

Computer Graphics and Product modelling

ME 735

Submitted by: Group 9

213104014	Galla Rama Srinivas (Team Leader)
213101001	Deepak Kumar Thakur
213100053	Thirumalasetty Sravan Sai Kumar
213100061	Yogesh Shrikant Sale
213101002	Bijay Kumar Shah

Under the Guidance of

Prof. Dr. Sanjay S. Pande



Department of Mechanical Engineering
INDIAN INSTITUTE OF TECHNOLOGY BOMBAY

Powai, Mumbai 400076

02, November 2021

Abstract:

In this Project, we are defining a 2D layout with obstacles and defined pathways and we are working on finding an optimized path to cover the distance between the user defined starting and final destination points in a path where distance travelled is the minimum of all the possible paths between the two said points. The shortest (optimized) path is generated using Dijkstra's Algorithm with Python as the coding software.

Introduction:

Project Title:

Graphical representation of shortest travel path in the given layout between two locations.

Objective:

Generate a 2D based layout and find optimized path on the layout to move from one location to other in the shortest path and interpret the same graphically.

Deliverable:

1. Graphical work to replicate the layout (in 2D)/2D simulator-based layout generation.
2. Optimization algorithm to move from one point to other in shortest or best path.
3. Merging Graphical part with Optimization.

Programming Languages used:

Python (libraries used: NumPy, matplotlib)

Algorithm/Methodology:

1. Generate a fixed 2D layout image of size 20X20 pixels. Layout can be modified as needed to a size of N X N pixels.
2. Allocate a value of “0” to the obstacles and “1” to the pathways.
3. Identify nodes in the image of the layout.
Note: Nodes are the intersections of the pathways present in the layout.
4. Rearrange the nodes in ascending order and record the distances between the nodes in a L X L matrix (L=Number of nodes) as an adjacency list.
5. Convert adjacency list to Dictionary list for the implementation of Dijkstra’s algorithm.
6. Adjust the dictionary list to provide input to Dijkstra’s algorithm as a graph with vertices defined.
7. Dijkstra’s Algorithm – taken from Stack overflow.
8. Input start and end vertices for starting and ending points of the path respectively. We can also randomize the input for starting and end vertices.
9. Call Dijkstra’s algorithm to compute the optimized path and print out the node vertices along the optimized path.
10. Invert the Y-axis of the layout for plotting the optimized path using Matplotlib, as the layout has inverted Y axis and we would like to locate origin on bottom-left part of the layout.
11. Covert nodes to points on the layout for Graphical representation.
12. Plot an animated path from starting to end point on the layout.

Results and Conclusion:

1. Successfully generated a layout and the shortest path between 2 nodes in the layout using Dijkstra's algorithm.
2. Plotted the layout and optimized path graphically as the output of the Algorithm.

As declared in the objective, we were successfully able to Generate a 2D based layout and find optimized path on the layout to move from one location to other in the shortest path and interpret the same graphically.

References:

1. Dijkstra's Algorithm: <https://stackoverflow.com/questions/22897209/dijkstras-algorithm-in-python/61078380#61078380>
2. Dijkstra's Algorithm - Wikipedia: https://en.wikipedia.org/wiki/Dijkstra%27s_algorithm