Visibility Detection among moving points in a Map in presence of Static Obstacles

---

A thesis submitted in partial fulfillment of the requirement for the degree of MASTER OF COMPUTER APPLICATION in the Department of Computer Science & Engineering, Jadavpur University

By

Sayan Ghosh

Roll No: 001910503019
Registration No: 149882 of 2019-20

Under the supervision of

DR. Chintan Kumar Mandal
Assistant Professor,
Department of Computer Science & Engineering
Faculty of Engineering and Technology
Jadavpur University

Faculty of Engineering and Technology
Department of Computer Science & Engineering
Jadavpur University

To Whom It May Concern

This is to certify that Sayan Ghosh, a student of MCA, from theDepartment of Computer Science & Engineering, under the Faculty of Engineering and Technology, Jadavpur University has done a thesis report under my supervision, entitled as "**Visibility Detection among moving points in a Map in presence of Static Obstacles**". The thesis is approved for submission towards the fulfillment of the requirements for the degree of Master of Computer Application, from the Department of Computer Science & Engineering, Jadavpur University for the session 2021-22.

Dr. Chintan Kumar Mandal
(Supervisor)
Computer Science & Engineering
Jadavpur University

Countersigned:

Prof. Anupam Sinha
(Head of the department)
Computer Science & Engineering
Jadavpur University

Prof. Chandan Mazumdar
(Dean)
Faculty of Engineering & Technology
Jadavpur University

Faculty of Engineering and Technology
Department of Computer Science & Engineering
Jadavpur University

Certificate Of Approval

The foregoing thesis is hereby approved as a credible study of an engineering subject carried out and presented in a manner satisfactory to warrant its acceptance as a prerequisite to the degree for which it has been submitted. It is understood that by this approval the undersigned do not necessarily endorse or approve any statement made, opinion expressed or conclusion drawn therein, but approve this thesis only for the purpose for which it is submitted.

Signature of Internal examiner

Signature of External examiner

Declaration Of Originality & Compliance Of Academic Ethics

I hereby declare that this thesis contains literature survey and original research work done by me, as part of my MCA studies. All information in this document has been obtained and presented in accordance with academic rules and ethical conduct.

Name: **Sayan Ghosh**
Roll No: 001910503019
Project Title: Visibility Detection among moving points in a Map in presence of Static Obstacles

____

Signature with date

Acknowledgement

I would like to express my deepest gratitude to my advisor and guide Dr. Chintan Kumar Mandal, for his excellent guidance, caring, patience, and providing me with an excellent atmosphere for doing research. I strongly believe that I got a lot of encouragement and inspiration from him throughout the project. With his invaluable guidance, this work is a successful one. I am equally grateful to Dr. Anupam Sinha, Head of The Department, Computer Science & Engineering, Jadavpur University, for his support towards our department. Last but not least, I would like to thank my parents and all respected teachers for their valuable suggestions and helpful discussions.

Regards,

Sayan Ghosh
Roll Number: 001910503019
Registration Number: 149882 of 2019-20

Signature with date

Contents

Abstract

Visibility Algorithms and their studies have a prominent place in Computational Geometry and Robotics. Visibility algorithms mainly rely on visibility graphs. Geometric algorithms problems can fall in three categories for points and geometric structures: offline algorithms, online and dynamic algorithms and kinetic systems. Offline algorithms are those where the points or geometric structures are given; online and dynamic algorithms are such that the points or geometric structures can be added or deleted from the system; and kinetic systems are when the points or geometric structures move while trying to maintain a basic geometric system.

In this work, we present an algorithm for visibility detection between a moving source and a fixed destination in presence of n randomly moving points and static convex polygonal obstacles in a map. It is shown through a probabilistic analysis that the proposed algorithm runs with $O(n^2)$ generally but runs in $O(n)$ as time progresses.

Chapter 1

Introduction

Visibility detection and finding optimal shortest path is a well known problem in the field of robot motion planning. Extensive studies and research have already been done in this regards. Visibility detection in many cases requires the construction of visibility graph. Visibility graph is a graph data structure in which the vertices are points or robots in motion and there is an edge between two vertices if the robots are visible to each other.

Most of the visibility graph algorithms runs in $O(n^2)$ and the optimal algorithm runs in $O(n \log n) + E$,where E is the number of edges in the visibility graph which in worst case can be $O(n^2)$ when all robots are visible to each other. Since this problem is more about visibility rather than path it chooses not to build visibility graph rather use some clever technique which runs in $O(n^2)$ in general but as the robots moves randomly and proceed with time the algorithm takes $O(n)$ in most of the timestamp which will be proved by probabilistic analysis of the algorithm.

Visibility analysis has a long history in the architecture and urban planning fields. [2] suggested that isovists and isovist fields are related to [5] model of visual perception because they capture the variation of visual fields that informs the spatial understanding of a person moving in an environment [5]. [1] offered a computational scheme for defining and measuring isovist. [13] proposed an innovative visibility analysis method, visibility graph analysis, by drawing on graph-based representation from space syntax [6] and social network theory [14]. Researchers from the fields of geography, architecture, and urban planning have also explored 3D isovist. Several studies have developed the $3D$-isovist algorithms and assessment tools [4] [9]. Visibility analysis is of particular interest in museum studies. Such studies have often focused on the relationship between museum layout and the structure of visitor paths [3] [7] [11]. Some studies reported that the museum layout and display setting influence visitors' exploration patterns and potential narrative understanding [12]. Patterns of co-visibility were also the focus of attention in an analysis of the changing interior design of the second floor of the High Museum of Art in Atlanta, GA [15]. [10] showed that visitors' assessment of the clarity of the presentation of a pictorial theme is associated with the degree of co-visibility of member works in virtual exhibition environments

Despite advances in the methodology and algorithms of visibility graph analysis, previous studies have two major limitations. First, visibility graph analysis and the corresponding software Depth map developed by Turner cannot deal with $3D$ environments. [8] tried to overcome this constraint by manually linking vertical connections.

This work presents an algorithm for visibility detection of n moving points in a map in presence of convex polygonal obstacles. The points can be considered as moving robots among which one is considered as source. The goal is to conclude if this source robot can communicate with a fixed tower called destination point. The objective is not to find any shortest path of communication but rather a Boolean question that if the source is visible to destination ?

Chapter 2

Proposed Algorithm

Suppose there are n randomly moving points (including the source) in 2D plane and a fixed destination point along with some convex polygonal obstacles. Here the algorithm below detects whether the source is visible to the destination directly or indirectly.

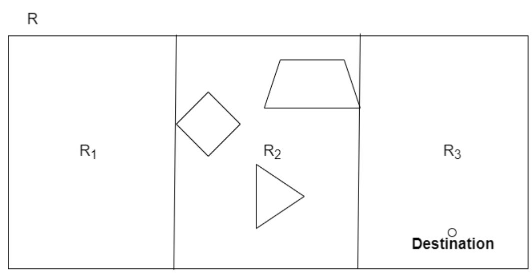

Figure 2.1: The map

The step by step flow of the algorithm is as follows:

STEP 1: All the convex polygonal obstacles and the points are enclosed in a bounding rectangle which divides the map in three sub rectangles R_1, R_2 and R_3.

In each timestamp the remaining step are followed.

STEP 2: If the source is in R_3 it is readily visible.

STEP 3: If the source is in R_1 or R_2 check visibility with R_3.

STEP 4: If STEP 3 fails check visibility of the source through R_1.

STEP 5: If STEP 4 fails check visibility of the source through R_2.

STEP 6: If STEP 5 fails check visibility of the source through R_1 and R_2.

STEP 7: If STEP 6 fails check visibility of the source through R_2 and R_1.

STEP 8: If STEP 7 fails there is no visibility between the source and destination.

STEP 9: Move all the n points randomly and follow STEP 2 to STEP 8 again.

The description of the above methodology for the proposed algorithm is as follows:-

All the convex polygons are enclosed in a bounding rectangle R. The bounding rectangle is constructed as follows:- Let x_{min} and x_{max} are the minimum and maximum x-coordinates and y_{min} and y_{max} are the minimum and maximum y-coordinates of the convex polygons. Then the height of the bounding rectangle will be $|(y_{max} + \epsilon) - (y_{min} - \epsilon)|$ where $\epsilon > 0$ and the length will be $|x_{min} - \delta| + \delta + |x_{max} + \delta|$ where $\delta = |x_{max} - x_{min}|$.

The length is chosen like this to make the area of all the three sub-rectangles equal to each other. Now as it can seem in below picture there are three sub-rectangles which are referred to as regions. Let the left most region be R_1 the middle one be R_2 and the right most be R_3.

All the concerned points are initially inside the bounding rectangle R and they move randomly in each timestamp inside R. So, without loss of generality let the fixed destination D is in the region R_3. And for all other points are separated according to their region. Any point (x, y) is in R_1 if $x < x_{min}$, in R_2 if $x_{min} \leq x \leq x_{max}$ and in R_3 if $x > x_{max}$.

Orientation of three points $P(x_1, y_1)$, $Q(x_2, y_2)$ and $X(x_3, y_3)$ is denoted here by (P, Q, X). Orientation can be of three types counterclockwise, clockwise and collinear as shown below.

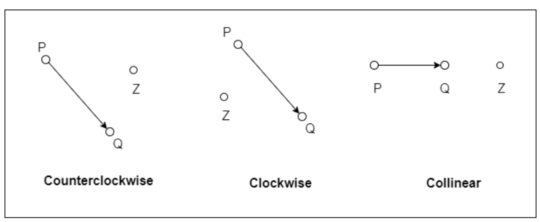

Figure 2.2: Orientation

Any three points always have any one these three orientation which is verified by the following quantity $\tau = (y_2 - y_1)(x_3 - x_2) - (y_3 - y_2)(x_2 - x_1)$. If $\tau = 0$ then (P, Q, X) are collinear, if $\tau > 0$ then (P, Q, X) are clockwise and if $\tau < 0$ then (P, Q, X) are counterclockwise.

The visibility of two points is being concluded by checking intersection of the line segment through these points with all the edges of all polygons. There is visibility if the line segment do not intersect with any edges of the all the polygons or is collinear with any edge. Intersection checking is done by verifying orientation of the points. Let $P(x_1, y_1)$ and $Q(x_2, y_2)$ are the points whose visibility is to be determined. Let $A(x_k, y_k)$ and $B(x_p, y_p)$ makes an edge of any polygon. Now, the line segment PQ intersects with AB if and only if orientation of (P, Q, A) and (P, Q, B) are different and orientation of (A, B, P) and (A, B, Q) are also different i.e, $\tau_{PQA} \neq \tau_{PQB}$ and $\tau_{ABP} \neq \tau_{ABQ}$. In this case there is no visibility between P and Q and in all other cases either PQ and AB do not intersect or they are collinear. Hence P and Q are visible in all other cases.

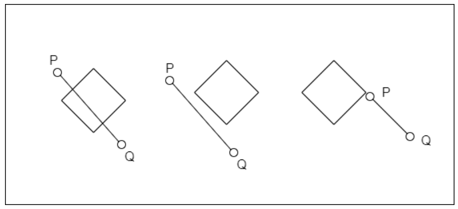

Figure 2.3: Visibility

Clearly, in the above figure the points P and Q are not visible in the first case and are visible in the other two cases.

If the source is R_3 then the source are destination are readily visible.

If the source is in R_1 or R_2 at first the visibility is checked directly with R_3. Then through R_1 or R_2, R_1 and R_2 and R_2 and R_1.

Let the source lie in R_1. Then the following checks are followed for visibility detection between source and destination.

1. First check the visibility of the source with each points in R_3 and if it is visible with at least one point in R_3 then the source and destination are visible else proceed to next step.

2. The visibility of source and destination through R_1 is now checked. Since the source is in R_1 the source is visible with each point in R_1 and let there are α number of points in R_1. Now the visibility of these α points with each point in R_3 is checked. If any one of the α point is visible with at least one point in R_3 then the source and destination are visible else proceed to next step.

3. The visibility of source and destination through R_2 is now checked. If the source is not visible to any point in R_2 proceed to next step else let the source is visible with β number of points in R_2. Now the visibility of these β points with each point in R_3 is checked. If any one of the β points is visible with at least one point in R_3 then the source and destination are visible else the visibility of these β points with other points of R_2 are checked. If they are not visible with any other point in R_2 proceed to next step else let these β points are visible to β_1 points in R_2. Then the visibility of these β_1 points with all points in R_3 is checked. If any one of these β_1 points is visibility to at least one point in R_3 then the source and destination are visible else proceed to next step.

4. The visibility of source and destination through R_1 and R_2 is now checked. If $\alpha = 1$ i.e, only source is there in R_1 then this step is skipped and proceed to the next one. If $\alpha > 1$ then the visibility of these α points with all points in R_2 is checked. If there is no visibility proceed to next step else let these α points are visible with β_2 points in R_2. Now the visibility of these β_2

points with all points in R_3 is checked. If any one of these β_2 points is visible with at least one point in R_3 then the source and destination are visible else the visibility of these β_2 points are checked with other points in R_2. If there is no visibility then proceed to next step else let these β_2 points are visible with β_3 points of R_2. Then the visibility of these β_3 points with all points in R_3 is checked. If any one of these β_3 points is visibility to at least one point in R_3 then the source and destination are visible else proceed to next step.

5. The visibility of source and destination through R_2 and R_1 is now checked. If $\beta = 0$ it is finally concluded that the source and destination are not visible. If $\beta \neq 0$ then the visibility of these β points with all points in R_1 is checked. If there is no visibility it is finally concluded that the source and destination are not visible else let these β points are visible with at least one point in R_1. Now all points are visible with all other points in R_1. So, the visibility of all points of R_1 with all points of R_3 is checked. If any one point of R_1 is visible with at least one point of R_3 then the source and destination are visible else it is finally concluded that the source and destination are not visible.

Let the source is in R_2. Then the following checks are followed for visibility detection between source and destination.

(i) First check visibility of the source with each points in R_3 and if it is visible with at least one point in R_3 then the source and destination are visible else proceed to next step.

(ii) The visibility of source and destination through R_2 is now checked. If the source is not visible to any point in R_2 proceed to next step else let the source is visible with β number of points in R_2. Now the visibility of these β points with each point in R_3 is checked. If any one of the β point is visible with at least one point in R_3 then the source and destination are visible else the visibility of these β points with other points of R_2 are checked. If they are not visible with any other point in R_2 proceed to next step else let these β points are visible to β_1 points in R_2. Then the visibility of these β_1 points with all points in R_3 is checked. If any one of these β_1 points is visibility to at least one point in R_3 then the source and destination are visible else proceed to next step.

(iii) The visibility of source and destination through R_1 is now checked. If the source is not visible to any point in R_1 proceed to next step else let the source is visible with α number of points in R_1. Now since all points in R_1 are visible to all other points in R_1 the visibility of all points in R_1 is checked with each point in R_3. If any one of the point in R_1 is visible with at least one point in R_3 then the source and destination are visible else proceed to next step.

(iv) The visibility of source and destination through R_1 and R_2 is now checked. If $\alpha = 0$ this step is skipped and proceed to the next one. If $\alpha \neq 0$ then the visibility of these α points with all points in R_2 is checked. If there is no visibility proceed to next step else let α points are visible with β_2 points in R_2. Now the visibility of these β_2 points with all points in R_3 is checked and if any one of these β_2 points is visible with at least one point in R_3 then the source and destination are visible else the visibility of these β_2 points are checked with other points in R_2. If there is no visibility then proceed to next step else let these β_2 points are visible with β_3 points of R_2. Then the visibility of these β_3 points with all points in R_3 is checked. If any one of these β_3 points is visibility to at least one point in R_3 then the source and destination are visible else proceed to next step.

(v) The visibility of source and destination through R_2 and R_1 is now checked. If $\beta = 0$ it is finally concluded that the source and destination are not visible. If $\beta \neq 0$ then the visibility of these β points with all points in R_1 is checked. If there is no visibility it is finally concluded that the source and destination are not visible else let β points are visible with at least one point in R_1. Now all points are visible with all other points in R_1. So, the visibility of all points of R_1 with all points of R_3 is checked. If any one point of R_1 is visible with at least one point of R_3 then the source and destination are visible else it is finally concluded that the source and destination are not visible.

Probabilistic analysis of the Algorithm: The orientation checking and intersection checking (as mentioned in Section 3) between two line segment runs in $O(1)$ time. But to check visibility of any two points intersection checking with all the edges of all the polygons are conducted and so visibility checking between any two points takes $O(q)$ time. So, checking visibility of one points with n other points runs in $O(qn)$ which is asymptotically $O(n)$ as $q << n$. So, in this whole discussion visibility checking will involve a constant factor of q which is ignored as q is almost constant and $O(q)$, $O(qn)$ and $O(qn^2)$ are asymptotically $O(1)$, $O(n)$ and $O(n^2)$ respectively.

This analysis aims to find how much of time the algorithm runs in $O(n^2)$ in a long run with some Assumptions which are as follows:

(a) In any timestamp no points is outside the bounding rectangle.

(b) The total number of vertices (say k) of all the polygonal obstacles are very less compared to the number of points, i.e, $k << n$.

(c) The total number of edges (say q) all the polygons combined is also very less as k is less. Hence, $q << n$.

(d) The chance of source appearing in R_1, R_2 and R_3 is equally likely.

(e) The chances of occurrence of the three events (mentioned below) is equally likely.

(i) All three regions R_1, R_2 and R_3 contains points proportional to n

(ii) Any two regions contains points proportional to n and the other one contains constant (compared to n) number of points.

(iii) Any one region contains points proportional to n and other contain constant (compared to n) number of points.

Let A, B and C denotes the event that the source lies in the region R_1, R_2 and R_3 respectively. Since, A, B and C are equally likely. (Assumption (d))

$$\therefore P(A) = P(B) = P(C) = \frac{1}{3}$$

Let Y be the event that the algorithm runs in $O(n^2)$. Then Y can occur in one of the three mutually exclusive ways $A \cap Y$, $B \cap Y$ and $C \cap Y$. Since, these are mutually exclusive it can be written,

$$\begin{aligned} P(Y) &= P(A \cap Y) + P(B \cap Y) + P(C \cap Y) \\ &= P(A)P(Y|A) + P(B)P(Y|B) + P(C)P(Y|C) \\ &= \frac{1}{3}P(Y|A) + \frac{1}{3}P(Y|B) + \frac{1}{3}P(Y|C) \\ &= \frac{1}{3}[P(Y|A) + P(Y|B) + P(Y|C)] \end{aligned} \tag{2.1}$$

Finding the conditional probabilities $P(Y|A)$, $P(Y|B)$ and $P(Y|C)$ which denotes the probability that the algorithm runs in $O(n^2)$ where it is already known that the source is in R_1, R_2 and R_3 respectively.

1. Calculating $P(Y|C)$: If the source is R_3 in destination which can be verified by the check $x_{source} > x_{max}$ and if so it is concluded that the source and destination are visible. This is $O(1)$ operation.

$$\therefore P(Y|C) = 0$$

2. Calculating $P(Y|A)$: If the source is in R_1 then to conclude visibility the following steps are followed,

   (A) First check visibility of the source with each points in R_3 and if it is visible with at least one point in R_3 then the source and destination are visible else proceed to next step. This operation runs in $O(n)$ in worst case.

   (B) The visibility of source and destination through R_1 is now checked. Since the source is in R_1 the source is visible with each point in R_1 and let there are α number of points in R_1. Let there are γ points in R_3. Now the visibility of these α points with each point in R_3 is checked. If any one of the α point is visible with at least one point in R_3 then the source and destination are visible else proceed to next step. Now three situations arises here.

i. If both α and γ are proportional to n then the algorithm runs in $O(n^2)$.
ii. If one of α or γ is proportional to n and the other is constant the algorithm runs in $O(n)$.
iii. Both α and γ are constant then the algorithm runs in $O(1)$.

(C) The visibility of source and destination through R_2 is now checked. If the source is not visible with any point in R_2 proceed to next step else let the source is visible with β number of points in R_2. Let there are γ points in R_3. Now the visibility of these β points with each point in R_3 is checked. If any one of the β point is visible with at least one point in R_3 then the source and destination are visible else the visibility of these β points with other points of R_2 are checked. If they are not visible with any other point in R_2 proceed to next step else let these β points are visible to β_1 points in R_2. Then the visibility of these β_1 points with all points in R_3 is checked. If any one of these β_1 points is visibility to at least one point in R_3 then the source and destination are visible else proceed to next step. Now six situations arises here.

i. If β_1 is not there and both β and γ are proportional to n then the algorithm runs in $O(n^2)$.
ii. If β_1 is not there and one of β or γ is proportional to n and the other is constant the algorithm runs in $O(n)$.
iii. If β_1 is not there and both β and γ are constant then the algorithm runs in $O(1)$.
iv. If β_1 is proportional to n and any one or both of β and γ are proportional to n the algorithm runs in $O(n^2)$.
v. If β_1 is constant and any one or both of β and γ are proportional to n the algorithm runs in $O(n)$.
vi. If all β, β_1 and γ are constant then the algorithm runs in $O(1)$.

(D) The visibility of source and destination through R_1 and R_2 is now checked. If $\alpha = 1$ i.e, there is only the source in R_1 then this step is skipped and proceed to the next one. If $\alpha > 1$ then the visibility of these α points with all points in R_2 is checked. If there is no visibility proceed to next step else let these α points are visible with β_2 points in R_2. Now the visibility of these β_2 points with all γ points in R_3 is checked. If any one of these β_2 points is visible with at least one point in R_3 then the source and destination are visible else the visibility of these β_2 points are checked with other points in R_2. If there is no visibility then proceed to next step else let these β_2 points are visible with β_3 other points of R_2. Then the visibility of these β_3 points with all points in R_3 is checked. If any one of these β_3 points is visibility to at least one point in R_3 then the source and destination are visible else proceed to next step. Now seven situations arises here.

i. If β_3 is not there and β_2 is proportional to n and any one or both of α and γ are proportional to n the algorithm runs in $O(n^2)$.
ii. If β_3 is not there and β_2 is constant and any one or both of α and γ are proportional to n the algorithm runs in $O(n)$.
iii. If all α, β_1 and γ are constant then the algorithm runs in $O(1)$.
iv. If all four of α, β_2, β_3 and γ are proportional to n then the algorithm runs in $O(n^2)$.
v. If any three of α, β_2, β_3 and γ are proportional to n then also the algorithm runs in $O(n^2)$.
vi. If any one or both of β_2 and β_3 are constant then the algorithm runs in $O(n)$.
vii. If all four of α, β_2, β_3 and γ are constant then the algorithm runs in $O(1)$.

(E) The visibility of source and destination through R_2 and R_1 is now checked. If $\beta = 0$ it is finally concluded that the source and destination are not visible. If $\beta \neq 0$ then the visibility of these β points with all points in R_1 is checked. If there is no visibility it is finally concluded that the source and destination are not visible else let any one of these β points are visible with at least one point in R_1. Now all points are visible with all other points in R_1. So, the visibility of all α points of R_1 with all γ points in R_3 is checked. If any one point of R_1 is visible with at least one point of R_3 then the source and destination are visible else it is finally concluded that the source and destination are not visible . Now six situations arises here.

i. If α is proportional to n and any one or both of β and γ are proportional to n the algorithm runs in $O(n^2)$.
ii. If α is constant and any one or both of β and γ are proportional to n the algorithm runs in $O(n)$.
iii. If all α, β, and γ are constant then the algorithm runs in $O(1)$.

Clearly, to conclude the visibility of source with destination where it is known that source is in R_1 the algorithm runs all or some of the steps mentioned through in A to E. Now the algorithm runs in $O(n^2)$ if any one of the steps runs in $O(n^2)$ and in $O(n)$ if all of them runs in $O(n)$. So if the source is in R_1 there are only two possibilities either the algorithm runs in $O(n^2)$ or in $O(n)$. Since it is assumed that the events of distribution of points are equally likely (Assumption (e)),

$$\therefore P(Y|A) = \frac{1}{2}$$

3. Calculating $P(Y|B)$: If the source is in R_2 then to conclude visibility the following steps are followed,

(A) First check visibility of the source with each points in R_3 and if it is visible with at least one point in R_3 then the source and destination are visible else proceed to next step. This operation runs in $O(n)$ in worst case.

(B) The visibility of source and destination through R_1 is now checked. If the source is not visible with any point in R_1 proceed to next step else let the source is visible with α number of points in R_1. Let there are γ points in R_3. Now since all points in R_1 are visible to all other points in R_1 the visibility of all points in R_1 is checked with each point in R_3. If any one of the point in R_1 is visible with at least one point in R_3 then the source and destination are visible else proceed to next step.

    i. If both α and γ are proportional to n then the algorithm runs in $O(n^2)$.
    ii. If one of α or γ is proportional to n and the other is constant the algorithm runs in $O(n)$.
    iii. Both α and γ are constant then the algorithm runs in $O(1)$.

(C) The visibility of source and destination through R_2 is now checked. If the source is not visible with any point in R_2 proceed to next step else let the source is visible with β number of points in R_2. Now the visibility of these β points with each point in R_3 (γ) is checked. If any one of the β point is visible with at least one point in R_3 then the source and destination are visible else the visibility of these β points with other points of R_2 are checked. If they are not visible with any other point in R_2 proceed to next step else let these β points are visible to some β_1 points in R_2. Then the visibility of these β_1 points with all points in R_3 is checked. If any one of these β_1 points is visibility to at least one point in R_3 then the source and destination are visible else proceed to next step. Now six situations arises here.

    i. If β_1 is not there and both β and γ are proportional to n then the algorithm runs in $O(n^2)$.
    ii. If β_1 is not there and one of β or γ is proportional to n and the other is constant the algorithm runs in $O(n)$.
    iii. If β_1 is not there and both β and γ are constant then the algorithm runs in $O(1)$.
    iv. If β_1 is proportional to n and any one or both of β and γ are proportional to n the algorithm runs in $O(n^2)$.
    v. If β_1 is constant and any one or both of β and γ are proportional to n the algorithm runs in $O(n)$.
    vi. If all β, β_1 and γ are constant then the algorithm runs in $O(1)$.

(D) The visibility of source and destination through R_1 and R_2 is checked. If $\alpha = 0$ this step is skipped and proceed to the next one. If $\alpha \neq 0$ then the visibility of these α points with all points in R_2 is checked.

If there is no visibility proceed to next step else let these α points are visible with β_2 points in R_2. Now the visibility of these β_2 points with all points in R_3 (γ) is checked. If any one of these β_2 points is visible with at least one point in R_3 then the source and destination are visible else the visibility of these β_2 points are checked with other points in R_2. If there is no visibility then proceed to next step else let these β_2 points are visible with β_3 other points of R_2. Then the visibility of these β_3 points with all points in R_3 is checked. If any one of these β_3 points is visibility to at least one point in R_3 then the source and destination are visible else proceed to next step. Now seven situations arises here.

i. If β_3 is not there and β_2 is proportional to n and any one or both of α and γ are proportional to n the algorithm runs in $O(n^2)$.

ii. If β_3 is not there and β_2 is constant and any one or both of α and γ are proportional to n the algorithm runs in $O(n)$.

iii. If all α, β_1 and γ are constant then the algorithm runs in $O(1)$.

iv. If all four of α, β_2, β_3 and γ are proportional to n then the algorithm runs in $O(n^2)$.

v. If any three of α, β_2, β_3 and γ are proportional to n then also the algorithm runs in $O(n^2)$.

vi. If any one or both of β_2 and β_3 are constant then the algorithm runs in $O(n)$.

vii. If all four of α, β_2, β_3 and γ are constant then the algorithm runs in $O(1)$.

(E) The visibility of source and destination through R_2 and R_1 is now checked. If $\beta = 0$ it is finally concluded that there is no visibility. If $\beta \neq 0$ then the visibility of these β points with all points in R_1 is checked. If there is no visibility it is finally concluded that the source and destination is not visible else let these β points are visible with at least one point in R_1. Now all points are visible with all other points in R_1. Let there are α_1 points in R_1. So, the visibility of all points of R_1 with all points in R_3 is checked. If any one point of R_1 is visible with at least one point of R_3 then the source and destination are visible else it is finally concluded that the source and destination are not visible. Now six situations arises here.

i. If α_1 is proportional to n and any one or both of β and γ are proportional to n the algorithm runs in $O(n^2)$.

ii. If α_1 is constant and any one or both of β and γ are proportional to n the algorithm runs in $O(n)$.

iii. If all α_1, β, and γ are constant then the algorithm runs in $O(1)$.

Clearly, to conclude the visibility of source with destination where it is known that source is in R_2 the algorithm runs all or some of the steps mentioned through in A to E. Now the algorithm runs in $O(n^2)$ if any one of the steps runs in $O(n^2)$ and in $O(n)$ if all of them runs in $O(n)$. So if the source is in R_1 there are only two possibilities either the algorithm runs in $O(n^2)$ or in $O(n)$. Since it is assumed that the events of distribution of points are equally likely (Assumption (e)),

$$\therefore P(Y|B) = \frac{1}{2}$$

Hence, Equation 1 yields,

$$\begin{aligned} P(Y) &= \frac{1}{3}[\frac{1}{2} + \frac{1}{2} + 0] \\ &= \frac{1}{3} \times 1 \\ &= \frac{1}{3} \end{aligned}$$

NOTE: All the calculations above are done assuming that the destination is in R_3. But the result remains same irrespective of the position of the destination. If destination is in R_1 then $P(Y|A) = 0$ and $P(Y|B) = P(Y|C) = \frac{1}{2}$ and if the destination is in R_2 then $P(Y|B) = 0$ and $P(Y|A) = P(Y|C) = \frac{1}{2}$.

Hence in one trial of the experiment (referred to the running of the algorithm) the probability that the algorithm runs in $O(n^2)$ is $\frac{1}{3}$.

Now let X be the random variable denoting that the algorithm runs in $O(n^2)$. Clearly, the algorithm either runs in $O(n)$ or $O(n^2)$. So, X follows Binomial Distribution with probability of success $p = \frac{1}{3}$ and probability of failure $1 - p = \frac{2}{3}$.

Hence, the probability mass function of X is given by,

$$P(X = k) = \binom{x}{k} p^k (1-p)^{x-k}$$

which denotes the probability of getting exactly k successes in x independent trial of the experiment.

Now, the expectation of X is given by,

$$\begin{aligned}
E(X) &= \sum_{k=0}^{x} k \binom{x}{k} p^k (1-p)^{x-k} \\
&= \sum_{k=1}^{x} k \frac{x!}{k!(x-k)!} p^k (1-p)^{x-k} \\
&= \sum_{k=1}^{x} \frac{x!}{(k-1)!(x-k)!} p^k (1-p)^{x-k} \\
&= xp \sum_{k=1}^{x} \frac{(x-1)!}{(k-1)!(x-k)!} p^{k-1} (1-p)^{x-k} \\
&= xp \sum_{y=0}^{x-1} \frac{(x-1)!}{y!(x-1-y)!} p^{y} (1-p)^{x-1-y} \text{ , } where\ y = k-1 \\
&= xp(p+1-p)^{x-1} \\
&= xp \times (1)^{x-1} \\
&= xp \qquad (2.2)
\end{aligned}$$

Here, $p = \frac{1}{3}$. So, Equation 2 yields,

$$E(X) = \frac{x}{3}$$

So, if the experiment is conducted x number of time the probability of getting success (the probability that the algorithm runs in $O(n^2)$) is $\frac{1}{3}$ which is roughly equal to 33.33% of x. Hence the algorithm runs in $O(n^2)$ in 33.33% of the cases and runs in $O(n)$ in 66.67% of the cases in long run.

Chapter 3

Conclusion

The work involves detecting visibility between a moving source and a fixed destination in presence of some moving points and static polygonal obstacles. Since the objective is to only determine visibility and no to find shortest path hence visibility graph is not constructed which takes $O(n^2)$ extra space. Since the points are randomly moving this algorithm exploit this fact and expectedly runs in $O(n)$ in long run. Further study can be conducted for finding the shortest path on the same problem. It can also be extended to 3D and running time analysis could be done. The work also opens the path of not using visibility graphs where only visibility is concerned. Lastly other computational geometry problems like closest pair, convex hull etc. of moving points can also be analysed in this way.

Bibliography

[1] Michael Batty. Exploring isovist fields: space and shape in architectural and urban morphology. Environment and planning B: Planning and Design, 28(1):123–150, 2001.

[2] Michael L Benedikt. To take hold of space: isovists and isovist fields. Environment and Planning B: Planning and design, 6(1):47–65, 1979.

[3] Yoon Kyung Choi. The morphology of exploration and encounter in museum layouts. Environment and Planning B: Planning and Design, 26(2):241–250, 1999.

[4] Dafna Fisher-Gewirtzman, Michael Burt, and Yigal Tzamir. A 3-d visual method for comparative evaluation of dense built-up environments. Environment and Planning B: Planning and Design, 30(4):575–587, 2003.

[5] JJ Gibson. The ecological approach to visual perception. mahwah, nj: L, 1979.

[6] B Hillier and J Hanson. The social logic of space. cambridge: Press syndicate of the university of cambridge, 1984.

[7] Bill Hillier and Kali Tzortzi. Space syntax: the language of museum space. A companion to museum studies, pages 282–301, 2006.

[8] Christoph Hölscher, Martin Brösamle, and Georg Vrachliotis. Challenges in multilevel wayfinding: A case study with the space syntax technique. Environment and Planning B: Planning and Design, 39(1):63–82, 2012.

[9] Anastasia Koltsova, Bige Tunçer, and Gerhard Schmitt. Visibility analysis for 3d urban environments. 2013.

[10] Yi Lu and John Peponis. Exhibition visitors are sensitive to patterns of display covisibility. Environment and Planning B: Planning and Design, 41(1):53–68, 2014.

[11] Sophia Psarra. Spatial culture, way-finding and the educational message. The Impact of Layout on the Spatial, Social and Educational Experiences of Visitors to Museums and Galleries In: McLeod, S ed. Reshaping

Museum Space: Architecture, Design, Exhibitions. London and New York: Routledge, pages 78–94, 2005.

[12] Sophia Psarra. Architecture and Narrative: The formation of space and cultural meaning. Routledge, 2009.

[13] Alasdair Turner, Maria Doxa, David O'sullivan, and Alan Penn. From isovists to visibility graphs: a methodology for the analysis of architectural space. Environment and Planning B: Planning and design, 28(1):103–121, 2001.

[14] Duncan J Watts and Steven H Strogatz. Collective dynamics of 'small-world'networks. nature, 393(6684):440–442, 1998.

[15] Pegah Zamani and John Peponis. Co-visibility and pedagogy: innovation and challenge at the high museum of art. The Journal of Architecture, 15(6):853–879, 2010.