

# SHORTEST PATH FINDER IN REAL TIME APPLICATION

DILPREET KAUR, DR.P.S MUNDRA

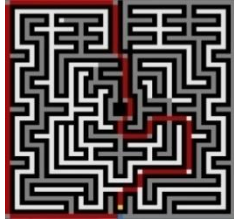
## I Abstract

Ant colony optimization (ACO) is a class of optimization algorithms modeled on the actions of an ant colony. ACO methods are useful in problems that need to find paths to goals. Artificial 'ants'—simulation agents—locate optimal solutions by moving through a parameter space representing all possible solutions. The basic idea in ant colony optimization algorithms (ACO) is to imitate the cooperative behavior of real ants to solve optimization problems. ACO metaheuristics have been proposed by M. Dorigo. They can be seen as multiagent systems in which each single agent is inspired by the behavior of a real ant. Traditionally, ACO have been applied to combinatorial optimization problems and they have achieved widespread success in solving different problems. The main interest of real ant's behavior is that simple ants using collective behavior perform complex tasks such as transportation of food and finding shortest paths to the food sources. ACO algorithms mimic the principle that using very simple communication mechanism, an ant colony is able to find the shortest path between two points. – The purpose of this paper is to design a robot controller which is able to pursue a given goal with obstacle-avoiding capability in which the two tasks, i.e. aiming at the goal and avoiding obstacles, solving the problem is performed. For a given ROBOT, the path is chosen according to the obstacles coming the way. When facing an obstacle, there is an equal probability for every robot to choose the left or right path.

**Keywords:** ACO (Ant Colony Optimization), GIS (geographical information system), Image compression.

## II Introduction

Graphs can be imagined as a set of points (vertices) interconnected by lines (edges). The edges can have specified direction (orientation). Paths are sequences of vertices and edges between them. Path-finding between two graph points is used in many areas, like GPS navigation, CAD systems (auto-routing in design of circuit boards), and applications of artificial intelligence, computer games, virtual reality, military forces or robotics. The main aim is to pass through environment in secure form and to avoid of obstacles. Examples of this need are autonomous robots on distant planets without the possibility to be controlled in real time because of latencies in signal sending, or automatic vehicles control. Another example of path-finding are strategic computer games, mostly with computer opponent. Path-finding is also widely used in finding best routers interconnection for data transmission in many kinds of computer networks. Some user-friendly visualization of results of path-finding, which is realized by application of computer graphics techniques. A geographical information system (GIS) is a software and geographical data designed for effective gathering, retaining, editing, processing, analysis and visualization of all kinds of geographical information. Generally, it is a spatial specialized IS. Being an IS, GIS has to provide information in graphical (e.g. location of the hotel on the map) and nongraphical (e.g. fees, room equipment) way that supports fast searching and actualization of data. Nowadays, graphical information are stored as vector graphics (two-dimensional), but the trends are clearly and forecasting the 2 generation GIS as an IS working with 3D objects and surfaces.

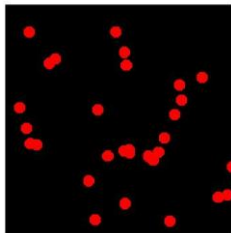


**Figure1:** Shows how a algorithm finds its path through different hurdles.

### III Methodology

To reach the path in minimum time and with accuracy at our destination we implement this optimization .In this we are completing our task through robot by taking a satellite image. IT works as follow:

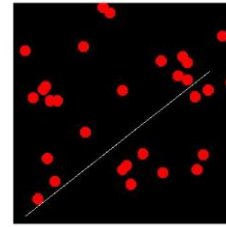
- A. In order to solve a problem at moon through robot we have firstly taken a satellite image of the path and the destination. In satellite image we have seen many obstacles coming on way (fire balls,) ,we have marked them as red balls.



**Figure2:** Satellite zoomed image taken from earth to show different paths to robot. Red balls are the possible hurdles coming on way to robot like fireballs, meteors, etc.

- B. Now these balls are the obstacles coming in the way of robot while going from earth to moon. So when we'll define the start up and destination points for it, and if we send the robot as it is to the path then it may strike with one of the obstacle before reaching the destination. This straight line

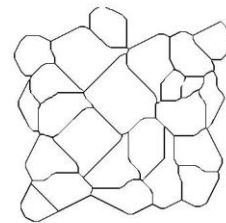
how the maximum obstacles coming on the



way.

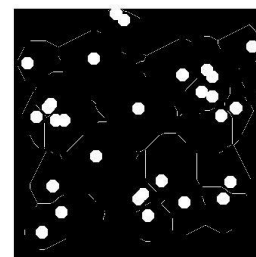
**Figure3:** Shows the straight path that if the robot is sent as it is it can strike with any of the ball and destroyed, so to avoid this situation we use aco.

- C. Now the satellite image is zoomed and optimization shows the possible routes for the robot to reach the destination safely. Image is first converted from rgb to gray scale and then to black and white, these lines are the possible routes through various fire balls, etc coming the way.



**Figure4:**Thinned lines shows the possible path for the going robot to reach on time and come back safely.

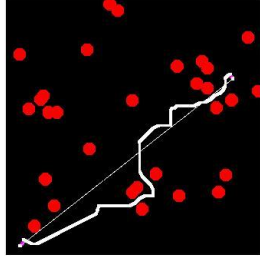
- D. Now the problem is the holes in between the routes through which fire balls can enter to strike the robot, so for that we have used kannan Kan method. Every obstacle is considered as an object, and it will be dilated and then eroded to fill the holes.



**Figure5:**After finding routes the dilated

and the eroded images are added to get the dummy image.

- E. Finally we can send our robot between the two defined points i.e from earth to moon to complete the task efficiently.



**Figure6:** Finally also starts from the startup point and to reach end point shortly Nearest Route Track Time is time taken to attach nearest startup point to actual startup point, Route Track Time is time taken to find the shortest path i.e. loop execution time and Nearest End Route Track Time is time taken to attach nearest end point to destination point.

#### IV Graphs

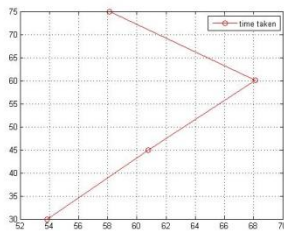
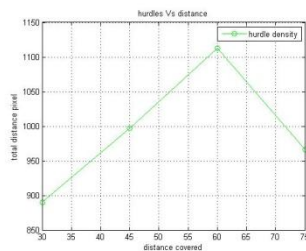


Figure7,8: Showing variations with number of hurdles and time taken and distance pixel depending on the startup and end points.



#### V Results

s. n	HURDLE DENSITY	NEAREST ROUTE TRACK TIME	ROUTE TRACK TIME	NEAREST END ROUTE TRACK TIME	TOTAL TIME	TOTAL DISTANCE PIXEL
1.	30	2.1372	157.1821	2.2284	53.84	890
2.	45	0.6399	181.3083	0.4228	60.79	997
3.	60	4.7850	194.7420	4.7850	68.10	1113
4.	75	3.8864	168.2956	2.1831	58.12	966

#### VI Conclusion

. Obstacle avoidance, however, is local in nature, and a higher priority temporarily should be placed on avoiding a collision with the obstacle than taking the shortest path toward the goal. This paper presents a planning algorithm and its application to traffic. The algorithm copes with the uncertainty of traffic conditions by stochastic modeling of travel delay on networks. The algorithm determines paths between two points that optimize a cost function of the delay data probability distribution. It can be used to find paths that maximize the probability of reaching a destination within a particular travel deadline. For such problems, standard shortest-path algorithms do not work because the optimal substructure property does not hold. Our algorithm can be integrated into on-board navigation systems as well as route-finding website.

#### VII References

1. Jerald Jariyasunant" Algorithm for finding optimal paths in a public transit network with real-time data" 2010
2. Anand Ranganathan, Nesime Tatbul" Real-time Route Planning with Stream Processing

- Systems: A Case Study for the City of Lucerne" 2011
3. Rahim A. Abbaspour, Farhad Samadzadegan" An Evolutionary Solution for Multimodal Shortest Path Problem in Metropolises" 2010
  4. FANG Yuan CHEN Jiemin," Realtime emergency route generating algorithm in tunnel" 2011
  5. Mohannad Abid Shehab Ahmed" OPTIMUM SHORT PATH FINDER FOR ROBOT USING QLEARNING" 2011
  6. Christian Plagemann, Varun Ganapathi" Real-time Identification and Localization of Body Parts from Depth Images".
  7. Shlomo Zilberstein, Stuart J. Russell" Anytime Sensing, Planning and Action: A Practical Model for Robot Control" 2010
  8. D. R. Wichmann, B. C. Wünsche" Automated Route Finding on Digital Terrains".