SHORTEST PATH PROBLEM

Anuja Vane, Dhaval Chauhan, Saylee Bhide

Motivation:

Generating an optimal solution to any problem in any field is fundamental to our life. Minimizing the use of resources and time is of utmost importance! Finding the shortest path falls in the same line. When you are driving from City A to City B using Google maps, Google maps automatically routes you to the path that takes the least time to reach the destination. This routing is based on the problem of finding the shortest path. This problem is also used to compute the shortest path on the road networks of the USA and Europe. In the field of networking, the shortest path algorithm is used to find the minimum delay. The other applications of this problem include Robotics, VLSI, and transportation.

Idea:

Our project is based on the Euclidian shortest path problem in Computational Geometry. The problem space contains a set of Rectilinear obstacles in Euclidian space and a starting point and an ending point. The Euclidian shortest path algorithm should find a path between the two points such that no point in path intersects any of the obstacles. Building on this, our problem space will contain random Rectilinear obstacles constructed by the user or drawn by the user – thus allowing randomness in the problem state. Also, we will be showing the visualization of a rectangular box moving from the starting point to the ending point without intersecting any of the obstacles. To move through the search space, the rectangular box should adjust its orientation by rotating and translating, and checking if this new stance would allow it to pass the bottleneck created by obstacles. The challenge is that the Rectilinear obstacles will be user defined which makes versatility of the tool's algorithm an essential aspect.

Duties of each Team Member:

Since algorithm is the crux of a software tool, all three of us will put their collective minds in designing algorithms and debating the correctness of those algorithms. Similarly, implementation of the idea will be carried out simultaneously by all three of us using a modular developmental model. However, Anuja and Dhaval will work on implementation of the core algorithm while Saylee will work on the User Interface of the interactive tool. We all will be responsible for thorough testing and documentation of the modules we develop.

Project Phases and General Timeline:

- Week 5 to Week 8 – Algorithm design and analysis.
- Week 7 to Week 8 – UI design and implementation.
- Week 9 and Week 10 – Implementation of the algorithm.
- Week 11 – Integration and Testing.
- Week 12 – Bug fixing, Regression testing and final documentation.

References and guides:

1. Deriving an Obstacle-Avoiding Shortest Path in Continuous Space: A Spatial Analytic Approach - Insu Hong.
2. Computing Shortest Paths among Curved Obstacles in the Plane - Danny Z. Chen, Haitao Wang.
3. Euclidean Shortest Paths in the Presence of Rectilinear Barriers – D. T. Lee, F. T. Preparata.