Exploring Unknown Polygonal Environments with Discrete Visibility

Subir Kumar Ghosh

School of Technology & Computer Science
Tata Institute of Fundamental Research
Mumbai 400005, India
ghosh@tifr.res.in
Overview

1. Robotic exploration problem
2. Visibility polygon
3. Continuous and discrete visibility
4. Exploring an unknown polygon: Continuous visibility
5. Exploring an unknown polygon: Discrete visibility
6. Exploring an unknown polygon: Bounded visibility
Robotic exploration problem

- Imagine that a robot is to explore the interior of a collapsed building, which has crumbled due to an earthquake, in order to search for human survivors.
- It is clearly impossible to have knowledge of the building’s interior geometry prior to the exploration.
- Thus, the robot must be able to see, with its on-board vision sensors, all points in the building’s interior while following its exploration path.
- In this way, no potential survivors will be missed by the exploring robot.

In our model for robotic exploration, we consider

- an unknown polygonal environment $P$ with holes as the search space, and
- the robot as a moving point in $P$. 
The visibility polygon of $P$ from a point $p$ (denoted as $VP(P, p)$) is the set of all points of $P$ that are visible from $p$.

In other words, for every point $z \in P$, if the line segment joining $z$ and $p$ lies inside $P$, then $z$ belongs to $VP(P, p)$.

In order to explore or see all points $P$, the robot computes visibility polygons from different locations inside $P$ such that every point of $P$ belongs to at least one such visibility polygon.

Continuous and discrete visibility

If the robot computes visibility polygons from each points on its path, we say that $P$ is explored under continuous visibility.

If the robot computes visibility polygons from a selected set of points on its path, we say that $P$ is explored under discrete visibility.
Efficiency of on-line algorithms

Suppose the polygon $P$ is not known apriori and the point robot can compute the visibility polygon of $P$ from its current position using visual sensors.

The robot wants to see all points of $P$ with minimum cost. Cost can be the length or the number of links in the path that the robot has traveled starting from its initial position.

Efficiency of the on-line algorithm:

\[
\text{Competitive ratio} = \frac{\text{cost of the on-line algorithm}}{\text{cost of the off-line algorithm}}.
\]
Algorithms for exploring an unknown environment with continuous visibility


Searching for the kernel in an unknown polygon with continuous visibility

Starting from the initial position $s$, the problem is to design a competitive strategy to walk into the kernel of $P$.

A street (also called LR-polygon) is a polygon for which two boundary chains from start to target are mutually weakly visible. So, the entire street is visible from any path from s to t.

The problem is to find a path from s to t such that the competitive ratio is the minimum.

Algorithms for searching a street


Exploring unknown polygons: discrete visibility

In the remaining part of the lecture, we present exploration algorithms and their competitive ratios from the following papers.


Motivation for discrete visibility

Many on-line computational geometry algorithms for exploring unknown polygons assume that the visibility region can be determined in a continuous fashion from each point on a path of a robot. Is this assumption reasonable?

1. Autonomous robots can only carry a limited amount of on-board computing capability.
2. At the current state of the art, computer vision algorithms that could compute visibility polygons are time consuming.
3. The computing limitations suggest that it may not be practically feasible to continuously compute the visibility polygon along the robot’s trajectory.
4. For good visibility, the robot’s camera will typically be mounted on a mast and such devices vibrate during the robot’s movement.
5. Hence for good precision the camera must be stationary while computing visibility polygons.

It seems feasible to compute visibility polygons only at a discrete number of points.
Exploration cost

Is the cost associated with a robot’s physical movement dominate all other associated costs?

The essential components that contribute to the total cost required for a robotic exploration can be analyzed as follows. Each move will have two associated costs as follows.

1. There is the time required to physically execute the move. If we crudely assume that the robot moves at a constant rate, $r$, during a move, the total time required for motion will be $rD$, where $D$ is the total path length.

2. In an exploratory process where the robot has no apriori knowledge of the environment’s geometry, each move must be planned immediately prior to the move so as to account for the most recently acquired geometric information. The robot will be stationary during this process, which we assume to take time $t_M$.

3. Since the robot is stationary during each sensing operation, we assume that it takes time $t_S$. 

Let $N_M$ and $N_S$ be respectively the number of moves and the number of sensor operations. that are required to complete the exploration of $P$.

Hence the total cost of an exploration is equated to the total time, $T$, required to explore $P$:

$$T(P) = t_M N_M + t_S N_S + rD$$

Now, $(t_M N_M + t_S N_S)$ can be viewed as the time required for computing and maintaining visibility polygons by computer vision algorithms, which is indeed a significant fraction of $T(P)$ because computer vision algorithms consume significant time on modest computers in a relatively cluttered environment.


Therefore, we assume that the overall cost of exploration is proportional to the cost for computing visibility polygons. The criteria for minimizing the cost for robotic exploration is to reduce the number of visibility polygons that the on-line algorithms compute.
An exploration algorithm

- We present an exploration algorithm that a point robot can use to explore an unknown polygonal environment $P$ under discrete visibility.

- In order to explore $P$, the robot starts from a given position, and sees all points of the free space incrementally.

- It may appear that it is enough to see all vertices and edges of $P$ in order to see the entire free-space. However, this is not the case.

- Three views from $p_1$, $p_2$ and $p_3$ are enough to see all vertices and edges of $P$ but not the entire free-space of $P$. 
(i) Let $S$ denote the set of viewing points that the algorithm has computed so far. (ii) The triangulation of $P$ is denoted as $T(P)$. (iii) The visibility polygon of $P$ from a point $p_i$ is denoted as $VP(P, p_i)$.

**Step 1:** $i := 1; \ T(P) := \emptyset; \ S := \emptyset; \text{ Let } p_1 \text{ denote the starting position of the robot.}$

**Step 2:** Compute $VP(P, p_i)$; Construct the triangulation $T'(P)$ of $VP(P, p_i); \ T(P) := T(P) \cup T'(P); \ S = S \cup p_i;$

**Step 3:** While $VP(P, p_i) - T(P) = \emptyset \text{ and } i \neq 0$ then $i := i - 1;
Step 4: If $i = 0$ then goto Step 7;
Step 5: If $VP(P, p_i) - T(P) \neq \emptyset$ then choose a point $z$ on any constructed of $VP(P, p_i)$ lying outside $T(P)$;
Step 6: $i := i + 1; p_i := z;$ goto Step 2;
Step 7: Output $S$ and $T(P)$;
Step 8: Stop.
The algorithm needs $r + 1$ views. Competitive ratio is $(r + 1)/2$, where $r$ denotes the number of reflex vertices of the polygon.

**Open Problem:** Can the bound be improved?
Convex robot exploration

We wish to design an algorithm that a convex robot $C$ can use to explore an unknown polygonal environment $P$ (under translation) following the similar strategy of a point robot. $C$ needs more than $r + 1$ views for exploration.

**Open problem:** Can one derive an upper bound on the number of views for a convex robot exploration?
The art gallery problem

An art gallery can be viewed as a polygon $P$ with or without holes with a total of $n$ vertices and guards as points in $P$.

Victor Klee asked in 1976: How many guards are always sufficient to guard any polygon with $n$ vertices?

The minimum vertex, point and edge guard problems for polygons with or without holes (including orthogonal polygons) are NP-hard.

Approximation algorithms


2. A. Efrat and S. Har-Peled, Guarding galleries and terrains, Information Processing Letters, 100 (2006), 238-245. Running time and the approximation ratio: (i) For simple polygons, $O(nc_{opt}^2 \log^4 n)$ expected time and $O(\log c_{opt})$, where $c_{opt}$ is the number of vertices in the optimal solution. (ii) For polygons with $h$ holes, $O(nhc_{opt}^3 \text{polylog } n)$ expected time and $O(\log n \log(c_{opt} \log n))$.

Suppose an optimal exploration algorithm for a point robot has computed visibility polygons from points $p_1, p_2, \ldots, p_k$.

We know that (i) $\bigcup_{i=1}^{k} V(P, p_i) = P$, (ii) $p_i \in V(P, p_j)$ for some $j < i$ and (iii) $k$ is minimum. So, $P$ can be guarded by placing stationary guards at $p_1, p_2, \ldots, p_k$.

The exploration problem for a point robot is the Art Gallery problem for stationary guards with additional visibility constraint (ii).

Our exploration algorithm for a point robot is an approximation algorithm for this variation of the Art Gallery problem.

**Open Problem**

Can one prove that the exploration problem, like the Art Gallery problem, is NP-hard?
Stationary guards in polygons with additional visibility constraint

In the standard art gallery problem, $\lfloor \frac{n}{3} \rfloor$ stationary guards are sufficient and sometime necessary for guarding $P$ containing no holes.

Suppose, guards $g_1, g_2, \ldots, g_k$ are placed in $P$ for security reasons in a such way that each guard $g_i$ for $i > 1$ is visible at least from one other guard $g_j$ for $i < j$.

In that case, $\lfloor \frac{n}{3} \rfloor$ guards are not sufficient. It has been proved by Hernandez-Penalver that $\lfloor \frac{n}{2} \rfloor - 1$ guards are not only necessary but also sufficient.
In the standard art gallery problem for a polygon $P$ contains $h$ holes, $\left\lfloor \frac{n+h}{3} \right\rfloor$ stationary guards are sufficient and sometime necessary.

If the guards also have to satisfy the visibility constraint between them as stated above, then $\left\lfloor \frac{n+h}{3} \right\rfloor$ guards are not sufficient.
We conjecture that \( \lfloor \frac{n+2h}{3} \rfloor \) guards are sufficient for this problem.


A watchman route in a polygon $P$ is a polygonal path such that every point of $P$ is visible from some point on the path.

The path of the robot produced by our exploration algorithm is a watchman route inside $P$, and therefore, it can also be used as an inspection path of autonomous inspection for subsequent traversal.

Exploring an unknown polygon: Bounded visibility

Computer vision range sensors or algorithms, such as stereo or structured light range finder, can reliably compute the 3D scene locations only up to a depth $R$. The reliability of depth estimates is inversely related to the distance from the camera. Thus, the range measurements from a vision sensor for objects that are far away are not at all reliable.

Therefore, the portion of the boundary of a polygonal environment within the range distance $R$ is only considered visible from the camera of the robot.

Vertices of restricted visibility polygon from $p_i$ with range $R$ are $u_1, u_2, \ldots, u_{12}$. 
An exploration algorithm using restricted visibility polygons

- The algorithm starts by computing the restricted visibility polygon $RVP(P, p_1)$ from the starting position $p_1$.

- It chooses the next viewing point $p_i$ on a constructed edge or a circular edge of $RVP(P, p_{i-1})$ for $i \geq 1$ till a boundary point $z$ of $P$ becomes visible.
Taking $z$ as the next viewing point $p_i$, $RVP(P, p_i)$ is computed. Taking viewing points along the boundary of $P$ in this fashion, restricted visibility polygons are computed till all points of this boundary of $P$ become visible.

The process of computing restricted visibility polygons ends once the entire $P$ is explored.
The maximum number of views needed to explore the unknown polygon $P$ with $h$ obstacles of size $n$ is bounded by

$$\left\lfloor \frac{8 \times \text{Area}(P)}{3 \times R^2} \right\rfloor + \left\lfloor \frac{\text{Perimeter}(P)}{R} \right\rfloor + r + h + 1.$$ 

The competitive ratio of the algorithm is

$$\left\lfloor \frac{8\pi}{3} + \frac{\pi R \times \text{Perimeter}(P)}{\text{Area}(P)} + \frac{(r+h+1) \times \pi R^2}{\text{Area}(P)} \right\rfloor.$$ 

Open problem

Can one improve the competitive ratio of the algorithm?
Exploration and Coverage Algorithms


Concluding remarks

In this talk, we have presented exploration algorithms under discrete visibility. It can be seen that our algorithms can be implemented easily. We expect that our algorithm will also perform efficiently in practice.