then constructed by taking the union of all line segments in the partitions $\{D(P, H_{ij}) | 1 \leq i < j \leq k\}$. The construction of $D(P, H_{ij})$ (see Fig 4.2) is as

Figure 2: The subpartition $D(P, H_{12})$

follows: we first superimpose P_i and P_j to get a preliminary partition $P_i \bigcup P_j$ of the coordinate plane. $D(P, H_{ij})$ is then formed by taking the visible edge decompositions of each cell in $P_i \bigcup P_j$. (Note that we do not take the visible edge decomposition of the region formed by the intersection of the exteriors of P_i and P_j.) As stated earlier, a cell C of a partition is a maximally connected region containing no line segments and its boundary $B(C)$ is the set of segments bounding it.

Lemma 4 *Let C be a cell of $D(P, H_{ij}$. Then exactly one of the following holds: (i) $V(p+h_i) = V(p+h_j)$ for all coordinates $p \in C$ or, (ii) $V(p+h_i) \neq V(p+h_j)$ for all coordinates $p \in C$.*

Proof: If $C = ext(P_i) \bigcap ext(P_j)$, then $V(p+h_i) = V(p+h_j) = \phi$ and we are done. For the other case, let C_1 be the cell of $P_i \bigcup P_j$ containing C. Let $S \subset B(C_1)$ be the subset of edges of C_1 visible from every point in C (recall that C is a cell of the visible edge decomposition $E(C_1)$). If an edge $e \in S$ belongs to just P_i, then $V(p+h_i) \neq V(p+h_j)$ for all coordinates $p \in C$. On the other hand, if all edges in S belong to both P_i and P_j then a robot at coordinate p will see the same visibility polygon irrespective of whether it was initially at h_i or h_j. ■

Lemma 5 *$D(P, H)$ satisfies the following:*
(i) The cells of $D(P, H)$ are polygonal regions, and
(ii) $H(p) = H(q)$ for every two coordinates p, q contained in a cell C of $D(P, H)$.

Proof: Every cell $C \in D(P, H)$ is equal to the intersections of cells $C_{ij} \in D(P, H_{ij})$ containing it.

Since C_{ij}'s are polygons, C will be a polygon itself. For the second part, assume two points p, q contained in C such that $H(p) \neq H(q)$. Choose a pair of hypotheses (h_m, h_n) such that they belong to the same class in $H(p)$ but not in $H(q)$. Then $p, q \in C_{mn} \in D(P, H_{mn})$ are two points such that $V(p+h_m) = V(p+h_n)$ but $V(q+h_m) \neq V(q+h_n)$, thereby contradicting lemma 4 above. ■

Theorem 2 *For a simple polygon P and set of hypotheses $H = \{h_1, h_2, \ldots, h_k\}$, the visibility partition decomposition $D(P, H)$ can be computed in polynomial time and has cardinality $O(k^4 n^8)$.*

Proof: That $D(P, H)$ is a visibility partition decomposition follows from lemma 5. The cardinality of $P_i \bigcup P_j$ is $O(n^2)$ and therefore the cardinality of $D(P, H_{ij})$ is $O(n^4)$. Hence $D(P, H)$ is formed by at most $O(k^2 n^4)$ lines and is of total cardinality $O(k^4 n^8)$. The algorithm for computing $D(P, H)$ follows from its construction above. ■

Corollary 1 *By ignoring an arbitrarily small positive cost, robot movement can be assumed to be piecewise linear with breakpoints at edges of $D(P, H)$.*

Proof idea: For each edge e of $D(P, H)$, the first time the robot hits e (at an endpoint or interior) it can visit the interior of each cell adjacent to the relative interior of e at arbitrarily small cost. New information is acquired only at these limited times. Hereafter we assume that when the robot is at point q it can acquire all information available within arbitrarily small neighborhoods of q.

5 Halving Curves

Notation $Maj(p)$ denote the maximum size class of visibility partition $H(p)$. $Blocked(p)$ denotes the class C of $H(p)$ with $V(C) = \phi$.

Definition 6 *A coordinate p is called half-traversable iff $|Blocked(p)| \leq \frac{1}{2}|H|$. P_H^* is the maximally connected subset of half-traversable coordinates containing the origin.*

Notation $C(u, v)$ denotes a curve in the coordinate plane with end points u and v. $C(u, .)$ means that the other endpoint of C is unspecified. For a curve C, $|C|$ denotes its length.

Definition 7 *A halving curve is a curve $C(\mathbf{0}, .) \in P_H^*$ such that $|\bigcap_{x \in C} Maj(x)| \leq \frac{1}{2}|H|$.*

Lemma 6 *Let $C \in P_H^*$ be a halving curve. A robot can correctly eliminate at least $\lceil \frac{1}{2}|H| \rceil$ hypotheses by tracing C.*

Proof: Consider a robot moving along curve C and continuously observing its environment. If the robot hits the boundary of P at coordinate $x \in C$, it localizes to a set of size at most $|Blocked(x)| \leq \frac{1}{2}|H|$ (since $x \in P_H^*$). If the observation V taken by the robot at coordinate $p \in C$ is different from the majority observation $V(p, Maj(p))$, the robot half-localizes to a subset of $H \backslash Maj(p)$ which has size at most $\frac{1}{2}|H|$. The only other possibility is that the robot reaches the other end point of C without hitting the polygon P or taking a non-majority observation. However the set of active hypotheses in this case $|H| = |\bigcap_{x \in C} Maj(x)| \leq \frac{1}{2}|H|$ (since C is a halving curve) and hence the robot half-localizes. ■

Lemma 7 *There exists a halving curve of length at most HALF-OPT(P,H).*

Proof: Let S be an optimal strategy for HALF-LOCALIZE(P,H). Imagine a robot guided by S which stops as soon as it half-localizes. Let $C(\mathbf{0}, p_f)$ be the maximum length path traced by the robot in the coordinate plane for any (initial) position in H. By definition $|C| \leq$ HALF-OPT(P,H). Let H_x denote the set of active hypotheses when the robot reaches coordinate $x \in C$. For $x \in C \backslash p_f$, $|H_x| > \frac{1}{2}|H|$ since otherwise the robot would have stopped at x itself. Therefore $|Blocked(x)| = |H| - |H_x| \leq \frac{1}{2}|H|$ and hence $C \in P_H^*$.

Finally we show that the set $I = \bigcap_{x \in C} Maj(x)$ is of size at most $\frac{1}{2}|H|$. For this assume a robot initially located at some $h \in I \subset H$. Guided by S, the robot would have followed path C and observed $V_x = V(x, Maj(x))$ for all $x \in C$ (since $I \subset Maj(x)$). But then $|I| = |\bigcap_{x \in C} G(x, V_x)| = |H_{p_f}| \leq \frac{1}{2}|H|$ and hence C satisfies the lemma. ■

Corollary 2 *Let $C_{P,H}^*$ denote the minimum length halving curve. Then tracing $C_{P,H}^*$ is an optimal strategy for HALF-LOCALIZE(P,H).*

6 Reference Point Set

We are aiming to extract a polynomial size set of points on which to solve a group Steiner problem. The piecewise linear curves defined by these points come within a constant factor of the optimal halving curves.

Notation. uv denotes the straight line segment joining points u and v. As before, the *boundary* $B(C)$ of a cell $C \in D(P,H)$ is the set of line segments bounding it. The *closure* $Clos(C)$ of a cell C is defined as $C \bigcup B(C)$. The *interior* of a cell is the cell minus its boundary. A cell $C \in P^*_H$ iff every coordinate in C lies in P^*_H. L denotes the set of line segments $\bigcup_{C \in P^*_H} B(C)$. V denotes the set of end points of segments in L. A *vertex* is a coordinate $v \in V$. For a point v and line segment l, let $\pi(v,l)$ denote the point closest to v on l. For a set of points X and set of line segments Y, $\pi(X,Y)$ denotes the set of points $\{\pi(p,l)|p \in X, l \in Y\}$.

The reference point set $\mathcal{Q}^0_{P,H} = \{\mathbf{0}\} \bigcup V \bigcup \pi(V,L) \bigcup \pi(\pi(V,L),L)$. The size of $Q^0_{P,H}$ is at most $O(|V||L|^2)$ and it can be computed in polynomial time.

Definition 8 *A curve $C(\mathbf{0},.) \in P^*_H$ is said to cover a cell $C_1 \in D(P,H)$ if $C \bigcap Clos(C_1) \neq \phi$. The cover of C, $Cover(C)$, is the set of all cells $C_1 \in D(P,H)$ covered by C.*

For a piecewise linear curve $C(u,v)$, $BP(C)$ denote the set of its break points. For two curves $C_1(u,v)$ and $C_2(v,w)$, $C_1 \oplus C_2$ denotes the curve with end points u,w formed by concatenating C_1 and C_2. For a set S of cells in P^*_H, $L^*_S \in P^*_H$ denotes the shortest curve $C(\mathbf{0},.)$ with one end point at origin such that $S \subseteq Cover(L^*_S)$. We write the two end points of L^*_S as **0** and e. For a curve C, $C[u,v]$ denotes the portion of C with end points u and v.

Lemma 8 *(i) L^*_S is piecewise linear, and*
(ii) $BP(L^*_S) \bigcup \{\mathbf{0}, e\} \subseteq \bigcup_{C \in S} B(C)$.

Proof Idea: Part (i) is trivial. For the second part, consider a break point $b \in BP(L^*_S)$ in the interior of a cell $C \in D(P,H)$. Let $b_1, b_2 \in C$ be points infinitesimally preceding and succeeding b on L^*_S. Then the curve $L^*_S[\mathbf{0}, b_1] \oplus b_1 b_2 \oplus L^*_S[b_2, e]$ is strictly shorter than L^*_S (by triangle inequality) and covers S (a contradiction). ■

The *anchor* of a break point $b \in L^*_S$ is the line segment $l_b \in L$ containing it. If b is a vertex, we arbitrarily choose one of the line segments containing it as its anchor. A break point is called a *reflection point* if it lies in the interior of its anchor. Clearly, every break point of L^*_S is either (i) a vertex $v \in V$ or, (ii) a reflection point.

Lemma 9 *Let r^-, r^+ be break points (or end points) immediately preceding and succeeding a reflection point $r \in L^*_S$. Then r^- and r^+ lie on the same side of the line containing its anchor l_r. Further r^- and r^+ lie on opposite sides of the line perpendicular to l_r at r.*

Proof: Suppose r^-, r^+ lie on opposite sides of l_r. Let $C_1, C_2 \in D(P,H)$ be cells containing the portion of L^*_S in the neighborhood of r. Choose points $p \in r^- r, q \in r r^+$ such that pr, rq lie in $Clos(C_1)$ and $Clos(C_2)$ respectively and pq intersects l_r. Then the curve obtained by replacing $pr \oplus rq$ by pq covers S and is strictly shorter than L^*_S (a contradiction). Hence r^-, r^+ lie on same side of l_r.

For the second part, suppose r^-, r^+ lie on same side of the line perpendicular to l_r at r. Choose point $r' \in l_r$ on the same side of r as r^-, r^+. Let r_1, r_2 be points where the line perpendicular to l_r at r' intersects $r^- r, r r^+$ respectively. By choosing r' close enough we can ensure that $r_1 r, r r_2 \in Clos(C)$. Then the the curve formed by replacing $r_1 r \oplus r r_2$ by $r_1 r' \oplus r' r_2$ covers S and is strictly shorter than L^*_S (a contradiction). ■

Lemma 10 *(i) The anchor l_r of a reflection point $r \in L^*_S$ intersects L^*_S only at r.*
(ii) If end point e is not a vertex, its anchor l_e intersects L^*_S only at e.

Proof: Suppose l_r also intersects L^*_S at r'. Let $C_1, C_2 \in D(P,H)$ be cells adjacent l_r. By lemma 9, the portion of L^*_S in the neighborhood of r lies completely in one of $Clos(C_1)$ or $Clos(C_2)$. Let r_1, r_2 be points infinitesimally preceding and succeeding r on L^*_S such that triangle $\triangle r_1 r r_2$ lies completely in $Clos(C_1)$. Then the curve $L^*_S[\mathbf{0}, r_1] \oplus r_1 r_2 \oplus L^*_S[r_2, e]$ covers S (it covers C_2 at r') and is strictly shorter than L^*_S (a contradiction). ■

Figure 3: Lemmas 11 and 12

Lemma 11 *Let $uv \in P_H^*$ be a line segment such that v lies in the interior of $l_v \in L$. Let v' be some other point on l_v. Then either:*
(i) $uv' \in P_H^*$ and $Cover(uv) \subseteq Cover(uv')$, or
(ii) there exist points $v'' \in vv', u' \in uv''\backslash\{u\}$ such that $u' \in V$, $uv'' \in P_H^*$ and $Cover(uv) \subseteq Cover(uv'')$.

Proof: Let $v_t, t \in [0,1]$ denote the point $tv + (1-t)v'$. Let t_0 be the largest t such that $Cover(uv) \subseteq Cover(uv_t)$ and $uv_t \in P_H^*$. If t_0 is 1, we satisfy part (i) of the lemma. Theorefore assume $0 \leq t_0 < 1$.

Take $t' > t_0$ arbitrarily close to t_0. By definition of t_0, there exists a cell $C \subseteq Cover(uv_{t_0})$ such that either $C \notin Cover(uv_{t'})$ or $uv_{t'} \in P_H^*$. In either case it is easy to show that there exists a vertex $u_1 \in uv_{t_0}\backslash\{u\}$. Take $v'' = v_{t_0}, u' = u_1$ (see Fig 6(a)). ■

Lemma 12 *Let $C(u,v) \in P_H^*$ be a piecewise linear curve such that v lies in the interior of a line segment $l_v \in L$. Let v' be some other point on l_v. Then there exists a piecewise linear curve $C'(u,v') \in P_H^*$ such that*
(i) $cover(C) \subseteq Cover(C')$,
(ii) $BP(C') \subseteq BP(C) \bigcup V$ and,
(iii) $|C'| \leq |C| + |vv'|$.

Proof: Let $b \in BP(C) \bigcup \{u\}$ be the break point immediately preceding v. Define a sequence $\{(u_i, v_i) | i \geq 0\}$ as follows:
(i) $u_0 = b, v_0 = v$.
(ii) If $Cover(u_i v_i) \subseteq Cover(u_i v')$ and $u_i v' \in P_H^*$, set $u_{i+1} = u_i, v_{i+1} = v'$. Otherwise let $v'' \in v_i v', u' \in u_i v''\backslash\{u_i\}$ be points as promised in Lemma 11 above. Set $u_{i+1} = u', v_{i+1} = v''$.

Let $C_i(u, v_i)$ denote the curve $C(u,b) \oplus bu_1 \oplus \ldots \oplus u_{i-1}u_i \oplus u_i v_i$ (see Fig 6(b)). The following properties are easy to establish by induction: (i) $BP(C_i) \subseteq BP(C_{i-1}) \bigcup V$, (ii) $Cover(C_i) \supseteq Cover(C_{i-1})$ and (iii) $|C_i| \leq |C_{i-1}| + |v_{i-1}v_i|$. (Use lemma 11 and the triangle inequality).

Since $u_i \in V\backslash\{u_0, u_1, \ldots, u_{i-1}\}$, there exists a $k \leq |V|$ such that $v_j = v'$ for all $j \geq k$. Complete the proof by showing that $C' = C_k$ satisfies the lemma. ■.

Lemma 13 *Let $C(u,v) \in P_H^*$ be a piecewise linear curve such that v lies in the interior of a line segment $l_v \in L$. Then there exists a piecewise linear curve $C'(u,v') \in P_H^*$ such that*
(i) $v' \in l_v$. v' is either an end point of l_v or C' is perpendicular to l_v at v'.
(ii) $Cover(C) \subseteq Cover(C')$,
(iii) $BP(C) \subseteq BP(C') \bigcup V$ and
(iv) $|C'| \leq |C|$.

Proof Idea: Let $b \in BP(C) \bigcup \{u\}$ be the point immediately preceding v. Define a sequence $\{(u_i, v_i, v_i') | i \geq 0\}$ as follows:
(i) $u_0 = b, v_0 = v$. Set $v_0' = \pi(v_0, l_v)$.
(ii) If $Cover(u_i v_i) \subseteq Cover(u_i v_i')$ and $u_i v_i' \in P_H^*$, set $u_{i+1} = u_i, v_{i+1} = v_i', v_{i+1}' = v_i'$. Otherwise let $v'' \in v_i v_i', u' \in u_i v''\backslash\{u_i\}$ be points as promised in Lemma 11 above. Set $u_{i+1} = u', v_{i+1} = v'', v_{i+1}' = \pi(u_{i+1}, l_v)$.
The rest of the proof is similar to lemma 12, showing that for some $0 \leq k \leq |V|$, the curve $C_k(u, v_k) = C(u,b) \oplus bu_1 \oplus \ldots \oplus u_{i-1}u_i \oplus u_i v_i$ works. ■

Lemma 14 *Let r be a reflection point of L_S^* and let r^- and r^+ be break points immediately preceding and succeeding it. Then there exists a point $r' \in l_r \bigcap Q_{P,H}^0$ such that $|rr'| \leq |r^- r| + |rr^+|$.*

Proof: Let l_r denote the anchor of r. Let Π^-, Π^+ be lines perpendicular to L_r passing through r^-, r^+ respectively. Let H^-, H^+ be half-spaces containing r bounded by Π^-, Π^+ respectively. H denotes the convex region $H^- \bigcap H^+$.

Figure 4: Proof of lemma 14

Let v be a vertex in H. Clearly $r' = \pi(v, l_r) \in \mathcal{Q}^0_{P,H}$ satisfies the lemma, since $|rr'| \leq |r^-r^+| \leq |r^-r| + |rr^+|$ (by triangle inequality). We complete the proof by finding a vertex $v \in H$. The only non-trivial case is shown in Fig 6: l_r intersects π^- and π^+ and both r^- and r^+ lie in the interior of their anchors l_{r^-} and l_{r^+} respectively. Let $v^-, v^+ \in V$ denote the end points of l_{r^-} and l_{r^+} lying in $\overline{H^-}$ and $\overline{H^+}$ respectively. Suppose on the contrary that $v^- \in \overline{H^+}$ and $v^+ \in \overline{H^-}$. Let q be the point where r^-v^- intersects Π^+. Since L is non-intersecting and l_{r^-} intersects L^*_S only at r^- (by lemma 10), q must lie on the portion of Π^+ below (see Figure) r^+. Consider the quadrilateral Q formed by r, r^-, q and r^+. Clearly $Q \in H$. Further since L is non-intersecting and l_{r^+} intersects L^*_S only at r^- (lemma 10(ii) if r^+ is an endpoint, else by lemma 10(i)), the segment r^+e^+ lies completely in Q. Therefore $e^+ \in H \bigcap V$ (a contradiction). ■

Theorem 3 *(Geometric Approximation Theorem) There exists a piecewise linear curve $L_S(\mathbf{0}, e') \in P^*_H$ such that*
(i) $S \subseteq Cover(L_S)$,
(ii) $B(L_S) \bigcup \{\mathbf{0}, e'\} \subseteq \mathcal{Q}^0_{P,H}$, and
(iii) $|L_S| \leq 5|L^*_S|$.

Proof: Let $r_1, r_2, \ldots, r_{k-1}$ be the reflection points of L^*_S. Let $l_i \in L$ denote the anchor of r_i. Choose points $r'_i \in \mathcal{Q}^0_{P,H} \bigcap l_i$ such that $|r_ir'_i| \leq |r^-_i r_i| + |r_ir^+_i|$ (use lemma 14). For purposes of proof take $r_0 = \mathbf{0}$ and $r_k = e$ as the two endpoints of L^*_S and define $r'_0 = r_0, r'_k = r_k$. Let $C_i = L^*_S[r_{i-1}, r_i], 0 < i \leq k$ denote the subpath of L^*_S from r_{i-1} to r_i.

Let $C'_i(r_{i-1}, r'_i)$ be the piecewise linear curve promised by lemma 12 under the substitution $C = C_i, u = r_{i-1}, v = r_i$ and $v' = r'_{i-1}$. Apply lemma 12 again to obtain $C''_i(r'_{i-1}, r'_i)$. (Take $C = C'_i, u = r'_i, v = r_{i-1}$ and $v' = r'_{i-1}$ in this case). The following properties follow from lemma 12 itself:
(i) $B(C_i) \subseteq B(C'_i) \bigcup V \subseteq B(C''_i) \bigcup V \subseteq V$,
(ii) $Cover(C_i) \subseteq Cover(C'_i) \subseteq Cover(C''_i)$ and
(iii) $|C''_i| \leq |C'_i| + |r_{i-1}r'_{i-1}| \leq |C_i| + |r_{i-1}r'_{i-1}| + |r_ir'_i|$.

We show that $L_S = \oplus^k_{i=1} C''_i$ satisfies the theorem. Since $Cover(L_S) = \bigcup^k_{i=1} Cover(C''_i) \supseteq \bigcup^k_{i=1} Cover(C) \supseteq Cover(L^*_S)$, L_S covers S. The set of break points $B(L_S) \subseteq (\bigcup^k_{i=1}(B(C''_i) \bigcup \{r'_{i-1}, r'_i\} \subseteq V \bigcup \{r'_1, r'_2, \ldots, r'_{k-1}\}$. Since $r'_i \in \mathcal{Q}^0_{P,H}$, $B(L_S) \subseteq \mathcal{Q}^0_{P,H}$. Finally the length of L_S, $|L_S| = \sum_{i=1}^{k} |C_i| \leq \sum_{i=1}^{k}(|C_i| + |r_{i-1}r'_{i-1}| + |r_ir'_i|) = |L^*_S| + 2\sum_{i=1}^{k-1}|r_ir'_i|$. Since r'_i was chosen according to lemma 14, this is at most $|L^*_S| + 2\sum_{i=1}^{k-1}(|r^-_i r_i| + |r_ir^+_i|) \leq 5|L^*_S|$. Clearly the starting point origin lies in V. If the other endpoint $r_k = e$ lies in the interior of a line segment l_e, replace C''_k by the curve promised by lemma 13 under the substitution $C = C''_k, u = r_{k-1}, v = e, l_v = l_e$. ■

7 Group Steiner Problem and Main Result

Rooted $\frac{1}{2}$-Group Steiner Problem We are given a graph $G = (V, E)$ with a cost function $c : E \rightarrow \Re^+$ on edges, a special vertex (call it *root*) $r \in V$ and sets of vertices (called *groups*) $g_1, g_2, \ldots, g_k \subset V$. A tree T covers a group g if $T \bigcap g \neq \phi$. The objective is to find the min-cost tree containing r and covering at least half the groups.

Theorem 4 *[7, 5, 4] There exists a $O(\log n. \log max_i|g_i|)$ factor approximation algorithm $\mathcal{A}'$ for the rooted $\frac{1}{2}$-Group Steiner problem where n is the number of vertices in G.*

For points $p_1, p_2 \in P_H^*$, we denote the shortest path in P_H^* joining them by $Path(p_1, p_2)$. $d(p_1, p_2)^{P_H^*}$ denotes the length of $Path(p_1, p_2)$. Augment the reference point set as follows: for every pair of points $p_1, p_2 \in \mathcal{Q}_{P,H}^0$, we add all intersection points of line segment p_1p_2 with segments in L. Let $\mathcal{Q}_{P,H}$ denote the augmented set of points. Recall that when the robot visits a point $p \in \mathcal{Q}_{P,H}$ with infinitesimal additional distance it acquires information about all cells of $D(P, H)$ adjoining it. Hence we can assume that $Maj(p), p \in \mathcal{Q}_{P,H}$ is equal to $\bigcap_{p \in Clos(C)} Maj(C)$, where C denotes a cell of $D(P, H)$.

Definition 9 *INSTANCE $\mathcal{I}_{P,H}$: Take G as the complete graph on $V = \mathcal{Q}_{P,H}$. Define the cost of an edge (q_1, q_2) as $d(q_1, q_2)^{P_H^*}$. Take the root as the the origin $\mathbf{0} \in \mathcal{Q}_{P,H}$. Make k groups where $H = \{h_1, h_2, \ldots, h_k\}$. Group g_i consists of all coordinates $q \in \mathcal{Q}_{P,H}$ such that $h_i \notin Maj(q)$.*

Lemma 15 *Every tree $T = (V', E')$ rooted at $\mathbf{0}$ covering at least half the groups in $\mathcal{I}_{P,H}$ can be converted to a halving curve of length at most $2c(T)$.*

Proof: Let $v_0 = \mathbf{0}, v_1, v_2, \ldots, v_l$ be the vertices of V' in the order they are visited by a depth-first search starting from the root $\mathbf{0}$. We show that $C = \bigoplus_{i=1}^{l} Path(v_{i-1}, v_i)$ is the required halving curve. Since each edge is traversed at most twice, $|C| = \sum_{1}^{l} d(v_{i-1}, v_i)^{P_H^*} \leq 2.c(T)$. Further $h_i \notin \bigcap_{v \in V'} Maj(v)$ if T covers group g_i. Since T covers at least half the groups in $\{g_1, g_2, \ldots, g_k\}$ this implies $|\bigcap_{v \in V'} Maj(v)| \leq \frac{1}{2}|H|$. ■

Lemma 16 *The optimal solution to $\mathcal{I}_{P,H}$ has cost at most $5.|C_{P,H}^*|$, where $C_{P,H}^*$ denotes the optimal halving curve.*

Proof: Let S be the set of cells covered by $C_{P,H}^*$. Let L_S be the path covering S as promised by theorem 3. For $C \in S$, let p_C denote the first point where L_S intersects its boundary $B(C)$. Let $p_1, p_2, \ldots, p_{|S|}$ be an ordering of these points according to their occurrence on L_S. Since the end points and break points of L_S are in $Q_{P,H}^0$, $p_i \in \mathcal{Q}_{P,H}$. Then the path $\mathbf{0}, p_1, p_2, \ldots, p_{|S|}$ is a solution to instance $\mathcal{I}_{P,H}$. Its cost is at most $\sum_{i=1}^{|S|} d(p_{i-1}, p_i)^{P_H^*} \leq |L_S| \leq 5|C_{P,H}^*|$. ■

Theorem 5 *Algorithm $\mathcal{A}$ computes a halving curve of length at most $O(\log^2 n).|C^*|$ in time polynomial in n and k.*

Proof: Let $T = (V', E')$ be the solution to $\mathcal{I}_{P,H}$ found by algorithm $\mathcal{A}$. By theorem 4 and lemma 16, $c(T) \leq O(\log n. \log max_i|g_i|).c(T^*) = O(\log^2 n)|C_{P,H}^*|$ where T^* is the optimal solution to $\mathcal{I}_{P,H}$. In the next step the algorithm just mimics the proof of lemma 15 above to get a halving curve of length at most $2.c(T) = O(\log^2 n).|C_{P,H}^*|$. Since computing the reference point set $Q_{P,H}^{\epsilon}$, making instance $\mathcal{I}_{P,H}$ and algorithm $\mathcal{A}'$ take polynomial time, $\mathcal{A}$ is a polynomial time algorithm. ■

Theorem 6 *Let P be the map polygon and H the set of active hypotheses. Let C be the halving curve computed by algorithm $\mathcal{A}$. A robot tracing curve C correctly eliminates at least $\lceil \frac{1}{2}|H| \rceil$ hypotheses from H. It travels distance at most $O(\log^2 n)$HALF-OPT(P,H) where $k = |H|$, n is the size of P and HALF-OPT(P,H) is the cost of an optimal strategy for HALF-LOCALIZE(P,H).*

Proof: Combine lemmas 6, 7 and theorem 5. ■

Theorem 7 *Let P be the map polygon and H the initial set of hypotheses. A robot guided by strategy* **RHL** *correctly determines its initial position $p_0 \in P$ by traveling at most $O(\log^2 n \log k)OPT(P, H)$ distance where $k = |H|$ and n is the number of vertices in P.*

Proof: Follows from theorem 6 and lemma 2. ■

7.1 Extension to Polygons with Holes

Here we sketch an extension of our algorithm. When the map polygon contains polygonal holes, the cells of subpartition $D(P, H_{ij})$ will be polygons with holes rather than just simple polygons. A vertex of such a cell may see more than one continuous portion of a particular edge through the windows formed by different obstacles. Therefore the visible edge partition needs to be augmented by adding rays $R(v, e_1(v)), R(v, e_2(v)), \ldots, R(v, e_k(v))$ where v is a vertex of the map polygon (including the vertices on holes) and e_i's are the end points of the various portions (each portion will still be a line segment) of e visible from v. This leads to a decomposition of cardinality $O(n^4)$ as before. The visibility partition decomposition $D(P, H)$ can then be formed by taking intersections of all subpartitions $D(P, H_{ij})$ and hence is of polynomial cardinality. We can verify that the geometric approximation lemma and other lemmas hold for polygonal cells with holes, and therefore our algorithm extends with the same approximation guarantee but a concomitant increase in running time.

8 Conclusion

The most important open problem is to close the $\log^2$-factor gap between the upper and lower bounds. The Group Steiner problem has recently been proved to be hard to approximate within a $\log^2 k$-factor (here k is the number of groups) by Halperin *et al.*. [8]. The close relation between half-localization and Group Steiner opens up the possibility of proving a polylogarithmic factor hardness of approximation. A complementary problem in the other direction is to improve the approximation factor and running time of our algorithm.

References

[1] Ricardo A. Baeza-Yates, Joseph C. Culberson, and Gregory J. E. Rawlins. Searching in the plane. *Information and Computation*, 106(2):234–252, 1993.

[2] I. J. Cox. Blanche - an experiment in guidance and navigation of an autonomous robot vehicle. In *IEEE Trans. on Robotics and Automation*, pages 193–204, 1991.

[3] Dudek, Romanik, and Whitesides. Localizing a robot with minimum travel. In *SODA: ACM-SIAM Symposium on Discrete Algorithms (A Conference on Theoretical and Experimental Analysis of Discrete Algorithms)*, 1995.

[4] Jittat Fakcharoenphol, Satish Rao, and Kunal Talwar. A tight bound on approximating arbitrary metrics by tree metrics. In *Proceedings of the thirty-fifth ACM symposium on Theory of Computing*, 2003.

[5] G. Kortsarz G. Even and W. Slany. On network design: fixed charge flows and the covering steiner problem. In *SWAT, pages 318-329, 2002*.

[6] C. Wetzler G. Weib and E. Von Puttkamer. Keeping track of position and orientation of moving indoor systems by correlation of rangefinder scans, 1994.

[7] Garg, Konjevod, and Ravi. A polylogarithmic approximation algorithm for the group steiner tree problem. In *SODA: ACM-SIAM Symposium on Discrete Algorithms (A Conference on Theoretical and Experimental Analysis of Discrete Algorithms)*, 1998.

[8] E. Halperin and R. Krauthgamer. Polylogarithmic inapproximability. In *Proceedings of the 35th ACM Symposium on Theory of Computing*, pages 585–594, 2003.

[9] Jon M. Kleinberg. The localization problem for mobile robots. In *IEEE Symposium on Foundations of Computer Science*, pages 521–531, 1994.

[10] R. Motwani L. Guibas and P. Raghavan. The robot localization problem in two dimensions. In *Proceedings 3rd ACM-SIAM Symposium on Discrete Algorithms*, pages 259–268, 1992.

[11] C. Papadimitriou and M. Yannakakis. Shrotest paths without a map. *Theoretical Computer Science*, 84:127–150, 1991.

[12] W.D. Rencken. Concurrent localisation and map building for mobile robots using ultrasonic sensors. In *In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2129–2197, 1993.

[13] Simmons, R. Goodwin, K. Haigh, S. Koenig, and J. O'Sullivan. A layered architecture for office delivery robots. In *In Proceedings of the First International Conference on Autonomous Agents*, 1997.

[14] W. Thompson, H. Pick, B. Bennett, M. Heinrichs, S. Savitt, and K. Smith. Map-based localization: The 'drop-off' problem. In *Proc. DARPA Image Understanding Workshop*, pages 706–719, 1990.

[15] C. Tovey and S. Koenig. Gridworlds as testbeds for planning with incomplete information. In *Proceedings of the National Conference on Artificial Intelligence*, pages 819–824, 2000.

[16] C. Ming Wang. Location estimation and uncertainty analysis for mobile robots. In *Autonomous Robot Vehicles (Ed: I. J. Cox and G. T. Wilfong)*. Springer-Verlag, Berlin, 1991.